UNIVERSITY OF PANNONIA

Veszprém, Hungary

New Methods and Algorithms for Testing Web Accessibility

by

Jinat Ara

DOI:10.18136/PE.2025.949

Supervisor: Prof. Cecilia Sik-Lanyi (PhD, DSc)

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the
Doctoral School of Information Science and Technology
Department of Electrical Engineering and Information Systems

27th May 2025

NEW METHODS AND ALGORITHMS FOR TESTING WEB ACCESSIBILITY

The thesis was prepared for the award of a doctoral degree (PhD) within the framework of the Doctoral School of Information Science and Technology at University of Pannonia

in the discipline of Computer Science

written by: **JINAT ARA**

Supervisor: CECILIA SIK-LANYI

I recommend the dissertation for acceptance: yes / no.

……………………….
supervisor

I recommend the dissertation for peer review.

……………………….
chair of the DDHC

The PhD-candidate has achieved .……. % at the public debate.

The composition of the Final Examination Committee:

chair:……….…………………………………………………………………….

reviewers:.……………………………………………………………………

members:……………………………………………………………………

Veszprém,

……………………….
chair of the committee

Qualification of degree: …...............................

Veszprém,

……………………….
chair of the UDHC

Declaration of Authorship

I, JINAT ARA, declare that this thesis titled, 'NEW METHODS AND ALGORITHMS FOR TESTING WEB ACCESSIBILITY' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ______________________________

Date: ______________________________

Abstract

Since the start of the World Wide Web in 1996, web accessibility evaluation has been an important aspect of web development to increase social inclusion for people with special needs. In the last few years, a significant effort has been observed to provide a useful web accessibility evaluation approach (hybrid and automated) to reduce the inconsistency of identifying accessibility issues for end-users. Although the developed approaches are significant, a few issues with webpage accessibility cannot be identified through the existing accessibility testing approaches due to several challenges and limitations. In this situation, it must identify which aspects are important to incorporate in the proposed solutions in order to make the developed solutions more effective, as it allows users to make their webpage accessible to people with disabilities.

Therefore, several existing studies from the state-of-the-art literature have been evaluated in this thesis, addressing the first research question **R-Q1:** What are the challenges and drawbacks of the existing web accessibility testing process? The evaluation of existing literature represents the frequent issues observed in their coverage, explainability, and reporting including (i) inappropriate guideline selection, (ii) ambiguities in guideline understanding, (iii) avoiding user and expert suggestions as evaluation criteria, (iv) limited consideration of semantic perspectives, and (v) unwillingness to incorporate the updated engineering methods which reduce the effectiveness of the developed approaches. Also, such issues may represent the result in an unclear manner, from which users might be confused, and that could directly hamper the possible adoption of the particular approach. Therefore, following the existing limitations identified with R-Q1, some straightforward yet precise solutions have been presented in this thesis to facilitate the web accessibility evaluation process from both hybrid and automated perspectives.

Regarding the hybrid evaluation, three different approaches have been proposed addressing the second research question **R-Q2:** How can we improve the effectiveness of the hybrid web content accessibility testing process to address its current limitations? Firstly, an integrated approach is presented considering several automated evaluation tools and human evaluation. Four automatic web accessibility evaluation tools (Mauve++, Nibbler, WAVE, and WEB accessibility tools) and human observation, such as system usability testing and expert testing, have been considered to evaluate the webpages. Secondly, a variable magnitude approach is presented to compute the accessibility score of webpages, considering the output of automated solutions and the requirements of people with disabilities. Two different components, such as the output of Mauve++ and Taw evaluation tools, have been considered as input variables and integrated to compute the overall accessibility score, considering the weight coefficient technique. Thirdly, a machine learning (ML)-based approach is presented that considers two ML classifiers: Random Forest (RF) and Decision Tree (DT) to evaluate the accessibility of the webpage. This ML-based approach has been experimented on a custom dataset to compute the overall accessibility score. The performance of ML methods was validated through the confusion matrix and classification report results.

The proposed three hybrid approaches were experimented on several webpages from different domains. Besides, expert evaluation has been incorporated to evaluate the experimental webpages to justify the evaluated results of the proposed approaches. The experiment results and expert evaluation validated the ability of the proposed approaches to provide an effective accessibility evaluation result.

Regarding the automated evaluation, at first, an automated accessibility evaluation framework has been

proposed addressing the third research question **R-Q3:** How can we improve the effectiveness of the automated web accessibility evaluation process to address the limitations of the available automated web accessibility testing tools? The proposed framework considers several accessibility aspects to improve the evaluation results such as (i) simplification of the updated web content accessibility guidelines, (ii) incorporation of all success criteria in the evaluation process, (iii) incorporation of user requirements/opinions with expert suggestions as an additional evaluation criterion, (iv) incorporation of separate complexity analysis algorithms for textual and non-textual feature analysis focusing semantic aspects, (v) categorization of the evaluated guidelines in terms of user evaluation and expert evaluation and (vi) representation of the evaluation result with the overall accessibility score along with specific accessibility scores for each disability type. Second, following the proposed framework, an automated web content accessibility testing tool has been developed addressing the fourth research question **R-Q4:** How can we increase the effectiveness of an automated web content accessibility assessment tool focusing on advanced engineering techniques? The developed automated web content accessibility evaluation tool assesses webpages by taking into account common features of the structural and visual elements of webpages that are part of the HTML Document Object Model (DOM) structure. In order to predict a webpage's accessibility status, three distinct algorithms are implemented to analyze web features/objects, considering both semantic and non-semantic aspect,s focusing on different NLP techniques and auxiliary methods.

Furthermore, to validate the effectiveness of the proposed framework and the developed automated tool, a comparative evaluation and experimental work have been conducted addressing the fifth research question **R-Q5:** How can the progress made with the developed tool be verified in terms of its effectiveness aspects? First, to do the comparative evaluation, the proposed framework and the developed tool were validated by comparing them with existing automated solutions, considering several functional properties. The comparative evaluation determines some distinct features that represent its significance and novelty. Second, to do the experimental work, 20 university webpages in Hungary were evaluated through the developed automated tool. The evaluated result of the developed tool was further assessed by comparing the result with a user study. The evaluation results of the developed automated tool and the user study depict that the proposed tool has the ability to compute the evaluation result that aligns with the participant's perception. This two-phase evaluation result shows that the developed tool has several advanced properties and potential to predict the accessibility issues of the tested webpages.

The presented approaches, frameworks, and the developed automated tool show potential to efficiently support web developers, accessibility engineers, and other practitioners in incorporating accessibility into their development and research. Although the contribution of the proposed approaches is significant, it is not free from limitations. The limitations were identified as experimentation was performed on small sets of webpages to validate the proposed approaches, thus, further studies are suggested to consider a large number of webpages from different domains and invite more experts to evaluate the result, considering the same set of data and parameters. With this, the approach presented will hopefully contribute to the socio-political objective of enhancing awareness of web accessibility to foster the inclusion of people with disabilities.

Tartalmi összefoglaló

A világháló 1996-os indulása óta a weblapok akadálymentességi értékelése a webfejlesztés fontos szem pontja, hogy növelje a társadalmi befogadást a speciális igényű emberek számára. Az elmúlt néhány évben jelentős erőfeszítések voltak megfigyelhetők egy jól használható webes akadálymentesség értékelési megközelítés (hibrid és automatizált) kidolgozására, amely csökkenti a végfelhasználók számára az akadálymentességi problémák azonosításának következetlenségét. Bár a létrehozott megközelítések jelentősek, a weboldalak akadálymentességével kapcsolatos néhány probléma számos kihívás és korlátozás miatt nem állapítható meg a meglévő akadálymentességi tesztelési szemlélettel. Ebben a helyzetben meg kell határozni, hogy mely szempontokat fontos beépíteni a javasolt megoldásokba annak érdekében, hogy a fejlesztett megoldások hatékonyabbak legyenek, mivel lehetővé teszi a felhasználók számára, hogy weblapjukat a fogyatékkal élő emberek számára akadálymentessé tegyék.

Disszertációmban a legfrissebb szakirodalomból számos fellelhető kutatást értékeltem, amelyek felvetették az első kutatási kérdést. **R-Q1:** Melyek a meglévő webes akadálymentesítési tesztelési folyamat kihívásai és hátrányai? A létező szakirodalom értékelése bemutatja a terjedelmükben, magyarázhatóságukban és jelentésükben megfigyelt gyakori problémákat, beleértve (i) az irányelvek nem megfelelő kiválasztását, (ii) az irányelvek értelmezésének kétértelműségét, (iii) a felhasználói és szakértői javaslatok, mint értékelési kritériumok mellőzését, (iv) a szemantikai szempontok korlátozott figyelembevételét, és (v) a vonakodást a korszerűsített mérnöki módszerek beépítésében, ami csökkenti a kidolgozott megközelítések hatékonyságát. Emellett az ilyen problémák az eredményt nem egyértelmű módon jeleníthetik meg, ami a felhasználókat összezavarhatja, és ami közvetlenül akadályozhatja az adott megközelítés esetleges elfogadását. Ezért az R-Q1 esetében azonosított meglévő korlátok után ebben a disszertációban néhány egyszerű, de pontos megoldást mutatok be a webes akadálymentességi értékelési folyamat megkönnyítése érdekében, mind a hibrid, mind az automatizált tesztelés szempontjából.

A hibrid értékeléssel kapcsolatban három különböző megközelítést javasoltam, amelyek a második kutatási kérdéshez vezettek. **R-Q2:** Hogyan javíthatjuk a hibrid webes akadálymentességi tesztelés folyamatán azok hatékonyságát, hogyan javítsuk a jelenlegi korlátaikat? Először is, egy integrált megközelítést mutatok be, amely több automatizált értékelő eszközt és humán értékelést vesz figyelembe. A weboldalak értékeléséhez négy automatikus webes akadálymentesítési értékelő eszközt (Mauve++, Nibbler, WAVE és WEB akadálymentesítési eszközt) és humán megfigyelést, például rendszerhasználhatósági tesztelést és szakértői tesztelést vettem figyelembe. Másodszor, egy változó nagyságrendű megközelítést mutatok be a weboldalak akadálymentességi pontszámának kiszámítására, figyelembe véve az automatikus megoldások kimenetét és a fogyatékkal élők igényeit. Két különböző automatikus tesztelőt, a Mauve++ és a Taw automatikus tesztelők által adott értékeket tekintettem bemeneti változóknak, és ezeket integráltam a teljes akadálymentességi pontszám kiszámításához, figyelembe véve a súly koefficiens technikát. Harmadszor, egy gépi tanuláson (machine learning, ML) alapuló megközelítés kerül bemutatásra, amely két ML osztályozót vett figyelembe: A weboldalak elérhetőségének értékeléséhez a véletlen erdő (Random Forest RF) és a döntési fa (Decision Tree DT) osztályozókat vettem figyelembe. Ez az ML alapú megközelítést egy egyedi adathalmazon kísérleteztem a teljes akadálymentességi pontszám kiszámításához. Az ML módszerek teljesítményét az igazságmátrix és az osztályozási jelentés eredményén keresztül validáltam.

A javasolt három hibrid megközelítést különböző területről származó weboldalakon kísérleteztem ki. Emellett szakértői értékelést is alkalmaztam a kikísérletezett weboldalak értékeléséhez, hogy igazolni

tudjam a javasolt megközelítések értékelt eredményeit. Kísérleti eredmények és a szakértői értékelés igazolta, hogy a javasolt megközelítések képesek hatékony akadálymentességi értékelési eredményt nyújtani.

Az automatizált értékelés tekintetében először egy automatizált akadálymentesítési értékelési keretrendszerre tettem javaslatot, amely a harmadik kutatási kérdést vetette fel **R-Q3:** Hogyan javíthatjuk az automatizált webes akadálymentesítési értékelési folyamat hatékonyságát, hogy javítsuk a rendelkezésre álló automatizált webes akadálymentesítési tesztelési eszközök korlátait? A javasolt keretrendszer több akadálymentességi szempontot is figyelembe vesz az értékelési eredmények javítása érdekében, mint például (i) az aktualizált Web Akadálymentességi Útmutató irányelveinek egyszerűsítése, (ii) az összes sikerkritérium beépítése az értékelési folyamatba, (iii) a felhasználói követelmények/vélemények beépítése a szakértői javaslatokkal együtt, mint további értékelési kritérium, (iv) külön komplexitáselemző algoritmusok beépítése a szöveges és nem szöveges jellemzők elemzéséhez, a szemantikai szempontokra összpontosítva, (v) az értékelt iránymutatások kategorizálása a felhasználói értékelés és a szakértői értékelés szempontjából, és (vi) az értékelési eredmény megjelenítése az általános akadálymentességi pontszámmal, valamint az egyes fogyatékossági típusokra vonatkozó egyedi akadálymentességi pontszámokkal. Másodszor, a javasolt keretrendszert követve egy automatizált Web Akadálymentességi tesztelő eszközt fejlesztettem ki, amely a negyedik kutatási kérdésre válaszolt:**R-Q4:** Hogyan növelhetjük egy automatizált Web Akadálymentességi értékelő eszköz hatékonyságát a fejlett mérnöki technikákra összpontosítva? A kifejlesztett automatizált webes akadálymentességi értékelő eszköz a weboldalakat a weboldalak szerkezeti és vizuális elemeinek közös jellemzőit figyelembe véve értékeli, amelyek a HTML dokumentum objektum modell (DOM) szerkezetének részét képezik. A weboldalak akadálymentességi státuszának előrejelzése érdekében három különböző algoritmus került bevezetésre a webes jellemzők/objektumok elemzésére, figyelembe véve mind a szemantikai, mind a nem szemantikai szempontokat, különböző NLP technikákra és segédmódszerekre összpontosítva.

Továbbá, a javasolt keretrendszer és a fejlesztett automatizált eszköz hatékonyságának validálása érdekében összehasonlító értékelést és kísérleti munkát végeztem, mely elvezetett az ötödik kutatási kérdéshez. **R-Q5:** Hogyan ellenőrizhető a kifejlesztett eszköz továbbfejlesztése a hatékonysági szempontok szempontjából? Először is, az összehasonlító értékelés elvégzéséhez a javasolt keretrendszert és a kifejlesztett eszközt a meglévő automatizált megoldásokkal való összehasonlítással validáltam, figyelembe véve számos funkcionális tulajdonságot. Az összehasonlító értékelés meghatároz néhány különálló jellemzőt, amelyek a jelentőségét és újdonságát képviselik. Másodszor, a kísérleti munka elvégzéséhez 20 magyarországi egyetemi honlapot értékeltem a kifejlesztett automatizált eszközzel. A kifejlesztett eszköz kiértékelt eredményét tovább értékeltem az eredménynek egy felhasználói vizsgálattal való összehasonlításával. A kifejlesztett automatizált eszköz és a felhasználói értékelés eredményei azt mutatják, hogy a javasolt eszköz képes olyan értékelési eredményt kiszámítani, amely összhangban van a résztvevő észlelésével. Ez a kétfázisú értékelés eredménye azt mutatja, hogy a kifejlesztett eszköz számos fejlett tulajdonsággal rendelkezik, és képes előre jelezni a vizsgált weboldalak akadálymentességi problémáit.

A bemutatott megközelítések, keretrendszerek és a fejlesztett automatizált eszköz lehetőséget kínálnak arra, hogy hatékonyan támogassák a webfejlesztőket, az akadálymentesítéssel foglalkozó mérnököket és más szakembereket abban, hogy az akadálymentesítést beépítsék fejlesztéseikbe és kutatásaikba. Bár a javasolt megközelítések hozzájárulása jelentős, nem mentesek a korlátoktól. A korlátok azért merültek fel, mert a javasolt megközelítések validálása érdekében a kísérleteket kis számú weboldalon végeztem, ezért további tanulmányok javasoltak, amelyekben nagyszámú, különböző területekről származó weboldalt vesznek figyelembe, és több szakértőt kérnek fel az eredmény értékelésére, ugyanazon adatokat és paraméterekét figyelembe véve. Ezzel a bemutatott megközelítés remélhetőleg hozzá fog járulni ahhoz a társadalmi politikai célhoz, hogy a fogyatékkal élők befogadásának elősegítése érdekében növeljük a webes akadály mentességgel kapcsolatos tudatosságot.

Zusammenfassung

Seit dem Start des World Wide Web im Jahr 1996 ist die Evaluierung der Zugänglichkeit von Webseiten ein wichtiger Aspekt der Webentwicklung, um die soziale Eingliederung von Menschen mit besonderen Bedürfnissen zu verbessern. In den letzten Jahren wurden erhebliche Anstrengungen unternommen, um einen nützlichen (hybriden und automatisierten) Ansatz für die Evaluierung der Zugänglichkeit von Webseiten zu entwickeln, um die Inkonsistenz bei der Ermittlung von Zugänglichkeitsproblemen für Endnutzer zu verringern. Obwohl die entwickelten Ansätze signifikant sind, können einige Probleme mit der Zugänglichkeit von Webseiten aufgrund verschiedener Herausforderungen und Einschränkungen nicht durch die bestehenden Zugänglichkeitstestansätze identifiziert werden. In dieser Situation muss ermittelt werden, welche Aspekte wichtig sind, um sie in die vorgeschlagenen Lösungen einzubeziehen, damit die entwickelten Lösungen effektiver werden, da sie es den Benutzern ermöglichen, ihre Webseiten für Leuten mit Behinderungen zugänglich zu machen.

Daher wurden in dieser Arbeit mehrere bestehende Studien aus der aktuellen Literatur ausgewertet, um die erste Forschungsfrage **R-Q1:** zu beantworten: Was sind die Herausforderungen und Nachteile des bestehenden Testverfahrens für Barrierefreiheit im Internet? Die Auswertung der vorhandenen Literatur zeigt die häufigen Probleme auf, die bei der Vollständigkeit, der Erklärbarkeit und der Berichterstattung beobachtet wurden einschließlich (i) ungeeignete Richtlinien Auswahl, (ii) Unklarheiten beim Richtlinien Verständnis, (iii) Vermeidung von Nutzer:innen- und Expert:innenvorschlägen bei der Evaluierung, (iv) eingeschränkte Berücksichtigung semantischer Perspektiven und (v) die mangelnde Bereitschaft, aktualisierte technische Methoden einzubeziehen, was die Wirksamkeit der entwickelten Ansätze verringert. Darüber hinaus können diese Aspekte das Ergebnis in einer unklaren Weise darstellen, was die Benutzer verwirren und die mögliche Annahme des jeweiligen Ansatzes direkt behindern könnte. Aus diesem Grund wurden in dieser Arbeit im Anschluss an die mit R-Q1 identifizierten Einschränkungen einige einfache und dennoch präzise Lösungen vorgestellt, um den Prozess der Evaluierung der Barrierefreiheit von Webseiten sowohl aus hybrider als auch aus automatisierter Sicht zu erleichtern.

Im Hinblick auf die hybride Evaluierung wurden drei verschiedene Ansätze vorgeschlagen, um die zweite Forschungsfrage **R-Q2:** zu beantworten: Wie können wir die Effektivität des hybriden Testverfahrens für die Zugänglichkeit von Webinhalten verbessern, um ihre derzeitigen Einschränkungen zu überwinden? Erstens wird ein integrierter Ansatz vorgestellt, der mehrere automatische Evaluierungswerkzeuge und eine von Menschen durchgeführte Evaluierung berücksichtigt. Vier automatische Tools zur Evaluierung der Barrierefreiheit (Mauve++, Nibbler, WAVE und WEB accessibility tools) und menschliche Beobachtungen wie System-Usability-Test und Expertenevaluierung herangezogen wurden zur Evaluierung der Webseiten herangezogen. Zweitens wird ein Ansatz mit variabler Größe vorgestellt, um die Zugänglichkeitsevaluierung von Webseiten unter Berücksichtigung der Ergebnisse automatisierter Lösungen und der Anforderungen von Leuten mit Behinderungen zu berechnen. Zwei verschiedene Komponenten, wie z.B. die Ausgabe von Mauve++ und Taw- Evaluierungstools, wurden als Eingangsvariablen betrachtet und integriert, um die Gesamtevaluierung der Zugänglichkeit unter Berücksichtigung der Gewichtskoeffiziententechnik zu berechnen. Drittens wird ein auf maschinellem Lernen (ML) basierender Ansatz vorgestellt, der zwei ML-Klassifikatoren berücksichtigt: Random Forest (RF) und Decision Tree (DT), um die Zugänglichkeit der Webseite zu bewerten. Dieser ML-basierte Ansatz wurde an einem benutzerdefinierten Datensatz erprobt, um die Gesamtevaluierung der Zugänglichkeit zu berechnen. Die Leistung der ML-Methoden wurde anhand der Konfusionsmatrix und des Klassifizierungsberichts validiert.

Die vorgeschlagenen drei hybriden Ansätze wurden experimentell an mehreren Webseiten aus verschiedenen Bereichen erprobt. Außerdem wurden die experimentierten Webseiten von ausgewertet Expertenevaluierung herangezogen wurden, um die Ergebnisse der vorgeschlagenen Ansätze zu rechtfertigen. Die experimentellen Ergebnisse und die Expertenevaluierung bestätigten die Fähigkeit der vorgeschlagenen Ansätze, ein effektives Ergebnis für die Evaluierung der Barrierefreiheit zu liefern.

In Bezug auf die automatisierte Evaluierung wurde zunächst ein Rahmen für die automatisierte Evaluierung der Barrierefreiheit vorgeschlagen, um die dritte Forschungsfrage **R-Q3:** zu beantworten: Wie können wir die Effektivität des automatisierten Evaluierungsprozesses der Barrierefreiheit verbessern, um die Grenzen der verfügbaren automatisierten Testwerkzeuge für die Barrierefreiheit zu überwinden? Der vorgeschlagene Rahmen berücksichtigt mehrere Aspekte der Barrierefreiheit, um die Evaluierungsergebnisse zu verbessern, wie z.B. (i) Vereinfachung der aktualisierten Richtlinien für die Barrierefreiheit von Webinhalten, (ii) Einbeziehung aller Erfolgskriterien in den Evaluierungsprozess, (iii) Einbeziehung von Benutzerinnen - und Benutzeranforderungen/Meinungen mit Expertinnen- und Expertenvorschlägen als zusätzliches Evaluierungskriterium, (iv) Einbeziehung separater Komplexitätsanalysealgorithmen für die Analyse textlicher und nicht-textlicher Merkmalsanalyse indem Fokussierung auf semantischen Aspekten, (v) Kategorisierung der bewerteten Richtlinien in Bezug auf die Nutzerinnen- und Nutzerevaluierung und die Expertinnen- und Expertenevaluierung und (vi) Darstellung des Evaluierungsergebnisses mit der Gesamtevaluierung der Barrierefreiheit Punktzahl zusammen mit spezifischen Barrierefreiheitsevaluierungen Partituren für jeden Behinderungstyp. Zweitens wurde auf der Grundlage des vorgeschlagenen Rahmens ein automatisiertes Werkzeug zur Prüfung der Zugänglichkeit von Web-Inhalten entwickelt, das die vierte Forschungsfrage **R-Q4:** Wie können wir die Effektivität eines automatisierten Werkzeugs zur Evaluierung der Zugänglichkeit von Web-Inhalten erhöhen, indem wir auf fortgeschrittene Ingenieurtechniken fokussieren? Das entwickelte automatische Tool zur Evaluierung der Zugänglichkeit von Webinhalten bewertet Webseiten unter Berücksichtigung gemeinsamer Merkmale der strukturellen und visuellen Elemente von Webseiten, die Teil der HTML Document Object Model (DOM) Struktur sind. Um den Zugänglichkeitsstatus einer Webseite vorherzusagen, werden drei verschiedene Algorithmen zur Analyse von Webmerkmalen/Objekten implementiert, die sowohl semantische als auch nicht-semantische Aspekte berücksichtigen sich auf verschiedene NLP-Techniken und Hilfsmethoden konzentrieren.

Um die Wirksamkeit des vorgeschlagenen Rahmens und des entwickelten automatisierten Werkzeugs zu validieren, wurden außerdem eine vergleichende Evaluierung und experimentelle Arbeiten durchgeführt, die sich mit der fünften Forschungsfrage **R-Q5:** befassen: Wie können die Fortschritte des entwickelten Werkzeugs in Bezug auf seine Wirksamkeitsaspekte überprüft werden? Für die vergleichende Evaluierung wurden zunächst das vorgeschlagene Rahmenwerk und das entwickelte Werkzeug durch einen Vergleich mit bestehenden automatisierten Lösungen unter Berücksichtigung verschiedener funktionaler Eigenschaften validiert. Bei der vergleichenden Evaluierung wurden einige Merkmale ermittelt, die die Bedeutung und Neuartigkeit des Tools darstellen. Zweitens wurden für die experimentelle Arbeit 20 Universitätswebseiten in Ungarn mit Hilfe des entwickelten automatischen Werkzeugs bewertet. Das Ergebnis der Evaluierung des entwickelten Tools wurde durch den Vergleich mit einer Nutzerstudie bewertet. Die Evaluierungsergebnisse des entwickelten automatischen Tools und der Nutzerstudie zeigen, dass das vorgeschlagene Tool in der Lage ist, ein Evaluierungsergebnis zu berechnen, das mit der Wahrnehmung der Teilnehmer übereinstimmt. Dieses zweistufige Evaluierungsergebnis zeigt, dass das entwickelte Tool über mehrere fortschrittlichen Eigenschaften und das Potenzial verfügt, die Zugänglichkeitsprobleme der getesteten Webseite vorherzusagen.

Die vorgestellten Ansätze, Frameworks und das entwickelte automatisierte Werkzeug haben das Potenzial, Webentwickler, Ingenieure für Barrierefreiheit und andere Praktiker:innen bei der Integration von Barrierefreiheit in ihre Entwicklung und Forschung effizient zu unterstützen. Obwohl der Beitrag der vorgeschlagenen Ansätze bedeutend ist, ist er nicht frei von Einschränkungen. Die Einschränkungen wurden dadurch identifiziert, dass die Experimente mit kleinen Mengen von Webseiten durchgeführt wurden, um die vorgeschlagenen Ansätze zu validieren. Daher werden weitere Studien vorgeschlagen, um eine große Anzahl von Webseiten aus verschiedenen Bereichen zu berücksichtigen und mehr Expert:innen einzuladen, die Ergebnisse unter Berücksichtigung der gleichen Daten und Parameter zu bewerten. Auf diese Weise wird der vorgestellte Ansatz hoffentlich zu dem gesellschaftspolitischen Ziel beitragen, das Bewusstsein für die Barrierefreiheit von Webseiten zu schärfen, um die Integration von Menschen mit Behinderungen zu fördern.

Acknowledgements

First and foremost, I would like to express my gratitude and gratefulness to Almighty Allah (the almighty God) for allowing me and helping me to successfully complete my doctoral thesis work in time.

I would like to express my gratitude to my supervisor Prof. Dr. Cecilia Sikne Lanyi, for her enormous support, guidance, and loving attitude that helped me to continue my work throughout my PhD journey.

Also, I would like to show my gratitude to all the members of the Doctoral School of Information Science and Technology, University of Pannonia, Hungary, for their guidance throughout the journey.

I am grateful to my friends and family especially my parents and two sisters, who have always been beside me in all situations. Their motivation and helpful attitude motivated me to continue my PhD studies.

Also, I would like to thank the Hungarian Government for allowing me to study in this beautiful country by awarding me the Stipendium Hungaricum (SH) scholarship for conducting my PhD study. Beside the SH scholarship, successfully, I have successfully secured additional research funding under the University Research Scholarship Program (EKÖP 2024/2025), University of Pannonia.

Additionally, throughout my PhD journey, some of my research activities were financially supported by the CA19104–Advancing Social Inclusion through Technology and EmPowerment project.

Contents

List of Figures

List of Tables

Abbreviations

W3C	**W**orld Wide Web Consortium
WAI	**W**eb Accessibility Initiative
WCAG	**W**eb Content Accessibility Guidelines
WHO	**W**orld Health Organization
EU	**E**uropean Union
UN	**U**nited Nation
ML	**M**achine Learning
NLP	**N**atural Language Processing
WCAEE	**W**eb Content Accessibility Evaluation Environment
WWW	**W**orld Wide Web
ATAG	**A**uthoring Tool Accessibility Guidelines
HCI	**H**uman-Computer Interaction
EDBA	**E**valuator-Decision-Based Assignment
HTML	**H**yper Text Markup Language
ACT	**A**ccessibility Conformance Testing
VCS	**V**isual Complexity Score
DOM	**D**ocument Object Model
RIAs	**R**ich Internet Applications
NCAM	**N**ational Center for Accessible Media
XML	**E**xtensible Markup Language
XSLT	**E**xtensible Stylesheet Language Transformations
CVD	**C**olor Vision Deficiency
UI	**U**ser Interface
CAPTCHA	**C**ompletely Automated Public Turing test to tell Computers and Humans Apart

MAUVE	**M**ultiguideline Accessibility Usability Validation Environment
JSON	**J**avaScript Object Notation
CSS	**C**ascading Style Sheets
URL	**U**niform Resource Locator
SUS	**S**ystem Usability Scale
SARS-CoV-2	**S**evere Acute Respiratory Syndrome Coronavirus 2
API	**P**rogrammatic Interfaces
RF	**R**andom Forest
DT	**D**ecision Tree
CV	**C**ross-Validation
TP	**T**rue Positive
FP	**F**alse Positive
TN	**T**rue Negative
FN	**F**alse Negative
SD	**S**tandard Deviation
AI	**A**rtificial Intelligence
GUI	**G**raphical User Interface

Symbols

ξ	Significance level/threshold value
α	Input URL
ϕ	Empty set
n	Number of samples
X	Targeted tested webpage
β, κ	Number of passed guidelines
φ, λ	Number of failed guidelines
γ, μ	Number of not-tested guidelines
δ, χ	Number of not-decided guidelines
Φ	Severity
ω_β, ω_φ, ω_γ, ω_δ	Weight coefficients [Mauve++]
ω_κ, ω_λ, ω_μ, ω_χ	Weight coefficients [Taw]
ϵ	Severity factor
ρ_1	Passed ratio
ρ_2	Failed ratio
ρ_3	Not tested ratio
ρ_4	Not decided ratio
$\rho_{(Mauve)}$	Mauve score
$\rho_{(Taw)}$	Taw score
ρ	Overall accessibility score

Chapter 1

Introduction

1.1 Overview of Web Accessibility and its Objective

According to the World Wide Web Consortium (W3C) report, the accessibility of the Web is a broad and extensible term associated with people who have disabilities, incompetent skills, or situational-induced impairment [1]. This initiative's objective is to ensure the accessibility of digital platforms, which means people with special needs should be able to access, navigate, interact, and contribute to the information available on the Web/Internet, electronic resources/materials, and computers. The Web or the Internet platform generally refers to the digital services of webpages, whereas accessibility of the web refers to the design and development of webpages in a way that is effective for people with disabilities and without disabilities [2]. In general, webpage is a digital platform for sharing varieties of information (healthcare, education, e-Commerce, etc.) with people. However, sometimes these digital resources are not properly organized (such as broken links, out-of-date content, dilated images, etc.). From the perspective of digital services, digital resources should be organized to make them accessible and barrier-free to people with disabilities [3]. From this perspective, the accessibility ensurement of digital platforms or webpages has been addressed multiple times by the Web Accessibility Initiative (WAI), and they set their mission to coordinate international, technical and human efforts to improve and ensure web accessibility globally [4]. With this mission in mind, WAI launched a set of accessibility guidelines called Web Content Accessibility Guidelines (WCAGs) [5][6]. However, the web accessibility guidelines suggest that a webpage should have an advanced design and the latest technology for development so that people with disabilities can perceive, understand, navigate, and interact with the Web more efficiently [7]. A detailed description of the WCAG has been demonstrated in Chapter 2.

Although W3C and WAI initiated their great effort in the area of web accessibility, some recent studies showed that most webpages even fail to maintain basic accessibility requirements or minimum accessibility standards [8][9]. As a result, people with disabilities experience several difficulties with access to the web. For example, web content information might be difficult to read and understand, the meaning of the placement of the user interface elements might be difficult to identify or remember, and some interactive designs (dropdown menu, sub tasks, landing page, etc.) might make the content partially or completely inaccessible. In addition, since accessibility problems are distinct according to the type of disability, problems or difficulties might vary from person to person and in different situations. Among several scenarios of difficulties, more particularly, people with vision problems have difficulties understanding content written in small font and a specific theme (italic and bolded). People with color blindness have difficulties recognizing specific colors, and people with cognitive difficulties have issues understanding the meaning of some complex or advanced words, notations, abbreviations, and alerts. Also, people with motion difficulties have issues with scrolling and pointing the drop-down menu. In that case, people with disabilities are forced to spend more time on webpages to find their required information than people without disabilities.

In a recent report, statistics have shown that a significant number of users with disabilities are active on online platforms, and this number is steadily increasing [10]. About 81% of users with various disabilities still experience difficulty, and sometimes, they can't perceive the content effectively[11]. According to World Bank statistics, one billion people (15% of the world population) across the world have some disabilities such as visual problems (vision impairment or vision loss), auditory problems (hearing disability), physical disabilities (disorder in a body part), speech problems (disorder speech production), cognitive or neurological problems (disorder in cognitive functions) and others [12]. According to the World Health Organization (WHO) report, among the whole population of the world, 3.2% of people have visual impairment, 6% of people have auditory or hearing difficulties, 2.6% of people have neurological difficulty, and 1% of people have physical disabilities or require a wheelchair [13].

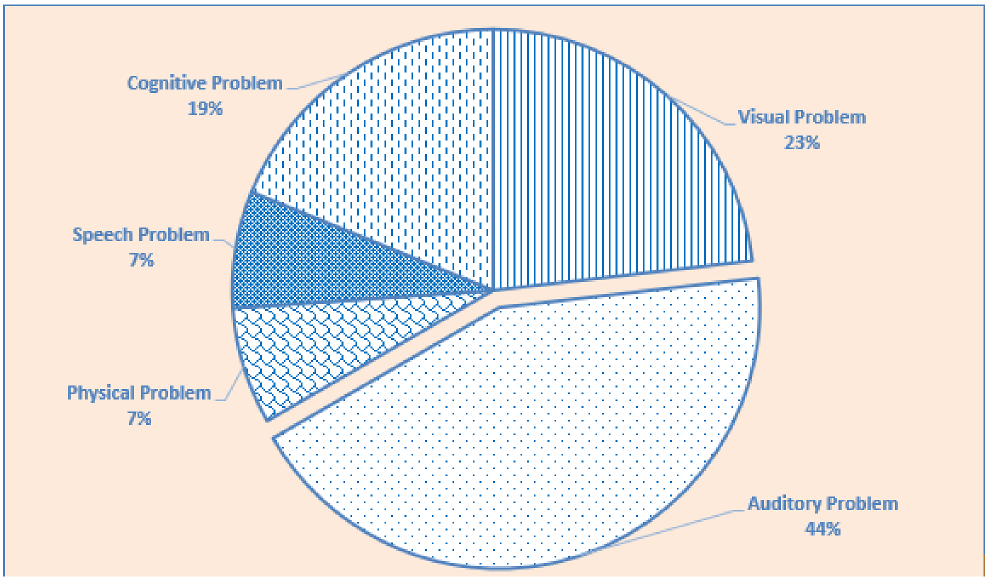


FIGURE 1.1: Worldwide ratio of people with disability

Based on WHO statistics, Figure 1.1 shows that for every 100 people, 44 people have auditory problems, 23 people have visual problems, 19 people have cognitive problems, and 7 people have speech and physical problems across the world. According to statistics from the European Union (EU), in Europe, one in six people have some disabilities and are actively involved online to search for information [14]. The United Nations (UN) reported that, in the Asian region, more than 10% people with disabilities are actively involved in the internet platforms for information [15]. Thus, to ensure accessible digital services or make the web platform completely accessible to these large numbers of people with disabilities, web accessibility aims to ensure consideration of the requirements of people with disabilities in the design and implementation of the web application.

1.2 Overview of the limitations of web accessibility research

In recent years, various aspects have motivated researchers to conduct studies on digital accessibility. The extension and increased availability of the web for multiple purposes (e.g., information search), the representation of the content (e.g., video, audio), and the emergence of new platforms (e.g., Internet of Things) and technologies (e.g., mobile, computer, tablets) are significant aspects to reinforce the investigation of the digital information platform. In particular, since the very beginning of the digital revolution, digital resources have become the fundamental source for citizens to access information such as education, health care, government, news, and other information such as entertainment and sports [16][17]. In recent days, a large amount of data has grown faster in the cloud, and people are rapidly engaging online [18][19][20]. Therefore, for ensuring online information, accessibility is crucial for technologists and researchers. With this aim, numerous research studies have been conducted that focus on how the Web platform can be completely accessible to the community.
Addressing this concern, most of the studies focused on the statistical analysis of Web accessibility and the development of tools and applications to measure web accessibility. Here, statistical analysis of web accessibility refers to studies that focus on incorporating statistical measurement to determine the accessibility of a particular webpage. In addition, accessibility measurement tools and application development refer to the complete development of a particular approach or process to evaluate webpages to determine their accessibility. The tool development process can be automated or a hybrid process. In an automated process, accessibility measurement can incorporate existing automated accessibility testing tools, or a completely new automated tool can be developed for accessibility measurement. In addition, the hybrid testing process allows the incorporation of both automated tools and human perception to measure the accessibility of the webpages. However, for the design and development methods of accessibility investigation tools for webpages, number of studies related to hybrid and automated evaluation processes is limited compared to statistical analysis due to lack of clarity, poor authenticity, and increased additional costs of investigation. Besides, studies for the development of hybrid and automated webpage accessibility testing applications are not significant compared to other processes, such as accessibility testing using existing tools. In addition, those studies found that the web accessibility tool design and development process have issues related to engineering aspects. Mostly, accessibility testing tools largely neglect advanced engineering assets in their consideration. Due to the lack of advanced engineering assets in consideration, studies also found that most existing tools or evaluation approaches are not useful to end users, hindering their effectiveness in the community. Therefore, this introduces the importance of investigating different web accessibility evaluation processes and tool development

approaches to determine an effective method and framework addressing how advanced engineering aspects can be integrated into the evaluation and development process to validate webpage accessibility effectively.

1.3 Research objective

Nowadays, web designers and developers are trying to incorporate several complex functionalities (e.g., dynamic functionalities, drop-down menu) and components (e.g., images, videos) into their webpages to make them more interactive. Although these interactive features are prominent in the attraction of more people, they limit the accessibility concept for users with disabilities [21][22]. Therefore, the demand for an effective tool to evaluate these interactive components in terms of accessibility has increased in the community. Thus, it is increasingly important to evaluate several existing accessibility testing processes to determine their possible vulnerabilities and, by addressing those vulnerabilities, propose an effective accessibility testing approach to validate the accessibility of the webpages. To achieve this purpose, the objective of this research is five-fold as follows:

- **Objective 1:** To understand, evaluate, and determine the strengths and limitations of existing web accessibility evaluation approaches. This objective corresponds to R-Q1.
- **Objective 2:** To minimize the finding limitations and improve the effectiveness of the hybrid web accessibility evaluation result, propose multiple effective hybrid web accessibility evaluation approaches. This objective corresponds to R-Q2.
- **Objective 3:** To minimize the finding limitations and improve the effectiveness of the automated web accessibility evaluation result, propose an effective automated web content accessibility evaluation framework. This objective corresponds to R-Q3.
- **Objective 4:** Following the proposed automated evaluation framework, develop an automated web content accessibility evaluation tool or environment for automatically validating webpage accessibility. This objective corresponds to R-Q4.
- **Objective 5:** To determine the effectiveness of the developed automated evaluation tool, validate the developed tool by comparing it with existing models using their functional properties, and conduct a user study. This objective corresponds to R-Q5.

1.4 Overview of research methodology

The web accessibility research area is evolving rapidly due to technological advances and continuous changes in the knowledge and techniques of web development. Therefore, determining appropriate strategies and incorporating them interactively is an important and, at the same time, difficult task for web researchers. The record of past literature shows that several solutions have been found in the literature addressing different perspectives on the area of web accessibility research. However, a significant number of studies do not contribute to improving the web accessibility evaluation process or the evaluation results. To address this research gap, this Ph.D. research aims to concentrate on different web accessibility evaluation processes or tools design and development for the end users to determine the accessibility status of their web resources (e.g., webpages). The entire investigation has been conducted following several phases that are shown in Figure 1.2 and demonstrated its overview in the following.

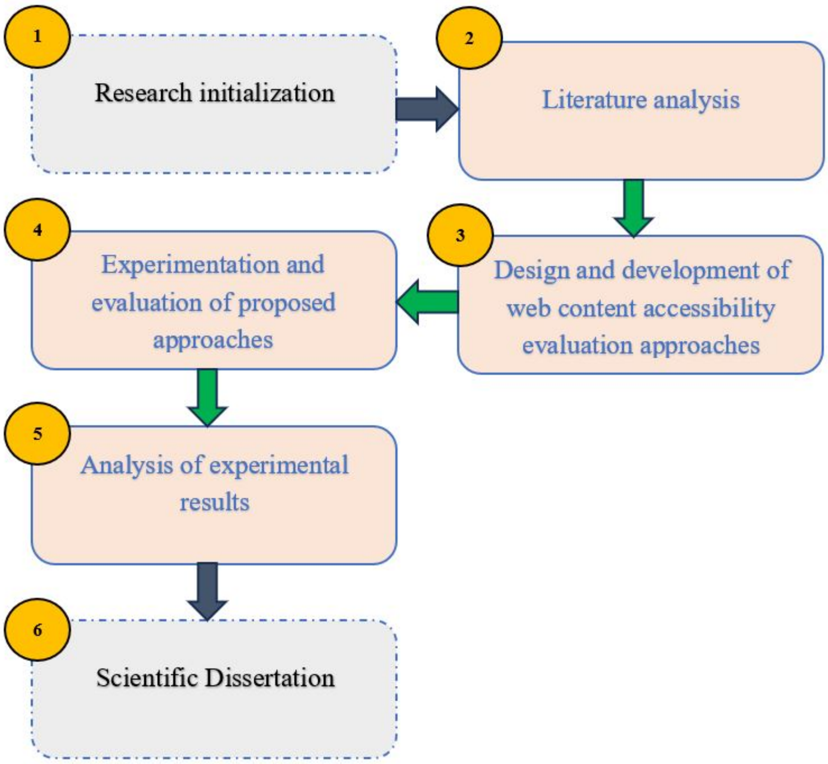


FIGURE 1.2: Research flow diagram

- **Phase-I (Research initialization):** The primary objective of this initial phase is to initialize the research topic, determine the research roadmap and finalize the research strategies to enhance the overall contribution to scientific research and successfully submit the dissertation.
- **Phase-II (Literature analysis):** This phase aims to investigate the literature on the area of web content accessibility evaluation considering the identification of research gaps and their current challenges. Focusing on this goal, all considered, evaluated, or relevant studies were scientific literature published in peer-reviewed journals and conferences around the world. This phase of the research helped to understand the research topic from both theoretical and practical perspectives, provided research direction, and helped to formulate research questions to complete the research with sufficient results.
- **Phase-III (Design and development of web content accessibility evaluation approaches):** This phase aims to design and develop different web content accessibility evaluation approaches (hybrid and automated) to minimize the challenges addressed in similar existing processes. Different advanced features, techniques, and strategies have been considered to propose the new Web Content Accessibility Evaluation Process, considering hybrid and automated solutions.

- **Phase-IV (Experimentation and evaluation of proposed approaches):** The primary goal of this phase is to experiment with the proposed approaches resulting from Phase III. All proposed approaches (hybrid and automated solutions) are tested on a sample of data considering several webpages from different domains to determine their accessibility status. This experiment helps to evaluate and determine the effectiveness of the proposed approaches or solutions.
- **Phase-V (Analysis of experimented results):** This phase aims to analyze the experimented result returning from Phase IV to validate the proposed approaches and to determine whether the proposed approaches are capable enough to improve the limitations addressed in the existing literature. This phase also helps to determine the effectiveness and prove the ability or capability of the proposed solutions.
- **Phase-VI (Scientific dissertation):** After completing Phase I to Phase V, this phase objective is to present detailed findings of the proposed approaches with their potential to contribute and minimize the research gap addressed in the field of web accessibility research area.

1.5 Research questions and hypothesis

To achieve the objectives mentioned above, five research questions have been followed throughout the investigation. The research questions with their hypothesis are the following.

R-Q1: What are the challenges and drawbacks of the existing web accessibility testing process?
H-1: Several issues associated with the effectiveness of existing approaches make the evaluation process biased.

R-Q2: How can we improve the effectiveness of the hybrid web content accessibility testing process to address its current limitations?
H-2: Incorporating several techniques, such as the participation of user and expert testing, the concept of variable magnitude alternation, and machine learning (ML) techniques, can be helpful in improving the effectiveness of the hybrid web content accessibility testing process.

R-Q3: How can we improve the effectiveness of the automated web accessibility evaluation process to address the limitations of the available automated web accessibility testing tools?
H-3: The design and development process could be facilitated by incorporating a wide range of elements, including suitable guidelines, user and expert requirements, and advanced engineering techniques to increase the effectiveness of the automated web accessibility evaluation process.

R-Q4: How can we increase the effectiveness of an automated web content accessibility assessment tool focusing on advanced engineering techniques?
H-4: The enhancement of the effectiveness of the tool may be greatly impacted by implementing separate algorithmic evaluations for arbitrary and non-arbitrary items focusing on semantic approaches by considering advanced engineering techniques, such as Natural Language Processing (NLP).

R-Q5: How can the progress made with the developed tool be verified in terms of its effectiveness aspects?

H-5: The effectiveness of the developed tool can be determined and represented by its functional properties. The results of user evaluations may also be useful in providing the actual scenario of effective components.

1.6 Innovation of this research

The prime contributions of this dissertation focused on introducing different approaches to solve the issues of the accessibility evaluation process of web content. The main contributions are presented as follows:

- Presented a detailed analysis to determine the strengths and limitations of the existing web accessibility evaluation process (can be found in Chapter 2).
- Presented an integrated approach to incorporate (i) multiple automated tools and (ii) human observation (user testing and expert testing) to improve the performance of the hybrid web accessibility evaluation process (can be found in Chapter 3).
- Presented an approach to apply variable magnitude alteration to improve the performance of the hybrid web accessibility evaluation process (can be found in Chapter 3).
- Presented an approach to apply Machine Learning (ML) techniques to improve the performance of the hybrid web accessibility evaluation process (can be found in Chapter 3).
- Proposed a new automated web accessibility evaluation framework to improve the effectiveness of the automated web content accessibility evaluation results (can be found in Chapter 4).
- Introduced a new automated web content accessibility evaluation tool named Web Content Accessibility Evaluation Environment (WCAEE) to evaluate and improve the performance of the automated evaluation results (can be found in Chapter 4).
- Presented a validation approach of the proposed or developed WCAEE tool considering a comparative evaluation focusing on functional properties with existing tools and incorporating a user study (can be found in Chapter 4).
- Answered all research questions with the decision about the corresponding hypothesis considering each thesis group (which can be found in Chapter 5).

1.7 The structure of this research

The entire thesis work has been demonstrated in several chapters. In Chapter 2, accessibility concepts in the web evaluation process, their importance, several related works from the existing literature, and their associated issues that are responsible for reducing the effectiveness of existing solutions are presented. Considering the raised issues, Chapter 3 focuses on hybrid techniques with several new hybrid approaches to improve the effectiveness of current hybrid evaluation systems to minimize their associated issues. After that, Chapter 4 focuses on automated techniques, concentrating on the automated evaluation framework with a developed automated Web Content Accessibility Evaluation Environment (WCAEE) tool to address the limitation and improve the effectiveness of existing automated accessibility testing tools. Also, Chapter 4 concludes the validation process of the developed WCAEE automated tool. Finally, Chapter 5 concludes the thesis work by providing details of the significance of the proposed solutions, the answer to each research question, and a decision about each hypothesis with their determined thesis group. Furthermore, Appendix A, Appendix B and Appendix C presents all additional Figures, Tables and resources that support the dissertation, respectively.

Chapter 2

Background of Study

The World Wide Web (WWW) was designed in 1989 to make information accessible to people [23]. The WWW is an advanced platform for instant sharing of information between large numbers of people [24]. Nowadays, the WWW provides various services (information sharing, communication, administration tasks, etc.) for different sectors such as education, healthcare, and the e-Commerce sector. The variety of services offered on the Web is unprecedented. This global platform has enormous functionalities and allows for a wide spectrum of devices, highlighting the uniqueness of the Web [25]. Although the WWW can potentially improve global standards, it might not be accessible to people with disabilities if its use is not designed properly. Therefore, according to the opinion of the universal accessibility specialist, the use of the WWW, particularly the web platform, should be designed with regard to all the requirements of people with disabilities to keep this digital platform accessible to them [26]. With this concern, several web accessibility guidelines and evaluation methodologies have been introduced in the past few decades. However, these existing methodologies or approaches for web content accessibility evaluation have several limitations that can reduce the effectiveness of the developed solutions or approaches. Thus, addressing the introduced guidelines and the proposed solutions, including their potentials and limitations, the first aim of this chapter is to provide a brief description of web content accessibility evaluation insights, including standard guidelines and related studies from existing literature. The second aim of this chapter is to identify the limitations or drawbacks of existing approaches or solutions and their associated challenges to answer the first research question (R-Q1), which helps to understand web accessibility insights in detail as follows.

R-Q1: What are the challenges and drawbacks of the existing web accessibility testing process?

2.1 Basic Definitions

All the presented definitions are in the context of the web accessibility assessment and testing process:

Hybrid Evaluation: Hybrid evaluation is the process of integrating automated and manual evaluation, considering automated evaluation tools and concurrently involving end-user and expert evaluation.

Crowdsourcing Approach: Crowdsourcing evaluation is the practice of combining multiple automated assessment tools with multifaceted manual evaluation that simultaneously involves the user and expert perspectives.

