DESIGN AND IMPLEMENTATION OF A WEB-BASED BUDGET AND EXPENSE MANAGEMENT SYSTEM

BY

OWOSENI, SAMUEL KOREDE

(17CH023283)

THIS PROJECT WAS SUBMITTED TO THE DEPARTMENT OF COMPUTER AND INFORMATION SCIENCES, COLLEGE OF SCIENCE AND TECHNOLOGY, COVENANT UNIVERSITY OTA, OGUN STATE.

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE BACHELOR OF SCIENCE (HONORS) DEGREE IN MANAGEMENT INFORMATION SYSTEM (MIS)

JUNE, 2022

CERTIFICATION

It is hereby certified that this project, by Owoseni Samuel Korede (17CH023283), was supervised by me and submitted to the Department of Computer and Information Sciences, Covenant University, Ota, Ogun State, Nigeria.

Mrs. Theresa Abiodun
Supervisor

____
Signature and Date

Professor Olufunke O. Oladipupo
Head of Department

____
Signature and Date

DEDICATION

This project is dedicated to Almighty God, for his help, strength, wisdom and grace over my life. Father I am grateful. Also, to everyone that helped in making this project a success. Thank you very much.

ACKNOWLEDGEMENT

My profound and greatest thanks go to Almighty God, for always having my back through this entire process. For the strength, insight and health, Father I am forever grateful. You are the best.

I want to say a big thank you to my supervisor Mrs. Theresa Abiodun. Thank you very much for creating out time from your schedule to guide and assist me on this project. I would like to say a big thank you to the department of Computer and Information Sciences of Covenant University for the opportunity to study here.

Finally, I would like to say thank you to my parents, Engr. & Mrs. Owoseni for endless encouragement and support, you are the best parents I could ask for.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Inaccurate or irrational assumptions can render a budget unrealistically fast. Budgets can limit the flexibility of decision-making. Budgets must be revised as circumstances change. Most budget and expense management systems also disregard the need for an efficient monitoring and review process. The aim of this study is to develop a web-based budget and expense management system that will enhance planning and savings. The development of the system utilized an agile software development methodology. The frontend and backend of the web application will be constructed using Hypertext Markup Language (HTML), Cascading Style Sheet (CSS), JavaScript, PHP, and APIs. User data will be stored on the MySQL Server. This project emphasizes the need for adopting digital technology into personal budgeting and cost monitoring. However, it might be enhanced due to the project's limitations and implementation difficulties. For future work on this project, it would be more effective and efficient if the spending monitoring system was coupled with user bank accounts. I would also suggest that the system be set up as a mobile web app instead of a web app. This way, users will be able to use features that the web app doesn't have yet.

CHAPTER ONE
INTRODUCTION

1.1 Background of the study

Budgeting is a management method that individuals or entities use to ensure the financial dimension of their priorities, revenues, expenses, and outcomes at the management level and measure economic performance by comparing results outcomes (Hinojosa, 2018).

Budget Management System (BMS) was developed to make it easier for agencies, Treasury, and other stakeholders to exchange data, monitoring and reporting on the current fiscal year's budget, maintaining forward forecasts for future budget periods; and preparing annual Budget statements and other documents containing Budget data (Hinojosa, 2018). A budget control system makes it possible to keep track of individual finances, make accurate forecasts, or meet individuals' financial goals (Jadhav, Rutuja, Trupti & Damayanti, 2022).

Manual procedures have been used for budgeting, making it impossible to get budget demands from all company areas together. These manual procedures are based on creating non-standardized physical and digital records, which effectively obstructs the proper flow of information (Hinojosa, 2018).

A budget is a financial forecast for a set period of time, often a year. Additionally, it may include projected sales and profits, resource quantities, costs and expenses, as well as assets, liabilities, and cash flows. Businesses, governments, families, and other organizations utilize it to communicate their operations' strategic plans (Capucao, Palaoag, & Sierra, 2020).

A budget is a list of expected costs and ideas on how to meet them and the overall sum of money set aside for a particular purpose. A budget surplus leaves money on the table for future usage. The process of reviewing, arranging, and tracking an individual's costs and expenditures is known as budget management. External allocation guidelines must be followed when managing a budget (Capucao *et al*., 2020).

