

---

**WalletWise Team**

---

**WalletWise**  
**Use-Case Specification**

**Version 1.1**

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

## Revision History

Date	Version	Description	Author
03/May/24	1.0	First version of the Use-Case Specification	WalletWise Team
16/May/24	1.1	Add alternative flows for the use cases: Add new expenses, Enter expense manually, Scan with OCR, Enter with text.	WalletWise Team

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

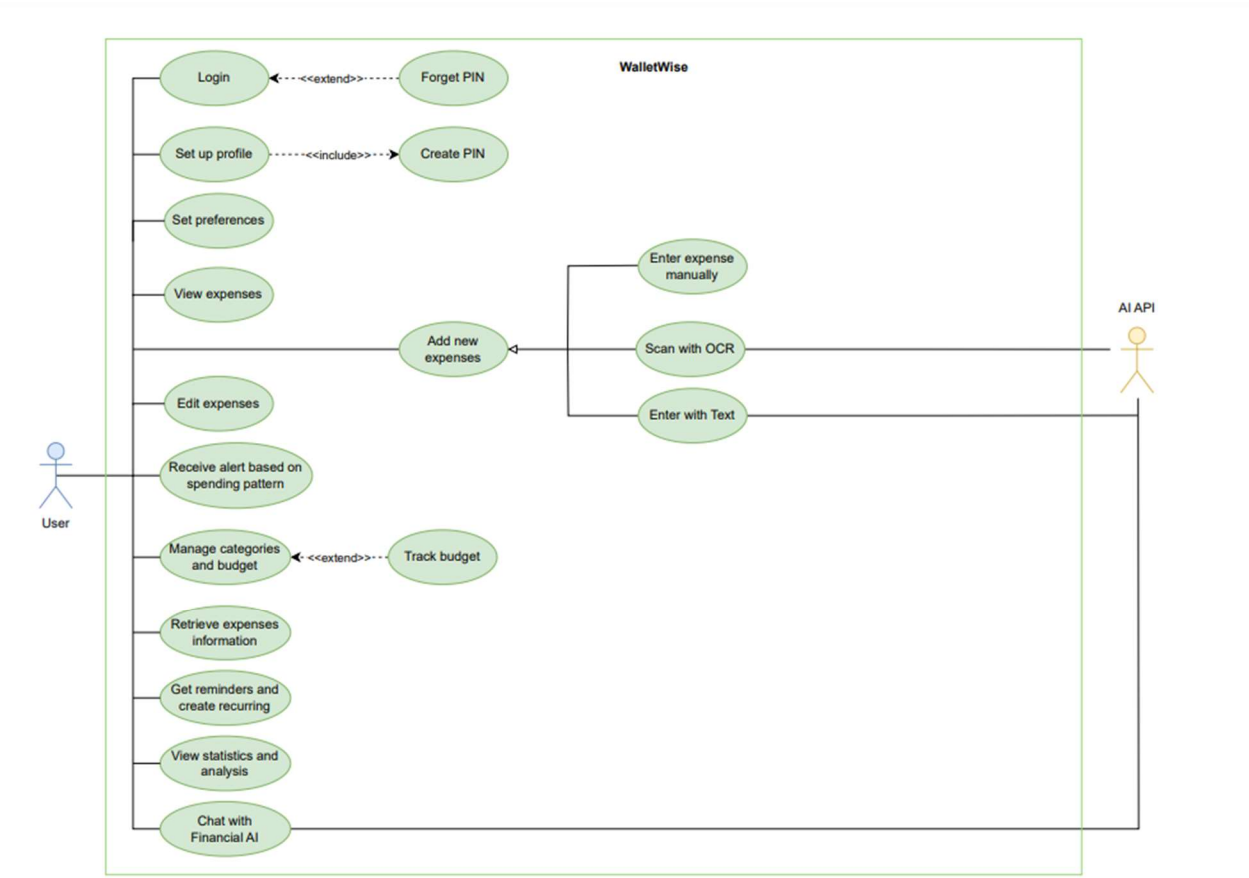
# Table of Contents

- 1. Use-case Model ..... 4**
- 2. Use-case Specifications ..... 4**
  - 2.1 Use-case: Log in .....4
  - 2.2 Use-case: Forget PIN .....5
  - 2.3 Use-case: Set up profile .....5
  - 2.4 Use-case: Create PIN.....6
  - 2.5 Use-case: Set preferences .....7
  - 2.6 Use-case: View expenses .....7
  - 2.7 Use-case: Edit expenses .....8
  - 2.8 Use-case: Add new expenses .....8
  - 2.9 Use-case: Enter expense manually .....9
  - 2.10 Use-case: Scan with OCR ..... 10
  - 2.11 Use-case: Enter with text..... 11
  - 2.12 Use-case: Receive alert based on spending patterns ..... 12
  - 2.13 Use-case: Manage categories and budgets ..... 12
  - 2.14 Use-case: Track budgets..... 13
  - 2.15 Use-case: Retrieve expenses information ..... 14
  - 2.16 Use-case: Get reminder and create recurring ..... 15
  - 2.17 Use-case: View statistics and analysis ..... 16
  - 2.18 Use-case: Chat with the financial AI assistant ..... 16

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

# Use-Case Specification

## 1. Use-case Model



## 2. Use-case Specifications

### 2.1 Use-case: Log in

Use case Name	Log in.
Brief description	This use case describes how the user logs into the application using valid credentials.
Actors	User
Basic Flow	<ol style="list-style-type: none"><li>1. The user clicks the WalletWise application icon.</li><li>2. The system displays the login page presenting the login form requesting the PIN.</li><li>3. The user enters a valid PIN.</li><li>4. System verifies the entered credentials against stored user data.</li><li>5. If the credentials are valid and match existing records, the system grants access to the corresponding interface.</li></ol>

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

Alternative Flows	<p><b>Alternative flow 1: The user forgets PIN</b></p> <ol style="list-style-type: none"> <li>From #2 of the basic flow, the user clicks on the 'Forget PIN' button.</li> <li>The user will be navigated to the page and asked for the information.</li> </ol> <p><b>Alternative flow 2: The user enters incorrect PIN</b></p> <ol style="list-style-type: none"> <li>The system displays an error message stating, "Invalid PIN".</li> <li>From #3 of the basic flow, the user is prompted to re-enter PIN.</li> <li>Continue #4 in the basic flow.</li> </ol>