Heuristic Approach: Heuristic evaluation is the practice of integrating multifaceted manual evaluation that simultaneously involves expert and non-expert (end-user) evaluation, rather than focusing on automated assessment.

Automated evaluation: Automated evaluation is the process of automatically evaluating the source code of webpages using computer programs against accessibility issues, following standard guidelines.

Declarative Model: Instead of outlining how to conduct the tests, the declarative model describes the fundamental requirements that must be addressed in the development process using the suggested sets of atomic tests to create complicated rules.

Ontological Model: The Ontological model organizes and represents the semantic knowledge of Web prototypes with semantic design techniques and related taxonomy so that the accessibility of web prototypes can be conveniently determined.

Algorithmic Evaluation: Algorithmic evaluation is the process of evaluating web prototypes through distinctive algorithms that represent a set of finite rules to confirm that each prototype satisfies accessibility requirements following accessibility standards.

2.2 Accessibility Standards

Digital accessibility is a process to ensure the availability of online tools or content for users [27]. The primary objective of digital accessibility is to create an accessible, operable, and interactable online platform to provide equal information access opportunities for people with disabilities [28][29]. Several aspects might initiate barriers to implementing and ensuring digitally accessible platforms or tools, or content, such as limited accessibility knowledge and its guidelines. Sometimes organizational barriers and parameters, such as organization size, capital, and cost, influence accessibility issues. Addressing these issues, the governments and organizations of several countries declared various guidelines, standards, and conformance levels for stakeholders [30]. Following these guidelines, associate authorities might overcome critical issues and ensure digital accessibility. Generally, accessibility guidelines are introduced by the government of several countries and various public and private institutes (such as WCAG, Section 508, EN 301 549, YD/T 1761–2012, WAI-ARIA, BITV, ISO 9241, and ATAG are prominent) to develop an accessible solution (e.g. application, webpages, software, etc.).

The [1]Web Content Accessibility Guideline (WCAG) was introduced by the Web Accessibility Initiative of the World Wide Web Consortium with several success criteria under 13 guidelines. [2]Section 508 is accessibility requirements rules published by the US government to make digital resources accessible. [3]EN 301 549 is a European accessibility requirement, suitable for public procurement of ICT products

[1]http://www.w3.org/WAI/

[2]https://www.regulations.gov/accessibility

[3]https://en.wikipedia.org/wiki/EN_301_549

and services in Europe. [4]YD/T 1761–2012 refers to the standards of Chinese Technical requirements for Web accessibility that primarily focus on ensuring accessibility in the digital platform. Besides, the [5]WAI-ARIA standard published by W3C defines a set of guidelines for HTML attributes to improve semantic accessibility. [6]BITV is a German standard that is issued focusing on WCAG 2.0 to make the webpage and application accessible to people with disabilities by ensuring perceivable, operable, understandable, and robust guidelines. Similarly, [7]ISO 9241 provides requirements for accessible development throughout the application development life cycle. It concerns both hardware and software components for interactive design and development. [8]Authoring Tool Accessibility Guidelines (ATAG) is WCAG and User Agent Accessibility Guidelines-based instruction to design and develop accessible web content. Among these guidelines, WCAG is the most widely used accessibility standard. WCAG is a documented guide that explains all accessibility criteria and step-by-step recommendations on the implementation, improvement, and measurement of accessibility to provide a better user experience, especially for people with disabilities. The WAI, W3C-based guideline, first developed the WCAG standards to make the Web accessible [1]. By 2024, WAI published five versions of the WCAG standard, including WCAG 1.0, WCAG 2.0, WCAG 2.1, WCAG 2.2, and WCAG 3.0 (draft version). WCAG 3.0 is the most sophisticated standard, currently available as a working draft for web developers (front and back end) and designers to develop usable and accessible web content [31].

In 1999, W3C released the first version of WCAG 1.0 with three priorities, 14 guidelines, and 65 checkpoints [32]. In 2008, W3C released the second version of standards/guidelines, including 61 success criteria and 12 guidelines under four principles: perceivable, operable, understandable, and robust, concerning three conformance levels: Level A, Level AA, and Level AAA [33]. Furthermore, the W3C published an updated version of the WCAG 2.0 principles, namely the WCAG 2.1 standard, in 2018 [6]. It has all the principles, guidelines, success criteria, and conformance levels similar to WCAG 2.0, but added one new guideline and 17 new success criteria. Therefore, completion of the WCAG 2.1 standard ensures the fulfillment of WCAG 2.0, and is followed by more accessibility concerns. The significant update in WCAG 2.1 is the 'Operable' principle. In this principle, a new guideline has been added with six success criteria.

In 2021, W3C extended the WCAG 2.1 guideline and released WCAG 2.2, an updated version [34]. In this version, in the Operable principle under guideline (2.4), three new success criteria have been added. In December 2021, the last modified version of WCAG (3.0, working draft) was published, now in progress, waiting for the final draft of guidelines [31]. Figure 2.1 shows the WCAG standard with principles, success criteria, and conformance levels. For a brief discussion on success criteria and conformance level, the author refers the reader to [34]. In addition, all WCAG versions followed the three conformance levels (A, AA, and AAA) to classify web content. By following the WCAG standard, developers and designers can make digital content accessible to a wide range of people with disabilities, including blindness, low vision or vision impairments, deafness and hearing loss, limited movement, dexterity, speech disabilities, sensory disorders, cognitive and learning disabilities, photosensitivity and combinations of these [35]. Nowadays, ensuring an accessible web and improving the user experience are crucial for web

[4]https://www.chinesestandard.net/PDF.aspx/YDT1761-2012
[5]https://www.w3.org/TR/wai-aria-1.2/
[6]https://www.continualengine.com/blog/bitv-compliance-for-accessibility/
[7]https://www.iso.org/standards.html
[8]https://www.w3.org/TR/ATAG20/

engineers, researchers, and developers. According to the researchers' opinions, more research needs to be done in the upcoming years focusing on accessibility standards to improve the accessibility of digital platforms [36].

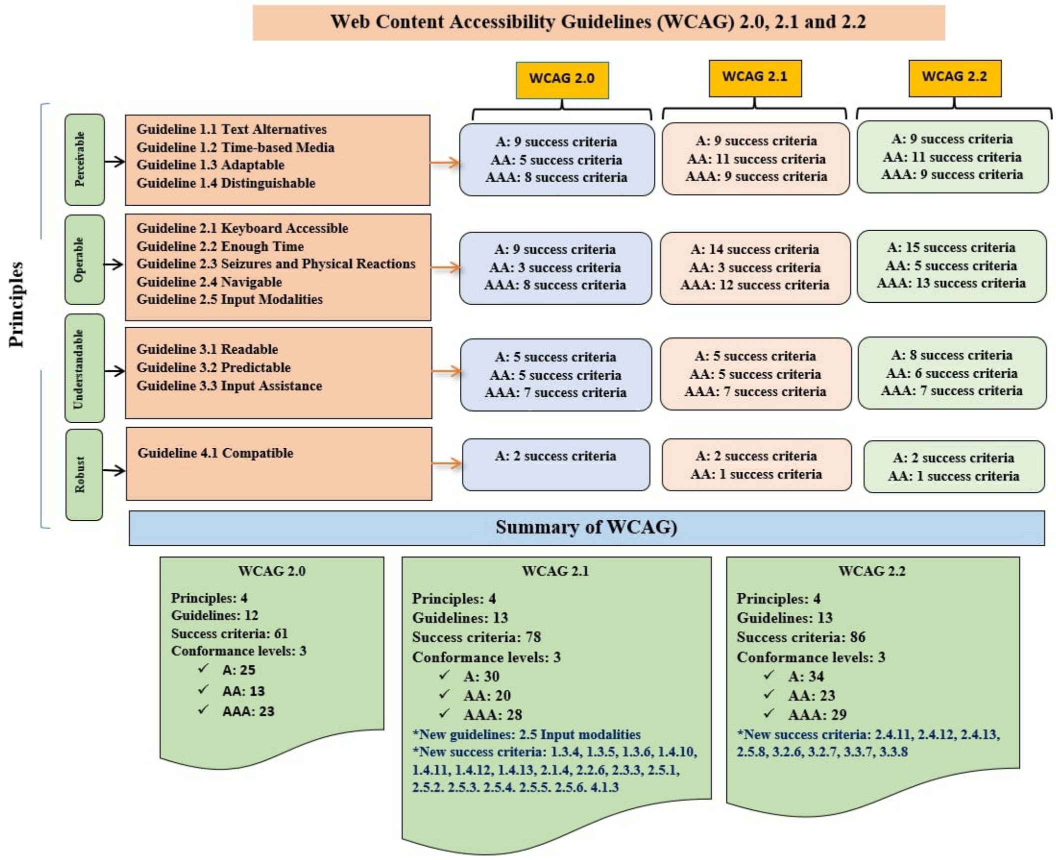


FIGURE 2.1: Overview of web content accessibility guidelines (WCAG) version 2.0, 2.1 and 2.2

2.3 Methodological contribution of existing web content accessibility research

Accessibility computation and validation are important research areas that are aligned with the domain of human-computer interaction (HCI) [37]. The HCI domain focuses on effective interaction between humans and computers, while accessibility computation focuses on effective service to people with disabilities, to empower them through computer technology [38]. The objective of accessibility computation is to incorporate computer technology into human activity, as computer technology can reduce people's dependency on human assistance that might be expensive, complex, or unavailable on demand. In particular, computer technology can be a great advantage for people with disabilities by providing different advanced services that could replicate human interaction. Therefore, with the help of advanced technology, the global mandate is to ensure the design and development of webpages that consider accessibility standards and follow accessibility practices to provide equal access to resources. The record of previous literature shows that the majority of the proposed approach evaluated the effectiveness of webpages in terms of their quality (e.g. broken link, interactivity), usability (e.g. HTML page, aesthetic, design, page size) and accessibility [39][40][41]. Concerning the increasing number of people with disabilities, recently researchers from different backgrounds (e.g., technology, education, neurodevelopment) have sought to evaluate the accessibility of online platforms (e.g., webpages) [42]. Generally, in the past few decades, several techniques or methods have been introduced to evaluate web accessibility. Two Web Accessibility Evaluation Techniques are more popular and effective: (i) hybrid web accessibility evaluation (both automatic and manual judgment) and (ii) automated web accessibility evaluation (through a computer program).

The hybrid web accessibility evaluation/testing process performs automatic and manual evaluation (e.g., end-user evaluation) concurrently, as shown in Figure 2.2. Sometimes, automated and manual evaluation particularly involves user evaluation and expert testing, which is also known as hybrid testing. It helps compare and improve multiple testing results simultaneously. Sometimes, automated tools cannot determine all accessibility issues according to accessibility guidelines and require human judgment. Therefore, manual evaluation or observation helps to identify the limitations of automated accessibility testing tools.

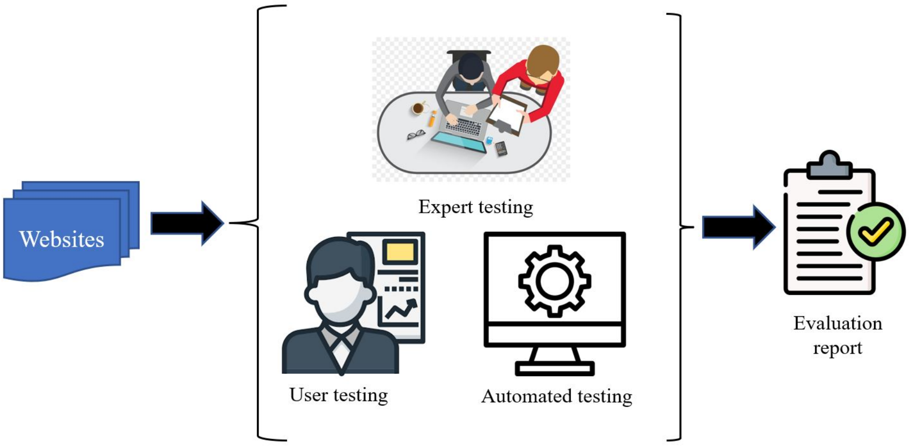


FIGURE 2.2: View of Hybrid Web Accessibility Testing process

In particular, five different scenarios can be raised in the hybrid web accessibility evaluation/testing process as follows:

Scenario-1: Incorporation of → User testing ∪ Expert testing ∪ Automated testing

Scenario-2: Incorporation of → User testing ∪ Expert testing

Scenario-3: Incorporation of → User testing ∪ Automated testing

Scenario-4: Incorporation of → Expert testing ∪ Automated testing

Scenario-5: Incorporation of → Automated testing_1 ∪ Automated testing_2 ∪.......... ∪ Automated testing_n

Furthermore, automated testing refers to the validation of accessible features of Web content through computer programs against accessibility guidelines, as shown in Figure 2.3. Also, the automatic evaluation technique allows the tester to evaluate the accessibility of webpages by running the evaluation tools in the browser. It also refers to web accessibility testing software that determines the quality of the webpage based on predefined web accessibility standard guidelines.

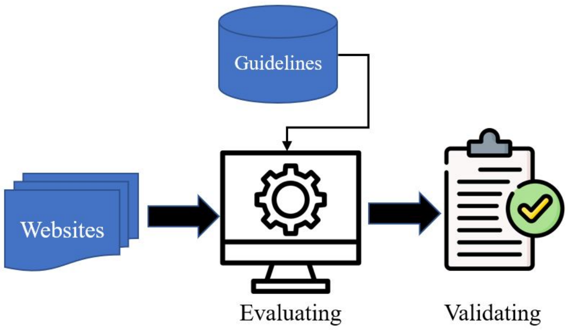


FIGURE 2.3: View of Automated Web Accessibility Testing

According to this basic understanding, several solutions from the existing literature on the area of hybrid and automated solutions for web accessibility testing have been demonstrated in the following sections with the aim of determining their potentials and limitations that might help to propose more advanced solutions in future to facilitate such developments and improve the performance of the future web accessibility evaluation results.

2.3.1 Hybrid evaluation approaches

Generally, accessibility evaluation of webpages is a difficult task that requires incorporating several aspects (web features, requirements of people with disabilities, expert opinion, etc.) to reduce the shortcomings of accessibility testing results. To avoid ineffective evaluation results, researchers suggested that hybrid evaluation could be effective in mitigating such issues to improve the precision of the evaluated result and retain the fairness of the evaluation process, as it allowed multiple investigation resources. In other words, the hybrid evaluation process is known as evaluating webpages in terms of automated, user, and expert evaluation. In addition, it incorporates the user and expert requirements and suggestions to improve the effectiveness of the evaluation process. From the past literature, several researchers have conducted hybrid evaluations of web accessibility, which can be divided into two categories: Crowdsourcing systems and Heuristic approaches, which have been described in detail in the following sections.

2.3.1.1 Crowdsourcing System (CS)

Generally, crowd-sourcing approaches allow multiple resources to be incorporated together to evaluate the accessibility of webpages. However, most of the existing research considered crowdsourcing approaches as the incorporation of multiple automated evaluation tools, rather than implementing multidimensional sources, which reduces the effectiveness of the tested results. For example, evaluating some specific issues (e.g., access, navigation, etc.) with disabilities requires manual assessment, where manual assessment could be from both expert and non-expert/user perspectives, because often considering only automated evaluation tools could result in inaccurate validation reports. Therefore, recently, numerous studies have been conducted focusing on various issues to improve the effectiveness of the crowdsourcing system. For instance, Songet al. [43] developed a crowdsourced-based web accessibility evaluation system that merges user and expert evaluators with an inferencing technique to produce reliable and accurate accessibility evaluation results to improve user and expert assessment processes. Their provided accessibility reports will help web designers to find accessibility issues and solutions together. Besides, Mohamad et al. [44] proposed a hybrid approach to assess accessibility of webpages, incorporating accessibility knowledge, automated tools assessment, and expert feedback to validate multiple webpages. They used a set of rules through a rule engine to perform the inferencing process and a decision support system to compute the accessibility evaluation report that makes the process effective. Furthermore, Li et al. [45] addressed that automated testing is effective, although few checkpoints/success criteria of web content accessibility guidelines require manual judgment. However, in manual judgment/testing, the main challenges are associated with a burdensome and excessive workload for the evaluator. Addressing these challenges, the authors of this paper proposed an advanced crowdsourcing-based web accessibility evaluation process that is effective for facilitating the manual testing process (e.g., user and expert testing). The proposed technique configured and simplified the evaluation system, focusing on the learning system, the task assignment system, and the task review process.

Furthermore, addressing accessibility and usability issues, Tahani Alahmadi [46] proposed a crowdsourcing system for web accessibility evaluation, considering subjective and objective measurements. The author of this study incorporated several accessibility and usability criteria, automated systems, and human incorporation to reduce the amount of effort and time for interactivity on the webpage during the evaluation process. To provide flexible and open support for different accessibility difficulties, Broccia et al. [47] presented a crowdsourcing system to evaluate webpage accessibility, including the results of automated accessibility testing and usability testing. They incorporated usability testing to validate the results qualitatively and quantitatively. Acosta-Vargas et al. [10] also proposed a heuristic method based on the user barrier computation process to increase user satisfaction, productivity, security, and effectiveness. The proposed approach was developed by incorporating UX Checker, evaluators (experts in web accessibility), and users with low vision. Another study by Martins et al. [48] noted several accessibility and usability problems associated with existing guidelines that are difficult to pinpoint using automated methods alone. Thus, they suggested a hybrid process for evaluating web accessibility that combines automatic evaluation (using the ACCESSWEB platform) and manual tasks (e.g., user and expert evaluation).

To improve the effectiveness of the task assignment process, the crowdsourcing system could be divided into tedious micro-tasks. Most crowdsourcing systems validate webpages using an expert who has high-level accessibility knowledge. They addressed that the validity and reliability of the evaluation result could be significantly improved by incorporating non-experts, as they will observe the webpages from an end-user viewpoint. Therefore, Song et al. [49] introduced a new crowdsourcing system implementing the Golden Set Strategy and Time-Based Golden Set Strategy. They incorporated automated testing for a few checkpoints and manual testing for other checkpoints. Manual evaluation was performed by non-experts, where all the associated tasks were distributed through the Golden Set Strategy, and task completion time was allocated according to the Time-Based Golden Set Strategy. The evaluation result shows that the evaluation time is reduced to half of the expert evaluation time, and improves the evaluation accuracy.

In addition, Alexander Hambley [50] addressed that automated tools have limited coverage of accessibility issues that reduce the acceptance of the accessibility evaluation result. Therefore, the author proposed an accessibility testing system incorporating pre-existing evaluation tools in terms of sampling, clustering, and developer testing. The proposed approach is effective in reducing the inaccuracy of the evaluation results.

Although these approaches are effective, some issues reduce their effectiveness. For example, since only a small subset of WCAG checkpoints are possible to evaluate using these methodologies, the results may not accurately reflect the accessibility perspective; it is challenging to implement because it takes a lot of effort, time, financial support, and empirical validation; a wide array of evaluation tools are available, though a few tools are possible to incorporate to these systems that might alter the evaluation result depending on the other evaluation tools taken into account; considering a limited number of evaluation metrics may bias the evaluated result; lack of accessible knowledge may skew the evaluation process and results because the majority of the evaluation process involves human evaluation; in some cases, they avoid expert evaluation due to several complexities that may reduce the effectiveness of the evaluation results.

2.3.1.2 Heuristic Approach (HA)

Heuristic approaches generally allow human inspection in various ways to evaluate the accessibility of the web. According to past research, several studies have focused on heuristic approaches to evaluate the effectiveness and accessibility issues of the Web platform. For example, Li et al. [11] addressed that a certain number of checkpoints during the evaluation of web accessibility require human inspection, such as volunteer participation or expert opinion. Due to the lower level of expertise and inappropriate knowledge, the evaluation task might seem complicated and return poor evaluation results. To address this issue, they proposed a heuristic approach considering a task assignment strategy called Evaluator-Decision-Based Assignment (EDBA) to enhance the selection of participants and experts by using evaluators' prior evaluation records and knowledge of their areas of competence.

Furthermore, Giraud et al. [51] argued that adherence to accessibility standards does not guarantee that a website is fully accessible to users with disabilities, particularly users with blindness. They noted that usability standards are crucial to making a webpage accessible to users with disabilities. One critical criterion for improving the usability of websites is redundant information filtering. They proposed a heuristic approach that focuses on redundant and irrelevant information filtering, including participants with blindness. They found that eliminating redundant information and information filtering enhances webpage accessibility, user satisfaction, and navigation performance.

Furthermore, Mazalu and Cechich [52] highlighted that it is important to consider the requirements of both the developer and the end user to encourage accessibility support for people with disabilities. To evaluate the intelligent feature specifically for users with visual impairment, they proposed a web accessibility assessment methodology that incorporates a multi-agent system. The proposed system was validated through several assessment metrics and statistical results. Although these approaches are effective, several factors, including a limited number of assessment features, guideline implementation, cost-associated difficulties, and the expert assessment process, reduced the effectiveness of the outcome of these proposed systems.

2.3.2 Automated Evaluation Approaches

The W3C has offered a list of automated web accessibility testing tools in recent years. Unfortunately, these tools are not frequently updated to use the most recent version of accessibility guidelines to keep up with the latest technology [53]. For example, every day, mobile browsing is growing it's popularity, but in some cases, due to the technological design and development process, they are not usable or viewable on different devices and screen resolutions. In general, automated web accessibility testing refers to code-oriented evaluation to validate the accessibility features of webpage content through computer programs against accessibility guidelines. In other words, automatic testing is a process of automatically executing a set of tasks to validate a set of patterns of webpages. The importance of automated accessibility testing has increased in recent times as it reduces testing time, minimizes associated cost, and speeds up the testing process [44]. Besides, it allows one to test multiple webpages without any issues. Focusing on these advantages, several research works have been conducted to develop automated accessibility testing tools or methods in the context of webpage evaluation that are classified into three categories: (i) declarative model, (ii) ontological model, and (iii) algorithmic evaluation. All of these evaluation categories are described in detail in the following sections.

2.3.2.1 Declarative Model (DM)

Despite having several advantages of automated accessibility testing, a few researchers claimed that existing automated testing processes have several limitations. For example, most of the existing automated accessibility testing tools do not support browser plugins and require the installation of multiple packages, making the evaluation process challenging and discouraging users from utilizing them. Boyalakuntla et al. [54] developed an automated accessibility evaluation tool to assist with web accessibility testing through command-line and browser plug-in facilities to address these challenges. It supports WCAG 2.1 and WCAG 2.2, focusing on ARIA, color contrast, Hyper Text Markup Language (HTML) checking, and interaction-related issues. It displays a list of errors and repair suggestions with a snippet of code. Although the proposed approach is effective, some issues limit its effectiveness as they assess webpages in terms of 16 success criteria of WCAG 2.1, and 2.2, even though additional success criteria must be implemented or verified to represent the full picture of the accessibility situation; they do not compute an overall accessibility score that might not accurately reflect the accessibility situation of the tested webpage. Besides, Jens Pelzetter [55] addressed how the vagueness of accessibility requirements causes several anomalies in the results of automated accessibility testing. This author proposed a declarative model to evaluate the accessibility status of websites by incorporating small test sets, evaluating considering the Accessibility Conformance Testing (ACT) rule set, and ontology modeling. Although the proposed system is capable, certain potential issues have been observed that could restrict the effectiveness of the evaluation result. For example, ontology modeling introduces ambiguity during the testing process, which reduces the effectiveness of the result. Also, implementing ACT rules is quite difficult as it requires resources and experience, which may not be convenient for practitioners.

Furthermore, several previous studies have asserted that accessibility also depends on the visual complexity of a webpage, although it is barely considered in the literature [56]. In a recent study, Rojbi et al. [57] addressed the fact that most of the visual elements are not accessible to end users, which marked accessibility of the web as a challenging task. Others asserted that understanding visual complexity is an emerging requirement for web accessibility evaluation, although many automated tools do not consider it due to its associated weaknesses, problems, or difficulties. For example, as picture descriptions are written manually or generated automatically, they may not be appropriate or suitable for understanding for people with disabilities. Especially, it might be difficult for people with vision impairment to interpret the content of an image due to an incorrect image description. Taking this into account, Michailidou et al. [58] proposed an automated tool to assess the visual complexity of the content and generate the Visual Complexity Score (VCS) based on common aspects of an HTML Document Object Model (DOM) to predict and visualize the complexity of the webpage in the form of a pixelated heat map. Considering the same issue, another example was given by Duarte et al. [59], where they also used web elements to measure their level of accessibility from visual perspectives. They used two specific features, such as menu and list elements, to evaluate according to their defined role in order to check whether their identifiable role can be determined to define their accessibility. In another study, Doush et al. [60] presented an approach to predict whether the given webpage is accessible in terms of specified Rich Internet Applications (RIAs), as it has a great effect on improving the accessibility of visual components. Also, Raju Shrestha [61] proposed a neural network framework for automatic evaluation of image descriptions according to the principles of the National Center for Accessible Media (NCAM). To increase the effectiveness of the proposed framework, the author also incorporated expert knowledge (people who understand image accessibility) and the universal design process. Although these two proposed systems can accurately predict accessibility issues, especially for those with visual impairments, a few issues reduce the effectiveness of the suggested solution, such as evaluating images for visual complexity, the proposed systems consider a small number of assessment features, incorporate a small number of guidelines and checkpoints, and do not compute the overall accessibility score to indicate the accessibility status of the evaluated component. Besides, accessibility-related studies on the web are not limited to considering only visual perspectives. In particular, in a recent work, Duarte et al. [62] proposed an automated approach that helps to understand the semantic meaning of web elements and their correlation with their provided textual format. It takes visual components, particularly images, and evaluates them to determine the similarity between visual components and their textual description according to web content accessibility guidelines. This could allow us to understand how complicated the presented component is for the user.

Research also found related to automatic measurement of accessibility issues of webpages of different aspects and dimensions [60][63][64]. Most studies presented their approaches considering several webpage elements, such as RIAs visual elements like links, images, menus, buttons, dropdown menus, etc. To evaluate this webpage's elements, most approaches typically use HTML source code, which is converted into a DOM structure to evaluate web features to measure accessibility issues or webpage complexity. Although another study conducted by Freire and Fortes [65] presented an automated approach for dynamic webpage evaluation considering Extensible Markup Language (XML) and Extensible Stylesheet Language Transformations (XSLT) scripting documents.

2.3.2.2 Ontological Model (OM)

According to several studies, the process of color-intensive webpage designing can alter the webpage accessibility for people with color impairment, such as the perception of color differentiation. In order to increase accessibility and interaction of people with color vision deficiency (CVD) with the Web, Bonacin et al. [66] designed an ontology-based framework for the adaptive interface development. Using this method, it is possible to identify the ideal recoloring interface for CVD users based on their preferences. Additionally, Robal et al. [67] addressed that the user interface of webpages should be developed with the requirements of end users in mind so that users can navigate the structure easily and smoothly and understand the information being shown to them. To ensure these aspects, they developed an ontology-based automated evaluation of the user interface (UI) of the webpages to determine the accessibility and usability of the UI.

Furthermore, to assess the accessibility of webpages, Hilera et al. [68] designed a universal architecture focusing on web services and semantic web technologies. They incorporated multiple evaluation tools and generated results according to the semantic similarity of multiple reports obtained by each tool. Similarly, Ingavélez-Guerra et al. [69] provided a strategy based on ontology and knowledge modeling that supports accessibility analysis and evaluation of learning objects, highlighting the relevance of knowledge representation about learning objects with a focus on WCAG.

These developments are effective, although a few difficulties made these improvements less successful, such as frequent updating and adding new guidelines, which are arduous processes in ontology-based solutions that practitioners may not want to employ. Without a professional review of the validated result, the evaluation results may be misleading, and the user may not find them fully acceptable.

2.3.2.3 Algorithmic Evaluation (AE)

Recently, a few researchers and academics have expressed concerns about the quality of webpages, which requires extra attention to satisfy the demands of end users. They claimed that the quality of the webpages also indicates how accessible a webpage is. With this in mind, Rashida et al. [9] presented an automatic evaluation process by presenting three algorithms for the content evaluation, loading time, and overall performance attributes that are generally ignored in many approaches. The results of the experiments demonstrate the usefulness of the suggested automated tool. In addition, to tackling the same problem, Abdullah Alsaeedi [70] presented an algorithmic evaluation framework that would incorporate different automated accessibility testing tools to evaluate the accessibility of a webpage. It enables the selection of sets of evaluation tools prior to the test and compares the accessibility status of the old and new versions of a given webpage.

Furthermore, in relation to the semanticity of the web content, Duarte et al. [59] reported that current automated web accessibility testing tools are unable to evaluate rules and techniques semantically and incorrectly evaluate the web content. To mitigate this problem, they proposed an automated tool that determines the similarity between content and its textual description from the perspective of web content accessibility guidelines. The semantic similarity measurement is performed using the SCREW algorithm to measure similarity between a set of textual descriptions of web content. They represent the accessibility of web content in terms of their computed similarity score.

Algorithmic evaluation is effective in automated accessibility evaluation, although the mentioned methodologies have several limitations that restrict their usefulness and applicability. For example, assessment features are very few, and other features must be taken into account in the evaluation process. To validate the algorithmic solution or the evaluation result, a wide array of user and expert interventions is required to validate the evaluated results and improve the acceptability of the end user.

2.4 Observed limitations in the existing solutions

Recently, researcher contributed their effort to develop an effective approach to validate the web, identify accessibility issues, and compute accessibility barrier scores. Among several approaches, hybrid approaches and several automated web accessibility testing tools have been developed to provide interactive accessibility reports about the tested webpages. Referring to the solutions developed in the past literature (Section 2.2), studies related to hybrid evaluation approaches were classified into Crowdsourcing System (CS) and Heuristic Approach (HA), where CS can be grouped into nine major drawbacks, and HA can be grouped into four drawbacks. Besides, automated evaluation approaches are classified into Declarative Model (DM), Ontological Model (OM), and Algorithmic Evaluation (AE), where DM can be grouped into seven main drawbacks, OM and AE can be grouped into four drawbacks, respectively, as presented in Table 2.1. Taking into account Table 2.1, it can be concluded that the existing solutions have several disadvantages that make the developed techniques less effective. For example, in the context of hybrid evaluation, specifically for the crowdsourcing system, findings reported that the crowdsourcing system has several issues with user and expert assessment, consideration of accessibility and usability criteria, task distribution, assessment time minimization, and cost reduction that might hamper the evaluation process and limit the advancements of the developed crowdsourcing system. Besides, a few drawbacks to heuristic approaches have been identified, including limited evaluation checkpoints and assessment features, cost-sensitiveness, and lack of an expert assessment process. Most proposed solutions only considered the identification of errors or highlighted the violated guidelines. The effectiveness of the developed methods was constrained by not considering the computation of the overall accessibility score. Most works focused on all types of disabilities. In some works, only people with vision impairments were considered.

TABLE 2.1: Categorical Overview of Selected Papers for Review

References	Identified drawbacks
Hybrid evaluation approaches	
Crowdsourcing System (CS)	
Song et al. [43]	(CS_1). A limited number of checkpoints
Song et al. [43] Mohamad et al. [44] Li et al. [45] Tahani Alahmadi [46] Giovanna et al. [47] Acosta-Vargas et al. [10]	(CS_2). Not time efficient/time-consuming
Mohamad et al. [44]	(CS_3). Absence of empirical validation
Acosta-Vargas et al. [10] Li et al. [11]	(CS_4). Cost-sensitive
Tahani Alahmadi [46] Hambley [50]	(CS_5). Difficult to perceive
Giovanna et al. [47] Martins et al. [48]	(CS_6). A limited number of evaluation tools have incorporated
Song et al. [49]	(CS_7). Evaluation results could be inappropriate due to the lack of accessibility knowledge (CS_8). A limited number of evaluation metrics might bias the performance
Heuristic Approach (HA)	
Li et al. [11]	(HA_1). Not cost-efficient (HA_2). A limited number of evaluation checkpoints
Giraud et al. [51]	(HA_3). A limited number of assessment feature
Mazalu and Cechich [52]	(HA_4). Expert opinion/assessment required
Automated evaluation approaches	
Declarative Model (DM)	
Boyalakuntla et al. [54] Shrestha [61]	(DM_1). Evaluate against a limited number of guidelines (DM_2).Absence of accessibility score computation
Jens Pelzetter [55]	(DM_3). Ontology modeling introduces ambiguity (DM_4). Implement guidelines that are not a standard guideline (DM_5). Lack of consideration of user and expert evaluation
Michailidou et al. [58]	(DM_6). A limited number of assessment features
Shrestha [61]	(DM_7). Limited checkpoints considered
Ontological Model (OM)	
Bonacin et al. [66] Robal et al. [67] Ingavélez-Guerra et al. [69]	(OM_1). Expert opinion/assessment is required
Robal et al. [67]	(OM_2). Update the ontology is challenging and requires a level of expertise (OM_3). Adding new guidelines is a complex and laborious task
Hilera and Timbi-Sisalima [68]	(OM_4). Observed inconsistency in the knowledge base and database
Algorithmic Evaluation (AE)	
Rashida et al. [9] Duarte et al. [59]	(AE_1). A limited number of assessment features (AE_2). No expert evaluation
Abdullah Alsaeedi [70]	(AE_3). A limited number of tools have incorporated (AE_4). Missing possible web accessibility violations and the corresponding error messages generation function

In contrast, in automated evaluation, more specifically in the declarative model, it was noted that consideration of a limited number of guidelines and checkpoints, fewer assessment features, and a lack of consideration of user and expert evaluation for accessibility assessment and validation are factors that make these processes less effective. Also, a lack of attention has been observed in most of the tools to what samples or objects of the webpage they have evaluated and what criteria they have used to assess those webpage objects. Besides, for the ontological model, the main limitations were associated with a lack of consideration of expert opinion/assessment, ontology updating issues for adding new guidelines, which are complex, laborious, and challenging tasks for the developer, and inconsistency in the knowledge base and database that might alter the evaluated results. For the algorithmic evaluation process, a limited number of assessment features, a lack of user and expert evaluation, and limited statistics for accessibility error computation are the primary factors in reducing their effectiveness.

However, these developed approaches provide interactive accessibility reports about the tested webpage by incorporating some specific attributes of webpages according to the Web Content Accessibility Guidelines (WCAG) to evaluate their accessibility status. Though the latest version of WCAG 2.2 is a complete guideline for accessibility features, it has limited consideration about some issues with people with disabilities such as whether the webpage is active or deactivated, webpage has a manual text size adjustment option, manual font family adjustment option, manual color adjustment option, user information requirement, Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) issues, usefulness of internal/external links, used images, inserted video and audio content. As these features are not directly possible to evaluate automatically, WCAG does not provide a clear indication about these aspects, and without considering these aspects, it is not possible to develop an effective automated web content accessibility evaluation tool.

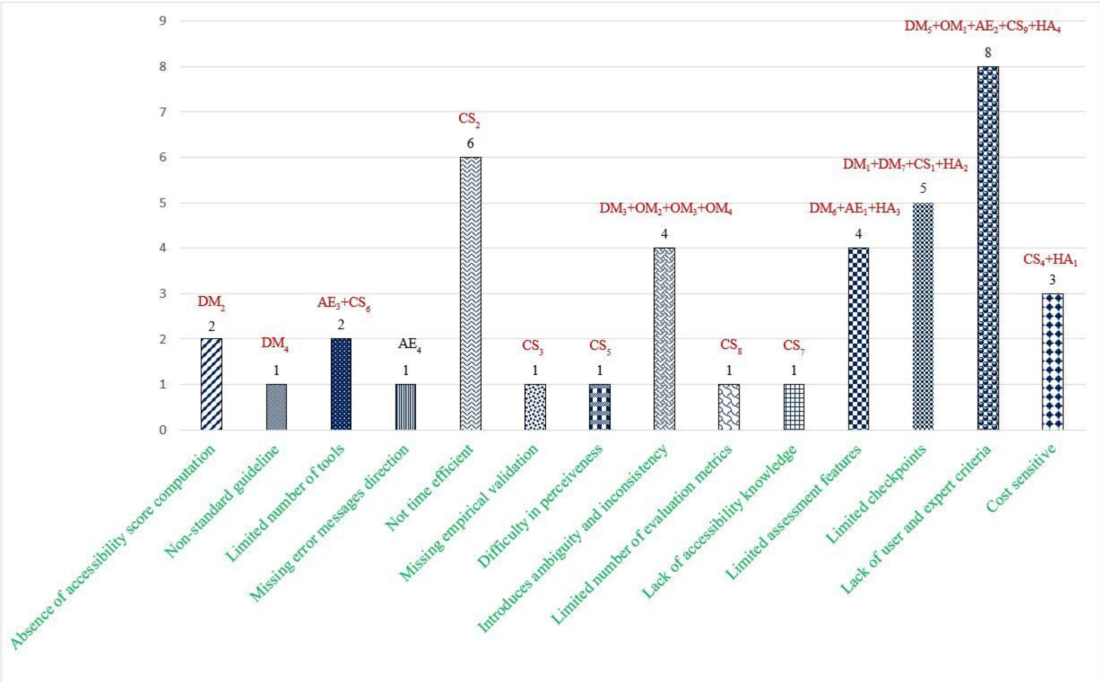


Figure 2.4: Number of reviewed studies on each identified limitation or challenge

According to the observation, Figure 2.4 represents the number of studies on each identified limitation of the reviewed studies from existing literature. This figure depicts that several issues appeared in the existing approaches that make the evaluation results ineffective, which addresses a further concern of the web researcher. In summary, to answer the first research question (R-Q1), the observation result concludes that the following drawbacks and challenges have been observed frequently in multiple developed solutions focusing on hybrid and automated evaluation that reduce their effectiveness:

Drawbacks in Hybrid Evaluation:

- Lack of consideration of user and expert criteria,
- Lack of consideration of accessibility and usability criteria,
- Most of the solutions are not cost and time-efficient,
- Limited checkpoints and assessment feature consideration,
- Difficulties in the task distribution process.

Drawbacks in Automated Evaluation:

- Difficulties in guideline (WCAG) understanding that are written in natural language format,
- Consideration of only a limited number of WCAG success criteria and web features,
- Insufficient attention is given to user behavior, user requirements, and expert suggestions,
- Challenges in mapping process among success criteria and webpage features,
- Semantic features of webpages and related engineering techniques are not given enough attention,
- The process of accessibility evaluation and score visualization is ambiguous, thus, it is difficult to determine which criteria have been looked into and which have been skipped and
- Terminologies used in assessments are ambiguous and do not accurately reflect their intended meaning.

2.5 Conclusion

In this Chapter, I attempted to take a small step toward contributing to web accessibility research by evaluating and determining the vulnerabilities or drawbacks of the existing solution to address a new direction for future developments. The analysis of this chapter presented several existing hybrid and automatic web accessibility testing and evaluation processes of the focused area of research in the last decade for ensuring the inclusion of accessible web content. Moreover, I analyzed several studies from the existing literature collected from several scientific databases using the developed list of keywords. According to the observation, inappropriate guideline selection, ambiguities in guideline understanding, avoiding user and expert suggestions as evaluation criteria, limited consideration of semantic perspectives, and unwillingness to incorporating updated engineering methods are major concerns for upcoming development to improve the performance and effectiveness of the developed solutions to provide reliable reports of the tested webpages.

Chapter 3

Hybrid Web Content Accessibility Evaluation Methods

In the last few decades, researchers from different countries (US, UK, etc.) have performed accessibility research on the web platform and suggested that a significant number of official and private portals have accessibility issues. Most countries have no strict rules or monitoring systems to control these inaccessible webpages that create massive barriers for people with disabilities. Researchers emphasized conducting comprehensive research on webpages to validate their accessibility. Several organizations have shown their impressive effort in the hybrid evaluation process. However, several limitations have been addressed in the current available hybrid web accessibility testing process, as mentioned in Chapter 2. Therefore, this chapter aims to present three new distinct hybrid web content accessibility evaluation approaches for improving the web accessibility evaluation process to answer the second research question as follows:

R-Q2: How can we improve the effectiveness of the hybrid web content accessibility testing process to address its current limitations?

According to the discussion of the earlier Chapter 2, crowdsourcing and heuristic approaches are the two main branches of the hybrid testing or evaluation process. Crowdsourcing approaches generally allow for the integration of both automated testing (incorporating automated tools) and human evaluation (performed by hiring people, including experts and users) to evaluate the accessibility of web. Besides, heuristic approaches generally enable only human inspection in various ways to evaluate the accessibility of the web. From our empirical analysis and personal observation, crowdsourcing approaches are effective compared to heuristic approaches, as heuristic approaches only allow humans to evaluate, which limits the evaluation process to a certain number of checkpoints, and assessment features require high cost, and might not be time-effective solution. Therefore, for the hybrid evaluation process, this thesis focuses on crowdsourcing approaches to facilitate the web content accessibility evaluation.

Generally, existing crowdsourcing approaches are not free of limitations, which is addressed in Chapter

2. Most of the existing crowdsourcing approaches are limited to the incorporation of i) lack of user and expert assessment, ii) consideration of a limited number of accessibility and usability criteria, iii) difficulty in the task distribution process, iv) not time effective, and v) not cost-effective that might hamper the evaluation process and limit the advancement of their developed crowdsourcing system. Addressing those issues, this chapter aims to present three new crowdsourcing evaluation methods for the web content accessibility evaluation process that can help to improve the existing limitations and facilitate the evaluation process. The proposed crowdsourcing evaluation methods are (i) Integrated approach, (ii) Variable magnitude approach, and (iii) Machine Learning approach. All of these proposed approaches are described in the following sections.

3.1 Integrated Approach

Generally, the objective of the proposed integrated approach is to integrate both automated and human evaluation, considering the implementation of multiple automated tools with user and expert assessment, focusing on accessibility and usability criteria. As most existing approaches have focused automated tools without considering user opinion and expert feedback, integrating both aspects might improve the evaluation results. Besides, selecting multiple automated web accessibility testing tools is a dynamic procedure, it can be changed according to the nature of the investigation. Thus, appropriate automated tool selection might facilitate the overall evaluation results. The overall goal of this approach is to propose a new integrated approach to evaluate and identify the frequent accessibility problems that might help webpage owners identify the shortcomings of their webpages. The proposed approach is divided into two steps: in step 1, evaluation is performed through multiple automated web accessibility testing tools, and in step 2, evaluation is performed through human observation through questionnaires (such as system usability testing and expert testing). Figure 3.1 shows the workflow diagram of the proposed integrated approach for evaluating web content accessibility.

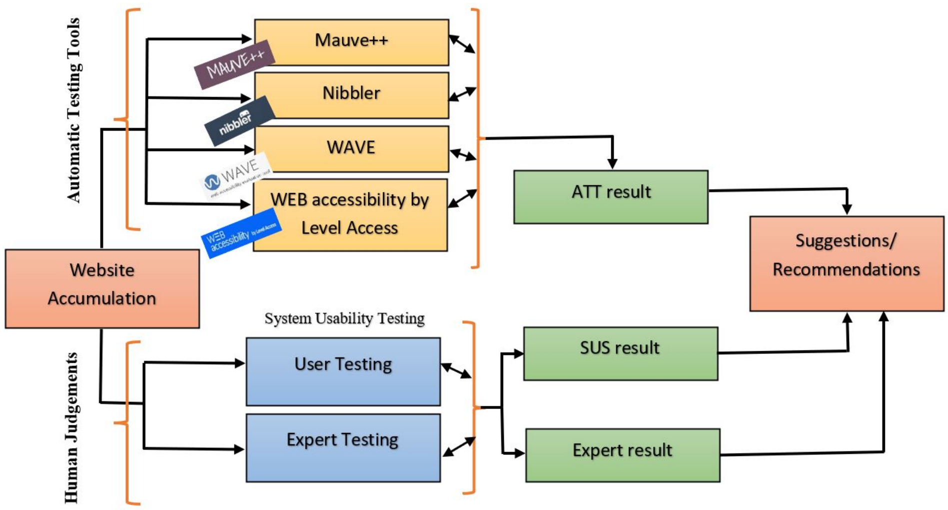


FIGURE 3.1: Workflow diagram of the proposed integrated approach

3.1.1 Webpage Accumulation

Webpage accumulation is the process of collecting webpages according to a specific group or targeted domain for performing the evaluation. To perform the integrated evaluation, webpage accumulation is the first step to consider, as shown in Figure 3.1.

3.1.2 Automatic Accessibility Testing Protocols

3.1.2.1 Mauve++ Tool

Multiguideline Accessibility Usability Validation Environment (MAUVE) is an environment for web content accessibility evaluation according to WCAG standards [71]. The researchers of HIIS Lab developed and provided it to the community as an open-access testing environment. It helps to determine webpage accessibility through WCAG standards. It focuses on three conformance levels: A, AA, and AAA to test webpages. MAUVE++ accessibility testing tool checks against all three conformance level requirements during the evaluation process of webpages. MAUVE++ allows running the software in a browser plugin, desktop, smartphone, iPad, tablet, etc. It is possible to download the result in JavaScript Object Notation (JSON) format. It generates the report through the total number of successful checkpoints, warning checkpoints, and erroneous checkpoints. The overall accessibility percentage generated is based on the ratio of the number of checkpoints passed and the number of checkpoints tested. Generally, it provides errors based on HTML and Cascading Style Sheets (CSS) rules violations according to the WCAG guidelines. It can be tested on both single-page and dynamic webpages simultaneously without interruption. Figure A.1 (Appendix A) shows the web accessibility result of the example tested webpage through the Mauve++ testing environment. The detailed result is shown in Section 3.1.4.1.

3.1.2.2 Nibbler Tool

Nibbler is a free webpage accessibility testing tool that accesses the webpage Uniform Resource Locator (URL) and generates the accessibility result [72]. In other words, Nibbler is a 'bot' or 'automated computer program' that works by looking at the domain name to find the linked webpages. It calculates the overall accessibility score on a 10-point scale based on four key prototypes (accessibility, experience, marketing, and technology). Each prototype analyzes several aspects such as code quality, headings, internal links, mobiles, page titles, URL format, amount of content, Facebook page, freshness, images, popularity, printability, server behavior, Twitter, analytics, domain age, incoming links, meta tags, and social interest to determine the overall accessibility score. Figure A.2 (Appendix A) shows the screenshot of the Nibbler tool for the tested webpage. The detailed result is shown in Section 3.1.4.2.

3.1.2.3 WAVE Tool

WAVE is a free, open-access, single-page web accessibility checker [73]. It is performs by checking the number of errors, contrast errors, alerts, features, structural element issues, Accessible Rich Internet Application (ARIA), and text size in terms of normal and large contexts. WebAIM developed WAVE for both webpage and plugin versions. It follows all the WCAG guidelines, including WCAG 1.0, WCAG 2.0, and WCAG 2.1 standards. Figure A.3 (Appendix A) shows the view of the WAVE test result for the tested webpage. It summarizes the entire problem and all the violations in detail and provides potential solutions via the source code as shown in Figure A.3 (Appendix A). The detailed result is shown in Section 3.1.4.3.

3.1.2.4 WEB Accessibility Tool

WEB accessibility is another popular webpage accessibility testing tool that highlights the graphical view of the original webpage [74]. It is free, open-source software, and publicly available for the entire browser (Chrome, Firefox, etc.) as a testing medium. It calculates accessibility percentage through compliance score, number of identified violations, and number of automated tests, and suggests the additional tests that should be carried out. Additionally, it provides details about identified violations concerning their severity level. Figure A.4 (Appendix A) shows the result of the tested webpage by the WEB accessibility tool environment. The detailed result is shown in Section 3.1.4.4.

3.1.3 Questionnaire-Based Human Observation

According to the past research [19][75][76], automatic web accessibility testing tools are effective. Though automatic tools have highly effective measurements, several issues remain unrevealed by these tools. Thus, human observation is performed through two testing systems: User testing and expert testing to validate the accessibility issues, whereas user testing is conducted through system usability testing.

3.1.3.1 User Testing

User testing is the process of human testing, incorporating end users who have sufficient knowledge about internet platforms and engage with the online platform frequently. User testing is performed through the System Usability Scale (SUS), as through this system it is possible to understand the effectiveness of the webpages for a particular group of people [77]. In this work, user testing is conducted by inviting university students. They have sufficient internet browsing knowledge and are familiar with webpage navigation. The system usability measuring scale was defined from 1 to 5, where 1 is Strongly Disagree and 5 is Strongly Agree, to provide usability feedback. Table 3.1 shows an example of a questionnaire-based SUS format used in this research. The detailed result is shown in Section 3.1.4.5.

TABLE 3.1: System Usability testing questionnaire with evaluation scale

Questions	Scale
1. I think that I would like to use this webpage frequently	1/2/3/4/5
2. I found this webpage unnecessarily complex.	1/2/3/4/5
3. I thought this webpage was easy to use.	1/2/3/4/5
4. I think that I would need assistance to be able to use this webpage.	1/2/3/4/5
5. I found the various functions in this webpage were well integrated.	1/2/3/4/5
6. I thought there was too much inconsistency on this webpage.	1/2/3/4/5
7. I would imagine that most people would learn to use this webpage very quickly	1/2/3/4/5
8. I found this webpage very cumbersome/awkward to use.	1/2/3/4/5
9. I felt very confident using this webpage.	1/2/3/4/5
10. I needed to learn a lot of things before I could get going with this webpage.	1/2/3/4/5

3.1.3.2 Expert Testing

Expert testing is the process of professional testing with sufficient scientific knowledge of the web accessibility domain and criteria to identify the existing problems of a particular webpage and make recommendations to upgrade the inaccessible functions. Twelve questions were considered with "Yes/No" answering option to provide expert responses according to their observation. Table 3.2 shows an example of a questionnaire-based expert testing format used in this research. The detailed result is shown in Section 3.1.4.5.

