IJOBTECH
INTERNATIONAL JOURNAL OF SMART BUSINESS AND TECHNOLOGY

Vol. 1 No. 1 (2025)

A Design Thinking Approach to the Design and Development of the MoneyMate Website for Personal Financial Management

Anisa Amalia[1], Dwi Ariyanti[2], Laurenta Dini Aprinda[3], Rilla Shadina Arifin[4]

[1 2 3 4] *Informatics Study Program, Faculty of Computer Science, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Rungkut Madya, Surabaya 60294, Indonesia*

Abstract

Money Mate is a web-based personal finance management application designed to assist users in efficiently recording and managing their daily financial transactions. This study applies the design thinking method, which includes the stages of empathy, problem definition, ideation, prototyping, and testing. The development process was based on the author's personal experience of difficulties in tracking income and expenses. The resulting application features core functions such as transaction recording, real-time display of total income, total expenses, balance summary, and graphical financial reports. Testing indicates that the application meets user needs and can improve financial awareness and literacy. This application is expected to be an effective solution in supporting personal financial management and has the potential for further development using advanced technologies.

Keywords: *Finance Management; Web Application; Design Thinking*

Article History:

Received	:	26 June 2025
Revised	:	10 October 2025
Accepted	:	23 October 2025
Published	:	31 October 2025

Introduction

The rapid advancement of technology in the era of globalization, particularly in Indonesia, has driven the emergence of various web and mobile-based applications that accelerate the dissemination of information and simplify everyday tasks. Today, digital technology has permeated nearly every aspect of people's lives, including personal financial management. There is significant potential to develop effective and practical digital solutions, driven by the increasing accessibility of the internet and the widespread use of mobile devices.

Although manual record-keeping is still widely used in the context of personal financial management, it is often inefficient and time consuming. Common issues include slow recording processes, the risk of data loss, and difficulties in conducting financial analysis. Therefore, there is a need for a mobile financial management application that is easy to use for all groups, enabling users to manage their finances in a detailed and controlled manner.

Every individual needs to implement personal financial management to balance income and expenses, meet daily needs, and avoid financial failure. However, many people—especially the younger generation—struggle with managing their finances due to low financial literacy and a consumptive lifestyle, which is often triggered by the ease of access to e-commerce platforms that provide and serve various needs online, as well as the influence of social media. In fact, managing, spending, saving, and avoiding waste are essential aspects of sound financial management that can improve well-being and financial stability [9].

Personal financial management is considered an art of managing financial resources by individuals to achieve goals effectively and efficiently for their personal well-being. To support this need, utilizing technology in the form of a web application may be the most appropriate choice. A web-based financial management system offers advantages in terms of data security, ease of use, and speed in recording transactions [6].

MoneyMate: Financial Manager is a web application developed to address these needs. This application enables users to record financial transactions, view summaries of income and expenses, and display data visualizations in the form of charts. Additionally, it includes an authentication system that protects user data and ensures privacy. Through this approach, the application is expected to help users make more logical financial decisions and gain better control over their financial situation.

[1] **CONTACT**: dwiariyanti613@gmail.com

Problem Formulation

The issues addressed in this project include:

1. How to develop a web application that can effectively record and manage financial transactions?
2. How can monthly income and expense trends be visualized using graphs?
3. How can a system be developed to display real-time summaries of balance, differences, and expense percentages?
4. How to provide a search feature so that users can easily find the desired transaction data?
5. How to ensure the security of financial data through a user authentication system?

Objectives

The objectives of this application development are:

1. To build a web application capable of recording, managing, and reporting financial transactions digitally.
2. To provide financial visualisation features in the form of monthly bar charts to facilitate users in analyzing their financial data.
3. To display structured financial summary information such as balance, total income, expenses, and the difference in real time.
4. To provide CRUD and search features of financial transaction data.
5. To implement login, registration, and logout systems to ensure the security of user data.

Benefits

The expected benefits of developing the MoneyMate application are as follows:

- For Users:
  - Facilitates users in recording and monitoring income and expenses digitally without relying on manual methods.
  - Motivate users to better manage their finances through engaging visualisation and summary features.
- From a Personal Financial Analysis Perspective:
  Provides graphical representations that help users understand their monthly financial patterns, helping to make better financial decisions.
- In Terms of Efficiency and Productivity:
  Improves the efficiency of the financial recording and reporting process in real time.
- Data Security:
  An authentication system involving login and registration ensures that users financial data is secure and protected.

Scope

The scope of the MoneyMate application development includes:

- Platform:
  The application is web-based and accessible through desktop and laptop browsers.
- Core Functions:
  - Income and expenses are recorded as financial transactions.
  - Displays of total balance, financial differences, and percentages of expenditure.
  - Monthly bar chart visualization of income and expenses.
  - Displays information about the latest transactions.
- Security Features:
  A system that uses user authentication for login, registration, and logout processes.
- Limitations:
  - The application is only used for personal financial management and not for large-scale corporate financial operations.
  - It does not include advanced account management features, such as password deletion or multi-level authorization.

Methodology

The research method used in the development of the MoneyMate application is **Design Thinking**. Design Thinking is a user-centered approach aimed at solving problems and providing effective solutions. This method was chosen because it aligns with the main goal of the MoneyMate application, which is to assist users in managing their personal finances. Based on user needs, the application is designed to help record transactions, view financial reports, and receive monthly notifications as part of the evaluation process. Figure 1 illustrates the stages of the Design Thinking process.

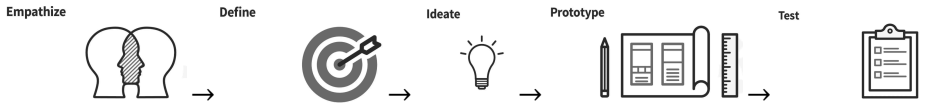

Figure 1. Stages of the design thinking process.

The stages carried out in the Design Thinking method are as follows[12]:

1. Empathize
   This is the initial stage, where the goal is to understand user needs. In this stage, the author approached the problem based on personal experience without involving respondents or conducting surveys, and performed observations of existing financial management applications.
2. Define
   This stage involves analyzing and understanding user needs in order to offer suitable solutions. Based on personal experience, one of the main problems faced by the author was the difficulty in managing and recording daily income and expenses.
3. Ideate
   This is the process of generating ideas to address the problems identified. At this stage, the author focused on generating the idea to develop the *MoneyMate* application and determining its core features. The planned features for the MoneyMate application include: landing page, start page, dashboard/home, add transaction, financial report page, graphical financial visualization, and account page.
4. Prototype
   This stage involves creating an initial design or wireframe of the application using tools such as Figma. The author implemented the previously defined features into a wireframe model of the MoneyMate application. After completing the wireframe, a mockup (final application layout) was created by selecting the appropriate color scheme, fonts, and icons to be used. Once the mockup was completed, a clickable prototype was developed to simulate the real flow of the application, which was then tested.
   Following the prototyping process using Figma, the system was developed into a static/dynamic website using HTML, CSS, JavaScript, and PHP.
5. Test
   This is the final stage, where the prototype is tested to ensure that the MoneyMate application design functions properly and meets user needs. This stage also allows for revisions if design errors or usability issues are identified.

Analysis and Discussion

Database Design

Entity Relationship Diagram (ERD) is a diagram used in designing databases. This diagram describes the relationship between entities or objects and their attributes. Thus, ERD serves as a model for describing the relationship of data in a database based on the main data objects that are interconnected in relationships. Entity-Relationship is one of the database modeling methods used to generate conceptual schemas for system semantic data types/models. Where systems often have relational databases, and their provisions are top-down. The diagram to describe this Entity-Relationship model is called an Entity-Relationship diagram, ER diagram, or ERD [1].

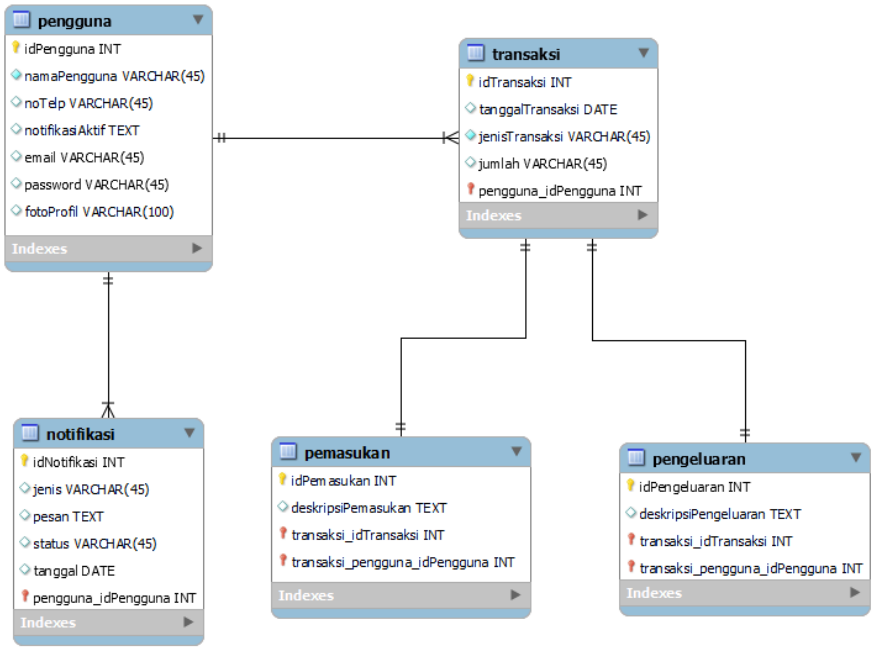

Figure 2. Entity Relationship Diagram (ERD) of personal finance management system.

