**A Budgeting App for Young Adults**

**BY**

**Terry Wambui**

**167293**

Submitted to the Strathmore Institute of Management and Technology in partial fulfilment of the requirements for the award of the Diploma in Business Information Technology

**Diploma in Business Information Technology**

**Strathmore University**

**07/08/2024**

**Declaration and Approval**

I **Terry Wambui** declare that this work has not been previously submitted and approved for the award of a Diploma in business information Technology by this or any other University. To the best of my knowledge and belief, the proposal contains no material previously published or written by another person except where due reference is made in the document itself.

**Student Name: Terry Wambui**

**Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Supervisor’s Name: Philip Neri**

**Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Abstract**

BudgetBuddy focuses on addressing the financial management challenges encountered by young adults who often struggle with effective budgeting due to unique financial pressures and a lack of tailored tools. Despite advancements in digital financial solutions, these demographics continue to face issues such as overwhelming debt, unpredictable income, and the complexities of managing both personal and business finances. Traditional budgeting methods and many existing apps fail to meet their specific needs, resulting in financial instability and stress. This study aims to develop a user-friendly budgeting app (BudgetBuddy) that leverages modern technology to provide real-time tracking, automated expense categorization, and personalized financial insights, ultimately promoting better financial health and stability for these key groups. BudgetBuddy is a budget app made for young adults in Kenya, with features that cater to local needs as well as technological growth. M-Pesa integration allows for secure and convenient mobile transactions which are the norm in this area. It gives personalized advice using artificial intelligence when coaching clients financially so they can make better decisions about their money.  BudgetBuddy also makes saving fun by providing rewards once you hit your budget target. The application supports both the Kiswahili and English languages to make it reach many people, and has an offline feature that allows tracking of money while moving. It is free and sends notifications for bills and debts so as not miss any financial obligation. The applied development approach that will be utilized is the Agile Methodology.

**Table of Contents**

Contents

[**Chapter 1: Introduction** 1](#_Toc173930108)

[**1.1: Background** 1](#_Toc173930109)

[**1.2: Problem Statement** 2](#_Toc173930110)

[**1.3: Aim** 2](#_Toc173930111)

[**1.4: Specific Objectives** 2](#_Toc173930112)

[**1.5: Research questions** 2](#_Toc173930113)

[**1.6: Justification** 3](#_Toc173930114)

[**1.7: Scope and Limitation** 3](#_Toc173930115)

[**Chapter 2: Literature Review** 4](#_Toc173930116)

[**2.1: Introduction** 4](#_Toc173930117)

[**2.2: Budgeting methods** 4](#_Toc173930118)

[**2.3: Challenges that young adults face with budgeting** 4](#_Toc173930119)

[2.3.1 Financial Literacy 4](#_Toc173930120)

[2.3.2 Inexperience with Long-term Planning 4](#_Toc173930121)

[2.3.3 High Living Cost 5](#_Toc173930122)

[**2.4: Existing budgeting apps, methods and tools for young adults** 5](#_Toc173930123)

[2.4.1 Monefy 5](#_Toc173930124)

[2.4.2 MoneyManager 6](#_Toc173930125)

[2.4.3 Tala 7](#_Toc173930126)

[**2.5: Gaps of the existing apps, methods and tools** 8](#_Toc173930127)

[**2.6: Conceptual Framework** 9](#_Toc173930128)

[**Chapter 3: Research Methodology** 10](#_Toc173930129)

[**3.1: Introduction** 10](#_Toc173930130)

[**3.2: Agile development methodology** 10](#_Toc173930131)

[3.2.1: Product Backlog 11](#_Toc173930132)

[3.2.2 Sprint Planning 11](#_Toc173930133)

[3.2.3 Sprint Backlog 11](#_Toc173930134)

[3.2.4 Sprint 11](#_Toc173930135)

[**3.2.5 Tested and Integrated Product Component** 12](#_Toc173930136)

[3.2.6 Sprint Review 12](#_Toc173930137)

[3.3 Tools to be used 12](#_Toc173930138)

[3.4 Reasons for choosing Scrum Model 12](#_Toc173930139)

[3.5 Deliverables 13](#_Toc173930140)

[3.5.1 System Modules 13](#_Toc173930141)

[3.5.2 Users and Roles 13](#_Toc173930142)

[**Chapter 4: System Analysis and Design** 14](#_Toc173930143)

[**4.1: Introduction** 14](#_Toc173930144)

[**4.2: System Analysis** 14](#_Toc173930145)

[4.2.1: Functional Requirements 14](#_Toc173930146)

[4.2.2: Non-Functional Requirements 15](#_Toc173930147)

[**4.3 System Design** 15](#_Toc173930148)

[**4.4 System Analysis Diagrams** 15](#_Toc173930149)

[4.5.1 Use Case Diagram 15](#_Toc173930150)

[4.5.2 Context Diagram 17](#_Toc173930151)

[4.5.4 Data Flow Diagram Level 1 18](#_Toc173930152)

[4.5.6 Entity Relationship Diagram 19](#_Toc173930153)

[4.5.7 System Architecture 20](#_Toc173930154)

[4.5.8 Wireframes 21](#_Toc173930155)

[**Chapter 5: System Implementation Testing** 23](#_Toc173930156)

[**5.1: Introduction** 23](#_Toc173930157)

[**5.2: System Implementation** 23](#_Toc173930158)

[5.2.1: Systems backend 23](#_Toc173930159)

[5.2.2: System’s front end 25](#_Toc173930160)

[5.2.3: Database Tables 27](#_Toc173930161)

[**5.3: System Testing** 29](#_Toc173930162)

[5.3.1: Black box testing 29](#_Toc173930163)

[5.3.2: Functionality testing 29](#_Toc173930164)

[**5.5: Test Cases** 29](#_Toc173930165)

[**5.6: Test Results** 30](#_Toc173930166)

[**Chapter 6: Conclusions and Recommendations for Future Work** 30](#_Toc173930167)

[**6.1: Conclusions** 30](#_Toc173930168)

[**6.2: Recommendations** 31](#_Toc173930169)

[**6.3: Future Work** 31](#_Toc173930170)

[**References** 32](#_Toc173930171)

[**Appendix** 33](#_Toc173930172)

**List of Figures**

[**Figure 2.1 Monefy App** (Monefy, 2021) 6](#_Toc173929667)

[**Figure 2.2 Money Manager** (Money Manager, 2010) 7](#_Toc173929668)

[**Figure 2.3 Tala App** (tala.co) 8](#_Toc173929669)

[**Figure 2.4: Conceptual Framework** 9](#_Toc173929670)

[**Figure 5.1: Login session 23**](#_Toc173929671)

[**Figure 7: Gantt Chart** 32](#_Toc173929672)

# **Chapter 1: Introduction**

## **1.1: Background**

Most youngadults lack basic financial education in managing their money and debt. Traditional budgeting methods are often cumbersome and prone to errors, and research indicates a need for more engaging tools and financial knowledge among young adults (University of Illinois at Urbana-Champaign, 2018). Nowadays people do not carry around a small notebook to take note of the details of every transaction in their daily life. In addition, with the amount of work and studies done throughout the day, it is difficult for them to recall the exact amount of every expense made. This has led to decreased interest in financial tracking, making budgeting later on more troublesome. This can lead to poor financial decisions, hindering their ability to achieve financial goals and build a secure future.

Existing budgeting apps often fail to address the unique financial needs and behaviors of young individuals. Young adults struggle with overwhelming student loans, uncertain income, and a lack of financial knowledge, making it challenging to plan and manage their finances. Moreover, many users find existing apps complex or not user-friendly, leading to inconsistent use. The absence of comprehensive, user-friendly, and specifically designed budgeting solutions worsens financial instability, resulting in increased debt, poor financial health, and a diminished quality of life. This highlights the pressing need for innovative budgeting tools tailored to these demographics, providing the necessary support and education for achieving financial well-being.

The rise of mobile technology and the increasing adoption of smartphones have provided new opportunities for developing more accessible and user-friendly budgeting solutions (Wong & Najib, 2022). To help young adults cultivate the habit of monitoring their daily spending, BudgetBuddy will be used for tracking expenses and simplify the task of recording every transaction making it effortless. It will become an essential tool for today’s individuals. Users will be able to update income and expenses through an interface, allowing them to visualize their financial situation over a specific period. BudgetBuddy aims to make the process of tracking finances engaging and uncomplicated, providing a convenient solution for managing money on the go.

## **1.2: Problem Statement**

Effective financial management is a significant challenge for many young adults often leading to stressful financial situations and hindering success. Many young adults struggle with managing their finances due to lack of financial knowledge, resulting in poor budgeting, high debt, and inadequate savings. Startup founders face their own financial challenges, trying to balance personal and business finances without free necessary tools (Hezretov, 2018). This can lead to stress, reduced quality of life, and difficulties in achieving entrepreneurial success, which can have broader economic impacts. Research shows that financial instability is a major cause of mental health issues and reduced productivity highlighting the urgent need to address this issue comprehensively.

Researchers emphasize that without proper financial management tools, young adults are at risk of financial failure, leading to business closures, further impeding financial recovery and growth (Wong & Najib, 2022). Financial literacy programs and effective budgeting tools are crucial in addressing these risks. Key stakeholders stress the need for accessible and user-friendly financial management solutions, including financial institutions, educational bodies, and mental health organizations. By providing the right tools and knowledge, I can try and help create a financially resilient and economically vibrant society.

## **1.3: Aim**

To create a web-based system for young adults that will improve their budgeting skills.

## **1.4: Specific Objectives**

1. To analyze the challenges that young adults face with budgeting
2. To review current existing budgeting apps, methods and tools used by young adults
3. To design and develop a web-based budgeting system for young adults
4. To evaluate and test the budgeting app.

## **1.5: Research questions**

1. What are the challenges that young adults face with budgeting?
2. What are the current existing budgeting apps, methods and tools used by young adults?
3. How will the web-based budgeting system be designed and developed?
4. How can the web-based budgeting app be evaluate and tested?

