AI system personalizes software interfaces in real time using user behavior and feedback.

(Improving browsing experience for Users with Visual Disabilities.)

TMP-24-25J-296

Project Proposal Report

H.L.Sandun Geemal - IT17117210

B.Sc. (Hons) in Information Technology Specializing in Information Technology

Faculty of Computing

Sri Lanka Institute of Information Technology
Sri Lanka

August 2024

AI system personalizes software interfaces in real time using user behavior and feedback.

(Improving browsing experience for Users with Visual Disabilities.)

TMP-24-25J-296

Project Proposal Report

B.Sc. (Hons) in Information Technology Specializing in
Information Technology

Faculty of Computing

Sri Lanka Institute of Information Technology
Sri Lanka

August 2024

Declaration

I declare that this is my own work, and this proposal does not incorporate without acknowledgementany material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously publish or written by another person expect where the acknowledgement is made in the text.

Name	Student ID	Signature
H.L. Sandun Geemal	IT17117210	Sandun

The above candidate is carrying out research for the undergraduate Dissertation under my supervision.

…………………………..

Signature of the Supervisor:

(Mr. Thusithanjana Thilakarthna)   Date:

…………………………..

Signature of the Co-supervisor:

(Ms. Rivoni De Zoysa)   Date:

Abstract

As the world becomes more and more computerized ordinary people with vision impairments experience great difficulties when it comes to their interaction with computers. The research proposed in this paper aims primarily at the creation of an adaptive AI-assisted system designed for instant customization of graphical user interfaces, with an example use of the browsing functionality for the disabled, namely the visually impaired. The goal of occupying user behavior and feedback is to make web content more accessible to the disabled in particular, the color-blind, and short-sighted users. To achieve this, the technique to be used in the proposed solution will be the interactive tests to identify the users and their required vision and perception patterns that will allow the interface to be dynamic in order to correspond with user vision demands.

There are a number of subsystems that constitute the overall system architecture and they are data capturing, Machine learning algorithm, and dynamic changes to the interface. User interaction data will be collected, and the results will be stored in MongoDB database and analyzed by using python and machine learning. In this case, we will determine trends of users' behaviour in order for the system to set required parameters of the browser like, size of text, contrast and colors according to profiles of users. That is why the system will utilize CSS variables and WAI-ARIA standards so that adjustments will be effective and SEP-compliant.

Integration of feedback mechanisms will be incorporated to ensure that user experience is constantly created and improved. Self-made online questionnaires as well as other measurement methods will be applied to investigate the satisfaction of the users and the efficacy of the adaptive aspects. This iterative development cycle will help to factor in changes as users change meaning that the system will grow with the users. Finally, it is the hope of this research to make the digital world a level playing field for the users with visual impairment to be enabled and perform their tasks online optimally as any other user.

Keywords: Visual Impairments, Adaptive AI System,User Interfaces,Customization Graphical User Interfaces (GUIs), Accessibility,Color Blindness, Short-Sightedness

Table of Contents

Contents

List of Figures

List of Tables

List of Abbreviations

AI	Artificial intelligence
AUI	Adaptive user interfaces
JAWS	job Access With Speech

1.0 Introduction

1.1 Background

Making websites accessible has become a key part of web development, as people are recognizing how important it is to design for everyone, especially those with disabilities like vision problems.The World Health Organization estimates that over 2.2 billion people worldwide suffer from some form of visual impairment or blindness, making accessibility a critical factor in digital inclusion.While guidelines like the Web Content Accessibility Guidelines (WCAG) provide standards for improving accessibility, much of the existing literature highlights gaps in real-time adaptability and personalized solutions tailored to the needs of visually impaired users.

Recent AI and machine learning breakthroughs have created an entirely new possibility for interface personalization. Adaptive systems can learn from user gestures by using hardware collection and presentation techniques. You can gather data for your own system programs in this way, and each new piece of data will be used by the next algorithm to better serve the needs of its users. For instance, an embodiment of the approach is using interactive tests to diagnose a particular visual impairment, and modifying text size, contrast and color scheme according to individual preferences. This real-time adjustability not only makes Web sites accessible to users with visual disabilities but also gives them a systemic control over their environment. Users can engage with digital content in a manner that suits their unique requirements.