Figure 2 is the Entity Relationship Diagram (ERD) for the user-based financial management system. This diagram consists of six main entities: users, transactions, income, expenses, notifications, and relationships between entities. The user entity stores user data such as name, email, phone number, password, and notification status. The transaction entity records the financial activities performed by the user, including the date, type of transaction (income/expense), and amount. The income and expense entities each store details of the financial transactions made, such as descriptions and relationships to transaction entities. Meanwhile, notifications store information on warnings or messages sent to users regarding certain financial activities. The relationship between entities shows that each transaction and notification is directly connected to one user, and each transaction can have details on both income and expenses. This structure is designed to support organized financial records that are directly connected to the users of the system.

System Design

System design is an important stage in the development of information systems, which is carried out after the analysis stage is complete, and includes describing the features and relationships between objects in the system [5].

One of the tools used in the system design process is the Unified Modeling Language (UML). UML is used to describe the visualization of the system to be developed in a clear and structured manner. UML diagrams consist of several types, including use case diagrams, activity diagrams, sequence diagrams, and class diagrams, each of which represents a different perspective of the system. The following section will explain the UML diagrams used.

Use Case Diagram

Use case diagram is a model resulting from system design analysis to explain what is needed by the system. These needs will later be used by users, so that the system design can be clearly reflected [7]. The following use case diagram has been created for the web-based personal finance manager.

Figure 3. Use case diagram.

This use case diagram shows the interaction between the user and the system in the financial transaction management application. Users can register, login (with authentication), add, view, edit, and delete transactions, as well as view transaction charts and log out. Each main action such as adding or editing transactions involves internal system processes such as saving data, displaying pages, sending notifications, or displaying graphs. The transaction recapitulation feature also runs automatically when the user views the chart or modifies the data. This diagram summarizes the main flow and system support for each user activity in a structured way.

Activity Diagram

Activity diagram is a visual tool used in system modeling to describe the flow of work or activities that occur in a real process [2].

1. Activity Diagram Registration

This activity diagram explains the flow of the user registration process on the system. The process starts when the user opens the website, then the system displays the login page. The user then selects the register option and fills in data such as full name, email, and phone number. After that, the user presses the register button. The system then validates the data entered. If the data is valid and the registration process is successful, the user will be directed to the login page. Otherwise, if an error occurs, the system will display an error message. This diagram illustrates the interaction between the user and the system sequentially from the beginning to the end of the registration process. The activity diagram of the registration flow in the figure 4.

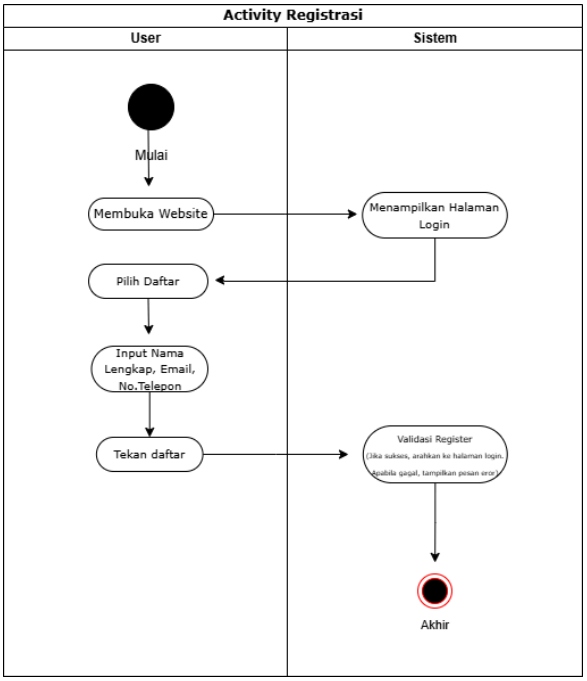

Figure 4. Activity diagram registration.

2. Activity Diagram Login

This activity diagram illustrates the flow of the user login process on the system. The process starts when the user opens the website, then the system displays the login page. After that, the user is asked to enter an email. The email data is then validated by the system. If the login process is successful, the user is directed to the main page. However, if the login fails, the system will display an error message. This diagram shows the sequence of steps that occur during the login process, from start to finish. The login flow activity diagram in Figure 5.

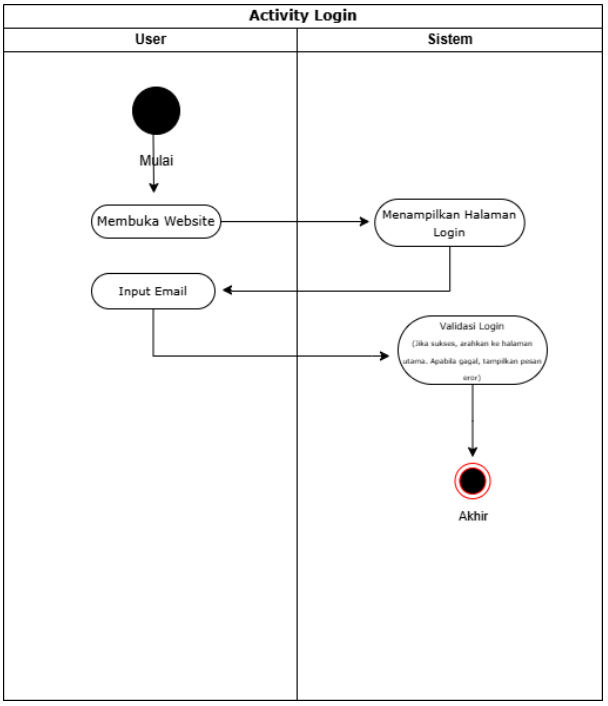

Figure 5. Activity diagram login.

3. Activity Diagram of Add Transaction

This activity diagram explains the process flow of adding income and expenditure transactions in the system. The process begins when the user opens the "Add Transaction" menu. The system then displays a menu of options, and the user selects the type of transaction whether it is income or expenditure. After that, the user fills in the transaction details such as date, amount, and description, then presses the save button.

Next, the system validates the data entered. If the data is valid, the system saves the transaction into the database, updates the total income or expenses, and displays a confirmation message to the user. This diagram illustrates the interaction between the user and the system from the beginning to the end of the transaction addition process. As for the activity diagram of the added transaction flow in the figure 6.

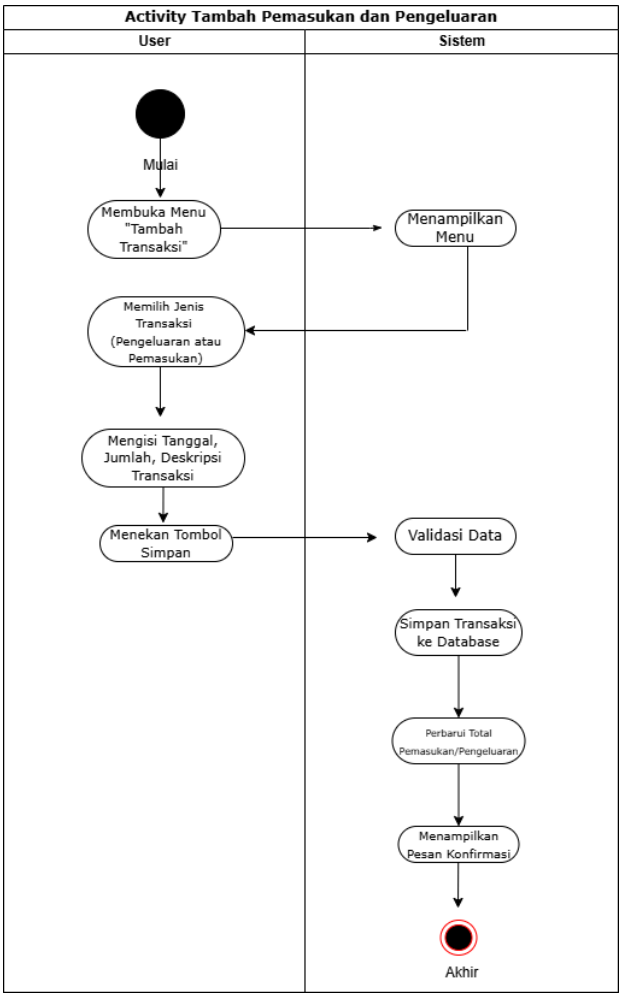

Figure 6. Activity diagram off add transaction.

