APPLIED INFORMATION TECHNOLOGY AND COMPUTER SCIENCE
e-ISSN: 2773-5141

AITCS

Vol. 5 No. 2 (2024) 538-557
https://publisher.uthm.edu.my/periodicals/index.php/aitcs

The Development of LearnPro: A Mobile App That Makes Learning More Effective and Enjoyable

Ching Kai Yuan[1], Rozanawati Darman[2*]

[1,2] *Fakulti Sains Komputer dan Teknologi Maklumat,*
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author: zana@uthm.edu.my
DOI: https://doi.org/10.30880/aitcs.2024.05.02.029

Article Info

Received: 13 June 2024
Accepted: 28 September 2024
Available online: 15 December 2024

Keywords

LearnPro, Mobile Application Development, Flutter Framework, Firebase Technologies, Effective Study App, Personalized Learning Model, Gamified Learning, Pomodoro

Abstract

LearnPro revolutionizes personalized learning with an object-oriented approach within the Flutter framework, ensuring a seamless educational experience. Using Firebase technologies like Firestore for database management and Firebase Authentication for secure access, LearnPro guarantees both functionality and data integrity. It empowers users with user-generated study plans and the Pomodoro timer technique to optimize productivity and focus. The user-defined Roulette Reward Mechanism adds gamification, enhancing motivation. The Analytics module, which graphs focus time, provides insights into study habits, enabling users to improve their strategies. The Notification and Reminder module keeps users organized with timely notifications for upcoming tasks and study sessions. By integrating these features, LearnPro creates a user-friendly and adaptive learning environment, empowering self-directed learners and students to take control of their educational journeys and achieve their goals.

1. Introduction

The LearnPro App project emerges as a response to the pressing need for personalized learning solutions in today's technology-driven landscape. Traditional educational systems often fall short in accommodating the diverse learning styles and preferences of individual students, prompting a demand for more flexible and user-centric tools. LearnPro steps in to fill this gap by offering a comprehensive mobile application tailored to provide personalized learning experiences. Leveraging the robust Flutter framework and Firebase technologies, LearnPro ensures a seamless user experience while prioritizing data security and scalability. With its core focus on empowering users to take control of their learning journey, LearnPro allows for the creation of customized study plans and effective session management through features like the Pomodoro timer.

Moreover, LearnPro introduces innovative elements such as the Roulette Reward Mechanism for motivation and the Focus Time Analytics module for progress tracking. By embracing a user-centric approach, LearnPro aims to address the challenges of traditional educational systems, providing learners with the tools they need to succeed. Through its objectives of design, development, and rigorous testing, LearnPro strives to not only improve learning efficiency but also foster sustained motivation and provide valuable insights for educational research. Positioned as a user-friendly and adaptive solution, LearnPro has the potential to revolutionize education by offering personalized support, real-time progress monitoring, and effective time management tools tailored to the needs of modern learners.

2. Related Work

This chapter delves into the integration of key educational concepts within the LearnPro application to amplify the learning experience. The adoption of personalized learning, reward learning, and the Pomodoro Technique within LearnPro contributes to a more dynamic and effective learning environment tailored to individual needs and preferences.

2.1 Personalized Learning Model

Personalized learning is a transformative approach in education that recognizes the unique needs and preferences of individual learners. Traditional educational models often follow a one-size-fits-all approach, which can overlook the diverse learning styles and paces of students[1]. At LearnPro, we use this logic to create a system that embraces personalized learning principles, allowing users to tailor their learning experiences according to their specific goals and schedules.

Through features like personalized study plans and adaptive learning pathways, LearnPro empowers learners to take control of their educational journey, fostering greater engagement and motivation. Studies have shown that personalized learning strategies significantly enhance student outcomes by promoting deeper engagement and understanding[2]. LearnPro leverages these strategies by integrating user-centric features such as the Pomodoro timer, which helps learners manage their study sessions effectively. By incorporating technology into the learning process, LearnPro creates a dynamic and adaptive environment that caters to the diverse needs of modern learners[3]. Additionally, the app's Focus Time Analytics module provides users with valuable insights into their study patterns, enabling them to make informed decisions and optimize their learning strategies[4].

2.2 Reward Learning

Reward learning is a fundamental aspect of human behavior that influences motivation, decision-making, and learning outcomes. Within the LearnPro mobile app, reward learning plays a central role in encouraging and reinforcing positive learning behaviors. By offering rewards, acknowledgments, and achievements for completing learning tasks or reaching educational milestones, LearnPro leverages reward learning principles to motivate users and enhance their learning experience. This approach taps into the anticipation of positive outcomes, providing users with a sense of accomplishment and progress[5].

The incorporation of reward learning principles within LearnPro not only incentivizes users to engage with the app but also fosters a supportive and encouraging learning environment. Through gamified elements like the Roulette Reward Mechanism, users are encouraged to set their own goals and rewards, making the learning process more enjoyable and meaningful[6]. By aligning with human motivational mechanisms, LearnPro maximizes user engagement and promotes a sustainable approach to self-directed learning.

2.3 Pomodoro Technique

The Pomodoro Technique, a time management method emphasizing focused work and regular short breaks, involves breaking work into intervals, typically 25 minutes, called "Pomodoro," followed by a 5-minute break. After completing a set of Pomodoro, a longer break of 15-30 minutes is taken[7]. Key features include setting a specific work duration, regular breaks for recharging, and using a timer for effective interval management.

The Pomodoro Technique Timer module in the LearnPro App aligns with these principles, aiding users in effective time management, maintaining focus during learning sessions, and preventing burnout. Offering a structured approach to work and study intervals, it contributes to improved time management and concentration for self-learners. This feature supports the LearnPro App's objective of enhancing efficiency and motivation in pursuit of individual learning objectives.

3. Methodology

According to Yakovyna et al.[8], the methodology of software development is a system of principles, as well as a set of ideas, concepts, approaches, methods and tools that determine the style of software development.

3.1 Object-Oriented Analysis and Design (OOAD)

The mobile application development undertaken in this project followed the OOAD methodology[9].OOAD serves as a robust framework for scrutinizing and architecting software systems based on object models. Objects, within this context, are construed as software representations of real-world entities, encapsulating properties, behaviors, and states. Consistent with the definition of objects in Object-Oriented Programming (OOP) languages, the OOAD methodology leverages Unified Modeling Language (UML) for diagrams such as use case diagrams, class diagrams, sequence diagrams, and activity diagrams. This methodology significantly enhances software development efficiency by advocating the use of reusable components and design patterns, thereby reducing development time

and effort. Figure 1. visually outlines the phases of the OOAD methodology in the project, encompassing object-oriented analysis, design, implementation, and testing phases[10]. The subsequent sections delve into the specific activities conducted at each stage of the OOAD methodology and expound upon the techniques or tools employed in the process.