A budget, in most cases, allocates specific amounts of money to various funding needs. A budget also keeps track of income or profit. Maintaining positive cash flow while staying under the budget limit is a constant balancing act for budgeting. When a budget becomes unmanageable, the manager must figure out how to increase or decrease expenditures (Jeseviciute-Ufartiene, 2014).

Comprehensive budget management is a concept that refers to a set of complete, finished, and successful management programs focused on a business's internal budget. The plan begins with the organization's primary strategic objective, proceeds via a scientific budgeting procedure and a fair organization guarantee system, balances responsibilities, and integrates all areas of the organization's operations and administration entirely. It is a closed-loop system that includes defining objectives, budgeting, controlling execution, analyzing and evaluating the company's operations, and implementing management activities based on strategy decomposition and project planning. After many years of management practice and accumulation, most businesses have gradually developed a relatively perfect comprehensive budget management method based on the financial management business plan. Simultaneously, there is still inconsistency in the development of applications in terms of total budget management in IT support systems related to regulations, and lack of adequate support for business standards. Within the context of financial centralization, promoting and attaining budget management, accuracy and consistency have become inevitable requirements in the current environment (Ma, 2016).

The inability to provide a reasonable return on investment is a core corporate aim, and the budgeting process involves future profitability projections. A firm must have a plan to cope with the uncertainty of the future. A person or business that does not plan opts to deal with the future ad hoc and can only react to events as they occur. On the other hand, most businesses develop a plan for the actions they will take in reaction to predicted incidents. While the budget has grown in importance as a financial planning tool in contemporary businesses, there are specific persistent issues with present budget management: Due to a lack of information about budget management, the process of "editorial budget-budget, implementation-budget, and adjustment-budget analysis" cannot be carried out rigorously; The foundation work for budgeting is incomplete, and the specific evaluation indicator is unclear. There is no system in place to monitor and track the budget's execution. All of this demonstrates how critical it is to establish budgeting information management. (Yang & Wang, 2014).

Businesses might utilize budget-to-actual comparisons to evaluate individual performance. For example, the normal variable cost of creating a personal computer is budgeted at IBM. This number may be compared to the real cost of producing personal computers to aid in evaluating the performance of personal computer manufacturing management and employees. This type of comparison will be carried out in a subsequent chapter (Ugochukwu, Emedosi,Emoh & Chinagorom, 2020).

1.2 Statement of the problem

In recent years, manual budgeting techniques have been adopted, causing a significant difficulty in budget forecasting and leading to inaccuracy due to assumptions that are relatively comparable to the operational situation of businesses and individuals. The majority of budgets do not represent the entity's strategy, are cost-driven rather than value-driven, negatively affect the entity's flexibility and capacity to react to change, are bureaucratic, and limit creativity. Due to inaccuracy, decisions made at the end of each financial year and during the budget preparation process do not take into account unpredictable business activities.

Most budget and expense management systems also do not pay attention to how important it is to have an effective monitoring and evaluation process. This is a crucial part of performance-based budgeting and can be measured to ensure that people are held accountable and that decisions are made based on facts.

1.3 Aim and Objectives of the study

The aim of this study is to design a web-based budget and expense management system that will improve planning and saving. This will be achieved through the following objectives:

i. To review relevant literature on budget and expense management systems.
ii. To design and model the proposed system.
iii. To implement and evaluate the budget and expense management system.

1.4 Methodology.

The following methodologies will be used to achieve each objective:

- **Objective i: To review relevant literature on budget and expense management systems.**

  A literature review of past similar projects will be carried out by studying article, and exploring internet information. A study of existing systems will be done by registering on some of the platforms and interacting with them to identify their limitations and also checking out reviews on the internet.

- **Objective ii: To design and model the proposed system**

  Figma will be used to construct the system's design, and Unified Modeling Language (UML) will be used to model diagrams.

- **Objective ii: To implement and evaluate the budget and expense management system**

  The web application will be developed using Hypertext Mark-up Language (HTML), Cascading Style Sheet (CSS), JavaScript, PHP and APIs for the frontend and backend. MySQL Server will be used to hold user data. The system will be evaluated by testing the user experience and functionalities of the prototype of the budget and expense management system.

1.5 Significance of the study

The significance of this study is to help individuals resolve the problems associated with budgeting, saving, and planning. The solution supplied by this project will help to eliminate wasteful spending by incorporating monitoring, assessment, and appraisal process into existing budget and planning software or web application. It will provide better and more efficient operational processes, making it easier for the individual or business to analyze their previous budgeting activities and make proper decisions.