4. Activity Diagram for Updating and Deleting Transactions

This activity diagram illustrates the process of editing or deleting transactions in the system. The process starts when the user opens the transaction history menu. The system then displays a list of transaction history that has been done. The user selects the transaction they want to edit or delete, then makes changes to the data or deletes it. After that, the system updates or deletes the transaction data in the database according to the action selected by the user. Furthermore, the system also updates the total income or expenditure and displays a confirmation message to the user as a sign that the process has been completed. As for the activity diagram of the update and delete transaction flow in the picture 7.

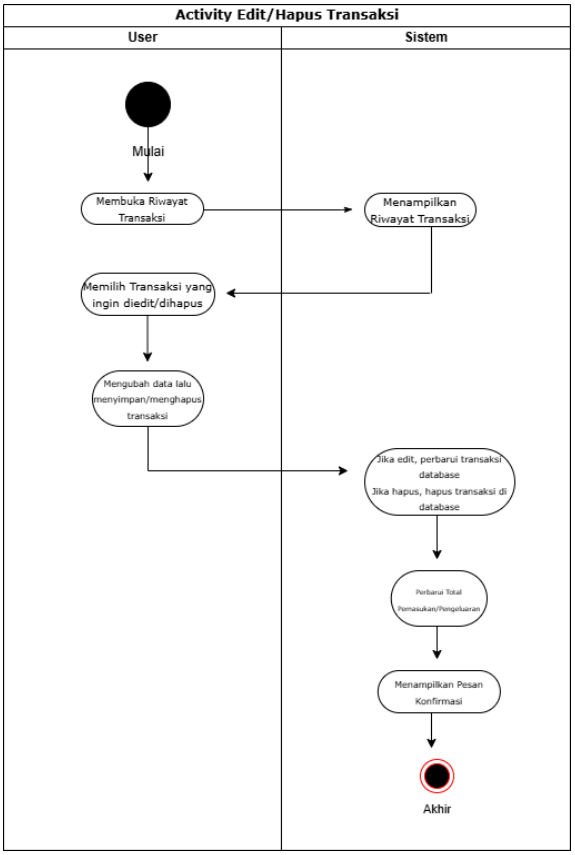

Figure 7. Activity diagram for updating and deleting transactions.

5. Activity Diagram Show Transaction List

This activity diagram illustrates the process flow when users view financial reports in the system. The process starts when the user opens the "Financial Report" menu, then the system displays the financial report page. The system then retrieves income and expenditure data for the last 12 months from the database, calculates total income, total expenditure, and the difference. After that, the system displays an annual graph in the form of a bar chart and provides a financial summary of the last month. The user can view the graph and summary, then press the "Back" button to be directed back to the main page. This diagram illustrates the sequential interaction between the user and the system during the financial report generation process. As for the activity diagram of the flow of displaying the list of transactions in the figure 8.

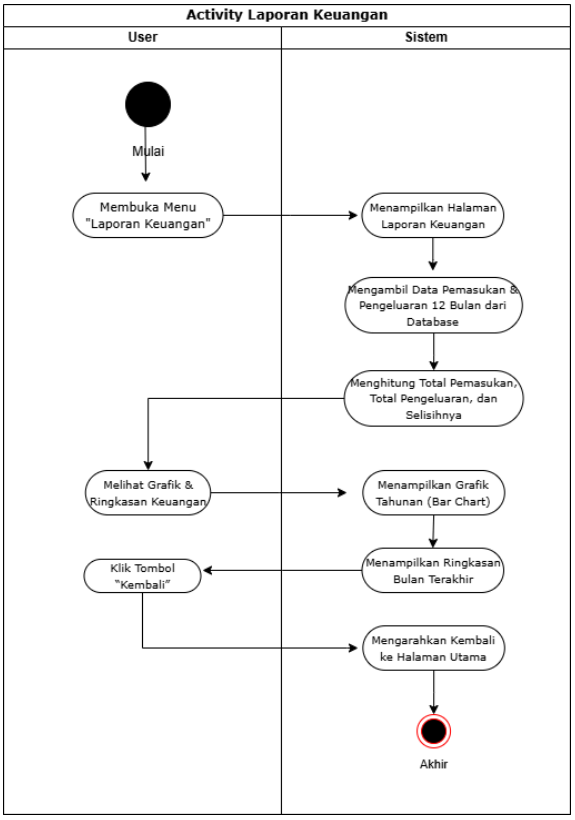

Figure 8. Activity diagram show transaction list.

6. Activity Diagram Logout

This activity diagram explains the flow of the logout process in the system. The process begins when the user opens the profile menu and selects the logout option. After that, the user is asked to confirm the logout. If the user confirms, the system will delete the user's session from the system to make sure the user is really logged out. After the session is deleted, the system then directs the user back to the login page as the final step of the logout process. This diagram illustrates a brief interaction between the user and the system in the process of logging out of the application. As for the activity diagram of the logout flow in the figure 9.

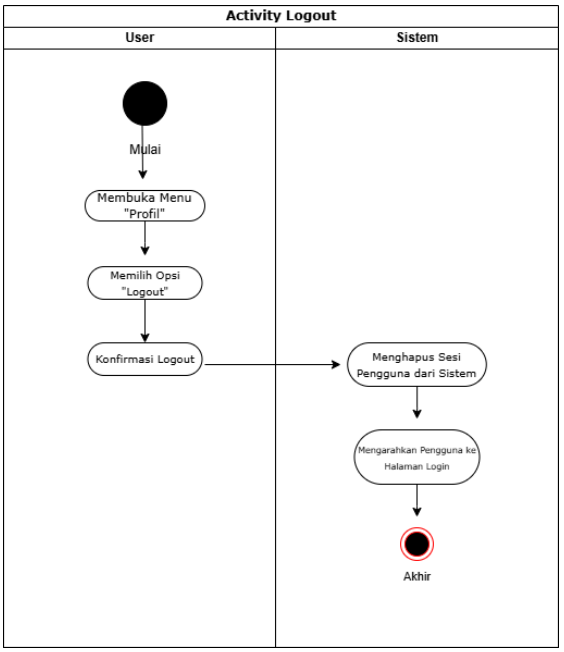

Figure 9. Activity diagram logout.

Sequence Diagram

Sequence diagram is a diagram that displays the results of interactions that occur from responses between objects that send messages to each other [2].

1. **Sequence Diagram Registration**

This sequence diagram describes the flow of user account registration. The user fills in the data and presses the register button. The system validates the input; if the data is incomplete, an error message is displayed. If complete, the account is saved to the database and a verification link is sent to the user's email. Once the user clicks the link, the system updates the verification status and redirects the user to the login page. This process ensures that only verified accounts can log in to the system. The registration flow sequence diagram is shown in Figure 10.

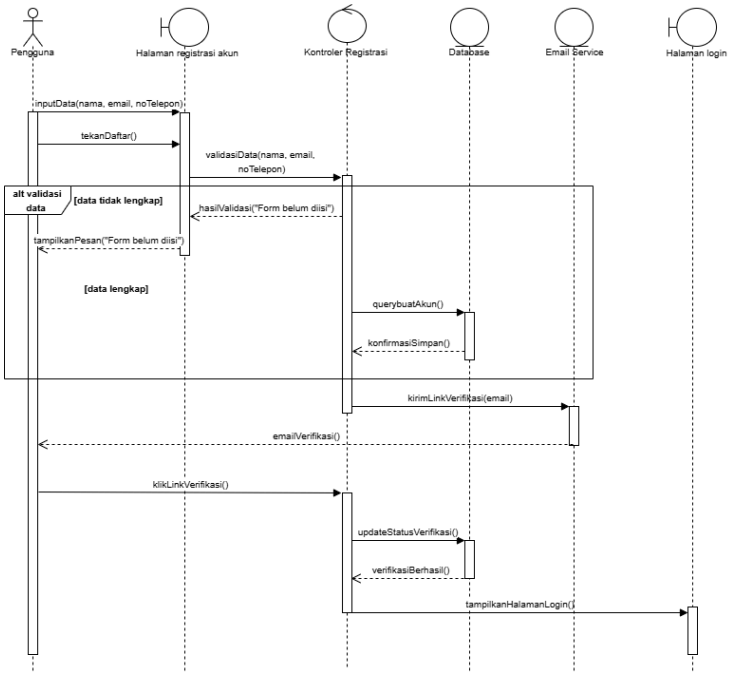

Figure 10. Sequence diagram registration.

