

## **Agile Software Development**

# **Budgeteer.io**

This comprehensive Software Specification Document outlines the requirements for our expense management system application, "Budgeteer.io." Developed by our software development team, it serves as a blueprint for designing and implementing the system, detailing the software requirements and specifications.

CMSC 129 - SOFTWARE ENGINEERING 2  
Second Semester, A.Y. 2022-2023

### **Members:**

Borromeo, Blesse Rachelle  
Cimafranca, JV Renz  
Hupp, James Ezra  
Lescano, Renmar  
Lupac, Coebe Austin

May, 2023

## II. Table of Contents

Title .....	1
Table of Contents .....	2
Problem Statement .....	3
Project Objectives .....	3
App Description .....	3
Project Limitations .....	4
Project Scope and Iterations .....	4
User Stories .....	6
Functional and Non-Functional Requirements .....	8
Gantt Chart .....	9
Use Cases .....	10
Test Results .....	15
Risk Identification and Management .....	17
Competitive Analysis .....	19
Recommendations .....	19

### **III. Problem Statement**

In the wake of the pandemic and the government's inadequate management, our country is grappling with a mounting inflation crisis. This unfortunate situation has resulted in steep price hikes, particularly in essential goods and transportation, as measures to offset the economic instability. As a result, it has become crucial to exercise prudent financial planning and budgeting to ensure our money is not quickly depleted amid these challenging circumstances.

### **IV. Project Objectives**

The main objective of this project is to develop a comprehensive web application that serves as an effective strategy to empower users in budgeting and managing their financial transactions and expenses.

Specifically, this project aims:

- To enable users to accurately track and record their current financial resources and expenditures.
- To provide users with valuable guidance and tools to make informed decisions when allocating their budget, ensuring wise and responsible financial choices.
- To offer a tracking system to monitor account transactions on a daily, weekly, monthly, and yearly basis, fostering a comprehensive understanding of individual spending habits and patterns.
- To facilitate users in planning for their future expenses by offering insightful projections and recommendations, encouraging proactive financial planning and discouraging impulsive spending.

By accomplishing these objectives, the proposed web application will significantly enhance users' ability to manage their finances effectively, fostering financial stability and a more secure financial future.

### **V. App Description**

*Budgeteer.io*, the ultimate expense management system designed to revolutionize the way individuals and businesses track and manage their finances. Our user-friendly app empowers

users with a seamless experience to effortlessly input, categorize, and organize their expenses, providing a streamlined approach to financial management.

With *Budgeteer.io*, keeping your spending in check has never been easier. Our intuitive interface allows you to effortlessly set and track budgets, ensuring you stay on target and achieve your financial goals. Whether you're aiming to save money, improve your budgeting skills, or gain a better understanding of your spending habits, Budgeteer.io offers the tools and insights you need to make informed financial decisions.

Take charge of your financial future with *Budgeteer.io*. Whether you're an individual or a business, our innovative expense management system is here to empower you with the tools and insights needed to achieve financial success. Start your journey towards financial freedom today with Budgeteer.io, where simplicity meets powerful functionality.

## **VI. Project Limitations**

The system is currently in a developmental stage and does not have a publicly accessible domain. Furthermore, it lacks the capability to verify email addresses. When it comes to expense tracking, the system relies on manual input and is unable to synchronize with bank accounts or automatically process receipts and other files for seamless addition to the expense list. Moreover, it supports only one currency at a time, lacking flexibility for users managing multiple currencies simultaneously.

Recommendations for the overall improvement of the application will be provided in the recommendations section of this paper.

## **VII. Project Scope and Iterations**

### **Iteration 1:**

*Iteration 1* of the software project focused on laying the foundation for the software by developing and implementing a secure login system, a robust database, and intuitive buttons, all aimed at creating a functional and user-friendly system.

### **Iteration 2:**

*Iteration 2* of the software project was built upon the foundation created in Iteration 1 by adding an Expenses and Incomes Form, Edit/Delete Entries, and Calendar, all aimed at

enhancing the functionality of the software and providing users with powerful tools to manage their finances.