Pre-conditions	The user must have set up and created a PIN.
Post-conditions	The user successfully logs in and gains access to the application homepage.

## 2.2 Use-case: Forget PIN

Use case Name	Forget PIN.
Brief description	This use case describes the process for a user who has forgotten their PIN to regain access to the application.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>The user clicks on the "Forgot PIN" button.</li> <li>System navigates the user to the PIN recovery page.</li> <li>The user provides the necessary information.</li> <li>The system verifies the provided information against the existing user database.</li> <li>The user enters a new PIN.</li> <li>The user successfully changed the PIN.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The user provides incorrect information</b></p> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the system displays an error message indicating that the provided information is incorrect.</li> <li>The user is requested to re-enter the necessary information.</li> <li>Continue #4 in the basic flow.</li> </ol>
Pre-conditions	The user has forgotten their PIN.
Post-conditions	The user successfully regains access to the application.

## 2.3 Use-case: Set up profile

Use case Name	Set up profile.
Brief description	This use case describes the process for the user to set up their profile including personal details such as name, gender, currency.

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>1. The user opens the WalletWise application for the first time.</li> <li>2. The user enters their name in the designated field.</li> <li>3. The user selects their gender from the available options (e.g., Male, Female, Other).</li> <li>4. The user selects their preferred currency from the available options (e.g., USD, EUR, GBP, VND).</li> <li>5. The user saves the changes to their profile.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The user enters missing information</b></p> <ol style="list-style-type: none"> <li>1. From #4 of the basic flow, the system displays an error message indicating that the provided information is incorrect.</li> <li>2. The user is requested to re-enter the necessary information.</li> <li>3. Continue #4 in the basic flow.</li> </ol>
Pre-conditions	The user opens the application for the first time, or they have reset the application.
Post-conditions	The user's profile is successfully created.