2. **Sequence Diagram Login**

This sequence diagram explains the user login process based on email. The user fills in the login form and the system validates the email entered. The controller then matches the data to the database. If the email is not found, the system displays the error message “Email not found”. However, if the email is valid and matched, the user data is sent back and the system displays the dashboard page. This diagram emphasizes data validation as an important step in the login process. The login flow sequence diagram in Figure 11.

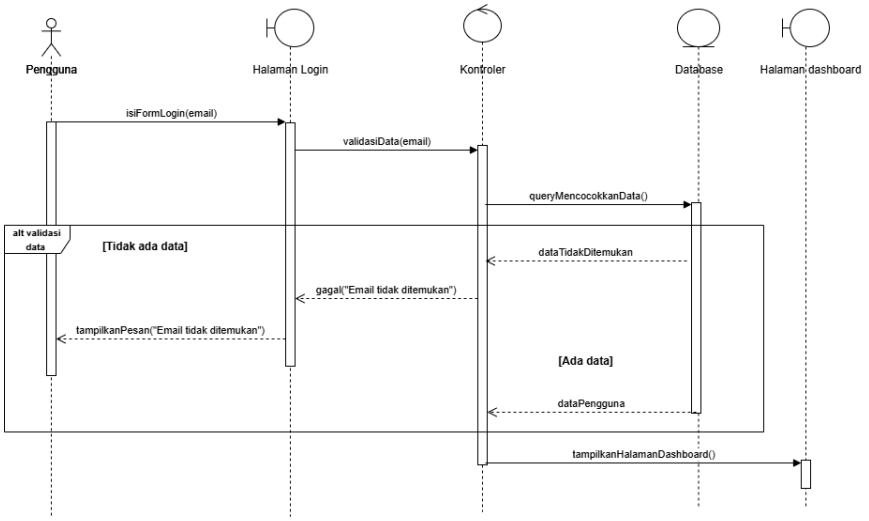

Figure 11. Sequence diagram login.

3. **Sequence Diagram of Add Transaction**

This sequence diagram illustrates the flow of the process of adding transactions by users. The user fills in the transaction form (type, date, amount, description) then presses the "Add" button. The system validates the data; if there is something missing, it displays the message "The form has not been filled in". If the data is complete, the controller saves the transaction to the database, then displays the message "Add transaction successfully" and updates the transaction list with the new data. This diagram emphasizes the importance of validation before data saving. The sequence diagram of the added transaction flow in Figure 12.

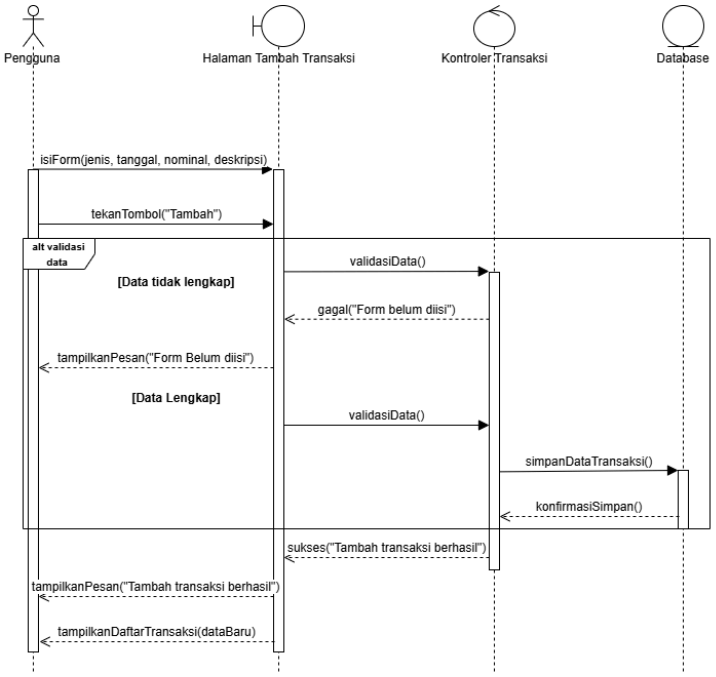

Figure 12. Sequence diagram of add transaction.

4. Sequence Diagram Show Transaction

This sequence diagram describes the process flow when the user views the transaction list. The user selects the type of transaction (income/expenditure), then the system sends a request to the controller to retrieve data from the database. If the data is not found, the system displays the message "No transaction yet". If the data is available, the transaction list is displayed along with the success message "List of transactions this month". This diagram shows two possible responses: empty data or available data. The sequence diagram of the show transaction flow in Figure 13.

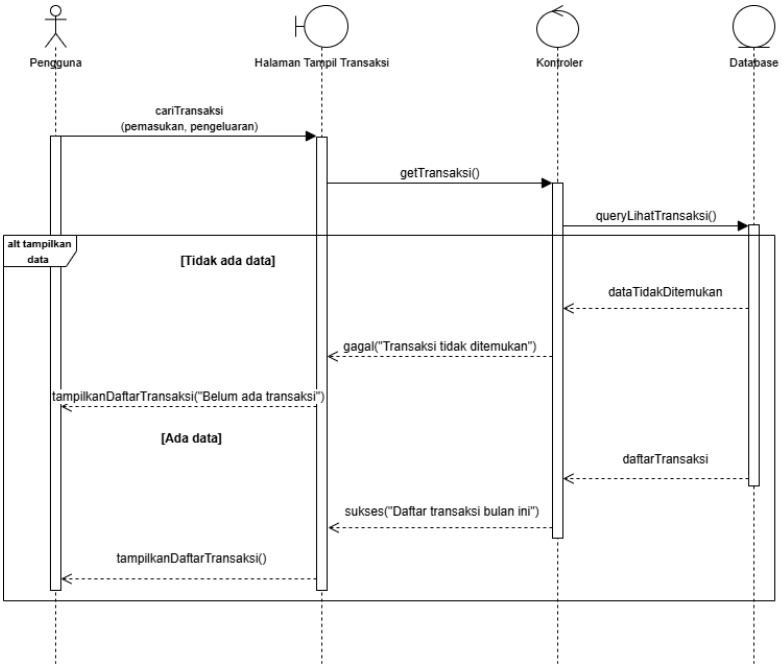

Figure 13. Sequence diagram show transaction.

5. Sequence Diagram of Update Transaction

This sequence diagram illustrates the process when a user edits a transaction. The user presses the edit button, then the system retrieves transaction data based on the ID. If the data is not found, the message "Transaction not found" is displayed. If the data exists, the edit form is displayed and the user submits the changes. The system then updates the data in the database, and if successful, displays a notification and the update result. This diagram emphasizes two main conditions: data is not found or the edit process is successful. The sequence diagram of the transaction update flow in Figure 14.

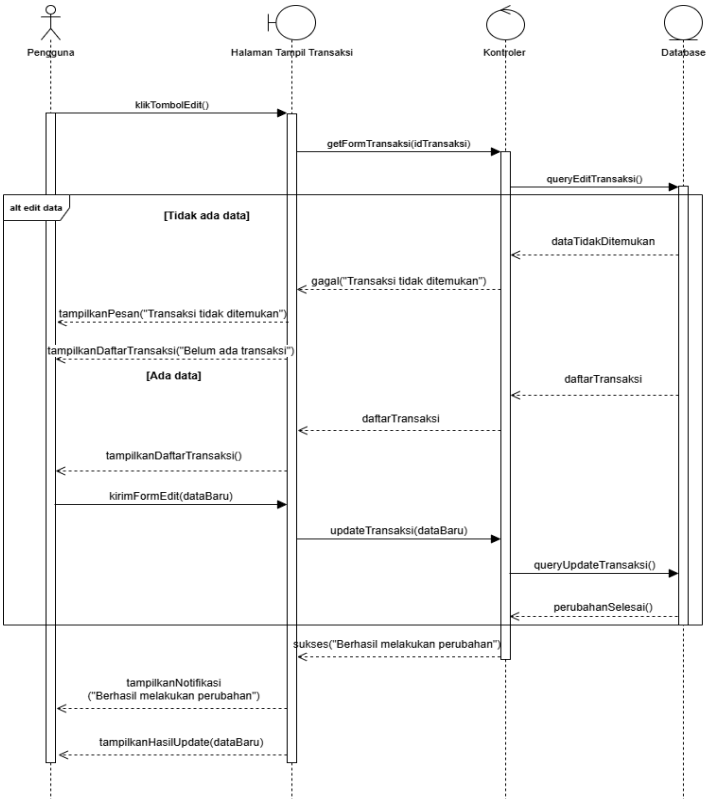

Figure 14. Sequence diagram of update transaction.