Additionally, there are real-time feedback loops to be integrated into adaptive systems for enhancing the user experience. Researches have shown that systems capable of processing user feedback are able to change interface settings (such as for instance text size, contrast or color schemes) what makes browsing experience more interactive and responsive(not only on web pages background). It is this real-time adaptability that makes dealing with the changeable-content behavior of users—who are operating in different environments than just when they go to a “site”—so important, hence an event-driven model being applied. Through the use of WebSockets and tools such as Firebase, developers can help ensure changes are applied in real-time to greatly improve digital interface usability.

1.2 Literature Review

Studied have revealed that adaptive user interfaces have the ability to enhance users interaction up to a great extent, by reducing the cognitive load and enhancing compatibility for its users and especially the ones with visual impairments. Gullà et al. (2015) have presented an approach of user modelling in interfaces which is extremely important for designing interfaces which meet the needs of visually impaired users because it involves using user profiles[1]. The authors point out that communication and graphic interfaces are not only utilitarian tools but user-centered designs that take into account user's demographics, psychographic features, and behavioral patterns on the Internet.

Further research has explored the design of human-AI interactions within accessibility technologies. Bigham et al. (2011) focus on real-time AI-driven tools that can dynamically adapt web content based on user feedback[2]. Their work suggests that the application of advanced artificial intelligence might help overcome some of the existing issues that are relevant to visually impaired users – for example, to adapt the contrast of the colors and texts or the size of the given texts, in a real time, which would make the process of browsing more convenient.This aligns with the objectives of developing interactive tests and utilizing data analysis to create a more personalized and accessible browsing experience.

Another significant paper in the area is the review of the AI web accessibility tools. Gleason et al. (2021) similarly devote much attention to the methods that AI can use to make modifications to the graphical user interface for the visitors, who are blind or have low vision such as the ability to change the color contrast and the font size. From the results of their work, the authors emphasize that engagement with the help of machine learning concepts contributes to the improvement of the interaction process because individualization corresponds to the browsing session. Thus, for one to appreciate how relaxing developmental constraints of current systems yields positive results in respect to usability for all, particularly for the challenged users who have impaired vision.

Finally, Jiang, Zhou, and Liu also demonstrated in research conducted in 2019, that such an approach requires befitting web accessibility solutions. In their work, they also opine about the use of machine learning to develop structure that adapts to the use of the user in question. They contend that such systems can make the Internet better for use by persons with disability since the systems will help in the easy and efficient access of data. This evidence is a clear reminder that people's needs are constantly changing, hence the need to incorporate the design interfaces that are also equally changing.

1.2 Research Gap

The research on improving web accessibility through AI and machine learning has made considerable advancements, especially in terms of enhancing usability for users with visual impairments. Previous solutions have primarily focused on static or semi-dynamic approaches to adjusting web content, which lack the ability to fully personalize the browsing experience in real-time. For instance, Lazar and Stein (2017) emphasized the importance of AI-driven user interfaces that adapt to the needs of visually impaired users but did not incorporate real-time data collection to dynamically optimize the interface. Similarly, Bigham, Ladner, and Borodin (2011) proposed human-AI interaction models that adjust web settings such as contrast and text size; however, these systems often rely on predefined configurations rather than continuously evolving in response to real-time user input.

The novelty of the proposed system lies in its ability to dynamically adjust browser settings and content presentation based on real-time user data, providing a more responsive and personalized browsing experience. By incorporating interactive tests, this system can identify specific visual impairments such as color blindness, or short-sightedness. This real-time feedback loop not only optimizes the browsing experience but also enables continuous adaptation and customization, which goes beyond the capabilities of previous solutions.

This innovation fills the research gap by offering a more fluid, adaptive, and data-driven approach to web accessibility, ensuring that the browsing experience is continuously optimized to meet the specific needs of users with visual disabilities.

There is several tools that already developed targeting visual impaired users.A range of these solutions like screen readers (JAWS, NVDA) are centered around converting text to speech and browsing a web page with no mechanisms for real-time adaption in response to individual behaviour or preferences. For example, JAWS is great at reading content aloud but only if the HTML that supports it (this is not always well-structured due to modern web design) allows for a true understanding of what can actually be interacted with on a page. Complex web applications offer little separation to account for this, so visually impaired users are often left behind and frustrated by the slow pace of content delivery. Additionally, WAVE is essential for identifying accessibility issues but do not offer immediate solutions or adaptations. They require manual intervention to implement changes, which can be a barrier for users who may not have the technical skills or knowledge to make the necessary adjustments.