**Iteration 3:**





*Iteration 3* of the software project expanded on the features developed in Iterations 1 and 2 by adding critical functionality such as Expense Statistics. The goal was to provide users with more tools to manage their finances and expand the capabilities of the software.

**Iteration 4:**




*Iteration 4* of the software project focused on testing and deployment to ensure that the software is fully functional and ready for release, which is a vital part of the development process for meeting end-user needs and expectations.

## VIII. User Stories

ITERATION #1	
<b>Login System</b>  <b>As a User,</b> <b>I want to</b> be able to create a secure login profile, <b>So that</b> I can keep my personal information safe.	Priority:  Must Have  Story Points: 5
<b>Database</b>  <b>As a Software Developer</b> <b>I want to</b> develop and implement a robust database system, <b>So that</b> the software can efficiently store and retrieve data without any errors or data loss.	Priority:  Must Have  Story Points: 13
<b>Buttons</b>  <b>As a User,</b> <b>I want to</b> be able to easily navigate through the software using intuitive buttons, <b>So that</b> I can use the software to complete tasks quickly.	Priority:  Must Have  Story Points: 5


ITERATION #2	
<b>Expenses Form</b>  <b>As a User,</b> <b>I want to</b> be able to add and track my expenses in the software, <b>So that</b> I can keep track of my spending habits.	Priority:  Must Have  Story Points: 5
<b>Edit/Delete Entries</b>  <b>As a User,</b> <b>I want to</b> be able to edit my expenses in the software, <b>So that</b> I can make adjustments to my spending habits and maintain accurate financial records.	Priority:  Must Have  Story Points: 5
<b>Incomes Form</b>  <b>As a User,</b> <b>I want to</b> be able to set an income, <b>So that</b> I can monitor my spending and stay within my financial limits.	Priority:  Should Have  Story Points: 8
<b>Custom Date Feature</b>  <b>As a User,</b> <b>I want to</b> be able to see my expenses in a daily, weekly, monthly, and yearly customized view <b>So that</b> I can easily track my spending over time and make informed financial decisions based on my spending patterns.	Priority:  Should Have  Story Points: 13


Priority Value: (1 as the Highest Priority)

1.  Must Have    3.  Could Have  
 2.  Should Have




Story Points Scale:

- 1 - takes 1-2 days      5 - takes 3-5 days      13 - takes 7-10 days  
 3 - takes 2-3 days      8 - takes 5-7 days

ITERATION #3	
<b>Expense Statistics</b>  <b>As a User</b> <b>I want to</b> be able to view expense statistics in the software, <b>So that</b> I can analyze my spending patterns and make informed financial decisions.	Priority:  Must Have  Story Points: 13

ITERATION #4	
<b>Software Testing</b>  <b>As a Software Developer,</b> <b>I want to</b> perform comprehensive testing of the software, <b>So that</b> I can ensure the software functions as intended and meets the needs of end-users.	Priority:  Must Have  Story Points: 8

Priority Value: (1 as the Highest Priority)

1.  Must Have    3.  Could Have
2.  Should Have

Story Points Scale:

- |                           |                           |                             |
|---------------------------|---------------------------|-----------------------------|
| <b>1</b> - takes 1-2 days | <b>5</b> - takes 3-5 days | <b>13</b> - takes 7-10 days |
| <b>3</b> - takes 2-3 days | <b>8</b> - takes 5-7 days |                             |

## **IX. Functional and Non-Functional Requirements**

The system includes the following functional requirements:

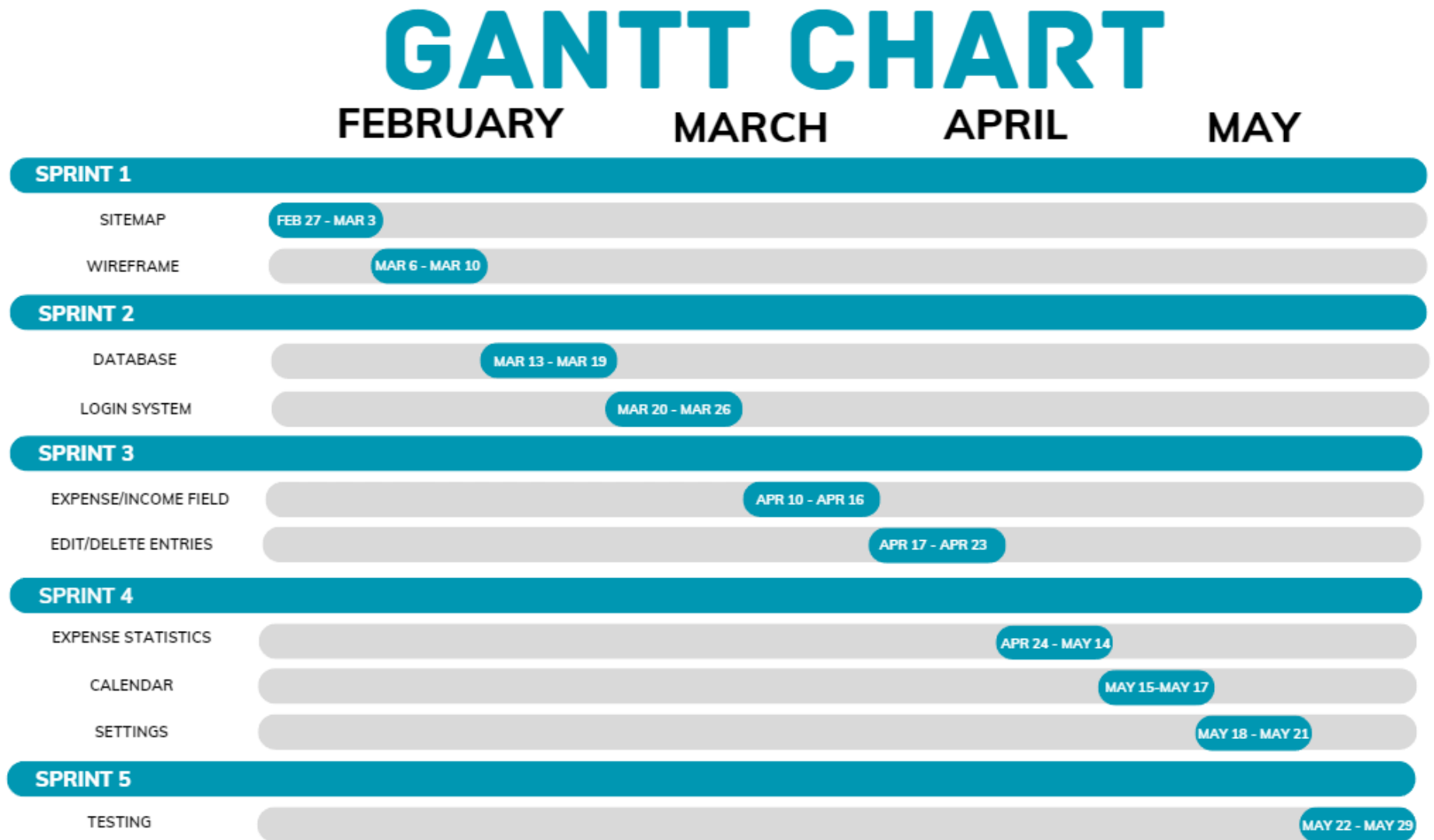
1. Users must be able to browse different website features by clicking a side navigation panel.
2. Users must be able to add, modify, or delete various expense and income list fields in the system.
3. Users must be able to view expenditure statistics at any valid time range.
4. Users must be able to formulate and set budget goals based on the expenditure statistics.
5. Users must be able to access their accounts by entering their email address and password

The system includes the following non-functional requirements:

1. Users must find the website easy to navigate, with functions that are easily understood.
2. Users' personal information must be stored securely and accessible only by authorized individuals.
3. Users must be able to view the history of expense and income entries.
4. The website must load within three seconds when accessed with a fast internet speed.
5. The website must be accessible 24/7, and in the event of system downtime, it must be restored within 24 hours.