6. Sequence Diagram of Delete Transaction

This sequence diagram describes the process when a user deletes a transaction. The user presses the delete button, then the system sends a request to the controller to delete data based on the transaction ID. If the data is not found, the system displays the message "Transaction not found". If the data is found and successfully deleted, the system displays the message "Transaction successfully deleted" and updates the transaction list. This diagram illustrates two possibilities: failed transaction or successful deletion. The sequence diagram of the delete transaction flow in Figure 15.

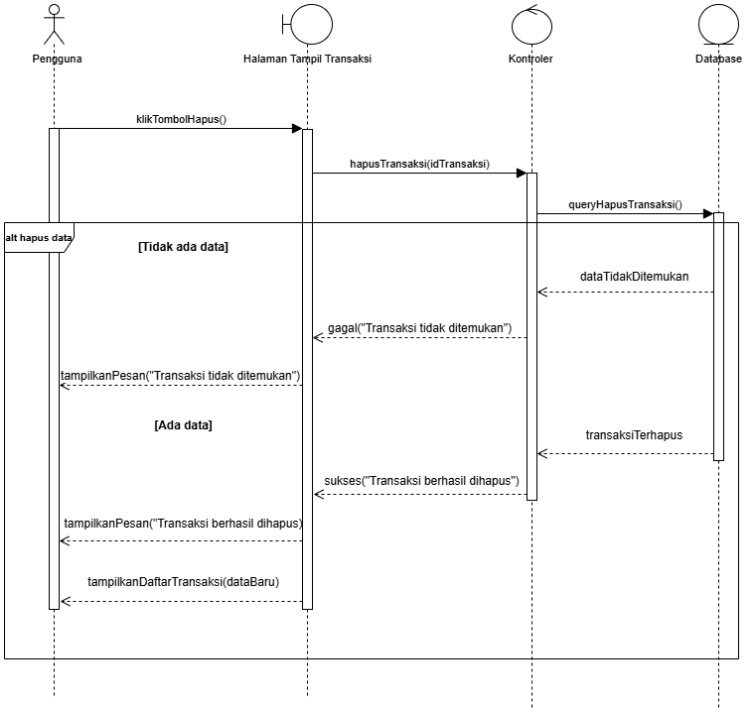

Figure 15. Sequence diagram of delete transaction.

7. Sequence Diagram of Show Monthly Transaction Chart

This sequence diagram describes the process when the user views the transaction chart. The user clicks the chart icon from the dashboard, then the system opens the chart page and requests data to the controller. The controller retrieves the data from the database. If there are no transactions, the system displays the message "No transactions". If the data is available, the current month's transaction graph will be displayed. The graph displayed is a bar chart that shows the comparison between income and expenses, making it easier for users to understand their monthly financial condition. This diagram emphasizes two possible conditions: the graph fails to display because the data is empty, or successfully displays if the data is available. The sequence diagram of the flow of displaying monthly transaction charts in Figure 16.

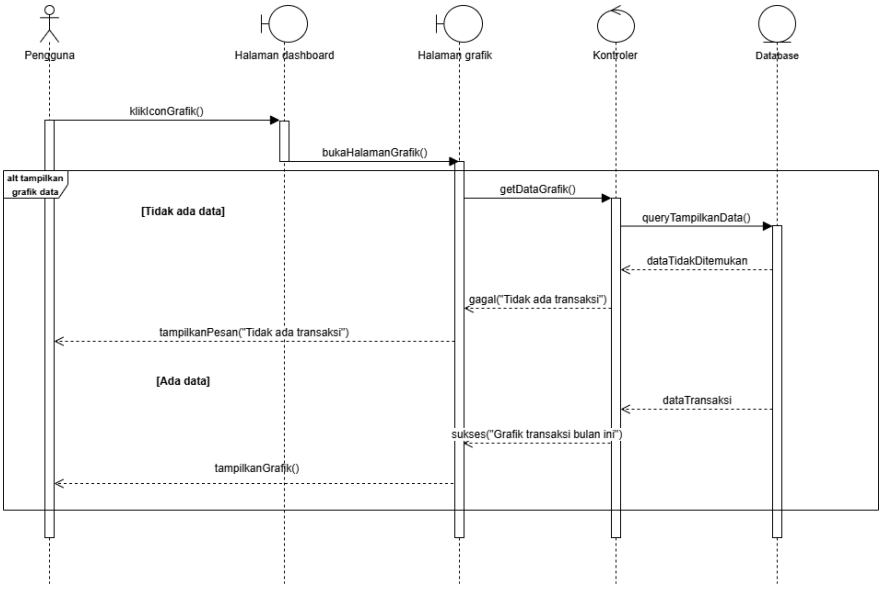

Figure 16. Sequence diagram of show monthly transaction chart.

8. Sequence Diagram Logout

This sequence diagram illustrates the flow of the logout process in an application system. The process starts when the user presses the logout button on the logout page. After that, the system sends a command to the controller to process the logout. At this stage, there are two possible flows. First, if the user cancels the logout process, the system will redisplay the previous page that the user is accessing. Second, if the user continues the logout process, the system will display the login page as a sign that the user has left the

system. This diagram shows the sequence of interactions between actors and system components in sequence according to the process that occurs. The logout flow sequence diagram in Figure 17.

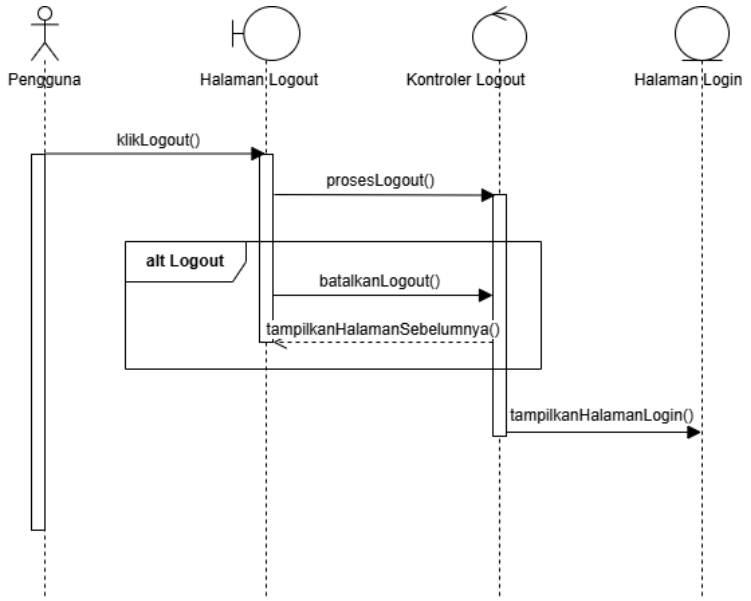

Figure 17. Sequence diagram logout.

Class Diagram

A class diagram is a visual representation of the structure and description of interrelated classes, packages, and objects, illustrating their connections, such as inheritance, association, and more [7]. The class diagram in Figure 18.

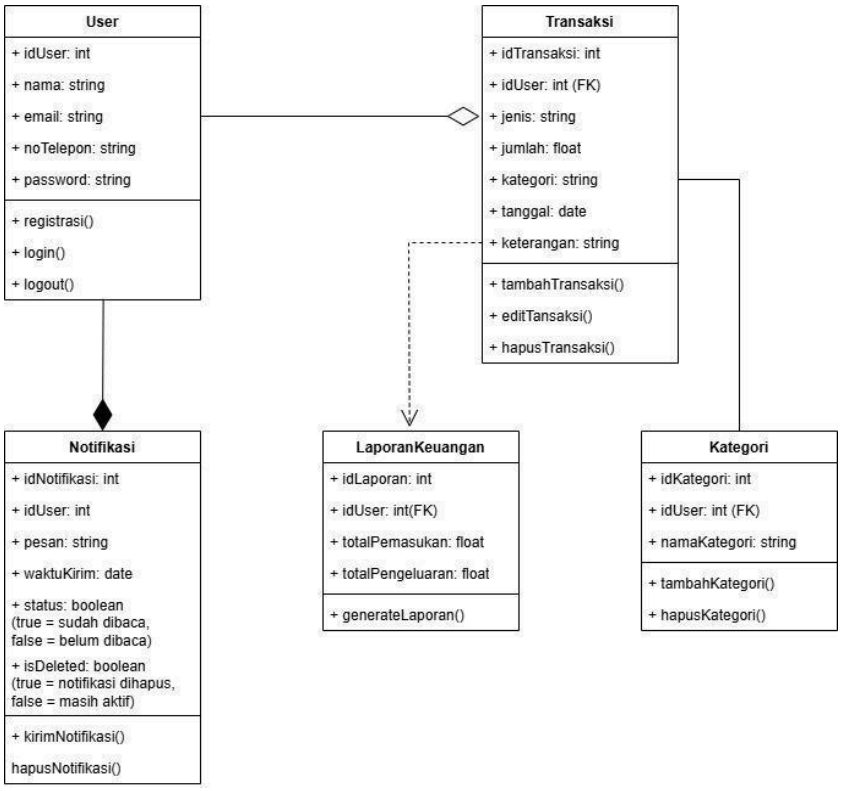

Figure 18. Class diagram.