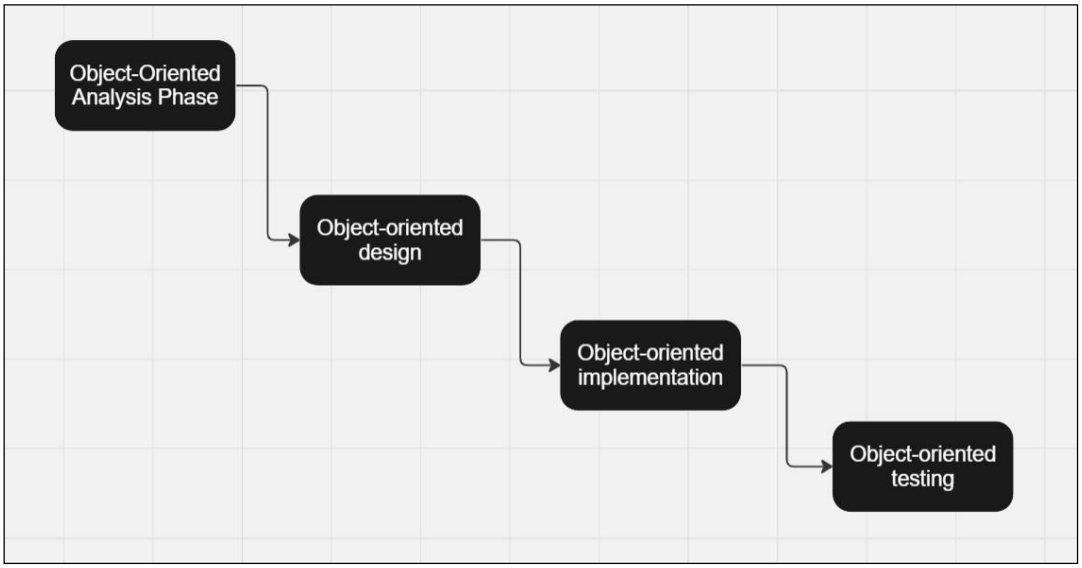


Fig 1. OOAD Methodology Phase[10]

3.1.1 Object-Oriented Analysis Phase

The object-oriented analysis phase marks the project's inception, centering on the identification of system requirements and proposing viable solutions. Commencing with a brainstorming session to determine the project title, a meticulously prepared draft project proposal, inclusive of project introduction, objectives, problem statements, and scope, underwent discussions with the supervisor to ensure project title acceptance. The primary focus during this phase was on analyzing system requirements related to the proposed mobile application, elucidating challenges encountered during self-learning, and determining functional and non-functional requirements.

3.1.2 Object-Oriented Design Phase

The object-oriented design phase transformed the collected information and requirements into a well-structured system design. UML diagrams, including use case diagrams, sequence diagrams, activity diagrams, and class diagrams, were created using the tools of Lucidchart. The mobile application database and data dictionary were sculpted through class diagram via Lucidchart, and the user interface was conceptualized using Figma to craft a wireframe and prototype.

3.1.3 Object-Oriented Implementation Phase

The object-oriented implementation phase involved the translation of the design into executable code, employing Android Studio and Flutter as the coding editor platform. This pivotal phase witnessed the development of mobile application interfaces and modules, with system integration implemented to birth a fully functional application. Firebase played a crucial role in developing and configuring the mobile application database, integrated based on the data dictionary.

3.1.4 Object-Oriented Testing Phase

The object-oriented testing phase played a pivotal role in validating the LearnPro mobile application's functionality. Through a meticulously crafted test plan, the team conducted a thorough assessment of features and performance, aiming to address errors and bugs for a seamless user experience. User Acceptance Testing (UAT), with insights from students and self-learners, was crucial for refining the application. The testing process scrutinized key functionalities like data storage, updates, and access, promptly correcting identified errors. Interface testing ensured user-friendliness, supporting effective learning. As a vital validation step, UAT engaged end users, enabling organizers and volunteers to actively participate in approving the software system before its transition to the production environment [11]. This meticulous approach guarantees alignment with user requirements, ensuring a smooth deployment for widespread use.

3.2 System Development Workflow

The system development workflow provides an organized framework for ensuring the efficient delivery of high-quality systems while also serving as a roadmap for the entire process. The OOAD methodology consists of four distinct phases, each of which has particular tasks and outputs listed in Table 1.

Table 1 *Mobile Application Development Activities and their Task*

No.	Phase	Task	Output
1	Object-oriented Analysis	1. Generate ideas for the project title 2. Prepare a draft project proposal 3. Discuss with the supervisor 4. Prepare the final project proposal 5. Determine the requirements 6. Study the existing system 7. Compare the existing system and proposed system	• Project Title • Project proposal • Project literature review • Comparison table between existing system and proposed system
2	Object-oriented design	1. Design UML diagrams (Class Diagrams, Use Case Diagrams) 2. Design User Interface wireframes 3. Create a prototype 4. Design the database schema 5. Design the UAT form	• UML Diagrams • Wireframes • Prototype • Database Schema Draft • UAT Form
3	Object-oriented implementation	1. Develop system interfaces (UI/UX development) 2. Develop system modules (backend and frontend development) 3. Integrate system components 4. Connect the system with the database	• Functional LearnPro mobile app • Connected and Functional Database
4	Object-oriented testing	1. Conduct system functional testing 2. Distribute UAT form to users 3. Debug and fix errors identified during testing	• Test Plan • Collected UAT Forms with User Feedback • Error-free LearnPro mobile app

3.3 Project Planning

To ensure the timely completion of the project, a project plan was formulated. This plan was visualized through a Gantt chart, created using Microsoft Project. The Gantt chart represents tasks or activities on a timeline, illustrating their start and end dates and indicating the duration required for each task within OOAD methodology phases. Refer to Appendix A for the detailed Gantt chart outlining the project plan.

4. Analysis and Design

This chapter will discuss the analysis and design of the project. The analysis and design will cover system requirement analysis and Design. The system requirement analysis contains Unified Modelling Language (UML) diagrams. The design contains interface design and database design.

4.1 System Requirement Analysis

System Requirement Analysis is a crucial phase in LearnPro App development, covering functional and non-functional requirements, user expectations, UML diagrams, requirement traceability, and entity relationship diagrams. This comprehensive analysis ensures a clear understanding of the app's architecture, behavior, and alignment with user needs, facilitating a seamless development process and the creation of a robust, user-centric learning platform.

4.1.1 Functional and Non-Functional Requirements

In the development of software systems, understanding and delineating the requirements is a critical phase that sets the foundation for the entire project. These requirements can be broadly categorized into two main types

which is Functional and Non-Functional. Functional requirement refers to tasks and implementations of the software system [12]. Table 2 represents the functional requirements of the LearnPro App.

Table 2 *Functional requirements for LearnPro App*

No.	Function	Functionality
1	User Authentication	The application authenticates users to access the system through login and registration.
2	Forget Password	The application provides users with a way to reset their forgotten passwords.
3	Dashboard	The dashboard gives users a quick summary of important information, making it easy to understand and use.
4	Profile	The application displays user information and settings in a dedicated section.
5	Goals Management	The application allows users to perform CRUD operations for goals.
6	Tasks Management	The application allows users to perform CRUD operations for tasks.
7	Pomodoro Timer	The application enables users to use the Pomodoro Timer for focusing on tasks.
8	Rewards Management	The application allows users to perform CRUD operations for rewards.
9	Spin Wheel	The application allows users to spin a roulette to get rewards.
10	Notification and Reminder	The application provides users with the capability to manage notifications and reminders to enhance their productivity and engagement.
11	View Report	The application allows users to view a summary of completed tasks and focus time over a week.
12	Tutorial	The Tutorial feature aims to onboard users smoothly into the application, providing them with a guided introduction to its key features and functionalities.

Non-functional requirements represent the performance and operating environment of the system [12]. The non-functional requirements of the LearnPro App are shown in Table 3.

Table 3 *Non-Functional Requirements for LearnPro App*

No.	Requirement	Description
1	Performance Requirement	The application must respond to user interactions (e.g., login, task creation) within 2 seconds for a responsive user experience. Additionally, the system should be scalable to handle a growing number of users, with the ability to scale resources and databases to maintain performance.
2	Reliability Requirement	The application should maintain an uptime of at least 99.9% to ensure reliable access for users. Furthermore, the system should demonstrate fault tolerance, recovering gracefully from errors to minimize service disruptions.
3	Usability Requirement	The application must provide a consistent and intuitive user interface across all modules to enhance overall usability and user experience.
4	Security Requirement	User data, including login credentials and personal information, should be encrypted during transmission and storage for data security. Robust mechanisms for user authentication and authorization must be implemented to prevent unauthorized access to the system.

