AI System Personalizes Software Interfaces in Real-time Using User Behavior and Feedback

(Software User Interface Personalization through User Feedback in Real-time)

TMP-24-25J-296

Project Proposal Report

Silva H.S.N – IT21324406

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

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

(Software User Interface Personalization through User Feedback in Real-time)

TMP-24-25J-296

Project Proposal Report

Silva H.S.N – IT21324406

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

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 acknowledgement any 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
Silva H.S.N	IT21324406	

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

…………………………..

Signature of the Supervisor:

(Mr. Thusithanjana Thilkarathna) Date: 22/08/2024

…………………………..

Signature of the Co-supervisor:

(Ms. Rivoni De Zoysa) Date: 22/08/2024

Abstract

The purpose of this study is to develop and evaluate a system for Software User Interface (UI) Personalization driven by real-time user feedback. This research addresses the critical need for adaptive and responsive user interfaces that cater to user preferences, thereby enhancing user satisfaction and engagement. The study investigates best practices for integrating user feedback into an AI-driven UI personalization system, emphasizing continuous improvement through iterative feedback loops. The proposed system collects and analyzes real-time feedback, using both quantitative and qualitative measures to dynamically adapt the UI according to user behavior and preferences. By establishing robust feedback mechanisms, the AI system can continuously learn and improve, offering tailored UI suggestions that align with evolving user needs. The basic design of the study involves deploying an AI model capable of real-time adaptation, supported by a comprehensive framework for feedback collection, analysis, and implementation. The findings suggest that effective integration of user feedback into UI personalization not only improves user satisfaction but also drives higher engagement by creating a more intuitive and user-centric software experience. This research concludes that real-time feedback mechanisms are essential for the continuous evolution of personalized interfaces, ensuring that the software remains relevant and user-friendly over time.

Keywords: Software User Interface Personalization, AI-driven Personalization, User Feedback, Real-time Feedback, Continuous Improvement, User Engagement, Adaptive Interfaces, Feedback Mechanisms, Quantitative and Qualitative Feedback, Dynamic UI Adaptation

Table of Contents

List of Figures

List of Tables

List of Abbreviations

UI	User Interface
AI	Artificial Intelligence
NLP	Natural Language Processing
NLTK	Natural Language Toolkit

1.0 Introduction

1.1 Background

User interfaces (UI) have changed over time from fixed designs that were the same for everyone to more flexible and personalized experiences. Initially, UIs were developed with a broad audience in mind, but as user needs became more complex and diverse, the limitations of static interfaces became apparent. Personalization emerged as a critical solution, allowing interfaces to adapt to individual user preferences, behaviors, and contexts.

In recent years, adding Artificial Intelligence (AI) to UI design has greatly improved how personalized user experiences can be. AI-driven systems leverage vast amounts of user data to predict preferences and adjust interfaces, accordingly, leading to more engaging and satisfying user experiences [1]. This paradigm shift is evident in various applications, from web interfaces to mobile applications, where the ability to tailor experiences in real-time has become a competitive advantage. A key advancement in this area is the use of AI models that learn from user feedback. These models are trained on datasets that incorporate user feedback, enabling continuous learning and improvement. By dynamically adjusting the UI in response to real-time feedback, these systems ensure that the interface remains relevant and user-friendly as needs evolve [2]. The capability of AI to adapt in real-time is crucial in today’s digital environments, where user preferences can shift rapidly and unpredictably.

Applying real-time feedback in UI personalization involves goes beyond adjusting the appearance. It encompasses a broader approach, including the analysis of user behavior, processing both quantitative and qualitative feedback, and dynamically adapting the interface to enhance usability and accessibility [3]. Creating a continuous feedback loop is crucial for these systems because it helps the interface constantly improve and adapt to the user's changing needs.

References to the integration of AI in UI personalization have highlighted the benefits of such systems, including increased user engagement, satisfaction, and overall usability. As the technology continues to evolve, the potential for more sophisticated and intuitive user interfaces grows, making real-time AI-driven personalization a key area of research and development [4].

1.2 Literature Review

The literature on AI-driven UI personalization is extensive, reflecting the growing interest in this area of research. One of the foundational studies in this field examines the use of machine learning techniques for adaptive user interfaces, highlighting the potential of AI to create more responsive and personalized experiences [1]. This study underscores the importance of incorporating user feedback into the design process, as it allows the AI to learn and improve continuously.

