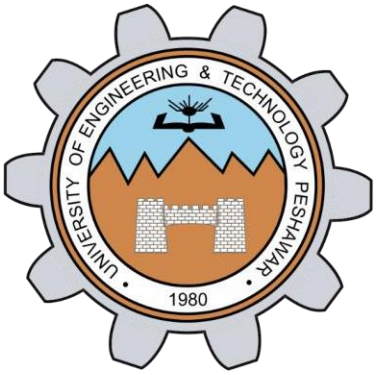
UNIVERSITY OF ENGINEERING AND TECHNOLOGY, PESHAWAR PAKISTAN

Main Campus



Software Engineering Lab

Feasibility Report (Fall 2025)

Submitted By

Abdul Baseer	**22PWBCS0913**
Muhammad Mohsin	**23PWBCS0973**
Saad Abdullah	**23PWBCS1055**
Hooria Altaf	**23PWBCS0969**

(22pwbcs0913@uetpeshawar.edu.pk)

(23pwbcs0973@uetpeshawar.edu.pk)

(23pwbcs1055@uetpeshawar.edu.pk)

(23pwbcs0969@uetpeshawar.edu.pk)

Supervisor : Ms. Kanwal

DEPARTMENT OF COMPUTER SCIENCE & IT

UNIVERSITY OF ENGINEERING AND TECHNOLOGY, PESHAWAR, PAKISTAN

Website Accessibility Simulator

Abstract

The project focuses on improving the user experience of disabled groups using websites. The proposed system will allow developers to simulate various disability conditions within the websites environment. This will enable the developers to experience firsthand how the disabled person navigates his website. Thus allowing the developer to make better decisions and improving user experience drastically. Given the current open-source tools and information the project is considered feasible within a three month timeframe.

Introduction
2. Feasibility Analysis

2.1 Technical Feasibility

The system will be developed using Javascript for the backend and MongoDB database.
Hardware: University already has servers and computer labs with internet connectivity.
Software: Open-source tools will be used, minimizing additional expenses furthermore much of the components required for this project have not yet been developed by anyone, thus those unavailable components will be developed from scratch using JavaScript.
Conclusion: Technically feasible within a three month timeframe.

2.2 Economic Feasibility

Cost estimation: Direct labour costs are low because university students will be working on the project. The primary expenses consist of software tools (mostly open-source), domain registration, and hosting. Web development tools and accessibility principles are already well-known to the development team and end users (developers). Using the simulator interface could require some preliminary instruction or documentation.

Expected benefits

Cost Savings: By identifying accessibility problems early in the design process, developers can lower the cost of fixes after the product is released.

Enhanced Efficiency: Provides quick, simulated feedback without the need for physical testing with assistive devices.

Increased Market Reach: Websites can reach a wider audience by making themselves more accessible, which could lead to higher engagement and income.

2.3 Operational Feasibility

User acceptance and readiness:
Technically proficient web developers are the target audience, and they are likely to use tools that increase the accessibility of their product. High acceptability is ensured by the project's direct resolution of a prevalent accessibility testing pain point.

Training and Support Are Necessary:
There will be little training needed. The UI of the simulator is made to be easy to use. The tool's brief help or guide module will be enough for users to comprehend every feature.

Fit with organizational processes:
The simulator fits seamlessly into the existing web development workflow:
Can be used during the testing or QA phase.
Compatible with commonly used technologies (MERN stack, JavaScript).
Aligns with global standards like WCAG 2.1.

2.4 Schedule Feasibility

Estimated project timeline:
Requirement Gathering: 2 weeks

Design: 2 weeks
Development: 6-8 weeks
Testing and Debugging: 2 weeks
Total Estimated Duration: 12-14 weeks
Deadline Feasibility:
The project can be completed within deadlines if milestones are tracked efficiently.
Potential risks causing delay:
Difficulties integrating with third-party web apps.
Simulation effects that are incompatible with some browsers.
Delays in features for report generation or testing.

2.5 Legal Feasibility

Data Privacy/Security Issues:

It is essential to handle any accessed data responsibly because the simulator will let developers enter URLs of potentially private or under-developed applications. No user information, URLs, or web content will be permanently stored or logged by the system. To prevent any unwanted access or data leaks, all simulations will operate in a sandbox. To reduce security risks, any processed data will be local to the user session.

Regulatory and Compliance Requirements:

The project complies with international accessibility standards, including the W3C's Web Content Accessibility Guidelines (WCAG) 2.1, which is referenced in the proposal. Even though the simulator isn't currently meant for public use, if it is, it must handle user data in accordance with GDPR or other local data protection regulations. Furthermore, the tool's legal consequences are low because it assists with compliance rather than being used directly by people with disabilities; however, care must still be taken to ensure simulation accuracy and ethical representation of impairments.

3. Results and Discussion

Key Findings from Each Feasibility Aspect:

Technical Feasibility:

It is possible to simulate impairments with front-end technologies like React. Accessibility simulations can be supported by existing libraries (like the Web Speech API.

Operational Feasibility:

The system offers developers a useful tool that can enhance inclusivity in the real world by giving them immediate behavioral and visual feedback on accessibility issues. Both usability and instructional value were considered in the tool's design.

Legal Feasibility:

The technology functions in a manner that conforms to accessibility guidelines and avoids data collecting. Legal hazards are reduced because no sensitive user data is kept.

Challenges and Risks Identified:

• It can be difficult and risky to accurately simulate real impairments (particularly cognitive ones) in a fashion that is universally representative.
• To prevent legal misunderstandings, exported reports must be precise and cautious in their wording.

Assumptions Made:

• The simulator will be used by developers in morally responsible situations (e.g., testing their own apps or with consent).
• The required APIs, such as the Web Speech API for screen reader simulation, will be supported by the browsers being used.
• The simulator is not intended for official certification or evaluation of legal compliance; rather, it is intended solely for development and educational purposes.

4. Conclusion

Yes, the project is **legally and technically feasible** within the current scope.

The tool focuses on developer-side testing in a secure setting, avoids touching sensitive personal data, and complies with recognized web standards (WCAG). It bridges the gap between usability testing and accessibility compliance with designed simulations and exportable reports, making it a useful and workable solution for the development community.

References

[1]W. W. W. C. (W3C), "Web Content Accessibility Guidelines (WCAG) 2.1," 2018. [Online]. Available: https://www.w3.org/TR/WCAG21/.

[2] World Wide Web Consortium (W3C), “Web Content Accessibility Guidelines (WCAG) 2.1,” W3C Recommendation, 2018. [Online]. Available: https://www.w3.org/TR/WCAG21/

[3] N. R. Kamaruddin, M. S. Hashim, and N. A. Rahman, “Web Accessibility Evaluation Tools: A Systematic Review,” 2020 IEEE International Conference on Computer and Communication Engineering (ICCCE), Kuala Lumpur, Malaysia, 2020, pp. 59–64.