4.1.2 UML Diagram

The LearnPro App's UML diagram analysis reveals essential components for system development. The Use Case Diagram show in Appendix B illustrates 13 main user functions, including registration, task management, and rewards. The Sequence Diagram show in Appendix C depicts the order of method calls and responses, highlighting the dynamic collaboration between objects. The class diagram outlined in Appendix D serves as a visual representation of the software system's structure, depicting key classes such as Authentication, Users, Goals, Tasks, Pomodoro, Reward, Report, and Notification, facilitating communication, modeling, design clarity, code generation, documentation, and maintenance throughout the development process. Lastly, the Activity Diagram show in Appendix E provides a flowchart showcasing user interactions, navigation options, and dynamic behavior within the application. This comprehensive UML analysis ensures a clear understanding of the LearnPro App's structure and functionality, guiding the development process effectively.

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 538-557

4.2 Database Design

Designing a reliable and efficient database is a vital component of developing a mobile app system since it serves as the foundation for storing, managing, and retrieving data that is critical to the system's functionality.

4.2.1 Schema Database

The LearnPro mobile application's database schema is designed to support various features, including goals management, tasks management, profile, and Report. The schema includes Users Collection, Goals Collection, Tasks Collection, and Report Collection.

Table 4 *Users Collection*

No.	Attribute	Data Type	Size	Key	Description
1	id	VARCHAR	255	PRIMARY	Unique identifier for User
2	fullName	VARCHAR	255	NONE	User's full name.
3	email	VARCHAR	255	NONE	User's email address.
4	phoneNo	VARCHAR	255	NONE	User's phone number.
5	password	VARCHAR	255	NONE	User's password.
6	chance	VARCHAR	255	NONE	User's chance attribute (assuming this is a string).
7	profilePictureUrl	VARCHAR	255	NONE	User's profile picture URL

Users Collection in Table 4 stores information about each user. It includes details such as id, fullName, email, password, chance, and profilePictureUrl. The id serves as a unique identifier for each user. Users leverage this table to make Authentication, and profile Page. The table also supports data retrieval for personalization and customization features within the app.

Table 5 *Goals Collection*

No.	Attribute	Data Type	Size	Key	Description
1	id	String	Dynamic	PRIMARY	Unique identifier for each goal.
2	title	String	Dynamic	NONE	Title or name of the goal.
3	Note	String	Dynamic	NONE	Additional notes or details related to the task.
4	createdAt	DateTime	Dynamic	NONE	Date and time when the goal was created.
5	isCompleted	Boolean	Dynamic	NONE	Indicates whether the goal has been completed.

Goals Collection in table 5 allow users to organize tasks into meaningful lists. The Goals Collection, with attributes like id, title, Note, and createdAt enables users to create the objectives of the personalized learning plans.

Table 6 *Tasks Collection*

No.	Attribute	Data Type	Size	Key	Description
1	id	String	Dynamic	PRIMARY	Unique identifier for each task.
2	title	String	Dynamic	NONE	Title or name of the task.
3	priority	Integer	Dynamic	NONE	Priority level of the task.
4	note	String	Dynamic	NONE	Additional notes or details related to the task.
5	doneTimer	Integer	Dynamic	NONE	Total time spent on the task (in minutes).
6	guestTimer	Integer	Dynamic	NONE	Time allocated for the task (in minutes).
7	deadline	DateTime	Dynamic	NONE	Deadline for completing the task.
8	reminder	DateTime	Dynamic	NONE	Date and time of the task reminder, if set.
9	createdAt	DateTime	Dynamic	NONE	Date and time when the task was created.

Tasks Collection show in table 6 is at the core of the LearnPro app, enabling users to create and manage their learning tasks. Each task is characterized by attributes such as id, title, priority, note, doneTimer, guestTimer, deadline, reminder, createdAt. Users organize their learning activities, set priorities, and integrate Pomodoro timers to enhance focus.

Table 7 *Report Collection*

No.	Attribute	Data Type	Size	Key	Description
1	id	String	Dynamic	PRIMARY	Unique identifier for each report.
2	createdAt	DateTime	Dynamic	NONE	Date and time when the report was created.
3	taskId	String	Dynamic	NONE	Identifier of the task associated with the report.
4	goalId	String	Dynamic	NONE	Identifier of the goal associated with the report.

The Report Collection in table 7 is the LearnPro mobile app stores information about user reports, facilitating tracking and analysis of focus time of the user per week or month. This collection includes attributes designed to capture essential details about each report.

4.3 Implementation of Module

The LearnPro mobile app implementation is conducted in the Android Studio IDE using the Flutter framework. The Dart programming language is utilized to handle both the logic and design interface of the system. Firebase Firestore and Firebase Authentication are employed as the storage platforms for the system. The modules include user authentication, dashboard, real-time calendar, reservation management, hall management, user management, feedback, and user manual.

4.3.1 User Authentication Module

The LearnPro mobile app implementation is conducted in the Android Studio IDE using the Flutter framework. The application implements user authentication to ensure secure access to the system through login and registration functionalities. Users are required to authenticate themselves with valid credentials to gain access to the application's features. The Figure 2 show the user authentication interfaces to secure user access through login and registration.

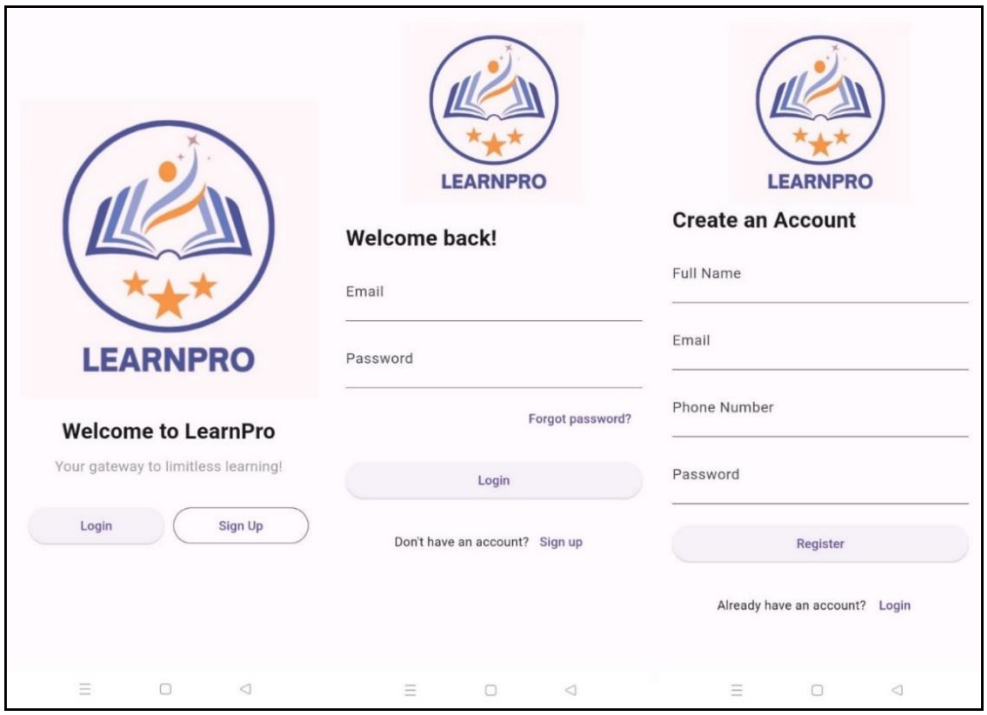


Fig 2. Welcome, Login, and Registration Pages

4.3.2 Forget Password Module

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 538-557

The forget password feature provides users with a seamless way to reset their forgotten passwords. Users can initiate the password reset process by providing their registered email address, and the system sends them a password reset link. The Figure 3 show the Forget password interface allow users to reset forgotten passwords.