Further research has explored the integration of real-time feedback mechanisms into UI personalization systems. For instance, a study on real-time user interface adaptation emphasizes the role of continuous feedback in enhancing the user experience [2]. The authors argue that by processing user feedback in real-time, the system can make immediate adjustments to the interface, resulting in a more fluid and responsive interaction.

Another significant contribution to the literature is the examination of feedback-driven AI systems for UI personalization. This research focuses on the importance of establishing feedback loops that enable the system to learn from user interactions and make iterative improvements [3]. The study provides insights into how these feedback loops can be designed and implemented to ensure that the UI remains aligned with user needs over time.

The integration of both quantitative and qualitative feedback in UI personalization has also been a topic of considerable interest. Research in this area highlights the value of combining different types of feedback to create a more holistic understanding of user needs [5]. By leveraging both quantitative data, such as user engagement metrics, and qualitative insights, such as user comments and surveys, AI-driven systems can deliver more nuanced and effective personalization.

Finally, a foundational paper on the automatic personalization of user interfaces using machine learning provides a comprehensive overview of the challenges and opportunities in this field [4]. The authors discuss various approaches to UI personalization and emphasize the importance of continuous learning and adaptation in creating interfaces that can evolve alongside the user.

1.3 Research Gap

The increasing integration of Artificial Intelligence in user interface (UI) personalization has led to significant advancements in enhancing user experience. However, there are still critical gaps in the current research that need to be addressed to fully leverage the potential of AI-driven user interface personalization, particularly in the context of real-time feedback mechanisms.

1. **Lack of Comprehensive Real-Time Feedback Integration**
   - Most studies focus on static or semi-static feedback integration where the UI is adjusted based on predefined user preferences or periodic user data collection. The lack of a real-time feedback loop means that these systems may not effectively adapt to the rapid changes in user behavior and preferences [1][2].
   - **Identified Gap:** There is a need for a more dynamic approach where user feedback is continuously collected and processed in real-time, allowing for instantaneous adjustments to the UI.
   - **Proposed Solution:** This research proposes the development of a system that continuously learns from user interactions and feedback, dynamically adapting the UI in real-time to enhance usability and user satisfaction.

2. **Limited Use of AI for Dynamic and Personalized UI Adjustments**
   - While some research has explored AI-driven personalization, the application is often limited to basic customization, such as theme changes or layout adjustments, without considering deeper contextual factors or user behavior analytics [3][4].
   - **Identified Gap:** The gap lies in the lack of AI models that can deeply analyze user feedback to offer a more contextually aware and personalized UI.
   - **Proposed Solution:** The proposed research will develop an AI model capable of not only analyzing user feedback but also incorporating contextual data to provide a truly personalized and adaptive user interface.

3. **Insufficient Attention to Continuous Learning and Adaptation**
   - Current systems that do implement AI-driven personalization often lack mechanisms for continuous learning, leading to interfaces that become outdated as user preferences evolve [2][4].

- **Identified Gap:** There is a significant need for systems that continuously adapt to users' changing needs, ensuring the interface remains relevant and efficient over time.
- **Proposed Solution:** The research will focus on establishing a continuous learning framework within the AI system, ensuring that the UI evolves in tandem with the user's behavior and feedback, thus maintaining high levels of engagement and satisfaction.

4. **Underutilization of Quantitative and Qualitative Feedback**
   - Most research has either focused on quantitative feedback, such as clicks or time spent on tasks, or qualitative feedback like user comments, but rarely integrates both for a more holistic personalization approach [1][3].
   - **Identified Gap:** The gap exists in the integration of both quantitative and qualitative feedback to create a more comprehensive and adaptive user interface.
   - **Proposed Solution:** This research aims to bridge this gap by integrating both types of feedback into the AI model, allowing for a more robust and responsive UI personalization.