TABLE 3.2: Expert testing questionnaire with evaluation scale

Questions	Scale
1. Is the webpage accessible for blind people/low vision people?	Yes/No
2. Is the webpage accessible for a person with a moving disability?	Yes/No
3. Is the link accessible or available?	Yes/No
4. Does the webpage information have a color deficiency?	Yes/No
5. Does the webpage require CAPTCHA?	Yes/No
6. Is the webpage accessible through a keyboard?	Yes/No
7. Is the webpage an English version webpage?	Yes/No
8. Does the webpage have a native language version?	Yes/No
9. Is the webpage responsive?	Yes/No
10. Does the webpage have inaccessible features?	Yes/No
11. Are links and buttons separable?	Yes/No
12. Does it allow manual font size adjustment?	Yes/No

3.1.4 Experimentation with Computed Results

To validate the proposed integrated method and perform the experimentation, government webpages related to the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) have been chosen from different countries across Europe and Asia to understand their accessibility for people with disabilities. All the 21 tested webpages URLs are given in Table B.1 (Appendix B). According to Figure 3.1, four automatic web accessibility testing tools (Mauve++, Nibbler, WAVE, and WEB accessibility) were employed, and user testing with expert testing through a questionnaire was incorporated. The details of the experiment results are described in the following sections.

3.1.4.1 Result of Mauve++ Test Tool

To evaluate the accessibility of the selected webpages, the Mauve++ test is employed based on WCAG 2.0 and WCAG 2.1 standards according to the POUR principles (P-Perceivable, O-Operable, U-Understandable, and R-Robust). Based on the error rate, the overall performance is calculated for conformance level AAA, as shown in Table 3.3.

TABLE 3.3: Mauve test: Government COVID information webpages accessibility testing result for WCAG 2.0 and WCAG 2.1

Webpage Info		**WCAG 2.0 (Level-AAA)**						**WCAG 2.1 (Level-AAA)**					
		****Per=Performance**											
		****High=H; **Medium=M; **Low=L**											
Country	**W-ID**	**P**	**O**	**U**	**R**	**Errors**	**Per**	**P**	**O**	**U**	**R**	**Errors**	**Per**
EU	WID-1.0	3	1	1	0	5 (0.05)	H	9	1	1	0	11 (0.011)	L
EU	WID-2.0	3	1	2	1	7 (0.07)	M	7	1	1	2	11 (0.011)	L
EU	WID-3.0	3	1	2	1	7 (0.07)	M	8	1	1	1	11 (0.011)	L
EU	WID-4.0	3	2	1	1	7 (0.07)	M	5	–	0	1	6 (0.06)	M
EU	WID-4.1	4	3	0	0	7 (0.07)	M	10	2	0	1	13 (0.013)	L
EU	WID-5.0	4	1	2	1	8 (0.08)	M	12	1	0	1	14 (0.014)	L
EU	WID-6.0	4	1	0	0	5 (0.05)	H	9	0	0	0	9 (0.09)	M
EU	WID-7.0	5	0	1	2	8 (0.08)	M	11	0	0	1	12 (0.012)	L
EU	WID-8.0	2	1	0	1	4 (0.04)	H	3	1	0	0	4 (0.04)	H
EU	WID-9.0	3	0	0	0	3 (0.03)	H	7	0	0	0	7 (0.07)	M
EU	WID-10.0	6	2	1	2	11 (0.011)	L	13	1	2	1	17 (0.017)	L
EU	WID-11.0	8	1	3	0	12 (0.012)	L	13	1	2	0	16 (0.016)	L
EU	WID-12.0	–	–	–	–	–	–	–	–	–	–	–	–
Asia	WID-13.0	8	1	2	1	12 (0.012)	L	10	2	1	0	13 (0.013)	L
Asia	WID-14.0	9	3	1	1	14 (0.014)	L	8	2	1	3	14 (0.014)	L
Asia	WID-15.0	6	2	2	1	11 (0.011)	L	6	1	2	1	10 (0.010)	L
Asia	WID-16.0	3	1	1	1	6 (0.06)	M	5	1	2	1	9 (0.09)	M
Asia	WID-17.0	8	2	2	1	13 (0.013)	L	8	2	1	1	12 (0.012)	L
Asia	WID-18.0	7	2	1	1	11 (0.011)	L	7	1	1	2	11 (0.011)	L
Asia	WID-19.0	0	0	0	0	0 (0.0)	H	0	0	0	0	0 (0.0)	H
Asia	WID-20.0	2	0	2	2	6 (0.06)	M	3	0	1	2	6 (0.06)	M

To evaluate the performance or webpage quality according to accessibility testing results, the webpage accessibility analysis results are classified into three categories: High, Medium, and Low. The error rate (%) is calculated from the total number of errors found during the test. Generally, fewer errors represent a high-performance webpage, and a higher number of errors represents a low-performance webpage. Therefore, to evaluate the error rate (%), a significance level is taken into consideration. From the empirical observation and the researcher's opinion, the significance level ξ=0.05 and ξ=0.09 is considered to evaluate the webpage performance. Those webpages are classified under $\xi \leqq 0.05$, referred to as 'High' quality webpages. If webpages are classified under ξ >0.05 to $\leqq$0.09, considered them as 'Medium' quality webpages. Furthermore, under ξ >0.09 significance level, webpages are considered 'Low' quality webpages. Table 3.3 shows that, according to WCAG 2.0 guidelines, among 21 webpages, only 5 webpages were classified under significance level ($\xi \leqq$0.05). These 5 are High-quality webpages according to WCAG 2.0 guidelines. After evaluating the result under ξ >0.05 to $\leqq$0.09 significance level, 8 webpages were medium-quality, and 7 were Low-quality webpages. For WCAG 2.1 guidelines, only 2 webpages were retrieved under the significance level ξ and marked as High-performance webpages. Besides, 5 webpages are classified as medium-quality, and 13 as low-quality webpages. Additionally, only 4% webpages (1 out of 21) passed the MAUVE++ test without any error according to WCAG 2.0 and WCAG 2.1 guidelines. For WCAG 2.0 standards, the remaining webpages failed with an average of 7 errors. Moreover, for WCAG 2.1 standards, all the webpages failed with an average error of 9. However, for the WCAG 2.1 guideline, average errors were comparatively higher than the WCAG 2.0 guideline. In both WCAG 2.0 and WCAG 2.1 guidelines, 1 webpage was unavailable (WID-12).

3.1.4.2 Result of Nibbler Tool

Nibbler is an automatic web accessibility testing tool. Table 3.4 shows the Nibbler test results of the selected webpages. The pages were evaluated according to their overall score. The evaluation falls into three assessment categories: High, Medium, and Low. Webpages with an overall score between 10 and 9.50 are considered high-quality webpages. If the page scores between 9.49 and 9.0, they are classified as a medium-quality webpages. Pages with an accessibility score between 8.99 and 0 are considered low-quality webpages. One webpage achieved a 9.70 accessibility score (WID-11.0), indicating a high-quality webpage. 3 webpages were determined to be medium quality webpages, and 17 webpages were low accessibility webpages. While testing, the Nibbler tool generated the test report for all the webpages.

TABLE 3.4: Nibbler test result for the tested Government COVID information webpages

****AS=Accessibility score**							
W-ID	**Overall score**	**Errors**	**AS**	**Experience**	**Marketing**	**Technology**	**Performance**
WID-1.0	9.10	8	9.4	8.6	7.5	9.6	Medium
WID-2.0	8.00	13	9.8	7.7	5.7	8.4	Low
WID-3.0	9.00	9	9.7	8.0	6.3	9.5	Medium
WID-4.0	8.70	19	9.6	8.8	9.4	8.0	Low
WID-4.1	8.70	19	9.6	8.8	9.4	8.0	Low
WID-5.0	8.70	12	9.7	8.1	7.1	8.4	Low
WID-6.0	6.90	20	7.3	5.1	5.5	7.1	Low
WID-7.0	8.60	10	9.7	7.4	5.8	9.0	Low
WID-8.0	8.90	13	9.4	8.1	7.4	9.2	Low
WID-9.0	8.70	16	9.9	9.3	8.5	9.2	Low
WID-10.0	9.0	15	7.8	8.1	9.7	8.7	Medium
WID-11.0	9.70	11	9.6	9.8	10	9.6	High
WID-12.0	8.50	18	9.0	9.2	8.7	9.2	Low
WID-13.0	8.60	26	8.5	9.4	9.2	8.5	Low
WID-14.0	8.70	17	9.1	9.6	8.8	8.6	Low
WID-15.0	7.80	15	7.5	7.6	6.4	8.0	Low
WID-16.0	8.50	17	9.5	8.2	6.7	8.2	Low
WID-17.0	8.0	23	8.6	8.1	6.8	7.9	Low
WID-18.0	7.90	19	8.9	7.1	5.6	8.1	Low
WID-19.0	8.10	11	8.3	8.1	6.0	8.6	Low
WID-20.0	8.90	19	9.6	8.7	9.2	8.3	Low

3.1.4.3 Result of WAVE Tool

An automatic web accessibility testing tool named 'WAVE' is employed to evaluate the accessibility and quality of webpages. WAVE evaluates the webpages based on six categories: Features, Errors, Alerts, Contrast errors, Structural elements, and ARIA. After experimenting, the analysis result revealed that (Table 3.5), among 21 webpages, only 3 webpages (WID-1.0, WID-5.0, WID-19.0) passed the accessibility test. The medium performance webpage has the lowest number of errors, 2 (WID-6.0, WID-8.0, and WID-20.0). Among 21 webpages, 6 have fewer than the average number of errors, 6. Four webpages have 6 to 11 errors, and five (WID-14.0, WID-13.0, WID-11.0, WID-18.0, and WID-15.0) have 25, 29, 31, 47, and 59 errors, respectively. Considering the contrast error result, sometimes significant contrast errors might affect content viewing and reading. The majority of the webpages (71%) have contrast errors, and the maximum contrast error is 100 found for WID-14.0. Some webpages are free from contrast errors, which are around 28%.

Table 3.5: Accessibility testing result of the selected Government COVID information webpages for WAVE tool

**SE=Structural Elements							
W-ID	**Features**	**Errors**	**Contrast Errors**	**Alerts**	**SE**	**ARIA**	**Performance**
WID-1.0	16	0	5	16	37	120	P
WID-2.0	3	3	3	16	25	108	F
WID-3.0	2	4	8	31	52	39	F
WID-4.0	4	3	3	2	22	16	F
WID-4.1	15	6	0	14	43	44	F
WID-5.0	48	0	2	6	36	18	P
WID-6.0	34	2	1	19	44	31	F
WID-7.0	16	4	27	14	41	49	F
WID-8.0	9	1	0	4	7	3	F
WID-9.0	23	4	0	31	20	3	F
WID-10.0	4	4	0	20	66	32	F
WID-11.0	16	31	16	19	35	29	F
WID-12.0	78	6	0	189	244	318	F
WID-13.0	29	29	15	473	52	26	F
WID-14.0	19	25	100	63	39	12	F
WID-15.0	8	59	39	5	42	2	F
WID-16.0	1	10	12	43	11	0	F
WID-17.0	35	11	22	33	30	28	F
WID-18.0	40	47	25	33	24	32	F
WID-19.0	1	0	0	3	0	0	P
WID-20.0	8	2	6	4	36	0	F

3.1.4.4 Result of WEB Accessibility Tool

The evaluation results of the WEB accessibility validation tool are presented in Table 3.6 with both compliance score and number of violations. Previous studies considered the compliance scores for webpage evaluation. Generally, a compliance score does not always depend on a specific attribute. In some cases, webpages have a high compliance score but also have several instances of violation, and sometimes the scenario is reversed. Therefore, this research didn't consider compliance scores. Only the violations/errors are considered. The highest number of errors was for WID-7.0, with an average of 104. The lowest number of errors was for WID-4.0, WID-5.0, WID-11.0, and WID-20.0, with an average violation of 1. Among 21 webpages, only 2 webpages (WID-2.0 and WID-12.0) passed the accessibility testing with 0 violations. Most webpages recorded growing violations, ranging from 2 to 104.

3.1.4.5 Results of System Usability Testing and Expert Testing

To do the system usability testing, a few foreign volunteers were invited to measure the usability of the webpages and provide their feedback according to ten questions as shown in Table 3.1. All the volunteers who participated in this testing were university third-year (of four-year studies) bachelor students from the Computer Science and Engineering department of Jahangirnagar University, Bangladesh. Every student had completed the 'User Interface Design' course. Thus, they were appropriate in observing the

TABLE 3.6: WEB accessibility tool result of the selected Government COVID information webpages, considering the number of violations

W-ID	Compliance score	No of violations	Performance
WID-1.0	84%	17	F
WID-2.0	92%	0	P
WID-3.0	86%	6	F
WID-4.0	87%	1	F
WID-4.1	81%	5	F
WID-5.0	88%	1	F
WID-6.0	83%	5	F
WID-7.0	83%	104	F
WID-8.0	85%	3	F
WID-9.0	86%	9	F
WID-10.0	86%	18	F
WID-11.0	90%	1	F
WID-12.0	92%	0	P
WID-13.0	75%	8	F
WID-14.0	78%	26	F
WID-15.0	80%	23	F
WID-16.0	77%	11	F
WID-17.0	78%	24	F
WID-18.0	76%	66	F
WID-19.0	85%	2	F
WID-20.0	90%	1	F

webpages according to the questionnaire. The number of students was 10, including 4 females and 6 male students between 19 and 22 years of age. To investigate 21 webpages, online participation was offered, and questionnaires and webpage information were shared with the students. To distribute the work, 10 students were classified into two groups: 6 belonged to group 1, and 4 belonged to group 2. Group 1 was responsible for evaluating European webpages, and Group 2 was responsible for validating Asian webpages. In group 1, 5 students were asked to validate 2 webpages each, and 1 student was assigned to investigate 3 webpages. Besides, in group 2, 4 students were responsible for testing 2 webpages each. The calculated average investigation time was 15 min/webpage.

After getting the feedback from the user, to calculate the SUS score [78] of the targeted webpages from the questionary-based user feedback, first, the score was calculated from the odd number of questions, Qscore (Odd)=Xn-1; where n=1,3,5,7,9 and then computed the score of the even number of questions, Qscore (Even)=5-Xn; where n=2,4,6,8,10. Then the SUS score was calculated through Equation 3.1.

$$SUS_{Score} = \sum_{n=1,3,5,7,9} (X_n - 1) * 2.5 + \sum_{n=2,4,6,8,10} (5 - X_n) * 2.5 \tag{3.1}$$

To determine the quality of the webpages, six quality parameters were stated based on the SUS score, as shown in Table 3.7. According to the quality parameter, Table 3.7 shows the SUS score of each webpage with evaluation remarks. After evaluating the webpages through the SUS score, only 14% webpages were classified as Excellent webpages (WID-12, WID-15, and WID-19), and 9% webpages (WID-2.0, WID-3) were found to be the Best quality webpages. Furthermore, only 28% of webpages (WID-1, WID-5, WID-6, WID-7, WID-14, WID-20) were Good quality webpages. These webpages are accessible to people who have some disabilities. However, the ratio of these accessible webpages is insufficient to ensure that all the webpages are accessible. Besides, the rest of the webpages have no adequate functionality or accessibility, and webpages operators must consider webpage accessibility. For instance, fair, poor, and worst webpage ratios are 19%, and 9%, respectively. This result suggests that a significant number of accessibility issues should be under consideration for further improvement.

TABLE 3.7: SUS result of the targeted Government COVID information webpages with quality assessment parameter

W-ID	Country	SUS Score	Remarks	SUS score	Quality assessment
WID-1.0	Estonia	75.0	Good	90-100	Best
WID-2.0	France	97.5	Best	80-89	Excellent
WID-3.0	Greece	100	Best	70-79	Good
WID-4.0	Ireland	40.0	Poor	50-69	Ok/Fair
WID-4.1	Ireland	25.0	Worst	30-49	Poor
WID-5.0	Latvia	72.5	Good	1-29	Worst
WID-6.0	Luxembourg	72.5	Good		
WID-7.0	Netherland	75.0	Good		
WID-8.0	Iceland	65.0	Ok/Fair		
WID-9.0	Schweiz	42.5	Poor		
WID-10.0	Croatia	57.5	Ok/Fair		
WID-11.0	Germany	65.0	Ok/Fair		
WID-12.0	Austria	82.5	Excellent		
WID-13.0	India	50.0	Ok/Fair		
WID-14.0	Philippines	70.0	Good		
WID-15.0	Jordan	85.0	Excellent		
WID-16.0	UAE	20.0	Worst		
WID-17.0	Kyrgyzstan	35.0	Poor		
WID-18.0	Maldives	35.0	Poor		
WID-19.0	Hong Kong	80.0	Excellent		
WID-20.0	Qatar	75.0	Good		

Furthermore, to do the expert testing, four experts were invited in this research to validate the selected webpages according to the given questionnaire, as shown in Table 3.2, to provide their responses based on their observations. The occupation of experts includes an associate professor, an assistant professor, a senior lecturer, and a Ph.D. student from the Department of Electrical Engineering and Information Systems, University of Pannonia, Hungary, and the Department of Computer Science and Engineering, Jahangirnagar University, Bangladesh. All the experts validated 21 webpages and provided their findings. Every expert had at least 5 years of research experience in web accessibility, multimedia design, human-computer interaction, and web informatics.

After getting feedback from the expert, to calculate the expert score of the targeted webpages from the questionary-based expert feedback, the final expert score is calculated through equations 3.2 and 3.3 where X is the targeted tested webpage; n is the number of questions for expert assessments and x is the number of experts.

$$Expert_{Score(X)} = \sum_{n=12}(x_{(n)}) + \sum_{n=12}(x_{(n)}) + \sum_{n=12}(x_{(n)}) + \sum_{n=12}(x_{(n)}) \tag{3.2}$$

$$Expert_{Final-score(X)} = \frac{Expert_{Score(X)}}{x = 4} \tag{3.3}$$

To evaluate the webpages, like the SUS score, six quality parameters are stated for expert opinion analysis as shown in Table 3.8. After evaluating the expert opinion for each webpage through the stated quality parameter, only 19% webpages were classified as Excellent webpages (WID-2, WID-8, WID-14, and WID-16), and 4% webpages (WID-19) were found to be the Best quality webpages. Furthermore, 33% of webpages (WID-1, WID-3, WID-5, WID-6, WID-10, WID-17, WID-20) were Good quality webpages. These webpages are sufficiently accessible to people who have some disabilities. However, the ratio of

Table 3.8: Expert result of the targeted Government COVID information webpages

W-ID	Country	Expert Score	Remarks	Expert score	Quality assessment
WID-1.0	Estonia	0.75	Good	0.90-1.0	Best
WID-2.0	France	0.83	Excellent	0.80-0.89	Excellent
WID-3.0	Greece	0.75	Good	0.70-0.79	Good
WID-4.0	Ireland	0.58	Ok/Fair	0.50-0.69	Ok/Fair
WID-4.1	Ireland	0.50	Ok/Fair	0.30-0.49	Poor
WID-5.0	Latvia	0.75	Good	0.01-0.29	Worst
WID-6.0	Luxembourg	0.75	Good		
WID-7.0	Netherland	0.66	Ok/Fair		
WID-8.0	Iceland	0.83	Excellent		
WID-9.0	Schweiz	0.58	Ok/Fair		
WID-10.0	Croatia	0.75	Good		
WID-11.0	Germany	0.50	Ok/Fair		
WID-12.0	Austria	0.66	Ok/Fair		
WID-13.0	India	0.50	Ok/Fair		
WID-14.0	Philippines	0.83	Excellent		
WID-15.0	Jordan	0.66	Ok/Fair		
WID-16.0	UAE	0.83	Excellent		
WID-17.0	Kyrgyzstan	0.75	Good		
WID-18.0	Maldives	0.50	Ok/Fair		
WID-19.0	Hong Kong	0.91	Best		
WID-20.0	Qatar	0.75	Good		

these accessible webpages is insufficient to ensure universal accessibility. Furthermore, the accessibility and functionality of the remaining webpages are inadequate, and webpage operators need to be very careful. For instance, 42% webpages are classified as Ok/Fair. This finding implies that a sizable portion of the webpage's accessibility problems should be considered for future development.

From the evaluation results of the integrated approach, at first, a few webpages were accessible through the automatic evaluation tools. However, after going through questionnaire-based human judgment (system usability and expert testing), some inaccessible functionalities were found in the tested webpages that were returned as accessible from the automatic testing. Regarding the user testing and expert testing, the quality of each webpage (Table 3.7, 3.8) can be determined, which is not possible to identify through any automatic testing tool. Through the automatic tool, it is possible to identify whether the webpage has passed or failed the accessibility testing. Automatic tools can not detect these issues and produce the inappropriate research output. Thus, human observation is crucial in identifying the unreported problems of automatic tools. Regarding the user (SUS testing) and expert testing, none of the webpages has manual font resize functionality, which could be a problem for people with poor vision. Some webpages have no accessibility option when using the TAB key, an excellent example could be, WID-18 has no movement function with the TAB key. This webpage is not accessible to people who have learning disabilities. Some webpages have keyboard/TAB key accessibility, but not for all the buttons or sub-menus, making the webpage inaccessible. Some webpages are not in English, and there are no translation options, such as WID-9, which has no English version or translation options. Additionally, some webpages have a color deficiency, and links and buttons are difficult to separate. Therefore, human judgment helps to find unexplored issues that are not possible through automatic tools, which depicts the effectiveness of the proposed integrated approach.

3.1.4.6 Final Evaluation Results

After evaluating each selected webpage through multiple automated web accessibility testing tools and human observation involving six measurement techniques (MAUVE++; Nibbler; WAVE; WEB Accessibility; User Testing, and Expert Testing), seven distinctive results have been found, as shown in Table 3.9. As all of these experimental results have different assessment parameters/metrics, thus, based on the researcher's perspective and empirical observation, different threshold values have been assigned to evaluate each assessment parameter to determine the overall evaluation score from these seven distinctive results. The threshold values for each assessment parameter has set from 1 to 0 where, for (High, Excellent, Pass (P)) assessment parameters, the threshold value is set to (1.0); for (Best) assessment parameter, the threshold value is set to (0.8); for (Medium, Good) assessment parameters, the threshold value is set to (0.7); for (Low, Ok/Fair) assessment parameters, the threshold value is set to (0.5) and for (Worst, Poor. Fail (F)) assessment parameters, the threshold value is set to (0.3) as shown in Table 3.9. According to the assigned threshold values for each assessment parameter, the overall evaluation score has been calculated following Equation 3.11, incorporating Equations 3.4 to 3.10.

$$EvaluationScore_{Mauve(WCAG2.0)} = A \tag{3.4}$$

$$EvaluationScore_{Mauve(WCAG2.1)} = B \tag{3.5}$$

$$EvaluationScore_{Nibbler} = C \tag{3.6}$$

$$EvaluationScore_{WAVE} = D \tag{3.7}$$

$$EvaluationScore_{WEB} = E \tag{3.8}$$

$$EvaluationScore_{UserTesting} = F \tag{3.9}$$

$$EvaluationScore_{ExpertTesting} = G \tag{3.10}$$

$$OverallEvaluationScore = \frac{A+B+C+D+E+F+G}{x=7} \tag{3.11}$$

Furthermore, to determine the accessibility status according to the calculated overall evaluation score, different ranges have been incorporated where a webpage is considered 'Accessible' if the overall evaluation score range is between 1.0 to $\geq$0.90; if the overall evaluation score range is between <0.90 to $\geq$0.80, then it classified as 'Likely accessible'; if the overall evaluation score range is between <0.80 to $\geq$0.50, then it classified as 'Partially accessible', and if the overall evaluation score range is <0.50, it classified as 'not Accessible'. Following these ranges, Table 3.9 shows the classified accessibility status for each tested webpage. This result depicts that the majority of webpages (14) are classified as Partially accessible, and that reflects most of the tested webpages have evaluation scores between <0.80 to $\geq$0.50. Besides, one (1) webpage is classified as Likely accessible, and six (6) webpages are classified as not Accessible. However, none of the webpages were found to be completely Accessible. These statistics reflect that accessibility criteria were not properly followed during the webpage development, which introduces serious accessibility issues for people with disabilities.

Table 3.9: Final evaluation result of the targeted webpages

*User Testing (UT), Expert Testing (ET), Overall evaluation score (OES), Accessibility status (AS)									
W-ID	**Mauve WCAG 2.0**	**Mauve WCAG 2.1**	**Nibbler**	**WAVE**	**WEB**	**UT**	**ET**	**OES**	**AS**
WID-1.0	High(1)	Lo(0.5)	Medium(0.7)	P(1)	F(0.3)	Good(0.7)	Good(0.7)	0.70	Partially accessible
WID-2.0	Medium(0.7)	Low(0.5)	Low(0.5)	F(0.3)	P(1)	Best(0.8)	Excellent(1)	0.68	Partially accessible
WID-3.0	Medium(0.7)	Low(0.5)	Medium(0.7)	F(0.3)	F(0.3)	Best(0.8)	Good(0.7)	0.57	Partially accessible
WID-4.0	Medium(0.7)	Medium(0.7)	Low(0.5)	F(0.3)	F(0.3)	Poor(0.3)	Ok/Fair(0.5)	0.47	Not accessible
WID-4.1	Medium(0.7)	Low(0.5)	Low(0.5)	F(0.3)	F(0.3)	Worst(0.3)	Ok/Fair(0.5)	0.44	Not accessible
WID-5.0	Medium(0.7)	Low(0.5)	Low(0.5)	P(1)	F(0.3)	Good(0.7)	Good(0.7)	0.62	Partially accessible
WID-6.0	High(1)	Medium(0.7)	Low(0.5)	F(0.3)	F(0.3)	Good(0.7)	Good(0.7)	0.60	Partially accessible
WID-7.0	Medium(0.7)	Low(0.5)	Low(0.5)	F(0.3)	F(0.3)	Good(0.7)	Ok/Fair(0.5)	0.50	Partially accessible
WID-8.0	High(1)	High(1)	Low(0.5)	F(0.3)	F(0.3)	Ok/Fair(0.5)	Excellent(1)	0.65	Partially accessible
WID-9.0	High(1)	Medium(0.7)	Low(0.5)	F(0.3)	F(0.3)	Poor(0.3)	Ok/Fair(0.5)	0.51	Partially accessible
WID-10.0	Low(0.5)	Low(0.5)	Medium(0.7)	F(0.3)	F(0.3)	Ok/Fair(0.5)	Good(0.7)	0.50	Partially accessible
WID-11.0	Low(0.5)	Low(0.5)	High(1)	F(0.3)	F(0.3)	Ok/Fair(0.5)	Ok/Fair(0.5)	0.51	Partially accessible
WID-12.0	–	–	Low(0.5)	F(0.3)	P(1)	Excellent(1)	Ok/Fair(0.5)	0.47	Not accessible
WID-13.0	Low(0.5)	Low(0.5)	Low(0.5)	F(0.3)	F(0.3)	Ok/Fair(0.5)	Ok/Fair(0.5)	0.44	Not accessible
WID-14.0	Low(0.5)	Low(0.5)	Low(0.5)	F(0.3)	F(0.3)	Good(0.7)	Excellent(1)	0.54	Partially accessible
WID-15.0	Low(0.5)	Low(0.5)	Low(0.5)	F(0.3)	F(0.3)	Excellent(1)	Ok/Fair(0.5)	0.51	Partially accessible
WID-16.0	Medium(0.7)	Medium(0.7)	Low(0.5)	F(0.3)	F(0.3)	Worst(0.3)	Excellent(1)	0.54	Partially accessible
WID-17.0	Low(0.5)	Low(0.5)	Low(0.5)	F(0.3)	F(0.3)	Poor(0.3)	Good(0.7)	0.44	Not accessible
WID-18.0	Low(0.5)	Low(0.5)	Low(0.5)	F(0.3)	F(0.3)	Poor(0.3)	Ok/Fair(0.5)	0.41	Not accessible
WID-19.0	High(1)	High(1)	Low(0.5)	P(1)	F(0.3)	Excellent(1)	Best(0.8)	0.80	Likely accessible
WID-20.0	Medium(0.7)	Medium(0.7)	Low(0.5)	F(0.3)	F(0.3)	Good(0.7)	Good(0.7)	0.55	Partially accessible

3.2 Variable Magnitude Approach

In past literature, several studies suggested the variable magnitude approach as a hybrid approach for web content accessibility evaluation, as it is effective in identifying the accessibility of webpages in depth. According to their suggestions, Kuppusamy and Balaji [79] computed the accessibility barrier of several webpages through a variable-magnitude approach. They computed the accessibility barrier of several webpages through a variable-magnitude approach, considering multiple automated tools (AChecker and WAVE) output. Their proposed approach was effective; however, some limitations have been found that could reduce the effectiveness of their evaluated results. For example, for severity-factor selection, they chose a constant value of 0.1, which should be applicable for all the variables; however, they did not incorporate the severity factor for the initial variable (likely error, error). In other cases, the dimension of the input set of two algorithms was different (3:2), which might lead to an inappropriate calculation of the overall score. In addition, for different assessment parameters, they did not classify the dependent and independent parameters and their potential. For example, among known errors, likely errors, and potential errors, they did not address which error potentially reduces the accessibility or prime factor for the inaccessible scenario. Treating the three errors collectively might not be effective in identifying the actual accessibility scenario of webpages.

Addressing these aspects, a variable-magnitude approach has been proposed to improve the evaluation results by minimizing the addressed issues. To do so, the proposed variable magnitude approach considers the evaluation results of two automated tools (Mauve++ and TAW) as input variables to calculate the accessibility score by altering the weight of the input variables, based on their importance, and integrating multiple variables' results to compute the final score. Mathematical statistics, such as threshold measurement, are employed to classify the computed score in assessing the accessibility of the evaluated webpages. Furthermore, expert evaluation has been incorporated to justify the overall computed scores. The details of the proposed variable magnitude approach are described in the following section.

3.2.1 Automated Accessibility Tools

For applying this variable magnitude approach, the tool selection is dynamic. Many aspects could be considered during the appropriate automated accessibility testing tool selection. In this case, all the selected tools provide the necessary variables to compute the result via the proposed approach. Two criteria have been chosen that allow for the identification of the appropriate tools to incorporate into this work: (i) the result should specify which success criteria or guidelines have passed, failed, not been tested, or not been decided, and (ii) the report should highlight all the investigated guidelines. Considering these two criteria, six tools were evaluated: WAVE, Nibbler, MAUVE++, Web accessibility, AChecker, and TAW, according to their popularity and capability in performing the validation process. Unexpectedly, among these six tools, Mauve++ and TAW fulfilled the selected criteria and were incorporated into the proposed approach. The Mauve++ and TAW tools highlight all the tested guidelines and represent their results in terms of four terminologies: "Pass", "Fail", "Not Tested", and "Not Decided", where "Pass" refers to the guidelines that were properly implemented in the webpage, following the WCAG; "Fail" refers to the guidelines that were not properly implemented in the webpage, following the WCAG; "Not Tested" refers to the guidelines that were not considered for implementation in the webpage, according to the WCAG; and "Not Decided" refers to the guidelines that were undetected, according to the WCAG. The details of these terminologies can be found in [80]. In addition, the secondary reason is their ability to provide the evaluation results structurally, which allows us to integrate the results via our proposed approach. In addition, these two tools allow interaction with the web through programmatic interfaces (API) that make the testing process unbiased and reduce complexity. The API provides interaction through the browser interface and fetches the tested result by entering the URL of the tested webpage. Both tested tools offer a browser plugin.

3.2.1.1 Mauve++ Tool

It is an open-source accessibility-evaluation environment that allows users to test a particular webpage through its URL [71]. It validates webpages against the Web Content Accessibility Guidelines (WCAG), providing various platforms (such as desktop, iPad, mobile phone, and tablet) to perform the evaluation process. A detail of this tool can be found in section 3.1.2.1.

3.2.1.2 TAW Tool

It is another open-source accessibility evaluation framework that allows the evaluation of webpage accessibility status without charging costs. It generates an evaluation report that mentions the total number of issues on the webpage that are related to accessibility aspects in terms of the number of "Passed", "Failed", "Not Tested", and "Not Decided" success criteria [81]. Figure A.5 (Appendix A) shows the accessibility-evaluation result of the tested webpage via TAW, in terms of their assessment terminologies.

3.2.2 Questionnaire-Based Expert Testing

Expert testing was incorporated into this work to validate the evaluated results of the proposed approach. Expert testing allows identification of several accessibility issues by incorporating human judgment [82]. Expert testing was conducted through ten questionnaires with three assessment parameters: "Accessible", "Partially Accessible", and "Not Accessible". Table 3.10 shows the prepared questionnaire for the expert evaluations. Each of the questions was associated with five disability types of vision impairment, color-blind issues, motion difficulty, hearing difficulty, and cognitive difficulty. Additionally, if experts find webpages partially accessible or not accessible, then the expert is asked to provide a descriptive opinion about what aspects made the webpage inaccessible or partially accessible. In addition, the expert was asked to submit valuable suggestions to improve such inaccessible scenarios for future consideration.

Table 3.10: Questionnaire for expert evaluation with assessment parameter

Questions	Assessment Parameter
Q1. Is the webpage accessible to low-vision people?	Accessible/Partially accessible/ Not accessible
Q2. Is the webpage accessible to color-blind people?	Accessible/Partially accessible/ Not accessible
Q3. Is the webpage accessible to a person with a motion disability?	Accessible/Partially accessible/ Not accessible
Q4. Is the webpage accessible to a person with a hearing disability?	Accessible/Partially accessible/ Not accessible
Q5. Is the webpage accessible to a person with a cognitive disability?	Accessible/Partially accessible/ Not accessible
Q6. Does the webpage require user information?	Accessible/Partially accessible/ Not accessible
Q7. Does the webpage have keyboard functionality?	Accessible/Partially accessible/ Not accessible
Q8. Does the webpage have manual font size adjustment functionality?	Accessible/Partially accessible/ Not accessible
Q9. Does the webpage have inaccessible features?	Accessible/Partially accessible/ Not accessible
Q10. Is it possible to distinguish between links and buttons?	Accessible/Partially accessible/ Not accessible

3.2.3 Accessibility Score Computation

The proposed variable magnitude approach combines the results of two automatic accessibility-testing tools. The first output is the evaluation result of several components computed by Mauve++, and the second output is the validation result of multiple components computed by TAW. Then, the computed results of Mauve++ and TAW are integrated via the accessibility computation algorithm. The computed accessibility score represents the magnitude of the accessibility status of the tested webpages for people with disabilities.

3.2.3.1 Mauve++ Accessibility Score

To compute the accessibility score of the selected webpage, the URL of the tested webpage is passed as input to Mauve++. The rule set is selected as Web Content Accessibility Guideline 2.1, considering level AAA. The output of Mauve++ is generated according to four components, as shown in Equation 3.12.

$$Evaluation\_Components, kk = [\beta, \varphi, \gamma, \delta] \tag{3.12}$$

In Equation 3.12, β represents the number of passed guidelines, φ indicates the number of failed guidelines, γ represents the number of not-tested guidelines, and δ refers to the number of not-decided guidelines. Based on the severity of the tested guidelines, Equation 3.13 shows the guidelines in descending order of severity.

$$Severity\_Order = \Phi[\beta] > \Phi[\varphi] > \Phi[\gamma] > \Phi[\delta] \tag{3.13}$$

Following the severity order, four different weight coefficients ω_β, ω_φ, ω_γ, and ω_δ together with a severity factor ϵ, are introduced to compute the accessibility score. β is multiplied by its weight coefficient ω_β and severity factor ϵ, as shown in Equation 3.14, to compute the ratio of the passed guidelines, ρ_1.

$$Pass\_Ratio, \rho_1 = [\beta * \omega_\beta * \epsilon] \tag{3.14}$$

The weight coefficient of the failed guidelines ω_φ is calculated by scaling down the weight coefficient of the number of pass guidelines ω_β by multiplying the severity factor ϵ, as shown in Equation 3.15.

$$Weight\_Coffeciecnt, \omega_\varphi = [\omega_\beta * \epsilon] \tag{3.15}$$

φ is multiplied by its weight coefficient ω_φ, to compute the ratio of the failed guidelines, ρ_2, as shown in Equation 3.16.

$$Fail\_Ratio, \rho_2 = [\varphi * \omega_\varphi] \tag{3.16}$$

The weight coefficient of the not-tested guideline ω_γ is calculated by scaling down the weight coefficient of the number of failed guidelines ω_φ by multiplying the severity factor ϵ, as shown in Equation 3.17.

$$Weight\_Coffecient, \omega_\gamma = [\omega_\varphi * \epsilon] \tag{3.17}$$

γ is multiplied by its weight coefficient ω_γ to compute the not-tested ratio, ρ_3, as shown in Equation 3.18.

$$Not\_Tested\_Ratio, \rho_3 = [\gamma * \omega_\gamma] \tag{3.18}$$

The weight coefficient of the not-decided guidelines ω_δ is calculated by scaling down the weight coefficient of the number of not-decided guidelines ω_γ by multiplying the severity factor ϵ, as shown in Equation 3.19.

$$Weight\_Coefficient, \omega_\delta = [\omega_\gamma * \epsilon] \tag{3.19}$$

δ is multiplied by its weight coefficient ω_δ to compute the not-decided ratio, ρ_4, as shown in Equation 3.20.

$$Not\_Decided\_Ratio, \rho_4 = [\delta * \omega_\delta] \tag{3.20}$$

After computing each component's score (ρ_1, ρ_2, ρ_3, ρ_4), the score of Mauve, $\rho_{(Mauve)}$ is computed through Equation 3.21. The ComputeAccessibility.Mauve function represents the full procedure as shown in Algorithm 1 (Figure 3.2).

$$Mauve\_Score, \rho_{(Mauve)} = [\rho_1 - \sum(\rho_2 + \rho_3 + \rho_4)] \tag{3.21}$$

Algorithm 1: ComputeAccessibility.Mauve

Input: URL (α)

Output: Accessibility Score $\rho_{(Mauve)}$

```
Function ComputeAccessibility.Mauve ()
  dataMauve = invoke Mauve (α)
  if <dataMauve ≠ Ø> then
   do preprocessing (dataMauve)
   Initialize:
    ϵ = 0.2//set severity factor
    β = dataMauve.count (pass)//fetch pass result count
    φ = dataMauve.count (fail)//fetch fail result count
    γ = dataMauve.count (not-tested)//fetch not-tested result count
    δ = dataMauve.count (not-decided)//fetch not-decided result count
    if β = = 0, then
      if φ = = 0, then
        if γ = = 0, then
          if δ = = 0, then
            ρ(Mauve) = 0
            return ρ(Mauve)
   else
     else
       else
            ρ1 = β*ωβ*ϵ//multiply by weight and severity factor
            ωφ = ωβ*ϵ
            ρ2 = φ*ωφ
            ωγ = ωφ*ϵ
            ρ3 = γ*ωγ
            ωδ = ωγ*ϵ
            ρ4 = δ*ωδ
            ρ(Mauve) = (ρ1 − ∑ (ρ2 + ρ3 + ρ4))
            return ρ(Mauve)
End
```

FIGURE 3.2: Algorithm for Mauve++ Computed Score

3.2.3.2 TAW Accessibility Score

The accessibility score computation by TAW is based on four components, κ, λ, μ, and χ. These four components refer to the number of passed, failed, not-tested, and not-decided guidelines, similar to the returned component by Mauve. Like the accessibility-barrier computation via Mauve, TAW-based accessibility-barrier computation considers variable weight and severity factors.

The four variable weights are ω_κ, ω_λ, ω_μ, ω_χ, and their severity factor is ϵ. The weight coefficients ω_κ, ω_λ, ω_μ, and ω_χ, are scaled by their severity factors ϵ, as shown in Algorithm 2 (Figure 3.3). The pass, fail, not-tested, and not-decided ratio computations are shown in Equation 3.22.

$$\rho_1 = [\kappa * \omega_\kappa * \epsilon]; \rho_2 = [\lambda * \omega_\lambda]; \rho_3 = [\mu * \omega_\mu]; \rho_4 = [\chi * \omega_\chi] \tag{3.22}$$

The score of TAW, $\rho_{(Taw)}$, is computed through Equation 3.23. The ComputeAccessibility.TAW function represents the full procedure, as shown in Algorithm 2 (Figure 3.3).

$$Taw\_score, \rho_{(Taw)} = [\rho_1 - \sum(\rho_2 + \rho_3 + \rho_4)] \tag{3.23}$$

Algorithm 2: ComputeAccessibility.TAW

Input: URL (α)

Output: Accessibility score count $\rho_{(Taw)}$

```
Function ComputeAccessibility.TAW ()
  dataTaw = invoke Taw (α)
  if <dataTaw ≠ Ø> then
    do preprocessing (dataTaw)
    Initialize:
    ϵ = 0.2//set severity factor
     κ = dataTaw.count (pass)//fetch pass result count
     λ = dataTaw.count (fail)//fetch fail result count
     μ = dataTaw.count (not-tested)//fetch not-tested result count
     χ = dataTaw.count (not-decided)//fetch not-decided result count
      if κ = = 0, then
        if λ = = 0, then
          if μ = = 0, then
            if χ = = 0, then
              ρ(Taw) = 0
              return ρ(Taw)
      else
        else
          else
              ρ1 = κ*ωκ *ϵ
              ρ2 = λ* ωλ//where ωλ = ωκ*ϵ
              ρ3 = μ*ωμ//where ωμ = ωλ*ϵ
              ρ4 = χ*ωχ//where ωχ = ωμ*ϵ
              ρ(Taw) = (ρ1 − ∑ (ρ2 + ρ3 + ρ4))
              return ρ(Taw)
End
```

FIGURE 3.3: Algorithm for TAW Computed Score

3.2.3.3 Overall Accessibility Score

The overall accessibility score is computed based on the mathematical summation of the accessibility counts via Mauve++ and TAW, as shown in Equation 3.24. The complete procedure for the overall accessibility score is described in Algorithm 3 (Figure 3.4) under the ComputeScore function.

$$Overall\_accessibility\_score\_count, \rho = \frac{\sum(\rho_{(Mauve)} + \rho_{(Taw)})}{n} \tag{3.24}$$

Algorithm 3: Compute Accessibility

Input: URL (α)

Output: Accessibility Score Count ρ

Function *ComputeScore ()*

$\rho_{(Mauve)} = ComputeAccessibility.Mauve\ (\alpha)$

$\rho_{(Taw)} = ComputeAccessibility.TAW\ (\alpha)$

$\rho = \frac{\sum(\rho_{(Mauve)} + \rho_{(Taw)})}{n}$ *//where n = 2*

return *(ρ)*

End

FIGURE 3.4: Algorithm for Overall Computed Score

3.2.4 Experimentation with computed results

The proposed variable-magnitude approach was experimented with and validated by considering the home pages of hospitals and clinical webpages in Hungary. The hospitals and clinics were chosen from the capital city, other large cities, and small towns in Hungary. The URLs of the considered 16 hospitals and clinics' webpages are listed in Table B.2 (Appendix B).

3.2.4.1 Result of MAUVE++ Accessibility Tool

To compute the accessibility score through the proposed variable magnitude approach, first, the accessibility scores were computed by incorporating Mauve $\rho_{(Mauve)}$ via the ComputeAccessibility.Mauve process (Algorithm 1). The computed results of 16 healthcare institutions' webpages are shown in Table 3.11. The visual illustration of the computed accessibility score is shown through a bar graph in Figure A.6 (Appendix A) with their corresponding webpage ID as a reference.

3.2.4.2 Result of TAW Accessibility Tool

The TAW [$\rho_{(Taw)}$] accessibility score was computed through ComputeAccessibility.TAW algorithm (Algorithm 2). The computed results are shown in Table 3.12. Following the visual representation of the computed accessibility score by Mauve, Figure A.7 (Appendix A) shows the visual representation of the computed accessibility score by TAW, considering a bar graph.

TABLE 3.11: Computed score of $\rho_{(Mauve)}$ for the selected healthcare institution webpages

Webpage ID	Pass	Fail	Not Tested	Not Decided	$\rho_{(Mauve)}$
ID-1	21	11	42	8	13.43
ID-2	20	12	42	8	7.19
ID-3	25	13	35	9	49.36
ID-4	22	15	37	8	7.04
ID-5	20	13	38	11	9.45
ID-6	24	12	37	9	45.22
ID-7	21	19	32	10	40.32
ID-8	25	11	36	10	69.56
ID-9	22	12	38	10	20
ID-10	19	12	43	8	18.70
ID-11	23	12	38	9	31.78
ID-12	25	11	37	9	69.33
ID-13	27	8	41	6	139.08
ID-14	20	11	41	10	3.90
ID-15	23	10	41	8	50.26
ID-16	23	15	35	9	6.26

TABLE 3.12: Computed score of $\rho_{(Taw)}$ for the selected healthcare institution webpages

Webpage ID	Pass	Fail	Not Tested	Not Decided	$\rho_{(Taw)}$
ID-1	14	12	6	18	0.44
ID-2	17	7	7	19	53.31
ID-3	14	12	5	19	1.34
ID-4	16	7	9	18	41.47
ID-5	16	6	9	19	47.61
ID-6	14	9	10	17	12.99
ID-7	16	5	9	20	53.76
ID-8	25	11	36	10	16.32
ID-9	15	7	9	19	32.64
ID-10	14	6	10	20	29.12
ID-11	15	6	9	18	40.5
ID-12	15	11	5	19	13.43
ID-13	13	7	9	21	17.47
ID-14	14	7	10	19	23.74
ID-15	13	9	8	20	8.32
ID-16	14	5	14	17	30.91

3.2.4.3 Result of Overall accessibility computation

The overall accessibility score (ρ) is computed through ComputeScore algorithm as shown in Algorithm 3. Table 3.13 shows the overall computed scores with their IDs. Figure 3.5 shows the graphical representation of the computed overall accessibility score through a bar graph. Every graph is represented with its corresponding reference ID. To evaluate the computed overall accessibility scores, a threshold value is set between ξ=0 to <40; 40 to <70; and 70 to $\leq$100. If the computed accessibility score is under the range of ξ=0 to <40, it would be considered as not accessible webpages. If the score is between the range of ξ=40 to <70, the webpage will be considered as likely accessible. If the computed score ranges between ξ=70 to $\leq$100, it will be a potentially accessible webpage.

According to the computed accessibility ratio (Table 3.13/Figure 3.5), the observation outcome is that the majority of the webpage's accessibility score is between the range of ξ=0 to <40 which depicts that they have low accessibility scores and are not accessible for people with disabilities. It represents a poor concern in the context of universal accessibility. Besides, two webpages have accessibility scores between the range of ξ=40 to <70, classified as likely accessible (ID-8, ID-12), and one webpage has an accessible score between the range of ξ=70 to $\leq$100, determined as potentially accessible (ID-13). This result shows the need for universal accessibility.

TABLE 3.13: Computed Accessibility Score of ρ for the selected healthcare institution webpages

Webpage ID	Accessibility Score	Status
ID-1	6.94	Not accessible
ID-2	23.05	Not accessible
ID-3	25.35	Not accessible
ID-4	17.21	Not accessible
ID-5	19.08	Not accessible
ID-6	29.10	Not accessible
ID-7	6.71	Not accessible
ID-8	42.94	Likely accessible
ID-9	26.32	Not accessible
ID-10	5.20	Not accessible
ID-11	36.14	Not accessible
ID-12	41.38	Likely accessible
ID-13	78.27	Potentially accessible
ID-14	13.82	Not accessible
ID-15	29.29	Not accessible
ID-16	18.58	Not accessible

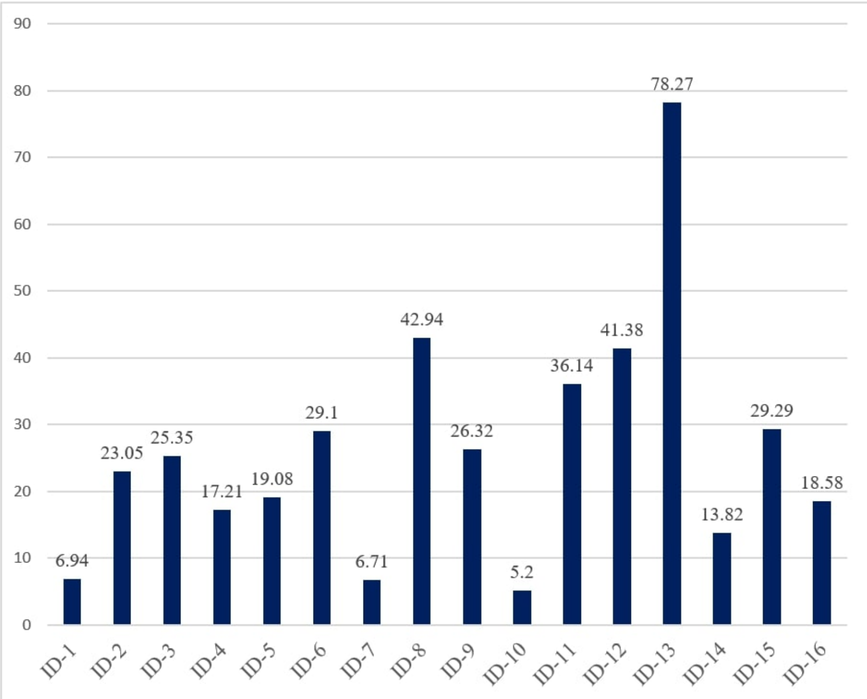


FIGURE 3.5: Bar graph of computed overall accessibility score count, ρ of the selected healthcare institution webpages

3.2.4.4 Result of Expert Evaluation

To carry out the expert evaluation, four experts were invited to evaluate the selected webpages and present their reports in terms of the three assessment criteria. All experts from the Department of Electrical Engineering and Information Systems, University of Pannonia, Hungary. Each of the experts had more than 6 years of experience in the field of "web accessibility", "human-computer interaction", "multimedia design", and "web informatics". The expert evaluation results concluded in Table 3.14 according to ten questions. The evaluation results were categorized into three criteria: (a) if a webpage is categorized as accessible by the expert for at least nine questions (evaluation criteria (accessible) $\geq$ nine questions), then the webpage is considered to be accessible; (b) if a webpage is categorized as accessible by the expert for less than nine questions but equal to or more than seven questions (evaluation criteria (accessible) $<$ nine questions to $\geq$ seven questions), then the webpage is considered to be partially accessible; (c) if a webpage is categorized as accessible by the expert for less than seven questions (evaluation criteria (accessible) $<$ seven questions), then the webpage is considered not to be accessible. The results show that most of the webpages were not accessible, and very few were accessible for people with disabilities (Table 3.14).