This class diagram illustrates the structure of a web-based personal finance management system, comprising five main classes: User, Transaction, Category, FinancialReport, and Notification. The User class represents the application's users and includes functionalities for registration, login, and logout. The Transaction class stores data for user income and expenses. This class has an aggregation relationship with User, indicating that transactions depend on the existence of a user, but can logically persist even if the user object is removed from memory. The Category class allows users to group transactions by specific types (e.g., food, salary, transportation). The relationship between Transaction and Category is an association, as one transaction is linked to one category, and a category can be used by

multiple transactions. The FinancialReport class is used to calculate and display the user's total income and expenses. This class has a dependency relationship with Transaction, meaning that financial reports do not directly store transaction data but rely heavily on the information provided by transactions for their calculation processes. Meanwhile, the Notification class is responsible for sending messages or reminders to users regarding their financial activities. This class has a composition relationship with User, implying that notifications can only exist if the associated user exists and will be deleted if the user object is removed.

The relationships among these classes indicate that all transaction, category, report, and notification data are directly connected to the user. This system is designed to help users monitor and manage their finances more neatly, structured, and efficiently through a web-based application.

Implementation Results

The implementation results of this web-based financial management application present several pages designed to simplify the management process.

1. Homepage

   The homepage of the MoneyMate website features the website's logo, which depicts a wallet with a graph and a heart, reflecting the application's purpose of financial management. The application's name is highlighted on the page. The background design uses a gradient of green, pink, and orange colors. This homepage serves as the identity of the MoneyMate website. The layout of the homepage is shown in Figure 19.

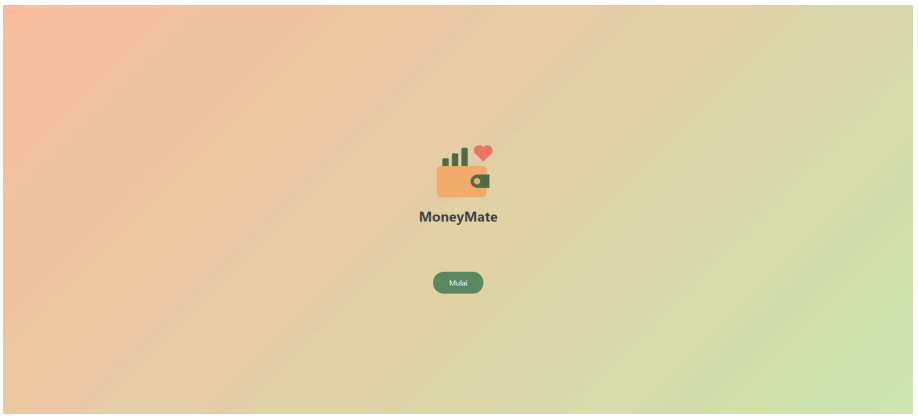

Figure 19. Homepage.

2. Login Page

   The login page presents an authentication form for users to enter the email address they registered during account creation. If a user hasn't registered an account yet, they must sign up first by clicking 'Daftar' (Sign Up) at the bottom. The login page is shown in Figure 20.

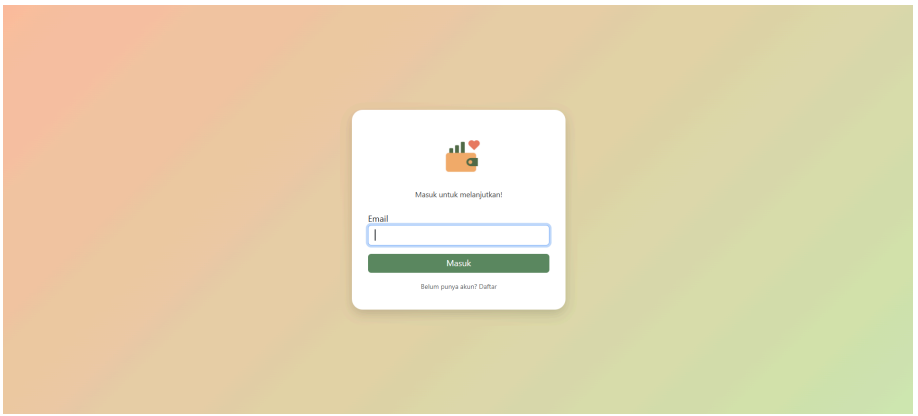

Figure 20. Login page.

3. Account Registration Page

   The account registration page displays a form for users to fill out during the new account creation process. The layout of the account registration page is shown in Figure 21.

Figure 21. Account registration page.

4. Dashboard

The user dashboard is the main page displayed after a user successfully logs in. On this page, users can view their latest transaction list, notifications, current balance, spending percentage, and a motivational quote. The user dashboard is shown in Figure 22.

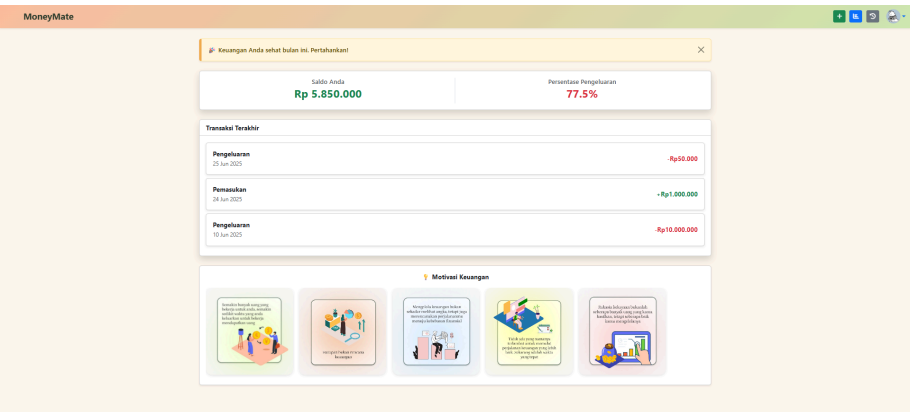

Figure 22. Dashboard.

5. Add Transaction Page

The Add Transaction page is designed for users to record their financial activities, whether it's income or expenses. This page is a core component of the finance management application, as it enables users to manage their finances by logging all financial movements. On this page, users will input the transaction date, type, amount, and a description. After filling in all the required fields, users can click the 'Simpan' button to store the new transaction. The layout of the Add Transaction page is shown in Figure 23.

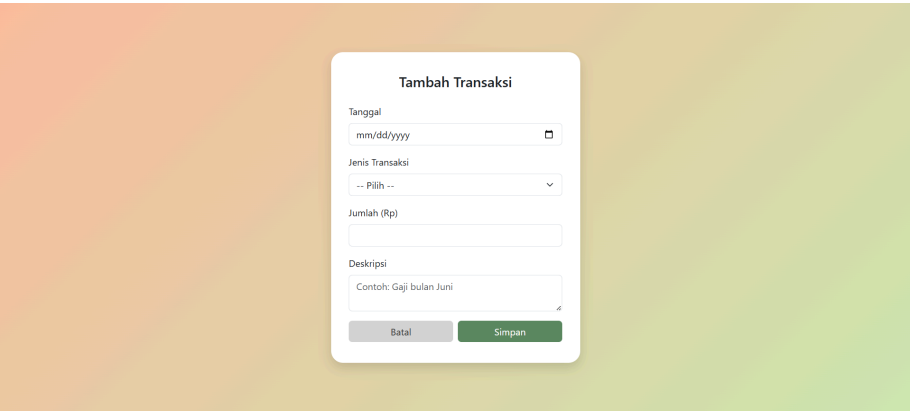

Figure 23. Add transaction page.

6. Edit Transaction Page

The Edit Transaction page is used to modify or update previously recorded financial activities, whether they are income or expenses. After making changes to the fields, users simply click the 'Simpan' button to store the updated data. The layout of the Edit Transaction page is shown in Figure 24.