Table 1: Research Gap Among Existing Software User Interface Personalization Systems

Identified Research Gaps	**Current Approaches in Literature**	**Proposed Research Solutions**
Real-time feedback integration.	Limited to periodic data collection or static feedback mechanisms [1][2].	Development of a real-time feedback loop for continuous UI adaption.
Dynamic and personalized UI adjustments using AI.	Basic customization without deep contextual analysis [3][4].	AI model that analyzes both contextual and behavioral data for deeper personalization.
Continuous learning and UI adaption.	Lack of continuous learning mechanisms leading to outdated interfaces [2][4].	Implementing a continuous learning framework to keep the UI relevant and adaptive over time.
Integration of quantitative and qualitative feedback.	Focus on either quantitative or qualitative feedback, rarely both [1][3].	Developing a holistic model that incorporates both types of feedback for a comprehensive UI personalization.

1.4 Research Problem

The complexity of user needs in digital environments has led to limitations in current UI designs, with traditional static methods struggling to adapt to evolving user preferences and behaviors. This study explores real-time user feedback integration in AI systems for dynamic UI personalization, aiming to develop a system for immediate adjustments based on user behavior or preferences. The research explores methods to ensure that AI-driven UI personalization remains relevant and adaptive over time. Many existing AI models lack mechanisms for continuous learning, resulting in personalization that may become outdated as user needs evolve. This study seeks to establish a framework that allows the AI to evolve alongside the user, ensuring that the interface remains engaging and effective.

Furthermore, the study examines how both quantitative and qualitative user feedback can be integrated into AI models to enhance UI personalization. Current approaches often focus on either quantitative data, such as clicks and time spent, or qualitative feedback, such as user comments, in isolation. By combining these two types of feedback, the research aims to create a more comprehensive and responsive UI personalization system. Finally, the impact of real-time AI-driven personalization on user satisfaction and engagement is evaluated. While the effectiveness of AI-driven personalization is often measured through static metrics, this study will assess the dynamic adjustments' effects on user satisfaction and engagement, providing insights into the benefits and limitations of this approach.

By addressing these research problems, the study aims to fill existing gaps in the literature and contribute to the development of more effective and user-centered AI-driven UI personalization systems.

2.0 Objectives

2.1 Main Objective

Real-time Feedback Integration for UI Personalization.

To develop a system that integrates real-time user feedback into AI-driven processes for dynamic user interface (UI) personalization. This system will enable continuous adaptation of the UI based on the user feedback, thereby enhancing user satisfaction and engagement.

2.2 Specific Objectives

The primary objective of this functionality is to develop an AI-driven system that personalizes user interfaces (UI) in real-time based on continuous user feedback. By dynamically adapting the UI to meet user preferences, the system aims to enhance overall user satisfaction and engagement.

SMART Objective:

- **S - Specific:** Develop an adaptive UI system that personalizes interfaces in real-time based on user feedback, including mechanisms for feedback collection, processing, dynamic UI adjustments, a continuous feedback loop, and personalized suggestions.
- **M - Measurable:** Success will be gauged by metrics such as feedback processing speed, accuracy of UI adjustments, user satisfaction scores, and the relevance of personalized UI suggestions.
- **A - Achievable:** Achieved through the integration of existing technologies and frameworks like React.js, Python, and TensorFlow, ensuring feasibility within the project's scope.
- **R - Realistic:** This goal is realistic, leveraging the team's expertise and existing tools, making it attainable within the given resources and timeframe.
- **T - Time-bound:** The objective will be met through a phased approach, with the system expected to be fully operational by the project's end, adhering to a clear timeline for each component.

3.0 Methodology

3.1 System Overview

The proposed AI-based plugin for web applications is designed to enhance user experiences by personalizing software interfaces and interactions in real time. This system primarily focuses on normal users and visually impaired users. Normal Users are focused for offering dynamic interface customization driven by continuous user feedback in real-time.

The system uses two primary data sources,

1. **Existing Dataset** *(Web Satisfaction Survey in Kaggle)*- An initial AI model is trained using a pre-existing dataset containing user interaction data, serving as the foundation for understanding user behaviors.
2. **Real-Time User Feedback** - Feedback is continuously collected via in-app surveys and pop-up forms, allowing the system to adapt interfaces based on evolving user preferences.