Table 3.14: Expert assessment/evaluation results of the selected healthcare institution webpages

Webpage ID	**Expert opinion**
ID-1	Not accessible
ID-2	Not accessible
ID-3	Not accessible
ID-4	Partially accessible
ID-5	Not accessible
ID-6	Not accessible
ID-7	Partially accessible
ID-8	Partially accessible
ID-9	Not accessible
ID-10	Not accessible
ID-11	Not accessible
ID-12	Accessible
ID-13	Accessible
ID-14	Not accessible
ID-15	Not accessible
ID-16	Not accessible

From the descriptive opinions of the expert, it could be concluded that not-accessible webpages have several issues in relation to people with disabilities; more specifically, provided information was not organized, there were absences of a manual translation and a manual font-size-adjustment option, the search functions were not responsive, buttons were not visible or understandable, no navigation direction was provided, the content font size was small and hard to read even for normal people, very dark colors were used, and additional user information was required in some cases. These issues limit the acceptability of these webpages for people with disabilities. In terms of disability types, the majority of the webpages were not accessible to people with vision impairment or cognitive disabilities. Although automated evaluation and analytical statistics were useful in predicting the overall accessibility score, it is quite a difficult task to identify particular issues associated with disability. Thus, the expert evaluation was significant for reporting the unreported accessibility issues of people with disabilities.

From the expert analysis report, it can be concluded that the webpage designers and developers did not properly comply with the web content accessibility guidelines. Web designers and developers ignored the accessibility guidelines during development, even though the associated accessibility issues could have been easily fixed. Our objective was to emphasize that every disability type and its requirements should be considered throughout the design and development phase, as people with disabilities contribute to the economic development of a country. Therefore, it is encouraged to all designers, developers, and practitioners who are working with any development aspects (e.g., game development, web development, software development, and mobile application development) to follow the Web Content Accessibility Guideline (WCAG) standards, issued by the World Wide Web Consortium (W3C), throughout their webpage development process.

3.2.4.5 Final Evaluation Results

The final evaluation results in terms of the accessibility status of the tested webpages from both the proposed variable magnitude approach and the expert evaluation or opinion depict that the majority of the tested webpages (11 webpages out of 16 webpages) are classified as Not accessible from both perspectives, as shown in Table 3.15. Besides, three webpages (ID-4, ID-7, ID-8) are classified differently from two different perspectives, although none of them are classified as accessible for people with disabilities. Additionally, two webpages (ID-12, ID-13) were classified significantly differently, where experts found these two webpages accessible for people with disabilities, but the proposed variable magnitude approach found accessibility issues on those webpages. This might happen as the proposed variable magnitude approach incorporates multiple automated tools that evaluate the webpage source code through algorithmic evaluation, and experts present their opinions according to their expertise. Therefore, some issues lie in the source code of the webpages, which is somehow challenging to determine from the individual perspective. However, these statistics show that a significant number of webpages are not accessible for people with disabilities, which reduces universal access for the community with different disabilities.

TABLE 3.15: Final evaluation result of the targeted webpages

W-ID	ρ(Mauve)	ρ(Taw)	Overall score	Accessibility status (Proposed approach)	Accessibility status (Expert opinion)
ID-1	13.43	0.44	6.94	Not accessible	Not accessible
ID-2	7.19	53.31	23.05	Not accessible	Not accessible
ID-3	49.36	1.34	25.35	Not accessible	Not accessible
ID-4	7.04	41.47	17.21	Not accessible	Partially accessible
ID-5	9.45	47.61	19.08	Not accessible	Not accessible
ID-6	45.22	12.99	29.10	Not accessible	Not accessible
ID-7	40.32	53.76	6.71	Not accessible	Partially accessible
ID-8	69.56	16.32	42.94	Likely accessible	Partially accessible
ID-9	20.0	32.64	26.32	Not accessible	Not accessible
ID-10	18.70	29.12	5.20	Not accessible	Not accessible
ID-11	31.78	40.5	36.14	Not accessible	Not accessible
ID-12	69.33	13.43	41.38	Likely accessible	Accessible
ID-13	139.08	17.47	78.27	Potentially accessible	Accessible
ID-14	3.90	23.74	13.82	Not accessible	Not accessible
ID-15	50.26	8.324	29.29	Not accessible	Not accessible
ID-16	6.26	30.91	18.58	Not accessible	Not accessible

3.3 Machine Learning-based approach

Nowadays, web researchers have shown their active participation in considering machine learning (ML) methods to evaluate the quality and usability of webpages. By addressing the potential of ML approaches in web platforms, the prime aim of this section is to propose an ML approach for performing accessibility evaluation of web platforms.

From the literature study (Chapter 2), the existing web accessibility evaluation approaches incorporate some specific attributes of webpages according to the Web Content Accessibility Guidelines (WCAG) to evaluate their accessibility status. Though the latest version of WCAG 2.2 [83] is a complete guideline for accessibility features, it has limited consideration about some issues with people with disabilities such as whether the webpage is active or deactivated, webpage has a manual text size adjustment option, manual font family adjustment option, manual color adjustment option, user information requirement, CAPTCHA issues, the usefulness of internal/external links, used images, inserted video and audio content. As these features are not directly possible to evaluate automatically, WCAG does not provide a clear indication about these aspects, and without considering these aspects, it is impossible to ensure complete accessibility of the developed webpages. In that manner, our prime focus is to evaluate webpages considering these additional criteria through ML algorithms and compute the overall accessibility score based on the selected additional criteria. The objective of this work is to observe the performance of the selected two ML classifiers (Random Forest (RF), and Decision Tree (DT)) to identify their effectiveness and then evaluate webpage accessibility according to the classification result. Figure 3.6 shows the working diagram that illustrates the materials and methods used in this study.

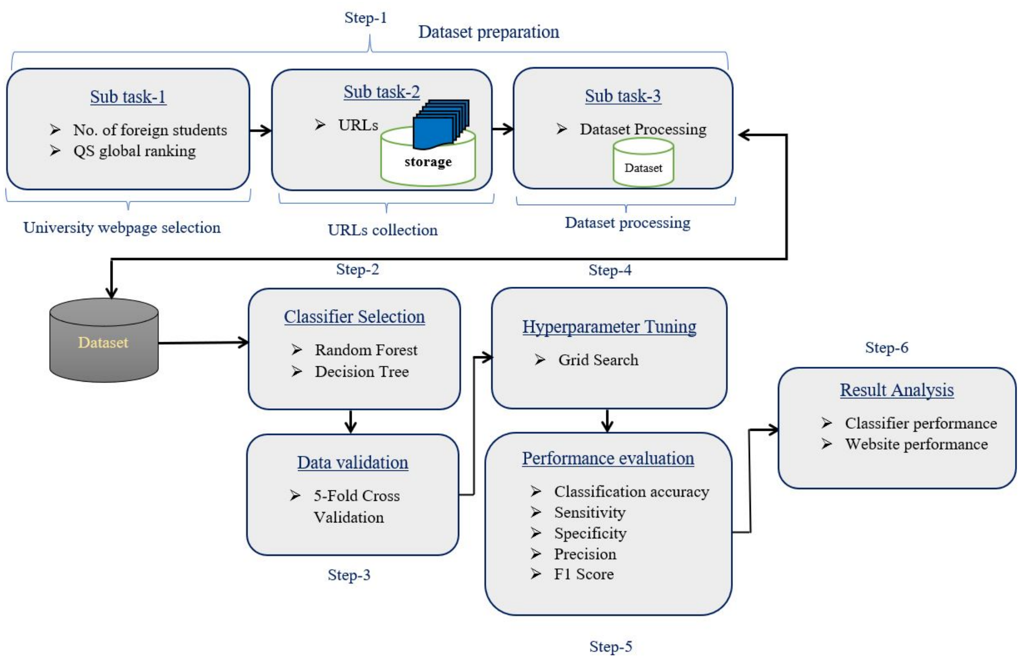


FIGURE 3.6: The System Architecture of the proposed ML-based approach

3.3.1 Dataset preparation

The prime challenges of this work related to the prepared dataset as there is no dataset has been found that evaluated webpages according to ten additional criteria such as availability, manual text size adjustment option, manual font family adjustment option, manual color adjustment option, user information requirement, CAPTCHA, usefulness of internal/external links, images, inserted video, and audio content. Thus, a custom dataset was prepared according to ten aspects. For dataset preparation, a preliminary survey was conducted to understand the importance of the selected ten (10) criteria in terms of their effectiveness for people with disability to represent the accessibility of the webpages. Six users participated in this survey, including vision problems (4) and cognitive problems (2), as the selected eight features are more likely to cause difficulty to these groups of people with disabilities. All the participants are active on the Internet platform for their professional work and daily activities. All the participants are between 25-50 years old. In the online survey, participants were asked to provide their opinions about whether these ten aspects are useful for them to understand the webpage content effectively. All the users expressed their positive opinion that these aspects are useful for understanding the web content and can improve their internet browsing experience. According to the positive feedback from the preliminary survey, a dataset was prepared to incorporate human observation, where 23 participants observed the selected five university webpages according to the selected ten aspects. All participants were university students from the Department of Electrical Engineering and Information Systems at the University of Pannonia, Hungary. All the participants had adequate knowledge of interactive design and web development. To select the five university webpages, the Webometrics university ranking [84] was considered for selecting the top five universities in Hungary. Then, the top five universities were ranked according to their number of international students and QS world ranking. The selected five university webpages are **Web-1:** https://u-szeged.hu/english (No. of foreign students: 5000; QS ranking: 551); **Web-2:** https://unideb.hu/en (No. of foreign students: 4000; QS ranking: 600); **Web-3:** https://www.elte.hu/en/ (No. of foreign students: 3000; QS ranking: 700); **Web-4:** http://www.bme.hu/?language=en (No. of foreign students: 1900; QS ranking: 801); **Web-5:** https://www.ceu.edu/ (No. of foreign students: 962; QS ranking: 124). To evaluate the selected webpages, online participation was arranged where all the information were shared with the participants including the prepared 10 questions related to the considered ten criteria (shown in Table 3.16) and webpage resources that need to be observed. All the participants observed the webpages and answered each question according to their understanding.

TABLE 3.16: Selected questions for the preliminary survey

Questions	**Response**
Question-1: Is the webpage available?	Yes/No
Question 2: Does the webpage have a manual text size adjustment option?	Yes/No
Question 3: Does the webpage have a manual font family adjustment option?	Yes/No
Question 4: Does the webpage have a manual color adjustment option?	Yes/No
Question 5: Does the webpage require user information?	Yes/No
Question 6: Does the webpage require CAPTCHA?	Yes/No
Question 7: Are the internal/external links useful?	Yes/No
Question 8: Are the webpage images useful?	Yes/No
Question 9: Is the webpage video content useful?	Yes/No
Question 10: Is the webpage audio content useful?	Yes/No

After obtaining the responses from users for five selected webpages, it was labeled in terms of 'Accessible', 'Partially Accessible', and 'Not Accessible' metrics where all positive responses or ten positive responses were labeled as accessible; having > 6 negative response (out of 10) were labeled as Not accessible and rest of the responses having ≤ 6 negative response (out of 10) were labeled as Partially Accessible. Figure 3.7 shows the dataset preparation flowchart to represent the entire process in detail. In total, twenty-three (23) responses were recorded for each dataset related to eight webpage features and classified into three levels of accessibility status. The used custom dataset is named as [1]ML-Dataset-University-WebPages. However, the five tested webpages observation results were incorporated into the proposed system to conduct the experimental analysis. The implemented datasets are categorical; thus, LabelEncoding was used to encode the data (label/categories). LabelEncoding is a popular categorical data encoding process. The LabelEncoding was used through the sklearn LabelEncoder () function.

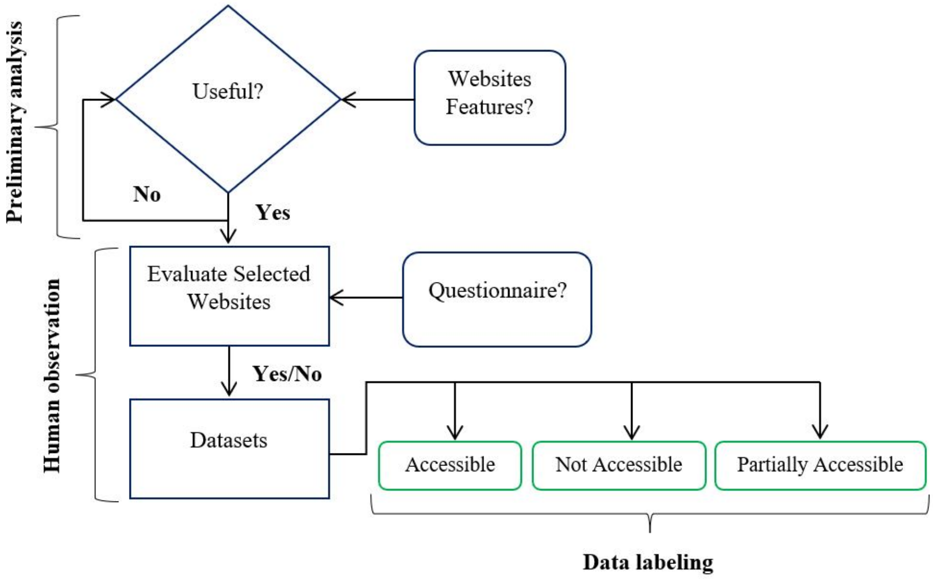


FIGURE 3.7: Flowchart of Dataset preparation

3.3.2 Methods and Materials

The methods and materials for system architecture and design have been described through step-2, step-3, step-4, and step-5 (according to Figure 3.6). These four steps are described in the following subsections.

3.3.2.1 ML Classification Algorithms

Machine learning is a branch of artificial intelligence that employs statistics, probabilities, absolute conditionality, boolean logic, and unconventional optimization strategies to learn and classify patterns

[1]https://drive.google.com/file/d/12QuZI7VgEBv2a-Ec0AkZ-bRxHRoX-zbU/view?usp=drive_link

through predictive models [84]. ML has both supervised and unsupervised models for classification and regression problems. Thus, the two most commonly used supervised ML classifiers or algorithms were incorporated, such as Random Forest (RF) and Decision Tree (DT).

Random Forest Classifier: Random Forest (RF) is a popular and most effective supervised machine learning classifier in classification and regression problems. The prime objective of RF is to reduce classification errors. It performs by building a decision tree by taking samples randomly. It takes the majority of the voting results to classify the output. It aggregates the decisions by taking the average results of all the trees to improve the predictive accuracy and control the over-fitting problem. The advantage of the random forest classifier is to handle the data set containing continuous variables for regression and categorical variables in classification [85]. However, it provides the best performance for categorical variables in classification problems.

Decision Tree Classifier: Decision Tree (DT) is a supervised machine-learning algorithm for classification problems. It is one of the most standard and commonly used ML algorithms. The prime goal of this algorithm is to predict the output value considering the target value and represent the solved problem in the form of a tree representation called a decision tree with a leaf node or decision node [86]. A decision tree has internal and external nodes, where the internal node is responsible for decision-making. In the decision tree, leaf nodes represent the class label, and the internal nodes represent the attributes. The main objective of using the DT in this work is to predict the target class instances using the decision rule learned from the prior data. In the build decision tree, root nodes classify the instances with different features, where root nodes have multiple branches, and the leaf nodes represent the classification result. The DT chooses a node according to the highest information gain among all the attributes. The best way to information-gain is to calculate entropy. Entropy is the quantified measurement of the amount of uncertainty of random instances, as shown in equation 3.25, where X is the random instance, X_i is the possible outcomes, and $P(X_i)$ is the outcome probability. By the entropy value for any random instance, the information gain was calculated through equation 3.26.

$$Entropy, H(X) = -\sum_{i=1}^{n} P(\frac{x}{i}) log P(\frac{x}{i}) \tag{3.25}$$

$$InformationGain = 1 - Entropy \tag{3.26}$$

3.3.2.2 Hyperparameter Tunning

It is noteworthy that there are several machine learning classifiers, and every ML classifier requires different constraints, weights, or learning rates to generalize the data patterns. Failure of the appropriate parameter selection might lead to differences in the final result. For example, every iteration of the classifier resulted in a different accuracy. Therefore, it is crucial to make appropriate parameter selections to solve the ML problem and improve classification accuracy. The proper parameter selection method is hyperparameter tuning or hyperparameter optimization. Hyperparameter tuning is the optimal parameter selection process for an ML model. It is crucial for any ML model implementation as it directly optimizes the performance of ML classification. It allows for defining all possible parameters for testing all the combinations to maximize the classification results [87]. Also, this optimization process allows Cross-Validation (CV) to estimate the generalization performance. A crucial aspect that needs to be mentioned is that every ML classifier has different default parameter requirements. Several hyperparameter optimization methods are available to select the best parameter. Herein, a grid search approach is used to evaluate classification accuracy with different combinations of parameters using a 5-fold CV. Grid search is a popular method for parameter fitting [88], which is implemented with the Scikit-learn library. It allows the selection of the optimization parameter following the default parameter. For details on how parameters influence the decision of ML models, literature is suggested proposed by Wu et al. [89]. This study employed Grid search as its most traditional hyperparameter optimization technique. It is also called a parameter sweep. It is the most effective but time-constrained procedure that takes longer than other optimization techniques and provides better results. Seven relevant parameters for RF and DT have been selected to reduce the complexity of the Grid search. The optimal values are shown in Table B.3 (Appendix **??**)) which lists the selected parameters of RF, and DT with definition, default values, grid values, and the optimal values for the optimization process.

3.3.2.3 Cross-Validation (CV)

The most effective way to validate the performance of an ML model is to train a model with available data and test its classification performance using a newly separated dataset. Another popular technique is the Train-Test Split. It is the process of data splitting before model development and using the separated data for performance validation. However, these processes require a substantial amount of data for validation. Generally, CV is used to evaluate the performance of learning algorithms or models by partitioning data into a training set for pattern learning and a testing set for model evaluation [86]. The prime idea is to split the dataset into training and test sets according to the user-defined number of partitions, such as K-fold. First, the dataset is divided into k folds, where the k-1 fold is used for training and the remaining fold for testing. In the experiment, the entire dataset is divided into 70:30, where 70% data was used for training and 30% for testing purposes. The model is trained on the training set considering a five-fold partitioning setting, and evaluated on the model through the testing set for performance measurement. For hyperparameter tuning, five-fold partitioning is applied on each fold of the training set.

3.3.2.4 Performance Evaluation Metrics

The evaluation metrics for measuring the classifier's performance are derived from the binary confusion matrix, as represented in Figure 3.8. The True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) counts are applied to calculate the classification accuracy, precision, recall, and f1 score of the classifier.

Here,

True Positive (TP) is the number of predicted instances as positive which are originally positive.

False Positive (FP) is the number of predicted instances as positive but are originally negative.

True Negative (TN) is the number of predicted instances as negative that are originally negative.

False Negative (FN) is the number of predicted instances as negative that are originally positive.

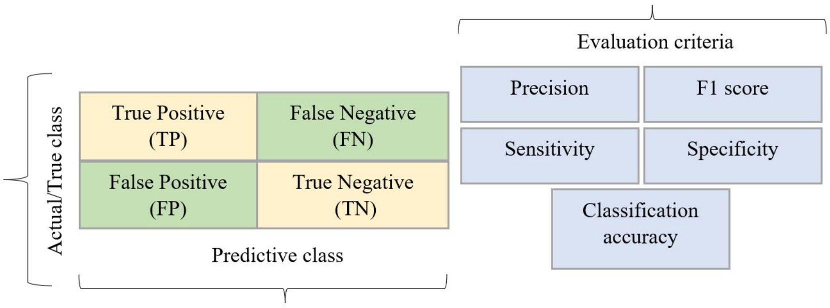


FIGURE 3.8: Confusion Matrix with several evaluation metrics

The evaluation criteria, such as precision, f1 score, sensitivity, specificity, and classification accuracy, have been explained in the following through equations [3.27 - 3.31].

Classification accuracy: Classification accuracy is the proportion of the number of correctly classified samples (Equation 3.27).

$$Classification\_Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{3.27}$$

Precision: Precision is the proportion of the samples that are actually true (equation 3.28).

$$Precision = \frac{TP}{(TP + FP)} \tag{3.28}$$

Sensitivity: Sensitivity is the proportion of total correctly predicted samples by the learning algorithm (equation 3.29).

$$Sensitivity = \frac{TP}{(TP + FN)} \tag{3.29}$$

Specificity: Specificity is the proportion of the correctly predicted negative samples of all negative samples (equation 3.30).

$$Specificity = \frac{TN}{(TN + FP)} \tag{3.30}$$

F1 score: F1 score is the harmonic mean of precision and recall. To achieve the best performance, F1 score should be one. Besides, for the lowest performance, it's usually zero (equation 3.31).

$$F1\_Score = \frac{2 * (Precision * Sensitivity)}{(Precision + Sensitivity)} \tag{3.31}$$

3.3.3 Experimentation with computed results

Experimentation with different datasets were performed considering the Python programming language in a jupyter notebook environment. The experiments were run on a computer with an integrated 2.5 GHz processor and 8 GB of RAM. I employed two ML classifiers: Random Forest and Decision Tree for their efficiency and reliability [90]. Every experiment was performed by taking datasets in CSV format as input to the ML classifier and splitting the dataset into a 70:30 ratio. The confusion matrix of experimental datasets is tabulated in Table 3.17 for two selected classifiers: Random Forest (RF) and Decision Tree (DT), respectively. Table 3.17 also elucidates different measurements for classification result assessment, such as precision, sensitivity, specificity, F1 score, and overall accuracy, to illustrate the effectiveness of the selected ML classifier. This table shows that the Random Forest (RF) classifier performs well for all five tested datasets. To compute the accessibility score, the score of each class (0, 1, 2) is quantified based on the number of samples of predicted data (Table 3.17) as shown in Equation 3.32. Also, I incorporated the severity score based on the importance of three classes as shown in equation 3.33. The computed score of each class has been scaled down by multiplying its severity level as shown in Equation 3.34 and computing the final score through Equation 3.35.

$$Accessible, (\alpha) = TP_0; Partially\_Accessible, (\beta) = TP_1; Not\_Accessible, (\gamma) = TP_2 \tag{3.32}$$

$$\epsilon_\alpha = 0.2, \epsilon_\beta = 0.1, \epsilon_\gamma = 0.01 \tag{3.33}$$

$$Accessible = [\alpha * \epsilon_\alpha]; Partially\_Accessible = [\beta * \epsilon_\beta]; Not\_Accessible = [\gamma * \epsilon_\gamma] \tag{3.34}$$

$$Total\_accessibility\_score = \frac{(Accessible + Partially\_Accessible) - NotAccessible}{n} * 100 \tag{3.35}$$

Table 3.17: Classification results of five datasets using RF and DT

*Accessible (0); *Not Accessible (1); *Partially Accessible (2)									
*Precision (P); *Sensitivity (S); *Specificity (Sp); *F1 score (F1); *Accuracy (Ac)									
Random Forest (tested data)									
		Predictive class			**Classification Metrics**				
Datasets	**Actual class**	**(0)**	**(1)**	**(2)**	**P**	**S**	**Sp**	**F1**	**Ac**
	Accessible (0)	3	0	0					
Dataset1	Not Accessible (1)	0	9	0	1.0	0.95	0.99	1.0	0.97
	Partially Accessible (2)	0	2	9					
	Accessible (0)	3	0	0					
Dataset2	Not Accessible (1)	0	8	0	1.0	0.86	0.99	1.0	0.99
	Partially Accessible (2)	0	0	12					
	Accessible (0)	5	0	0					
Dataset3	Not Accessible (1)	0	4	0	1.0	0.86	0.99	1.0	0.97
	Partially Accessible (2)	0	2	12					
	Accessible (0)	1	0	0					
Dataset4	Not Accessible (1)	0	7	0	1.0	0.94	0.99	0.97	0.99
	Partially Accessible (2)	0	0	15					
	Accessible (0)	4	0	0					
Dataset5	Not Accessible (1)	0	9	0	1.0	1.0	1.0	1.0	1.0
	Partially Accessible (2)	0	0	10					
Decision Tree (tested data)									
	Accessible (0)	2	0	0					
Dataset1	Not Accessible (1)	0	8	2	0.97	0.97	0.97	0.96	0.95
	Partially Accessible (2)	0	1	10					
	Accessible (0)	4	0	0					
Dataset2	Not Accessible (1)	0	7	3	0.98	0.95	0.93	0.95	0.94
	Partially Accessible (2)	0	0	9					
	Accessible (0)	5	0	0					
Dataset3	Not Accessible (1)	0	5	0	1.0	1.0	1.0	1.0	1.0
	Partially Accessible (2)	0	0	13					
	Accessible (0)	2	0	0					
Dataset4	Not Accessible (1)	0	12	0	0.96	0.94	0.98	0.98	0.94
	Partially Accessible (2)	0	1	8					
	Accessible (0)	4	0	0					
Dataset5	Not Accessible (1)	0	6	3	0.91	0.92	0.94	0.94	0.92
	Partially Accessible (2)	0	0	10					

The computed accessibility score for each classifier with its average score and standard deviation (SD) is shown in Table 3.18. According to the computed accessibility score, dataset 3 has a higher accessibility score than the other experimented datasets. Besides, dataset 4 experienced the lowest accessibility score. In the context of accessibility, the higher the accessibility score, the lower the accessibility barrier. In contrast, the lower the accessibility score, the higher the accessibility barrier. Also, the standard deviation (SD) represents the difference between the computed results of two implemented classifiers. According to SD values, the highest difference among the two classifiers was observed for datasets 4 and 5.

TABLE 3.18: Accessibility scores count of tested datasets

Dataset	RF	DT	Avg. Accessibility Score	SD
Dataset-1	47%	44%	45.5%	2.12
Dataset-2	57%	53%	55%	2.23
Dataset-3	72%	75%	73.5%	2.12
Dataset-4	25%	36%	30.5%	7.77
Dataset-5	53%	58%	55.5%	3.53

Figure 3.9 shows the graphical representation of the computed accessibility score (for both RF and DT), their average accessibility score, and SD values for each dataset, where the results of each dataset are referenced with their associated dataset/webpage. It shows that the webpage of the University of Debrecen (Dataset 4/Web-4) has the lowest accessibility score among other selected university webpages. In contrast, the webpage of Eotvos Lorand University (Dataset 3/Web-3) has the highest accessibility score. From these computed scores, it is noteworthy that none of the selected university webpages were found completely accessible or barrier-free for people with several disabilities in accordance with the selected features.

FIGURE 3.9: Graphical representation of the computed accessibility score of five selected university websites

The experimental result shows that the classification performance of the RF classifier is more significant than DT classifier. The average accessibility score shows that Eotvos Lorand University has higher accessibility features (according to the selected features in this research work) than other university webpages. However, the computed score of other selected university webpages was poor, which represents that most of the tested university webpages are not accessible to people with disabilities in terms of the experimented aspects/features. To improve accessibility in accordance with the selected aspects, the selected university webpages need to improve their quality to ensure the complete accessibility objective. In addition, concerning the performance of ML classifiers or models, it is interesting to note that ML classifiers or models are significant in accessibility evaluation of university webpages. However, throughout the experimentation, this study has some limitations associated with single-page validation and a small dataset. Therefore, a further investigation is required, focusing on the current limitation that will be considered in future work.

3.4 Conclusion

In this Chapter, three different hybrid evaluation approaches have been presented that have been introduced in this research work, such as (i) an integrated approach, (ii) a variable magnitude approach, and (iii) a machine learning-based approach. Several webpages from different domains have been considered for the experimentation to evaluate the effectiveness of the proposed approaches. From the experimentation and evaluation results, it can be concluded that the proposed and presented approaches have the potential to improve the effectiveness of the evaluation result to facilitate the overall web accessibility evaluation process.

Chapter 4

Automated Web Content Accessibility Evaluation Methods

Since 1996, accessibility evaluation of the Web platform has been an important aspect of web development to increase social inclusion for people with special needs. Several Web accessibility evaluation and testing tools have been developed to automatically evaluate webpages to identify barriers for people with disabilities. The existing automated tools are significant as they aim to represent accessibility issues effectively. However, several studies in the existing literature claimed that some issues with the accessibility of webpages cannot be identified through the existing automated accessibility testing tools due to several issues related to guideline selection, guideline modeling, lack of consideration of additional criteria, and semantic perspectives. Therefore, the reported results may be unclear and inappropriate for some users. Such limitations are critical factors that reduce the effectiveness of the developed tools. These issues cause unwillingness to use a particular tool or the possible adoption of other tools.

According to the findings of Chapter 2, it could be depicted that most automated web accessibility testing processes have several issues that hinder their effectiveness. In particular, ontological models are time-consuming solutions and have a greater possibility of raising ambiguity. Therefore, focusing on the declarative model and algorithmic evaluation process, there is an emerging need to propose an interactive accessibility testing process to develop an updated automated accessibility testing tool to mitigate the existing shortcomings of the current tools.

Therefore, in this Chapter, as shown in Figure 4.1, the first objective is to propose an inclusive automated accessibility testing framework considering the determined aspects that can facilitate the improvement of the available limitations of automated accessibility testing systems. The second objective is to present the automated web content accessibility testing tool development process following the proposed framework, with its validation result as the third objective. The overall goal of this Chapter is to answer the research questions **RQ3**, **RQ4**, and **RQ5** as follows.

R-Q3: How can we improve the effectiveness of the automated web accessibility evaluation process to address the limitations of the available automated web accessibility testing tools?

R-Q4: How can we increase the effectiveness of an automated web content accessibility assessment tool focusing on advanced engineering techniques?

R-Q5: How can the progress made with the developed tool be verified in terms of its effectiveness aspects?

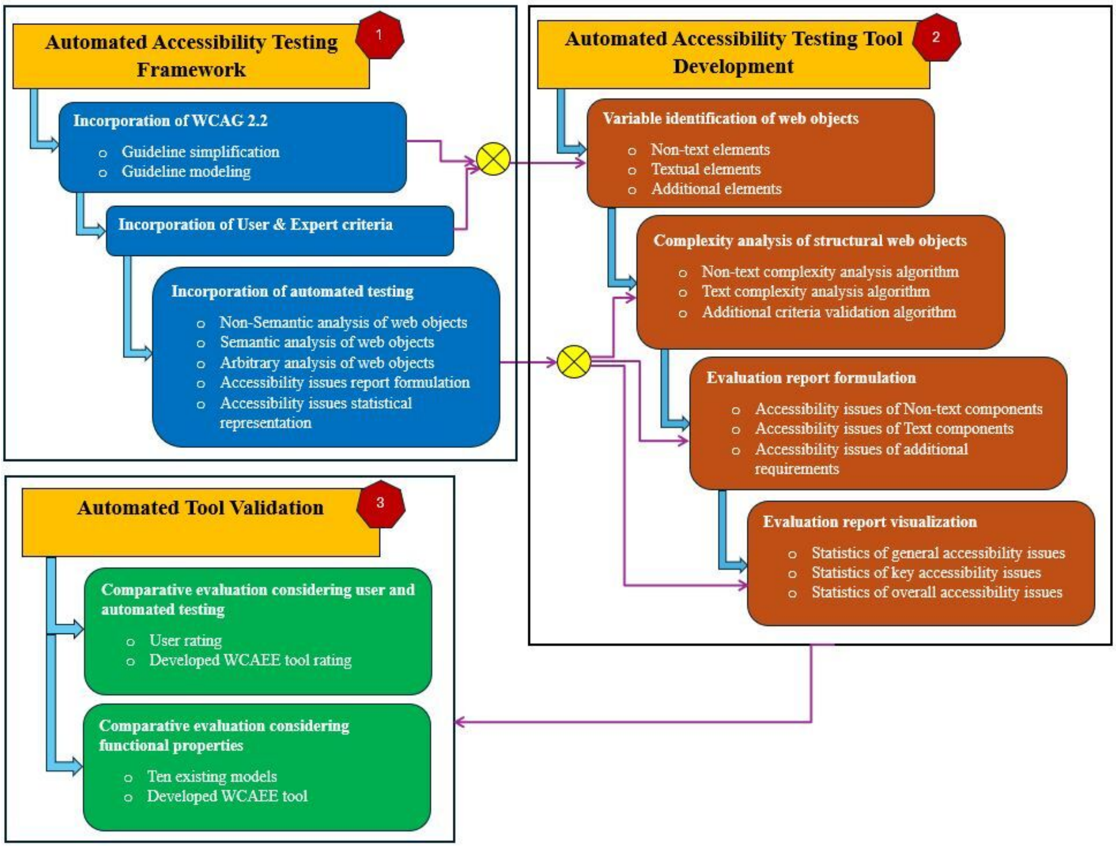


FIGURE 4.1: Workflow diagram

4.1 An Inclusive Framework for Automated Web Content Accessibility Evaluation

According to the observation and the reported findings, this section proposes a framework considering several crucial aspects that need to be included in the development process to limit the existing issues of the current automated web accessibility evaluation tools to answer the third research question as follows:

R-Q3: How can we improve the effectiveness of the automated web accessibility evaluation process to address the limitations of the available automated web accessibility testing tools?

The proposed accessibility evaluation framework demonstrates five aspects, considering several criteria to improve ambiguities, such as accessibility guidelines, user and expert suggestions, guideline simplification, automated testing, and issue identification and visualization. The proposed approach focuses on different aspects, including standard guidelines, additional criteria, computation, and visualization of the evaluation report. Thus, it could be beneficial to facilitate the evaluation and representation process of the computed results as reliable, acceptable, and fair. In addition, compared to the existing system, the aspects addressed in the proposed system are not considered in the existing systems, which makes the proposed system distinctive. The proposed framework will be useful for web practitioners and web researchers to understand the web evaluation process in detail. According to several existing shortcomings listed in Chapter 2, the general observation is that the main aspects leading to incorrect perception, encoding, and development of the accessibility evaluation tool are:

- Understanding difficulties of natural language formatted web content accessibility guidelines
- Limited consideration of user requirements and expert suggestions
- Lack of semantic concern

These addressed issues make the evaluation results less credible and less acceptable. Most accessibility testing tools only check a specific number of WCAG success criteria, which is around 50% of the total guidelines. As a result, it restricted the evaluation process, and the overall evaluation result might be inaccurate. As many web accessibility guidelines cannot be assessed automatically, they do not specify whether guidelines require user/expert testing. This could also lead to an incorrect calculation and the formulation of the evaluation report. Without incorporating user requirements/opinions and expert suggestions during the development of accessibility testing tools, the evaluation process may overlook some crucial aspects and inadvertently inflate the final accessibility score. Besides, a lack of consideration of semantic criteria may reduce the effectiveness of the evaluated results. Therefore, to minimize such issues, an accessibility aspects framework for automated web accessibility testing considering the following aspects could help the development procedure and improve the evaluation process with accurate results.

- Simplifying the updated version of the web content accessibility guideline to represent the knowledge of the guideline more easily and effectively.
- Incorporating all success criteria into the evaluation process to make the evaluation results more effective while improving the fairness of the evaluation results.
- Incorporating user requirements/opinions with expert suggestions during the evaluation process as an additional criterion for evaluation.
- Incorporating separate complexity analysis algorithms for textual feature and non-textual feature analysis, focusing on semantic aspects to improve the effectiveness of the evaluated result.
- Categorizing the evaluated guidelines, considering the evaluation of the user and expert when the guideline is not applicable for automatic evaluation.
- Displaying evaluated results with the overall accessibility score with specific accessibility scores for each types of disabilities.

Referring to the findings, Figure 4.2 shows the proposed automated web accessibility evaluation framework. The proposed framework considers several aspects related to the selection of appropriate accessibility guideline, consideration of user and expert suggestions (additional evaluation feature) and

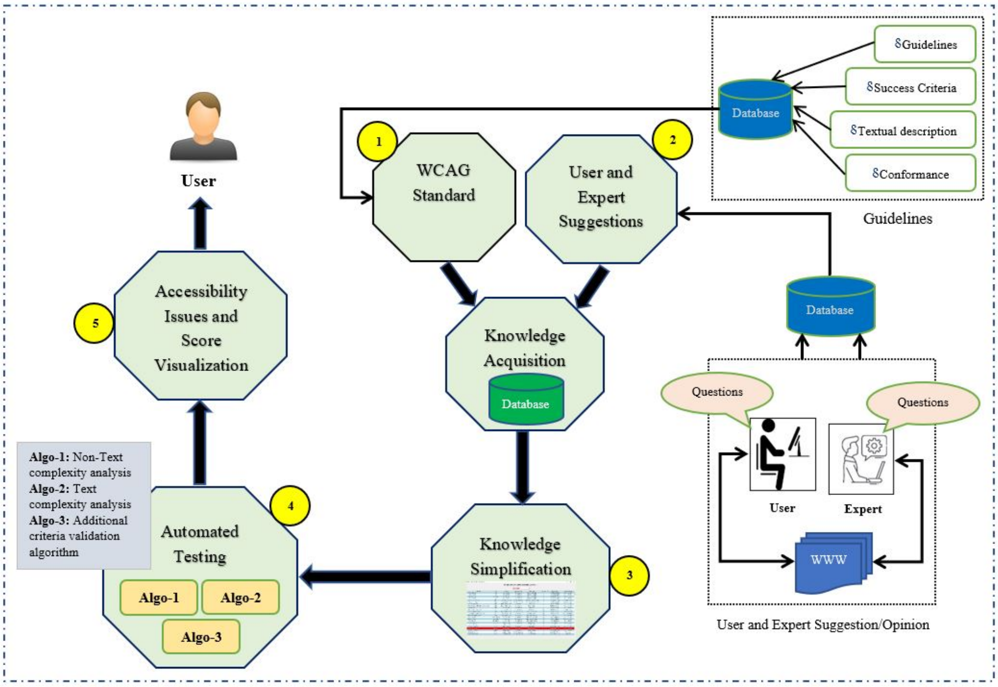


FIGURE 4.2: The proposed automated accessibility testing framework

simplification of guideline knowledge prior to the algorithmic coded process that facilitates the appropriate accessibility evaluation scores computation.

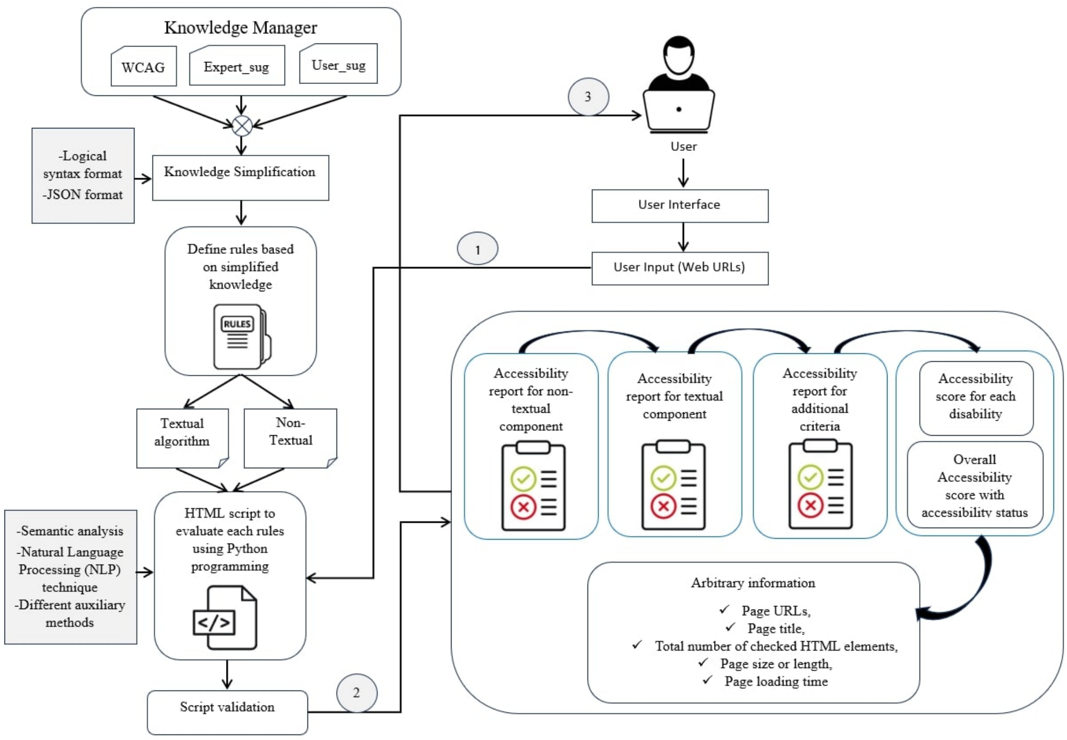


FIGURE 4.3: Use case diagram of the proposed automated accessibility testing framework

Figure 4.3 shows a use case diagram that explains the specifics of the accessibility assessment procedure performed by the proposed framework. All aspects shown in Figure 4.2 are discussed in the following sections:

4.1.1 Aspects of Web Content Accessibility Guideline

Several governments and organizations from different countries have presented various accessibility guidelines in recent years. Referring to existing studies [91][92], the Web Content Accessibility Guideline (WCAG) is the most sophisticated and widely accepted guideline. WCAG has several versions, which is the most used guideline by most of the existing testing tools. Its newer version (WCAG 2.2) has more features/success criteria than its previous versions WCAG (1.0, 2.0, 2.1). It contains 13 guidelines with 87 success criteria that are distributed into three conformance levels: A, AA and AAA where 33 success criteria are assigned to level A, 24 success criteria are assigned to level AA, and 30 success criteria are assigned to level AAA. The conformance levels ensure the priority of the success criteria in terms of their importance, where A refers to what might be included based on the development/evaluation criteria, AA refers to what should be and AAA refers to what must be included. However, to make the evaluation process effective and correct, it is crucial to incorporate all success criteria under three conformance levels. Therefore, in our proposed automated accessibility testing framework, every success criterion of the updated WCAG 2.2 guideline is considered that could facilitate the improvement of the overall evaluation result. The summary of WCAG 2.2 is shown in Figure 4.4. Detailed information on WCAG can be found in [93].

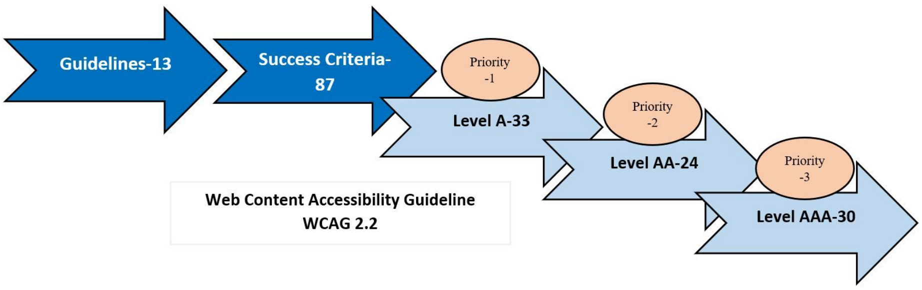


Figure 4.4: Overview of WCAG 2.2 (the latest version of WCAG)

4.1.2 Aspects of User Opinions and Expert Suggestions

From the evaluation results of existing studies (Chapter 2), user opinions and expert suggestions are the most common limitation in their work. As the development of automated web accessibility testing tools is limited to a specific guideline, the evaluation process might not consider all the difficulties associated with disabilities. Sometimes, incorporation of every success criterion of guidelines is not possible through automated means. Therefore, considering user opinion and expert suggestion might be a valuable resource to identify additional requirements as an additional evaluation criterion with WCAG during the development of an automated accessibility testing tool.

Generally, user opinion refers to the expressed opinion of users based on the difficulties they have encountered during the experimentation or web evaluation process [94]. Depending on the type of experiment and the nature of user opinion collection, user opinions may be collected in different forms. Questionnaire-based evaluation (questions asked to the user) is the most common and effective way to obtain user opinion [95]. Therefore, 10 users were interviewed including 6 students (M.Sc. students are aged between 24-28) and 4 end-users (2 persons are older adults aged between 63-71, and 2 persons are service holders aged between 35-42), who frequently access webpages for their daily activities. Two questions were asked during the interview and requested to provide their valuable opinion. The interview was taken via Zoom meeting and it took about 30-35 minutes for each. All the asked questions are as follows.

Q-1: What kind of features would you expect that could be helpful to navigate the webpage from your perspective?
Q-2: Why do you prefer these features? How could it be helpful to navigate the webpage?

All participants mentioned several aspects that are frequently encountered during webpage navigation and introduced accessibility issues. However, many of their suggested criteria are related to WCAG. All of those criteria were excluded and considered only 5 criteria that are unique, and web content accessibility guidelines, including WCAG, do not address these aspects. The selected criteria are: manual text size and font adjustment availability, manual color adjustment availability, necessity of user information, and availability of textual and image CAPTCHA. However, the questionnaire used in this thesis work concerning these aspects helps us understand the user's perspective and obtain their particular requirement regarding every single aspect. Understanding user requirements might be helpful in improving the overall accessibility evaluation process.
In the context of expert suggestion, it refers to the improvement recommendation of prototypes of some particular aspects to avoid some unconsidered situations [96]. From the web content accessibility perspective, experts are the people who have a thorough understanding of accessibility standards as well as technical knowledge of webpage design and development process. Depending on their role, web developers, web designers, accessibility test specialists, UX/UI experts, and researchers are known as accessibility experts [97]. Some additional aspects such as word/sentence length, specific font family, font size, and other factors require fixed determinators to make the webpage accessible. To determine these aspects, expert suggestions or perspectives were considered as additional factors, as they have more expertise, experience, and critical judgment knowledge. As a result, I consulted/interviewed three specialists/experts to get their insightful opinions on how to enhance the functionality of the web accessibility assessment tool. I requested opinions on any additional factors that could be useful to add to the evaluation process but not included in the WCAG. Three experts were from the Department of Electrical Engineering and Information Systems, University of Pannonia, Veszprem, Hungary, and the Department of Computer Science, Jahangirnagar University, Savar, Bangladesh. Two experts have more than 20 years of experience in the domain of accessibility of digital platforms and one expert has more than 5 years of experience in this field. They were chosen as possible experts based on their background and expertise. They are aware of all versions of WCAG and have sufficient knowledge and experience to work with WCAG. To conduct this interview, three questions were selected that were asked during

the interview and requested to provide insightful suggestions. The interview was taken in person and the overall interview took 45-56 minutes for each person. The selected questions are as follows:

Q-1: What kind of arbitrary information do you think is useful to improve the performance of the automated accessibility evaluation results that are not mentioned in WCAG?
Q-2: What kind of non-arbitrary information has an impact on improving the effectiveness of the automated accessibility evaluation results that are not included in WCAG?
Q-3: Why do you suggest these attributes? How it facilitates the performance, please explain.

From expert suggestion, 17 additional factors are considered that are beyond the mentioned criteria in the web content accessibility guideline (WCAG) from both arbitrary, and non-arbitrary point of view. These additional requirements include Loading time; Paragraph length; Hyperlink ratio; Default Language; Webpage length; Server Status/Availability; User information; CAPTCHA; Multiple languages; Image ratio; Manual font adjustment option; Manual color adjustment option; Text Font family; Text Font size; Text pattern; Content type; and Number of audio/video content. By considering expert suggestions along with WCAG, including any new elements they recommend, it might be possible to enhance the webpages evaluation results. After obtaining additional criteria from users and expert suggestions/opinions, these criteria have been incorporated into the evaluation process or in the proposed framework with WCAG as additional rules or guidelines to improve the entire accessibility evaluation process.

4.1.3 Aspects of Guideline Simplification and Guideline Modelling

Guideline simplification is the text simplification process. As guidelines are written in natural language format, it allows simplification of existing guidelines to make the guidelines more comprehensible to users or associated authorities [98]. However, web content accessibility guidelines are written in natural language format and there is no logical representation of these guidelines, it is relatively difficult to understand and implement these guidelines in the development of web accessibility evaluation tools [99]. To understand these complex guidelines, adequate accessibility knowledge and high-level technical competence are required. In that regard, the concept of web content accessibility guidelines simplification has been considered, which can help to represent the guideline simply and effectively. In the guideline simplification process, all the WCAG success criteria have been categorized into eight criteria: guidelines, objects, attributes, components type, requirements, conformance level, beneficiary type and evaluation type/phase (as shown in Figure 4.5). This classification process might represent the simplified guideline more systematically, which will help encode every guideline and perceive the webpage feature appropriately through the developed accessibility evaluation algorithm. Also, guideline modeling helps semantically encoded of each guideline.

For guideline modeling, WCAG knowledge representation or simplification is performed by simplifying 87 checkpoints, incorporating domain (website design and development criteria with programming knowledge) and expert knowledge (website design and accessibility issues), which is performed applying knowledge acquisition and knowledge representation process (as shown in Figure 4.6).

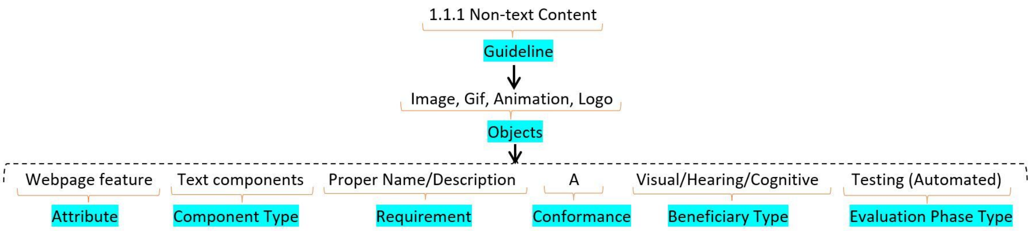


FIGURE 4.5: A simplification process of web content accessibility guideline

Knowledge acquisition is performed using domain knowledge and guideline knowledge, which are integrated to simplify natural language-formatted guidelines through accessibility engineering knowledge. Furthermore, in the knowledge representation process natural language formatted guideline is simplified according to logical syntax, represented through a JSON file format to encode into the coding process. However, this guideline simplification and modeling approach specifies accessibility guidelines considering several requirements and provides a complete direction that might help in designing to testing phase of the web evaluation tool development. In general, this guideline modeling approach is used to understand guidelines and requirements to facilitate the development process from the technical to the coding aspects.