## **1.6: Justification**

The app will address the issue of financial literacy which is a widespread problem, particularly among young adults and startup founders. Many individuals currently lack the fundamental financial knowledge needed to effectively manage their money, resulting to poor financial decisions and instability. It will also address the issue of poor budgeting skills, insufficient integration with banking systems and lack of goal-oriented budgeting features in financial planning tools for personal and business use. Further, it will empower young adults and startup founders to achieve financial stability and success, benefiting economy and society.

## **1.7: Scope and Limitation**

The app will enable young adults and startup founders to create their budget, categorize expenses and set their financial goals. It will have layers of security to protect users’ sensitive financial information. Moreover, the app will use open banking APIs to enable third-party applications to access consumer-permissioned data. It will also focus on budgeting, planning, and analysis to empower users to make informed financial decisions.

Although the app could provide financial tips, it would not get the advices from professional advisors nor would it provide any legal advice. The app will not directly facilitate transactions involving sensitive financial data, such as banking or investment transactions.

# **Chapter 2: Literature Review**

## **2.1: Introduction**

This chapter goes over how budgeting methods have been working in the past and recently. It discusses the current solutions and the gaps in these solutions. It also gives an overview of what the system is going to do.

## **2.2: Budgeting methods**

Budgeting is all about creating a plan to manage your money, so you know exactly where it's going and how it's being saved. It's a way for people to achieve financial stability and reach their financial goals. Basically, it's all about keeping track of your spending, controlling how much you spend, and putting money aside for savings and investments. Nowadays, young adults have lots of different ways to budget, using a mix of old-school and digital methods. Some use customizable spreadsheets in Excel or Google Sheets if they would like getting into details. And then there's the envelope system, where you divide your cash into different categories to help you stick to your budget. There are also some other cool tricks that young adults use to manage their money. Things like automatic savings and investment features, digital banking tools, strategies for paying off debt like the snowball and avalanche methods, and even going old-school with cash-only budgeting.All of these methods help young adults stay on top of their finances, avoid getting into debt, and save up for the future. They provide structure and clarity for your financial planning, so you can make the most of your hard-earned cash.

## **2.3: Challenges that young adults face with budgeting**

### 2.3.1 Financial Literacy

Many young adults struggle with managing their finances due to lack of financial knowledge, resulting in poor budgeting, high debt, and inadequate savings. Schools often do not prioritize financial literacy, leaving many young people unprepared for managing their personal finances. A strong foundation of financial literacy can help support various life goals, such as saving for education, using debt responsibly, and running a business (Fernando, 2024).

### 2.3.2 Inexperience with Long-term Planning

Many young adults find it challenging to plan for their future because they often focus on short term goals lack an understanding of financial matters and face uncertainties in the early stages of their careers. Immediate concerns, like repaying student loans covering rent and socializing tend to take precedence over saving for the term. Concepts like compound interest and investment strategies can seem daunting without teaching or guidance making it hard to prioritize term financial objectives. Additionally, the frequent job changes and relocations common among adults make it tough to establish and stick to term financial plans. The weight of debt adds another layer of complexity leaving room, for saving for tomorrow.

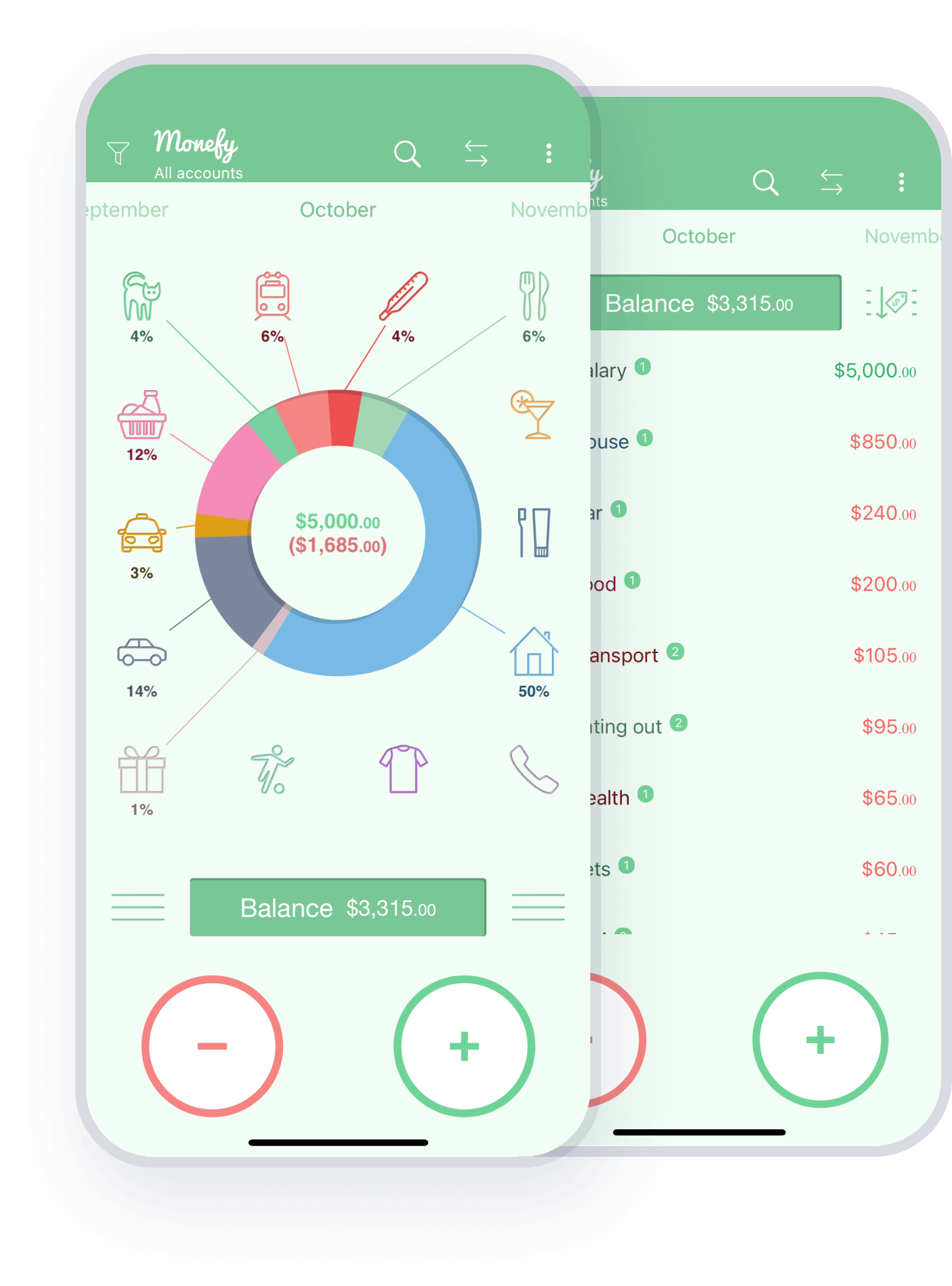
### 2.3.3 High Living Cost

The high cost of living poses a challenge, for adults who are trying to manage their finances efficiently. Rent and housing expenses in cities can eat up a portion of their income sometimes surpassing other financial priorities. Costs related to transportation whether for owning a car or using transport also contribute to the pressure. Basic utilities like electricity, water, internet and phone bills quickly add up along with grocery and food costs – especially when trying to maintain a diet. Moreover, healthcare expenses such, as insurance premiums, medications and medical appointments can be substantial. Are often underestimated when planning budgets.

## **2.4: Existing budgeting apps, methods and tools for young adults**

### 2.4.1 Monefy

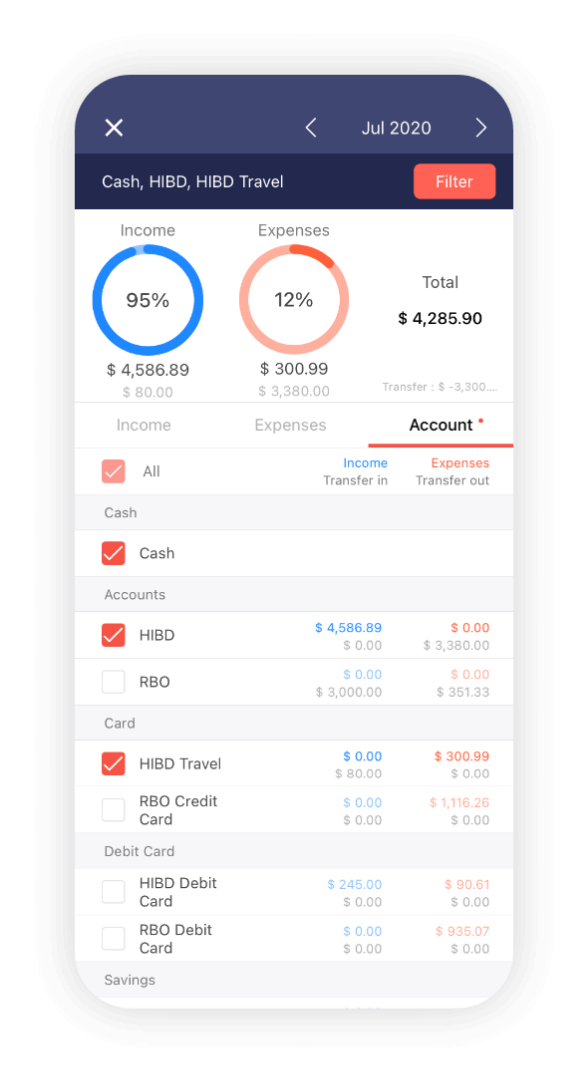
Monefy is a budget app and expense tracker for managing your money. It is easy to use however one does not understand it works. It doesn’t provide a guide on how to navigate through the app. Its free version has limited features, and users need to purchase the Pro version to unlock more advanced functionalities such as syncing between devices and adding multiple accounts. There is also no encryption on the app, one can easily the app and view one’s finances. Monefy does not automatically import transactions from bank accounts or credit cards, one inputs data manually which is cumbersome and prone to errors. Some users have reported issues with syncing data between devices, even with the Pro version. This can lead to discrepancies and data loss, which undermines the reliability of the app.