1.6 Scope and Limitations of the study

The scope of this study is to design a budget and expense finance management system that is more interactive and engaging and possibly closes gaps in existing budget and expense management systems. This project focuses only on budget management, leaving out other financial management forms. This study does not enable the product to sync to users' financial accounts.

1.7 Project Organization

This project is divided into five main chapters. Chapter One presents a general introduction, highlighting the problem statement, the study's aims, the methodology employed, the study's importance, and the project's limits. The second chapter summarizes numerous previous works, provides an overview of existing systems, and delves into various budgeting strategies. The third chapter discusses system analysis and design. It covers all the diagrammatic models that would aid in the system's construction. Chapter four implements and deploys the system based on the analysis and design presented in Chapter three. Chapter five is the conclusion chapter; recommendations for further study are included.

CHAPTER TWO
LITERATURE REVIEW

2.1. INTRODUCTION

Segment of selected literature review defines that we will discuss related previous investigations on this study. This chapter is focused on reviewing previously existing systems developed for Budget Management. The review identifies different approaches used and adopted in budgeting and recognizes the influence technology has on the planning and monitoring of budgets. It also specifies methods used to develop existing budget and expense management systems, reviews existing systems, and highlights their strengths and limitations.

2.2. REVIEW OF RELEVANT LITERATURE

Web-based systems are those that operate inside a web browser. It is a program that may be accessed using the Hypertext Transfer Protocol (HTTP). Online applications propose a reasonable approach to securely restoring unified data. Employees may be hired directly by the person or organization supervising them. Distributed computing allows employees to do the same thing repeatedly so that they do not lose personal time in the event of a disaster. We have a variety of web-based application samples (Lee, 2019) . Online forms, shopping carts, word processors, file conversion, file scanning, and email systems such as Gmail, Yahoo, and Outlook are all examples of Web applications. Google Apps and Microsoft 365 are both popular programs. Both individuals and businesses benefit from web-based systems since they help them with a variety of tasks: It helps individuals and business owners reduce the cost of updating software, they help to improve the security of online data, it helps to monitor monetary cycles, Web-based systems work with finance, they control the work process of individual staff and undertaking gatherings (Singh *et al.*, 2021).

In microeconomics, a budget refers to the exchange that occurs when one product is replaced by another. A surplus budget anticipates gains, a balanced budget anticipates revenues equaling expenditures, and a deficit budget anticipates expenses exceeding revenues—or the result of this trade-off. Budgeting is critical for tracking monthly spending, planning for life's unforeseen occurrences, and avoiding debt when making large-ticket expenditures. Maintaining an accurate record of your income and expenses does not have to be a pain. It does not require you to become a math guru, and it does not imply that you cannot afford the items you want. This simply means that you will have greater control over your finances and be aware of where your money is being spent. (Ma, 2016).

Governments, businesses, and individuals use budgets to estimate revenue and expenses over a certain period of time. A budget is a financial plan for a particular time period, typically a year, designed to improve one's financial situation dramatically.

A budget is essentially a sum of money set aside for a specific purpose. It is also known as a financial plan for a particular period. When we talk about budgeting, we are referring to the amount of money that a person has available to use and use for a specific purpose, whether it is for investment or simply to use. However, the issue is that people spend more than they save (Yang & Wang, 2014).

Governments were the pioneers of budgeting. They revealed a yearly account of income and expenditures that forecasted a surplus or deficit. As a result, it was evident that preparation was required to keep the results under control. In organizations, it gave control and restrictions. This strategy in business organizations developed gradually in popularity as the twentieth century progressed. The creation of a wide range of budgeting and administrative-accounting literature aided budgeting. Due to the importance of budgeting, the first International Conference on Budgeting was held in Geneva in 1930. This gave an opportunity for applied and speculative consultants to debate budget management approaches. Following World War II, the budgeting approach swiftly expanded throughout countries and businesses of all sizes. The popularity of budgeting has skyrocketed (Roy, 2020). There are four main types of budgets: incremental, activity-based, value proposition and zero-based budget. Each of these four budgeting strategies has its own set of benefits and drawbacks, which will be addressed in greater depth later in this chapter.