Product/ Tool	**Key features**	**Technology**	**Limitations**
JAWS	Screen reader with speech and Braille output	Text-to-speech, Braille display	Limited real-time adaptation, no dynamic interface adjustment
NVDA (NonVisual Desktop Access)	Free screen reader with text-to-speech conversion	Text-to-speech, keyboard navigation	Static interface, lacks real-time data-based adaptation
Readability	Simplifies web pages, adjusts text size and contrast	Browser extension	Manual adjustments required, lacks real-time adaptation
Mercury Reader	Removes clutter from web pages, customizable text and background	Browser extension	Limited to browser compatibility; no real-time adaptation
WAVE	Identifies accessibility issues, provides detailed reports	Web-based evaluation tool	Does not provide real-time adjustments; requires manual intervention

Table 1: Research Gap Among Existing Software/ Tools

1.3 Research Problem

The increasing reliance on digital technology has highlighted the necessity for accessible software interfaces, particularly for users with visual disabilities. As the digital landscape continues to evolve, the demand for adaptive user interfaces (AUIs) that can cater to the diverse needs of users has become paramount. AUIs are designed to adjust their functionalities based on user characteristics and preferences, allowing for a more personalized and engaging experience. This adaptability is essential for users with visual impairments, who often face significant barriers when interacting with standard web interfaces.

Despite the advancements in adaptive user interfaces, significant gaps remain in the existing literature, particularly concerning the dynamic adjustment of browser settings and content presentation based on real-time user data. Most current solutions focus on either static adaptations or require users to manually customize their settings, which can be a barrier for individuals with visual impairments. For instance, while some studies discuss the importance of designing adaptable systems that cater to diverse user needs, they primarily emphasize user profiles and predefined settings without addressing the need for real-time adjustments based on individual interactions. This limitation can hinder the effectiveness of these systems in providing a truly personalized experience for visually impaired users, who may require immediate and context-sensitive adaptations to their browsing environments.

In summary, the research problem lies in the lack of adaptive user interfaces that dynamically adjust browser settings and content presentation based on real-time user data. By providing interactive tests to identify visual impairments and automatically optimizing the browsing experience, this approach aims to fill the gaps identified in existing literature. The emphasis on real-time adaptation and customization represents a significant advancement in the development of accessible technologies for users with visual disabilities.

2.0 Objectives

2.1 Main Objective

Improving browsing experience for Users with Visual Disabilities.

To develop an system that dynamically adjusts browser settings and content presentation based on real-time user data, with a focus on improving the browsing experience for users with visual disabilities.

2.2 Specific Objectives

- Develop Interactive Tests for Visual Impairments

Apply machine learning models trained on the dataset to analyze user responses in real-time.

- Dynamic Adjustment of Browser Settings

Dynamically adjust text sizes, contrast settings, and color schemes based on user test results

- Develop Customizable User Interface

Allows users to customize their browser settings based on their accessibility preferences.

- Implement Real-Time Adaptation

ensuring immediate application of changes without requiring page reloads

3.0 Methodology

3.1 System Overview

The AI-driven web plugin aims to improve user experiences by customizing software interfaces and interactions in real time, with a focus on both normal users and those with visual impairments. System using two primary data sources

- Existing dataset (Ishihara MNIST dataset) - specialized collection of images inspired by the traditional Ishihara color blindness test, combined with elements of the classic MNIST dataset. It features numerals displayed in a manner similar to Ishihara plates, making it suitable for training machine learning models to detect color vision deficiencies, particularly focusing on recognizing patterns affected by color blindness.

- Interactive test results - By analyzing user responses through these tests, the system identifies specific accessibility needs, allowing for real-time adjustments to improve the browsing experience for visually impaired users.

The proposed research employs a comprehensive methodology that integrates various technologies and machine learning techniques to enhance the browsing experience for users with visual disabilities. The primary goal is to develop an adaptive user interface that dynamically adjusts based on real-time user data, ensuring a personalized and accessible online experience.1. Interactive Tests Development

The first task involves creating interactive tests using JavaScript and WebAssembly to identify visual impairments such as color blindness and short-sightedness. By utilizing libraries like D3.js, these tests will provide visual assessments and data visualizations that help users understand their specific needs. This foundational step is crucial for tailoring the browsing experience to individual users, as it allows the system to gather essential data on user capabilities.2. Data Collection and Analysis