**Figure 2.1 Monefy App** (Monefy, 2021)

### 2.4.2 MoneyManager

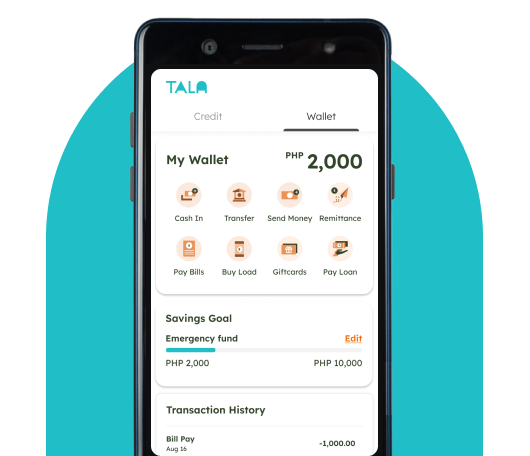
MoneyManager is a financial planning, expense tracking and financial management app. It has good reviews but lacks some basic features that could make the app more friendly. The user needs internet access to use the app, this is inconvenient for people are not always connected to the internet. It does not have a change or forgotten password feature after the first input, this makes it difficult for when one forgets their passcode, there is no way to change it. It is also a bit complicated to use for young adults and startup founders. It is more of an experienced businessperson kind of app. It also has ads and one needs to buy it to use it ad-free.



**Figure 2.2 Money Manager** (Money Manager, 2010)

### 2.4.3 Tala

Tala is a good app for managing small loans but it doesn't do well in full budgeting and long-term money planning. While it's great at tracking loan payments, it doesn't offer much in the way of sorting expenses, setting big money goals, or other deep budgeting tools. This can make it hard for users to get a full view of their money situation and plan for the future properly. To make up for these missing pieces, users might need to use other apps or tools that are better at detailed budgeting and planning for the long run.



**Figure 2.3 Tala App** (tala.co)

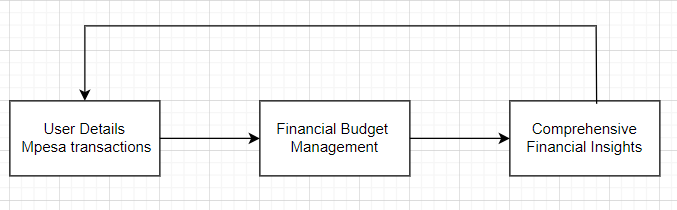
## **2.5: Gaps of the existing apps, methods and tools**

The effectiveness of Monefy, MoneyManager and Tala budgeting and financial management apps are hindered by some notable gaps. For example, Monefy has no navigation guide and as well needs a lot of manual input that is tedious. Its free version only comes with essential features locked behind a paywall such as syncing devices and multiple accounts whereas encryption is not included hence it fails to provide adequate security for user data.

MoneyManager is an excellent application although it requires consistent internet connection for it to work accordingly thus making it less convenient and secure because there is not any password recovery option. Additionally, this app is more complicated for young adults while ads ruin the experience unless someone opts to buy premium version.

Tala manages small loans effectively but lacks comprehensive budgeting and long-term financial planning since it does not have tools for setting significant financial goals neither categorizes expenses. Thusly, if they want a full picture of their economic situation or plan ahead properly, Tala users may need some other applications as well.

## **2.6: Conceptual Framework**



**Figure 2.4: Conceptual Framework**

# **Chapter 3: Research Methodology**

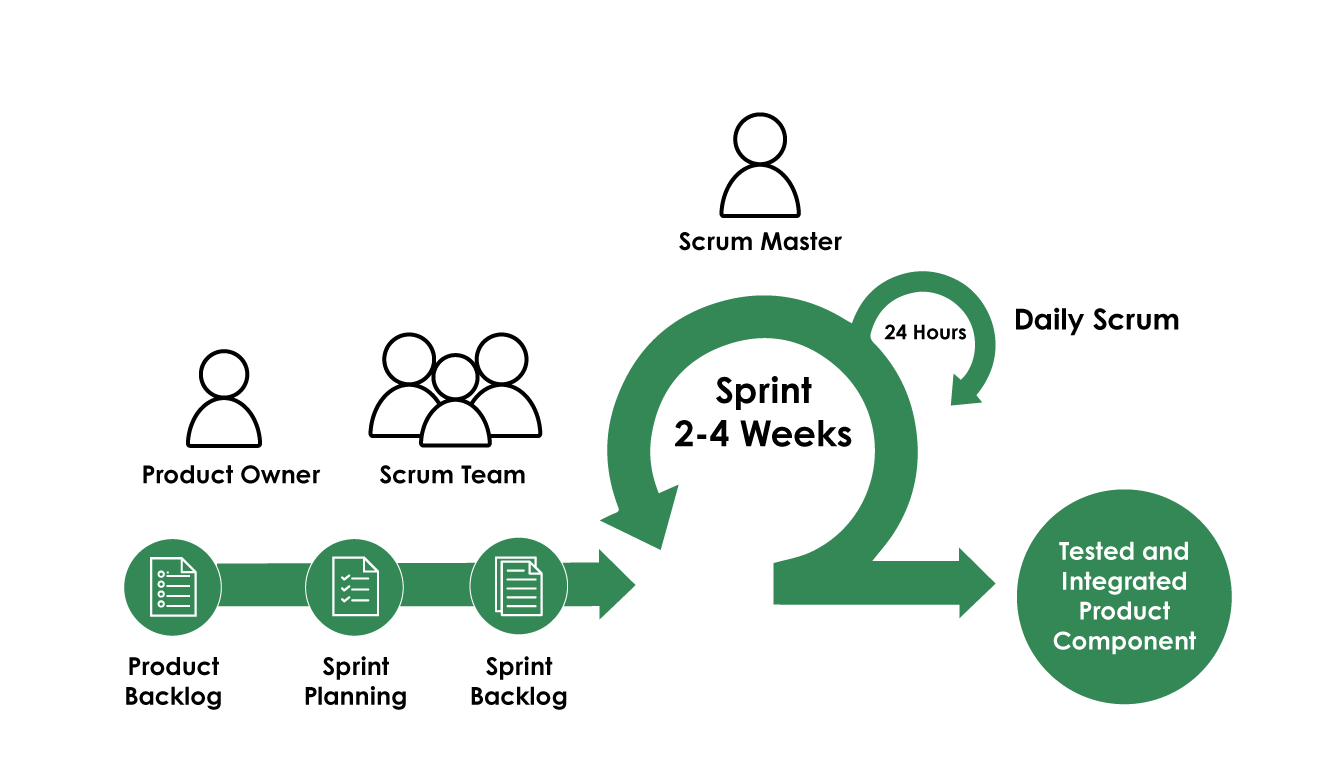
## **3.1: Introduction**

This project is going to use a structured system analysis and design (SSAD) approach. Additionally, it covers the adopted development approach and delve into the system analysis, design, and the corresponding deliverables.

## **3.2: Agile development methodology**

The applied development approach that will be utilized is the Agile Methodology. This is an iterative and incremental approach, emphasizing adaptability and teamwork. It also prioritizes flexibility, collaboration, and teamwork, fit for the ecommerce site which experiences evolving requirements and demands swift delivery (Sacolick, 2022). SSAD is based on structured programming techniques, which involve breaking down a software system into smaller, manageable components or modules.

This is an agile methodology that adheres to a structured framework which organized the work into timeboxed periods known as sprints which typically range from one to two weeks. User stories outlining the requirements, were authored, and prioritized in a backlog. The development team then assessed the backlog and committed to completing the highest-priority user stories within the sprint (Sacolick, 2022).



**Figure 3.1 Scrum** (One Beyond, 2024)

### 3.2.1: Product Backlog

The Product Owner will identify and prepare all the requirements and features that need to be in BudgetBuddy (Wrike,Inc, 2024). Some of the requirements and features could include M-Pesa integration, AI based financial coach, support for more than one language (Kiswahili and English), offline mode, gamification for savings plans, advanced reporting, bill reminders and advanced security features.

### 3.2.2 Sprint Planning

The Scrum Team and the Product Owner plan the next Sprint, the items in the Product Backlog are considered for inclusion in the Sprint (Wrike,Inc, 2024). For BudgetBuddy the Product Backlog will be considered and included in the Sprint things like M-Pesa integration setup or the AI based financial coach development.

### 3.2.3 Sprint Backlog

The tasks selected for implementation in the Sprint are now moved to the Sprint Backlog. The Scrum Team now knows exactly what tasks need to be completed (Sprint Backlog) to deliver a potentially shippable product increment at the end of the Sprint (Wrike,Inc, 2024). For BudgetBuddy, the Sprint Backlog have tasks like Develop M-Pesa transaction flow and user experience, design the user interface for the budget tracking feature or Initial setup of AI algorithms for financial coach suggestions.

### 3.2.4 Sprint

The Scrum Team implements the selected items in the Sprint Backlog within the Sprint time-box. The team holds a Daily Scrum (a short meeting held every day before they start work) to review progress made and any impediments the team may have (Wrike,Inc, 2024). For BudgetBuddy the Daily Scrum could be used to report things like progress on the M-Pesa integration, problems being experienced in the offline mode implementation or updates on the user interface designs.

## **3.2.5 Tested and Integrated Product Component**

By the end of each Sprint the Scrum Team should have a potentially shippable product increment. This means that the features (selected items from the Product Backlog) developed during the Sprint should have been tested and integrated into the BudgetBuddy app (Wrike,Inc, 2024). For example, at the end of a Sprint the M-Pesa transaction flow should be fully operational or the AI based financial coach should be able to give basic suggestions to users who have provided details for him.

### 3.2.6 Sprint Review

The team holds a Sprint Review at the end of each Sprint to showcase the potentially shippable product increment (new features of BudgetBuddy) to the relevant stakeholders and get their feedback on the product. For each Sprint the Scrum stages above are followed and at the end of it all BudgetBuddy would be a fully developed app with features that effectively meet the needs of its users. Following the Scrum stages provides a structured, iterative and adaptive approach to the development of BudgetBuddy.