Zero-based budgeting, one of the most commonly used budgeting systems, means that all agency budgets are zero and must be constructed from the ground up. Managers are required to justify all expenditures. The objective of zero-based budgeting is to eliminate any costs that are judged unnecessary for the productive (profitable) operations of the business. When a business is in severe need of cost reductions, such as during a financial reorganization or a significant economic or commercial crisis that forces it to dramatically cut its budget, the zero-based strategy is the prudent course of action. Discretionary costs are easier to manage than non-essential operating expenses using zero-based budgeting. However, because it is time-consuming, many businesses only utilize it rarely(Ugochukwu *et al*., 2020).

Incremental budgeting begins with the previous year's actual data and then adds or subtracts an amount to arrive at the current year's budget. Due to its simplicity and clarity, it is the most frequently used budgeting tool. If the primary expense drivers remain constant year after year, incremental budgeting makes sense. Its advantages are its simplicity, accuracy, and speed. However, some of its flaws include re-inventing the wheel, not paying attention to important changes in the future, and assuming that the base is correct.

Budgeting by activity is a top-down method of budgeting in which the number of inputs required to support the company's goals or outputs is defined. Budgeting in this manner is only feasible if each activity has discrete divisions that clearly indicate how resources are allocated. When shared resources (such as people, facilities, etc.) are involved, implementing an activity-based budget presents a challenge. It will be important to employ precise resource allocation mechanisms such as staff time charges, square footage assignment, and utility consumption, among others. Several advantages of this budgeting method include the following: its resources are properly matched to its service delivery; it acts as a basis for unit pricing; and the costliest activities are emphasized. Among the disadvantages are the following: resource allocation may not be precise; it requires extensive labor to separate each activity and the resources utilized, which is not feasible for services that require a flexible approach and/or where resources must be transferred between activities to meet demand (Lysiak *et al*., 2021).

Budgeting for value propositions is just another way of saying that everything in the budget brings value to the business. Although value proposition budgeting is not as precise as our ultimate budgeting choice, zero-based budgeting, it aims to eliminate unnecessary expenses.

Financial planning and forecasting are used to identify the resources that will be allocated to achieving strategic objectives and action plans. Numerous businesses continue to spend substantial resources on suboptimal management practices that fall short of meeting their operational and strategic goals. Understanding budgeting and forecasting is critical, but mastering their implementation is even more critical. Forecasting and budgeting include four critical components that provide major advantages to any firm (Singh, 2021).

While it may seem obvious, budgeting is not simple. It takes time to review previous successes and identify cost drivers. It is critical to avoid setting unrealistic objectives. For instance, it may seem unrealistic to believe that income may increase by 60% or that expenditures can be reduced by 50%. As a consequence, it is critical to have a logical expectation of the budget's prospective outcomes. Creating a company is not a simple undertaking. However, individuals who take a prudent approach to fiscal planning may anticipate reaping the benefits of compounding success. Of course, a budget does not only assist growth; it also has other objectives. Thus, a budget may emphasize revenue growth above debt reduction. To build a reasonable budget, you must first establish reasonable, quantitative, and measurable objectives. The budget may then be transformed into a tool for real accomplishment (Singh, 2021).

Both forecasting and budgeting are intended to provide a framework for the fiscal year ahead. Unexpected events often break the normal business cycle. COVID-19 is an excellent example. As a result, it is critical to evaluate a range of competencies and conduct scenario analysis while developing budgets and projections. Guerrouj suggests one rule of thumb. Always begin with a budget and forecasting. This is a circumstance in which everything happens as planned. Following that, develop a set of budgets for the worst-case and best-case situations. This is an excellent technique to prepare for many possibilities when addressing budgeting in this manner (Singh, 2021).

Inaccurate data may cause significant issues with forecasting and budgeting. It is critical to begin with that which has been resolved and secured. Negative effects may occur as a result of profit and loss budget. Always utilize trustworthy facts when creating a budget. Budgeting is necessary for forecasting. This is a critical notion. If a projection is constructed on the basis of an inadequate or erroneous budget, it is a recipe for catastrophe (Singh, 2021).