Once the interactive tests are conducted, the results and user interaction data will be stored in MongoDB for efficient data management. The analysis of this data will be performed using Python, leveraging machine learning models through scikit-learn. This analysis will identify specific accessibility needs and patterns in user behavior, allowing the system to make informed adjustments to the user interface.3. Dynamic Adjustment of Browser Settings

The system will implement features that dynamically adjust browser settings, including text sizes, contrast settings, and color schemes, based on the results of the interactive tests. This will be achieved using CSS variables and custom properties, ensuring that changes are visually effective and user-friendly. Additionally, the integration of WAI-ARIA standards will enhance semantic content accessibility, making the browsing experience more intuitive for visually impaired users.4. Customizable User Interface
To further improve user experience, a customizable user interface will be developed

using React and Material-UI. This interface will allow users to manage their accessibility preferences easily, enabling them to tailor their browsing experience according to their specific needs and preferences.5. Real-Time Adaptation

The proposed system will utilize WebSockets and Firebase for real-time data synchronization, allowing for immediate updates to browser settings without requiring page reloads. This real-time adaptation is essential for creating a seamless and responsive experience, ensuring that users receive instant adjustments based on their interactions.6. User Feedback and Improvement

Finally, user feedback will be integrated through forms and surveys developed with React and Node.js. This feedback will be essential for monitoring user satisfaction and identifying areas for improvement. Analytics tools like Matomo will help track user interactions and preferences, enabling continuous enhancement of the adaptive system.In summary, the methodology combines advanced web technologies and machine learning techniques to create a dynamic and responsive browsing experience for users with visual disabilities. By focusing on real-time adaptations and user-driven customization, this approach aims to significantly improve accessibility and usability in digital environments.