## X. Gantt Chart



## XI. Use Cases

<b>Use Case_No.</b>	1
<b>User case</b>	Landing Page
<b>Pre-condition:</b>	User must have stable internet connection
<b>Acceptance Criteria:</b>	System shows landing page
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Open the website</li> <li>2. System shows landing page</li> <li>3. Can go to register/login page</li> <li>4. End.</li> </ol>

<b>Use Case_No.</b>	2
<b>User case</b>	Sign up/Register
<b>Pre-condition:</b>	User must not have an existing active account in the database.
<b>Acceptance Criteria:</b>	User successfully registered.
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Open the website</li> <li>2. Proceeds to register page</li> <li>3. Enter the first name, last name, username, email address and password</li> <li>4. Enter the password again for confirmation</li> <li>5. System accepts all the data in each field</li> <li>6. Proceeds to log in</li> <li>7. End.</li> </ol>
<b>Extension</b>	<p>If email is taken:</p> <ol style="list-style-type: none"> <li>4. a. System detects email is taken</li> <li>4.a.1. System displays the message "An account with this e-mail address already exist."</li> <li>4.a.2. Return to Step #3</li> </ol> <p>If username is taken:</p> <ol style="list-style-type: none"> <li>5.a. System detects that username is taken</li> <li>5.a.1. System displays the message " An account with this username already exists."</li> </ol> <p>If confirm password is different from the given password:</p>

	<p>5.a.2 System detects that the confirm password is different from the given password</p> <p>5.a.3 System displays the message "Password you have entered does not match."</p> <p>If atleast one of the fields has no values:</p> <p>5.a.4. System detects that one or more of the fields are blank</p> <p>5.a.5. System requires the user to fill in the blank field</p>
--	--

<b>Use Case_No.</b>	3
<b>User case</b>	Login
<b>Pre-condition:</b>	User must have an existing active account in the database.
<b>Acceptance Criteria:</b>	User successfully logins. / User successfully enters the app and accesses the features.
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Open the website</li> <li>2. Proceeds to login page</li> <li>3. Enter the username</li> <li>4. System accepts the username</li> <li>5. Enter the password</li> <li>6. System accepts the password</li> <li>7. Proceeds to log in</li> <li>8. End.</li> </ol>
<b>Extension</b>	<p>If username or password is incorrect:</p> <ol style="list-style-type: none"> <li>6. a. System detects username or password is incorrect</li> <li>6. a. 1. System displays a message "The name or password you have entered is incorrect. Please try again."</li> <li>6. a. 2. Return to Step #3</li> </ol>

<b>Use Case_No.</b>	4
<b>User case</b>	Home page
<b>Pre-condition:</b>	User successfully logins in the system
<b>Acceptance Criteria:</b>	System shows home page with current date
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Open the website</li> </ol>

	<ol style="list-style-type: none"> <li>2. Proceeds to log in</li> <li>3. System shows home page with current date</li> <li>4. End.</li> </ol>
<b>Extension</b>	<p>If there is no transaction on the current date:</p> <ol style="list-style-type: none"> <li>3.a. System displays message “No Expenses recorded.” or “No Incomes recorded.”</li> </ol>

<b>Use Case_No.</b>	5
<b>User case</b>	Add Income
<b>Pre-condition:</b>	User successfully logins in the system
<b>Acceptance Criteria:</b>	User successfully added a income
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Proceed to “Income Section”</li> <li>2. User proceeds to input amount for income</li> <li>3. User sets the date and category of the income entry</li> <li>4. Users can opt to add a comment or description to the income entry</li> <li>5. System accepts income entry</li> <li>6. System displays a message “New Income added.”</li> <li>7. Income successfully set</li> <li>8. End.</li> </ol>
<b>Extension</b>	<p>If input income is empty / blank:</p> <ol style="list-style-type: none"> <li>5.a. System detects income is 0 or empty</li> <li>5.a.1. System displays message “Please fill in all required fields.”</li> <li>5.a.2. Return to Step #2</li> </ol> <p>If input income value is less than or equal to zero:</p> <ol style="list-style-type: none"> <li>5.a.1. System displays message “Please enter valid positive amount.”</li> <li>5.a.2. Return to Step #2</li> </ol>