### 3.3 Tools to be used

The following diagrams will be used use-case diagram, context diagram, data schema, sequence diagram, class diagram, data flow diagram (DFD), entity-relationship diagram (ERD). Draw.io will be used to draw the diagrams. For the front end, HTML will be used to define and structure content on the web pages. Visual studio as the code editor. CSS (cascading style sheets) to style and define the layout of the web pages created. JavaScript as the programming language. For the back end, PHP (hypertext pre-processor), to handle the administration of MySQL over the web. Xampp will be used for database.

### 3.4 Reasons for choosing Scrum Model

It lets the team adjust fast to changes and user feedback. This is key for an app like BudgetBuddy, where user needs and tech changes fast. The model brings on-going betterment through frequent Sprint Reviews. It helps to find and fix issues fast, and makes the app better over time. Also, Scrum helps to ship things faster by splitting the project into small sprints. Daily stand-ups and Sprint Reviews help with hard features like M-Pesa and AI-powered money help. The focus on user needs and feedback aligns the app with real user needs. And the clear steps and early risk control make for a strong and safe product. In the end, Scrum gives a strong frame for the smart and quick making of BudgetBuddy, (One Beyond, 2024).

### 3.5 Deliverables

This explains the app's development process as well as the completion deadlines for the various milestones. It also provides an estimate of when the system will be ready. It depicts the process of putting the system in place.

### 3.5.1 System Modules

Budget Buddy’s system architecture and modules will give a detailed blueprint of the innards of the platform. The breakdown of system modules which are designed to manage core budgeting functions like tracking budgets, classifying expenses, creating reports and interacting with users will be discussed here. These modules should be interconnected in such a way as to allow for smooth data flow within the system.

### 3.5.2 Users and Roles

Users and their respective roles will be clearly defined to accommodate different levels of administration and regular users, among other roles that may be required. Administrators will wield a general grasp of control, dealing with user access, permissions, and system configurations. Regular users are going to interact with the platform mostly for budget management, where the access levels to each are designated appropriately to keep the integrity of the data under control, as well as privacy.

# **Chapter 4: System Analysis and Design**

## **4.1: Introduction**

This chapter provides a detailed description of the proposed system’s structure, highlighting system analysis and design to facilitate better understanding. It also discusses the structure, behavior, and perspectives of this system through different modeling methods: use case diagramming, class diagramming, sequence diagramming, entity-relationship diagrams (ERDs), and database schema. Furthermore, it explains the data lifecycle within the system; such as data input, authentication, processing and display as well. For a comprehensive illustration of how various components are related to each other in a given system, context diagrams are used together with data diagrams and DFDs.

## **4.2: System Analysis**

The first part of the chapter focuses on the system requirements for BudgetBuddy. It introduces the financial management application’s specific requirements and needs. The functional and non-functional components of this software are covered in the system requirements, thereby providing a complete understanding of what constitutes effective implementation and operation.

### 4.2.1: Functional Requirements

The functional requirements of the budgeting system are covered in more detail in this section. It offers a description of the precise actions and abilities that the system is capable of.

Below are the requirements:

1. Users and Admin create an account, log in, validate email address and provide options for password recovery.
2. Users and Admin update their personal information and upload a profile picture.
3. Users add new transactions and specify the type
4. Users view a list of their financial transactions.
5. Users edit the details of existing transactions.
6. Users set their budgets and financial goals.
7. Users view their reports to analyze their financial data.
8. Admin manages users and can generate reports.

### 4.2.2: Non-Functional Requirements

Performance standards are crucial, requiring the system to maintain swift response times and robust handling of concurrent user interactions.

User interface and user experience. The interaction elements of the system are designed to be user-friendly and meets users’ needs.

Reporting. The system generates reports, e.g., sales report.

Reliability is essential for ensuring continuous availability and resilience against potential system failures or unexpected errors, thereby preserving user data integrity.

Maintainability focuses on modular design principles and comprehensive documentation, facilitating efficient updates, troubleshooting, and system maintenance.

Compatibility across various platforms and devices is crucial to ensure that the system operates seamlessly for all users, regardless of their preferred technology environment.

## **4.3 System Design**

System design is the process of designing the elements of a system such as the architecture, modules, and components, the different interfaces of those components, and the data that goes through that system. In this project, the diagrams used for defining and developing the system are Data Flow Diagrams (DFDs) and Entity-Relationship Diagrams (ERDs).

## **4.4 System Analysis Diagrams**

System analysis diagrams are diagrams used to capture high level business processes and early models of system behavior and elements. They provide a useful means of capturing the essential system characteristics and requirements.

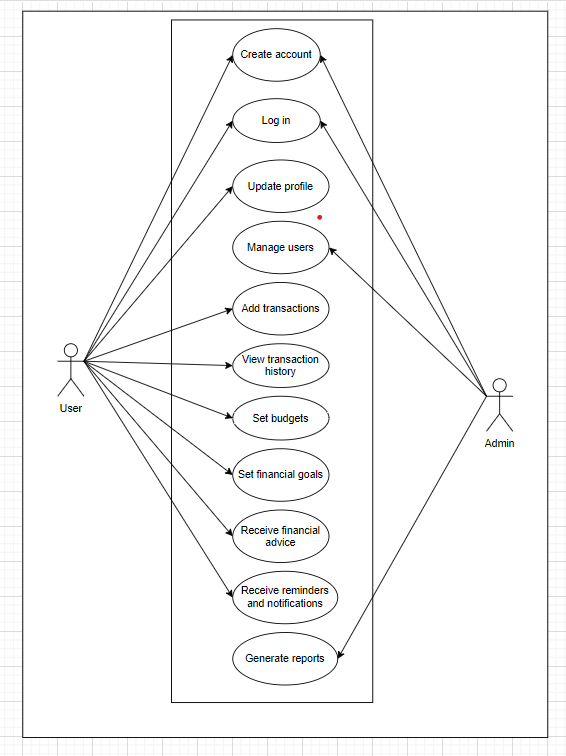
A system diagram is a visual model of a system, its components, and their interactions. With supporting documentation, it can capture all the essential information of a system’s design.

**4.5 System Design Diagrams**

A system design diagram is a visual model of a system, its components, and their interactions. With supporting documentation, it can capture all the essential information of a system’s design.

### 4.5.1 Use Case Diagram

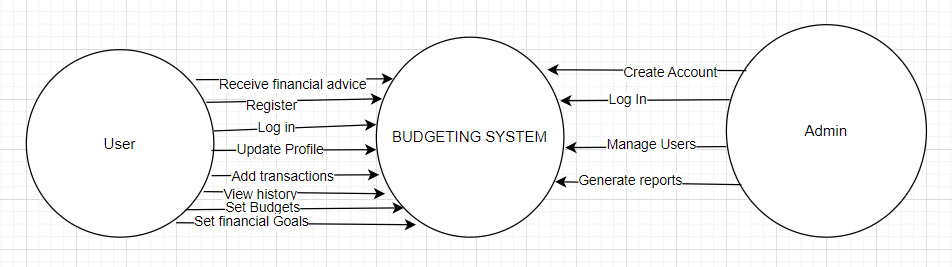
A use case diagram shows how users interact with the system and what features it offers. It is like a visual guide that outlines everything the application can do. In the diagram, we have different actors, like the main user who manages their personal finances and an admin who takes care of system stuff and user data. Each use case represents a specific function of the app, such as registering or logging in, handling transactions, checking transaction history, creating budget reports, setting and tracking financial goals, getting reminders and notifications, accessing AI-powered financial coaching, receiving personalized advice, managing offline features, and checking out the user dashboard.

****

**Figure 4.1 Use case diagram**

### 4.5.2 Context Diagram

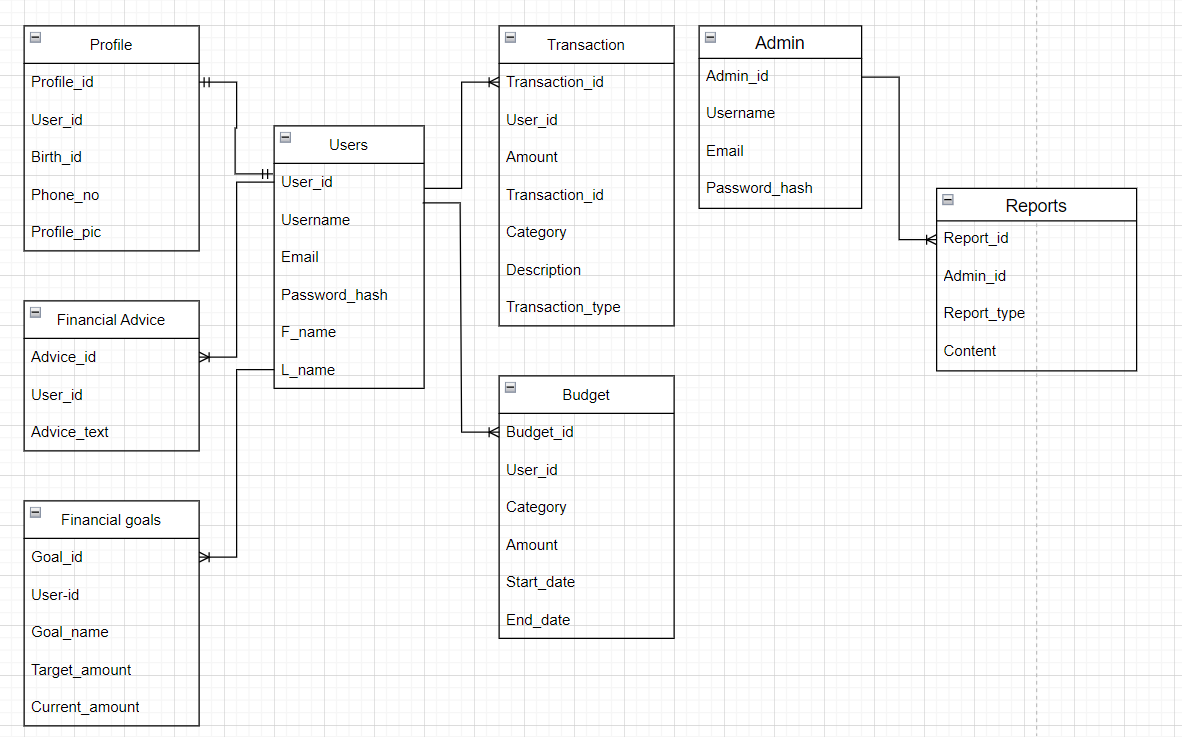
This diagram shows how a budgeting system interacts with its users and administrator. Users can do a bunch of things with the system like signing up, logging in, updating their profile, adding transactions, checking their history, setting budgets, and setting financial goals. They can even get some financial advice from the system. On the admin side, administrators can create accounts, log in, manage users, and generate reports. The diagram gives a clear picture of how the users, administrators, and the budgeting system all work together, showing how everything flows and what the system can do.