Collected feedback is processed using NLP tools like NLTK and spaCy to analyze user sentiments. TensorFlow classifies feedback, enabling the AI model to make real-time UI adjustments. Apache Kafka supports a continuous feedback loop, ensuring ongoing refinement of the user interface. Key technologies include React.js for front-end development, Python for back-end processing, and RESTful APIs to enable communication between system components. MongoDB is used for data storage and management, while NLTK, spaCy, and TensorFlow handle natural language processing and feedback analysis. Apache Kafka facilitates real-time data integration through a continuous feedback loop, and the development environment is supported by Visual Studio Code, with Git and GitHub used for version control and collaborative development.

This system ensures a personalized and adaptive user interface, enhancing user satisfaction and engagement.

3.2 System Architecture Diagram

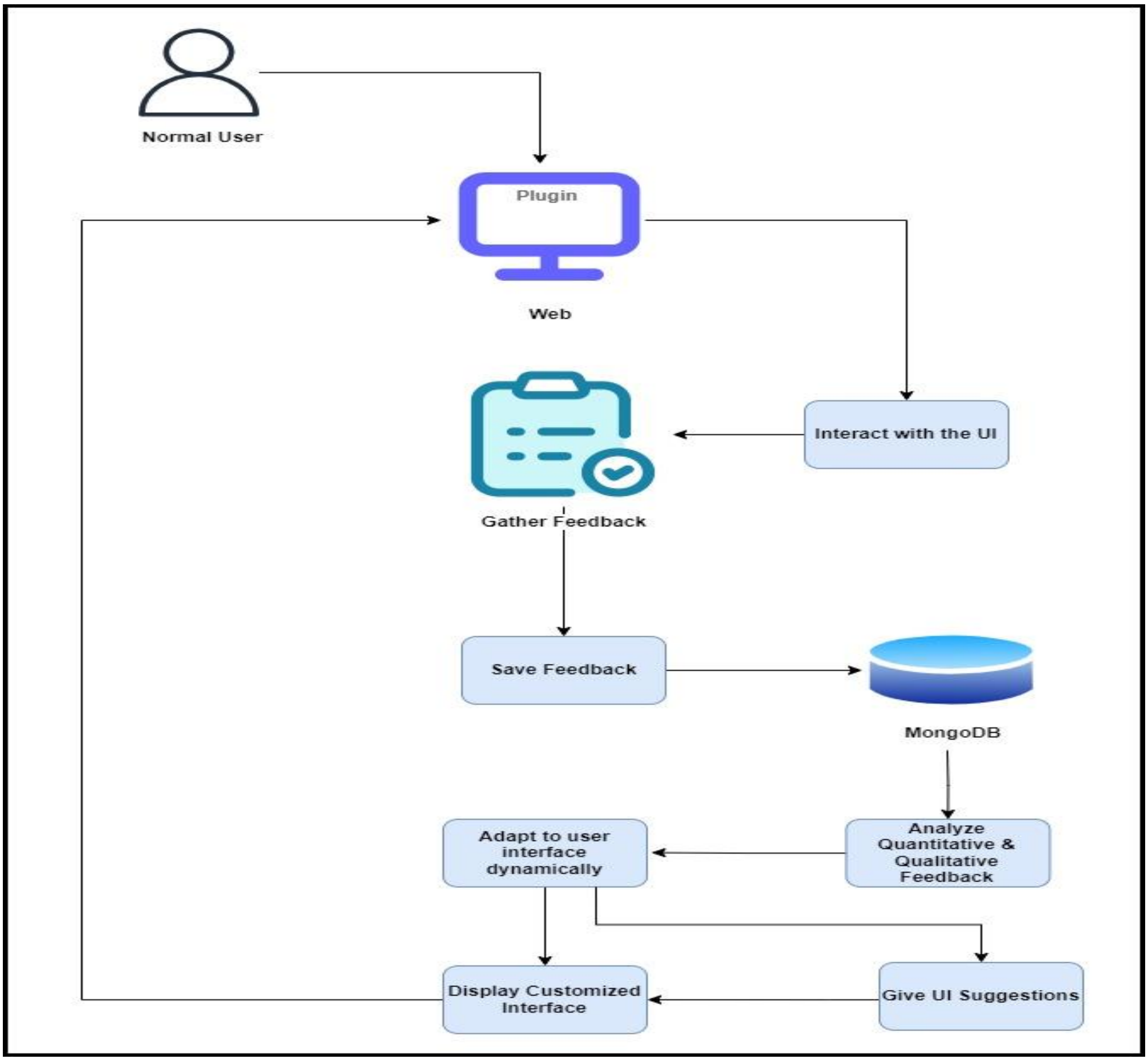

Normal User
Plugin
Web
Interact with the UI
Gather Feedback
Save Feedback
MongoDB
Analyze Quantitative & Qualitative Feedback
Adapt to user interface dynamically
Give UI Suggestions
Display Customized Interface