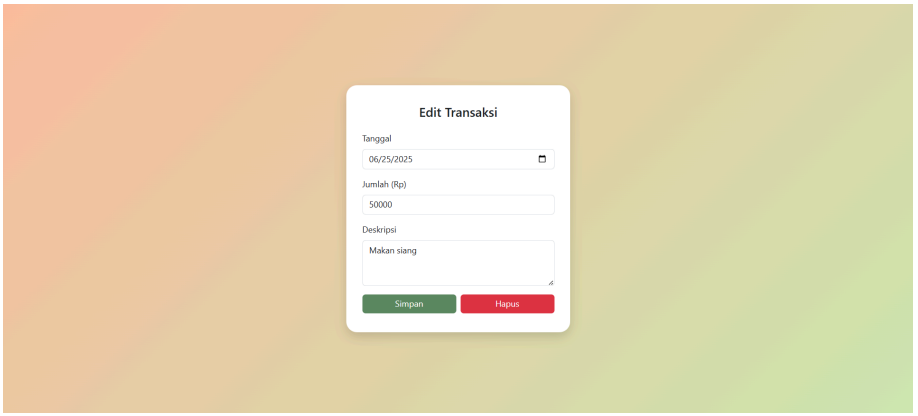

Figure 24. Edit transaction page.

7. Transaction Deletion Page

The transaction deletion page appears when a user clicks the "Hapus" button for a transaction in the list. The system then displays a confirmation dialog with a warning message, asking if the user genuinely wants to delete the transaction data. This feature is crucial for maintaining data integrity, preventing accidental deletions. The confirmation message, "Yakin ingin menghapus transaksi ini?", gives users an opportunity to cancel the action if they pressed the delete button by mistake. The transaction deletion process is shown in Figure 25.

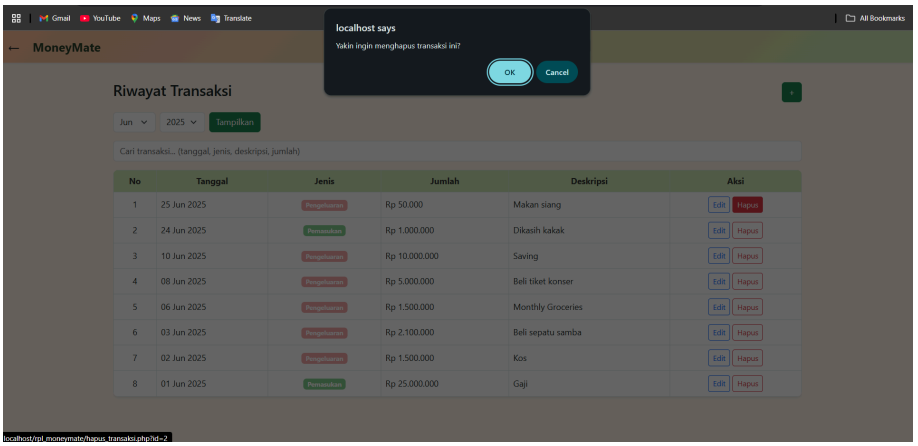

Figure 25. Transaction deletion page.

8. Transaction Display Page

The Transaction Display page shows all financial activities recorded by the user, including both income and expenses. This page provides detailed transaction data such as the date, transaction type, description, and amount recorded.

Additionally, users can search for transactions by keywords (date, type, description, amount) and filter by month and year to view transactions within a specific period. This feature allows users to efficiently review and manage their financial history. Each transaction entry also includes an "Edit" button to update information and a "Delete" button to remove data. The Transaction Display page is shown in Figure 26.

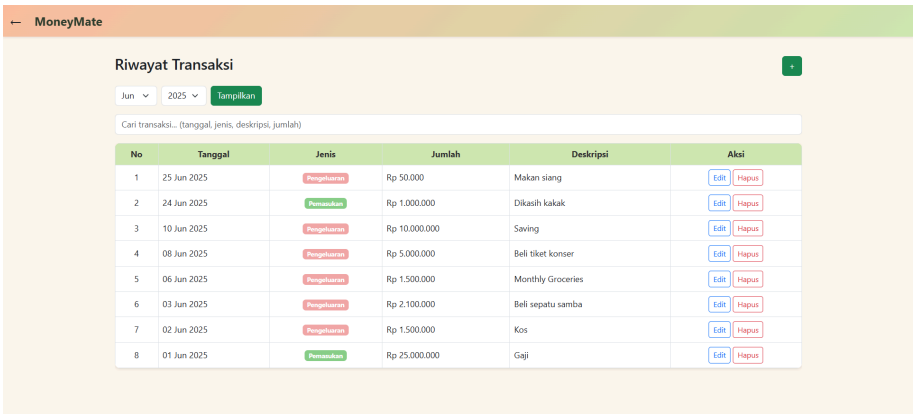

Figure 26. Transaction display page.

9. Financial Report Page

The Financial Report page offers users a comprehensive overview of their financial activities. It prominently features a bar chart that visually represents monthly income and expenses over a specific year, allowing users to quickly identify their financial trends. Below this graphical representation, a dedicated "Monthly Summary" section provides a detailed breakdown for the currently selected month. This summary typically includes the total income, total expenses, and the resulting net difference, enabling users to clearly understand their financial position for that period. This page enables users to monitor their financial performance effectively, both at a glance through the chart and in detail via the monthly summary. The Financial Report page is shown in Figure 27.

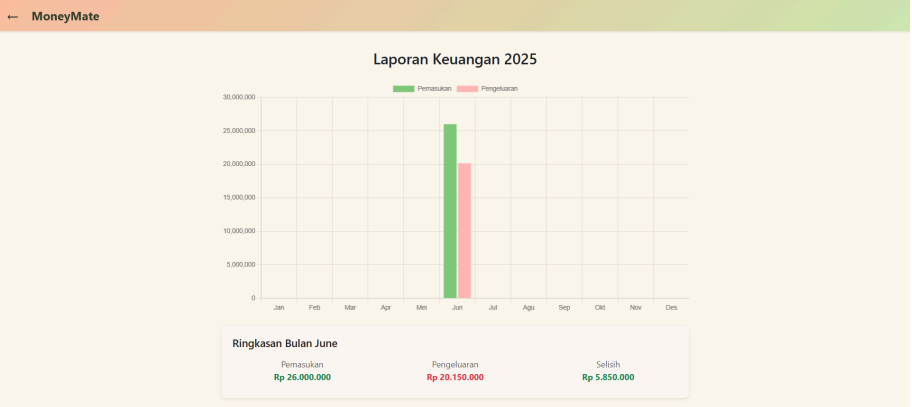

Figure 27. Financial report.

10. Profil Page

The Profile Page serves as a central hub for personal information and user account settings. At the top, it displays an avatar representing the user's profile picture, along with the user's name. The main section of the page contains several information fields that show the user's account details, specifically Name, Email, and Phone. Additionally, there is a toggle button labeled "Aktifkan Notifikasi Dashboard" (Activate Dashboard Notifications) which provides users with the flexibility to manage their notification preferences appearing on their homepage. To make changes to personal information, users can press the "Edit Profil" (Edit Profile) button available at the bottom. Meanwhile, the option to log out of the account is provided via the "Logout" button located in the top-right corner of the page. The appearance of the profile page is shown in Figure 28.

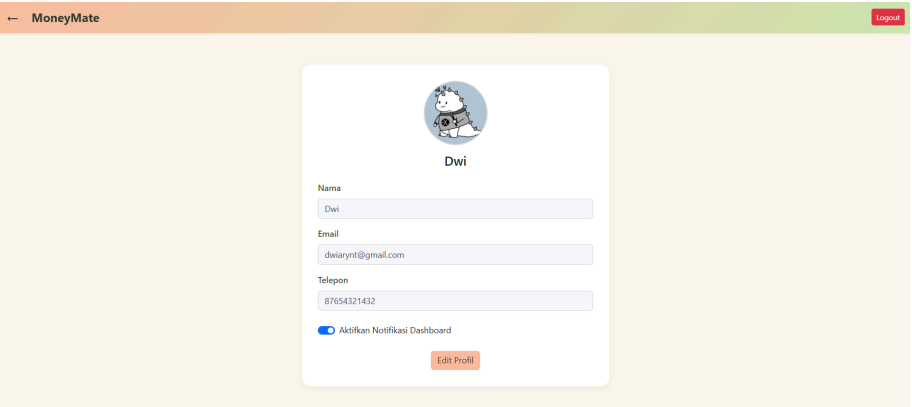

Figure 28. Profil page.

11. Edit Profile Page

The Edit Profile page allows users to update their personal information registered within the application. This page displays a form for Name, Email, and Phone Number, which are pre-filled with the user's current data. Additionally, there is a "Foto Profil" (Profile Photo) option with a "Choose File" button, enabling users to upload or change their profile picture. After making the desired changes, users can click the "Simpan" (Save) button to store the updates, or the "Batal" (Cancel) button to discard the changes and return to the previous page. The layout of the edit profile page is shown in Figure 29.