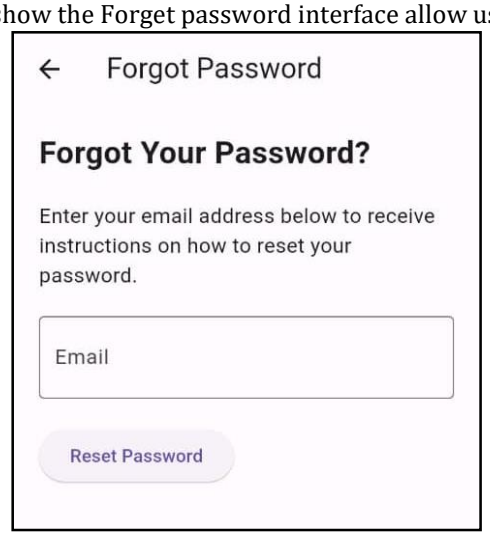


Fig 3. Forget Password Pages

4.3.3 Dashboard Module

The dashboard is designed to offer users a comprehensive summary of important information at a glance. It provides key insights, such as task progress, upcoming deadlines, and overall performance metrics. The Figure 4 show the Dashboard interface to display key information and insights for easy navigation and decision-making.

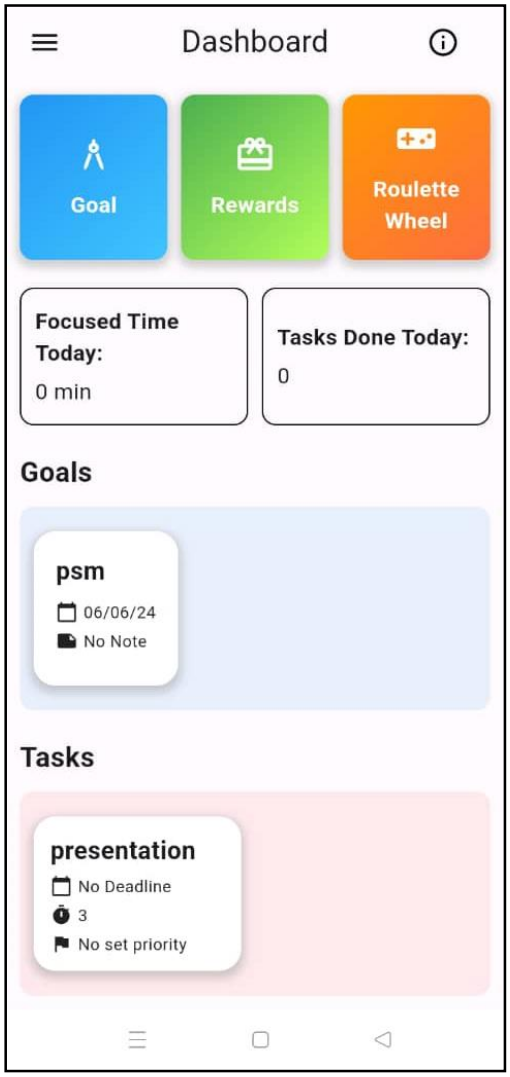


Fig 4. Dashboard Page

4.3.4 Profile Module

A dedicated profile section displays user information and settings. Users can view and update their profile details, including personal information. The Figure 5 show the Profile interfaces allow users to view and update their personal information.

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 538-557

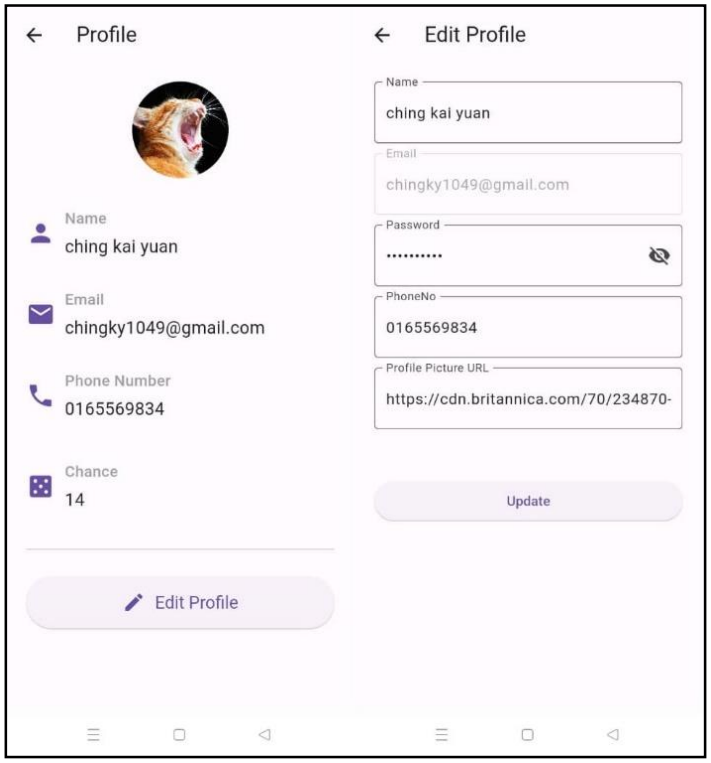


Fig 5. Profile Pages

4.3.5 Goals Management Module

This functionality allows users to create, read, update, and delete (CRUD) their goals. Users can set specific goals, track their progress, and modify them as needed. The Figure 6 show the Goals interfaces which enable users to manage their goals effectively.

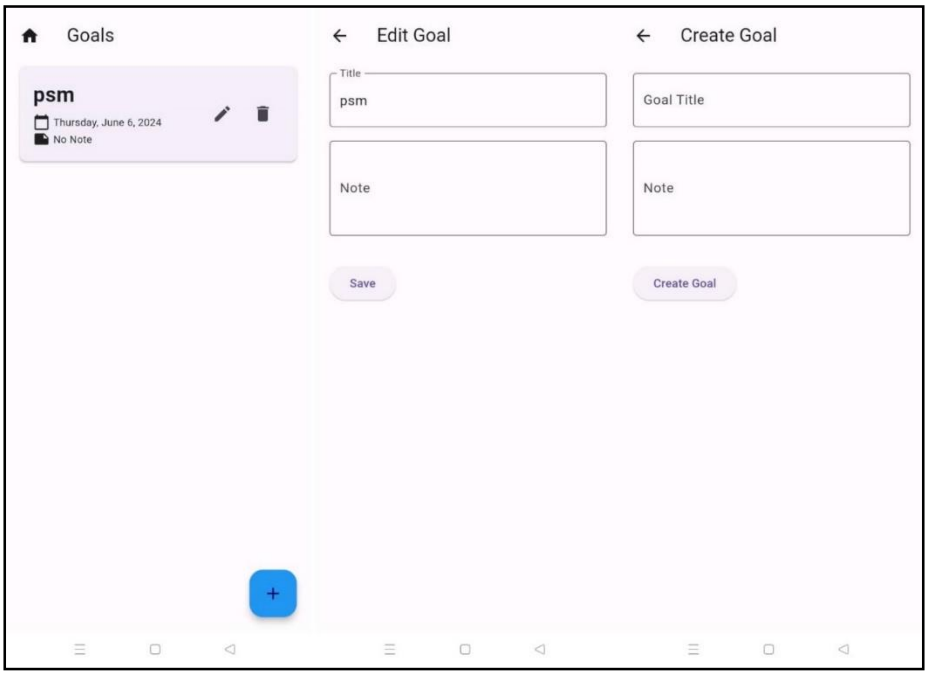


Fig 6. Goals Pages

4.3.6 Tasks Management Module

Users can efficiently manage their tasks within the application through CRUD operations. This includes creating, updating, deleting tasks, and marking them as completed or pending. The Figure 7 show the Tasks interfaces to facilitate effective task organization and execution.