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

The system depicted in the diagram illustrates the high-level architecture of real-time software UI personalization through user feedback. Initially, the AI model is trained using an existing dataset to establish a foundational understanding of user feedback. As users interact with the application, real-time feedback is continuously collected via in-app surveys and pop-up forms. This feedback is processed and analyzed using natural language processing (NLP) tools, such as NLTK and spaCy, and classified through TensorFlow. The system then dynamically adapts the user interface based on the insights gained, ensuring a personalized experience.

3.3 Identifying Existing Systems

Several existing systems have explored the integration of user feedback for personalized user interface (UI) design, focusing on enhancing user satisfaction and engagement. These systems typically rely on pre-defined rules or static models that are updated periodically based on collected user feedback. However, they often lack the ability to adapt the UI in real-time, leading to delays in addressing user preferences. The existing approaches also primarily focus on either visual aspect of the UI or general usability, without a comprehensive strategy to dynamically adjust the interface based on continuous and real-time user feedback.

Another common approach in current systems involves using machine learning models trained on historical user data to predict and implement UI changes. While effective to some extent, these models are often limited by their dependence on batch processing of feedback, which can result in outdated UI configurations. Additionally, the user feedback mechanisms in these systems are generally simplistic, relying on basic surveys or ratings, and may not fully capture the complexity of user preferences and feedback. This gap highlights the need for more advanced systems capable of real-time adaptation, continuous learning, and sophisticated feedback processing, which are essential for creating truly personalized user experiences.

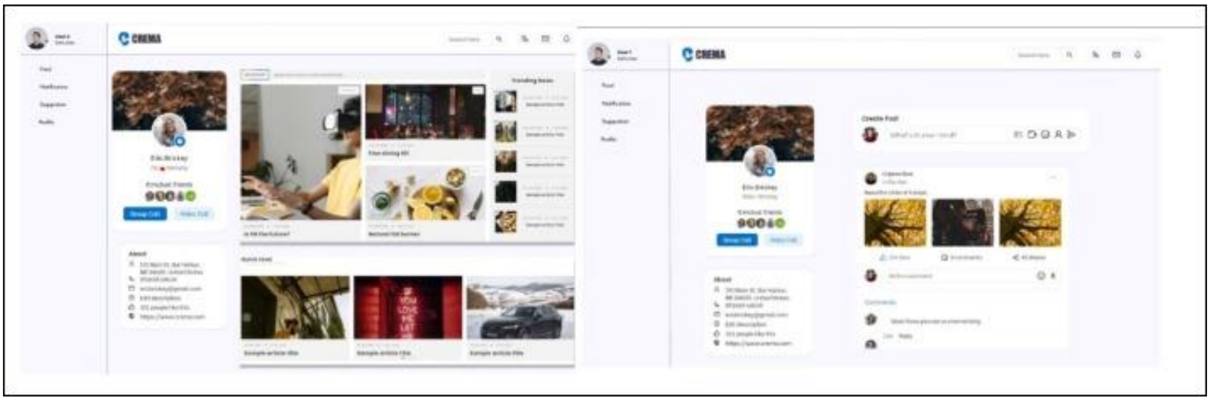


Figure 2: Software UI Personalization on an Existing System

3.4 Project Requirements

3.4.1 System Requirements

The Software User Interface Personalization through User Feedback system requires tools and infrastructure for real-time data collection, processing, and response, including robust data collection mechanisms, hardware, software, data, and network resources, seamless integration with existing web applications, and dynamic user interface adaptation.

3.4.1.1 Functional Requirements

1. **User Profile Management**

- **Requirement:** The system must include a robust user profile management feature that stores and manages individual user preferences and UI settings based on their feedback.
- **Sub-Tasks:**
  - **Profile Data Storage:** Implement a secure method for storing user-specific UI preferences and feedback history in a database (e.g., MongoDB).
  - **Session Management:** Ensure that when users log back into the system, their previous session's UI settings are retrieved and applied.
  - **User Authentication:** Implement authentication mechanisms that securely link user profiles with their personalized settings, ensuring privacy and data integrity.