****

**Figure 4.2: Context Diagram**

**4.5.3 Data Schema**

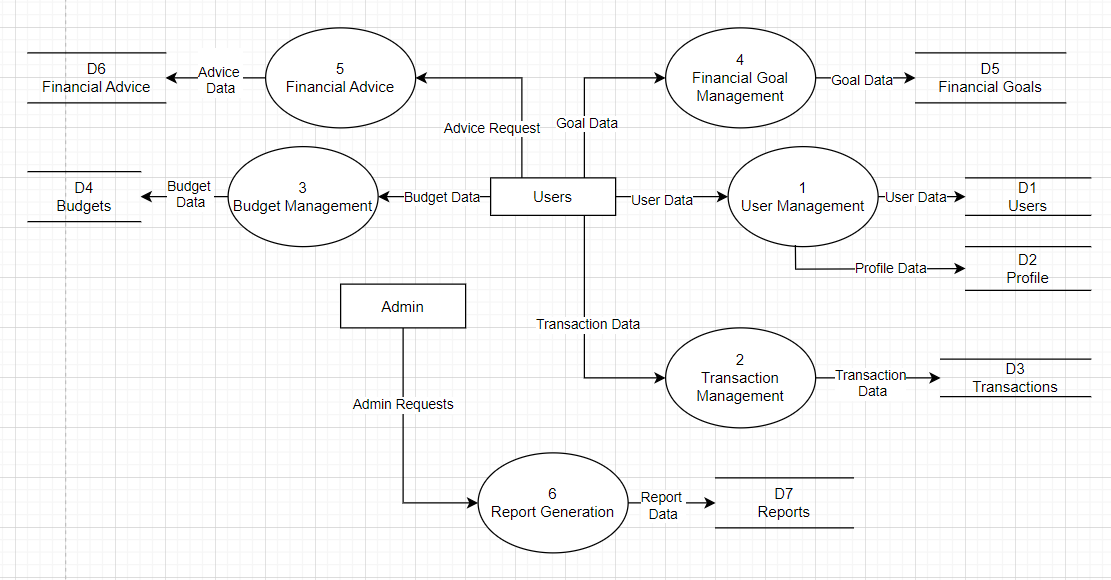
This shows a money system, joining user info, money moves, budgets and money targets. The `users` contains information like `username`, `email`, and `password\_hash`, linked to the `Profile` table for more info. User transactions are recorded in the `transaction` table, while the `Budget` table deals with budget categories and amounts. Financial goals are kept in the `financial goals` table, noting target and current amounts. Personalized financial advice is stored in the Financial Advice table. The admin table holds admin credentials, and the Reports table documents various reports. Relationships between these tables ensure data integrity and efficient retrieval, making BudgetBuddy a comprehensive financial management tool.

****

**Figure 4.3 Data Schema**

### 4.5.4 Data Flow Diagram Level 1

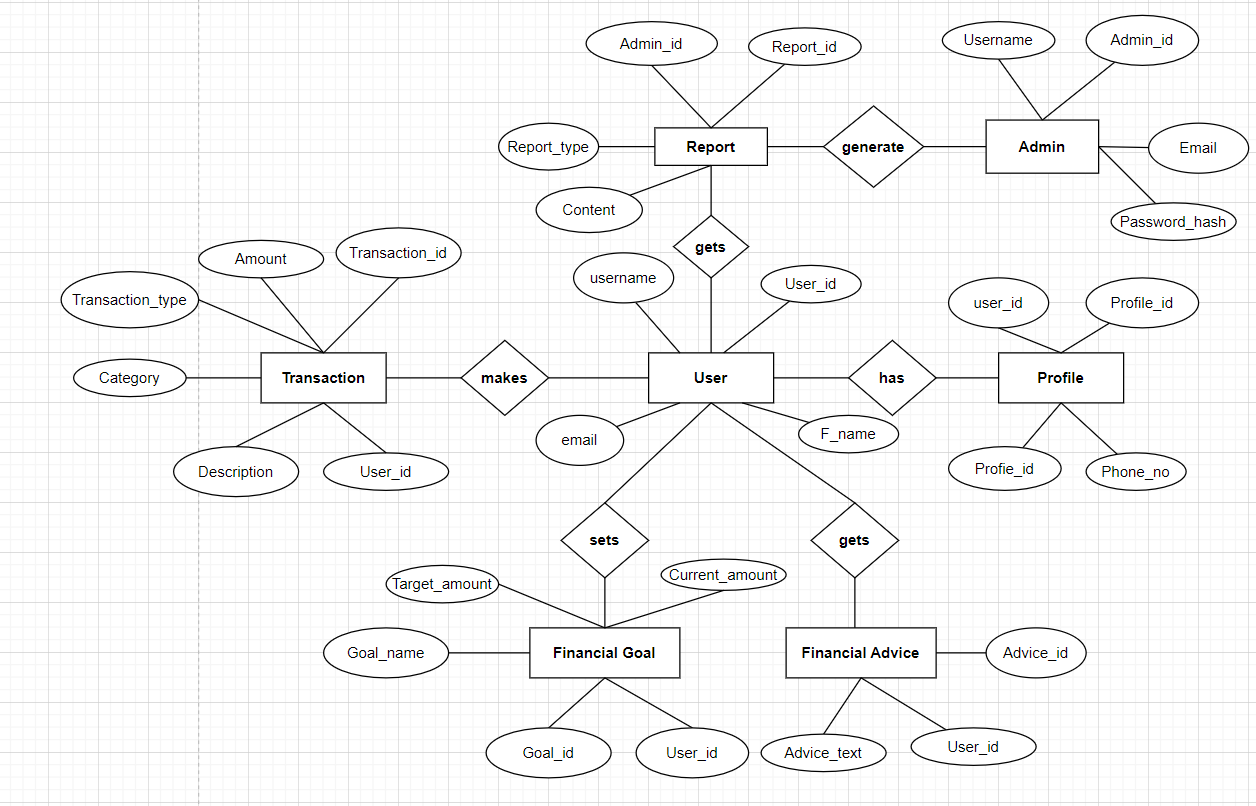
The Data Flow Diagram (DFD) depicts the interaction between different processes, data stores, and external entities within the system. The central entity, "Users," interacts with several key processes including User Management, Transaction Management, Budget Management, Financial Goal Management, Financial Advice, and Report Generation. User Management (Process 1) handles user data (D1) and profile data (D2), while Transaction Management (Process 2) deals with transaction data (D3). Budget Management (Process 3) manages budget data (D4), and Financial Goal Management (Process 4) oversees goal data (D5). Financial Advice (Process 5) provides advice data (D6), and Report Generation (Process 6) generates report data (D7). The admin entity can make requests to the system, primarily affecting Report Generation and indirectly impacting other processes. Each process exchanges specific data types with corresponding data stores and external entities to maintain the system's functionality and integrity.

****

**Figure 4.4 DFD 1**

### 4.5.6 Entity Relationship Diagram

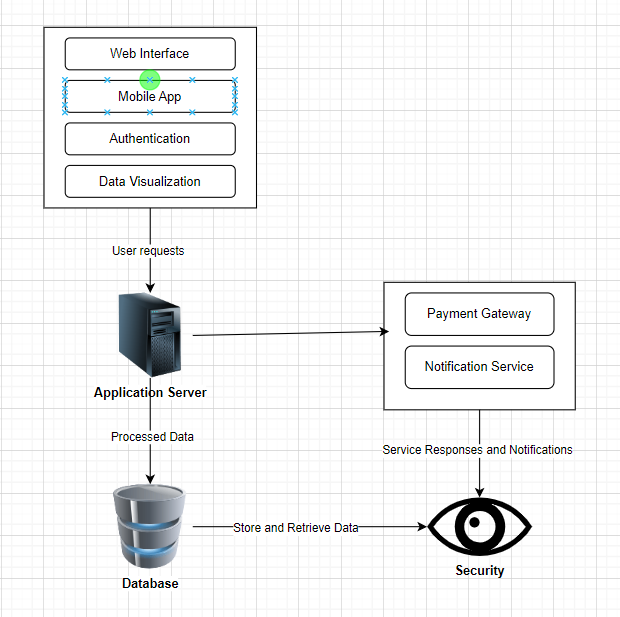
The ERD for BudgetBuddy indicates a structure showing the relationship between different system entities and their behavior. Emphasis is put on the interaction and data flow between users and their financial management activities. Positioned in the center of the diagram you will notice the entity named User, components User has relations which include Transaction, Financial Goal, Financial Advice, Profile as well as Report. For the User, these are interacting components that administer and offer functions above and beyond a handful of transaction features. Users can manage transactions, set financial goals, seek advice on financial matters, create and maintain a profile as well as view reports. Likewise, a separate yet closely related table to Transactions and Profiles may be that of Financial Goals. It wraps around Transactions seamlessly, as the transaction merely isn't just a purchase, but also invests in attaining a financial goal. This is detailed across aspects such as category, type, and more. Financial goals encompass the targeted amount, as well as the current amount, whereas financial advice has an array of specific recommendations. More user-specific data is stored in Profiles such as User info like name and phone number, while the entirety of details needed by the business is accessible from the transaction summary. Admins put together in-depth insights that are categorized by Report and are also viewed in unison with specific Users. Lastly, the ERD doesn't stop with User centric features but goes on to show Admins who are overlooking each application and establishing the rules for the system is happening.