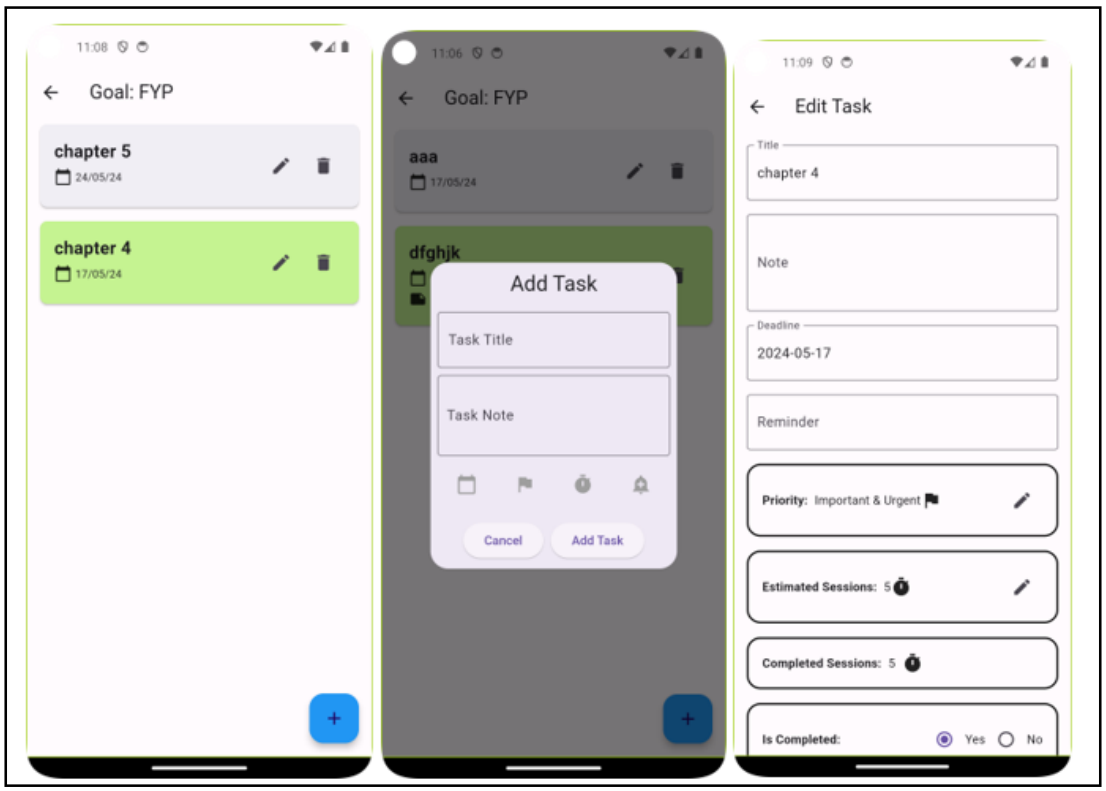


Fig 7. Tasks Pages

4.3.7 Pomodoro Timer Module

To enhance productivity and focus, the application integrates the Pomodoro Timer functionality. Users can manage their study or work sessions effectively with configurable timer settings. The Figure 8 show the pomodoro timer interfaces to improve productivity using the Pomodoro technique.

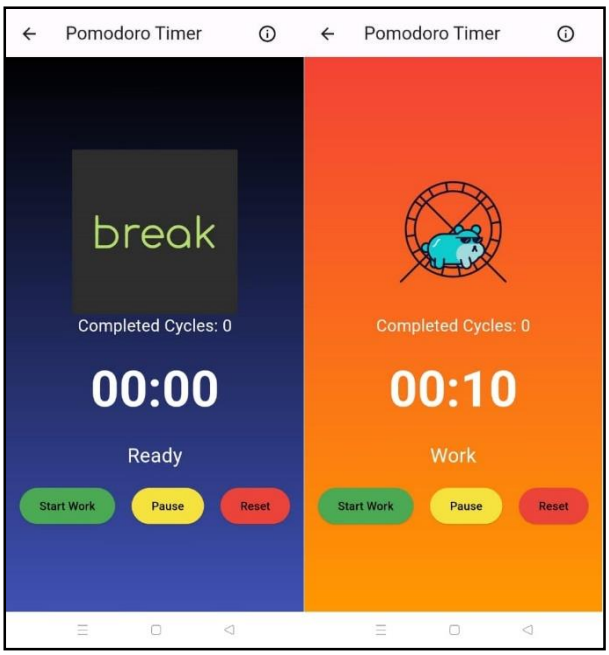


Fig 8. Pomodoro Timer Pages

4.3.8 Rewards Management Module

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 538-557

The application allows users to manage rewards through CRUD operations. Users can spin the wheel to earn rewards, motivating them to stay engaged and productive. The Figure 9 show the Rewards interface to manage and motivate users with rewards.

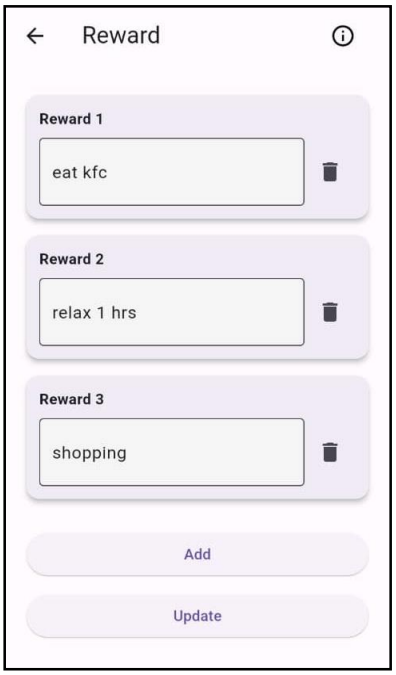


Fig 9. Rewards Management Pages

4.3.9 Spin Wheel Module

Users can spin a roulette wheel within the application to receive rewards, adding an element of gamification to the user experience. The Figure 10 show the Spin Wheel interface to incentivize user engagement and participation through gamification.

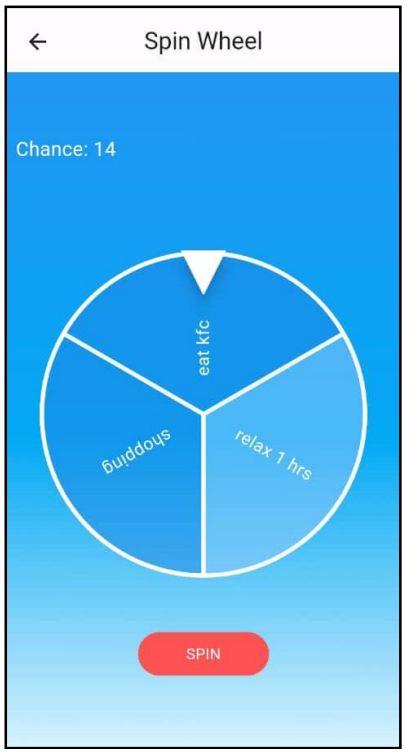


Fig 10. Spin Wheel Page

4.3.10 Report Module

The application allows users to view detailed reports summarizing their completed tasks and focus time over a specified period. Figure 11 show the Report interface provides valuable insights into their productivity and progress on graph.

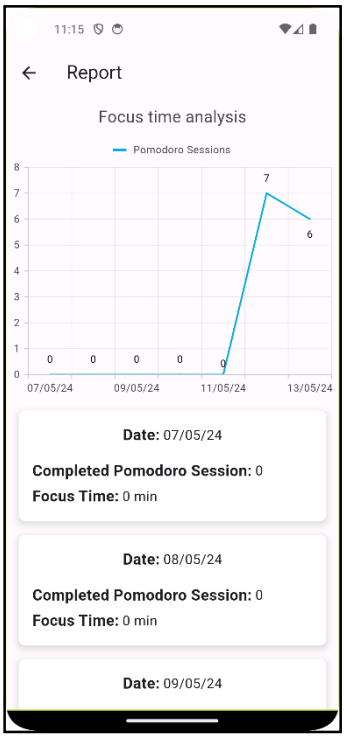


Fig 11. Report Page

4.3.11 Notification Module

The application includes a comprehensive notification interface to keep users informed and on track with real-time alerts and scheduled notifications. The Figure 12 show the Notification interfaces to keep users informed and reminded about important updates and activities.