<b>Use Case_No.</b>	6
<b>User case</b>	Add Expenses
<b>Pre-condition:</b>	User successfully logins in the system
<b>Acceptance Criteria:</b>	User successfully added an expense
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Proceed to the “Expense” section in the app.</li> </ol>

	<ol style="list-style-type: none"> <li>2. User proceeds to input amount for expense</li> <li>3. User sets the date and category of the expense entry</li> <li>4. Users can opt to add a comment or description to the expense entry</li> <li>5. System accepts expense entry</li> <li>6. System displays a message “New Expense added.”</li> <li>7. Expense successfully set</li> <li>8. End.</li> </ol>
<b>Extension</b>	<p>If input expense is empty / blank or 0:</p> <ol style="list-style-type: none"> <li>5.a. System detects budget is 0 or empty</li> <li>5.a.1. System displays message “Please fill in all required fields.”</li> </ol>

<b>Use Case_No.</b>	7
<b>User case</b>	View Expenditure Statistic
<b>Pre-condition:</b>	User successfully logs in the system
<b>Acceptance Criteria:</b>	User can view different statistics of the expenses
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Proceed to the “Statistics” Section</li> <li>2. System shows the different statistics and data based on the list</li> <li>3. System shows the history of income and expense transactions</li> <li>4. End.</li> </ol>
<b>Extension</b>	<p>If there are no expenses added:</p> <ol style="list-style-type: none"> <li>2.a. System shows “ No Expenses Recorded”</li> </ol> <p>If there are no incomes added</p> <ol style="list-style-type: none"> <li>3.a. System shows “No Incomes Recorded”</li> </ol>

<b>Use Case_No.</b>	8
<b>User case</b>	Edit/Delete Expense
<b>Pre-condition:</b>	User successfully logs in the system
<b>Acceptance Criteria:</b>	User can edit and modify or delete expenses list in the fields.
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Proceed to the “Statistics” section in the app.</li> <li>2. System shows the list of expenses</li> <li>3. System shows “No Expenses Recorded” if list are empty</li> </ol>

	<ol style="list-style-type: none"> <li>4. Edits the amount and comment/ description on the expense entry</li> <li>5. Deletes the expense entry</li> <li>6. System updates the final list</li> <li>7. End.</li> </ol>
<b>Extension</b>	<p>If the expense value is zero or less than zero:</p> <p>4.a. System shows “ Value must be greater than zero.”</p>

<b>Use Case_No.</b>	9
<b>User case</b>	Edit/Delete Income
<b>Pre-condition:</b>	User successfully logins in the system
<b>Acceptance Criteria:</b>	User can edit, modify or delete incomes list in the fields.
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Proceed to the “Statistics” section in the app.</li> <li>2. System shows the list of incomes</li> <li>3. System shows “No Incomes Recorded” if list are empty</li> <li>4. Edits the desired incomes</li> <li>5. Deletes the desired expenses</li> <li>6. System updates the final list</li> <li>7. End.</li> </ol>
<b>Extension</b>	<p>If there income value is zero or less than zero:</p> <p>4.a. System shows “ Value must be greater than zero.”</p>

<b>Use Case_No.</b>	10
<b>User case</b>	Calendar Feature
<b>Pre-condition:</b>	User successfully logins in the system
<b>Acceptance Criteria:</b>	User successfully views his/her expense in calendar mode
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Proceed to “Calendar Section”</li> <li>2. System displays calendar.</li> <li>3. End.</li> </ol>