## 2.4 Use-case: Create PIN

Use case Name	Create PIN.
Brief description	This use case outlines the process for a user to create a new PIN for their account.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>1. The user navigates to the "Create PIN" section after setting up their profile.</li> <li>2. The user enters the desired PIN.</li> <li>3. The user confirms the new PIN.</li> <li>4. The user selects pre-set security questions and answers them for authentication when they forget their PIN.</li> <li>5. System saves the PIN and updates the user's information accordingly.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The user's confirmed PIN doesn't match the initial entry</b></p> <ol style="list-style-type: none"> <li>1. From #3 of the basic flow, the system displays an error message indicating the mismatch.</li> <li>2. The user is requested to re-enter the new PIN and confirm it again.</li> <li>3. Continue #3 of the basic flow.</li> </ol> <p><b>Alternative flow 2: The user doesn't enter security question</b></p> <ol style="list-style-type: none"> <li>1. From #5 of the basic flow, the system displays an error message indicating that some information fields are missing.</li> <li>2. The user is requested to enter missing information.</li> <li>3. Continue #5 of the basic flow.</li> </ol>

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

Pre-conditions	The user has set up their accounts.
Post-conditions	The user successfully creates a PIN, enabling them to access their application using the PIN.

## 2.5 Use-case: Set preferences

Use case Name	Set preferences.
Brief description	This use case describes how the user can configure the application preferences.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>At the application homepage, the user selects the "Set Preferences" feature.</li> <li>The application displays available settings for configuring application preferences (e.g., user's name, gender; application features, ...).</li> <li>The user selects and specifies specific options for the preferences they want to configure.</li> <li>Upon completing the configuration, the user confirms and saves the changes.</li> <li>The system validates and stores the user's preference settings.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The user decides not to configure anything</b></p> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the user selects the option to cancel the process.</li> <li>The system returns the user to the homepage.</li> </ol> <p><b>Alternative flow 1: The user configures invalid or incomplete information</b></p> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the system displays error messages indicating the fields that need correction.</li> <li>The user corrects the required fields.</li> <li>Continue step #6 in the basic flow.</li> </ol>
Pre-conditions	The user must have a valid PIN to log into the application.
Post-conditions	The user successfully sets their preferences in the application.

## 2.6 Use-case: View expenses

Use case Name	View expenses.
Brief description	This use case describes how the user views their expenses within the application.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>The user selects the "View expenses" feature.</li> <li>The system switches to another screen displaying a list of the user's expenses.</li> <li>The user views detailed information about the expenses.</li> </ol>
Alternative Flows	<b>Alternative flow 1: No expenses recorded</b>

<b>WalletWise</b>	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

	<ol style="list-style-type: none"> <li>From #3 of the basic flow, the system displays a message indicating that there are no expenses to track.</li> <li>The user can choose to go back to the main interface or navigate to other features of the application.</li> </ol>
Pre-conditions	The user goes to the homepage.
Post-conditions	The user has successfully viewed expense information.

## 2.7 Use-case: Edit expenses

Use case Name	Edit expenses.
Brief description	This use describes how the user edits/deletes expenses within the application.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>The user selects the "View expense" feature.</li> <li>The system switches to another screen displaying a list of the user's expenses.</li> <li>The system displays an interface allowing the user to edit or delete the selected expense.</li> <li>The user edits or deletes the expense.</li> <li>The system updates the corresponding data.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The user wants to cancel the editing/deleting process</b></p> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the user can select the "Cancel" option or exit the editing or deleting interface.</li> <li>System displays a confirmation dialog box, requesting confirmation to cancel the editing or deleting process.</li> <li>The user confirms the cancellation request, and the system returns #3.</li> </ol>
Pre-conditions	The user goes to the homepage.
Post-conditions	The user has successfully edited or deleted expense information.

## 2.8 Use-case: Add new expenses

Use case Name	Add expenses.
Brief description	This use case describes how users add expenses into the application manually.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>The user selects the plus sign symbol.</li> <li>The system switches to another screen that allows users to enter new expenses.</li> <li>Users can choose from three types of additional expenses: Enter expense manually, Scan receipt with OCR, enter expenses using text.</li> <li>The system verifies and saves the new expense information.</li> </ol>



WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

Alternative Flows	<b>Alternative flow 1: Adding a new category during expense entry</b> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the user selects the option to add a new category.</li> <li>The user enters the category name and confirms</li> <li>The system validates the changes and updates the category list.</li> </ol>
Pre-conditions	The user goes to the homepage.
Post-conditions	The user has successfully entered new expenses information into the application.