2. **Collect Real-Time User Feedback**

- **Requirement:** The system must collect feedback from users during their interactions in real-time.
- **Sub-Tasks:**
  - **Design In-App Surveys:** Develop and implement surveys that appear contextually within the application.
  - **Implement Pop-Up Forms:** Trigger pop-up forms based on specific user actions to capture immediate feedback.

3. **Analyze User Behavior**

- **Requirement:** The system must analyze the collected feedback to understand user behavior and preferences.
- **Sub-Tasks:**
  - **Data Parsing:** Use NLP tools like NLTK and spaCy to parse textual feedback.

4. **Process and Analyze Feedback**

- **Requirement:** The system must classify and process feedback using machine learning algorithms.
- **Sub-Tasks:**
    - **Feedback Classification:** Employ TensorFlow to categorize feedback into actionable insights.
    - **Model Training:** Continuously train the model with new feedback to improve accuracy.
    - **Data Filtering:** Filter out noise and irrelevant data to focus on significant feedback.

5. **Adapt User Interface Dynamically**

- **Requirement:** The system must dynamically adjust the user interface in real-time based on feedback.
- **Sub-Tasks:**
    - **UI Adjustment Algorithms:** Develop algorithms that determine the best UI changes based on feedback.
    - **Real-Time Implementation:** Ensure that UI adjustments are made instantly as feedback is processed.

6. **Establish a Feedback Loop**

- **Requirement:** A continuous feedback loop must be established to ensure ongoing system adaptation.
- **Sub-Tasks:**
    - **Integrate Apache Kafka:** Use Kafka to manage real-time data streams and maintain a consistent feedback loop.

7. **Provide UI Suggestions Based on Feedback**

- **Requirement:** The system must generate personalized UI suggestions based on analyzed feedback.
- **Sub-Tasks:**
    - **Suggestion Algorithm Development:** Create algorithms that generate UI recommendations tailored to individual users.
    - **User Interface Presentation:** Implement a method for presenting these suggestions to users in a clear and intuitive manner.

3.4.1.2 Non-Functional Requirements

Performance

- The system must process and apply user feedback to the UI within milliseconds to ensure a seamless, real-time experience.

Scalability

- The system must be scalable to handle an increasing number of users and interactions without compromising performance or user experience.

Reliability

- The system must be highly reliable, ensuring that user feedback is consistently captured and applied without failure.

Usability

- The system must provide an intuitive and user-friendly interface for both collecting feedback and presenting personalized UI changes.

Security

- The system must ensure the security of user data, particularly personal preferences and feedback.

3.4.1.3 User Requirements

- As a User, I want to provide feedback easily so that the system can understand my preferences.
- As a User, I want the system to remember my preferences from previous sessions so that I do not have to reconfigure the interface each time I log in.
- As a User, I want the user interface to adapt dynamically based on my feedback so that it becomes more aligned with my needs and preferences.
- As a User, I want to see personalized UI suggestions based on my past behavior and feedback so that I can choose the options that best suit my workflow.
- As a User, I want the system to process my feedback quickly so that changes to the interface are applied immediately.
- As a User, I want to be notified of any major UI changes resulting from my feedback so that I’m aware of how my input is being used.

3.4.2 Expected Test Cases

Table 2: Expected Test Cases for Software UI Personalization through User Feedback