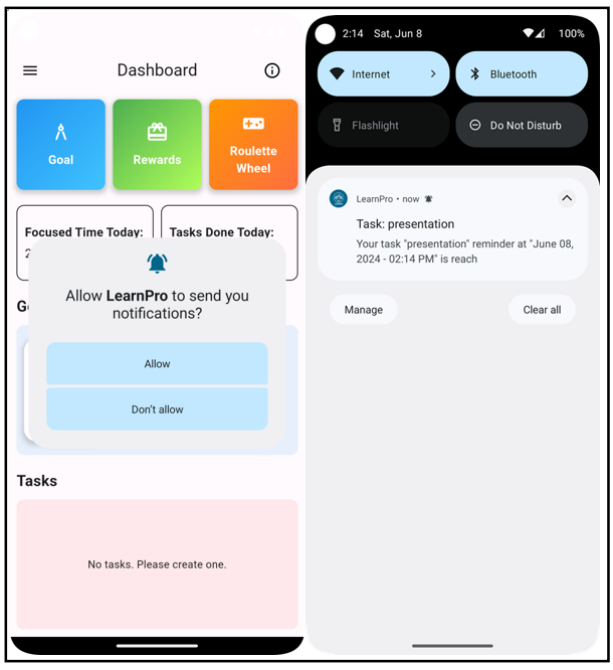


Fig 12. Notification Pages

4.3.12 Tutorial Module

The tutorial section provides a comprehensive walkthrough of the LearnPro mobile application's features, ensuring users can effectively navigate and utilize the app. The Figure 13 show the Tutorial interface to guide users through the app's features and functionalities.

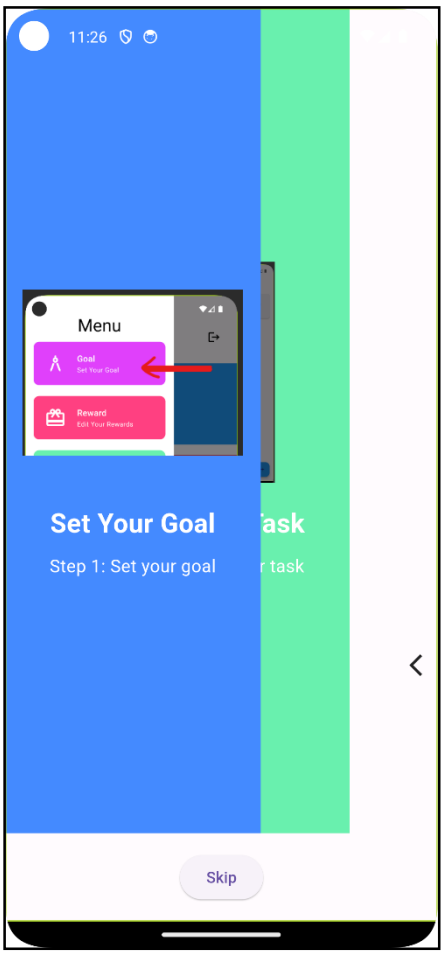


Fig 13. Tutorial Page

5. Testing

The Report Collection in the LearnPro mobile app stores information about user reports, facilitating tracking and analysis of focus time of the user per week or month. This collection includes attributes designed to capture essential details about each report.

5.1 Functional Testing

Functional testing is concerned with the functional requirements and ensuring that the system functions properly. Functional testing is performed to test and determine LearnPro mobile app whether it meets the functional requirements. Therefore, a comprehensive test plan is created as one of the functional testing strategies for the LearnPro mobile app.

5.1.1 Test Plan

The Test Plan outlines the testing strategy and approach for validating the functionality and reliability of the LearnPro mobile app. This plan encompasses various modules of the application, each with specific test cases designed to ensure that all features perform as expected. The table 8 below summarizes the results of the test cases for each module, including the Login Module, Registration Module, Forgot Password Module, and others. Each test case specifies the expected result, the actual result observed during testing, and the status of the test, whether it passed or failed. The successful execution of these test cases indicates that the application meets the specified requirements and is ready for deployment.

Table 8 *Test Plan of LearnPro Mobile App*

No.	Test Module	Expected Result	Status
1	**Login Module (TC_100)**	System authenticates user and displays appropriate messages based on input.	PASS
2	**Registration Module (TC_101)**	System creates a new account or displays an error message if the email is already in use.	PASS
3	**Forgot Password Module (TC_102)**	System sends a reset link for registered emails or displays an error for unregistered emails.	PASS
4	**Dashboard Module (TC_103)**	Dashboard displays a summary of important information such as tasks, goals, and rewards.	PASS
5	**Profile Module (TC_104)**	System displays user profile information and saves updates.	PASS
6	**Goals Management Module (TC_105)**	System performs CRUD operations on user goals.	PASS
7	**Tasks Management Module (TC_106)**	System performs CRUD operations on user tasks.	PASS
8	**Pomodoro Timer Module (TC_107)**	System starts, stops, and saves Pomodoro timer sessions.	PASS
9	**Rewards Management Module (TC_108)**	System performs CRUD operations on user rewards.	PASS
10	**Spin Wheel Module (TC_109)**	System spins the wheel and displays the reward won.	PASS
11	**Profile Management Module (TC_110)**	System performs CRUD operations on user profiles.	PASS
12	**View Report Module (TC_111)**	System displays a summary of completed tasks and focus time over the week.	PASS
13	**Notification Interface Module (TC_112)**	Users receive push and scheduled notifications as configured.	PASS
14	**Tutorial Module (TC_113)**	System displays the tutorial guide effectively.	PASS

6. Conclusion

In conclusion, the LearnPro mobile app presents a blend of advantages and disadvantages that shape its overall user experience. While it has successfully achieved its objectives, offering personalized learning experiences, effective time management tools, and motivational features, challenges such as platform dependence and limited offline functionality exist. Recommendations include expanding platform accessibility and enhancing customization options. Overall, LearnPro aims to provide a comprehensive and user-friendly learning environment for self-learners and students, with ongoing efforts focused on addressing existing limitations and enhancing user satisfaction.

6.1 System Advantages

The LearnPro mobile app offers several advantages that enhance the overall user experience and learning efficiency. It provides a personalized learning experience by allowing users to create tailored study plans that meet their individual needs, thereby increasing engagement and improving learning outcomes. The app includes effective time management tools, such as the Pomodoro timer and task management features, which help users manage their study sessions efficiently, leading to increased productivity. Additionally, motivational tools like the Roulette Reward Mechanism incentivize users to stay committed to their learning goals by offering personalized rewards, thereby boosting motivation and engagement.

6.2 System Disadvantages

Despite its strengths, the LearnPro mobile app has some disadvantages that may hinder the user experience. The app's platform dependence limits its accessibility to mobile devices, potentially alienating users who prefer desktop or web-based platforms. Additionally, the complete loss of functionality offline poses a significant challenge, as the app relies heavily on internet connectivity, making it unusable in areas with poor or no internet access. Furthermore, the app offers limited customization options, restricting users' ability to personalize their learning environment according to their preferences, which may reduce user satisfaction and engagement.