## 2.9 Use-case: Enter expense manually

Use case Name	Enter manual expense.
Brief description	This use case describes how users input expenses into the application manually.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>The user selects the plus sign symbol.</li> <li>The system switches to another screen that allows users to enter new expenses.</li> <li>System displays an interface allowing the user to add a new expense, including options for date, expense category, and amount.</li> <li>The system verifies and saves the new expense information.</li> </ol>
Alternative Flows	<b>Alternative flow 1: Missing required or invalid Information</b> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the system displays an error message and asks the user to re-enter.</li> <li>The user re-enters the invalid information or fills in all required information.</li> <li>Continue #4.</li> </ol> <b>Alternative flow 2: The user wants to cancel the entry process</b> <ol style="list-style-type: none"> <li>From #3 of the basic flow, they can select the "Cancel" option or exit the input interface.</li> <li>The system displays a confirmation message requesting confirmation to cancel the expense adding process.</li> <li>The user confirms the cancellation request and the system return #3.</li> </ol> <b>Alternative flow 3: Adding a new category during expense entry</b> <ol style="list-style-type: none"> <li>From #3 of the basic flow, the user selects the option to add a new category.</li> <li>The user enters the category name and confirms</li> <li>The system validates the changes and updates the category list.</li> <li>Continue #4</li> </ol>
Pre-conditions	The user goes to the homepage.
Post-conditions	The user has successfully entered new expenses information into the application.

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

## 2.10 Use-case: Scan with OCR

Use case Name	Scan with OCR.
Brief description	This use case describes how a user scans a receipt into the application with Optical Character Recognition (OCR) technology to automatically extract information from the receipt.
Actors	User and AI API
Basic Flow	<ol style="list-style-type: none"> <li>1. The user selects the "add expenses" symbol.</li> <li>2. The system switches to another screen that allows users to enter new expenses.</li> <li>3. The user selects the "Scan Receipts with OCR" feature.</li> <li>4. The system opens the mobile device's camera.</li> <li>5. The user places the receipt under the camera and takes a photo.</li> <li>6. The system uses OCR technology to recognize text on the receipt.</li> <li>7. The system automatically extracts information from the receipt and fills in the expense's information automatically.</li> <li>8. System displays an interface that allows the user to review the extracted information to ensure accuracy.</li> <li>9. The user reviews the extracted information and can optionally edit it, then confirms it.</li> <li>10. The system verifies and saves the new expense information.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The receipt is unclear or blurry</b></p> <ol style="list-style-type: none"> <li>1. From #7 of the basic flow, if the receipt is unclear or blurry due to poor lighting or low image quality, the system displays a notification message asking the user to capture the receipt again with better lighting or from a better angle.</li> <li>2. The user can choose to retry scanning the receipt or exit the scanning interface.</li> <li>3. If the user decides to capture the receipt again, continue #7.</li> </ol> <p><b>Alternative flow 2: The user wants to cancel the scanning process</b></p> <ol style="list-style-type: none"> <li>1. From #5 of the basic flow, the user selects the "Cancel" option or exits the scanning interface.</li> <li>2. The system displays a confirmation message requesting confirmation to cancel the scanning process.</li> <li>3. The user confirms the cancellation request and returns to the main page without making any changes.</li> </ol> <p><b>Alternative flow 3: Missing required or invalid Information</b></p> <ol style="list-style-type: none"> <li>1. From #10 of the basic flow, the system displays an error message and asks the user to re-enter.</li> <li>2. The user re-enters the invalid information or fills in all required information.</li> <li>3. Continue #10.</li> </ol> <p><b>Alternative flow 4: Adding a new category during expense entry</b></p> <ol style="list-style-type: none"> <li>1. From #9 of the basic flow, the user selects the option to add a new category.</li> </ol>

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

	<ol style="list-style-type: none"> <li>The user enters the category name and confirms</li> <li>The system validates the changes and updates the category list.</li> <li>Continue #10</li> </ol>
Pre-conditions	The user goes to the homepage.
Post-conditions	The user has successfully entered new expenses information into the application.