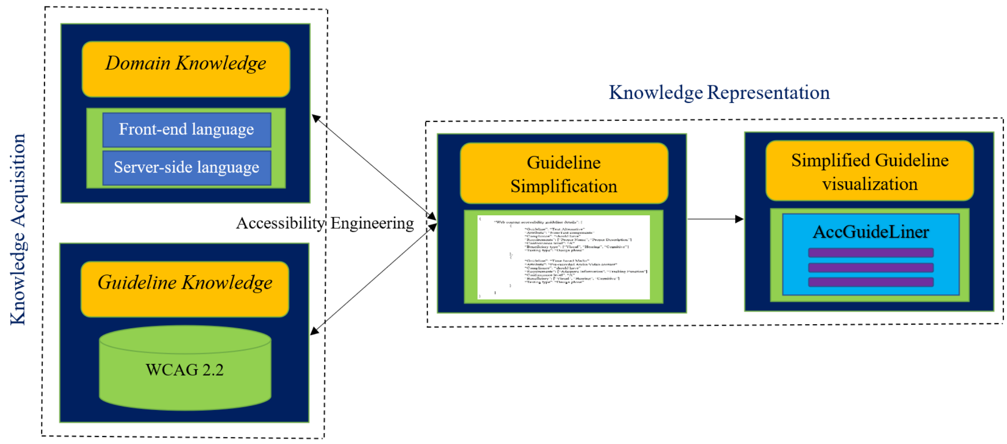


FIGURE 4.6: Guideline modeling approach

4.1.4 Aspects of Automated Testing

In the existing literature, different accessibility research groups proposed several prominent automated accessibility testing methodologies, including Accessibility Conformance Testing (ACT), Source code mining, Application Programming Interface (API) based testing, and Ontology based testing. However, these approaches have several limitations when it comes to conducting accessibility testing. For example, for the implementation of ACT rules, maintaining a unique ID is necessary for web attributes mapping to one or more WCAG success criteria, which is challenging [100]. In source code mining, the analyzed code might not be structured enough to extract relevant regularities due to poor precision and recall, and a large number of user involvement is required [101]. Besides, API testing is effective, but it does not allow interaction with real user activity and proceeds only with a raw request [102]. In an ontology-based evaluation, inconsistency in the knowledge base and database reduce the effectiveness of the result [103]. To overcome these difficulties, several studies reported that algorithmic evaluation of web content is important, especially for web accessibility testing [58][75]. Through the algorithmic evaluation process, it is possible to analyze webpage source code by incorporating every guideline offered by the Web Initiative Forum called Web Content Accessibility Guideline (WCAG). As WCAG derived most of the prototypes of a webpage including textual and non-textual content and features, two separate algorithms have been incorporated for complexity analysis of textual content and non-textual features that might improve the performance of the algorithm and provide a straight guideline about algorithm specification. To incorporate the textual, and non-textual algorithms, the most effective process is considered that is to parse the webpage's HTML code and analyze its features using Artificial Intelligence (AI) techniques that might be effective in assessing each web feature and validating the guidelines and other additional requirements (from user requirements, and expert suggestions) appropriately. Figure 4.7 shows the workflow diagram of the algorithmic evaluation process for accessibility testing of a particular webpage. This process is the most convenient algorithmic evaluation process for validating every guideline for every webpage feature and evaluating them in terms of four assessment terminologies such as 'Pass', 'Fail', 'Not tested', and 'Not detected' where Pass refers to those guidelines that have been followed and successfully implemented; Fail refers to those guidelines that have been followed, but wrongly implemented; Not detected refers to those guidelines that should be followed, but not implemented; and guidelines need user or expert testing refers to Not tested because the software fails to test the feature.

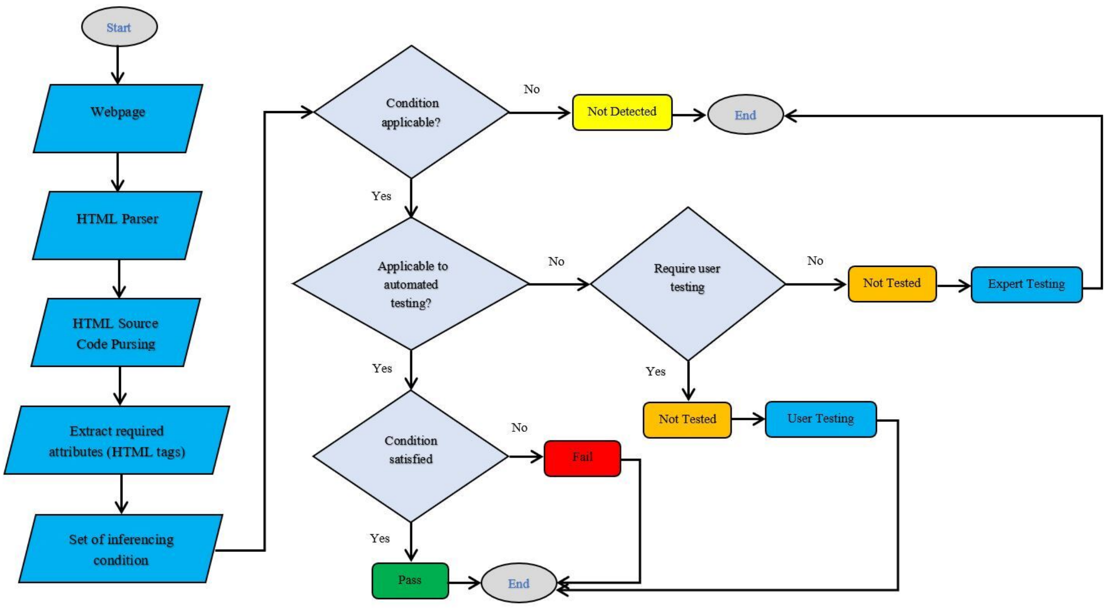


Figure 4.7: Workflow of algorithmic evaluation of accessibility testing

4.1.4.1 Non-Text Complexity

A non-text complexity algorithm could be a potential way to evaluate non-textual web elements to determine their accessibility status. According to WCAG 2.2, 12 success criteria are associated with non-text elements that validate the accessibility of web content. The determined criteria related to the non-text complexity are Audio/Video; Links; Display Orientation; Input field; Buttons; Headings; Header; Paragraph; Background Music; Keyboard access; Keyboard Character Key Shortcut; Search Field; Label; Dropdown Menu; Dialog box; Form; Status Message; Error Message; Error Suggestion. In Section 4.2.1.1, a detailed description of these web features with their description and implementation direction has been provided that has been incorporated into the proposed automated web content accessibility testing tool.

4.1.4.2 Text Complexity

To determine the accessibility of the webpage content in the context of textual complexity, the potential solution is to derive a text complexity algorithm that evaluates the HTML textual content feature through Natural Language Processing (NLP) to analyze each element associated with textual aspects, validate their accessibility status and determine the associated complexity regarding semantic manner. WCAG 2.2 stated that 35 success criteria are associated with textual elements, which are essential components to determining and ensuring accessibility and reducing the complexity of webpage surfing. The determined criteria related to textual complexity are Image; Pre-recorded/Live Audio and Video; Title; Words; Paragraph; Webpage Text; Buttons; Links; Headings and Labels; Language; Checkbox; Combo boxes. Section 4.2.1.1 provides a detailed description of these elements associated with textual aspects that were incorporated into the text complexity algorithm with their implementation direction.

4.1.4.3 Additional Criteria Validation

A number of studies have found that following web content accessibility guidelines alone is insufficient to address all problems pertaining to individuals with special needs [93][104]. Therefore, considering additional criteria along with accessibility guidelines for web content, it could be possible to present a more comprehensive picture of online accessibility. According to user and expert suggestions (can be found in Section 4.1.2), 17 criteria have been found that should be considered when developing the proposed tool in conjunction with the most recent version of the web content accessibility guidelines (WCAG 2.2). The identified additional criteria are Loading time; Paragraph length; Hyperlink ratio; Default Language; Webpage length; Server Status/Availability; User information; CAPTCHA; Multiple languages; Image ratio; Manual font and color adjustment option; Text Font family; Text Font size; Text pattern; Content type; and Number of audio/video content. A detailed description of these additional criteria has been provided in Section 4.2.1.1, which has been incorporated during the development of the proposed automated web accessibility testing tool.

4.1.5 Aspects of Accessibility Issues and Score Visualization

After performing the accessibility evaluation, overall evaluation report has been presented considering several statistics. In general, statistics are related to each algorithmic evaluation, the accessibility score of each type of disability, overall accessibility score with accessibility status and arbitrary information of the evaluated webpages.

Regarding the algorithmic evaluation, the first algorithm (Non-Text Complexity Analysis) evaluates the accessibility concerns of the webpage by considering its non-text components. It is able to assess 19 web objects in total, including their functionalities and other aspects. Similar to the first algorithm, the second algorithm (Tex-Complexity Analysis) analyzes all of the webpage's text components to determine how complicated or problematic they are from an accessibility standpoint. It is able to assess a total of 12 web objects considering the textual components. Following the implementation of these two algorithms, additional criteria were analyzed using the additional criteria assessment algorithm. It evaluates the accessibility perspective of 17 web objects. The algorithmic evaluation results were arranged into six different aspects considering Success Criteria, Conformance Level, Feedback, Result, Impairments Type and Improvement Direction. The algorithmic evaluation result is divided into six categories since in terms of accessibility support, most of the developed approaches [55][58][103] have no clear indication of the implemented techniques or success criteria and their conformance level. Therefore, users are unable to distinguish between accessibility features that have been implemented and those that are not covered by the developed approach. Also, without indicating the conformance level of each success criterion introduces difficulties in understanding the importance of a particular success criterion. Thus, to increase the effectiveness of the evaluation report, information on the conformance level is also provided with reference to the specific success criteria. After that, feedback regarding the evaluation status of each success criterion is given. Furthermore, in the context of result categorization, almost all the developed tools categorize the accessibility guidelines in terms of several terminologies such as passed, failed, cannot tell, known error, likely error, potential error, error, warning, success, and not applicable. These terminologies sometimes denote an uncertain outcome. For example, failed, cannot tell, error, warning and not applicable terminologies are considered a negative result. It represents that the guideline is either not fulfilled or not identified or difficult to identify. It can also mean that the webpage has structural defects or programmatic errors in the evaluation tool. Also, for likely error terminology, it is not clear whether it is an identified error or not. So, from these uncertain categorizations, it is difficult to understand the concluded accessibility score that could lead to misleading accessibility representation. Therefore, all the terminologies or assessment criteria are concisely categorized into PASSED, FAILED, NOT DETECTED, and NOT TESTED referring to each evaluated success criteria to calculate the accessibility score and to appropriately evaluate the accessibility status. Also, information is provided on each type of impairment related to each success criterion. This information indicates which group of individuals with specific needs, these particular success criteria are important to ensure that the web content is accessible to them. In the last, as majority of the developed tools don't indicate which success criteria can be implemented automatically, and which require manual or expert investigation as it is not possible to incorporate all the success criteria in an automated manner. Therefore, the textual improvement directions also offered to show which criteria the tool successfully validated and which

criteria need further verification or expert testing. It is recommended to do additional verification for FAILED success criteria and expert testing for NOT TESTED criteria.
Regarding the accessibility score for each type of disabilities, none of the selected tools provide the percentage of accessibility for each disability type, such as visual impairment, cognitive impairment, etc. It may be challenging for web practitioners to comprehend which types of disabilities were prioritized during the development of that particular webpage and which disability type needs to be prioritized for future improvement. Providing accessibility percentages for each type of disability can give a better understanding of how accessible a certain webpage is for a specific group of people. Furthermore, an overall accessibility score with accessibility status has also been added. Besides, summary of some arbitrary information is presented that helps to understand some basic information about the tested webpage such as page URLs, page title, total number of checked HTML elements, page size or length, and page loading time. These statistics help a wide array of people (i.e., end users, designers, developers, reactionaries, etc.) to understand the overall accessibility status of the tested webpage.

4.1.6 Proposed automated accessibility testing Framework validation

To validate the proposed framework, a comparative analysis has been performed with similar existing models considering several functional properties such as updated guidelines, user and expert criteria consideration, textual/non-textual component analysis, evaluated and not evaluation checkpoints feedback, overall accessibility score computation, and accessibility score computation for different types of disabilities. Regarding comparative assessment, the proposed framework is compared with existing eight (8) models that have been mentioned in state-of-the-art literature in terms of several functionalities as mentioned in Table 4.1.

TABLE 4.1: Comparative assessment results considering functional properties with existing model

References	**Assessment Functional Properties**						
Heading level	**C-1**	**C-2**	**C-3**	**C-4**	**C-5**	**C-6**	**C-7**
Jens Pelzetter (2018) [55]	(No)	(No)	(No)	(No)	(No)	(Yes)	(No)
Jens Pelzetter, (2020) [103]	(No)	(No)	(No)	(No)	(Yes)	(No)	(No)
Michailidou et al. (2021) [58]	(No)	(No)	(No)	(No)	(No)	(No)	(No)
Boyalakuntla et al. (2021) [54]	(Yes)	(No)	(No)	(No)	(No)	(No)	(No)
Raju Shrestha (2021) [61]	(No)	(No)	(No)	(No)	(No)	(No)	(No)
Robal et al. (2017) [67]	(No)	(No)	(No)	(No)	(No)	(Yes)	(No)
Hilera et al. (2016) [68]	(No)	(No)	(No)	(No)	(No)	(Yes)	(No)
Ingavélez-Guerra et al. (2018) [69]	(No)	(No)	(No)	(No)	(No)	(No)	(No)
Proposed Model	**(Yes)**	**(Yes)**	**(Yes)**	**(Yes)**	**(Yes)**	**(Yes)**	**(Yes)**

[*WCAG 2.2=C-1; *User requirements consideration=C-2; *Expert suggestion consideration=C-3; *Textual/non-textual component analysis=C-4; *Evaluated and not Evaluated Checkpoints feedback=C-5; *Overall Accessibility Score=C-6; *Accessibility Score for Disability type=C-7]

It depicts that for information regarding the updated version of accessibility guidelines, one model considered the updated guideline in their model (Boyalakuntla et al. [54]) and other models have no such concern. Concerning user and expert requirements and suggestions, none of the compared models were found to address such concerns in their evaluation process. Besides, for textual and non-textual component analysis, none of the models were found with such concern in their evaluation process. For evaluated and not evaluated guideline information, one model (Jens Pelzetter [103]) provides feedback about the evaluated and not evaluated checkpoints feedback. Even though other models did not compute the total accessibility score, three models (Jens Pelzetter [55], Robal et al. [67] and Hilera et al. [68]) were found to be concerned about the overall accessibility score into account. Lastly, concentrating on disability types in the computation of the accessibility score, none of the models generate an accessibility score for each type of disabilities. Out of all these factors, the proposed framework takes into account every element addressed in (Section 2.3) to enhance the accessibility assessment or evaluation outcome of a webpage. The hypothesis is that the proposed framework can offer a comprehensive and up-to-date view of webpage accessibility in comparison to other comparable models.

4.2 An Automated Tool Development and Implementation

Referring to the findings in Chapter 2, it is becoming clear that a modern and sophisticated web content accessibility testing tool is an emerging need to assess webpages to accurately reflect their current accessibility scenario. With this need taken into account, in this section, the primary objective is to answer the fourth research question (as presented in the following) by developing an automated web content accessibility testing tool by following the proposed accessibility evaluation framework described in 4.1 to evaluate web content to represent their accessibility status. The research question addressed in this section is the following.

R-Q4: How can we increase the effectiveness of an automated web content accessibility assessment tool focusing on advanced engineering techniques?

The detailed development and implementation process of the proposed tool are described in the following sections, where the development and implemented strategies are presented, including variable identification and algorithmic evaluation that have been followed to implement the proposed model as a tool for real-life experimentation. I titled the proposed and developed tool as Web Content Accessibility Evaluation Environment (WCAEE). The proposed tool has been addressed throughout the paper as WCAEE.

4.2.1 Web Content Accessibility Evaluation Environment (WCAEE) Tool Development

The literature suggests that a number of factors, such as the implementation of appropriate guidelines, the consideration of additional criteria, and the use of advanced engineering techniques, could facilitate the development process of automated web content accessibility testing tools [52][105][106]. By considering these aspects and facilitating the development process, it is crucial to propose an accessibility evaluation framework by considering an appropriate accessibility guideline, incorporating user and expert requirements and suggestions, and integrating guideline knowledge simplification to facilitate the appropriate accessibility evaluation score computation of the tested webpages.

Figure 4.8 shows the architecture of the first version of the developed WCAEE tool. It incorporated two separate algorithms for textual and non-textual web component analysis. After complexity analysis, it represented the generated textual complexity report, non-textual complexity report, and statistical report to the end user through three separate windows, which represent the overall accessibility score, accessibility score for each disability type, total number of passed, failed, not detected, and not tested guidelines.

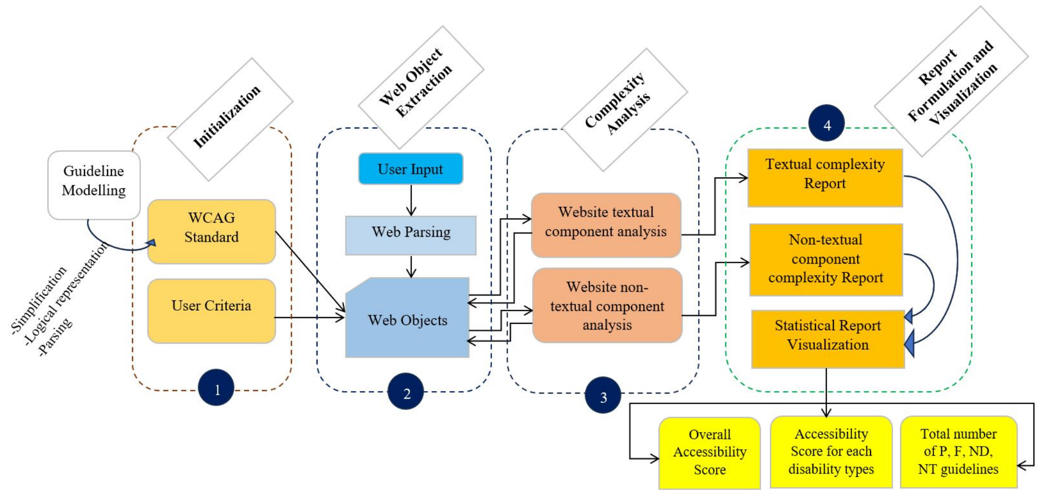


FIGURE 4.8: The system structure/views of the WCAEE tool (first version)

However, the first version of this model is not fully compatible in terms of computation time as all the textual, and non-textual criteria were validated through two algorithms that called several methods in multiple times which caused overloading. However, this issue can be resolved by implementing three separate algorithms for different web features. Also, issues found related to report representation, and visualization to the end users that reduce the effectiveness of the system. Therefore, to improve the performance and effectiveness of the first version of the model, a modified version is proposed that is considered to develop the final version of the WCAEE tool which is an advanced tool for evaluating web accessibility. It can be used as an automated tool to assess web content accessibility and determine its accessibility score.

The developed WCAEE tool consists of four phases, as shown in Figure 4.9. The process involves: (1) modeling the structural elements to facilitate the structuring of the guidelines and user requirements; (2) conducting a complexity analysis by loading the webpage and implementing three distinct complexity analysis algorithms; (3) formulating the report to provide an overall evaluation statistics that includes feedback and results related to each checkpoint of the implemented guidelines, as well as information on conformance level, impairment types, and improvement direction through four distinct window views; and (4) visualizing the report through various graphical analysis (graphs, charts, and other visual representations). The ensuing subsections provide a detailed description of the development process.

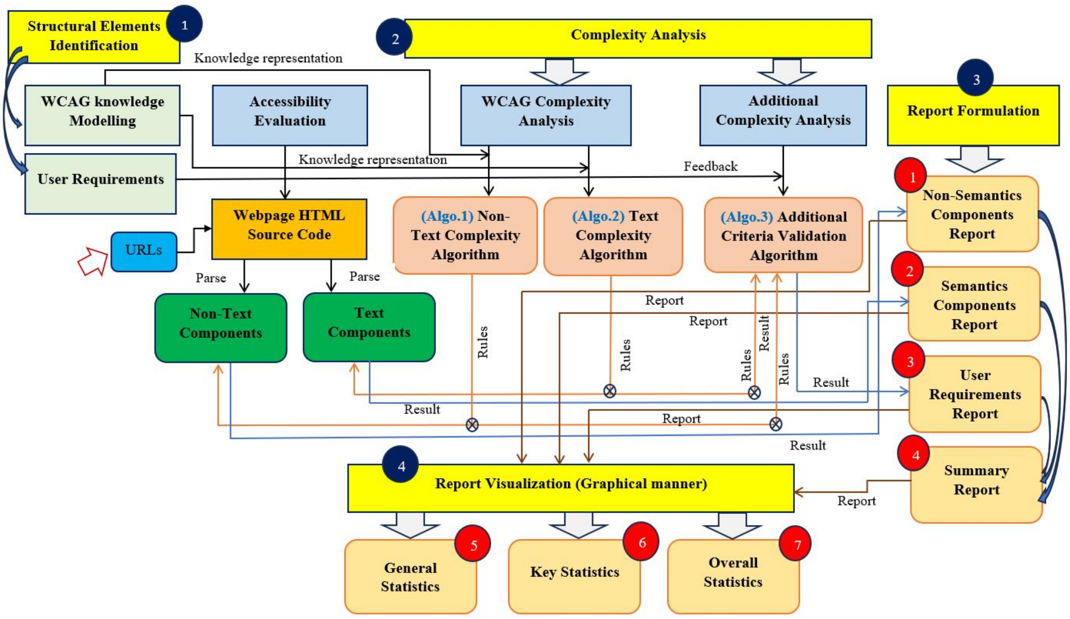


FIGURE 4.9: The system structure/views of the WCAEE tool (final version)

4.2.1.1 Structural elements: variable identification for complexity analysis

During the development process, standard web content accessibility guidelines and user assessments or requirements is considered for determining structural elements as these aspects provide a detailed view of important variables, also known as webpage objects, which are regarded as prime assets for evaluating webpage accessibility. The most recent version of the widely accepted web content accessibility guidelines (WCAG 2.2) is evaluated and a user study has been conducted to ascertain their requirements and assess the associated objects. In that case, it directs us to understand how web content accessibility guidelines and user requirements help to determine the structural elements to improve the accessibility issues or complexity.

The WCAG 2.2 assessment process emphasized the necessity of simplifying or modeling the guidelines because they are written in a natural language format and can be challenging to comprehend when attempting to develop evaluation tools or incorporate the guidelines into the evaluation process. As a result, AccGuideLiner is an advanced engineering tool (that has been described in Section 4.1.3) that is developed to simplify all 87 success criteria. This tool provides a complete view of simplified guidelines for a variety of criteria, including beneficiary type, evaluation type/phase, attribute, component type, requirements, and conformance level. The AccGuideLiner tool is one of the works that has been developed to support the WCAEE tool development. The Tkinter GUI toolkit was utilized in the development of AccGuideLiner within a Python environment. The details of the developed AccGuideLiner tool can be found in [30]. In guideline modeling, objects were referred to the specific webpage features that addressed particular success criteria; attributes indicated whether the variable was an HTML or CSS tag; the component type referred to textual (title, paragraph, etc.) and non-textual (header, footer, color, etc.) webpage features; requirements referred to information that was missing but could help to improve the issues that were detected; conformance level which referred to priority/importance of the success criteria; beneficiary type referred to the target audience or users connected with special requirements, and for whom that particular success criterion should be satisfied and evaluation type/phase referred to whether a specific success criterion can be evaluated automatically or requires manual evaluation. From guideline modeling, 31 objects or structural elements have been found that relate to 51 of the 87 WCAG 2.2 success criteria that are inherently evaluable. Based on expert knowledge, the remaining 36 success criteria need to be manually assessed. The identified 31 variables are classified into non-textual elements and textual elements. Besides, after assessing experts' suggestions, 17 additional variables have been identified that are crucial to consider in the proposed web accessibility evaluation process to address the identified criteria. All the determined non-textual, textual, and additional variables are described in the following. All the rules or directions about each addressed elements for web content accessibility evaluation are given in Table B.4 (Appendix B).

Non-Textual Elements

[Audio/Video]: When compared to text alone, audio and video are thought to be valuable mediums for representing information. Therefore, it is advised that audio and video files be included in the webpage to better convey the content and increase webpage usability and accessibility. In this work, the presence of any audio and video material was assessed through 'audio'/'video' elements with 'controls', and 'src' attributes.

[Links]: There are two types of hyperlinks, internal and external, that can be used to navigate webpages. Internal and external links are assessed in terms of their presence and assignment properly, as well as for any broken or unavailable conditions and appropriate length of observation. A maximum of 80 characters was taken into consideration when determining the link's length. Additionally, 'a' and 'link' elements with the 'href' property were used to analyze hyperlinks as well as links that point to other CSS files based on their name, role, and value.

[Display Orientation]: Referenced width is used to evaluate display orientation; for example, width should be set as device width, and the default or starting scale should be 1. The availability or visibility of webpages across a variety of platforms, including laptops, desktops, tablets, mobile devices, etc., can be enhanced by properly indicating orientation assignments.

[Input Field]: Input fields are the most common objects of webpages that require input based on their type. Therefore, a clearly stated purpose is essential to improve its usefulness to people and make it understandable and accessible. All the input fields of the webpages are evaluated through 'placeholder' and 'aria-label' attributes as they indicate or instruct the actual purpose of the input field.

[Buttons]: According to accessibility, buttons need to be labeled in a way that is accessible to those with disabilities such as people with visual impairments who use assistive technologies for navigating webpages. To evaluate the accessibility of buttons, it evaluated through the 'aria-label' attribute as this attribute specifies the button's purpose and makes the navigation process easier for people.

[Headings]: Headings are crucial components for accurately representing and enhancing the information's semantic quality. Inappropriate background and text color choices may make it challenging to correctly comprehend the information being displayed. Since red-black and red-green are the most challenging color combinations for individuals with special needs to navigate, thus the heading color was assessed in terms of these two color combinations.

[Header]: The header is one of the webpage's secondary core components, or characteristics, that is most crucial to accurately reflecting the content. Selecting the wrong background and text color can make it difficult to accurately understand the information that is being shown. Thus, all the headers were assessed based on the way the background and text colors blended. Similar to headings, all of the headers were assessed using two-color combinations, including red-black and red-green.

[Paragraph]: A paragraph is made up of several sentences put together to give the user all the information in the site content. Therefore, clearly representing this text is important. In majority of the time inaccessible color makes this text representation difficult for a specific group of people such as color-blind people or people who use assistive technology for navigating the webpages. Thus, the background and text colors of the paragraphs were assessed in relation to two color combinations, including red-black and red-green, which are reminiscent of the header.

[Background Music]: Background music is one of the responsible elements that contribute to accessibility issues. Therefore, it was advised by accessibility experts to avoid using any background music or audio to make web content accessible. Thus, background music was assessed using the HTML code's 'bgsound' element.

[Keyboard Access]: For people with special needs, having keyboard access is one of the most important requirements for an efficient online navigating experience. Keyboard access functionalities (Tab key) were evaluated through 'tabindex' global attribute with default 0 and -1 values for lang, title, link, div, header, ul, li, button, paragraph, heading, img, footer, and other attributes.

[Keyboard Character Key Shortcut]: People with special needs who use various assistive technologies to access webpages may encounter challenges if a webpage's content has designated keyboard character shortcuts which is referred to as a hidden trap. The 'accesskey' global attribute was used to assess keyboard character shortcuts for each element, including buttons, paragraphs, headings, labels, headers, footers, li, ul, and other elements.

[Search Field]: The search field is used in the webpages for consistent identification and navigation. Sometimes, search fields are not properly defined, and are not understandable and navigable, thus search field understandability was assessed using the 'placeholder' element and active status using the 'action' attribute. Besides, the 'role' attribute is used to assess the functionality of the status message.

[Label]: A descriptive, meaningful, and appropriate label description plays a crucial role in improving its accessibility. Among several techniques, a properly specified label name might be most effective and can improve its accessibility; thus, label name specification was assessed using the 'for' property on the 'label' element.

[Dropdown Menu]: Dropdown menus come in a variety of forms, including clicking and mouse hovering. However, in some cases, the dropdown menu's usefulness is diminished because assigned elements include missing information that causes the menu to behave improperly. In order to make sure that no information is missing for any of its elements and attributes, each element was examined through 'label', 'select' and 'option' using the 'value' attribute.

[Dialog Box]: The dialog box facilitates user involvement by providing instructions on how to resolve a particular problem, which is especially helpful when making decisions. In some cases, the dialog box returns null values due to missing information in the dialog box element. To confirm this, the 'dialog' element or tag was assessed considering its assigned information.

[Form]: Forms act as an important tool for collecting user information. Forms frequently malfunction because of a lack of instructions regarding desired input data. As a result, forms were assessed using two criteria: 'label' to specify the type of data that should be entered, and 'action' to indicate that the form is active and operating as intended. Additionally, the form's status message was assessed using a 'role' property, similar to a search field.

[Status Message]: Status message refers to the return message from the server during navigating the webpages. Sending or returning status messages from the server for every request represents the responsiveness of the server. Through the status message, the user is informed of the status of a particular action. In this system, the status message of all the input and search fields was considered to evaluate their accessibility. The 'role' attribute was considered while assessing the status message functionalities of any input fields or search fields to ascertain the functionalities they offered.

[Error Message]: If an error occurs, a message explaining the issue must be displayed to inform the user of the error's cause and suggest additional actions. To enable error generation, the defined error message needs to be suitable and accurately convey the command. To evaluate the error message accessibility, an error detection function, such as the true or false status of the defined 'aria-invalid' attribute of the 'input' element was considered to evaluate the error message.

[Error Suggestion]: According to WCAG, if an input error is automatically identified and the possible correction suggestions are detectable, then the automatic suggestions are provided to the user. However, among several solutions, some most common and effective tags used in HTML code can be useful to evaluate error suggestion functionalities. Thus, to evaluate the recommended error suggestion, the 'span' tag within the 'input' element was considered.

Textual Elements

[Image]: One of the essential components of web content that is regularly utilized to provide readers with descriptive information is an image. Images (Image, Gif, Animations, Logos, and decorative images) were evaluated through their alternative text whether it is assigned or not, as assistive technology users completely rely on it to comprehend the content of the images. This element was assessed by validating its alternate text using the 'image' element with the 'alt' attribute.

[Pre-recorded/Live Audio and Video]: These days, webpages usually have live or pre-recorded audio and video content attached in order to facilitate sharing information with the user. Users can effectively and valuably use the represented information using audio and video content. Information received by audio or video is more advantageous than information received through text, particularly for individuals with disabilities. Thus, all audio/video content (pre-recorded or radio webcasts) was assessed based on its accurate caption and a clear explanation that aids in resolving accessibility concerns. The pre-recorded/live Audio/Video was evaluated through ‘audio’/ ‘video’ elements with ‘audio controls’, ‘video controls’, and ‘alt’ attributes.

[Title]: Webpage titles also known as meta titles or title tags. The title of the webpage reflects its purpose to the user. In an HTML page, the title is defined by the title element. Thus, in the proposed system, the title was extracted using the ‘title’ element. As the title represents the purpose of a particular webpage, thus, to ascertain whether the title is accurate, descriptive, and appropriate, its presence was assessed and whether it follows a coherent or meaningful sequence.

[Words]: Words are the normal text that is used to represent the content or information of a webpage. The content or information presented should be clear and understandable in order to increase accessibility. Here, all the words of title and body content were evaluated in terms of their meaningfulness and word spacing ratio where the ratio was considered 0.16*12 pt as proper spacing between two words.

[Paragraph]: A paragraph is the body of a webpage which is also known as a descriptive explanation of the information. As this is one of the prime content to understand the purpose of a webpage, it should be meaningful, simple, precise, and contain useful but sufficient information that will help to understand the content. To evaluate their accessibility, the entire paragraph of a webpage has been extracted with the help of the ‘p’ element. Similar to the title, the body content was assessed based on the meaning sequence observation, which aided in determining its efficacy for the intended audience.

[Webpage Text]: The text on the webpage refers to the title and body of the text that is assessed using a number of semantic methodologies, including reading level, repetitions/redundancies, unusual words, acronyms, and mispronounced terms. The ability of the user to read the text without difficulty is referred to as their reading level, particularly for individuals with disabilities. Sustaining reading levels is crucial for enhancing web content accessibility. The ability to comprehend words or sentences without trouble is known as pronunciation. Sentences and meaningful words are essential for improving pronunciation ability. Complex and ambiguous words ought to be avoided in the text content of webpages. The shortened version of words or phrases is called an acronym. An example of an acronym is IT, which stands for information technology. While abbreviations might be useful in some contexts, using such condensed versions is inappropriate for accessibility because a person with impairments often struggles with these shorter forms of words or phrases. To grasp the meaning of an abbreviation, it is vital to supply a broad and extended form when it is unavoidable.

[Buttons]: The button is one of the advanced user interface components that help users interact with the system with a single tap. However, improper information such as non-clear functionalities makes the navigation process inaccessible. Therefore, buttons were evaluated in terms of their defined purpose through the ‘button’ tag using the ‘title’ attribute.

[Links]: Links are additional resources on a webpage that are intended to expand or improve the user's understanding of the content on the page. A webpage may include a number of links, both internal and external, to expand its content. All the internal and external links were assessed based on their stated purposes since these serve to illustrate the links' usefulness. The purpose of the link was assessed using the 'title' attribute on the 'href' element.

[Headings and Labels]: The headings and labels are the secondary vital aspects or qualities of the webpage that are most crucial for accurately expressing the information. A descriptive, meaningful, and appropriate description of headers and labels is suggested to enhance web accessibility and navigation process. All the headings and labels were assessed according to their stated purposes. The purpose was evaluated through their text component.

[Language]: Properly defined language is a basic requirement of any webpage. In some cases, developers prefer to define several sections or content through different languages hindering the consistency of the presented information. According to the accessibility perspective, using multiple languages on one webpage reduces the content's readability and consistency. Keeping a single language for all sections and paragraphs of the content is another crucial feature of accessibility that needs to be upheld. Thus, it is not advised to use different languages in content as this decreases webpage accessibility and makes a webpage less programmatically determinable. To ascertain whether the content is written in a single language or a blend of several languages, it was assessed using lang detector.

[Checkbox]: In the webpage, a square box with the ability to be checked or marked for any active operation is known as checkbox. In many cases, missing information raises challenges in deciphering the required input data, similar to a dropdown menu. To determine if any missing input data was discovered, the input data was assessed using 'input' and 'checkbox' components with 'value' attributes.

[Combo boxes]: A unique feature of webpages is the combo box, which lets users select an item from a vast list of possibilities to make it easier for them to find what they're looking for. Unfortunately, developers often forget to specify the correct name, which makes it challenging for those who use assistive technology. Accessibility problems are exacerbated when information in the presented list is missing. Thus, the 'input', 'select' and 'option' elements with their 'name' and 'value' properties were used to assess combo boxes based on their supplied name and item list.

Additional Elements

[Loading Time]: The average amount of time it takes for a webpage to load when a user searches or browses through their web address in the search panel is known as the webpage loading time. Webpage loading time acts as a crucial factor in webpage accessibility because users will become dissatisfied and webpage interactivity may be excluded if a page has problems and takes longer than usual to load. Thus, the webpage loading time was evaluated considering the standard or average loading time of 0.3 seconds.

[Paragraph Length]: A lengthy paragraph introduces difficulties in understanding web content for people with cognitive disability. To make the web content accessible to every group of people, it is recommended to limit its length to make it compressive to the user. Thus, the paragraph length was evaluated according to the standard ratio of 1500 words maximum.

[**Hyperlink Ratio**]**:** Hyperlinks are essential components of the webpage that give the user access to more information. However, individuals with special needs find it challenging to navigate webpages with an excessive number of hyperlinks in the text. Experts in accessibility estimated that no more than fifty hyperlinks are necessary to accurately convey the content and give the user all the information they need.

[**Default Language**]**:** Webpage default language was evaluated considering the English language of the webpage. As English is the global language, setting it as the default language on a webpage can increase its accessibility for the general public. The language of the webpage was evaluated through the 'lang' attribute to determine whether the webpage has an English version.

[**Webpage Length**]**:** The duration of a webpage includes both its content and display sizes, as well as navigation time. Similar to webpage loading time, webpage length plays a critical role in enhancing accessibility. People with disabilities may find it very difficult to use lengthy webpages due to mobility issues or cognitive difficulties that prevent them from viewing content for extended periods of time. Thus, enough information should be appropriately arranged within a set number of pages. According to the accessibility perspective, the webpage length was evaluated with a limit of 14 KB in mind.

[**Server Status/Availability**]**:** The purpose of webpage availability, also known as webpage uptime, is to ensure that users may view or navigate the page whenever they choose. In the event that a webpage is unavailable, its effectiveness could be diminished, and users could be less inclined to return often. Consequently, keeping a webpage up to date is essential to improving accessibility. The severe condition/availability was assessed in terms of being down, deactivated, or active. A key contributing factor to a webpage's accessibility reduction is that most users become disinterested in returning to it when the server is unavailable.

[**User Information**]**:** In general, user information relates to whether accessing a webpage involves logging in or registering. Certain webpages may ask for personal information from their users, including their location, email address, password, username, interests, etc. These make the webpage inaccessible as users might not consent to sharing their information and users with disabilities might misunderstand what these criteria actually entail. In that case, users might not want to continue this process to browse the page. Thus, login functionality is used to assess whether the webpage needed any login or registration information.

[**CAPTCH**]**:** Recently, CAPTCHA-enabled webpages have become increasingly popular, either for security reasons or to better understand user behavior. While some users find image-based CAPTCHA to be more helpful, others prefer text-based CAPTCHA. However, it is one of the hardest tasks for people with special needs which reduces webpage accessibility. To evaluate the accessibility of webpages in terms of the presence of CAPTCHA, the 'div' element with the 'id' attribute was used to determine whether CAPTCHA was present or being used.

[**Multiple Languages**]**:** Language is a unifying element of webpage that allows the community to access it from anywhere in the world. For a webpage to be user-accessible, it must be available in several languages and include a selection choice. To attain this goal, using 'nav' and 'ul' elements with 'li' and 'a' attribute, the multiple language option was assessed to see if the webpage specified multiple languages or not to make the webpage available to users of various languages.

[**Image Ratio**]: As previously mentioned, image is an essential component of a webpage. However, it may raise accessibility concerns, if the proper placement ratio is not maintained. Overuse of graphics or images in webpage content may make a webpage less accessible to users with cognitive impairments. Therefore, according to accessibility experts, a limit of 10 images can be a reasonable decision to preserve its accessibility from an accessibility standpoint.

[**Manual Font Adjustment Option**]: Enabling manual adjustments to text size or font size is a primary cause of decreased accessibility. As font type varies from user to user, thus the webpage was evaluated to see if it had a stated font adjustment option that allowed users to manually change the font to suit their preferences.

[**Manual Color Adjustment Option**]: Similar to font type, the type of color also varies from user to user. So, the evaluation was performed whether the page allowed the color adjustment option to be adjusted manually in terms of the user interest.

[**Text Font Family**]: An appropriate font family should be used as the default font family for webpage content in order to make the content accessible to people with all types of disabilities. Among several font families, only a few of them are considered as accessible to make the content accessible to every group of users. Thus, web content text and heading text were evaluated in terms of six specific font families such as Tahoma, Calibri, Helvetica, Arial, Verdana, and Times New Roman.

[**Text Font Size**]: To ensure content accessibility, the default font size of webpage content needs to be appropriate for all categories of individuals with disabilities. The accessibility expert stated that text font sizes should be 16 px (12 pt) or larger, thus the textual content's (header and body text) accessible font size was assessed by comparing it to the ($\geq$ 16 px) ratio.

[**Text Pattern**]: Text pattern refers to different styles of text representation such as italic, bold, etc. Unsuitable textual conventions may make the content challenging to read. The 'italic' text pattern can be perplexing to certain people. As a result, using suitable text patterns may make the content easier to understand and more accessible. The text pattern was evaluated considering 'bold', 'strong', 'Italic', 'emphasized', 'marked', 'subscript' and 'superscript' formats through 'b', 'strong', 'i', 'em', 'mark', 'sub' and 'sup' attributes.

[**Content Type**]: Content type refers to web content that should have a combination of textual content, image content, and video or audio content. To represent web content effectively to people with special needs, following the appropriate content type is essential to improve its accessibility to every group of people.

[**Number of Audio/Video Content**]: Similar to the image content, an excessive amount of audio and video content might reduce the accessibility of the web content. Therefore, in the expert's opinion, a maximum of 2 audio or videos is the optimal number of audio/video content that can improve the accessibility of the web content.

4.2.1.2 Accessibility Evaluation: complexity analysis of structural elements

As it is illustrated in Figure 4.9, after finalizing the structural elements, algorithmic evaluation was carried out to analyze the complexity of particular web elements considering the accessibility perspective. The developed tool technically performs the accessibility evaluation based on three different algorithms: (Algo.1) Non-Text Complexity Analysis Algorithm (see 4.2.1.2); (Algo.2) Text Complexity Analysis Algorithm (see 4.2.1.2); and (Algo.3) Additional Criteria Validation Algorithm (see 4.2.1.2). The Text Complexity Analysis Algorithm assessed the complexity of the webpage's textual components, while the Non-Text Complexity Analysis Algorithm assessed the difficulty of the non-textual elements of the tested webpage. Besides, the Additional Criteria Validation Algorithm highlights the complexity of the interactive components of the tested page.

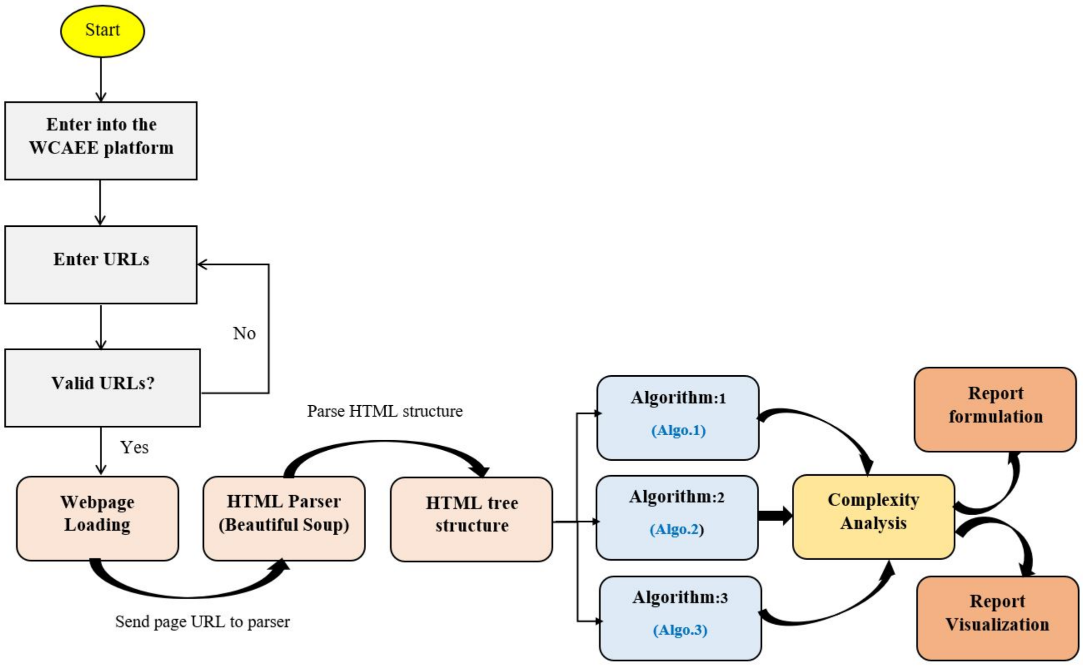


FIGURE 4.10: The work-flow diagram of the algorithmic evaluation process

Figure 4.10 illustrates the workflow of the algorithmic evaluation process. To evaluate the accessibility of a particular webpage, the web page's URL is first validated in the WCAEE platform to determine its accessibility. Once the URL has been validated, it is sent to an HTML parser (in this case, Beautiful Soup) to parse the HTML code and convert it into the HTML Document Object Model (DOM) structure. The full HTML source code is represented by the DOM structure as a tree structure view, which may be utilized to navigate the HTML elements in accordance with the required HTML tags. To assess the complexity of the corresponding elements that refer to various online objects, three distinct algorithms have been implemented. The evaluation report is generated based on the results of the complexity analysis using three different algorithms. Besides, graphical representation (a number of data visualization techniques) is also used to make the report effective, interactive, and helpful to the end user.

For implementing the developed WCAEE model, sublime text editor has been considered as a development framework and Python programming language to write the script to implement the proposed algorithms. The Windows 10 version has been taken into consideration which running on the 8th generation Intel Core i7 processor. The Tkinter package has been utilized to design user interfaces and numerous window views, as it is the most often used Python graphical user interface (GUI) library that enables the efficient creation of GUI programs. Additionally, an object-oriented interface such as the Tk GUI toolkit enables the integration of several objects into the user interface, which aids in creating and representing an interactive user interface for the user. A number of Python libraries have been used, which are mentioned in Table B.5 (Appendix B), to traverse the tested webpage, parse the HTML code, extract the information such as elements and attributes, and carry out the full evaluation.

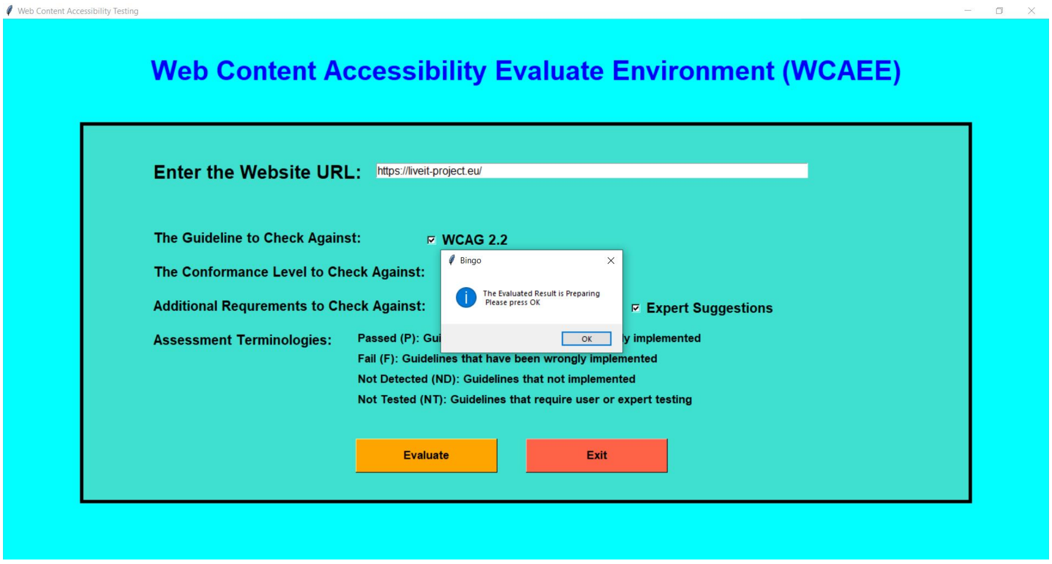


FIGURE 4.11: User interface of the WCAEE tool (URL validation)

Moreover, the user's initial step while using the WCAEE tool to assess a webpage's accessibility is to type or paste the webpage's URL into the URL validation screen. After that, the user must select/press the relevant button (Evaluate in orange) to proceed with the assessment procedure, as indicated in Figure 4.11. The user must click 'ok' to validate the warning after selecting the Evaluation button on the validation interface. Afterward, the URL of the page will be forwarded to the HTML parser, which translates the HTML source code into a tree structure including all of the element and attribute data. The three complexity analysis algorithms will then analyze each of the page's objects or characteristics and generate a report with details about the accessibility concerns and complexity of the tested page. For more information, a thorough overview of the three implemented algorithms is presented in the following sections.

■ Complexity Analysis Algorithms

Three separate complexity analysis algorithms have been implemented to develop the WCAEE tool and carry out the overall evaluation procedure. The first algorithm is called the Non-Text Complexity Analysis Algorithm, and it evaluates the accessibility concerns of the webpage by considering its non-text components. The Non-Text Complexity analysis algorithm is able to assess 19 web objects in total, including their functionalities and other aspects (see Section 4.2.1.1). Similar to the first algorithm, the second one is called the Tex-Complexity Analysis Algorithm and it analyzes all of the webpage's text components to determine how complicated or problematic they are from an accessibility standpoint. The Text-Complexity Analysis Algorithm is able to assess a total of 12 web objects (see Section 4.2.1.1) considering the textual components. After implementing these two algorithms, the additional criteria assessment algorithm has been used to analyze additional criteria that were received from user assessment. It considers 17 web objects (refer to Section 4.2.1.1) to assess their accessibility perspective. The following sections provide a detailed explanation of the developed algorithms along with their evaluation strategies.