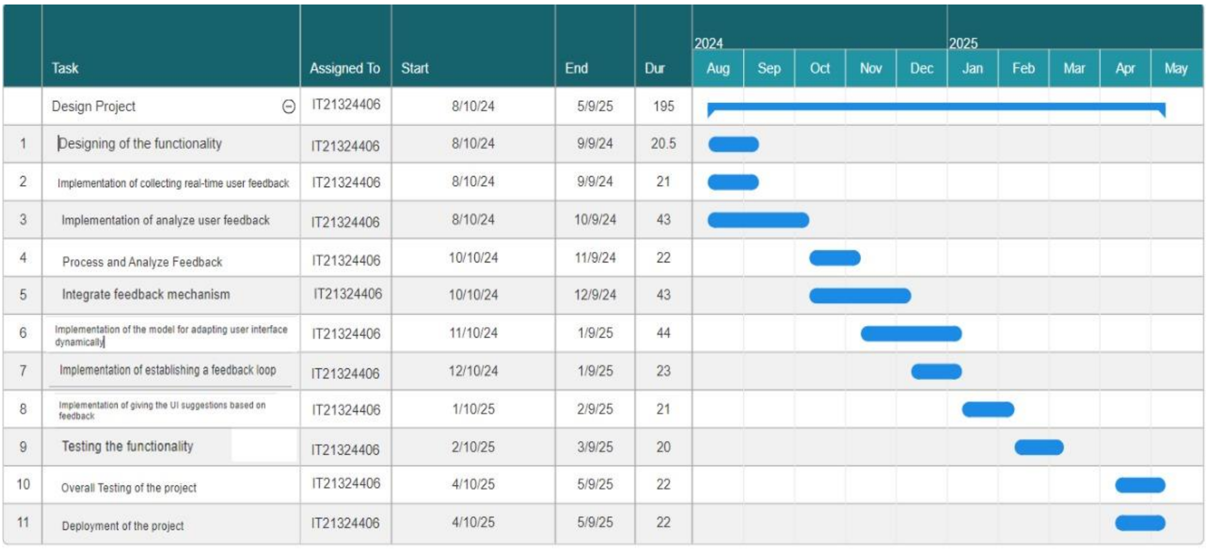
Test ID	Test Description	Test Input	Test Output
01	Verify that user feedback is correctly captured	User submits feedback through a pop-up form	Feedback is stored in the database and notify the user
02	Check if the system dynamically adapts the UI based on feedback	User requests a change in UI layout	UI layout is updated in real-time to reflect user preferences
03	Validate that user preferences are saved across sessions	User logs in, provides feedback, logs out, and logs in again	The UI reflects the preferences from the previous session
04	Ensure the system provides personalized UI suggestions	User interacts with the software multiple times	Personalized UI options are suggested based on the user's past behavior
05	Test feedback processing and analysis accuracy	User submits feedback with specific preferences	Feedback is correctly analyzed, and the UI is adjusted accordingly
06	Check system performance under high feedback load	Multiple users submit feedback simultaneously	System processes feedback without delay and updates UI in real-time
07	Validate that the system notifies users of major UI changes	User feedback results in significant UI modification	User receives a notification detailing the changes made to the UI

3.5 Testing

The testing process will encompass multiple stages to ensure the reliability and functionality of the Software User Interface Personalization through User Feedback in real-time. Initially, unit testing will be conducted to verify that individual components, such as feedback collection and UI adaptation modules, function correctly in isolation. Following this, integration testing will ensure that these components work together seamlessly, with a focus on the smooth flow of data from user feedback to real-time UI updates. Finally, system testing will validate the overall performance and usability of the system under real-world conditions, confirming that it meets all user requirements and operates efficiently across different scenarios.

3.6 Timeline

The timeline for implementing the Software User Interface Personalization through User Feedback functionality is structured to ensure a phased and systematic approach. Initially, the project will begin with requirements gathering and dataset preparation, which will take approximately two weeks. This will be followed by the development phase, where core components such as feedback collection, processing algorithms, and UI adaptation mechanisms will be implemented over the next months. After development, testing, including unit testing, integration testing, and system testing will be performed. The final phase will focus on deploying the system and refining the UI based on real-world feedback. In total, the project is expected to be completed within a ten-month period.



	Task	Assigned To	Start	End	Dur
	Design Project	IT21324406	8/10/24	5/9/25	195
1	Designing of the functionality	IT21324406	8/10/24	9/9/24	20.5
2	Implementation of collecting real-time user feedback	IT21324406	8/10/24	9/9/24	21
3	Implementation of analyze user feedback	IT21324406	8/10/24	10/9/24	43
4	Process and Analyze Feedback	IT21324406	10/10/24	11/9/24	22
5	Integrate feedback mechanism	IT21324406	10/10/24	12/9/24	43
6	Implementation of the model for adapting user interface dynamically	IT21324406	11/10/24	1/9/25	44
7	Implementation of establishing a feedback loop	IT21324406	12/10/24	1/9/25	23
8	Implementation of giving the UI suggestions based on feedback	IT21324406	1/10/25	2/9/25	21
9	Testing the functionality	IT21324406	2/10/25	3/9/25	20
10	Overall Testing of the project	IT21324406	4/10/25	5/9/25	22
11	Deployment of the project	IT21324406	4/10/25	5/9/25	22