****

**Figure 4.5 ERD**

### 4.5.7 System Architecture

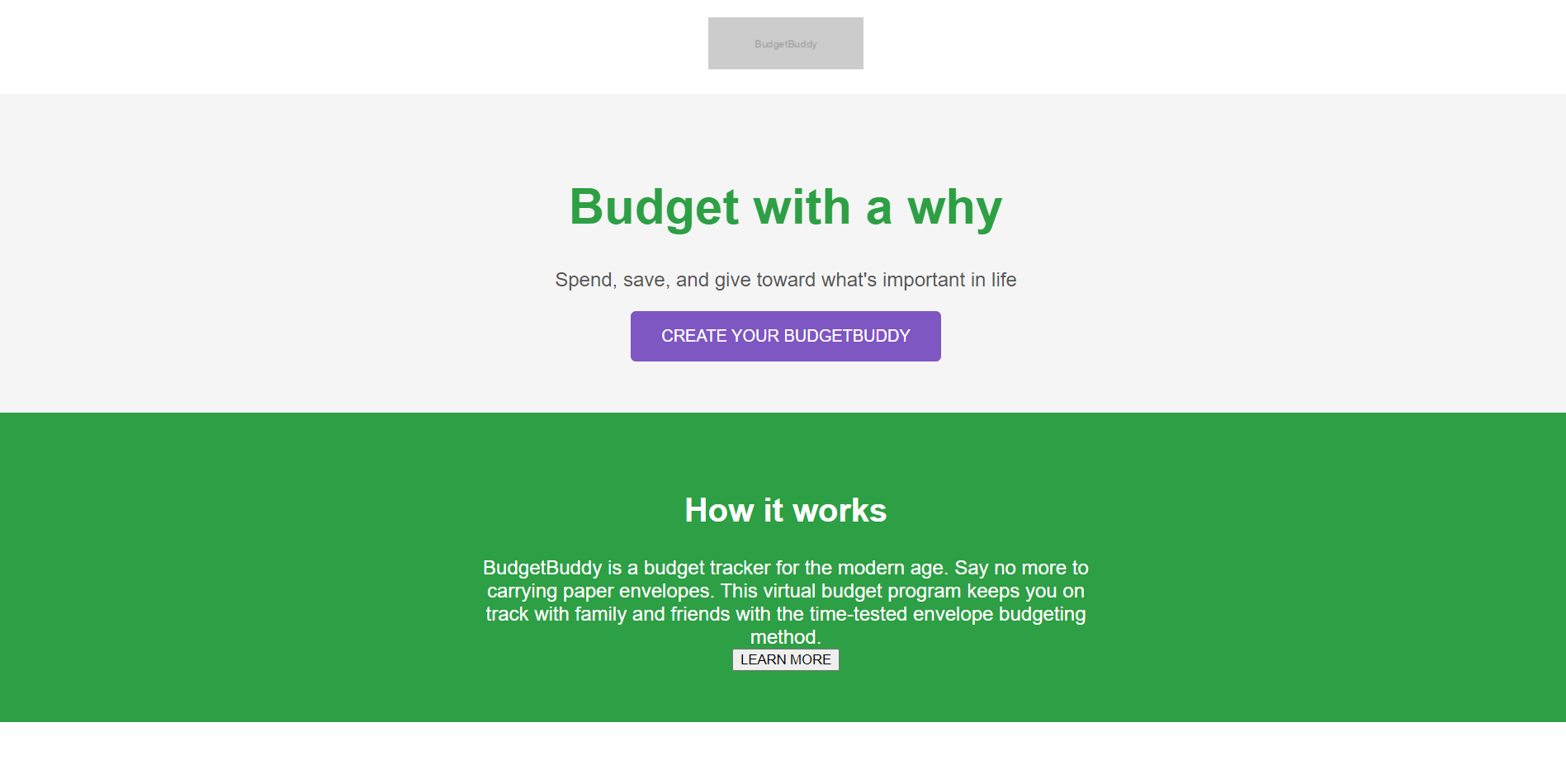
The application at the system architecture chart is for budgeting. In the diagram, we can see that a three-tier framework has been used. The three pieces are tightly interconnected processes representing same overall function service request and transaction cooperation. Next in the process diagram, is the connectivity spot. The user interface tier includes those traditional web-interface, including some sort of mobile application, a system to handle requests on the users. These user requests go to the server where the proper processing on their data is ensured. In the next tier, the user application server gives the data base for keeping and fetching data corresponds with the end user. It also interacts with external services including a payment gateway for buying and an issuing service to deliver updates to the customer. Later, security is throughout the network. Hiding the interaction of various sublayers to the last in every component. This system was built with the intention of creating a robust enough, secure, and easy-to-use budget tracking application, built for ease-of-use and handling complex transactions with easy-to-read real-time dashboards.

****

**Figure 4.6 System Architecture**

### 4.5.8 Wireframes

A Wireframe is a blueprint and visual guide that represents the skeletal framework of a website. It shows the page’s interface and specifically focuses on space allocation and prioritization of content.

****

**Figure 4.7 Wireframe**

# **Chapter 5: System Implementation Testing**

## **5.1: Introduction**

This chapter describes the system implementation and the testing that was conducted for the system modules to verify the system. It also describes the testing, the testing paradigm and outline of the test results.

## **5.2: System Implementation**

The system was built using PHP and MySQL, with Visual Studio Code as the development environment. CSS was employed for consistent visual design. The system stores textual and image data in a MySQL database. Authentication features, including registration and login, were implemented as the foundation, followed by the development of the system's core functionalities.

### 5.2.1: Systems backend

The system’s backend comprises of the logic behind the main functionalities of the system as shown below.

#### **Login sessions**

The below screenshot shows how a user is logged in by verifying their email and password against a database. If the conditions are met, it sets a session variable with the user's email, indicating a successful login, and returns an array with a success message and status. If the credentials are invalid, it returns an array with an error message and status.

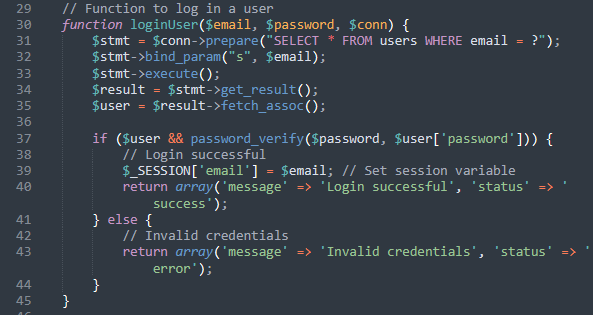


Figure 5.1: Login session

#### **Register sessions**

The below screenshot shows how to register a new user by first checking if the provided email already exists in the database. The email and hashed password parameters are bound to the prepared statement as strings, which is then executed. If the execution is successful, a success message is returned; otherwise, an error message is returned.



Figure 5.2: Register sessions

#### **Dashboard sessions**

The screenshot below shows the code behind the dashboard.php file, where the features of BudgetBuddy are displayed.

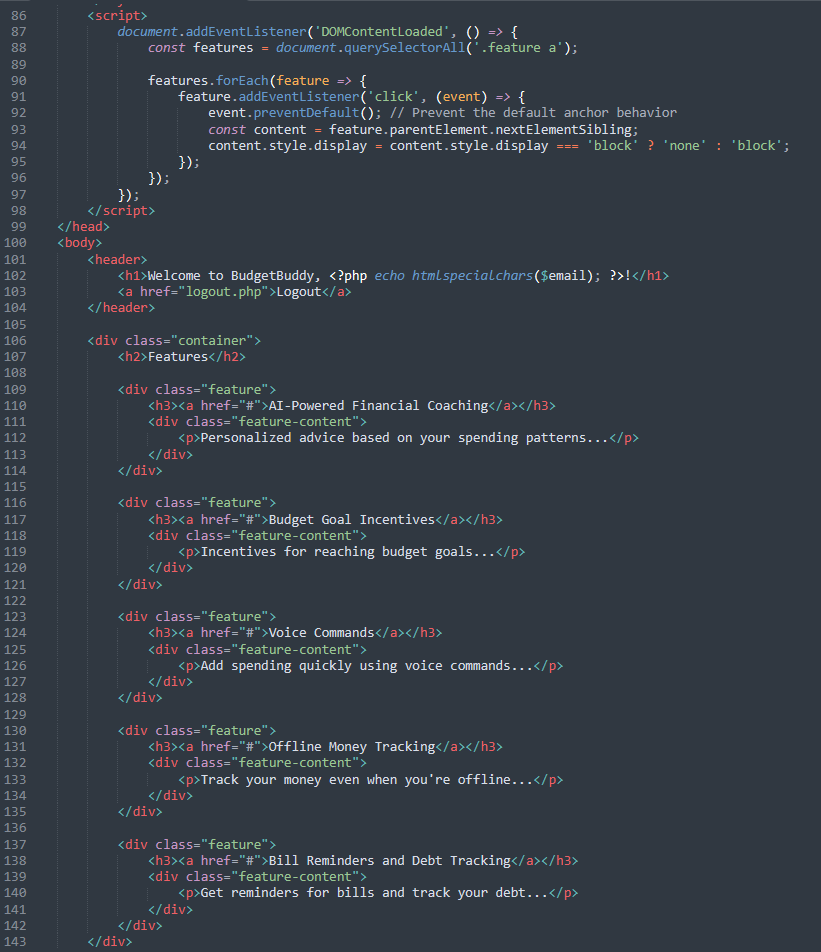


Figure 5.3: Dashboard session

### 5.2.2: System’s front end

The system’s frontend comprises of the logic behind creating the user interface of the system as shown below.

#### **Home page**

The below screenshot shows how the user interface for the home page looks like.