- **Algo-1. Non-Text-Complexity Analysis Algorithm**

With particular attention to the elements and attributes of HTML objects, the Non-Text-Complexity analysis method navigates all web objects through HTML tags as it recursively explores the HTML tree view of the tested or provided webpage. As demonstrated in the **(Algo.1) Non-Text Complexity Analysis Algorithm** (Figure 4.12), it conducts the evaluation process by executing multiple auxiliary methods to assess the objects and determine the accessibility score of the page. As the algorithm shows, in lines (2-4), the algorithm accesses each node in the HTML tree view to navigate the elements and attributes of each node. In line (5), it counts all the objects of the webpage as the counted statistic is used frequently in the evaluation process. In lines (6-8), it evaluates the keyboard access considering 0 and 1 values for navigation order, and also the keyboard character key shortcut is evaluated, in lines (9-11). The background sound or any background audio files are evaluated through the bgsound element in the node, in lines (12-14). For evaluating the display orientation, it considers the device's default width and 1.0 as an initial scale to identify as proper display orientation, in lines (15-21). Furthermore, it assesses the audio and video files in lines (22-26) based on their source attribute to ascertain their presence, as improper source files cause significant accessibility concerns. In lines (27-31), Hyperlinks and CSS reference links are evaluated in terms of their missing resources and length as longer than 80 characters might cause accessibility issues. The defined purposes of all input fields, including error message boxes and search fields are assessed in lines (32-46). The normal input fields are assessed based on its aria-label and placeholder attributes; error message boxes are assessed based on their aria-invalid and span attributes to determine the error message and error suggestion information; and the search fields are assessed based on its placeholder, action, and role attributes to determine its defined purpose and executable functionalities. Also, lines (47–51) analyzed the buttons' purpose using the aria-label attribute. Besides, in lines (52-56), the color or headings, header, and paragraph are evaluated in terms of red-green and red-black color combinations as these are the most inaccessible colors according to the accessibility experts. To ascertain if the label has a clear name or purpose, in line (57–61), label name attribution is assessed. Lines (62–68) are used to evaluate Dropdown menu in terms of their assignment of missing values. The dialog box is also evaluated in lines (69-73), considering their missing value or information during the assignment. Lastly, all the forms of the tested webpage are evaluated in lines (74-80) considering their text information, active status, and execution functionalities. Finally, the process of exploring the HTML tree view is stopped in lines (81–82).

Algorithm1: Non-Text Complexity Analysis Algorithm

```
HTMLNonTextElements (Node node) {
foreach node, do
  extract DocumentElements () {
    foreach element in node
      count elements (audio, video, links, img, etc.)
      if ([tabindex] attribute is equal to [0] or [-1]), then //evaluate keyboard access
         return true;
      end
      if ([accesskey] attribute is not in elements), then //evaluate keyboard character key shortcut
         return true;
      end
      if (<bgsound> element is not in node), then //evaluate background music
         return true;
      end
      if (node is <meta> element), do //evaluate display orientation
         if ([name] attribute is equal to 'viewport'), do
            if ([content] attribute is equal to 'width=device-width, initial-scale=1.0'), then
               return true;
            end
         end
      end
      if (node is <audio> or <video> element), do //evaluate audio and video content
         if ([src] attribute is not null), then
            return true;
         end
      end
      if (node is <a> or <link> element), do //evaluate hyperlinks and CSS reference links
         if ([href] attribute is not empty && len[href]>80 character), then
            return true;
         end
      end
      if (node is <input> element), do //evaluate input field
         if ([placeholder] && [aria-label] attributes are not empty), then
            return true;
         end
         if ([aria-invalid] attribute is true && [span] attribute is not null), then //evaluate error message and error suggestions
            return true;
         end
         if ([type] attribute is equal to 'search'), do //evaluate the search field
            if ([placeholder] && [action] attributes are not empty), do //evaluate active status
               if ([role] attribute is defined), then //evaluate status message functionalities
                  return true;
               end
            end
         end
      end
      if (node is <button> element), do
         if ([aria-label] attribute is specified), then //evaluate button purpose
            return true;
         end
      end
      if (node is <h> or < header > or <p>), do //evaluate headings, header and paragraph color
         if ([style] attribute has text and background color except ['red', 'green'] and ['red', 'black']), then
            return true;
         end

      end
      if (node is <label> element), do //evaluate label
         if ([for] attribute is not empty), then
            return true;
         end
      end
      if (node is <select> element), do //evaluate dropdown menu
         if (node is <option> element), do
            if ([value] attribute is not null), then
               return true;
            end
         end
      end
      if (node is <dialog> element), do //evaluate dialog box
         if (item.text is not empty), then
            return true;
         end
      end
      if (node is <form> element), do //evaluate form
         if ([action] attribute && item.text is not empty), do
            if ([role] attribute is defined), then //evaluate status message functionalities of form
               return true;
            end
         end
      end
    end
end
```

FIGURE 4.12: Webpages non-text components analysis algorithm used by the proposed WCAEE tool.

- **Algo-2. Text-Complexity Analysis Algorithm**

The Text-Complexity analysis algorithm, like Algo.1, traverses the HTML tree view of the tested web page recursively to navigate all the elements and attributes of HTML objects. It assesses every textual element that is described in Section 4.2.1.1. The Text-Complexity Algorithm recursively identifies all the elements and their associated attributes (depending on the requirements) according to the selected variables chosen for textual components and performs the evaluation process, as the aim of this algorithm is to evaluate only the textual objects of the tested webpage. The **(Algo.2) Text Complexity Analysis Algorithm** (Figure 4.13) comprises multiple auxiliary techniques that assess multiple web objects to determine the page's accessibility score. As the algorithm shows, in lines (2-4), it traverses the HTML tree view and accesses all the nodes to identify the elements and attributes of the tree view. In lines (5-9), it accesses all the image objects of the tested webpage using the img element and evaluates them in terms of their alternative text property. In lines (10-14), similar to the image alternative text property, the audio and video content are also evaluated considering a similar perspective. Lines (15–19) test the title of the webpage in terms of how accurately it describes and aids the user in understanding the webpage's goal. In lines (20-56), it retrieves the title and each paragraph of the webpage and assesses the textual content or text information considering several aspects. Accordingly, in lines (21–24), the spacing within the text or between words is assessed using the 0.16*12 ratio, where 12 is the standard font size for all web content and 0.16 is a constant that indicates that the amount of space should be 0.16 times greater than the font size. Tokenization and stop word removal techniques are used from the nltk platform to preprocess the text in lines (25–29). To make the text cleaner, also excluded special characters, digits, and annotations. After that, the meaningful sequence of the words is checked by applying the language tool-python library. The word occurrences have been counted in lines (30–33) to identify any duplicate words because they can reduce the accessibility of the text. Additionally, the Python language identification library langdetect is used to identify the text language in lines (34–37) and count the identified language objects to determine the presence of multiple languages. Additionally, the brown corpus is used in lines (38–41) to detect unique words; in lines (42-45), regular expression is considered for abbreviation recognition; and in lines (46–49), the words and wordnet corpus is used from the nltk platform to identify mispronounced words. Additionally, to compute the readability of the text, in lines (50-55), 8 readability observation statistics have been applied and considered their average score to determine the readability of the evaluated webpage text. Also, assessment of the assigned title property of the button, hyperlinks, and CSS reference links is performed in lines (57–61) to ascertain whether they have appropriate titles by referring to their precise purpose. All the function of headings and labels is assessed in lines (62–66) based on their textual characteristics. The webpage language is evaluated in lines (67–71) using their lang property. All of the comboboxes and checkboxes are assessed on lines (72–85) based on the presence of their name and value attributes. Lines (86-87) are the closing of the nodes and elements of the nodes.

Algorithm2: Text Complexity Analysis Algorithm

```
HTMLTextElements (Node node) {
foreach node, do
  extract DocumentElements () {
    foreach element in node
      if (node is <img> element), do   //evaluate image
         if ([alt] is not empty), then
            return true;
         end
      end
      if (node is <audio> or <video> element), do   //evaluate audio and video content
         if ([alt] attribute is not null), then
            return true;
         end
      end
      if (node is <title> element), do   //evaluate title content
         if (item.text is not null), then
            return true;
         end
      end
      if (node is <title> or <p> element), do   //evaluate webpage title and paragraph
         calculate the space between words (); //checking text spacing
         if (space >= 0.16*12), then
            return true;
         end
         do the preprocessing [tokenization; stop words removal; remove special characters, numbers, and notations]
         match words () with language-tool-python library; //checking words meaningful sequence
         if (matches is True), then
            return true;
         end
         count words occurrence (); //checking duplicate words
         if (word_occurrence >1), then
            return true;
         end
         detect words language () using langdetect language detection library; //checking text language
         if (langdetect is True && langdetect.count () is equal to1), then
            return true;
         end
         match words () with brown corpus; //checking unusual words
         if (matches is True), then
            return true;
         end
         match words () pattern through regular expressions; //checking abbreviation
         if (matches is False), then
            return true;
         end
         match words () with words and wordnet corpus; //checking mispronounced words
         if (matches is True), then
            return true;
         end
         readability_statistis= [flesch_kincaid(), flesch(), dale_chall(), ari(), coleman_liau(), gunning_fog(), spache(),
linsear_write()]
         compute words () readability through readability_statistis; //checking text readability
         compute the average_readability_level;
         if (average_readability_level >=9), then
            return true;
         end
      end
      if (node is <button> or <href> or <link> elements), do
         if ([title] attribute is specified), then   //evaluate button, hyperlinks and links title
            return true;
         end
      end
      if (node is <h> or < label > elements), do //evaluate headings, and label's purpose
         if (item.text is not empty), then
            return true;
         end
      end
      if (node is <html> element), do   //evaluate webpage language
         if ([lang] attribute is specified), then
            return true;
         end
      end
      if (node is <input> element), do
         if ([type] attribute is equal to 'checkbox'), do   //evaluate checkbox
            if ([value] attribute is not null), then
               return true;
            end
         end
         if (node is <select> element), do //evaluate combobox
            if (node is <option> element), do
               if ([name] attribute is not null && [value] attribute or item.text is not empty), then
                  return true;
               end
            end
         end
      end
  end
end
```

FIGURE 4.13: Webpages textual components analysis algorithm that used by the proposed WCAEE tool

- **Algo-3. Additional Criteria Validation Algorithm**

Algorithm3: Additional Criteria Validation Algorithm

```
calculate (node) {
  web_loading_time = end_time - start_time; //checking loading time
    if (web_loading_time <=0.3 sec), then
        return true;
    end
  calculate web_page_length () in KB; //checking webpage length
    if (web_page_length <=14 KB), then
        return true;
    end
  check web_page_ response.status_code; //checking severe status
    if (web_page_ response.status_code == 200), then
        return true;
    end;}
counts (node) {
  count textual words (); //checking text length
    if (word_count <=1500), then
        return true;
    end
  count hyperlinks (); //checking hyperlinks ratio
    if (hyperlinks <=50), then
        return true;
    end
  count images (); //checking image ratio
    if (images <=10), then
        return true;
    end
  count audios () && videos (); //checking audio/video ratio
    if (audios >=1 to <=2 or videos >=1 to <=2), then
        return true;
    end;}
HTMLInteractiveElements (Node node) {
foreach node, do
  extract DocumentElements () {
    foreach element in node
      if (node is <title> or <p> elements), do   //evaluate default language
        if ([lang] attribute is English), then
          return true;
        end
      end
      if (node is <label> elements), do   //evaluate user requirements
        if (element.text is not equal to 'Username' && 'Password'), then
          return true;
        end
      end
      if (node is <div> elements), do   //evaluate CAPTCHA
        if ([id] attribute is not equal to 'captchaBlock'), then
          return true;
        end
      end
      if (node is <nav> && <ul> && <li> && <a> elements), do   //evaluate multiple language option
        if([onclick] attribute is not empty), do
          count its occurrence ();
          if (occurrence >=2), then
            return true;
          end
        end
      end
      if (node is <p> or <h>elements), do   //evaluate font-family
        if ('font-family' is 'Tahoma' or 'Calibri' or 'Helvetica' or 'Arial' or 'Verdana' or 'Times New Roman'), then
          return true;
        end
      end
      if (node is <p> or <h>elements), do   //evaluate font-size
        if ('font-size' is '16px' or '17px' or '18px' or '19px' or '20px'), then
          return true;
        end
      end
      if (node is <p>elements), do   //evaluate text patern
        if (element.text has no <b> or <strong> or <i> or <em> or <mark> or <sub> or <sup> elements), then
          return true;
        end
      end
      if (<p> && <img> && <video> elements in node) //checking content type
        return true;
      end
    end
end;}}
```

FIGURE 4.14: Webpages additional components validation algorithm that uses the proposed WCAEE tool

The additional criteria validation algorithm performs a complexity analysis of interactive elements of the tested webpage through two important methods and a set of auxiliary methods. The first method 'calculate (node)' is a recursive method that performs arbitrary object evaluation. Referring to **(Algo.3) Additional Criteria Validation Algorithm** (Figure 4.14), in lines (2-13), webpage loading time, length, and server active status are evaluated in terms of a specified ratio. The process of calculating the webpage loading time is demonstrated in lines (2-5), webpage length is demonstrated in lines (6-9), and the server's active status is checked in lines (10-13). Similarly, the second method 'count (node)' counts page textual words, hyperlinks, images, audio, and videos as shown by lines (15-30). Once the elements have been counted, it is evaluated according to the determined ratio that has been designated as an accessible ratio. By iteratively going through each node in the HTML tree, this approach counts the number of page elements. From the auxiliary methods, webpage default language is evaluated as English in lines (35-39). Using the 'level' and 'div' elements in lines (40–49), the requirements for user data and CAPTCHA were also assessed. Additionally, the 'nav' element is used to evaluate the multiple language options to see if the webpage has specified multiple languages to provide multilingual access in lines (50–57). Additionally, the font size and font family are assessed on lines (58–67). Five font sizes and six widely recognized and accessible font families have been taken into account while assessing the text on the webpage. For text patterns, such as whether the text has any special text pattern that makes the content difficult for the user to understand, seven distinct patterns have been taken into consideration in lines (68-72). Finally, the content types such as the three major content types should be considered as mandatory types for any web content evaluated in lines (73-75). Lines (76-77) are the endings of the HTML tree node traversing.

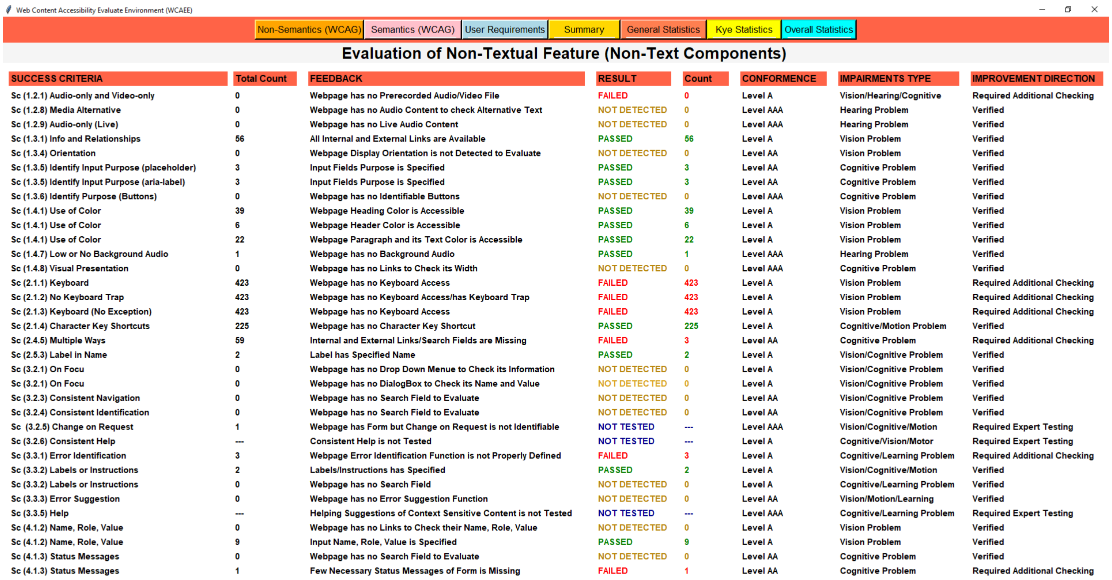
4.2.1.3 Accessibility report: evaluation report formulation of structural elements

This section aims to represent the outcome of each complexity analysis algorithm for non-textual, textual, and interactive elements. After applying the three algorithms and performing the complexity analysis, the results of Algo.1 and Algo.2 were organized into eight different aspects through eight columns with titles 'Success Criteria', 'Total Count', 'Feedback', 'Result', 'Count', 'Conformance Level', 'Impairments Type' and 'Improvement Direction' (as shown in Figure 4.15 and Figure 4.16). Besides, Algo.3 was structured into seven different aspects similar to Algo.1 and Algo.2 except for the Conformance level as shown in Figure 4.17. To classify conformance levels based on user requirements is difficult and may be inaccurate because it necessitates the involvement of accessibility professionals and a high degree of accessibility expertise. Figure 4.18 illustrates how the formulation of the assessment report is carried out via four distinct pages or window views, which are as follows:

Non-semantic, Semantic, and User Requirement Page: The output of the Non-Text Complexity Analysis Algorithm, Text Complexity Analysis Algorithm, and Additional Criteria Validation Algorithm for a tested website: https://liveit-project.eu/, is displayed on the non-semantic, semantic, and user requirement page, which is represented by Figure 4.15, 4.16 and 4.17. As shown in these figures, to structure and represent the output of each algorithm or evaluation result, the computed complexity result is first presented to the user by structuring with particular WCAG 2.2 success criteria that are relevant to or refer to the particular web objects (first column).

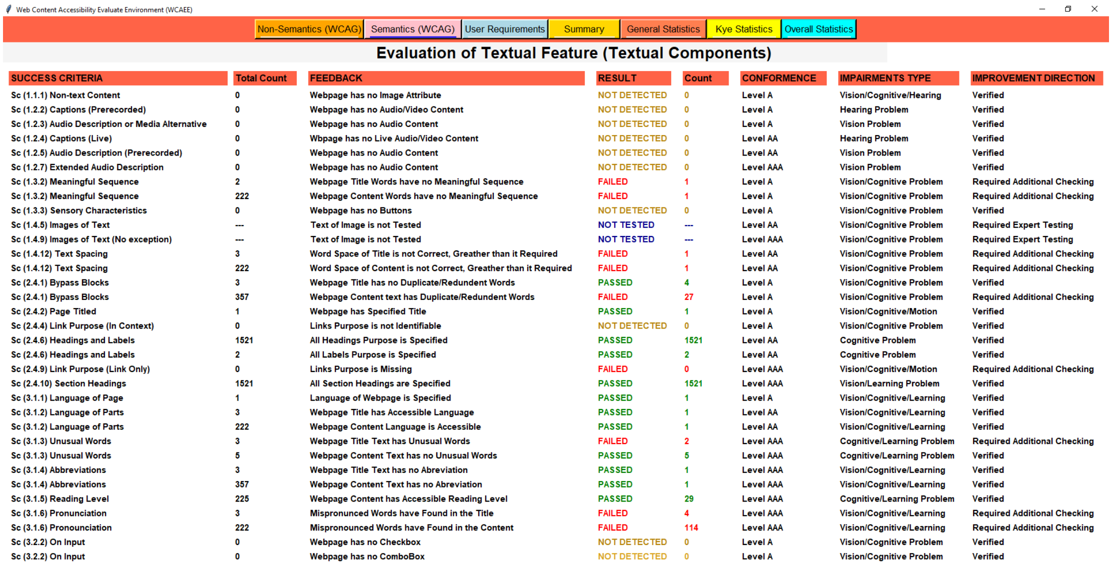
In the second, the total quantity of the particular items is displayed that were counted throughout the assessment procedure (second column). After that, it gave feedback regarding the evaluation status (third column). The evaluation result was then given in terms of PASSED, FAILED, NOT DETECTED, and NOT TESTED along with the number of objects that aligned with the result among the total counted objects (fourth and fifth column). To increase the effectiveness of the evaluation report, also information on the conformance level with reference to the specific success criteria is provided in the sixth column, as well as information on the types of impairments.

This information indicates for which group of individuals with specific needs, this particular success criteria are important to ensure that the web content is accessible to them (seventh column). It helps the user (both designer, developer, and end-user) to understand the importance of each success criterion. In the last, textual improvement directions are offered that show which criteria the tool is successfully able to validate and which criteria need further verification or expert testing. It is recommended to do additional verification for FAILED success criteria and expert testing for NOT TESTED criteria. Moreover, Figure 4.15, Figure 4.16 and Figure 4.17 depicts that **Algo.1 (Non-Text Complexity Analysis Algorithm)** can evaluate 27 success criteria (1.2.1; 1.2.8; 1.2.9; 1.3.1; 1.3.4; 1.3.5; 1.3.6; 1.4.1; 1.4.7; 1.4.8; 2.1.1; 2.1.2; 2.1.3; 2.1.4; 2.4.5; 2.5.3; 3.2.1; 3.2.3; 3.2.4; 3.2.5; 3.2.6; 3.3.1; 3.3.2; 3.3.3; 3.3.5; 4.1.2; 4.1.3), **Algo.2 (Text Complexity Analysis Algorithm)** can evaluate 24 success criteria (1.1.1; 1.2.2; 1.2.3; 1.2.4; 1.2.5; 1.2.7; 1.3.2; 1.3.3; 1.4.5; 1.4.9; 1.4.12; 2.4.1; 2.4.2; 2.4.4; 2.4.6; 2.4.9; 2.4.10; 3.1.1; 3.1.2; 3.1.3; 3.1.4; 3.1.5; 3.1.6; 3.2.2) and **Algo.3 (Additional Criteria Validation Algorithm)** can evaluate an additional 18 criteria which by following it is possible to improve the accessibility of the webpage. Besides, after formulating the evaluation result incorporating three different algorithms, the evaluation result is presented in the form of a summary report as shown in Figure 4.18.



SUCCESS CRITERIA	Total Count	FEEDBACK	RESULT	Count	CONFORMENCE	IMPAIRMENTS TYPE	IMPROVEMENT DIRECTION
Sc (1.2.1) Audio-only and Video-only	0	Webpage has no Prerecorded Audio/Video File	FAILED	0	Level A	Vision/Hearing/Cognitive	Required Additional Checking
Sc (1.2.8) Media Alternative	0	Webpage has no Audio Content to check Alternative Text	NOT DETECTED	0	Level AAA	Hearing Problem	Verified
Sc (1.2.9) Audio-only (Live)	0	Webpage has no Live Audio Content	NOT DETECTED	0	Level AAA	Hearing Problem	Verified
Sc (1.3.1) Info and Relationships	56	All Internal and External Links are Available	PASSED	56	Level A	Vision Problem	Verified
Sc (1.3.4) Orientation	0	Webpage Display Orientation is not Detected to Evaluate	NOT DETECTED	0	Level AA	Vision Problem	Verified
Sc (1.3.5) Identify Input Purpose (placeholder)	3	Input Fields Purpose is Specified	PASSED	3	Level AA	Cognitive Problem	Verified
Sc (1.3.5) Identify Input Purpose (aria-label)	3	Input Fields Purpose is Specified	PASSED	3	Level AA	Cognitive Problem	Verified
Sc (1.3.6) Identify Purpose (Buttons)	0	Webpage has no Identifiable Buttons	NOT DETECTED	0	Level AAA	Cognitive Problem	Verified
Sc (1.4.1) Use of Color	39	Webpage Heading Color is Accessible	PASSED	39	Level A	Vision Problem	Verified
Sc (1.4.1) Use of Color	6	Webpage Header Color is Accessible	PASSED	6	Level A	Vision Problem	Verified
Sc (1.4.1) Use of Color	22	Webpage Paragraph and its Text Color is Accessible	PASSED	22	Level A	Vision Problem	Verified
Sc (1.4.7) Low or No Background Audio	1	Webpage has no Background Audio	PASSED	1	Level AAA	Hearing Problem	Verified
Sc (1.4.8) Visual Presentation	0	Webpage has no Links to Check its Width	NOT DETECTED	0	Level AAA	Cognitive Problem	Verified
Sc (2.1.1) Keyboard	423	Webpage has no Keyboard Access	FAILED	423	Level A	Vision Problem	Required Additional Checking
Sc (2.1.2) No Keyboard Trap	423	Webpage has no Keyboard Access/has Keyboard Trap	FAILED	423	Level A	Vision Problem	Required Additional Checking
Sc (2.1.3) Keyboard (No Exception)	423	Webpage has no Keyboard Access	FAILED	423	Level A	Vision Problem	Required Additional Checking
Sc (2.1.4) Character Key Shortcuts	225	Webpage has no Character Key Shortcut	PASSED	225	Level A	Cognitive/Motion Problem	Verified
Sc (2.4.5) Multiple Ways	59	Internal and External Links/Search Fields are Missing	FAILED	3	Level AA	Cognitive Problem	Required Additional Checking
Sc (2.5.3) Label in Name	2	Label has Specified Name	PASSED	2	Level A	Vision/Cognitive Problem	Verified
Sc (3.2.1) On Focu	0	Webpage has no Drop Down Menue to Check its Information	NOT DETECTED	0	Level A	Vision/Cognitive Problem	Verified
Sc (3.2.1) On Focu	0	Webpage has no DialogBox to Check its Name and Value	NOT DETECTED	0	Level A	Vision/Cognitive Problem	Verified
Sc (3.2.3) Consistent Navigation	0	Webpage has no Search Field to Evaluate	NOT DETECTED	0	Level AA	Vision/Cognitive Problem	Verified
Sc (3.2.4) Consistent Identification	0	Webpage has no Search Field to Evaluate	NOT DETECTED	0	Level AA	Vision/Cognitive Problem	Verified
Sc (3.2.5) Change on Request	1	Webpage has Form but Change on Request is not Identifiable	NOT TESTED	---	Level AAA	Vision/Cognitive/Motion	Required Expert Testing
Sc (3.2.6) Consistent Help	---	Consistent Help is not Tested	NOT TESTED	---	Level A	Cognitive/Vision/Motor	Required Expert Testing
Sc (3.3.1) Error Identification	3	Webpage Error Identification Function is not Properly Defined	FAILED	3	Level A	Cognitive/Learning Problem	Required Additional Checking
Sc (3.3.2) Labels or Instructions	2	Labels/Instructions has Specified	PASSED	2	Level A	Vision/Cognitive/Motion	Verified
Sc (3.3.2) Labels or Instructions	0	Webpage has no Search Field	NOT DETECTED	0	Level A	Cognitive/Learning Problem	Verified
Sc (3.3.3) Error Suggestion	0	Webpage has no Error Suggestion Function	NOT DETECTED	0	Level AA	Vision/Motion/Learning	Verified
Sc (3.3.5) Help	---	Helping Suggestions of Context Sensitive Content is not Tested	NOT TESTED	---	Level AAA	Cognitive/Learning Problem	Required Expert Testing
Sc (4.1.2) Name, Role, Value	0	Webpage has no Links to Check their Name, Role, Value	NOT DETECTED	0	Level A	Vision Problem	Verified
Sc (4.1.2) Name, Role, Value	9	Input Name, Role, Value is Specified	PASSED	9	Level A	Vision Problem	Verified
Sc (4.1.3) Status Messages	0	Webpage has no Search Field to Evaluate	NOT DETECTED	0	Level AA	Cognitive Problem	Verified
Sc (4.1.3) Status Messages	1	Few Necessary Status Messages of Form is Missing	FAILED	1	Level AA	Cognitive Problem	Required Additional Checking

FIGURE 4.15: The view of the WCAEE tool for the tested LIVE IT webpage (Non-Textual elements)



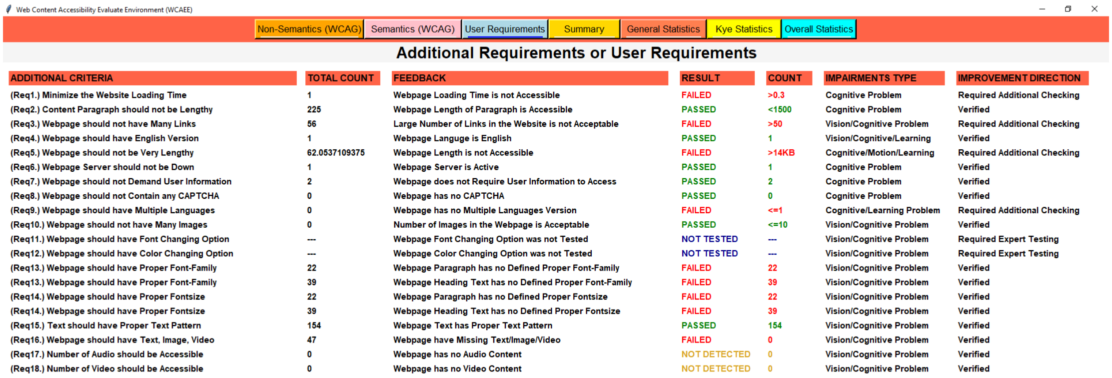
Web Content Accessibility Evaluate Environment (WCAEE)

Non-Semantics (WCAG) | Semantics (WCAG) | User Requirements | Summary | General Statistics | Kye Statistics | Overall Statistics

Evaluation of Textual Feature (Textual Components)

SUCCESS CRITERIA	Total Count	FEEDBACK	RESULT	Count	CONFORMENCE	IMPAIRMENTS TYPE	IMPROVEMENT DIRECTION
Sc (1.1.1) Non-text Content	0	Webpage has no Image Attribute	NOT DETECTED	0	Level A	Vision/Cognitive/Hearing	Verified
Sc (1.2.2) Captions (Prerecorded)	0	Webpage has no Audio/Video Content	NOT DETECTED	0	Level A	Hearing Problem	Verified
Sc (1.2.3) Audio Description or Media Alternative	0	Webpage has no Audio Content	NOT DETECTED	0	Level A	Vision Problem	Verified
Sc (1.2.4) Captions (Live)	0	Wbpage has no Live Audio/Video Content	NOT DETECTED	0	Level AA	Hearing Problem	Verified
Sc (1.2.5) Audio Description (Prerecorded)	0	Webpage has no Audio Content	NOT DETECTED	0	Level AA	Vision Problem	Verified
Sc (1.2.7) Extended Audio Description	0	Webpage has no Audio Content	NOT DETECTED	0	Level AAA	Vision Problem	Verified
Sc (1.3.2) Meaningful Sequence	2	Webpage Title Words have no Meaningful Sequence	FAILED	1	Level A	Vision/Cognitive Problem	Required Additional Checking
Sc (1.3.2) Meaningful Sequence	222	Webpage Content Words have no Meaningful Sequence	FAILED	1	Level A	Vision/Cognitive Problem	Required Additional Checking
Sc (1.3.3) Sensory Characteristics	0	Webpage has no Buttons	NOT DETECTED	0	Level A	Vision/Cognitive Problem	Verified
Sc (1.4.5) Images of Text	---	Text of Image is not Tested	NOT TESTED	---	Level AA	Vision/Cognitive Problem	Required Expert Testing
Sc (1.4.9) Images of Text (No exception)	---	Text of Image is not Tested	NOT TESTED	---	Level AAA	Vision/Cognitive Problem	Required Expert Testing
Sc (1.4.12) Text Spacing	3	Word Space of Title is not Correct, Greather than it Required	FAILED	1	Level AA	Vision/Cognitive Problem	Required Additional Checking
Sc (1.4.12) Text Spacing	222	Word Space of Content is not Correct, Greather than it Required	FAILED	1	Level AA	Vision/Cognitive Problem	Required Additional Checking
Sc (2.4.1) Bypass Blocks	3	Webpage Title has no Duplicate/Redundent Words	PASSED	4	Level A	Vision/Cognitive Problem	Verified
Sc (2.4.1) Bypass Blocks	357	Webpage Content text has Duplicate/Redundent Words	FAILED	27	Level A	Vision/Cognitive Problem	Required Additional Checking
Sc (2.4.2) Page Titled	1	Webpage has Specified Title	PASSED	1	Level A	Vision/Cognitive/Motion	Verified
Sc (2.4.4) Link Purpose (In Context)	0	Links Purpose is not Identifiable	NOT DETECTED	0	Level A	Vision/Cognitive Problem	Verified
Sc (2.4.6) Headings and Labels	1521	All Headings Purpose is Specified	PASSED	1521	Level AA	Cognitive Problem	Verified
Sc (2.4.6) Headings and Labels	2	All Labels Purpose is Specified	PASSED	2	Level AA	Cognitive Problem	Verified
Sc (2.4.9) Link Purpose (Link Only)	0	Links Purpose is Missing	FAILED	0	Level AAA	Vision/Cognitive/Motion	Required Additional Checking
Sc (2.4.10) Section Headings	1521	All Section Headings are Specified	PASSED	1521	Level AAA	Vision/Learning Problem	Verified
Sc (3.1.1) Language of Page	1	Language of Webpage is Specified	PASSED	1	Level A	Vision/Cognitive/Learning	Verified
Sc (3.1.2) Language of Parts	3	Webpage Title has Accessible Language	PASSED	1	Level AA	Vision/Cognitive/Learning	Verified
Sc (3.1.2) Language of Parts	222	Webpage Content Language is Accessible	PASSED	1	Level AA	Vision/Cognitive/Learning	Verified
Sc (3.1.3) Unusual Words	3	Webpage Title Text has Unusual Words	FAILED	2	Level AAA	Cognitive/Learning Problem	Required Additional Checking
Sc (3.1.3) Unusual Words	5	Webpage Content Text has no Unusual Words	PASSED	5	Level AAA	Cognitive/Learning Problem	Verified
Sc (3.1.4) Abbreviations	3	Webpage Title Text has no Abreviation	PASSED	1	Level AAA	Vision/Cognitive/Learning	Verified
Sc (3.1.4) Abbreviations	357	Webpage Content Text has no Abreviation	PASSED	1	Level AAA	Vision/Cognitive/Learning	Verified
Sc (3.1.5) Reading Level	225	Webpage Content has Accessible Reading Level	PASSED	29	Level AAA	Cognitive/Learning Problem	Verified
Sc (3.1.6) Pronunciation	3	Mispronunced Words have Found in the Title	FAILED	4	Level AAA	Vision/Cognitive/Learning	Required Additional Checking
Sc (3.1.6) Pronounciation	222	Mispronounced Words have Found in the Content	FAILED	114	Level AAA	Vision/Cognitive/Learning	Required Additional Checking
Sc (3.2.2) On Input	0	Webpage has no Checkbox	NOT DETECTED	0	Level A	Vision/Cognitive Problem	Verified
Sc (3.2.2) On Input	0	Webpage has no ComboBox	NOT DETECTED	0	Level A	Vision/Cognitive Problem	Verified

FIGURE 4.16: The view of the WCAEE tool for the tested LIVE IT webpage (Textual elements)



Web Content Accessibility Evaluate Environment (WCAEE)

Non-Semantics (WCAG) | Semantics (WCAG) | User Requirements | Summary | General Statistics | Kye Statistics | Overall Statistics

Additional Requirements or User Requirements

ADDITIONAL CRITERIA	TOTAL COUNT	FEEDBACK	RESULT	COUNT	IMPAIRMENTS TYPE	IMPROVEMENT DIRECTION
(Req1.) Minimize the Website Loading Time	1	Webpage Loading Time is not Accessible	FAILED	>0.3	Cognitive Problem	Required Additional Checking
(Req2.) Content Paragraph should not be Lengthy	225	Webpage Length of Paragraph is Accessible	PASSED	<1500	Cognitive Problem	Verified
(Req3.) Webpage should not have Many Links	56	Large Number of Links in the Website is not Acceptable	FAILED	>50	Vision/Cognitive Problem	Required Additional Checking
(Req4.) Webpage should have English Version	1	Webpage Languge is English	PASSED	1	Vision/Cognitive/Learning	Verified
(Req5.) Webpage should not be Very Lengthy	62.0537109375	Webpage Length is not Accessible	FAILED	>14KB	Cognitive/Motion/Learning	Required Additional Checking
(Req6.) Webpage Server should not be Down	1	Webpage Server is Active	PASSED	1	Cognitive Problem	Verified
(Req7.) Webpage should not Demand User Information	2	Webpage does not Require User Information to Access	PASSED	2	Cognitive Problem	Verified
(Req8.) Webpage should not Contain any CAPTCHA	0	Webpage has no CAPTCHA	PASSED	0	Cognitive Problem	Verified
(Req9.) Webpage should have Multiple Languages	0	Webpage has no Multiple Languages Version	FAILED	<=1	Cognitive/Learning Problem	Required Additional Checking
(Req10.) Webpage should not have Many Images	0	Number of Images in the Webpage is Acceptable	PASSED	<=10	Vision/Cognitive Problem	Verified
(Req11.) Webpage should have Font Changing Option	---	Webpage Font Changing Option was not Tested	NOT TESTED	---	Vision/Cognitive Problem	Required Expert Testing
(Req12.) Webpage should have Color Changing Option	---	Webpage Color Changing Option was not Tested	NOT TESTED	---	Vision/Cognitive Problem	Required Expert Testing
(Req13.) Webpage should have Proper Font-Family	22	Webpage Paragraph has no Defined Proper Font-Family	FAILED	22	Vision/Cognitive Problem	Verified
(Req13.) Webpage should have Proper Font-Family	39	Webpage Heading Text has no Defined Proper Font-Family	FAILED	39	Vision/Cognitive Problem	Verified
(Req14.) Webpage should have Proper Fontsize	22	Webpage Paragraph has no Defined Proper Fontsize	FAILED	22	Vision/Cognitive Problem	Verified
(Req14.) Webpage should have Proper Fontsize	39	Webpage Heading Text has no Defined Proper Fontsize	FAILED	39	Vision/Cognitive Problem	Verified
(Req15.) Text should have Proper Text Pattern	154	Webpage Text has Proper Text Pattern	PASSED	154	Vision/Cognitive Problem	Verified
(Req16.) Webpage should have Text, Image, Video	47	Webpage have Missing Text/Image/Video	FAILED	0	Vision/Cognitive Problem	Verified
(Req17.) Number of Audio should be Accessible	0	Webpage has no Audio Content	NOT DETECTED	0	Vision/Cognitive Problem	Verified
(Req18.) Number of Video should be Accessible	0	Webpage has no Video Content	NOT DETECTED	0	Vision/Cognitive Problem	Verified

FIGURE 4.17: The view of the WCAEE tool for the tested LIVE IT webpage (additional elements)

Summary Page: The evaluation report is presented in the summary page or report along with the total number of success criteria that were validated during the evaluation process, the number of success criteria that were checked against three different conformance levels (level A, level AA, and level AAA), and the number of success criteria that validated along with their Passed, Failed, Not Detected, and Not Tested ratios for each type of disability (cognitive, vision, hearing, motion, and learning). The summary report includes a total count of words along with the count of duplicate words, uncommon terms, mispronounced words, and counted abbreviations to provide a brief overview of the evaluated webpage. Additionally, the number of evaluated objects is displayed along with the occurrence number of each object. Besides, the total number of evaluated user requirements have presented with their individual Passed, Failed, Not Detected, and Not Tested ratios. For example, a summary report of a sample of the tested webpage (https://liveit-project.eu/) is shown in Figure 4.18.

It indicates that the developed tool assessed the tested webpage based on 51 WCAG 2.2 success criteria, of which 14 were successfully Passed, 13 Failed, 19 were found to be Not Detected, and 6 were Not Tested. Furthermore, in terms of conformance level, out of the 51 success criteria that were assessed, 23 were found under level A, 13 under level AA, and 18 under level AAA. Regarding the different types of disabilities, among the success criteria that were tested, 35 related to people with cognitive disabilities and vision impairments, respectively, 7 to people with hearing impairments, 6 to people with motion disabilities, and 11 to people with learning disabilities. Besides, focusing on the tested objects and counted words, 38 objects or features were tested, and 225 words were found on the evaluated webpage. Additionally, 18 criteria were evaluated from the user perspective along with the web content accessibility guideline where for the tested webpage 7 were marked as Passed, 6 as Failed, 3 as Not Detected, and 2 as Not Tested.

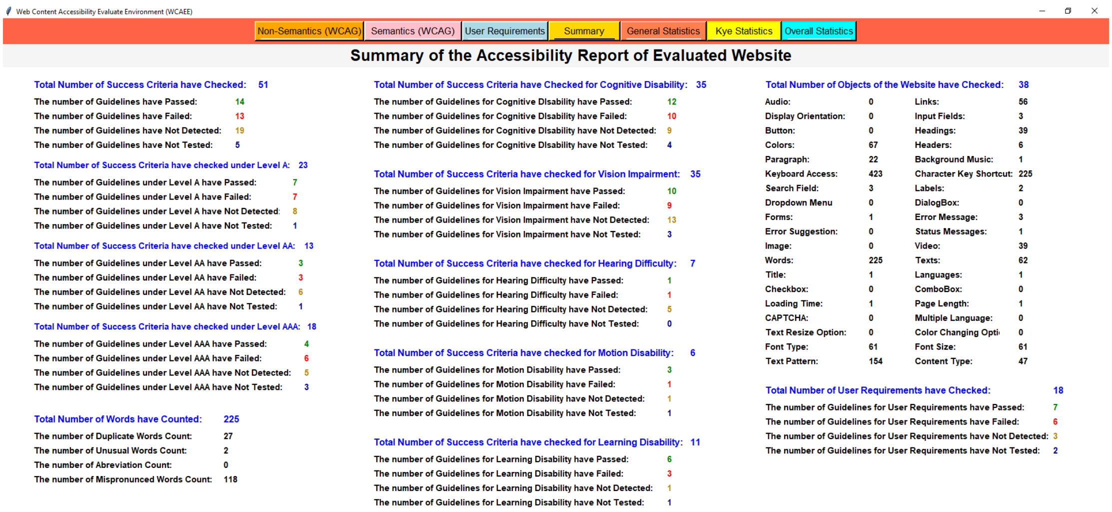


FIGURE 4.18: The view of the WCAEE tool for the tested LIVE IT webpage (evaluation summary)

4.2.1.4 Accessibility report: evaluation report visualization of structural elements

After formulating the evaluation report by three different algorithms and providing a summary report, the evaluation result is organized by considering several statistics and displaying the result through multiple graphical representations. For graphical representation, the results have been classified into three groups and presented through three different pages or window views such as 'General Statistics', 'Key Statistics', and 'Overall Statistics' pages which are represented in Figure 4.19, Figure 4.20, and Figure 4.21 and described in the following:

General Statistic Page: In the general statistics page, the evaluation result is represented in terms of evaluated and not evaluated WCAG, the total number of criteria from both WCAG and user requirements that have been considered in the evaluation process, the number of success criteria and user requirements those have evaluated successfully and require additional checking and expert testing in further. Also, in the general statistics report, the coverage ratio of successfully evaluated success criteria along with other success criteria has been provided that require additional checking and expert testing from WCAG in terms of three conformance levels. Also, the coverage of additional requirements or user requirements is presented on the general statistics page. For example, Figure 4.19 shows the general statistics of the tested webpage formulated by the developed WCAEE tool where all the statistics are represented through several bar charts, line graphs, and pie charts.

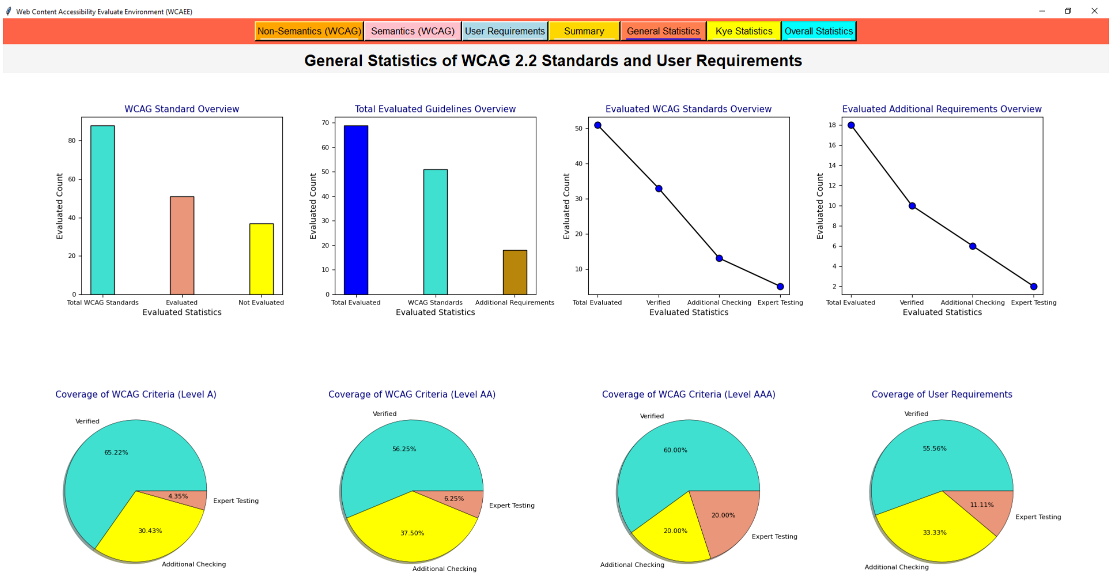


FIGURE 4.19: The view of the WCAEE tool for the tested LIVE IT webpage (general statistics)

In Figure 4.19, the bar chart in the upper left corner represents the number of success criteria that have been evaluated and not evaluated through the tool for the tested webpage in accordance with the total number of success criteria in WCAG. This chart shows that out of the 87 WCAG success criteria, 52 can be evaluated automatically by the developed tool, while the remaining criteria need to be observed manually. The second bar graph from the upper left represents the total number of evaluated criteria including WCAG and user requirements. This chart shows that the tool can evaluate 69 criteria total from both WCAG and additional requirements where 51 from WCAG and 18 from user requirements. Similarly, two-line graphs from the upper right corner represented the evaluated criteria (WCAG and additional requirements) in terms of verified, additional checking, and expert testing. It indicates that for both WCAG and user requirements, the highest number of success criteria was able to be verified indicating that the tool can assess these guidelines automatically. However, a few success criteria require further verification and expert testing in addition to the automatic review. Furthermore, the coverage ratio of evaluated criteria (WCAG and additional requirements) in terms of three evaluation statuses (verified, additional checking, and expert testing) is represented by the first three pie charts from the lower left corner in the lower portion of Figure 4.19. It shows that, for WCAG, the greatest number of success criteria that are automatically assessed under conformance level A is roughly 65.25 percent. Also, a significant portion of the success criteria identified as needing further investigation under conformance level AA is around 37.50 percent. Furthermore, a significant portion of the success criteria that require expert testing under conformance level AAA is roughly 20 percent. On the other hand, a high percentage of the criteria (about 55.56%) for the user requirements could be automatically assessed and effectively validated by the tool. These statistics help users gain a basic understanding of the web page's statistics in relation to the targeted success criteria, which aids in the comprehension of their future modification plan to enhance accessibility scenarios.

Key Statistic Page: On the key statistics page, the evaluation result is represented according to the conformance level from WCAG along with user requirements in terms of assessment terminologies (Pass, Fail, Not Detected and Not Tested), statistics of each evaluated web object or features that helps in understanding which types of objects have been considered during the evaluation process, and ratio of each disability type in terms of success criteria that have Passed, Failed, not tested and not detected. For example, Figure 4.20 shows the key statistics of the tested webpage that have been evaluated through the developed WCAEE tool where all the statistics are represented through several line graphs, bar charts, and pie charts.

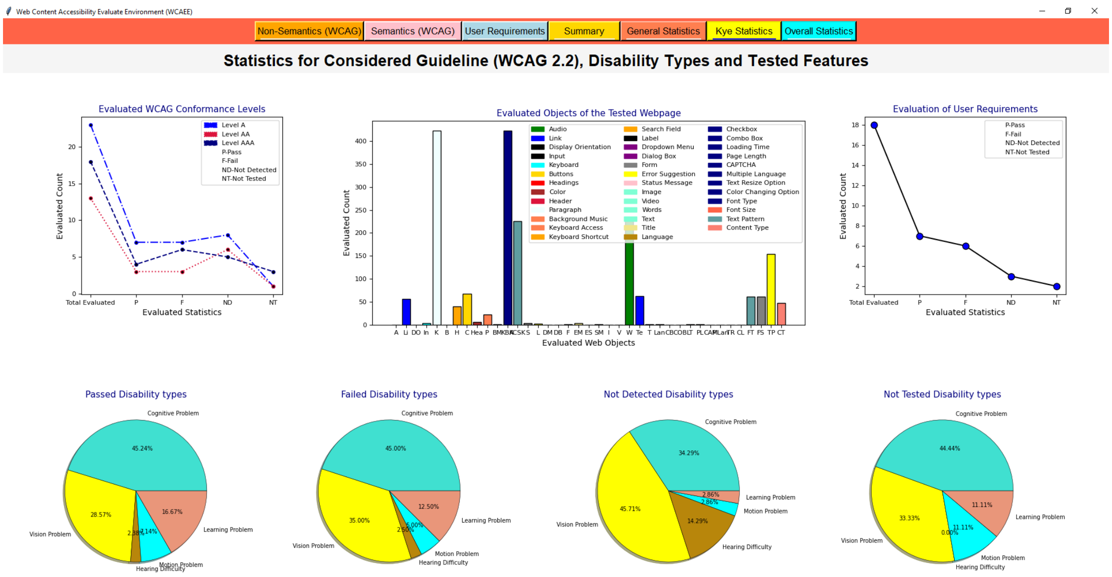


FIGURE 4.20: The view of the WCAEE tool for the tested LIVE IT webpage (key statistics)

In Figure 4.20, the line graph in the upper left corner represents the evaluated success criteria from WCAG for each conformance level (level A indicated through the blue line; level AA indicated through the red line; level AAA indicated through the navy blue line) in terms of four evaluation/assessment terminologies such as Passed (indicated as P), Fail (indicated as F), Not Detected (indicated as ND) and Not Tested (indicated as NT). It depicts that, for the tested webpage, a significant number of success criteria have Passed, Failed, and Not Detected for conformance level A while the maximum number of Not Tested success criteria was found for conformance level AAA. Similarly, the line graph in the upper right corner represents the statistics of user requirements in terms of four assessment terminologies. It indicates that for that particular tested webpage, the majority of the requirements have Passed but a significant number of criteria have also Failed that direct sufficient improvement is required in the future. The bar graph (in the middle) shows the frequency of each evaluated object on the tested webpage which helps the user in understanding which objects or features have been evaluated during the evaluation by the WCAEE tool. In the lower part of Figure 4.20, four pie charts represent the ratio of each type of disability associated with the evaluated guidelines and additional requirements in terms of four assessment terminologies. The first pie chart (from the lower left corner) shows the highest number of criteria Passed associated with cognitive problems whereas the second pie chart shows that the Failure ratio was also greater for cognitive problems. Besides, the third pie chart shows that the majority of the vision problem-related criteria were marked as Not-Detected which means that these criteria were not implemented in the evaluated webpage or the examined webpage did not use them. Finally, the fourth pie chart represents the Not-Tested criteria, which was significant in relation to cognitive issues. These four pie charts with the presented ratio will help user comprehend the accessibility scenario of their tested webpage according to each form of impairment.