It is critical to maintain a budget and not let it lay dormant on a desk. This is a strategy for achieving the business's objectives. The only way to ensure that your organization is heading in the right direction is to monitor its progress against the forecast and budget on a regular basis, including milestones in the budget and forecast is a good idea. Once these milestones are achieved, it is obvious that the firm is on track. These milestones demonstrate that the company is on track to meet management's objectives. Additionally, it may be used to evaluate employee performance. Occasionally, unanticipated occurrences occur that cause the organization's budgetary progress to be halted. It is critical to keep the forecast and budget current and to make any necessary adjustments (Ugochukwu *et al*., 2020).

Financial success for your business is dependent upon your skill and desire to define its operation as a commercial entity in the current economy. Your ability to apply current budgeting, planning and forecasting methodologies to your finance will determine your financial planning efficiency (Singh, 2021).

The web has changed the way that average citizens conduct their daily business. Individuals may access information on a variety of topics through the internet, place orders for things, and manage their bills over the web. The Budget Management System (BMS) was created to facilitate the sharing of data between departments, the Treasury, and other stakeholders for purposes such as budget analysis and reporting, forward forecasting for future budget periods, and the preparation of annual budget statements and other documents containing budget data. People who write a lot of things about the budget use data from the BMS. For example, the Cabinet and Treasury State of the Budget Reports and the Consolidated Fund Quarterly Statements are based on data from the BMS (Jeseviciute-Ufartiene, 2014). Web-based Budget Management Systems are web-based applications for budgeting and personal finance management. Budget and expense management systems on the web are online information systems that represent a concept that encompasses the overall management of an organization's activities. Individuals in the middle class lack the time necessary to manage their personal finances. Several of them are incapable of managing their income and results. While they recognize the need to seek counsel from an adviser, many are unwilling to pay for personal financial advising services due to the high cost. They require expedient and straightforward money management services (Mumpuni & Sukarno, 2014). It enables users to keep a detailed record of their spending, It removes human error in budgeting as a consequence of the process's pressure, It reduces the amount of time necessary to develop an accurate budget, It contributes to more transparency.

2.3. REVIEW OF EXISTING SYSTEMS

You Need a Budget, or YNAB, is a personal finance application that aims to assist you in managing your monthly budget while also increasing your financial literacy. The curriculum includes classes that will aid you in navigating some of the most challenging financial topics as you create your budget and manage your everyday finances. If you've been struggling to overcome unfavorable financial habits, YNAB can assist you by teaching you how to follow a few easy money management principles. Rather than utilizing typical budgeting techniques, you construct a budget based on your income, allocating each dollar to a specific activity. These positions have a variety of obligations, ranging from living expenses and recreation to debt repayment, savings, and investment. You must consider every dollar you earn and spend if you want to avoid leaving any money unaccounted for. The software is beneficial for individuals or couples who are budgeting together. It has both desktop and mobile interfaces, as well as the option to automatically sync your bank accounts or manually enter expenditures. In addition, it has tools to help you pay off debt and keep track of your progress.

Mint is a well-known free online personal finance system from Intuit that features a variety of easy-to-use financial planning and monitoring tools. Mint's free mobile apps for iPad, iPhone, Android, and Windows mobile devices complement the internet version. For many users, Mint is ideal because of its useful personal financial skills such as budgeting, transaction tracking, categorization, and bill reminders, but its absence of an account reconciliation feature makes it inappropriate for others. To begin using Mint, just establish an account and connect your bank accounts. Once your accounts are connected, the app will continue gathering transactions and data for several months. While the first operation will take a few minutes, subsequent updates will be automated. The program automatically categorizes transactions as they are received according to budget spending categories. While the automatic classification is frequently right, you may simply rename or reclassify transactions. Additionally, categorizing a transaction is uncomplicated. In addition, you can group transactions so that you can make more detailed financial reports or budgets. Users of Mint can take advantage of the following features: Easiness of use Budgeting tools that are adaptable allow for experimenting in a variety of settings. Email or text messaging is used to send financial summaries and notifications. Notifications of unusual account activity through email or SMS messages, bill reminders, At the bank level, data security is critical for account aggregation. customizable and the establishment of categories for expenditure and income. The Mint software, on the other hand, has a variety of downsides, including the following: There is no reconciliation of accounts available, and account registers do not maintain continuous balances. It does not support the usage of several currencies, making it impossible to link multiple savings goals to a single account.