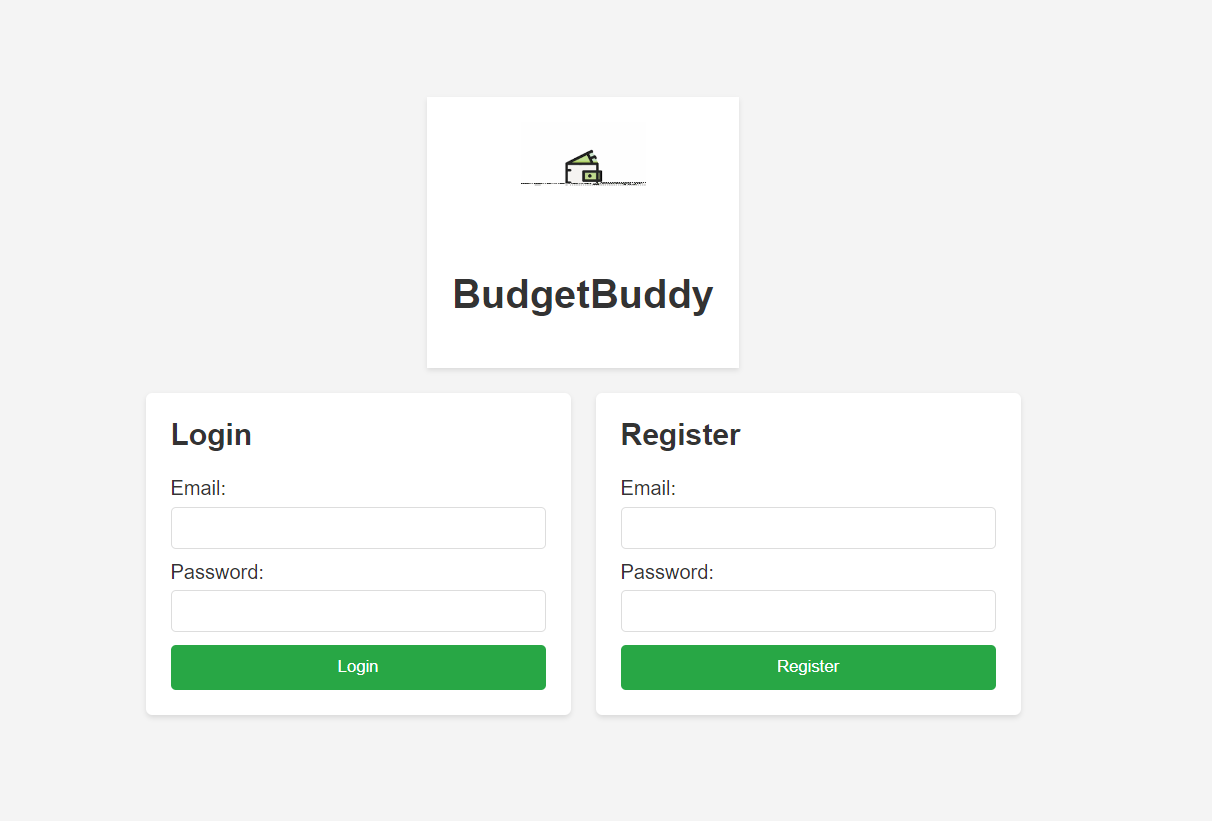


Figure 5.4: Home Page

#### **Admin Dashboard**

The below screenshot shows how the super-admin’s interface looks like.

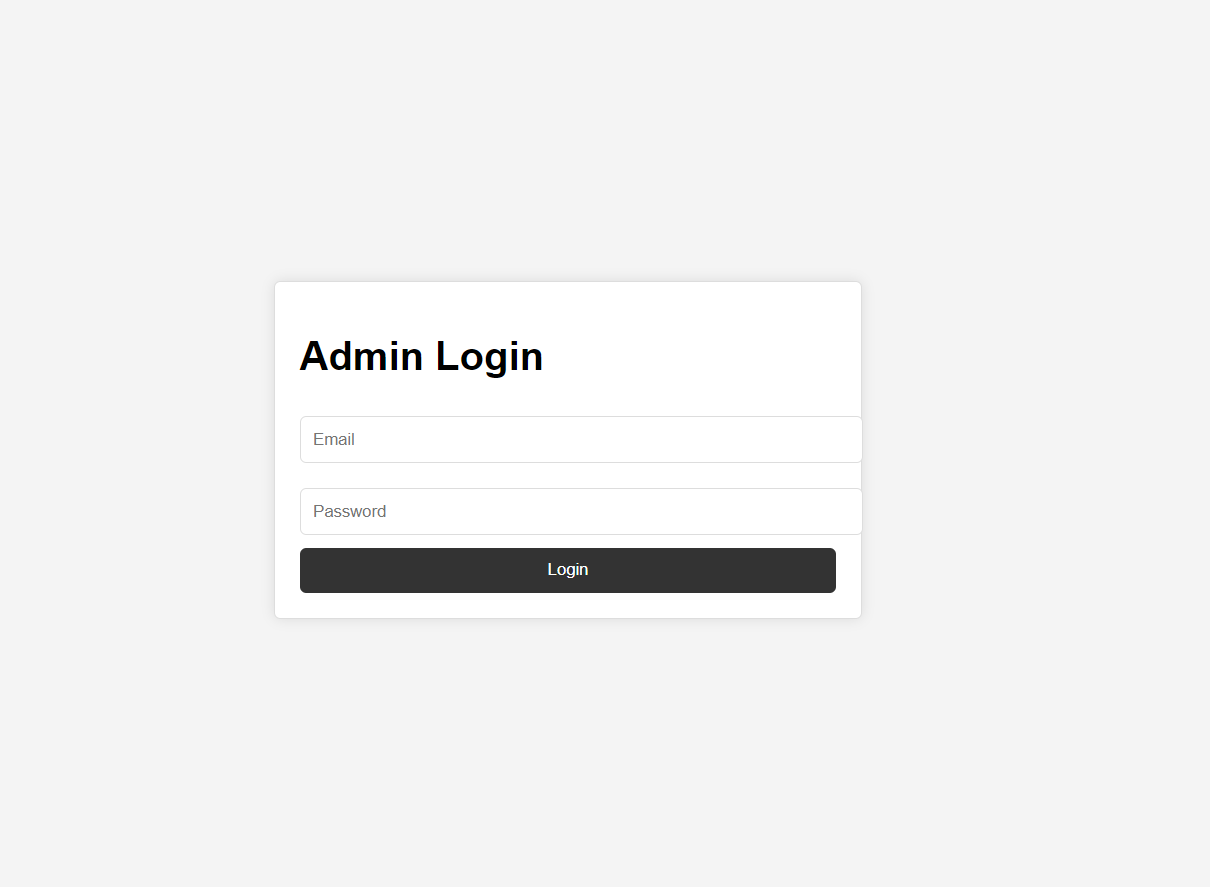


Figure 5.5: Admin Login

### 5.2.3: Database Tables

#### **5.2.3.1: Users Table**

This screenshot shows a database table named `users` within the `budgetbuddy` database, as viewed in phpMyAdmin. The table contains three columns: `id`, `email`, and `password`. The user interface offers options to edit, copy, delete, or export the data in the table.

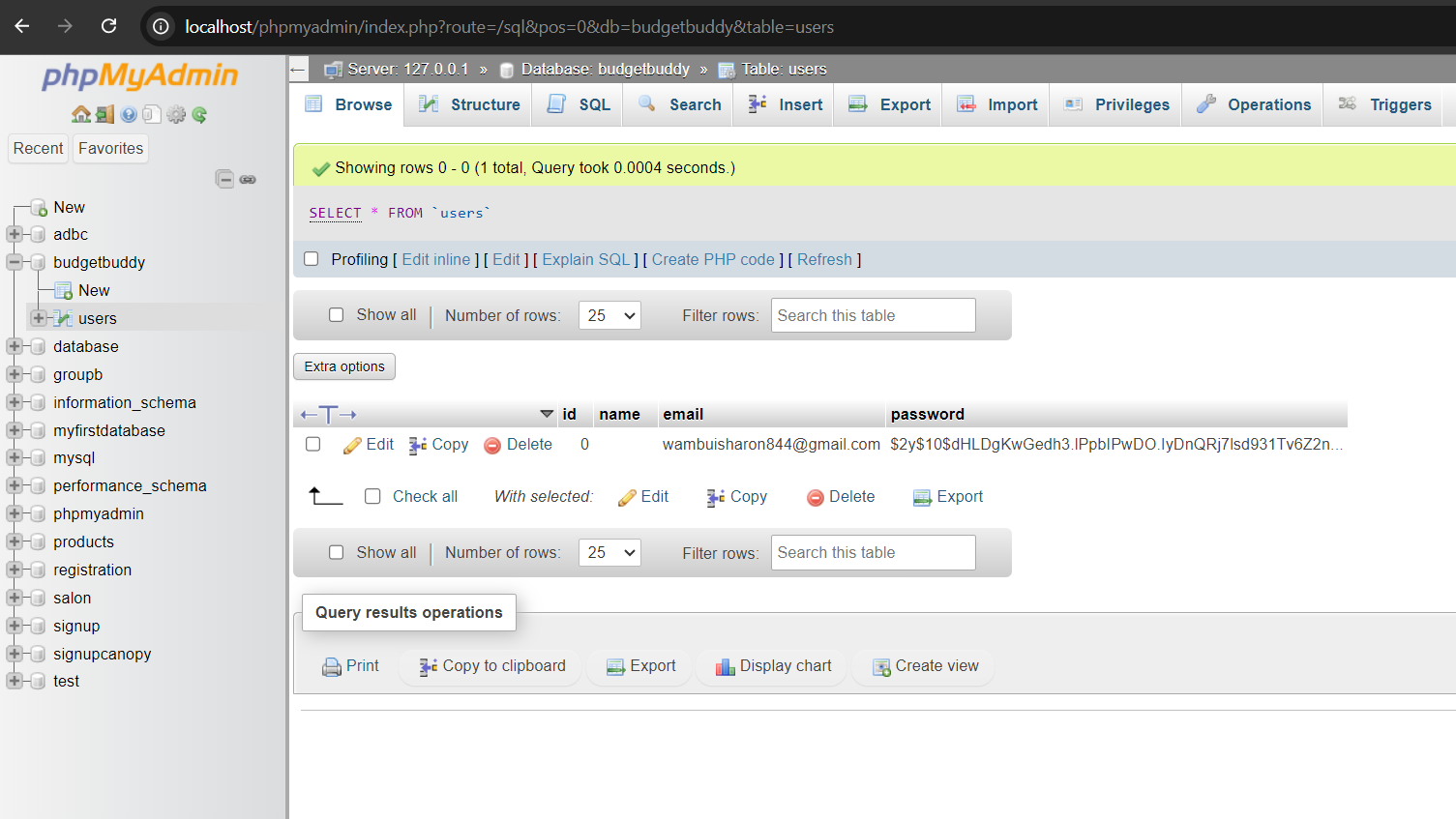


Figure 5.6: Users Table

#### **5.2.3.2: Admin Table**

The screenshot displays a phpMyAdmin interface connected to a MySQL database named "budgetbuddy."