Overall Statistic Page: In the overall statistics page as represented in Figure 4.21, some arbitrary information has been summarized that helps to understand some basic information about the tested webpage such as page URLs, page title, total number of checked HTML elements, page size or length, and page loading time. Besides, the major statistics also concluded in the overall statistics page such as evaluated success criteria in terms of each assessment terminologies, overall coverage of each disability type, statistics of several textual types, guidelines overview accordance with conformance level, accessibility score for each disability type, final accessibility status and overall accessibility score. All the statistics are represented through line graphs, pie charts, area charts, bar charts, and progress bars.
The assessment report of the tested webpage and the overall data are depicted in Figure 4.21 in terms of evaluated criteria (guidelines and user criteria). The evaluated criterion overview is presented in the upper left corner by the bar graph and pie chart, while the assessment result of the tested webpage is represented by other graphs. The bar graph shows that among the success criteria evaluated from WCAG, most of the WCAG success criteria that were analyzed came from conformance levels A and AAA. Next, the pie chart shows that the majority of the criteria (WCAG success criteria and user criteria) found in the tested webpage are related to issues with cognitive disabilities and vision impairments which are around 42.06% and 35.71% of the total evaluated criteria, respectively.
The area chart viewed from the lower left corner reveals that the tested webpage has several mispronounced and duplicated words or terms that seriously impair accessibility for those with cognitive disorders. Additionally, the line graph demonstrates that, although the Pass and Fail criteria were found to be roughly equal, a significant number of criteria were identified as Not-Detected in the tested webpage. It indicates that there are a number of missing or erroneous requirements on the tested webpage that should be taken into account in the future to improve accessibility.
After that, eight different progress bar charts showed the accessibility scores according to each type of disability (five types of disabilities), non-semantic (non-textual features), semantic (textual aspects), and additional criteria (user requirements). It shows that the tested webpage has the greatest level of accessibility concern for users with cognitive and vision impairments and the lowest level of concern

for users with learning, hearing, and mobility issues. Additionally, the majority of the semantic test criteria have Passed, indicating that the tested webpage is more accessible for semantic objects than for non-semantic objects. Besides, the accessibility score (77.78%) was prominent for the criteria related to the user requirements.

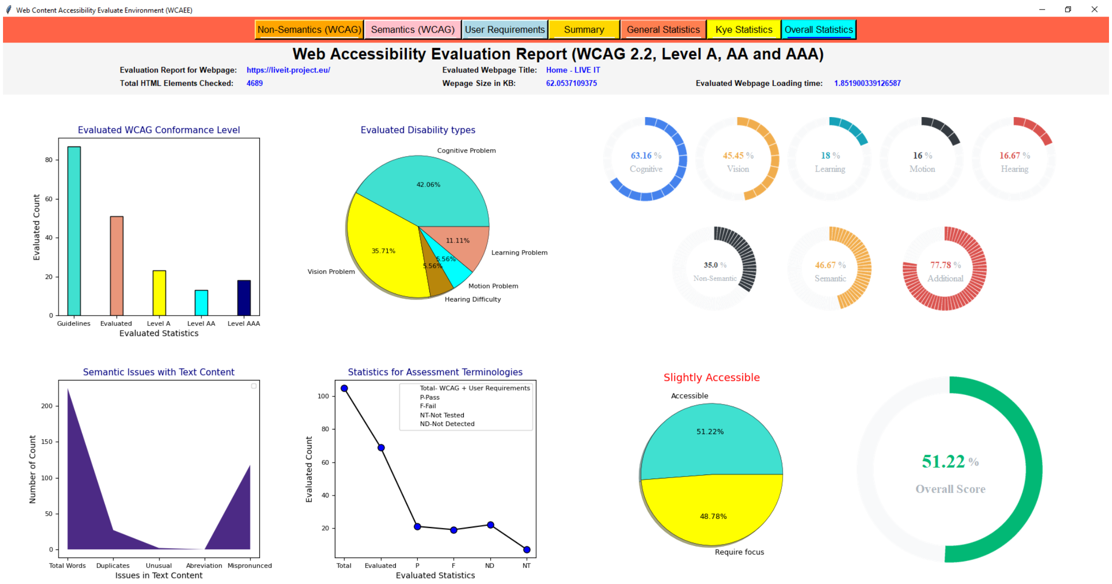


FIGURE 4.21: The view of the WCAEE tool for the tested LIVE IT webpage (overall statistics)

Finally, from the left lower part, the progress bar presents the computed overall accessibility score in percentage and the pie chart represents the accessibility status with the accessibility percentage along with the ratio of further improvement. Besides, to identify the accessibility status, the overall accessibility score has categorized according to four ratios: 'Accessible' if (OverallAccessibilityScore ≥90); 'Comparatively Accessible' if (OverallAccessibilityScore <90 && ≥80); 'Partially Accessible' if (OverallAccessibilityScore <80 && OverallAccessibilityScore ≥60); and 'Slightly Accessible' if (OverallAccessibilityScore <60). Furthermore, the overall accessibility score is computed through Equation 4.1. For the tested webpage, the overall accessibility score was 51.22% which indicates that the tested webpage has significant accessibility issues that require sufficient improvement in the future.

$$OverallAccessibility\_Score = \frac{Total\_passed}{Total\_failed + Total\_not\_detected} * 100 \tag{4.1}$$

4.2.2 Web Content Accessibility Evaluation Environment (WCAEE) Tool Evaluation

Some previous works claimed that accessibility issues of webpages depend on a wide array of web objects [51][107]. By considering those web objects in the accessibility evaluation process, it is possible to improve the accessibility status of the current web and provide a comprehensive accessibility evaluation report to the practitioners. Thus, to evaluate the developed WCAEE tool and to validate the accessibility evaluation report and computed accessibility score using the proposed tool, in this section, a two-phase experimental evaluation has been conducted. At first, a user-centric evaluation was performed considering user participation where users were asked to assess the sample webpages and rate them based on how well they understood the given criteria (details about design and procedure in Section 4.2.2.4). Second, the same sample of webpages evaluated through the proposed tool to evaluate and compute their accessibility score automatically. Furthermore, based on the user-given ratings, a comparative evaluation has been conducted to evaluate the computed score by the WCAEE tool. Moreover, the prime goal of this experimental evaluation is to evaluate the effectiveness of the computed results generated by the proposed tools in terms of the user perspective. This experimental study helps to answer the fifth research question as follows and make a decision about its corresponding hypothesis.

R-Q5: How can the progress made with the developed tool be verified in terms of its effectiveness aspects?

4.2.2.1 Methods of WCAEE tool evaluation

The entire evaluation process is performed in two phases. First, in the user-centric study, at first, the participants were asked to rate a sample of twenty (20) webpages based on their understanding or preference in terms of accessibility perspective. Second, the same sample of webpages was evaluated through the developed WCAEE tool to formulate their accessibility report. Then, user ratings were further compared with the accessibility scores computed by the WCAEE tool for the same sample of 20 webpages.

4.2.2.2 Participants

The user-centric study was conducted by inviting students (bachelor, master, and PhD) from the Department of Electrical Engineering and Information Systems, and the Department of Mathematics, University of Pannonia to test the sample webpages. The experiment was announced and shared in multiple Facebook study groups as well as the university's official Facebook page to invite students with a Google form registration link where students could write their names, preferred times, and suitable dates that they want to participate in the experiment. The experiment was conducted at the university building in the computer programming lab in the desktop version. Fifty (50) students initially verified their registration, however, seven (7) later withdrew from the test. Therefore, the test was performed by including 43 students which took 28 days long. However, among 43 students, 30 were in the 20-23 age range, 11 were in the 24-27 age range, and 2 were in the 28-30 age range. Sixteen (16) of them were foreign students, and the remaining students were Hungarians. Later, three (3) students were removed from the evaluation because they were unable to attend the test due to personal reasons. All the participants reported that they are active on the internet platform, and they use the internet almost every single day for their daily activities. 98% of them mentioned that they use the web for study, 60% for e-commerce portals for online purchasing, and 88% for chatting. Most participants use the internet for more than 30 hours a week, and none report using it for fewer than 8 hours. From their feedback, it is acceptable that all of the participants are familiar with the internet platform for every kind of browsing which increases the sample's generalizability.

Furthermore, for evaluating the same sample of selected webpages (see Section 4.2.2.6) through the proposed/developed WCAEE automated tool, two PhD students from the Department of Computer Science, University of Pannonia were invited to participate in this experimental evaluation. Both selected students are final-year students and have a high level of understanding about software protocols, and Internet platforms and they have experience with other existing automated web content accessibility evaluation tools for their research purposes. As they were already familiar with similar types of evaluation tools, thus it assures their ability to perform the evaluation effectively. However, similar to the user-centric evaluation, all the resources were shared with the two PhD students and invited them to participate in the experiment in the 3D Virtual and Digital Reality Research Laboratory, Department of Electrical Engineering and Information Systems, University of Pannonia. For a declaration, none of the international participants were invited to the experiment due to a limited financial budget.

4.2.2.3 Materials

The most popular top 20 Hungarian university webpages were selected for their accessibility assessment by the user and the developed WCAEE tool. In Table B.6 (Appendix B), the selected webpages are listed with URLs that were used in this study. To include in this study, only English version of the selected webpages was considered as many of the participants were international students (who are staying in Hungary for their studies) and all of the Hungarian students have proficiency in English. Besides, the URLs of the selected page were taken for the homepage only as the proposed WCAEE tool supports single page validation. Thus, the aim is to evaluate and screen the homepage only to balance the result between the user rating and the computed result by the tool.

4.2.2.4 Design and Procedure

For a user-centric study, a comprehensive guideline has been prepared on how to perform the evaluation process, what to evaluate, and how the assessment score should be determined based on what criteria govern the user evaluation process. For the instruction guideline, at first, 15-minutes consultation was performed with the participants before starting the experiment to give them a clear indication of the procedure. To make the experiment effective and less time-consuming, a list with participants' names and the associated webpage link was prepared that they had to evaluate. In that case, before starting the experiment, the file has been shared with the user including the evaluation criteria and scoring instructions prior to initiating the experiment. In this shared file, the forty participants were divided into two groups, with 20 people in each group. To obtain two evaluation scores for a single webpage from distinct participants, each group member was asked to assess two webpages. As a result, 40 participants rated 20 webpages twice, yielding two rating scores. To evaluate, all the users were asked to spend a minimum of fifteen minutes on the webpage, during which they were to examine each object and assess its functionality in terms of i) whether or not each web object is executable; ii) how the webpage is represented in terms of color, theme, structuring style, and content (text, image, and video/audio content); iii) whether or not any prototypes have been found that would indicate if the information is missing; and iv) whether or not the information provided is informative. According to the usefulness/importance/availability of the evaluated content, all the participants were asked to submit a score on a 1–5 point scale, where 1 represents the lowest score and 5 represents the best score or highest relevancy with accessibility.

For the WCAEE tool-centric evaluation, before beginning the experiment, 20-minutes discussion was conducted with two participants to make sure they understood the experimental process. All the participants were offered their user access to the file containing the evaluation guidelines before beginning the experiment. Two PhD student had to perform the experiment for all the selected webpages and report their results. For reporting the results, participants were asked to report based on three criteria i) whether any webpages are unavailable or not detectable through the WCAEE tool, ii) during the experiment, if they found any error in the valuation results and iii) determine the overall accessibility score. The entire experiment took an average of 42 minutes to evaluate and report the results of 20 Hungarian university webpages through the WCAEE tool.

4.2.2.5 User Rating Validation

As it is mentioned in the preceding section, all the invited users were from two groups to assess two distinct webpages, thus each page was scored twice by two different participants. As a result, as indicated in Table 4.2, two ratings were received for a single page from two distinct participants. For user rating validation, the aim is two-fold, first, (1) evaluate the correlation between two scores (Score A and Score B) for a single webpage to determine how consistent and correlated the user-given score is, and second, (2) carry out a reliability test to determine whether two scores are significantly and positively related to one another. The average score of scores A and B were calculated to find the best possible score between the two scores to perform the consistency and reliability test.

However, the Pearson Correlation Coefficient technique was used to assess the consistency and correlation between Score A and Score B given by the users. The results show that there was a significant positive correlation between the user-rated score A (group 1) and score B (group 2), with Pearson Correlation Coefficients of $r(18)=0.845$, and significant at $p<0.001$. Besides, the correlation was also evaluated considering the average score with each sample score. It indicates that a significant and positive relationship has been found between Score A, and the average score, with Pearson Correlation Coefficients of $r(18)=0.968$, and significant at $p<0.001$. Also, a significant and positive relationship has been found between Score B, and the average score, with Pearson Correlation Coefficients of $r(18)=0.845$, significant at $p<0.001$. Also, a reliability test was performed through Spearman's Correlation Coefficients technique. To verify the consistency of the two scores provided by two distinct participants, the correlation between scores A and B was calculated on a single page. With Spearman's Correlation Coefficients of $Rs=0.8902$ and significance at $p=0.001$, the bivariate correlation testing shows that there is a substantial correlation between rating score A and rating score B. Besides, the correlation of the average score with each rating score was evaluated which indicates that a strong and positive correlation was found between Score A and the average score with $Rs=0.9534$ which is significant at $p=0.001$. Also, a strong and positive correlation was found between Score B and the average score with $Rs=0.9677$ which is significant at $p=0.001$. These two statistical analyses depict that the user ratings are normally distributed and significantly correlated with each other which could be included in the final study.

Following a two-step rating correlation analysis of each sample, according to the experts' opinions, a threshold score is stated that acts as an indicator for categorizing the accessibility status of the user-provided rating. The threshold value is applicable for the average score as it denotes the optimal score between two ratings or scores. To determine the threshold, first, the accessibility status was categorized as Completely Accessible, Comparatively Accessible, Partially Accessible, and Slightly Accessible. Then the determined threshold value with a score of ≥4.5 (90%) denotes as Completely accessible, <4.5 (90%) to ≥3.75 (75%) denotes Comparatively accessible, <3.75 (75%) to ≥2.75 (55%) denotes Partially accessible, and <2.75 (<55%) denotes Slightly accessible. However, each page's accessibility status was categorized and assessed based on the specified threshold value, as indicated in Table 4.2. It depicts that according to the user opinion; the majority of the pages are partially accessible, and few are slightly accessible. This indicates that the tested webpages have serious accessibility issues that hinder their complete access.

Table 4.2: The accessibility scores generated by the participants (Score A, Score B, and Average Score) with classified accessibility status for tested webpages

Page ID	Score A (group-1)	Score B (group-2)	Avg. Score	Accessibility status
1	3.45	3.87	3.66	Partially Accessible
2	2.91	3.54	3.225	Partially Accessible
3	3.13	3.15	3.14	Partially Accessible
4	2.15	2.30	2.225	Slightly Accessible
5	3.05	3.15	3.1	Partially Accessible
6	1.2	1.50	1.35	Slightly Accessible
7	2.78	3.0	2.89	Partially Accessible
8	3.65	3.22	3.435	Partially Accessible
9	2.97	2.86	2.915	Partially Accessible
10	3.82	3.92	3.87	Comparatively Accessible
11	2.60	2.64	2.62	Slightly Accessible
12	2.59	2.51	2.55	Slightly Accessible
13	3.01	2096	3.985	Partially Accessible
14	2.39	2.53	2.46	Slightly Accessible
15	1.63	2.55	2.09	Slightly Accessible
16	3.0	2.68	2.84	Partially Accessible
17	1.59	2.05	1.82	Slightly Accessible
18	3.31	3.09	3.2	Partially Accessible
19	3.98	3.22	3.6	Partially Accessible
20	2.55	2.08	2.315	Slightly Accessible

4.2.2.6 WCAEE Tool Rating Validation

To evaluate the developed WCAEE tool rating/score, the selected webpages (listed in Table B.6 (Appendix B) were assessed by the proposed or developed tool and represented their computed accessibility score in Table 4.3. To perform the assessment of webpage through the WCAEE tool, two participants (PhD students) were invited to assess every distinct webpage separately, thus each page was experimented with twice by two different participants. As a result, two ratings were received from the WCAEE tool for a single page from the two-phase experiment. After that, the average of scores A and B were calculated to find the best possible score between the two scores to categorize their accessibility status.

To categorize the evaluation results, similar to Table 4.2, the computed accessibility scores were classified through sated threshold value into four statuses: Completely Accessible, Comparatively Accessible, Partially Accessible, and Slightly Accessible. Completely Accessible is denoted by a score of greater than or equal to 90% (≥90%); Comparatively Accessible is denoted by a score of greater than or equal to 75% (≥75%); Partially Accessible is denoted by a score of greater than or equal 55% (≥55%); and Slightly Accessible is denoted by a score of less than 55%. These statuses are also represented in Table 4.3. From this statistic, Table 4.3 depicts that the majority of the tested webpages are categorized as Partially Accessible, and Slightly Accessible. This indicates that none of the webpages are completely accessible which hinders the consistent accessing opportunities for people with disabilities that violates the accessibility criteria.

Table 4.3: The accessibility scores computed by the WCAEE tool (Score A, Score B, and Average Score) with classified accessibility status for each tested webpage

Page ID	Score A	Score B	Avg. Score	Accessibility status
1	58.97%	58.97%	58.97%	Partially Accessible
2	55.0%	52.5%	53.75%	Slightly Accessible
3	58.97%	58.80%	58.88%	Partially Accessible
4	51.22%	50.09%	50.65%	Slightly Accessible
5	51.22%	52.13%	51.67%	Slightly Accessible
6	40.91%	42.09%	41.5%	Slightly Accessible
7	40.89%	40.88%	40.88%	Slightly Accessible
8	58.97%	58.97%	58.97%	Partially Accessible
9	55.0%	54.05%	54.52%	Slightly Accessible
10	63.16%	63.16%	63.16%	Partially Accessible
11	40.91%	40.91%	40.91%	Slightly Accessible
12	43.18%	43.18%	43.18%	Slightly Accessible
13	67.57%	67.57%	67.57%	Partially Accessible
14	43.0%	44.0%	43.5%	Slightly Accessible
15	36.96%	35.90%	36.43%	Slightly Accessible
16	65.79%	65.79%	65.79%	Partially Accessible
17	46.51%	46.51%	46.51%	Slightly Accessible
18	63.16%	63.16%	63.16%	Partially Accessible
19	44.19%	44.19%	44.19%	Slightly Accessible
20	36.96%	36.93%	36.94%	Slightly Accessible

4.2.2.7 Comparative Evaluation (User rating Vs WCAEE rating)

After evaluating the selected webpages from both the user point of view and the developed WCAEE tool, the aim is to perform a comparative evaluation among both tested results to determine their correlation. According to the computed statistic, the correlation between the average score obtained from the user study and the computed score by the proposed model was evaluated. This comparative evaluation helps to determine whether any similarities are observed between two ratings to determine the alignment of user perception with the delivered results computed by the WCAEE tool. Both the user's rating score and score computed by the WCAEE tool with their accessibility status are presented in Table 4.4.
Table 4.4 shows that most of the tested webpages are categorized as slightly accessible and partially accessible based on both the user-provided rating or score and the score calculated by the suggested tool. Also, for the majority of the tested webpages, the accessibility score generated by the WCAEE tool is significantly related to the participants' or users' ratings in terms of their accessibility status.

Besides, there was no significant correlation seen between the computed score and user rating for a few webpages (ID-2, ID-5, ID-7, ID-9, ID-10, and ID-19), with respect to their classified accessibility status. It can happen as the developed tool evaluates a computer program or written script that acts following advanced techniques. Thus, human perception might be incorrect to observe some critical criteria such as the determination of accessible color, complex words, alternative tags, description of images, etc, those actually defined in the webpage source code and not visible to the user directly. Therefore, for some aspects or criteria, it is difficult to evaluate properly for the end users and often it might not be appropriate like a computer program.

Table 4.4: Comparative results with accessibility status of user-given score and computed score by the WCAEE tool

	User Evaluation		**Evaluation by WCAEE**	
	User Study (between 0-5)		**Computed by WCAEE (%)**	
Page ID	**Avg. Score**	**Accessibility Status**	**Avg. Score**	**Accessibility status**
1	3.66	Partially Accessible	58.97%	Partially Accessible
2	3.225	**Partially Accessible**	53.75%	**Slightly Accessible**
3	3.14	Partially Accessible	58.88%	Partially Accessible
4	2.225	Slightly Accessible	50.65%	Slightly Accessible
5	3.1	**Partially Accessible**	51.67%	**Slightly Accessible**
6	1.35	Slightly Accessible	41.5%	Slightly Accessible
7	2.89	**Partially Accessible**	40.88%	**Slightly Accessible**
8	3.435	Partially Accessible	58.97%	Partially Accessible
9	2.915	Partially Accessible	54.52%	Slightly Accessible
10	3.87	**Comparatively Accessible**	63.16%	**Partially Accessible**
11	2.62	Slightly Accessible	40.91%	Slightly Accessible
12	2.55	Slightly Accessible	43.18%	Slightly Accessible
13	2.985	Partially Accessible	67.57%	Partially Accessible
14	2.46	Slightly Accessible	43.5%	Slightly Accessible
15	2.09	Slightly Accessible	36.43%	Slightly Accessible
16	2.84	Partially Accessible	65.79%	Partially Accessible
17	1.82	Slightly Accessible	46.51%	Slightly Accessible
18	3.2	Partially Accessible	63.16%	Partially Accessible
19	3.6	**Partially Accessible**	44.19%	**Slightly Accessible**
20	2.315	Slightly Accessible	36.94%	Slightly Accessible

However, according to the presented statistics in Table 4.4, it can be concluded that the WCAEE tool could predict the accessibility score that could align with the participants' perception and have the potential to predict the accessibility of a specific tested webpage because the majority of the webpage's accessibility status was similar according to the user-given score and the optimally computed score by the tool.

4.2.3 Web Content Accessibility Evaluation Environment (WCAEE) Tool Validation

To validate the effectiveness of the developed WCAEE tool, a comparative evaluation has been performed considering several similar existing open-source automated tools considering different functional properties. The aim of this validation is to determine the significant improvements/functionalities that have been addressed in the developed tool that distinguish the developed tool from other existing automated tools. Also, this functional property-centric evaluation helps validate the effectiveness of the proposed and developed WCAEE tool in web content accessibility evaluation.

To perform the comparative evaluation, the developed tool is compared with ten (10) existing open-source automated web accessibility testing tools (**Accessibility Checker, AccessMonitor, aCe, AChecker, Bulk Accessibility Checker, MAUVE, Rocket Validator, TAW, WAVE, Nibbler**). These selected tools are practically developed tools that are implemented as real-world applications for the accessibility evaluation of webpages. The comparative evaluation of the selected tools was performed through four criteria. The selected criteria are the following:

Criteria-1: The first criterion is related to what standards or guidelines they have followed to execute the test, and how many success criteria, conformance level, and checkpoints can be supported by the tool. This criterion is critical because the evaluation result might be different based on the considered success criteria, conformance level, and checkpoints. The more success criteria, conformance level, and checkpoints a tool covers, the more complete and precise results are expected to return. Besides, the WCAG 2.2 guidelines have 87 success criteria with three levels of conformance type. Some of the success criteria can be evaluated automatically and some are not possible to implement in an automatic manner. Thus, the tool should clearly indicate the evaluated and not evaluated guidelines in the evaluation report. Besides, the incorporation of user and expert suggestions as an additional evaluation criterion is also a crucial factor in improving the effectiveness of the tool.

Criteria-2: The second criterion is related to whether the tool is able to provide information about the evaluated assessment features with their respective detected issues. With proper accessibility issues indication and complete direction about how to resolve those issues can significantly improve the effectiveness of the selected tool.

Criteria-3: The third criterion is related to how accessibility issues are categorized during evaluation. Improper and ambiguous investigation terminologies might reduce the effectiveness of the evaluation tool. Thus, the more a tool utilizes precise terminologies for the accessibility issues classification, the more the results will be understandable and acceptable to its end users.

Criteria-4: The fourth and final criterion is related to how the evaluation results are provided by the tool to its end users. For example, without taking into account the accessibility percentage of each type of disability, the overall accessibility score does not give a comprehensive picture of the accessibility condition of the specific webpages. A computed accessibility score with insights into accessibility in terms of each disability type can improve its acceptability to the end user. Thus, the tool should be capable of indicating accessibility of every type of disability with overall accessibility measures.

According to the selected four criteria, the comparative analysis with key attributes of the selected tools is summarized in Table 4.5. It depicts that concerning how the evaluation results are reported or represented to its users, the most common structure is to represent the result with its investigated success criteria, conformance level, and assessed features. The representation of the results following these criteria can improve its effectiveness. However, only one tool shows the result with success criteria,

four tools consider conformance levels to represent the results, and two tools consider the assessed features. Other tools don't consider these aspects and summarize the result in a tabular format. Besides, WCAG does not include every aspect that might be difficult for people with disabilities. Thus, other aspects that are not considered in WCAG need to be incorporated into the evaluation process along with WCAG. For example, webpage loading time, webpage appropriate length, ratio of arbitrary information such as links, images, forms, etc. Surprisingly none of the selected tools consider such aspects in their evaluation process that might not be effective for a large number of users with disabilities.
Regarding the supported accessibility guideline, five selected tools provide the considered guideline information that the tool is able to validate. Four of them validate against WCAG 2.1 and along with WCAG 2.1, one tool (AChecker) extended their validation ability against BITV 1.0; Section 508; and Stanca Act. However, the other five tools do not explicitly indicate the guidelines that they have implemented during webpage scanning. From the user's perspective, the possibility of knowing exactly which guideline is implemented would increase the effectiveness of the evaluated results. All the considered tools do not explicitly provide any information about which checkpoints are not able to be implemented automatically and require additional checking such as manual checking. Thus, the users can't understand which accessibility criteria are not covered by the tool, and which features have to be manually inspected to ensure complete accessibility of the evaluated webpages. Regarding the information on how to resolve the detected issues, most tools provide only the listed issues without explicitly mentioning the resolving procedure. Few tools provide W3C documentation links that are not useful for the end users or those who have no technical background. In this way, it reduces its effectiveness and reliability.

Table 4.5: Comparative assessment results of the WAEE tool with existing automated tools according to their functionalities/properties

Tools	**Criteria-1**					**Criteria-2**		**Criteria-3**		**Criteria-4**
	F-1	**F-2**	**F-3**	**F-4**	**F-5**	**F-6**	**F-7**	**F-8**	**F-9**	**F-10**
Accessibility Checker	Yes	No	No	No	No	No	No	Yes	No	No
AccessMonitor	Yes	No	Yes	No	No	No	No	Yes	No	No
aCe	No	No	No	Yes	No	No	No	No	No	No
AChecker	Yes	Yes	Yes	No	No	No	Yes	No	No	Yes
Bulk Accessibility Checker	No	No	No	No	No	No	Yes	No	No	Yes
MAUVE	Yes	No	Yes	No	No	No	No	Yes	No	Yes
Rocket Validator	No	No	No	No	No	No	No	No	No	Yes
TAW	Yes	Yes	Yes	No	No	No	No	No	No	Yes
WAVE	No	No	No	No	No	No	No	No	No	Yes
Nibbler	No	No	No	Yes	No	No	Yes	Yes	No	No
WCAEE	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

[*Considered/ Supported guidelines= F-1; *Success criteria= F-2; *Conformance level= F-3; *Assessed features= F-4; *Incorporation of User and Expert suggestions= F-5; *Information of not implemented guidelines= F-6; *Improvement Suggestions= F-7; *Overall accuracy= F-8; *Percentage regarding disability type= F-9; *Assessment terminologies= F-10]

In terms of metrics that are used for summarising the identified issues and computed results, some tools provide overall accessibility percentage or ratio of accessibility which helps end users to understand how accessible the evaluated web page/site is. However, none of the selected tools represents the accessibility score or percentage in terms of disability type. In such a way, it is not clear enough about how much accessible the evaluated webpage is for a particular disability type. The last considered feature is the assessment terminologies (used to categorize the result), six tools categorize the accessibility issues in terms of several terminologies. As every tool has different terminologies to categorize the results, a proper explanation should be indicated regarding their exact meaning to clarify their intention to the end users, though none of the tools provide a complete explanation of this matter. Besides, the four tools didn't explicitly define their assessment terminologies. Moreover, the comparison results show that none of the chosen tools encountered all the selected criteria when assessing their functional ability. In contrast, the developed tool satisfies all the criteria that have been selected for the comparative evaluation that validates its potential.

4.3 Conclusion

In this Chapter, first, an accessibility evaluation framework is proposed considering several aspects from the coding to the visualization part which lead web practitioners to understand the accessibility evaluation process and how it could facilitate to improvement of the accessibility evaluation result. The proposed framework has the potential to overcome the current accessibility evaluation tools' limitations. Additionally, none of the studies found in the literature addressed such aspects for web accessibility evaluation which completely represent the novelty of the proposed framework. Following the proposed framework, second, the developed end-to-end work is presented to provide a tool that can evaluate webpage objects and determine their level of accessibility. The developed tool is able to evaluate accessibility issues and generate accessibility scores for webpages based on common aspects of their HTML Document Object Model (DOM) considering structural and visual elements. Therefore, users can have an initial perception of the accessibility of the tested webpage, and web designers and developers can use this tool to take a complete direction to design and develop an accessible webpage maintaining usability and accessibility criteria. It is believed that this work is an important contribution to the area of web accessibility domain because by using this tool, it is possible to represent the evaluation results in an implicit way focusing on every group of users including special needs.

Chapter 5

Conclusion

In this thesis, several new methods are presented for the accessibility evaluation process of web content. In general, the accessibility evaluation process for web content is categorized into hybrid and automated evaluation processes. In the existing literature, numerous studies have been conducted, focusing on this area to facilitate the web content accessibility evaluation process. However, existing solutions have several ambiguities that reduce their effectiveness and acceptance in the community. Therefore, there is an increasing demand for new solutions to overcome these limitations. With this focus in mind, in this thesis, several studies are evaluated from the existing literature to determine the issues that are observed frequently to answer the first research question. The evaluation result provides a wide array of challenges and issues that hinder the effectiveness of existing web content accessibility evaluation approaches or solutions. Therefore, the first hypothesis (H-1) that corresponds to the first research question is accepted.

According to the findings, to improve the effectiveness of the hybrid testing process, three new methods are presented, such as i) Integrated approach, ii) Variable magnitude approach, and iii) ML-based approach. In the integrated approach, several existing automated evaluation tools have been integrated with user and expert testing to facilitate the evaluation process. In the variable magnitude approach, two separate algorithms are proposed to compute accessibility scores based on the four components or terminologies (pass, fail, not tested, and not decided) obtained from two automated accessibility testing tools, Mauve++ and TAW. To compute the accessibility score through the proposed algorithms, the severity factor has been considered which helps to scale down the weight of the components based on their importance. To generate the overall score, two algorithms scores were integrated. In the ML-based approach, two machine learning algorithms are implemented that were experimented on a custom dataset to improve the performance of the evaluation process. To validate the effectiveness of the three proposed methods, experimental work has been performed considering different webpages from different domains. The experimental results depicts that the proposed approaches are effective in improving the accessibility evaluation results of webpage content. Therefore, the second hypothesis (H-2) that corresponds to the second research question is accepted.

Besides, for automated evaluation, an inclusive framework has been presented that considers several aspects such as appropriate guideline selection, guideline modeling, user and expert assessment criteria as additional criteria, and several auxiliary methods with advanced techniques that didn't considered in existing literature. The proposed framework is validated with existing models according to their functional properties, which shows that the proposed framework has the potential to improve the effectiveness of the automated web content accessibility evaluation process. Therefore, the third hypothesis (H-3) that corresponds to the third research question is accepted.

Furthermore, following the proposed framework, an automated tool has been developed considering algorithmic evaluation and advanced engineering techniques that can be useful for web content accessibility evaluation. A user-centric evaluation and experimental evaluation have been performed to validate the effectiveness of the developed tool. At the same time, the developed tool is validated by comparing it with existing models and open-source tools considering their functional properties. The comparative and experimental results shows that developed tool is prominent to facilitate the performance of the evaluation and improve the effectiveness of the validated results. Therefore, the fourth hypothesis (H-4) corresponds to the fourth research question, is accepted.

5.1 New Scientific Results

In this thesis, different approaches have been studied to mitigate the existing issues related to the evaluation of web accessibility. In the following part of this thesis, I summarize the new scientific results considering three different thesis groups.

5.1.1 Thesis group-I: Web Content Accessibility Testing process considering advanced engineering aspects

This thesis group has been described in Chapter 2. In this thesis group, I investigated several proposed solutions for accessibility evaluation to identify frequently arising issues that limit the effectiveness of their development. According to the observation of my conducted research, it can be stated that:

Thesis-1: There are a few reference architectures for referring to accessible web design, development, and evaluation processes. Most frameworks do not consider aspects related to cost and feasibility, such as human input cost and the quality of inputs in terms of level of expertise, consistency, cost, and generality of results across different disabilities. Also, most frameworks focus on accessibility improvement for people with color vision deficiency and an accessibility testing and refinement process for the early phase of design and development. Besides, frameworks that consider only non-expert users raise inappropriate feedback regarding the wide spectrum of disabilities (all of the aspects presented in this thesis have been proven in Chapter 2, Section 2.4).

Sub-thesis-1: It would be beneficial to develop other reference architectures focusing on other contributing areas to solving the three problems: (a) Framework for the developer to identify and implement accessibility features to address accessibility issues, (b) Easy methods to understand and ensure accessibility requirements concerning every type of disability during the development phase, and (c) Updated automatic accessibility testing protocols incorporating the latest WCAG standards rules.

Sub-thesis-2: To overcome these problems, it can be noted that developing new methods and tools focusing on an accessibility evaluation strategy considering several aspects might help to improve the effectiveness of the further proposed solutions, such as: (a) Guideline selection, (b) User and expert suggestion consideration, (c) Guideline visualization, (d) Listing several webpage features that require special focus during tool development, and (e) Acceptable accessibility issue identification and visualization process.

The corresponding publications of this thesis group (I) are **A1, A9**.

5.1.2 Thesis group-II: Proposing a Hybrid Web Content Accessibility Testing process using integrated techniques, variable magnitude approach, and ML techniques

This thesis group has been described in Chapter 3. In this thesis group, I proposed three new hybrid web content accessibility testing approaches that have the potential to improve the effectiveness of the web accessibility evaluation process, which might be helpful for web practitioners in improving the accessibility of their webpages in the future. Based on the result of this thesis group, I formulate three theses as follows:

In the integrated approach,

Thesis-2: I improved the accessibility assessment process of web content by proposing an integrated approach that incorporates several automated evaluation tools and a user and expert assessment based on questionnaires. The proposed integrated approach is validated through experimental results, considering official websites related to COVID-19. The experimental result showed that integrating multiple automated testing tools and allowing user and expert assessment is crucial to improve the effectiveness of the evaluation result (this thesis has been proven in Chapter 3, Section 3.1).

In the variable magnitude approach,

Thesis-3: I improved the effectiveness of the hybrid web content accessibility evaluation result by proposing a variable magnitude approach, considering the output of multiple automated evaluation tools to measure the accessibility scores of webpages. The proposed variable magnitude approach considers the evaluation results of two automated tools as input variables, calculates the accessibility score by altering the input variables' weight according to their importance, and integrates the results of multiple variables to calculate the final score. The proposed variable-magnitude approach is validated by incorporating expert evaluation and experimental results, considering the webpages of hospitals and clinics. Expert evaluation and experimental results show that the proposed variable magnitude approach is effective in properly identifying the accessibility status of the tested webpages (this thesis has been proven in Chapter 3, Section 3.2).

In the machine learning-based approach,

Thesis-4: I emphasized the improvement of the hybrid web content accessibility evaluation result by proposing a Machine Learning (ML) approach considering ten additional evaluation criteria beyond the latest version of WCAG 2.2. The proposed ML approach was validated by training a model using a custom dataset that has been prepared according to the selected additional ten aspects obtained from user testing and incorporating two ML techniques (Random Forest (RF) and Decision Tree (DT)) to improve the effectiveness of the evaluated results. The proposed ML approach is experimented with considering the university webpage. The experimental results show that the proposed ML approach is significant in the web testing domain and able to represent the actual accessibility insights of the tested webpages in terms of the additional evaluation criteria (this thesis has been proven in Chapter 3, Section 3.3).

The corresponding publications of this thesis group (II) are **A2, A3, A4, A5, A6, A7**.

5.1.3 Thesis group-III: Developing an Automated Web Content Accessibility Testing Tool using knowledge simplification and advanced engineering techniques

This thesis group has been described in Chapter 4. In this thesis, the work has been presented with a two-fold objective. At first, I proposed an automated web content accessibility evaluation framework to facilitate the accessibility evaluation process of webpages. To facilitate the evaluation process, I incorporated a guideline modeling approach that helps to simplify the natural text guideline into a logical format. Furthermore, according to the proposed framework, I developed an automated accessibility testing tool that considers a wide array of aspects in the evaluation process to improve the effectiveness of the evaluated results. Based on the outcome of this thesis group, I have formulated two theses as follows:

Thesis-5: I proposed an automated accessibility evaluation framework addressing several accessibility aspects such as (i) Simplifying the updated web content accessibility guidelines, (ii) Incorporating all success criteria in the evaluation process, (iii) Incorporating user requirements/opinions with expert suggestions, (iv) Incorporating separate complexity analysis algorithms for textual feature, and non-textual feature, (v) Categorizing the evaluated guidelines in terms of user evaluation and expert evaluation, and (vi) Displaying the evaluation result with the overall accessibility score to improve the evaluation results by mitigating the limitations of existing solutions. The proposed framework is validated by comparing it with existing automated solutions, considering their functional properties (this thesis has been proven in Chapter 4, Section 4.1).

Thesis-6: I proposed a straightforward yet precise model that assesses webpage accessibility by taking into account common features of the structural and visual elements of webpages that are part of the

HTML Document Object Model (DOM) structure. To develop the proposed model as a real-life accessibility testing tool, three distinct algorithms have been implemented to analyze web features/objects, considering both semantic and non-semantic aspects. The developed tool, namely Web Content Accessibility Evaluation Environment (WCAEE) is compared to other tools (that already exist) considering several functional characteristics or properties and experimental work with a user study. This two-phase evaluation result shows that the developed tool has several advanced properties and the potential to predict the accessibility issues of the tested webpage (this thesis has been proven in Chapter 4, Section 4.2).

The corresponding publications of this thesis group (III) are **A8, A9, A10, A11, A12, A13, A14, A15**.

5.2 Discussion

Several recent studies, regarding web accessibility have found that the proportion of inaccessible webpages on internet is growing rapidly, which negatively impacts people with disabilities in their access to digital resources. In recent years, this issue has drawn the attention of researchers' in finding ways to improve this problem so that persons with impairments can gain better access options.

Among several possible solutions, the most effective technique is to demonstrate the limitations of a developed webpage by reviewing its features to determine its accessibility status in terms of several factors, like accessibility score, accessibility concerns, etc. Concerning this aspect, several studies performed investigations on different web accessibility evaluation approaches, unfortunately, their reported result emphasized that most of the existing solutions are not effective enough to determine the actual accessibility scenarios of current web platform.

Therefore, in this PhD thesis, the prime objective is to focus on different web accessibility testing solutions in order to improve the effectiveness of accessibility testing result. However, in order to address the current limitations of web accessibility testing process, my findings are aligned with integration of multidimensional evaluation resources, importance of variable significance, ML based optimisation, and incorporation of advanced engineering techniques such as appropriate guideline selection, user and expert suggestion incorporation, guideline modelling, and semantic improvement, to determine the proper accessibility result formulation.

Addressing the significance of the proposed approaches presented in this PhD thesis, my observation is, most of the existing approaches only focused on the automated tools-based evaluation, although, other testing aspects are also significant and can improve the evaluation result. Therefore, in the proposed integrated approach, I implemented both automated tools (Mauve, Nibbler, WAVE, and WEB accessibility) and human observation through questionnaires which categorised the proposed integrated approach uniquely. Additionally, this analysis will help to understand the effectiveness of the automated tools and human validation together. Besides, the proposed variable magnitude approach offers severity-factor selection option that help to choose a constant value for all the variables in terms of their significance. This process allows to determine the dependent and independent variables with their potential which help to identifying the actual accessibility scenario of webpages that is missing in the existing similar approaches. In addition, very few works focused on machine learning methods to evaluate the webpage quality. Most of them depict that the machine learning model is more effective in evaluating webpage quality than other approaches. Besides, none of the work performed the evaluation considering the additional criteria. Therefore, the proposed ML-based approach validates web accessibility using ML techniques focusing on the ten additional aspects related to the webpage feature such as availability, manual text size adjustment option, manual font family adjustment option, manual color adjustment option, user information requirement, CAPTCHA, usefulness of internal/external links, images, inserted video, and audio content which make the proposed approach unique than the existing systems. Furthermore, the implemented automated accessibility testing framework and the developed Web Content Accessibility Evaluation Environment (WCAEE) tool incorporates an updated version of Web Content Accessibility Guidelines (WCAG 2.2), the user and expert suggestions, and guideline simplification concept. It classified all the guidelines in terms of semantic and non-semantic aspects considering natural language processing (NLP) techniques. Also, to determining the proper visualization criteria, it focuses on every disability type. All of these addressed advanced engineering techniques makes the developed automated tool significant and unique compared to other existing similar automated tools.

Although the proposed approaches have the potential to evaluate webpage accessibility, they are not free from limitations. All the proposed approaches perform the evaluation considering single standard guideline (WCAG) that can be extended incorporating other guidelines. All the proposed approaches focus on the latest version of WCAG 2.2, which can be extended to implementing upcoming version of WCAG 3.0. Besides, the proposed models consider fewer user and expert evaluation, which can be extended by considering other criteria by incorporating more user and expert involvement. Although considering expert suggestions is appropriate, it is also crucial to justify expert suggestions by considering scientists' opinions of this field. Also, the limitation associated with limited number of automated tool consideration, single-page validation, a small dataset, can be address in the upcoming research. Besides, we do not consider any dynamic elements and their effect on our evaluation process to compute accessibility scores. Further studies are suggested for a better understanding of the impact of dynamic content on accessibility. Besides, due to the financial budget and limited time, at this stage, we validated the proposed tool considering students' evaluation results, although evaluating incorporating specialists would provide more suitable evaluation results. Further studies might be conducted with other domain webpages. Also, the proposed tool doesn't offer the downloading option of the evaluation report. Future study may extend the tool's functionality by adding the ability to download the evaluation reports.

However, to improve the current scenario of web in term of accessibility, authorities should pay more attention to accessibility issues during the design and development phase. More accessibility evaluation processes should be carried out in the future, as webpages are updated constantly. Therefore, frequent evaluation will help to identify and solve the emerging issues. In addition, further training should be organized for web designers, web developers, and other associated authorities to increase the awareness and importance of accessibility to make internet platforms accessible.