PocketGuard may assist you in avoiding excessive expenditure. This budgeting tool connects to all of your financial accounts and helps you keep track of your monthly expenditure in relation to your budget. This is one of the simplest apps to configure and link to your bank accounts. It keeps track of how much money you make, how much money you spend on regular payments and daily costs, and how much money you put into your savings account. PocketGuard even has a tool that allows you to monitor individual bills and identify savings possibilities. It monitors recurring bills from phone, television, and Internet providers, which may help you identify more affordable solutions. Additionally, it may assist you in locating subscriptions you forget you had and provide suggestions for what you might want to cancel. Not only does this program assist you in tracking your expenditure, it also assists you in reducing your spending.

Goodbudget, previously known as Easy Envelope Budget Aid or EEBA, is an excellent choice for couples who want to collaborate on the budgeting process. It is based on the well-known envelope budgeting theory and helps you create a proactive budget for all of your expenses and expenditures. Envelope budgeting is the process of splitting each month's costs into envelopes for each budget category. You can share and sync budgets across many platforms (web, iPhone, and Android), which makes it ideal for couples who share expenses but also works well for single budgeters. When creating new transactions, you may provide a variety of information, effectively segmenting your costs into various envelopes. Budget by category (referred to in the app as envelopes), with up to ten envelopes included for free. Each paycheck, add to your envelopes and you'll know exactly how much money you have left over for non-essential items. The system, however, does not link with individual bank accounts.

Wally has stunning images and graphics and is just focused on budgeting without a slew of extra features. This software assists you in tracking your income and expenditures while also providing a picture of your remaining budget to assist you in avoiding overspending. This app, which is more popular among Millennials, is available in both free and premium editions. One significant advantage of Wally is that it has built-in support for practically all international currencies, making it a more attractive alternative for people who reside outside the United States. Wally does not track your spending or do any type of assessment. If you're looking for a straightforward budgeting software without all the bells and whistles, this may be the tool for you.

QuickBooks is a widely used accounting and budgeting application for small businesses that is used by accountants worldwide. It has a variety of capabilities, ranging from spending monitoring to invoicing, and enables businesses to get a thorough picture of their budgets and predictions. QuickBooks is totally devoted to the effectiveness of budgeting and expense management. This means that it is packed with a number of beneficial features. Spending tracking, custom invoice templates, invoice sending, bill management, and payment are just a few of the top features. Automated download and classification of bank transactions, profit and loss statements, balance sheets, and other data, such as sales and expenses.

BillQuick is a company budgeting system that enables budgeting to occur even when you are not in close proximity to a computer. Budgeting, invoicing, and project management are all facilitated by this program, making it a fantastic alternative for creative teams. BillQuick is a software tool that is entirely dedicated to employee efficiency. It has an employee workload forecaster function that helps you have a better understanding of your staff and budget accordingly. Additionally, there are tools such as budgeting for projects. Creating invoices and charging clients to send invoices through email as PDF attachments. Keep track of personal leave. Keep track of overtime. Keep track of compensatory time and vacations, and send invoices through email as PDFs.

2.4. METHODS OF BUDGETING

Budgets imposed from the top-down are referred to as imposed budgets. It is determined independently of the final budget holder. It saves time since decisions are made by a small group of top executives. Junior managers may lack the necessary abilities to engage fully in the budget decision-making process. Senior managers have a clearer picture of the organization's strategic goals and available resources. Senior managers are more aligned with the organization's strategic goals and take a long-term perspective. Junior managers should include wiggle room into the budget to make it more manageable (Althnian, 2021).

A bottom-up budget is known as a participative budget. All budget holders can contribute. Bottom-up budgeting increases managers' likelihood of achieving budgeted goals and boosts their morale. Lower-level managers are more in touch with the company and have a better understanding of its specific issues, challenges, and opportunities.

Incremental budgeting is appropriate if the primary cost drivers do not change from year to year. Its advantages include; simplicity, accuracy and quickness. It is the most commonly used budgeting tool since it is simple and straightforward. However, some of its disadvantages includes; it compounds historic errors, it does not consider any necessary future changes and it assumes the base is accurate.