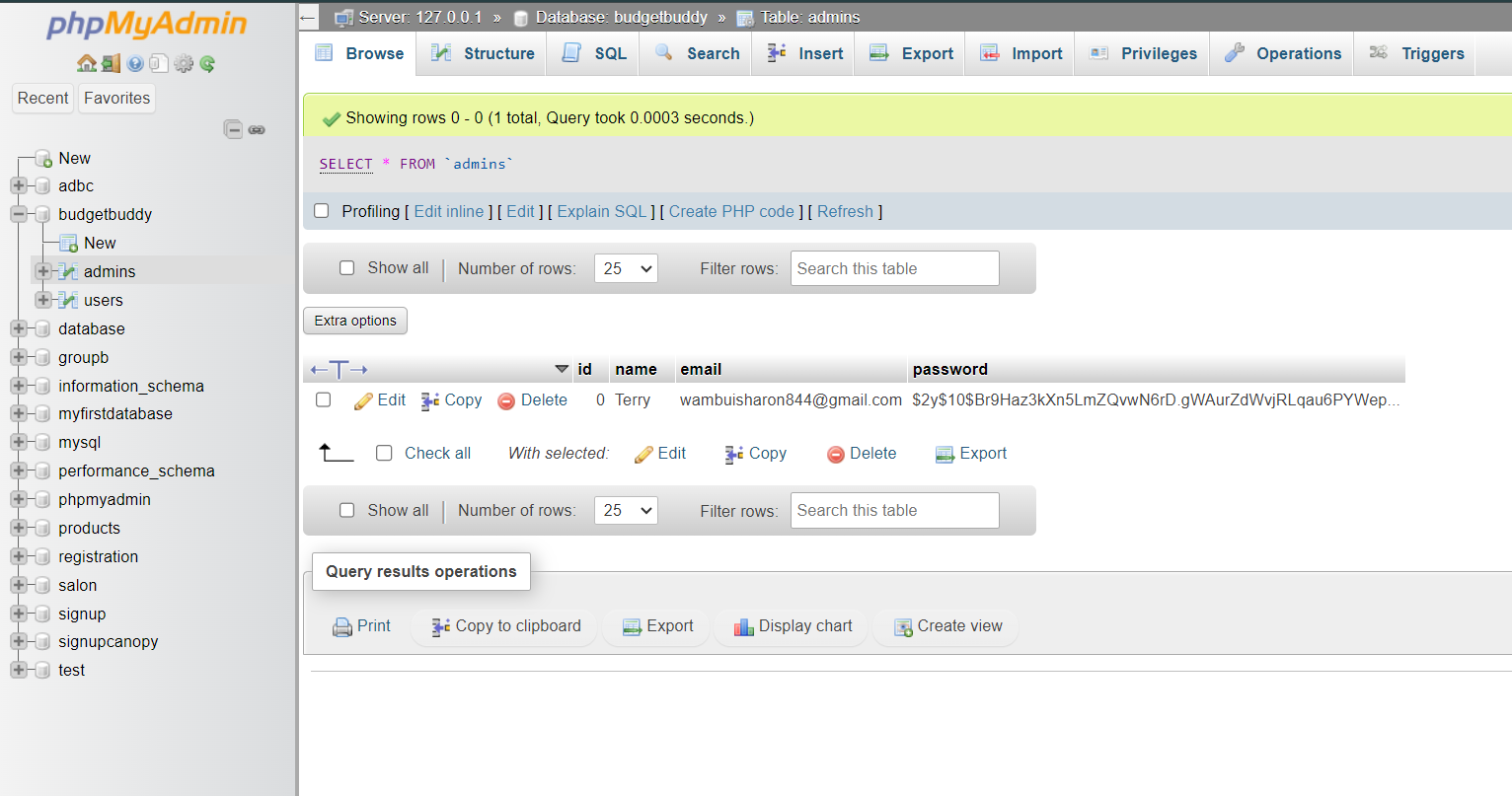


Figure 5.7: Admin Table

## **5.3: System Testing**

A critical step in the software testing process, system testing assesses the entire system as a whole. Its objective is to confirm that the system complies with the stated functional and non-functional requirements, functions as intended, and is prepared for deployment. Testing for the developed system was done by providing the correct input and checking whether the system behaved as expected. Incorrect input was also provided to see whether the system had error handling. All modules were tested to confirm whether they met the requirements.

### 5.3.1: Black box testing

Black box testing is a software testing technique that focuses on evaluating the functionality of a system without examining its internal code or implementation details. It treats the system as a black box, where inputs are provided, and outputs are observed, without considering how the system processes the inputs or generates the outputs. The goal of black box testing is to assess the system's behavior and ensure that it meets the specified requirements. Testing was done for all modules implemented in the system. The expected output was compared with the system’s actual output to understand the system’s behavior and identify possible errors.

### 5.3.2: Functionality testing

The system satisfies all the functional requirements by allowing farmers to register their farms and allowing only authorized users to login to the system. The system also allows users to log out successfully. It also allows the super admin to approve or delete accounts created by farmers and users. The system is able to store the information that users input in the local MySQL database, satisfying the storage requirement. The use of MySQL keeps the users’ passwords safe, even from administrators, thus satisfying reliable information to be loaded.

## **5.5: Test Cases**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case # | Description | Test Data | Expected Outcome |
| Registration Test | Registration with all required info | Field Form | Registration successful |
| Registration without filling all required field | Registration not successful |
| Login Test | Login with right credentials | Email  Password | Login is successful |
| Login with wrong credentials | Login not successful |
| Services | Check services | Choose service | Chosen service must display |

## **5.6: Test Results**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case # | Description | Test Data | Expected Outcome | Actual Result | Verdict (Pass or Fail) |
| Login Test | Login with the right credentials | Username  Password | Login is successful | Login successful | Pass |
| Login with wrong credentials | Login is not successful | Login failed, invalid username or password | Pass |
| Registration Test | Register a wrongly formatted email | Email | Enter correct email | Failed | Pass |

# **Chapter 6: Conclusions and Recommendations for Future Work**

## **6.1: Conclusions**

BudgetBuddy system was developed for the users and it includes AI powered financial coaching, budgeted-goal incentives to help setting goals easily, voice control on budgets also offline tracking of transaction amongst others. This conventional system of budgets constantly lacks in offering personalized and secured financial advice at the right time which results to ineffective overall expenditures planning & management. Tracking Spending, Budgeting Realistically and Staying Motivated: It can be challenging for users to track their expenses as well letting them set realistic budget goals or staying motivated while adhering to financing plans. To combat these issues, BudgetBuddy offers a complete solution in the form of an easy-to-use platform that tracks expenses more accurately and provides intelligent reminders & insights to users so they can better execute on their financial goals than ever before using existing systems.

## **6.2: Recommendations**

For the BudgetBuddy system to function optimally, it is recommended to use reliable web browsers such as Chrome, Mozilla Firefox, and Safari. Devices accessing the system should have a stable internet connection to ensure seamless functionality. Additionally, users are encouraged to regularly update their browsers and devices to the latest versions for the best user experience and enhanced security.

## **6.3: Future Work**

While BudgetBuddy is already changing the game when it comes to personal finance management, there are a ton of areas that future developers can continue expanding on. Improvements might involve setting up a financial approval process where users are required to verify their money through an identity document and in this way mitigating the chances for fraudulent activities. Furthermore, the system can be strengthened by enabling stronger authentications like minimum password lengths, email verification and two-factor authentication in order to secure users finance information as well. There is also potential to harness machine learning algorithms in future iterations, which would provide users with even more accurate and customized financial advice and predictions so that they can make truly informed decisions about their finances.

# **References**

Bank of America Corporation. (2024). Why and how to keep your personal and business finances separate. *Bank of America*.

Edgell. (2008).

Fernando, J. (2024). Key Takeaways. *Financial Literacy: What it is, and why it is so important to teach teens*.

Financial Services Authority. (2023). Benefits of Tracking Expenses You Should Know. *BCA*.

Hezretov, M. (2018). Motivation. *Budget Tracker Highly Customizable Budgeting Mobile Application*, 64.

Monefy. (2021). *Handy personal finance management tool for IOS and Android*.

Money Manager. (2010). Screenshots. *Money Manager Expense & Budget*.

Office, I. (2023). Bookkeeping Software vs. Manual Bookkeeping: Which is Right for Your Business? *inLinkedin*.

One Beyond. (2024). Agile Software Development Methodology. *One Beyond*.

Sacolick, I. (2022). What is Agile Methodology. *Modern Software Development* .

tala.co. (n.d.). Tala. *Tala App*.

University of Illinois at Urbana-Champaign. (2018). Many young adults lack financial literacy, economic stability, study finds. *Science Daily*.

Verma, E. (2018). Project Documentation and Its Importance in 2023. *Project Documentation and Its Importance in 2023*.

Wong, C., & Najib, M. (2022). *Personal Finance and Budgeting Mobile*, 16.

Wrike,Inc. (2024). Scrum Methodology. *Wrike*.

# **Appendix**

**Figure 7: Gantt Chart**