<b>Use Case_No.</b>	11
---------------------	----

<b>User case</b>	Settings
<b>Pre-condition:</b>	User successfully logins in the system
<b>Acceptance Criteria:</b>	User successfully views his/her credentials.
<b>Main Scenario</b>	<ol style="list-style-type: none"> <li>1. Proceed to “Settings”</li> <li>2. System displays the user's credentials.</li> <li>3. User can edit his/her credentials.</li> <li>4. End.</li> </ol>

## XII. Test Results

Use Case #	Use Case	Expected Result/Acceptance Criteria	Criteria/Condition	Results
UC-01	Landing Page	System shows landing page	<ul style="list-style-type: none"> <li>• “Start now” must redirect to sign up/register page and “Log in” to log in page.</li> </ul>	Passed
UC-02	Sign up/Register	User successfully registered.	<ul style="list-style-type: none"> <li>• Name input must only contain letters.</li> <li>• Input fields must not be blank</li> <li>• System should have error handlers.</li> <li>• Users cannot have multiple registrations in one username and/or email.</li> <li>• System informs users that they are successfully registered.</li> <li>• System should show the password when the checkbox is clicked.</li> </ul>	Passed Passed Passed Passed Passed
UC-03	Login	User successfully logins. / User successfully enters the app and accesses the features.	<ul style="list-style-type: none"> <li>• “create an account” must redirect to sign up/register</li> <li>• Users must input correct details and not leave any input fields blank.</li> <li>• System should have error handlers.</li> <li>• Login button must redirect to the home page when all input details</li> </ul>	Passed Passed Passed Passed

			<p>are correct.</p> <ul style="list-style-type: none"> <li>System should show the password when the checkbox is clicked.</li> </ul>	Passed
UC-04	Home Page	System shows home page with current date.	<ul style="list-style-type: none"> <li>System displays the message “No Expenses recorded.” or “No Incomes recorded.” when there are no transactions added.</li> <li>The daily transactions must be displayed according to their respective dates.</li> <li>Arrow dates must traverse the dates.</li> </ul>	<p>Passed</p> <p>Passed</p> <p>Passed</p>
UC-05	Add Income	User successfully added a budget.	<ul style="list-style-type: none"> <li>User needs to add the amount, set date and category, and an optional description.</li> <li>System informs users that they added an income.</li> <li>Added income is reflected in Home and Statistics</li> <li>“Save” button adds a new income.</li> <li>“Cancel” button redirects to home.</li> </ul>	<p>Passed</p> <p>Passed</p> <p>Passed</p> <p>Passed</p> <p>Passed</p>
UC-06	Add Expenses	User successfully added an expense.	<ul style="list-style-type: none"> <li>User needs to add the amount, set date, payment method, category, and an optional description.</li> <li>System informs users that they added an expense.</li> <li>Added expense is reflected in Home and Statistics</li> <li>“Save” button adds a new expense.</li> <li>“Cancel” button redirects to home.</li> </ul>	<p>Passed</p> <p>Passed</p> <p>Passed</p> <p>Passed</p> <p>Passed</p>
UC-07	View Expenditure Statistic	User can view different statistics of the expenses.	<ul style="list-style-type: none"> <li>User must input an expense and income first.</li> <li>System should display all balances, statistics details and charts.</li> </ul>	<p>Passed</p> <p>Passed</p>
UC-08	Edit/Delete	User can edit and	<ul style="list-style-type: none"> <li>User should add an expense first.</li> </ul>	Passed



	Expense	modify or delete expenses list in the fields.	<ul style="list-style-type: none"> <li>• “Edit” button let the user edit the value selected</li> <li>• “Delete” button deletes the value selected</li> <li>• Edited/Deleted values must be reflected in Home and Statistics</li> </ul>	Passed  Passed  Passed
UC-09	Edit/Delete Income	User can edit and modify or delete incomes list in the fields.	<ul style="list-style-type: none"> <li>• User should add an income first.</li> <li>• “Edit” button let the user edit the value selected</li> <li>• “Delete” button deletes the value selected</li> <li>• Edited/Deleted values must be reflected in Home and Statistics</li> </ul>	Passed Passed  Passed Passed
UC-10	Calendar Feature	User successfully views his/her expense in calendar mode.	<ul style="list-style-type: none"> <li>• System should show the monthly and current date.</li> </ul>	Passed
UC-11	Settings	User successfully views his/her credentials.	<ul style="list-style-type: none"> <li>• System must show the user's credentials.</li> <li>• System must update the credentials once it is edited.</li> </ul>	Passed  Passed