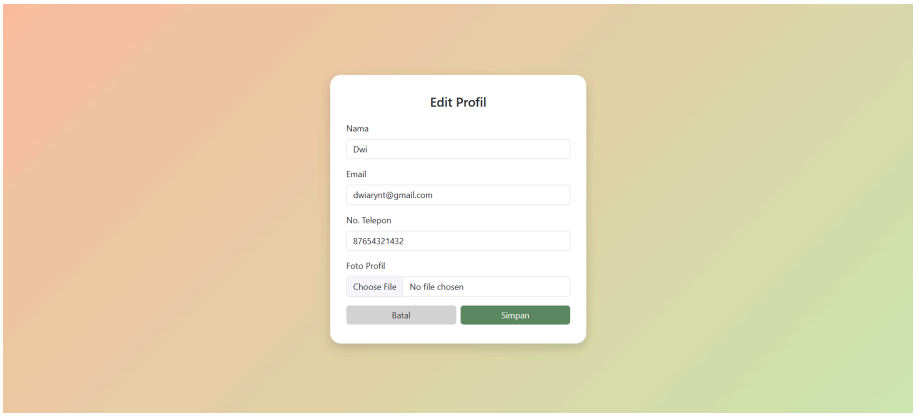

Figure 29. Edit profile page.

12. Logout Page

The logout page functions to terminate an active user session within the application. When a user clicks the "Logout" button, typically located in the top-right corner of the page, the system will display a logout confirmation page in the form of a dialog box or pop-up. This dialog presents a warning message stating, "Apakah Anda yakin ingin logout?" (Are you sure you want to log out?) to confirm the user's intent. Users are then given two options: pressing the "OK" button to proceed with the logout process, which will clear all active user session data and automatically redirect the user back to the login page or the application's main page; or pressing the "Cancel" button to abort the logout action and remain on the current page. This confirmation feature is specifically designed to prevent users from accidentally ending their session. The appearance of the logout confirmation page is shown in Figure 30.

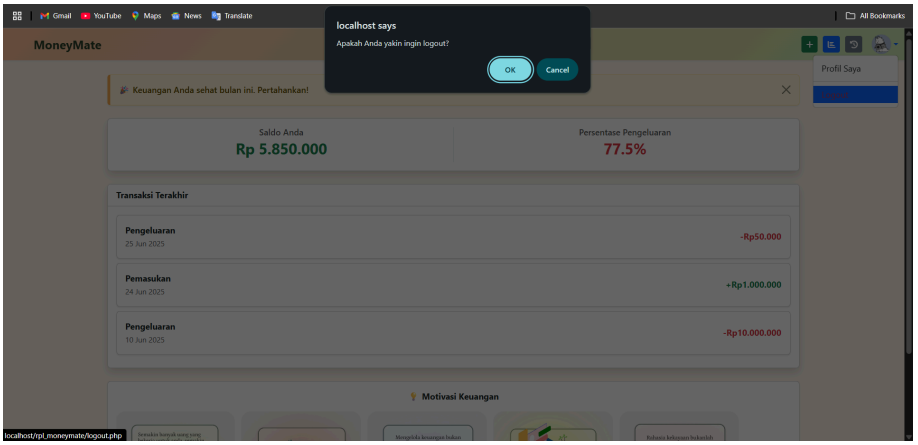

Figure 30. Logout page.

Program Testing

The implementation results of this web-based financial management application present several pages designed to simplify the management process.

Table 1. Testing the personal finance management website.

No	Function	Scenario	Expected Result	Output	Outcome
1	Login	Input correct email	Successfully logged in	Successfully logged in	Success
2	Dashboard	Access the dashboard page	Successfully displayed the dashboard page	Successfully displayed the dashboard page	Success
3	Transaction Display Page	Access the transaction page	Successfully displayed the transaction page	Successfully displayed the transaction page	Success
4	Add, Edit, Delete Transaction	Successfully add, edit, and delete transaction data in the transaction table	Successfully added, edited, and deleted transaction data in the transaction table	Successfully added, edited, and deleted transaction data in the transaction table	Success
5	Financial Report	Access the financial	Successfully displayed	Successfully displayed	Success

	Page	report page	the financial report page	the financial report page	
6	Profile Page	The user accesses their profile page and toggles the notification settings (e.g., turns notifications on or off). The change is saved.	The profile page is successfully displayed, and the notification setting is updated and reflected correctly.	The profile page is successfully displayed, and the notification setting is updated and reflected correctly.	Success
7	Edit Profile	The user navigates to the "Edit Profile" , makes changes to their personal information (e.g., name, email), and saves them.	The "Edit Profile" is displayed, and the updated profile information is successfully saved and reflected.	The "Edit Profile" is displayed, and the updated profile information is successfully saved and reflected.	Success
8	Logout	The user clicks the "Logout" button	The user is successfully logged out of their account and redirected to the login or home page.	The user is successfully logged out of their account and redirected to the login or home page.	Success

Conclusion

The development of the MoneyMate website as a personal financial management platform was carried out using a design thinking approach, encompassing the stages of empathy, definition, ideation, prototyping, and thorough testing. This process resulted in a web application equipped with features such as transaction recording, financial reporting, notifications, and a user-friendly interface. The successful development of this application is expected to enhance users' awareness in managing their finances effectively, while also serving as a foundation for future development of similar applications.

For future enhancements, the application holds potential for integrating artificial intelligence technologies to provide predictive, personalized financial analysis and recommendations.

References

[1] Akbar, I. S., & Haryanti, T. (2021). Pengembangan Entity Relationship Diagram Database Toko Online Ira Surabaya. *Jurnal Ilmiah Computing Insight*, *3*(2), 28–35. https://doi.org/10.30651/comp_insight.v3i2.12002.

[2] Billah, A. M. S., Fikriyah, M., Anisa, S., & Sutriyono. (2023). Perancangan Sistem Manajemen Keuangan Berbasis Web di CV. Tigana Sukses. *JORAPI : Journal of Research and Publication Innovation*, *1*(3), 727–735.

[3] Dewi, N. L. P. K., Agus, W. S. G., & Astiti, N. P. Y. (2021). Pengaruh Literasi Keuangan, Gaya Hidup Hedonisme, dan Pendapatan terhadap Pengelolaan Keuangan Mahasiswa UNMAS. *Jurnal Emas*, *2*(3), 74–85.

[4] Muaziz, I., Utomo, F. S., Krisbiantoro, D., & Setiawan, I. (2024). Desain Aplikasi Mobile Smart Farming dengan Pendekatan Design Thinking untuk Meningkatkan Produktivitas Pertanian. *JUSTIN (Jurnal Sistem Dan Teknologi Informasi)*, *12*(2), 338–344. https://doi.org/10.26418/justin.v12i1.75319.

[5] Narulita, S., Nugroho, A., & Abdillah, M. Z. (2024). Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS). *BRIDGE : Jurnal Publikasi Sistem Informasi dan Telekomunikasi*, *2*(3), 244–256. https://doi.org/10.62951/bridge.v2i3.174.

[6] Pratama, J., & Lestari, S. (2023). Web-Based Financial Management Systems: Security and Usability. *Journal of Information Security*, 18(1), 89-102.

[7] Ramdany, S. W., Kaidar, S. A., Aguchino, B., Putri, C. A. A., & Anggie, R. (2024). Penerapan UML Class Diagram dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web. *Journal of Industrial and Engineering System (JIES)*, *5*(1), 30–41. https://doi.org/10.31599/2e9afp31.

[8] Restyasari, N., Fuada, S., & Setyowati, E. (2023). Redesign UI/UX Aplikasi SH-UPI Menggunakan Kerangka Kerja Design Thinking. *Jurnal Pendidikan Teknologi Informasi (JUKANTI)*, *6*(2), 152–166.

[9] Sari, N. R., & Listiadi, A. (2021). Pengaruh Literasi Keuangan, Pendidikan Keuangan di Keluarga, Uang Saku terhadap Perilaku Pengelolaan Keuangan dengan Financial Self-Efficacy sebagai Variabel Intervening. *Jurnal Pendidikan Akuntansi (JPAK)*, *9*(1), 58–70. https://doi.org/10.26740/jpak.v9n1.p58-70.

[10] Setyasih, D., Santoso, S., Wahyuni, I., & Rahayu, E. F. (2024). Aplikasi Sistem Informasi Pencatatan Keuangan Berbasis Web Menggunakan Metode Rapid Application Development. *ICIT Journal*, *10*(1), 71–82.

[11] Fahrudin, R., & Ilyasa, R. (2021). Perancangan aplikasi “Nugas” menggunakan metode Design Thinking dan Agile Development. *JITTER : Jurnal Ilmiah Teknologi Informasi Terapan*, 8(1), 35-44. https://journal.widyatama.ac.id/index.php/jitter/article/view/714/547

[12] Widiyanti, Widyaswati, R., & Meiriyanti, R. (2020). Peran Pencatatan Keuangan Harian sebagai Filter Diri dalam Pengaturan Pengeluaran Pribadi. *Dinamika Sosial Budaya*, *22*(2), 258–271.