Chapter 6

The author's publications

[**A1**] **Jinat Ara**, Cecilia Sik-Lanyi, and Arpad Kelemen, "Accessibility Engineering in Web Evaluation Process: A Systematic Literature Review", Universal Access in the Information Society, Springer, 2023; pp.1-34. DOI: https://doi.org/10.1007/s10209-023-00967-2 (**Impact Factor: 2.1**)

[**A2**] **Jinat Ara** and Cecilia Sik Lanyi, "Accessibility Evaluation of COVID-19 Information Websites across Europe". IEEE 34th Neumann Colloquium (NC), pp. 126-133. IEEE, (2021)

[**A3**] **Jinat Ara** and Cecilia Sik Lanyi, Investigation of COVID-19 Vaccine Information Websites across Europe and Asia Using Automated Accessibility Protocols. International Journal of Environmental Research and Public Health, MDPI, 2022; 19(5), pp.2867. DOI: https://doi.org/10.3390/ijerph19052867 (**Impact Factor: 3.390**)

[**A4**] **Jinat Ara**, Cecilia Sik-Lanyi, and Arpad Kelemen, "An integrated variable magnitude approach for accessibility evaluation of healthcare institute webpages", Applied Science, MDPI, 2023; 13(2), pp.932. DOI: https://doi.org/10.3390/app13020932 (**Impact Factor:2.5**)

[**A5**] **Jinat Ara** and Cecilia Sik-Lanyi, "Accessibility evaluation of healthcare webpages in Hungary using accessibility barrier computation algorithm", 1st IEEE International Conference on Internet of Digital Reality (IoD), 23-24 June, 2022, Gyor, Hungary. DOI: 10.1109/IoD55468.2022.9986974

[**A6**] **Jinat Ara** and Cecilia Sik-Lanyi, "Webpage accessibility evaluation using Machine Learning Technique", 14th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2023, Budapest, Hungary. DOI: 10.1109/CogInfoCom59411.2023.10397496

[**A7**] **Jinat Ara**, and Cecilia Sik-Lanyi, "Computation of Accessibility Score of Educational Institute Webpages using Machine Learning Approaches", Infocommunications Journal, Joint Special Issue on Cognitive Infocommunications and Cognitive Aspects of Virtual Reality, 2024, pp.49-57. DOI: https://doi.org/10.36244/ICJ.2024.5.6. (**Impact Factor: 1.16**)

[A8] Jinat Ara and Cecilia Sik-Lanyi, "Artificial Intelligent in Web Accessibility: potentials and possible challenges". International Academic Conference on Engineering, Transport, IT and Artificial Intelligence, 5-6 August 2022, Vienna, Austria

[A9] Jinat Ara, Cecilia Sik-Lanyi, Arpad Kelemen, and Tibor Guzsvinecz "An Inclusive Framework for Automated Web Content Accessibility Evaluation", Universal Access in the Information Society, Springer, 2024, pp.1-27. DOI: https://doi.org/10.1007/s10209-024-01164-5 (**Impact Factor: 2.1**)

[A10] Jinat Ara and Cecilia Sik-Lanyi, "Algorithmic Evaluation: Accessibility of Assistive Technology Webpage Content", 1st IEEE International Conference on Internet of Digital Reality (IoD), 23-24 June, 2022, Gyor, Hungary. DOI: 10.1109/IoD55468.2022.9987021

[A11] Jinat Ara and Cecilia Sik-Lanyi, "AccGuideLiner: Towards a Modelling Approach of Web Accessibility Requirements following WCAG 2.2", In 2023 IEEE International Conference on Smart Information Systems and Technologies (SIST), Kazakhstan, 4-6 May 2023, pp.10-15. DOI: 10.1109/SIST58284.2023.10223541

[A12] Jinat Ara and Cecilia Sik-Lanyi, "Towards developing a framework for automated accessibility evaluation of web content from expert perspectives", Infocommunications Journal, (**Impact Factor: 1.16**) (Accepted)

[A13] Jinat Ara and Cecilia Sik-Lanyi, "A Declarative Model for Web Content Accessibility Evaluation Process", In International Conference on Computers Helping People with Special Needs, ICCHP 2024. Lecture Notes in Computer Science, vol-14750, pp.84-92. Springer, 2024. DOI: https://doi.org/10.1007/978-3-031-62846-7_10

[A14] Jinat Ara, and Cecilia Sik-Lanyi, "Automated evaluation of accessibility issues of webpage content: Tool and Evaluation", Scientific Reports, Springer, 2025; 15(1), pp.9516. DOI: https://doi.org/10.1038/s41598-025-92192-5 (**Impact Factor: 3.8**)

[A15] Jinat Ara, and Cecilia Sik-Lanyi, "A Comparative Study: Effectiveness, Reliability, Acceptability, and Fairness of Automatic Web Accessibility Evaluation Tools", Journal of Information Technology Case and Application Research, Taylor and Francis Ltd., (**Impact Factor: 1.30**), (Under review)

Appendix A

Additional Figures

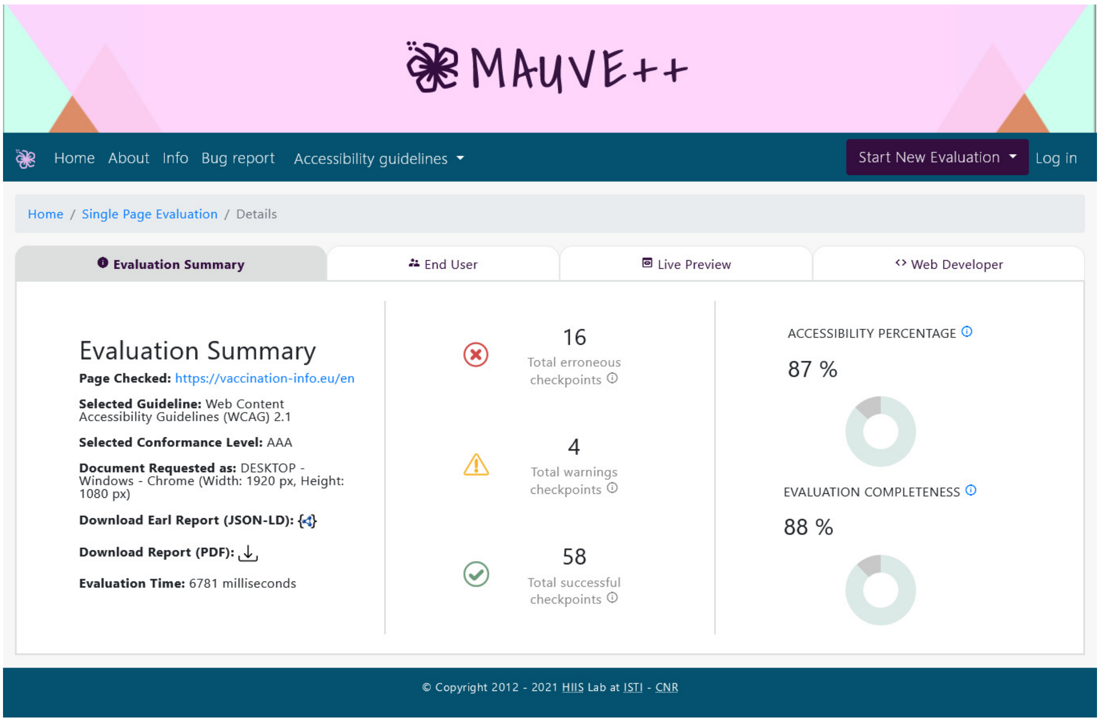


FIGURE A.1: Mauve web accessibility testing tool environment for the tested webpage

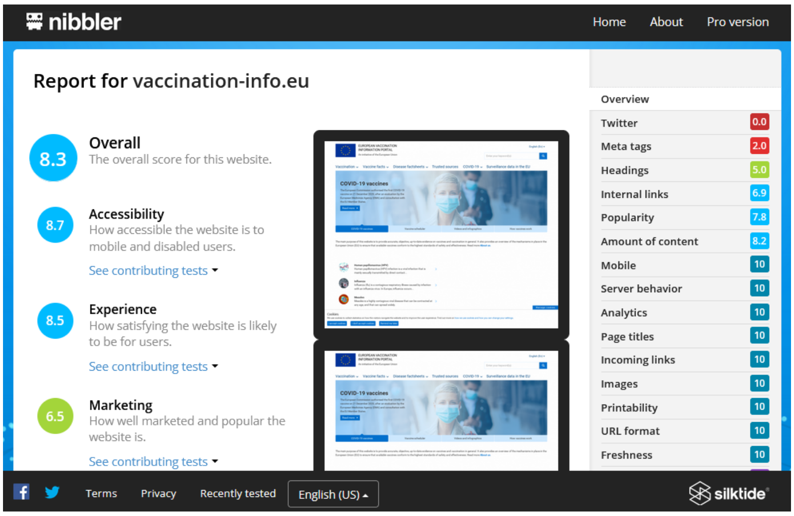


FIGURE A.2: Nibbler web accessibility testing tool environment for the tested webpage

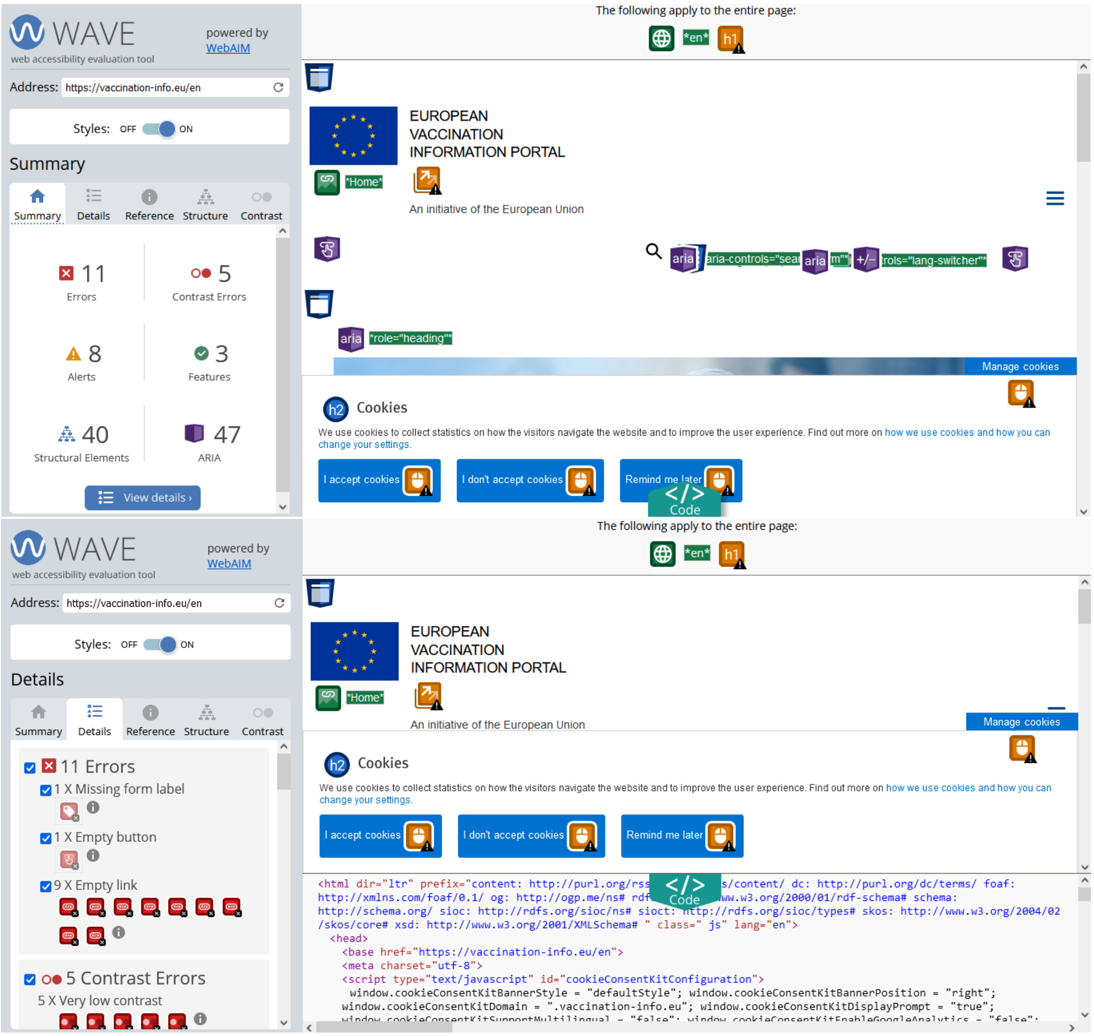


FIGURE A.3: WAVE web accessibility testing tool environment for the tested webpage

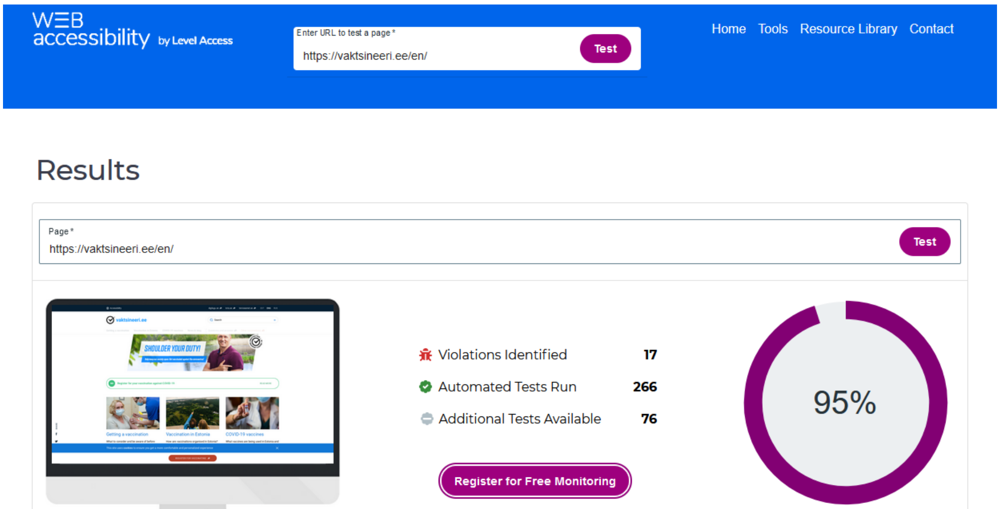


FIGURE A.4: WEB accessibility testing tool environment for the tested webpage

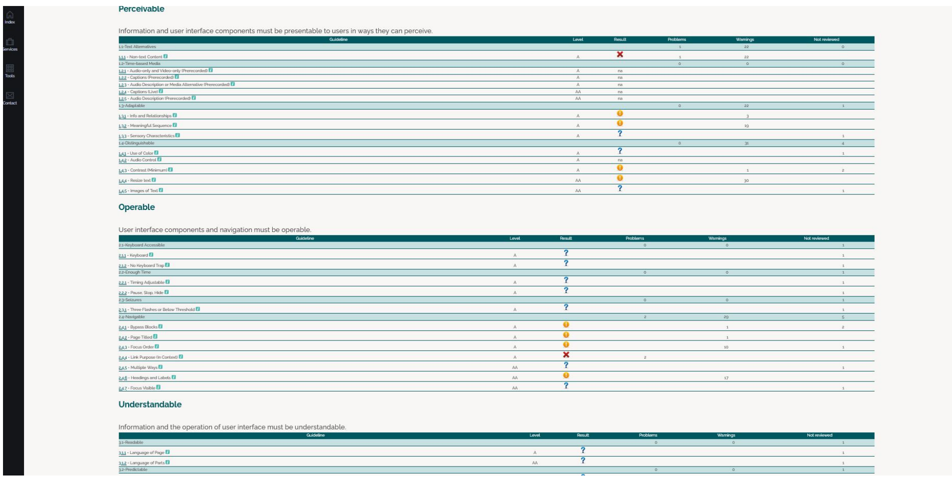


FIGURE A.5: TAW web accessibility testing tool environment for the tested webpage

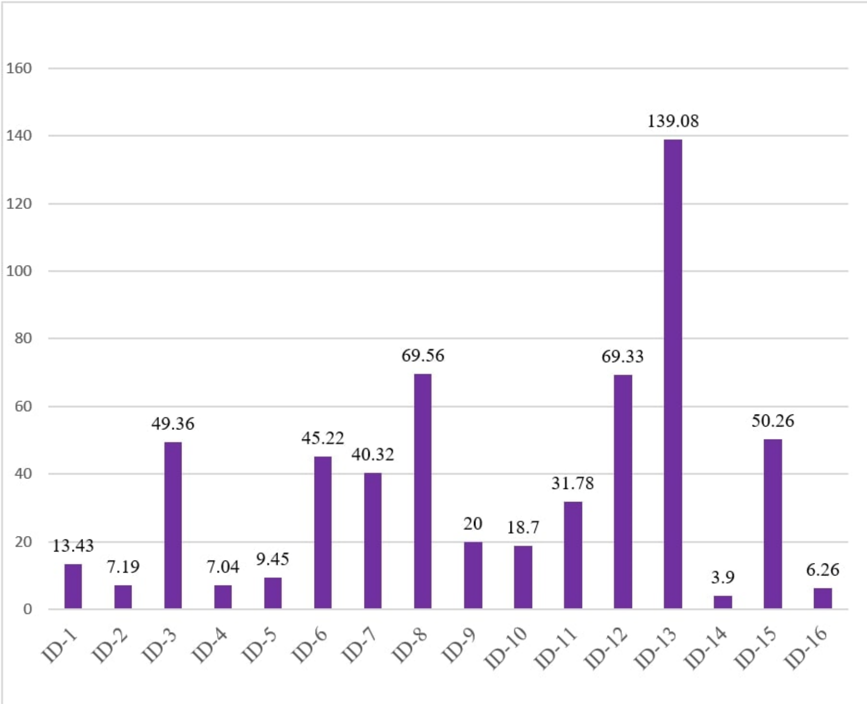


FIGURE A.6: Computed accessibility score by Mauve, (ρ_{Mauve}) for the tested webpage (Bar graph)

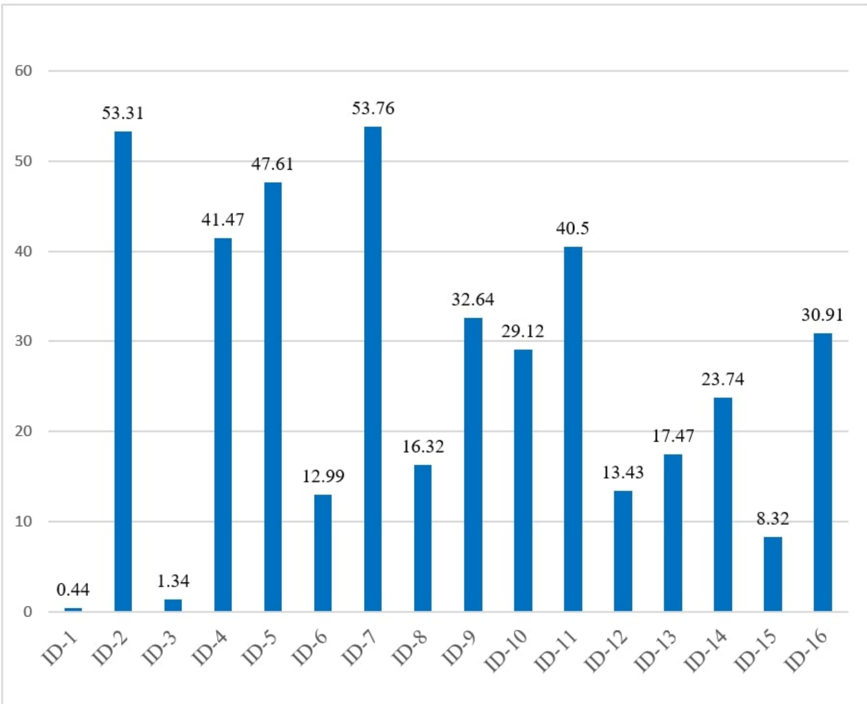


FIGURE A.7: Computed accessibility score by TAW, (ρ_{Taw}) for the tested webpage (Bar graph)

Appendix B

Additional Tables

Table B.1: List of experimented or tested Government webpages

W-ID	Type	Country	Region	Page-URL
WID-1.0	Govt.	Estonia	EU	https://vaktsineeri.ee/en/
WID-2.0	Govt.	France	EU	https://chis.cern/covid-19-vaccination-residents-france
WID-3.0	Govt.	Greece	EU	https://help.unhcr.org/greece/coronavirus/
WID-4.0	Govt.	Ireland	EU	https://www.hse.ie/eng/
WID-4.1	Govt.	Ireland	EU	https://www2.hse.ie/coronavirus/
WID-5.0	Govt.	Latvia	EU	https://covid19.gov.lv/index.php/en/vaccine
WID-6.0	Govt.	Luxembourg	EU	https://covid19.public.lu/en/vaccination.html
WID-7.0	Govt.	Netherland	EU	https://coronadashboard.government.nl/landelijk/vaccinaties
WID-8.0	Govt.	Iceland	EU	https://www.covid.is/statistical-information-on-vaccination
WID-9.0	Govt.	Switzerland	EU	https://www.ch.ch/de/coronavirus/
WID-10.0	Govt.	Croatia	EU	https://gov.hr/en
WID-11.0	Govt.	Germany	EU	https://www.bundesregierung.de/breg-en
WID-12.0	Govt.	Austria	EU	https://www.sozialministerium.at/
WID-13.0	Govt.	India	Asia	https://www.mohfw.gov.in/
WID-14.0	Govt.	Philippines	Asia	https://doh.gov.ph/
WID-15.0	Govt.	Jordan	Asia	https://corona.moh.gov.jo/en
WID-16.0	Govt.	UAE	Asia	https://covid19.ncema.gov.ae/en
WID-17.0	Govt.	Kyrgyzstan	Asia	http://med.kg/en/
WID-18.0	Govt.	Maldives	Asia	https://covid19.health.gov.mv/en/?c=0
WID-19.0	Govt.	Hong Kong	Asia	https://www.covidvaccine.gov.hk/en/
WID-20.0	Govt.	Qatar	Asia	https://www.gco.gov.qa/en/focus/covid-19/

TABLE B.2: List of experimented or tested healthcare institute webpages homepage

W-ID	**Webpage**	**Web page URLs**
ID-1	Országos Onkológiai Intézet	https://onkol.hu/
ID-2	Uzsoki Kórház	https://www.uzsoki.hu/
ID-3	Szegedi Közlekedési Társaság	htts://szkt.hu/szeged-megyei-jogu-varos-ii-korhaza
ID-4	Buda Health Center Budai Egészségközpont	https://bhc.hu/
ID-5	Heim Pál Children's Hospital	http://heimpalkorhaz.hu/
ID-6	Jahn Ferenc Dél-Pesti Kórház	https://www.delpestikorhaz.hu/
ID-7	Klinikai Központ Semmelweis Egyetem	https://semmelweis.hu/klinikaikozpont/
ID-8	Klinikai Központ Debreceni Egyetem	https://klinikaikozpont.unideb.hu/
ID-9	Szegedi Tudományegyetem	https://www.med.u-szeged.hu/
ID-10	Klinikai Központ Pécsi Tudományegyetem	https://kk.pte.hu/klinikak-intezetek
ID-11	Tormay Károly Egészségügyi Központ Tüdőszűrő	https://tormay.hu/
ID-12	B.A.Z Megyei Kórház és Oktató Kórház	https://www.bazmkorhaz.hu/
ID-13	Soproni Gyógyközpont	http://www.sopronkorhaz.hu/
ID-14	Petz Aladár Megyei Oktató Kórház	https://www.petz.gyor.hu/
ID-15	Veszprém Megyei Csolnoky Ferenc Kórház Nonprofit Zrt.	https://csfk.hu/
ID-16	Markhot Ferenc Teaching Hospital and Clinic	https://www.mfkh.hu/

Table B.3: The selected parameters of RF and DT with definition, default values, grid values, and the optimal values for the optimization process

Random Forest Parameters				
	Selected Values			
Parameter	**Definition**	**Default**	**Grid**	**Optimal**
criterion	The quality of a split	Gini	Gini, entropy	Gini
max_depth	Depth of a tree	None	None	None
max_features	Maximum feature of a tree	auto	auto, sqrt, log2	auto
min_samples_leaf	Minimum sample in a leaf node of a tree	1	1, 5, 8, 10, 15, 20	1
min_samples_split	Minimum samples to be split	2	2, 5, 10, 15, 20, 25	2
n_estimators	The number of estimators	100	100, 110, 120, 130, 140, 150	130
random_state	The random state of a node	None	None	None
Decision Tree Parameters				
	Selected Values			
Parameter	**Definition**	**Default**	**Grid**	**Optimal**
criterion	The quality of a split	Gini	Gini, entropy	Gini
max_depth	Depth of a tree	None	None	None
max_features	Maximum feature of a tree	None	None	None
max_leaf_nodes	The maximum leaf node of a tree	None	None	None
min_samples_leaf	Minimum sample in a leaf node of a tree	1	1,2,3,4,8,12, 18,20	2
min_samples_split	Minimum samples to be split	2	2,3,4,5,8,10, 15	2
random_state	The random state of a node	None	None	None

Table B.4: Webpages components validation guidelines/implementation directions used to develop the proposed WCAEE tool

Elements/Objects	**Non-Textual Elements**	**Textual Elements**	**Additional Elements**	**Related guidelines/ Improvement directions**
Audio/Video	✓			(1.2.1) Audio-only and Video-only; (1.2.2) Captions (Prerecorded); (1.2.3) Audio Description or Media Alternative (Prerecorded); [WCAG 2.2]
Links	✓	✓		(1.3.6) Identify Purpose; (1.4.1) Use of Color; (1.4.8) Visual Presentation; (2.4.4) Link Purpose (In Context); (2.4.9) Link Purpose (Link Only); (4.1.2) Name, Role [WCAG 2.2]
Display Orientation	✓			(1.3.4) Orientation [WCAG 2.2]
Input field	✓			(1.3.3) Sensory Characteristics; (1.3.5) Identify Input Purpose (placeholder); (placeholder); (2.1.1) Keyboard; (2.1.4) Character Key Shortcuts; (3.3.2) Labels or Instructions [WCAG 2.2]
Buttons	✓	✓		(1.3.3) Sensory Characteristics; (1.3.5) Identify Input Purpose (placeholder); (1.3.6) Identify Purpose (Buttons); (2.1.1) Keyboard; (3.3.2) Labels or Instructions [WCAG 2.2]
Headings	✓	✓		(1.4.1) Use of Color; (2.4.6) Headings and Labels; (2.4.10) Section Headings [WCAG 2.2]
Header	✓			(1.4.1) Use of Color; (2.4.10) Section Headings [WCAG 2.2]
Paragraph	✓	✓		(1.3.2) Meaningful Sequence; (1.4.1) Use of Color; (2.4.1) Bypass Blocks; (3.1.3) Unusual Words [WCAG 2.2]
Background Music	✓			(1.4.7) Low or No Background Audio [WCAG 2.2]
Keyboard access	✓			(2.1.1) Keyboard; (2.1.3) Keyboard (No Exception) [WCAG 2.2]
Keyboard Character Key Shortcut	✓			(2.1.4) Character Key Shortcuts [WCAG 2.2]
Search Field	✓			(2.4.5) Multiple Ways; (3.2.3) Consistent Navigation; (3.2.4) Consistent Identification; (3.3.2) Labels or Instructions; (4.1.3) Status Messages [WCAG 2.2]
Label	✓	✓		(1.3.3) Sensory Characteristics; (1.3.5) Identify Input Purpose (placeholder); (1.4.1) Use of Color; (2.1.1) Keyboard; (2.4.10) Section Headings; (2.5.3) Label in Name; (3.3.2) Labels or Instructions [WCAG 2.2]
Dropdown Menu	✓			(3.2.1) On Focus; (3.2.2) On Input [WCAG 2.2]
Dialog box	✓			(3.2.1) On Focus; (3.2.2) On Input [WCAG 2.2]

Elements/Objects	**Non-Textual Elements**	**Textual Elements**	**Additional Elements**	**Related guidelines/ Improvement directions**
Form	✓			(3.2.5) Change on Request; (4.1.3) Status Messages [WCAG 2.2]
Status Message	✓			(4.1.3) Status Messages [WCAG 2.2]
Error Message	✓			(3.3.1) Error Identification; (3.3.3) Error Suggestion [WCAG 2.2]
Error Suggestion	✓			(3.3.3) Error Suggestion [WCAG 2.2]
Image		✓		(1.1.1) Non-text Content; (1.4.5) Images of Text [WCAG 2.2]
Pre-recorded/ Live Audio and Video		✓		(1.2.1) Audio-only and Video-only; (1.2.2) Captions (Prerecorded); (1.2.3) Audio Description or Media Alternative (Prerecorded); (1.2.4) Captions (Live) [WCAG 2.2]
Title		✓		(1.3.2) Meaningful Sequence; (1.4.12) Text Spacing; (2.4.1) Bypass Blocks; (2.4.2) Page Titled [WCAG 2.2]
Words		✓		(1.3.2) Meaningful Sequence; (2.4.1) Bypass Blocks; (3.1.3) Unusual Words [WCAG 2.2]
Webpage Text		✓		(1.3.2) Meaningful Sequence; (2.4.1) Bypass Blocks; (3.1.3) Unusual Words; (3.1.4) Abbreviations; (3.1.5) Reading Level; (3.1.6) Pronunciation [WCAG 2.2]
Language		✓		(3.1.1) Language of Page [WCAG 2.2]
Checkbox		✓		(3.2.1) On Focus; (3.2.2) On Input [WCAG 2.2]
Combo boxes		✓		(3.2.1) On Focus; (3.2.2) On Input [WCAG 2.2]
Loading time			✓	Webpage loading time should be $\leq$ 0.3 seconds.
Paragraph length			✓	(1.3.3) Sensory Characteristics; (1.3.4) Orientation [WCAG 2.2]
Hyperlink ratio			✓	The page number of internal and external or hyperlinks must be $\leq$ 50
Default Language			✓	(3.1.1) Language of Page [WCAG 2.2]
Webpage length			✓	Webpage length should be $\leq$ 14 KB.
Server status/ Availability			✓	Webpage must be available or not down.
User information			✓	No 'Username' and 'Password'
CAPTCH			✓	(1.1.1) Non-text Content [WCAG 2.2]
Multiple languages			✓	Webpage should have multiple language versions.
Image ratio			✓	Webpage image number must be $\leq$ 10
Manual font adjustment option			✓	Webpage should have manual text and font size adjustment option.
Manual color adjustment option			✓	Webpage color pair should be accessible except ['red', 'green'] and ['red', 'black'] pair/combination
Text Font family			✓	Accessible font family are 'Tahoma' or 'Calibri' or Roman' 'Helvetica' or 'Arial' or 'Arial' or 'Verdana' or 'Times New Roman'
Text Font size			✓	(1.4.12) Text Spacing [WCAG 2.2]

Elements/Objects	**Non-Textual Elements**	**Textual Elements**	**Additional Elements**	**Related guidelines/ Improvement directions**
Text pattern			✓	No<b > or <strong> or <i> or <em> or <mark> or <sub> or <sup> elements/patterns
Content type			✓	Webpage content type should be a combination of paragraph, image, and video content
Number of audio/ video content			✓	Webpage number of audios should be ≥ 1 to ≤ 2 and videos should be ≥ 1 to ≤ 2

Table B.5: Important libraries that used to develop the proposed WCAEE tool

Python libraries	
✓from tkinter import*	✓from tkinter import messagebox
✓import os	✓import tkinter as tk
✓import tkinter as tkinter	✓from tkinter import ttk
✓import urllib.request	✓from bs4 import BeautifulSoup
✓import validators	✓import re
✓import nltk	✓nltk.download('punkt')
✓nltk.download('stopwords')	✓nltk.download('punkt') nltk.download()
✓from nltk.corpus import words	✓from nltk.corpus import wordnet
✓from nltk.corpus import wordnet as wn	✓from nltk.corpus import stopwords
✓stopwords.words('english')	✓stop_words = set(stopwords.words("english"))
✓from nltk.tokenize import word_tokenize, sent_tokenize	✓import language_tool_python
✓from nltk.tokenize import word_tokenize, sent_tokenize	✓import language_tool_python
✓import itertools	✓from readability import Readability
✓import requests as re	✓import matplotlib.pyplot as plt
✓import matplotlib.patches as mpatches	✓import requests
✓import time	✓from langdetect import detect, detect_langs
✓from langdetect import DetectorFactory	✓import language_tool_python
✓from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg	

Table B.6: Webpages lists that are used in the user study for validating the proposed WCAEE tool

Page ID	**Page Name**	**Page URL**
ID-1	Eötvös Loránd University	https://www.elte.hu/en/
ID-2	University of Szeged	https://u-szeged.hu/english
ID-3	Budapest University of Technology and Economics	https://www.bme.hu/?language=en
ID-4	University of Pécs	https://international.pte.hu/
ID-5	Széchenyi István University	https://admissions.sze.hu/welcome
ID-6	University of Miskolc	https://www.uni-miskolc.hu/en
ID-7	Semmelweis University	https://semmelweis.hu/english/
ID-8	Corvinus University of Budapest	https://www.uni-corvinus.hu/?lang=en
ID-9	University of Pannonia	https://international.uni-pannon.hu/
ID-10	Óbuda University	https://uni-obuda.hu/en/
ID-11	University of Sopron	https://international.uni-sopron.hu/home
ID-12	Budapest Business University	https://uni-bge.hu/en
ID-13	Budapest Metropolitan University	https://www.metubudapest.hu/
ID-14	IBS International Business School Budapest	https://www.ibs-b.hu/en/
ID-15	Pázmány Péter Catholic University	https://ppke.hu/en
ID-16	Hungarian University of Fine Arts	https://mke.hu/english/index.php
ID-17	Moholy-Nagy University of Art and Design	https://mome.hu/en/
ID-18	Eötvös József College	https://ejf.hu/en/welcome
ID-19	University of Kaposvár	https://kc.uni-mate.hu/
ID-20	University of Nyíregyháza	https://english.nye.hu/

Appendix C

Additional Resourcess

The developed Web Content Accessibility Evaluation Environment (WCAEE) tool can be found on GitHub at: https://github.com/Jinat-ara/WEB-Content-Accessibility-Testing-Environment

Bibliography

[1] World Wide Web Consortium (2022a). W3c—web accessibility initiative (wai). *Making the Web Accessible*. URL http://www.w3.org/WAI/.

[2] PDD Dominic, Handaru Jati, P Sellappan, and Goh Kim Nee. A comparison of asian e-government websites quality: using a non-parametric test. *International Journal of Business Information Systems*, 7(2):220–246, 2011.

[3] Alexandro Pinto, Luciana Sepúlveda Köptcke, Renata David, and Hannah Kuper. A national accessibility audit of primary health care facilities in brazil—are people with disabilities being denied their right to health? *International Journal of Environmental Research and Public Health*, 18(6):2953, 2021.

[4] World Wide Web Consortium (2022b). About w3c groups. URL https://www.w3.org/Consortium/activities.

[5] World Wide Web Consortium (2022c). Web content accessibility guidelines (wcag). URL https://www.w3.org/standards/.

[6] World Wide Web Consortium 2018. Web content accessibility guidelines (wcag) 2.1. URL https://www.w3.org/TR/WCAG21/.

[7] Panayiotis Zaphiris and R Darin Ellis. Website usability and content accessibility of the top usa universities. 2001.

[8] Abid Ismail and KS Kuppusamy. Web accessibility investigation and identification of major issues of higher education websites with statistical measures: A case study of college websites. *Journal of King Saud University-Computer and Information Sciences*, 34(3):901–911, 2022.

[9] Maliha Rashida, Kawsarul Islam, ASM Kayes, Mohammad Hammoudeh, Mohammad Shamsul Arefin, and Mohammad Ashfak Habib. Towards developing a framework to analyze the qualities of the university websites. *Computers*, 10(5):57, 2021.

[10] Patricia Acosta-Vargas, Luis Antonio Salvador-Ullauri, and Sergio Luján-Mora. A heuristic method to evaluate web accessibility for users with low vision. *IEEE Access*, 7:125634–125648, 2019.

[11] Liangcheng Li, Can Wang, Shuyi Song, Zhi Yu, Fenqin Zhou, and Jiajun Bu. A task assignment strategy for crowdsourcing-based web accessibility evaluation system. In *Proceedings of the 14th International Web for All Conference*, pages 1–4, 2017.

[12] Disability Inclusion Overview-World Bank group. URL https://www.worldbank.org/en/topic/disability#1.

[13] World Report on Disability. URL https://www.who.int/disabilities/world_report/2011/report.pdf.

[14] Marian Padure and Costin Pribeanu. Comparing six free accessibility evaluation tools. *Informatica Economica*, 24(1):15–25, 2020.

[15] RP Oliveira, Ana Paula Afonso, and Agostinho Sousa Pinto. E-commerce website accessibility evaluation: case study in portugal. *Şubat*, 10:2021, 2020.

[16] V Yukinori Nagano V Ken Suginome, V Koji Yoshimoto, and V Yumi Tsuchiya. Activities for improving web accessibility. *FUJITSU Sci. Tech. J*, 45(2):239–246, 2009.

[17] Ioannis Basdekis, Iosif Klironomos, Ioannis Metaxas, and Constantine Stephanidis. An overview of web accessibility in greece: a comparative study 2004–2008. *Universal Access in the Information Society*, 9:185–190, 2010.

[18] Balázs Csontos and István Heckl. Improving accessibility of cms-based websites using automated methods. *Universal Access in the Information Society*, 21(2):491–505, 2022.

[19] Cecilia Sik-Lanyi and Éva Orbán-Mihálykó. Accessibility testing of european health-related websites. *Arabian Journal for Science and Engineering*, 44(11):9171–9190, 2019. URL https://doi.org/10.1007/s13369-019-04017-z.

[20] M Nagaraju, Priyanka Chawla, and Ajay Rana. A practitioner's approach to assess the wcag 2.0 website accessibility challenges. In *2019 Amity International Conference on Artificial Intelligence (AICAI)*, pages 958–966. IEEE, 2019.

[21] Stephanie Hackett, Bambang Parmanto, and Xiaoming Zeng. A retrospective look at website accessibility over time. *Behaviour & Information Technology*, 24(6):407–417, 2005.

[22] Ravic Ringlaben, Marty Bray, and Abbot Packard. Accessibility of american university special education departments' web sites. *Universal Access in the Information Society*, 13:249–254, 2014.

[23] World Wide Web-Wikipedia. URL https://en.wikipedia.org/wiki/World_Wide_Web.

[24] Karwan Jacksi and Shakir M Abass. Development history of the world wide web. *Int. J. Sci. Technol. Res*, 8(9):75–79, 2019.

[25] Justin Murphy. Artificial intelligence, rationality, and the world wide web. *IEEE Intelligent Systems*, 33(1):98–103, 2018.

[26] Delia Ferri and Silvia Favalli. Web accessibility for people with disabilities in the european union: Paving the road to social inclusion. *Societies*, 8(2):40, 2018.

[27] Fatima Ezzahra Chadli, Driss Gretete, and Aniss Moumen. Digital accessibility: a systematic literature review. In *SHS Web of Conferences*, volume 119, page 06005. EDP Sciences, 2021.

[28] Aleksander Bai, Kristin Fuglerud, Rannveig A Skjerve, and Till Halbach. Categorization and comparison of accessibility testing methods for software development. *Transforming our World Through Design, Diversity and Education*, pages 821–831, 2018.

[29] Michael F Chiang, Roy G Cole, Suhit Gupta, Gail E Kaiser, and Justin B Starren. Computer and world wide web accessibility by visually disabled patients: Problems and solutions. *Survey of ophthalmology*, 50(4):394–405, 2005.

[30] Rachel Caldwell. Web-accessibility, e-learning, and academic libraries. *International Journal of Public Information Systems*, 2(1), 2006.

[31] Web content accessibility guidelines (WCAG) 3.0. URL https://www.w3.org/TR/wcag-3.0/.

[32] World Wide Web Consortium 1999. Web content accessibility guidelines (wcag) 1.0. URL https://www.w3.org/TR/WAI-WEBCONTENT/.

[33] World Wide Web Consortium 2008. Web content accessibility guidelines (wcag) 1.0. URL https://www.w3.org/TR/WCAG20/.

[34] World Wide Web Consortium 2021. Web content accessibility guidelines (wcag) 2.2. URL https://www.w3.org/TR/WCAG22/.

[35] Amiel A Dror, Nicole G Morozov, Eli Layous, Matti Mizrachi, Amani Daoud, Netanel Eisenbach, Doaa Rayan, Edward Kaykov, Hesham Marei, Masad Barhum, et al. United by hope, divided by access: country mapping of covid-19 information accessibility and its consequences on pandemic eradication. *Frontiers in Medicine*, 7:618337, 2021.

[36] Nam Wook Kim, Shakila Cherise Joyner, Amalia Riegelhuth, and Yeeun Kim. Accessible visualization: Design space, opportunities, and challenges. In *Computer graphics forum*, volume 40, pages 173–188. Wiley Online Library, 2021.

[37] Amaia Aizpurua, Simon Harper, and Markel Vigo. Exploring the relationship between web accessibility and user experience. *International Journal of Human-Computer Studies*, 91:13–23, 2016.

[38] Bujar Raufi, Mexhid Ferati, Xhemal Zenuni, Jaumin Ajdari, and Florije Ismaili. Methods and techniques of adaptive web accessibility for the blind and visually impaired. *Procedia-Social and Behavioral Sciences*, 195:1999–2007, 2015.

[39] Kalpana Devi and A Sharma. Framework for evaluation of academic website. *International Journal of Computer Techniques*, 3(2):234–239, 2016.

[40] Gholam Ali Montazer. University website quality improvement using intuitionistic fuzzy preference ranking model. *Quarterly of Iranian Distance Education Journal*, 1(2):9–30, 2018.

[41] Sukhpuneet Kaur, Kulwant Kaur, and Parminder Kaur. An empirical performance evaluation of universities website. *International Journal of Computer Applications*, 146(15):10–16, 2016.

[42] Zehra Yerlikaya and Pınar Onay Durdu. Evaluation of accessibility of university websites: A case from turkey. In *HCI International 2017–Posters' Extended Abstracts: 19th International Conference, HCI International 2017, Vancouver, BC, Canada, July 9–14, 2017, Proceedings, Part II 19*, pages 663–668. Springer, 2017.

[43] Shuyi Song, Jiajun Bu, Andreas Artmeier, Keyue Shi, Ye Wang, Zhi Yu, and Can Wang. Crowdsourcing-based web accessibility evaluation with golden maximum likelihood inference. *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW):1–21, 2018.

[44] Yehya Mohamad, Carlos A Velasco, Nikolaos Kaklanis, Dimitrios Tzovaras, and Fabio Paternò. A holistic decision support environment for web accessibility. In *Computers Helping People with Special Needs: 16th International Conference, ICCHP 2018, Linz, Austria, July 11-13, 2018, Proceedings, Part I 16*, pages 3–7. Springer, 2018.

[45] Liangcheng Li, Jiajun Bu, Can Wang, Zhi Yu, Wei Wang, Yue Wu, Chunbin Gu, and Qin Zhou. Crowdae: A crowdsourcing system with human inspection quality enhancement for web accessibility evaluation. In *Computers Helping People with Special Needs: 16th International Conference, ICCHP 2018, Linz, Austria, July 11-13, 2018, Proceedings, Part I 16*, pages 27–30. Springer, 2018.

[46] Tahani Alahmadi. A multi-method evaluation of university website accessibility: Foregrounding user-centred design, mining source code and using a quantitative metric. In *Proceedings of the 14th International Web for All Conference*, pages 1–2, 2017.

[47] Giovanna Broccia, Marco Manca, Fabio Paternò, and Francesca Pulina. Flexible automatic support for web accessibility validation. *Proceedings of the ACM on Human-Computer Interaction*, 4 (EICS):1–24, 2020.

[48] José Martins, Ramiro Gonçalves, and Frederico Branco. A full scope web accessibility evaluation procedure proposal based on iberian ehealth accessibility compliance. *Computers in Human Behavior*, 73:676–684, 2017.

[49] Shuyi Song, Jiajun Bu, Ye Wang, Zhi Yu, Andreas Artmeier, Lianjun Dai, and Can Wang. Web accessibility evaluation in a crowdsourcing-based system with expertise-based decision strategy. In *Proceedings of the 15th International Web for All Conference*, pages 1–4, 2018.

[50] Alexander Hambley. Empirical web accessibility evaluation for blind web users. *ACM SIGACCESS Accessibility and Computing*, (129):1–5, 2021.

[51] Stéphanie Giraud, Pierre Thérouanne, and Dirk D Steiner. Web accessibility: Filtering redundant and irrelevant information improves website usability for blind users. *International Journal of Human-Computer Studies*, 111:23–35, 2018.

[52] Rafaela Mazalu and Alejandra Cechich. Web accessibility assessment through multi-agent support for visually impaired users. *International Journal of Cooperative Information Systems*, 29(03): 2050005, 2020.

[53] Ahmed Hassan and Rasha Abd El Aziz. Social networks website accessibility for people with visual disability: An experimental study. *JRL of the Faculty of Commerce for Scientific Research*, 54(02), 2017.

[54] Kowndinya Boyalakuntla, Akhila Sri Manasa Venigalla, and Sridhar Chimalakonda. Waccess–a web accessibility tool based on the latest wcag 2.2 guidelines. *arXiv preprint arXiv:2107.06799*, 2021.

[55] Jens Pelzetter. A declarative model for accessibility requirements. In *Proceedings of the 17th International Web for All Conference*, pages 1–10, 2020.

[56] Mohammad Bajammal and Ali Mesbah. Semantic web accessibility testing via hierarchical visual analysis. In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pages 1610–1621. IEEE, 2021.

[57] Sami Rojbi, Anis Rojbi, and Mohamed Salah Gouider. Enhancing the accessibility of images on the web: A holistic view and new perspectives.

[58] Eleni Michailidou, Sukru Eraslan, Yeliz Yesilada, and Simon Harper. Automated prediction of visual complexity of web pages: Tools and evaluations. *International Journal of Human-Computer Studies*, 145:102523, 2021.

[59] Carlos Duarte, Inês Matos, and Luís Carriço. Semantic content analysis supporting web accessibility evaluation. In *Proceedings of the 15th International Web for All Conference*, pages 1–4, 2018.

[60] Iyad Abu Doush, Faisal Alkhateeb, Eslam Al Maghayreh, and Mohammed Azmi Al-Betar. The design of ria accessibility evaluation tool. *Advances in Engineering Software*, 57:1–7, 2013.

[61] Raju Shrestha. A neural network model and framework for an automatic evaluation of image descriptions based on ncam image accessibility guidelines. In *Proceedings of the 2021 4th Artificial Intelligence and Cloud Computing Conference*, pages 68–73, 2021.

[62] Carlos Duarte, Ana Salvado, M Elgin Akpinar, Yeliz Yeşilada, and Luís Carriço. Automatic role detection of visual elements of web pages for automatic accessibility evaluation. In *Proceedings of the 15th International Web for All Conference*, pages 1–4, 2018.

[63] Nádia Fernandes, Daniel Costa, Carlos Duarte, and Luís Carriço. Evaluating the accessibility of web applications. *Procedia Computer Science*, 14:28–35, 2012.

[64] João Dias, Diana Carvalho, Tânia Rocha, and João Barroso. Automated evaluation tools for web and mobile accessibility: proposal of a new adaptive interface tool. *Procedia Computer Science*, 204:297–304, 2022.

[65] André Pimenta Freire and Renata Pontin de Mattos Fortes. Automatic accessibility evaluation of dynamic web pages generated through xslt. In *Proceedings of the 2005 International Cross-Disciplinary Workshop on Web Accessibility (W4A)*, pages 81–84, 2005.

[66] Rodrigo Bonacin, Julio Cesar dos Reis, and Ricardo José de Araujo. An ontology-based framework for improving color vision deficiency accessibility. *Universal Access in the Information Society*, 21 (3):691–716, 2022.

[67] Tarmo Robal, Jevgeni Marenkov, and Ahto Kalja. Ontology design for automatic evaluation of web user interface usability. In *2017 Portland International Conference on Management of Engineering and Technology (PICMET)*, pages 1–8. IEEE, 2017.

[68] José R Hilera, Salvador Otón, C Martin-Amor, and C Timbi-Sisalima. Towards a service-based architecture for web accessibility federated evaluation. In *9th International Conference on Advances in Computer-Human Interactions (ACHI'16)*, pages 6–10, 2016.

[69] Paola Ingavélez-Guerra, Vladimir Robles-Bykbaev, S Oton, Pablo Vera-Rea, Jorge Galán-Men, Marlon Ulloa-Amaya, and JR Hilera. A proposal based on knowledge modeling and ontologies to support the accessibility evaluation process of learning objects. In *2018 Congreso Argentino de Ciencias de la Informática y Desarrollos de Investigación (CACIDI)*, pages 1–5. IEEE, 2018.

[70] Abdullah Alsaeedi. Comparing web accessibility evaluation tools and evaluating the accessibility of webpages: proposed frameworks. *Information*, 11(1):40, 2020.

[71] WMauve++. URL https://mauve.isti.cnr.it/about.jsp.

[72] Nibbler. URL https://nibbler.silktide.com/en_US/about.

[73] WAVE. URL https://wave.webaim.org/about.

[74] WEB accessibility. URL https://www.webaccessibility.com/.

[75] Zainab AlMeraj, Fatima Boujarwah, Dari Alhuwail, and Rumana Qadri. Evaluating the accessibility of higher education institution websites in the state of kuwait: empirical evidence. *Universal Access in the Information Society*, 20(1):121–138, 2021.

[76] Sarah Alismail and Wallace Chipidza. Accessibility evaluation of covid-19 vaccine registration websites across the united states. *Journal of the American Medical Informatics Association*, 28 (9):1990–1995, 2021.

[77] System Usability Scale. URL https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html.

[78] Majed Alshamari. Accessibility evaluation of arabic e-commerce web sites using automated tools. *Journal of Software Engineering and Applications*, 9(9):439–451, 2016.

[79] KS Kuppusamy and Venkatesan Balaji. Evaluating web accessibility of educational institutions websites using a variable magnitude approach. *Universal Access in the Information Society*, 22 (1):241–250, 2023.

[80] Marco Manca, Vanessa Palumbo, Fabio Paternò, and Carmen Santoro. The transparency of automatic web accessibility evaluation tools: design criteria, state of the art, and user perception. *ACM Transactions on Accessible Computing*, 16(1):1–36, 2023.

[81] Antonio Giovanni Schiavone and Fabio Paternò. An extensible environment for guideline-based accessibility evaluation of dynamic web applications. *Universal access in the information society*, 14(1):111–132, 2015.

[82] Omar Jawabreh, Ashraf Jahmani, B Maaiah, and A Basel. Evaluation of the contents of the five stars hotel website and customer orientation. *Information Sciences Letters*, 11(4):1077–1085, 2022.

[83] Jinat Ara and Cecilia Sik-Lanyi. Accguideliner: Towards a modelling approach of web accessibility requirements following wcag 2.2. In *2023 IEEE International Conference on Smart Information Systems and Technologies (SIST)*, pages 10–15. IEEE, 2023.

[84] Sebastian Raschka. Model evaluation, model selection, and algorithm selection in machine learning. *arXiv preprint arXiv:1811.12808*, 2018.

[85] Emma Izquierdo-Verdiguier and Raúl Zurita-Milla. An evaluation of guided regularized random forest for classification and regression tasks in remote sensing. *International Journal of Applied Earth Observation and Geoinformation*, 88:102051, 2020.

[86] Deepti Sisodia and Dilip Singh Sisodia. Prediction of diabetes using classification algorithms. *Procedia computer science*, 132:1578–1585, 2018.

[87] Andrius Vabalas, Emma Gowen, Ellen Poliakoff, and Alexander J Casson. Machine learning algorithm validation with a limited sample size. *PloS one*, 14(11):e0224365, 2019.

[88] Li Yang and Abdallah Shami. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, 415:295–316, 2020.

[89] Jia Wu, Xiu-Yun Chen, Hao Zhang, Li-Dong Xiong, Hang Lei, and Si-Hao Deng. Hyperparameter optimization for machine learning models based on bayesian optimization. *Journal of Electronic Science and Technology*, 17(1):26–40, 2019.

[90] Goran Matošević, Jasminka Dobša, and Dunja Mladenić. Using machine learning for web page classification in search engine optimization. *Future Internet*, 13(1):9, 2021.

[91] Jinat Ara, Cecilia Sik-Lanyi, and Arpad Kelemen. An integrated variable-magnitude approach for accessibility evaluation of healthcare institute web pages. *Applied Sciences*, 13(2):932, 2023.

[92] Sandra Souza Rodrigues, Patrick Eduardo Scuracchio, and Renata Pontin de Mattos Fortes. A support to evaluate web accessibility and usability issues for older adults. In *Proceedings of the 8th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-Exclusion*, pages 97–103, 2018.

[93] Jinat Ara, Cecilia Sik-Lanyi, and Arpad Kelemen. Accessibility engineering in web evaluation process: a systematic literature review. *Universal Access in the Information Society*, 23(2):653–686, 2024.

[94] Sukru Eraslan, Victoria Yaneva, Yeliz Yesilada, and Simon Harper. Web users with autism: eye tracking evidence for differences. *Behaviour & Information Technology*, 38(7):678–700, 2019.

[95] Jinat Ara, Md Toufique Hasan, Abdullah Al Omar, and Hanif Bhuiyan. Understanding customer sentiment: Lexical analysis of restaurant reviews. In *2020 IEEE Region 10 Symposium (TENSYMP)*, pages 295–299. IEEE, 2020.

[96] Nuha Awlad Karaim and Yavuz Inal. Usability and accessibility evaluation of libyan government websites. *Universal Access in the Information Society*, 18:207–216, 2019.

[97] Barbara Leporini, Marina Buzzi, and Marion Hersh. Video conferencing tools: Comparative study of the experiences of screen reader users and the development of more inclusive design guidelines. *ACM Transactions on Accessible Computing*, 16(1):1–36, 2023.

[98] Text simplification guidelines, inclusive design research centre. *OCAD University, Toronto, Ontario, Canada (1998).* URL https://snow.idrc.ocadu.ca/accessible-media-and-documents/text-simplification-guidlines/.

[99] Ombretta Gaggi and Veronica Pederiva. Wcag4all, a tool for making web accessibility rules accessible. In *2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*, pages 1–6. IEEE, 2021.

[100] Understanding the w3c's accessibility conformance testing (act) requirements. *Bureau of Internet Accessibility, East Greenwich, US*, 2023. URL https://www.boia.org/blog/understanding-the-w3cs-accessibility-conformance-testing-act-requirements.

[101] Kim Mens and Andy Kellens. Mining source code for design regularities. *Working Session on Industrial Realities of Program Comprehension (IRPC 2008)*, 2008.

[102] Abhinav Sharma, M Revathi, et al. Automated api testing. In *2018 3rd International Conference on Inventive Computation Technologies (ICICT)*, pages 788–791. IEEE, 2018.

[103] Jens Pelzetter. Using ontologies as a foundation for web accessibility tools. In *Proceedings of the 15th International Web for All Conference*, pages 1–2, 2018.

[104] Vincenzo Rubano, Fabio Vitali, et al. Experiences from declarative markup to improve the accessibility of html. In *Balisage: The Markup Conference*, pages 1–22, 2020.

[105] Milton Campoverde-Molina, Sergio Luján-Mora, and Llorenç Valverde. Process model for continuous testing of web accessibility. *IEEE access*, 9:139576–139593, 2021.

[106] Nicola Iannuzzi, Marco Manca, Fabio Paternò, and Carmen Santoro. Usability and transparency in the design of a tool for automatic support for web accessibility validation. *Universal Access in the Information Society*, 23(1):435–454, 2024.

[107] Iyad Abu Doush, Khalid Sultan, Mohammed Azmi Al-Betar, Zainab Almeraj, Zaid Abdi Alkareem Alyasseri, and Mohammed A Awadallah. Web accessibility automatic evaluation tools: to what extent can they be automated? *CCF Transactions on Pervasive Computing and Interaction*, 5 (3):288–320, 2023.