6.3 Recommendations

To address these disadvantages and enhance the overall user experience, several recommendations are proposed for the LearnPro mobile app. Firstly, expanding the platform accessibility to include desktop and web-based versions would cater to a wider audience and accommodate users who prefer alternative learning platforms. Secondly, improving offline functionality by incorporating features that allow access to certain functionalities or study materials without an internet connection would significantly enhance usability in areas with poor connectivity. Lastly, offering enhanced customization options, such as customizable themes, layouts, and notification preferences, would allow users to personalize their learning environment, thereby increasing engagement and satisfaction with the app.

Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Ching Kai Yuan, Rozanawati Darman;* ***data collection:*** *Ching Kai Yuan;* ***analysis and interpretation of results:*** *Ching Kai Yuan, Rozanawati Darman;* ***draft manuscript preparation:*** *Ching Kai Yuan, Rozanawati Darman. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] P. Grant and D. Basye, *Personalized learning: A guide for engaging students with technology*. International Society for Technology in Education, 2014.

[2] I. Y. Jibrin and K. M. Ibrahim, "Design of a Personalized Learning Management System: Enhancing Student Learning Preferences," 2023.

[3] S. O'connor and T. Andrews, "Smartphones and mobile applications (apps) in clinical nursing education: A student perspective," *Nurse Educ. Today*, vol. 69, pp. 172–178, 2018.

[4] M. Zotou, E. Tambouris, and K. Tarabanis, "Data-driven problem based learning: Enhancing problem based learning with learning analytics," *Educ. Technol. Res. Dev.*, vol. 68, no. 6, pp. 3393–3424, 2020.

[5] K. C. Berridge, "Reward learning: Reinforcement, incentives, and expectations," in *Psychology of learning and motivation*, vol. 40, Elsevier, 2000, pp. 223–278.

[6] H. Alamri, V. Lowell, W. Watson, and S. L. Watson, "Using personalized learning as an instructional approach to motivate learners in online higher education: Learner self-determination and intrinsic motivation," *J. Res. Technol. Educ.*, vol. 52, no. 3, pp. 322–352, 2020.

[7] R. Angelini and V. Magnifico, "Pomodoro," *Colt. Cult. Milan, Italy*, 2010.

[8] V. Yakovyna, M. Seniv, and I. Symets, "The Relation between Software Development Methodologies and Factors Affecting Software Reliability," in *2020 IEEE 15th International Conference on Computer Sciences and Information Technologies (CSIT)*, 2020, pp. 377–381. doi: 10.1109/CSIT49958.2020.9321937.

[9] M. Mukherjee, "Object-Oriented Analysis and Design," *Int. J. Adv. Eng. Manag.*, vol. 1, no. 1, pp. 1–11, 2016.

[10] tutorialspoint, "Object Oriented Design - Wikiversity." [Online]. Available: http://en.wikiversity.org/wiki/Object_Oriented_Software_Design

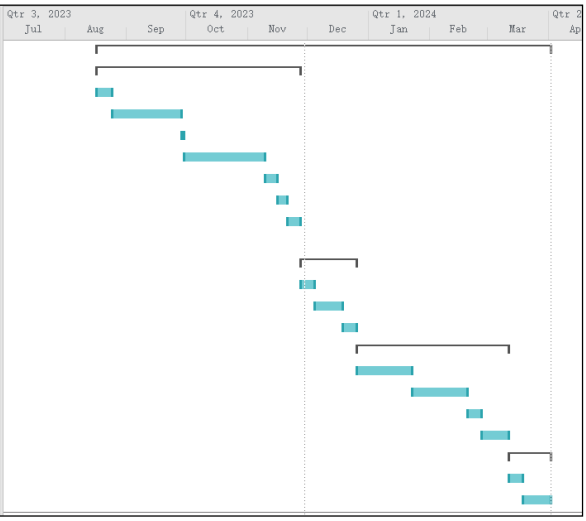
[11] T. Hamilton, "What is User Acceptance Testing (UAT)? with Examples," 2019, [Online]. Available: https://www.guru99.com/user-acceptance-testing.html

[12] H. Kaur and A. Sharma, "ANOVA Based Significance Testing of Non-functional Requirements in Software Engineering," *Int. J. Inf. Technol. Proj. Manag.*, vol. 10, no. 4, pp. 100–117, 2019.

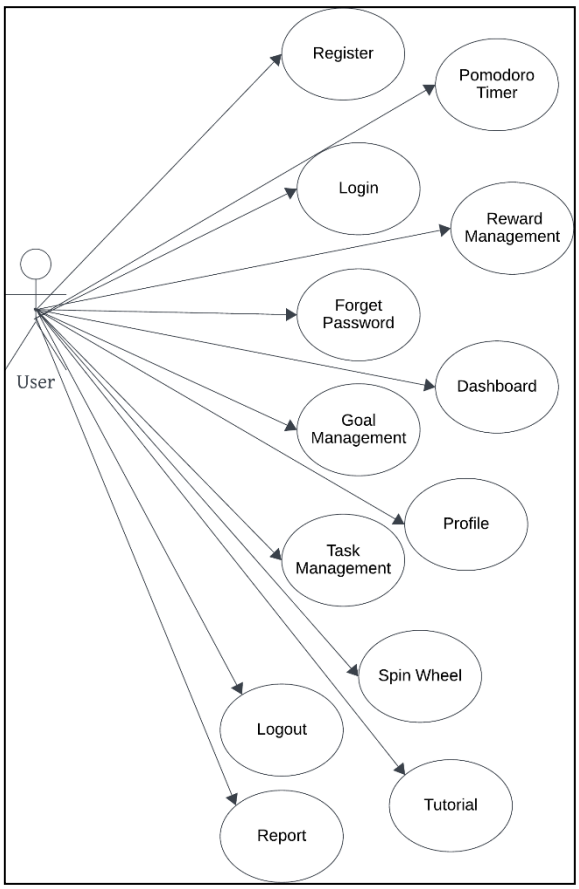
Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 538-557