## 2.11 Use-case: Enter with text

Use case Name	Enter with text.
Brief description	This use case describes how the user input expenses into the application by manually entering text, allowing the system to process and categorize expenses based on the provided information.
Actors	User and AI API
Basic Flow	<ol style="list-style-type: none"> <li>The user selects the plus sign symbol.</li> <li>The user selects the "Enter expenses using text" feature.</li> <li>The system displays a text input field for the user to enter expense details.</li> <li>The system processes the entered text to extract relevant expense details and categorizes them accordingly.</li> <li>The system displays an interface that allows users to review the extracted information to ensure accuracy.</li> <li>The user reviews the extracted information and can optionally edit it, then confirms it.</li> <li>The system verifies and saves expense details to the list.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The user wants to cancel the entry process</b></p> <ol style="list-style-type: none"> <li>From #5 of the basic process, if the user wants to cancel the expense entry process, they can select the "Cancel" option or exit the text input interface.</li> <li>The system displays a confirmation message requesting confirmation to cancel the expense entry process.</li> <li>The user confirms the cancellation request and returns to the main page without making any changes.</li> </ol> <p><b>Alternative flow 2: Missing required or invalid Information</b></p> <ol style="list-style-type: none"> <li>From #7 of the basic flow, the system displays an error message and asks the user to re-enter.</li> <li>The user re-enters the invalid information or fills in all required information.</li> <li>Continue #7.</li> </ol> <p><b>Alternative flow 3: Adding a new category during expense entry</b></p> <ol style="list-style-type: none"> <li>From #6 of the basic flow, the user selects the option to add a new category.</li> <li>The user enters the category name and confirms</li> <li>The system validates the changes and updates the category list.</li> <li>Continue #7</li> </ol>

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

Pre-conditions	The user goes to the homepage.
Post-conditions	The user has successfully entered new expenses information into the application.

## 2.12 Use-case: Receive alert based on spending patterns

Use case Name	Receive alert based on spending patterns.
Brief description	This use case describes how the user can receive alerts from the system when the projected expenditure exceeds the allowed threshold.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>1. System displays the alert.</li> <li>2. The user clicks on the “Ok” button to ignore the alert.</li> </ol>
Alternative Flows	<b>Alternative flow 1: The user wants to view statistics and analyzes</b> <ol style="list-style-type: none"> <li>1. From #2 of the basic flow, the user clicks on the “View” button to view details.</li> <li>2. System navigates to the statistics and analysis page.</li> </ol>
Pre-conditions	The user enables the feature in the set preferences page.
Post-conditions	The user successfully receives an alert from the system and views it.

## 2.13 Use-case: Manage categories and budgets

Use case Name	Manage categories and budgets.
Brief description	This use case describes how the user can categorize expenses and set monthly budgets for each category.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>1. At the application homepage, the user selects the “Manage Categories and Budgets” feature.</li> <li>2. At the "Manage Category and Budget" page, the user views existing categories and their assigned monthly budgets.</li> <li>3. The user manages the category and its assigned monthly budgets.</li> <li>4. The system validates the changes and updates the category and budget management accordingly.</li> </ol>
Alternative Flows	<b>Alternative flow 1: The user adds or edits a category and its assigned monthly budgets</b> <ol style="list-style-type: none"> <li>1. From #3 of the basic flow, the user fills in the name and monthly budgets for each category in the Category and Budget Management interface.</li> <li>2. The user saves or applies the modifications.</li> </ol>

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

	<p>3. Continue step #4 in the basic flow.</p> <p><b>Alternative flow 2: The user adds or edits a category and its assigned monthly budgets</b></p> <ol style="list-style-type: none"> <li>1. From #3 of the basic flow, the user selects the category or categories to be deleted from the catalog.</li> <li>2. The user confirms the deletion action.</li> <li>3. Continue step #4 in the basic flow.</li> </ol> <p><b>Alternative flow 3: The user enters invalid or incomplete information</b></p> <ol style="list-style-type: none"> <li>1. From #4 of the basic flow, the system displays error messages indicating the fields that need correction.</li> <li>2. The user corrects the required fields and resubmits the updated details.</li> <li>3. Continue step #4 in the basic flow.</li> </ol>
Pre-conditions	The user goes to the homepage.
Post-conditions	The user successfully manages categories and budgets within the application, including creating, editing, or deleting categories and their assigned monthly budgets, with changes validated and updated by the system accordingly.