3.2 System Architecture Diagram

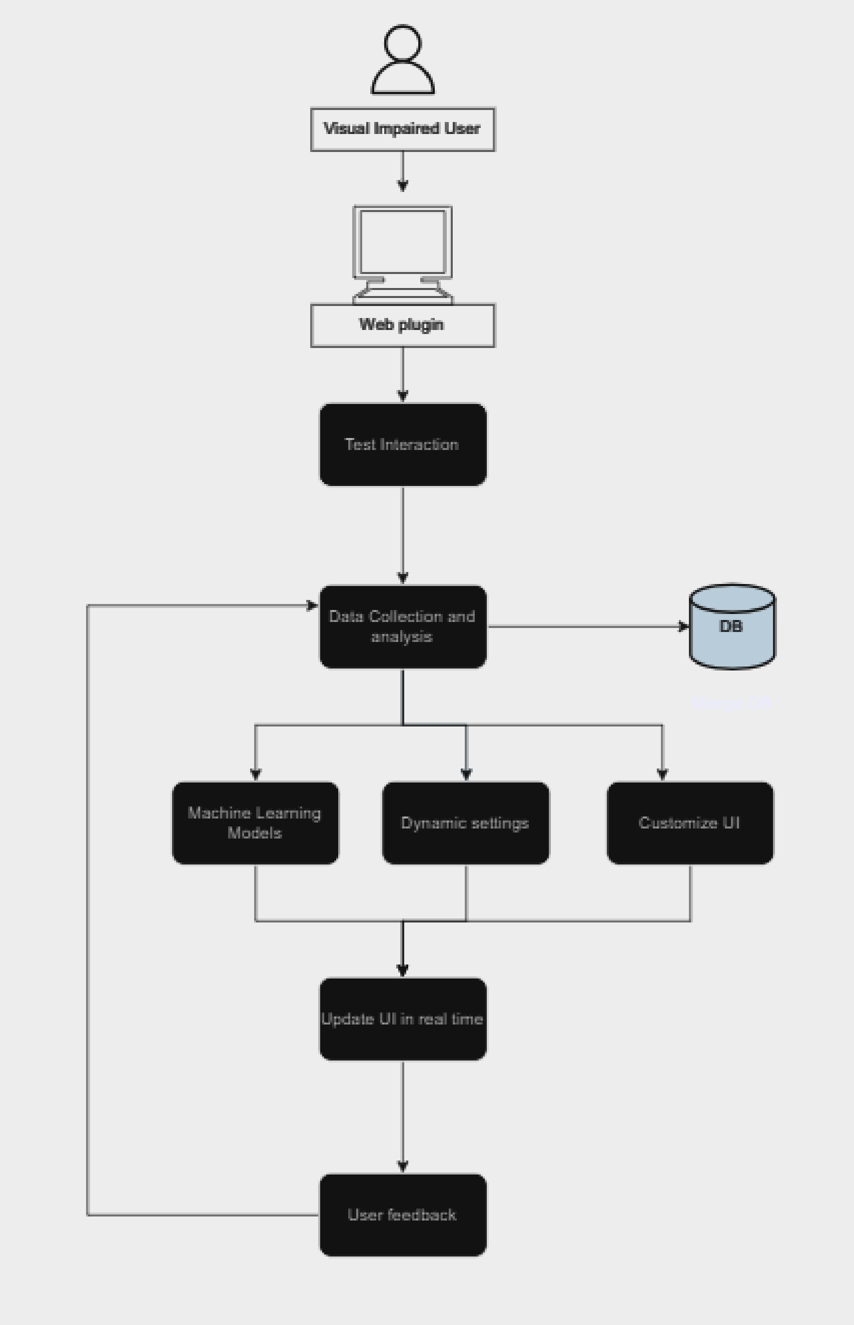


Figure 1: High-Level System Diagram for the Research Project

3.3 Identifying Existing Systems

Existing systems often focus on either static adaptations or require users to engage in manual customization, which can be particularly challenging for individuals with disabilities. For instance, while adaptable interfaces empower users to tailor their experience, they can also pose barriers for those with lower levels of digital literacy. Conversely, fully adaptive systems, which learn from user interactions, have the potential to reduce cognitive load and enhance usability but may raise concerns regarding privacy and user trust.Moreover, many current solutions do not incorporate real-time feedback mechanisms that allow for immediate adaptations based on user behavior. This limitation underscores the need for a more integrated approach that combines interactive assessments to identify visual impairments with dynamic adjustments to browser settings. By analyzing existing systems, it becomes evident that there is a significant gap in providing a comprehensive solution that dynamically adjusts to user needs in real-time, particularly for users with visual disabilities. The proposed research aims to address these gaps by developing a system that not only identifies individual visual impairments but also optimizes the browsing experience through continuous adaptation and user-driven customization.

3.4 Project Requirements

3.4.1 System requirements

It is clear that any adaptation of software user interface for the visually impaired imposes the need to provide tools and facilities for ongoing real time data capture, analysis, and feedback. This entails the put in place of effective systems of recording and collection of data, right and appropriate hardware and software and data and network technology respectively. In addition to a productive data processing capacity, the integration to existing web applications and the possibility of modifying the layout according to the distinct use cases are part of this system as well.

3.4.1.1 Functional Requirements

1. Interactive Tests

- Requirement: The system must provide interactive tests that accurately identify specific visual impairments, such as color blindness and short-sightedness, to tailor the browsing experience accordingly.
- Sub-Tasks:

  Test Development: Develop interactive tests using JavaScript and WebAssembly that assess various aspects of visual function, including color perception, contrast sensitivity, and visual acuity.

Test Accuracy: Ensure that the interactive tests provide reliable and valid results by incorporating established assessment methods and validated test stimuli.

Test Usability: Design the interactive tests to be user-friendly and accessible, minimizing cognitive load and providing clear instructions for participants.

2. Data Collection and Analysis

- Requirement: The system must collect and analyze user data from the interactive tests and browsing interactions to inform real-time adaptations and identify accessibility needs.
- Sub-Tasks:

Data Storage: Store user test results and interaction data securely in a MongoDB database for efficient retrieval and analysis.

Data Analysis: Apply machine learning models using Python and scikit-learn to analyze the collected data, identify patterns, and determine specific accessibility requirements for each user.

Adaptation Recommendations: Based on the data analysis, generate recommendations for dynamic adjustments to browser settings and content presentation to optimize the browsing experience for each user.

3. Dynamic Browser Adaptation

- Requirement: The system must dynamically adjust browser settings and content presentation in real-time based on user test results and interaction data to provide a personalized and accessible browsing experience.
- Sub-Tasks:

Adjustment Mechanisms: Implement features that allow for immediate changes to text sizes, contrast settings, and color schemes using CSS variables and custom properties.

Semantic Accessibility: Utilize WAI-ARIA standards to enhance semantic content accessibility, ensuring that the adapted content remains meaningful and navigable for users with visual impairments.

Real-Time Synchronization: Use WebSockets and Firebase to synchronize user data and browser settings in real-time, enabling instant updates without requiring page reloads.