Zero-based budgeting, one of the most commonly used budgeting approaches, suggests that all agency budgets are zero and must be developed from the bottom up. All expenses must be justified by the manager. Budgeting on a zero basis strives to remove any costs that are not regarded as necessary for the productive (profitable) operations of the firm. The zero-based method is effective when quick cost reductions are required, such as when a firm is undergoing financial restructuring or is suffering from a severe economic or commercial crisis that forces it to dramatically reduce its budget. It is advisable to budget for non-essential operating expenditures rather than for discretionary charges. However, many businesses still use it only a few times a year because of the time-consuming nature of the process (Ugochukwu *et al*., 2020).

CHAPTER THREE
SYSTEM ANALYSIS AND DESIGN

3.1 INTRODUCTION

System analysis is simply the study of a business problem domain to provide solutions, recommend solutions and specify business requirement. The intent of system analysis is to provide the project team with a comprehensive understanding of the problem and needs that triggered the project. System design involves defining the entire system architecture based on predefined requirements from the user.

This chapter provides a comprehensive overview of the approach utilized to create the Budget and Expense Management System. Using the notations of the Unified Modelling Language as the basis for modeling, the requirements and the many tools used are laid out in tables and shown in detailed graphs.

3.2 The Proposed System

The proposed system is a web-based budget and expense management system that would generate budget plans for individuals based on their income and display their savings over time. Users will be able to create monthly or weekly budgets and input expenses; the system will then generate a budget plan for them, and users will have the ability to adjust their budget. In addition, a graphical depiction will be given so that the user can judge whether or not they have properly followed the plan and saved money. Email notifications will be issued to users requesting new expenditure for the chosen time period. The consumer has the option to enable or disable these notifications, which will assist them in monitoring their budget.

3.2.1 Requirement Analysis

Requirement Analysis is a phase of the System Development Life Cycle (SDLC) that identifies the services that users demand from systems and the constraints that govern their operation and development. It ascertains what clients want from a new system and how the system should interact with them. A specification document is generated, which serves as an official statement of the features that will be added by the system developers. During this phase, mistakes and omissions should be minimized as they might anger consumers and result in costly changes. Typically, a system's requirements are categorized into functional and non-functional requirements.

3.2.2 Functional Requirements of the System

The functional requirements describe what the system should be able to do, how it should respond to inputs, and how it should behave in certain situations. The budget and expense management system's functional requirements include the following:

1. Users must be able to signup, login and log out of the system.
2. Users must be able to input and update income.
3. Users must be able to view and change budget plan.
4. Users must be able to receive reminders on their budget.
5. System must be able to generate budget for the user
6. Users can update their profile at will

3.2.3 Non-functional Requirements of the System

These are the system's overall features or attributes. They are the constraints on the services provided by the system.

1. The interface should scale in proportion to the size of the device's screen.
2. The website should be viewed on any device with internet connectivity.
3. The interface, including menus and settings, should be simple to understand and use for everyone.
4. The data of users should be protected against all types of security attacks.
5. The system should be capable of supporting further functionalities.
6. The System should produce accurate results at all times.

3.3 SYSTEM DESIGN

This includes the point-by-point graphical representation of the information and the whole framework's work process, which is shown in a diagram. It demonstrates communication between elements and entities that are within and outside the framework. It reveals the overall model, which includes the framework's details and structure in great detail. The plan is deconstructed into a cohesive and precise specification. The purpose is to create a theoretical representation of how the application operates and how the few framework modules react to various client behaviors or framework events.

3.3.1 Behavioral Modeling Diagram

The behavioral modeling diagram depicts the required behavior of the modeled system. The diagrams are used to convey information about the functionality of software systems. They include the following:

i. Sequence diagram
ii. Use case diagram
iii. Activity diagram

3.3.2 Structural modeling diagram

They only call attention to the aspects that must exist in the represented system, most notably the system's structure. Structure diagrams illustrate the static relationships between the elements in your system. i.e., the relationship between two objects. It displays information about the system's components—classes, objects, packages or modules, physical nodes, components, and interfaces. For instance, static aspects of a house include the presence and placement of walls, doors, windows, pipes, wires, and vents. Because structure diagrams show how a system is put together, they are often used to show how software is put together in systems. For instance, the component diagram illustrates the division of a software system into components and their interdependence. They are widely used to record the software architecture of software systems because they visually show the system's structure.

i. Class diagram
ii. Deployment diagram
iii. Object diagram

3.3.3 Use Case Diagram

Use case diagrams are used to visualize the system's activities. It often illustrates the system's users, referred to as actors, as well as the ways in which the client interacts with the system. Typically, it outlines the system's intended usage and illustrates the sequence of events that can be accomplished. It shows the relationship between use cases, the user, and the systems, but not the sequence in which the user and the system interact. In a use case diagram, the system is represented by a rectangle that defines its scope, illustrating what occurs within and outside the system; actors are often represented outside the system boundaries.