Figure 3: Gantt Chart for the system

3.7 Communication Management Plan

The Communication Management Plan outlines the strategies and channels that will be used to ensure effective and timely communication throughout the development of the Software User Interface Personalization through User Feedback functionality. Clear communication protocols will be established to facilitate the exchange of feedback, address issues promptly, and ensure that all team members are aligned with the project objectives and timelines.

3.7.1 Communication Objectives

The Communication Objective aims to maintain clear, concise, and timely communication with all stakeholders about the Software User Interface Personalization through User Feedback functionality development project, facilitating collaboration, minimizing misunderstandings, and meeting all expectations.

3.7.2 Communication Media

Effective communication is vital for the successful development of the Software User Interface Personalization through User Feedback functionality. Various communication media will be used to ensure clear, consistent, and timely information exchange among team members, stakeholders, and other relevant parties. The chosen communication tools will facilitate collaboration, documentation, and feedback collection throughout the project lifecycle.

Table 3: 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,	Scheduled virtual or

			Project Team	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.

4.0 Commercialization

The commercialization of this AI-based system for real-time software user interface personalization through user feedback and user behavior, will be approached in a way that aligns with the resources and capabilities of a university research project. The initial focus will be on developing a functional prototype that demonstrates the system's core features, including real-time adaptation of user interfaces based on feedback, user behavior and improve the browsing experience of the visually impaired users.

Commercialization efforts will involve sharing the system with academic and industry communities through conferences, publications, and tech exhibitions. This exposure can help attract interest and feedback, which can be instrumental in refining the system and exploring potential market applications. Given the scope of the project, the initial commercialization strategy will focus on licensing the technology to software companies or integrating the system into existing platforms as a plugin. This approach allows for gradual scaling while generating revenue to support ongoing development and research. Collaborations with industry partners, or academic institutions could also be explored to gain additional support, resources, and expertise needed to bring the system to market.

5.0 Budget

The budget section will outline the financial resources required for the project, including expenses for research, development, testing, and commercialization. This will cover costs for software, hardware, licenses, team salaries, and consultant fees. The aim is to provide a clear understanding of the financial needs to support the project's successful execution.

6.0 Summary

This proposal outlines the development of a key component of an AI-based system (plugin) designed to personalize software user interfaces in real-time, based on user feedback. The system focuses on enhancing user experiences, particularly for normal users by personalizing the UI based on the user feedback including the dynamic adaptation of UIs through machine learning models trained on user data. The project leverages advanced technologies such as React.js, Python, MongoDB, and TensorFlow. The commercialization plan involves creating a functional prototype, with potential for collaboration and licensing, ensuring the system's transition from research to real-world application.

References

[1] A. Khamaj and A. M. Ali, "Adapting user experience with reinforcement learning: Personalizing interfaces based on user behavior analysis in real-time," *Alexandria Engineering Journal,* vol. 95, no. 1, pp. 164-173, 2024.

[2] A. Costa, F. Silva and J. J. Moreira, "Towards an AI-Driven User Interface Design for Web Applications. Procedia Computer Science," *Procedia Computer Science,* vol. 237, no. 1, pp. 179-186, 2024.

[3] A. Namoun, A. Alrehaili, Z. U. Nisa, H. Almoamari and A. Tufail, "Predicting the usability of mobile applications using AI tools: the rise of large user interface models, opportunities, and challenges," *arXiv preprint arXiv,* vol. 213, no. 1, pp. 1312-1327, 2024.

[4] W. D, C. Anderson, P. Domingos, O. Etzioni, K. Z. Gajos, T. Lau and S. Wolfman, "Automatically personalizing user interfaces," no. 1, 2003.

[5] J. Hussain, A. U. Hassan, H. S. Muhammad Bilal, R. Ali, M. Afzal, S. Hussain and S. Lee, "Model-based adaptive user interface based on context and user experience evaluation," *Journal on Multimodal User Interfaces,* vol. 12, no. 1, pp. 1-16, 2018.