3.4.1.2 Non Functional Requirements

- Performance
  The system must provide a high-performance user experience, ensuring that all interactive tests and dynamic adjustments occur with minimal latency.

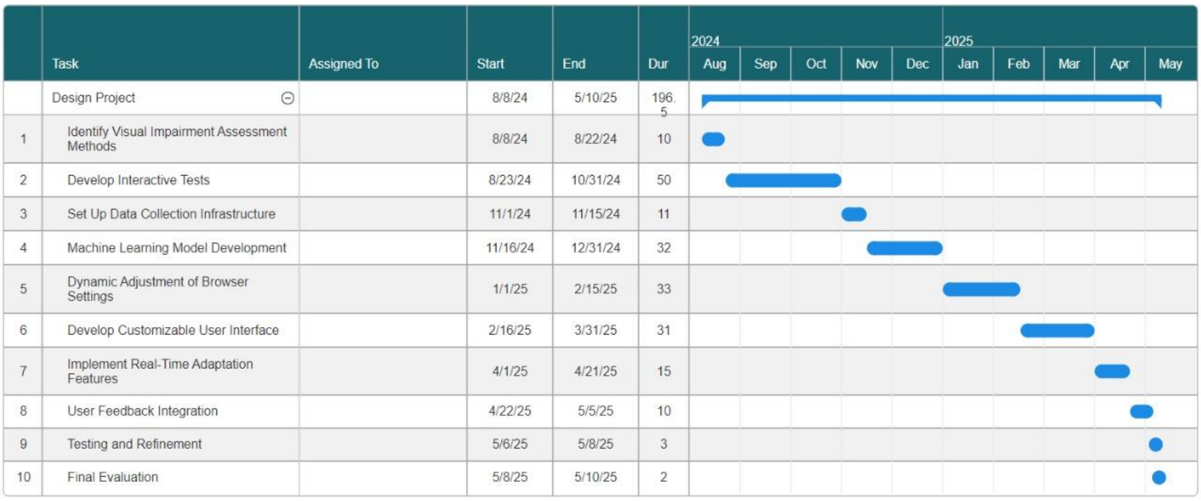
- Usability
  The system must be user-friendly, allowing users of varying technical abilities to navigate and utilize its features effectively.
- Security
  The system must implement robust security measures to protect user data and privacy.
- Scalability
  The system must be scalable to accommodate an increasing number of users and data without compromising performance.
- Reliability
  The system must be reliable, ensuring consistent availability and functionality for users

3.4 Expected test cases

Test ID	Test Description	Test Input	Test Output
01	Verify that interactive tests accurately identify visual impairments	User completes color blindness and short-sightedness tests	Test results correctly identify the user's visual impairments
02	Check if the system dynamically adjusts browser settings based on test results	User takes interactive tests and requests UI changes	Browser settings (text size, contrast, color scheme) are updated in real-time to reflect the user's needs
03	Validate that user test results and preferences are saved across sessions	User logs in, takes tests, logs out, and logs in again	The UI reflects the preferences and settings from the previous session based on the saved test results
04	Ensure the system provides personalized UI suggestions based on test results	User takes interactive tests and interacts with the software multiple times	Personalized UI options and suggestions are provided based on the user's test results and past behavior
05	Test interactive test accuracy and analysis	User takes tests with specific visual impairments	Test results accurately identify the user's visual impairments, and the UI is adjusted accordingly
06	Check system performance under high test load	Multiple users take interactive tests simultaneously	System processes test results without delay and updates UI settings in real-time
07	Validate that the system notifies users of major UI changes based on test results	User takes tests that result in significant UI modifications	User receives a notification detailing the changes made to the UI based on their test results
08	Verify that the interactive tests are user-friendly and accessible	Users with varying levels of technical ability take the tests	Users are able to easily understand and complete the interactive tests

Table 2: test cases

3.5 Timeline



	Task	Assigned To	Start	End	Dur
	Design Project		8/8/24	5/10/25	196.5
1	Identify Visual Impairment Assessment Methods		8/8/24	8/22/24	10
2	Develop Interactive Tests		8/23/24	10/31/24	50
3	Set Up Data Collection Infrastructure		11/1/24	11/15/24	11
4	Machine Learning Model Development		11/16/24	12/31/24	32
5	Dynamic Adjustment of Browser Settings		1/1/25	2/15/25	33
6	Develop Customizable User Interface		2/16/25	3/31/25	31
7	Implement Real-Time Adaptation Features		4/1/25	4/21/25	15
8	User Feedback Integration		4/22/25	5/5/25	10
9	Testing and Refinement		5/6/25	5/8/25	3
10	Final Evaluation		5/8/25	5/10/25	2