### XIII. Risk Identification and Management

#### PROJECT, PRODUCT, AND BUSINESS RISKS

RISK	AFFECTS	DESCRIPTION
Technical Risk	Project	There is a risk that the Expense Management System may encounter technical issues during development, such as bugs or compatibility issues. This could lead to delays and budget overruns.
Unforeseen Circumstances	Project	There is a risk that the project may be delayed due to unforeseen circumstances, such as illness of a team member, lack of resources, or external events. This could impact the project timeline and potentially cause budget overruns.

User Interface	Product	There is a risk that the user interface of the Expense Management System may be confusing or difficult to use, which could impact the usability of the product.
Performance Risk	Product	There is a risk that the Expense Management System may experience performance issues, such as slow loading times or errors, which could impact the user experience.
Market Demand	Business	There is a risk that the Expense Management System may not be in demand in the market, which could impact the financial viability of the product.
Regulatory Compliance	Business	There is a risk that the Expense Management System may not comply with relevant regulations, such as data protection laws, which could result in legal issues and financial penalties.

### RISK MANAGEMENT

RISK	AFFECTS	DESCRIPTION
Technical Risk	Project	Conduct thorough testing and quality assurance throughout the development process to identify and resolve technical issues immediately.
Unforeseen Circumstances	Project	Maintain open communication channels with stakeholders to promptly address any lack of resources or external events that may arise.
User Interface	Product	Conduct user research and usability testing to understand user needs and preferences, ensuring the user interface is intuitive and user-friendly.
Performance Risk	Product	Optimize code and database queries to improve system responsiveness and reduce loading times.
Market Demand	Business	Conduct market research to assess the demand for expense management systems

		and identify target customer segments.
Regulatory Compliance	Business	Stay informed about relevant regulations, such as data protection laws, and ensure the Expense Management System adheres to all necessary compliance requirements.

#### XIV. Competitive Analysis

	<b>Budgeteer.io</b>	<b>Wallet: Budget Expense Tracker</b>	<b>Goodbudget: Budget &amp; Finance</b>	<b>Budget planner-Expense tracker</b>
<b>Product/ Service</b>	Expense tracker	Expense tracker	Expense tracker	Expense tracker
<b>Target market</b>	People who aim to save money and improve budgeting skills.	People who want to save money, plan for the future, and want to see their finances in one place	People who want to have their home budget planned.	People who are concerned about how they spend their money.
<b>Strengths</b>	Empowers users with a seamless experience to effortlessly input, categorize, and organize their expenses.	Currency converter, bank synching	Available both in mobile and website	Inbuilt calculator
<b>Weaknesses</b>	It lacks the capability to verify email addresses.	Some features have bugs.	It constantly crashes and it can be complicated at first.	Requires payment for other features

#### XV. Recommendations

The system can greatly enhance security measures by implementing two-factor authentication, providing an additional layer of protection through verification processes. Furthermore, integrating an API for email validation would significantly enhance the

accuracy and legitimacy of user-provided email addresses, ensuring reliable communication channels. To ensure transparency and improve record-keeping, it is essential for the app's date system to accurately reflect transactions based on the date they are entered. By aligning the current date with transaction dates, users can have a clear and accurate overview of their financial activities over time. This functionality would assist in organizing and managing personal finances, allowing users to track their transaction history and compare it with other financial statements or records, promoting a more efficient and streamlined user experience. To cater to users who frequently travel internationally, the inclusion of a currency converter would be a valuable addition.