Appendix A: Gantt chart

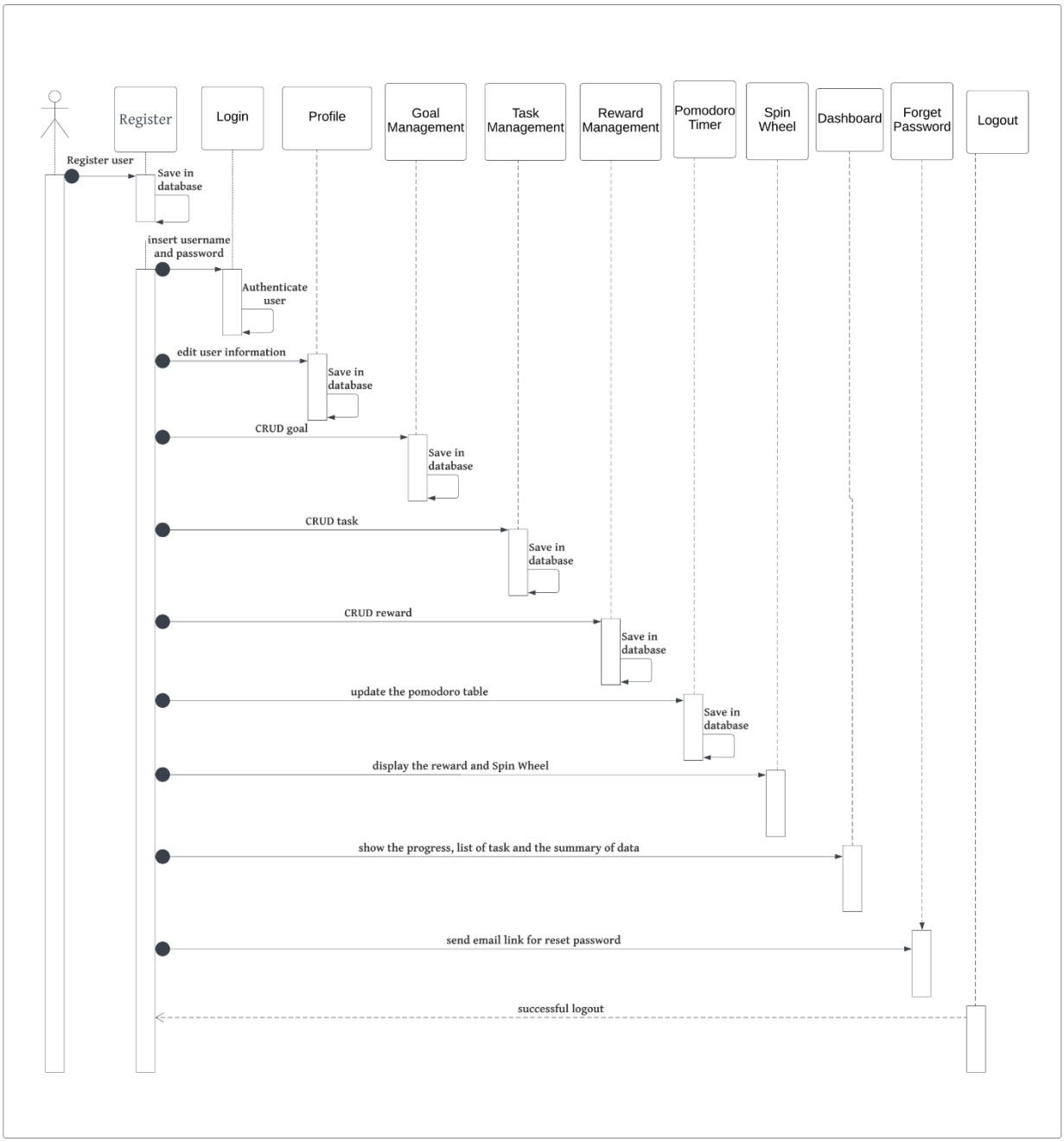
Task Mode	Task Name	Duration	Start	Finish	Pred
	LearnPro Mobile APP	**163 days**	**Thu 17/08/23**	**Mon 01/04/24**	
	Object-Oriented Analysis Phase	**73 days**	**Thu 17/08/23**	**Mon 27/11/23**	
	Generates ideas for project title	6 days	Thu 17/08/23	Thu 24/08/23	
	Prepare draft project proposal	25 days	Fri 25/08/23	Thu 28/09/23	
	Discuss with supervisor	1 day	Fri 29/09/23	Fri 29/09/23	
	Prepare project proposal	30 days	Sat 30/09/23	Thu 09/11/23	
	Determine the requirements	4 days	Fri 10/11/23	Wed 15/11/23	
	Study existing system	3 days	Thu 16/11/23	Mon 20/11/23	
	Compare existing system and proposed system	5 days	Tue 21/11/23	Mon 27/11/23	
	Object-oriented design	**20 days**	**Tue 28/11/23**	**Mon 25/12/23**	
	Design UML diagram	5 days	Tue 28/11/23	Mon 04/12/23	
	Design interface	10 days	Tue 05/12/23	Mon 18/12/23	
	Design database	5 days	Tue 19/12/23	Mon 25/12/23	
	Object-oriented implementation	**55 days**	**Tue 26/12/23**	**Mon 11/03/24**	
	Develop system interfaces	20 days	Tue 26/12/23	Mon 22/01/24	
	Develop system modules	20 days	Tue 23/01/24	Mon 19/02/24	
	Integrate system	5 days	Tue 20/02/24	Mon 26/02/24	
	Connect with database	10 days	Tue 27/02/24	Mon 11/03/24	
	Object-oriented testing	**15 days**	**Tue 12/03/24**	**Mon 01/04/24**	
	Test the system	5 days	Tue 12/03/24	Mon 18/03/24	
	Debug the faced error when testing	10 days	Tue 19/03/24	Mon 01/04/24	



Appendix B: Use Case Diagram

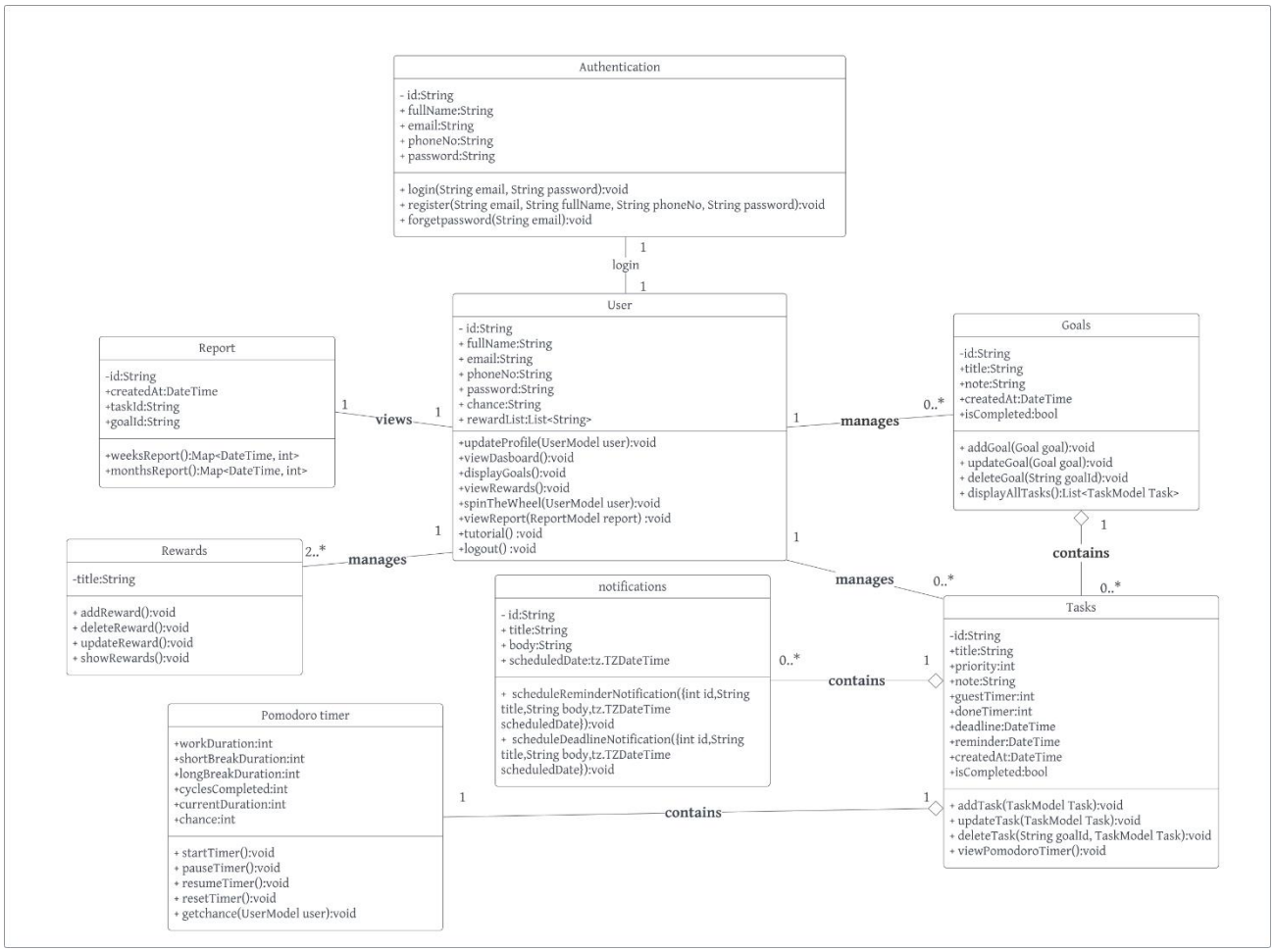


Appendix C: Sequence Diagram



Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 538-557

Appendix D: Class Diagram



Appendix E: Activity Diagram