Figure 2: Grantt chart of the system

Gantt chart that highlights the timeline for this project as well as the phases that were covered in the development of an adaptive user interface system for the visually impaired. They include the main activities, the targets and the time estimates that can guide the management of the tasks and the resources. It identifies smaller-scale actions, which categorize each task: Creating engaging tests, data gathering and processing, dynamic adjustment of browser settings, and incorporation of users' feedback. The Gantt chart shows the dependencies of tasks through arrows, and this is very helpful in the organization of tasks since it gives direction on the project process in order to complete it on time. It also benefits from that structure by allowing a better view of how the work is going, and where certain parts may be stalled, so that adjustments can be made to keep things going forward. In general, the Gantt chart is one of the tools that is needed to cope with the challenges of the project and that will guarantee the positive outcome of applying the adaptive user interface system.

3.6 Communication management plan

The Communication Management Plan describes how communication will be handled during the development of the Software User Interface Personalization through User Feedback feature. It sets up clear rules for sharing feedback, solving problems quickly, and making sure everyone on the team understands the project's goals and schedule. This will help make communication smooth and effective throughout the development process.

3.7 Communication Media

Communication Media	Purpose	Frequency	Participants	Description
Email	Formal updates, documentation sharing	As needed (at least weekly)	Supervisor, Co-supervisor, Team Members	Used for sharing project status reports, meeting summaries, and formal communication
Weekly Meetings	Status updates, issue resolution	Weekly	Supervisor, Co-supervisor, Project Team	Scheduled virtual or physical meetings to discuss progress, address issues, and plan upcoming tasks
Project Management Tools (e.g., Trello)	Task tracking and progress monitoring	Continuous	Project team	Enables tracking of project tasks, deadlines, and deliverables, ensuring the team stays on schedule
Shared Drive/Cloud Storage (e.g., Google Drive, OneDrive)	Document storage and sharing	Continuous	Project team	Centralized repository for storing project documents, code, designs, and other important files
GitHub/Git	Version control and code sharing	Continuous	Project team	Used for managing code versions, collaborative development, and code reviews.

Table 3: communication media

4.0 Commercialization

The commercialization of the AI-based system for real-time personalization of software user interfaces through user feedback and behavior will be aligned with the resources and capabilities typical of a university research project. The initial focus will be on creating a functional prototype that showcases the system's essential features, such as real-time adaptation of user interfaces based on feedback and user behavior, ultimately improving the browsing experience for visually impaired users.

To promote commercialization, the system will be presented to academic and industry audiences through conferences, publications, and technology exhibitions. This exposure aims to generate interest and gather feedback, which will be crucial for refining the system and exploring potential market applications. Given the project's scope, the initial strategy for commercialization will prioritize licensing the technology to software companies or integrating the system as a plugin into existing platforms. This approach allows for gradual scaling while generating revenue to support ongoing development and research. Collaborations with industry partners and academic institutions may also be pursued to obtain additional support, resources, and expertise necessary for bringing the system to market.

References:

[1] Gullà, F., Ceccacci, S., Germani, M., & Cavalieri, L. (2015). Design Adaptable and Adaptive User Interfaces: A Method to Manage the Information. In Ambient Assisted Living (pp. 47-54). Springer International Publishing. DOI: 10.1007/978-3-319-18374-9_5.

[2] Bigham, J. P., Ladner, R. E., & Borodin, Y. (2011). The Design of Human-AI Interaction in Accessibility Technology. In Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems (pp. 2035-2044). ACM. DOI: 10.1145/1978942.1979244.

[3] Jiang, Y., Zhou, Y., & Liu, W. (2019). Adaptive Web Accessibility via Machine Learning for Users with Disabilities. Universal Access in the Information Society, 18(3), 765-779. DOI: [10.1007/s10209-018-0632-1](https://doi.org

[4] Gleason, C., Ye, Y., Lee, C. P., & Wu, H. (2021). AI-Assisted Web Accessibility for Individuals with Visual Impairments. International Journal of Human-Computer Studies, 146, 102565. DOI: 10.1016/j.ijhcs.2020.102565.