## 2.14 Use-case: Track budgets

Use case Name	Track budgets.
Brief description	This use case describes a system feature wherein users' expenses are monitored against their designated monthly budgets. It alerts users when expenses surpass predetermined budget thresholds, facilitating effective budget management.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>1. The user begins recording expenses within each category.</li> <li>2. The alert is displayed to the user, indicating the category exceeded.</li> <li>3. User acknowledges the alert.</li> <li>4. The user clicks on the “Ok” button to ignore the alert.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The user has not set a monthly budget for a specific category</b></p> <ol style="list-style-type: none"> <li>1. From #1 of the basic flow, the system notifies the user about the absence of a predefined budget for that category.</li> <li>2. The user is informed that expenses in that category will not be monitored.</li> <li>3. Continues step #1 in the basic flow.</li> </ol> <p><b>Alternative flow 2: The user decides to update the monthly budget for a category.</b></p> <ol style="list-style-type: none"> <li>1. From #2 of the basic flow, the user selects the option to modify the budget for a specific category.</li> <li>2. The user enters the updated budget value.</li> <li>3. The system validates the entered budget value.</li> <li>4. If the entered value is invalid (e.g., non-numeric or negative), the system displays an error message.</li> </ol>

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

	<ol style="list-style-type: none"> <li>If the entered value is valid, the system proceeds to update the budget.</li> <li>The system confirms the successful update of the budget.</li> <li>Continues step #2 in the basic flow.</li> </ol>
Pre-conditions	<p>The user selects the option to track expenses.</p> <p>The user must have set monthly budgets for at least one expense category.</p>
Post-conditions	The user's expenses are successfully tracked against their set monthly budget and alerts are provided when expenses exceed predefined budget limits.

## 2.15 Use-case: Retrieve expenses information

Use case Name	Retrieve expenses information.
Brief description	Retrieves expenses information from bank/wallet transaction notifications and suggests users to input their income and expense details.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>The application retrieves money transaction notifications from the user's bank/wallet application.</li> <li>The application presents the user with a suggestion to input the expense details, category, including reviewing the amount.</li> <li>The user reviews the suggestions and input the expense details.</li> <li>The user confirms and the application saves the information to the user's expense history.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: The user ignores the suggestion</b></p> <ol style="list-style-type: none"> <li>From #3 of the basic flow, the user chooses to ignore the application's suggestion to add a new expense record based on a bank/wallet transaction.</li> <li>The application deletes the notification about that suggestion.</li> </ol> <p><b>Alternative flow 2: Incorrect Transaction Amount Identified</b></p> <ol style="list-style-type: none"> <li>From #2 of the basic flow, the application retrieves the wrong transaction's amount.</li> <li>The user manually corrects the transaction's amount.</li> <li>Continue #3 of the basic flow.</li> </ol>
Pre-conditions	<p>The user must have supported bank/wallet applications.</p> <p>The user must have enabled transaction notifications for their bank/wallet application and the application's transaction retriever feature.</p>
Post-conditions	The user's expense history is updated with the new expense information.

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

## 2.16 Use-case: Get reminder and create recurring

Use case Name	Get reminder and create recurring.
Brief description	This use case describes how the user can create, edit and delete a reminder for recurring expenses and get a reminder from the system.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>At the homepage, the user selects “Reminder and recurring” features.</li> <li>The user chooses to add a new reminder.</li> <li>System displays a form for users to fill in information about the reminder: <ul style="list-style-type: none"> <li>Expiry Date: User selects the expiry date of the recurring expense.</li> <li>Repeat Interval: User selects how often the expense repeats (e.g., weekly, monthly, yearly).</li> <li>Notification Interval: The user selects how many days before the expiry date they want to be notified (e.g., 2 days).</li> <li>Amount: User enters the amount to be paid.</li> <li>Description (optional): User may enter a description for the expense.</li> </ul> </li> <li>System verifies.</li> <li>The user confirms and the system adds a new reminder to list and stores the information.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: Missing required or invalid Information</b></p> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the system displays an error message and asks the user to re-enter.</li> <li>The user re-enters the invalid information or fills in all required information.</li> <li>Continue #4.</li> </ol> <p><b>Alternative flow 2: Edit reminders</b></p> <ol style="list-style-type: none"> <li>From #2 of the basic flow, the user chooses the reminder to edit.</li> <li>Continue #3.</li> </ol> <p><b>Alternative flow 3: Delete reminder</b></p> <ol style="list-style-type: none"> <li>From #2 of the basic flow, the user clicks on the “Delete” button on the reminder to remove it.</li> <li>System displays a notification to verify the action of the user.</li> <li>The user agrees to remove the reminder or not.</li> <li>System remove reminder and delete it from list.</li> </ol>
Pre-conditions	The user goes to the homepage.
Post-conditions	The user successfully adds, edits or deletes a reminder.

WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

## 2.17 Use-case: View statistics and analysis

Use case Name	View statistics and analysis.
Brief description	This use case describes how the system displays monthly and yearly financial charts and reports, along with providing detailed analysis reports on expense items and expense trends.
Actors	User
Basic Flow	<ol style="list-style-type: none"> <li>At the application homepage, the user selects “Statistics and Analysis” features.</li> <li>The user selects the desired timeframe (monthly or yearly).</li> <li>The system shows monthly and yearly expense charts, displaying total expenses and breakdowns by category.</li> <li>The user reviews the detailed analysis reports.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: User has no recorded expense data for the selected timeframe</b></p> <ol style="list-style-type: none"> <li>From #2 of the basic flow, the system notifies the user that there is no expense data available for the specified timeframe.</li> <li>The user selects another timeframe.</li> <li>Continue #2 in the basic flow.</li> </ol> <p><b>Alternative flow 2: User has no recorded expense data</b></p> <ol style="list-style-type: none"> <li>From #2 of the basic flow, the system notifies the user that there is currently no expense data available for analysis.</li> <li>The user is prompted to record expenses to enable analysis features.</li> <li>The user chooses to cancel and return to the application homepage.</li> </ol>
Pre-conditions	<p>The user has recorded at least one expense.</p> <p>The user goes to the homepage.</p>
Post-conditions	The user successfully reviewed the displayed monthly and yearly financial charts and analysis reports.

## 2.18 Use-case: Chat with the financial AI assistant

Use case Name	Chat with the financial AI assistant.
Brief description	The user chats with a financial AI assistant to receive information and get guidance related to their finances.
Actors	User and AI API
Basic Flow	<ol style="list-style-type: none"> <li>The user selects the "FinAI chatbot" feature.</li> </ol>



WalletWise	Version: 1.1
Use-Case Specification	Date: 16/May/24
UCS-WW	

	<ol style="list-style-type: none"> <li>The user initiates a conversation by typing a question, request, or topic of interest related to their finances.</li> <li>Based on the user's input, the financial AI assistant performs one or more of the following actions: <ul style="list-style-type: none"> <li>Provides statistics and analysis about the user's finances.</li> <li>Suggest the user budget plan.</li> <li>Offers helpful suggestions or guidance (budgeting tips, investment and saving recommendations, ...).</li> </ul> </li> <li>The user ends the chat session.</li> </ol>
Alternative Flows	<p><b>Alternative flow 1: Misunderstood the user's input</b></p> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the financial AI assistant misinterprets the user's intent.</li> <li>The AI chatbot asks for clarification by rephrasing the user's input.</li> <li>Continue #5 of the basic flow.</li> </ol> <p><b>Alternative flow 2: Lacks information to complete a task</b></p> <ol style="list-style-type: none"> <li>From #4 of the basic flow, the financial AI assistant lacks information to complete a task.</li> <li>The AI chatbot requests additional details from the user.</li> <li>Continue #5 of the basic flow.</li> </ol> <p><b>Alternative flow 3: The user asks follow-up questions</b></p> <ol style="list-style-type: none"> <li>After #5 of the basic flow, the user asks follow-up questions.</li> <li>The interaction continues until the user ends the chat session.</li> </ol>
Pre-conditions	<p>The user must have logged into the WalletWise application and have access to the "FinAI chatbot" feature.</p> <p>The user has an internet connection.</p>
Post-conditions	<p>The user has received the requested information, completed the desired financial task, or obtained helpful financial guidance.</p>