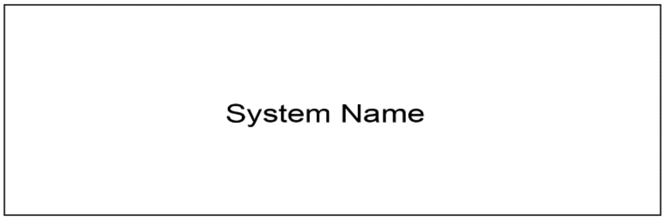


Figure 3.1: A use case system

The use cases are represented by an oval shape labeled with verbs describing the system's functionalities. They are contained within the rectangle to represent an action that the actor can perform in order to fulfill a task in the system. The following are types of use cases:

i. Base use case: A use case initiated by the actor.
ii. Extend use case: A use case that is triggered while the actor is performing the base use case. This use case is only triggered when a set of conditions are met. The use case for the quiz generator is an extended use case.
iii. Include use case: A use case that is invoked whenever an actor invokes a base use case. The use case for push notifications is an included use case.
iv. Generalized use case: It is an abstract representation of a collection of use cases that share common behaviors.
v. Specialized use case: It inherits the general use case's behaviors and adds specific ones. A typical use case scenario is represented in Figure 3.2.

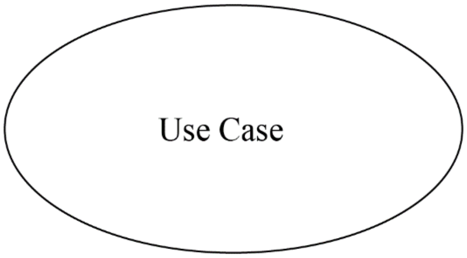


Figure 3.2: Diagram of a use case

Use cases in the Budget and Expense Management system:

i. Signup/Login: The system user will be required to login or register on the platform in order to receive access to the platform's services.

ii. Input income: After logging into the platform, the user will be requested to input his/her income for the month, which will be utilized to produce a budget for the user. The income may be updated over time.

iii. Generate Budget: After inputting income, the system will then generate a budget based on the budgeting rules (50 percent on needs, 30 percent on wants, and 20 percent for saving). A percentage will also be computed for each chosen need and want.

iv. Review Budget Plan: The customers will be able to check their budget plan and see whether it is to their satisfaction. They may also edit the plan and delete it if they wish to establish a fresh plan by themselves.

v. Notification: The user will be allowed to set the interval of the push notification as it pleases the user.

The system users are individuals, an organization, another system, or an external device. They are depicted by stick figures. This object is outside the system and is an external object. In a use case diagram, major actors initiate the usage of the system and are positioned on the left side of the system. Secondary actors respond to the main actors' actions and are placed on the right side of the system. In the image below, various actors are illustrated together with their respective actions.

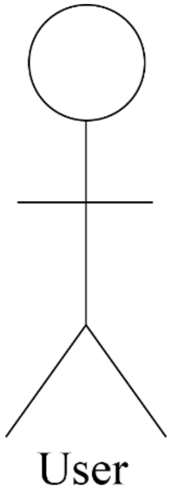


Figure 3.3: Diagram of an actor in a use case

The relationships between the various components of a use case diagram represent interactions. Relationships come in a variety of forms:

i. An association connection exists when a simple line is drawn between an actor and a usage instance. The user is associated with login, registration, income input, budget plan review, budget plan deletion, expense input, notification settings, and logout. The association relationship of a use case diagram is represented in Figure 3.4.

Figure 3.4: Diagram of association relationship

ii. Relationships between the base application and the extension application Depending on whether a user can login, input income, and add expenses, the system will operate the platform's budget plan generation; if the user can input income and add expenses, the system will generate a budget plan for the user; if the user cannot input income and add expenses, the system will generate no budget plan for the user. Thus, the budget generator use case expands the add expenses use case. The connection is represented in Figure 3.5 by a dashed line pointing to the base case and labeled '<<extend>>'.

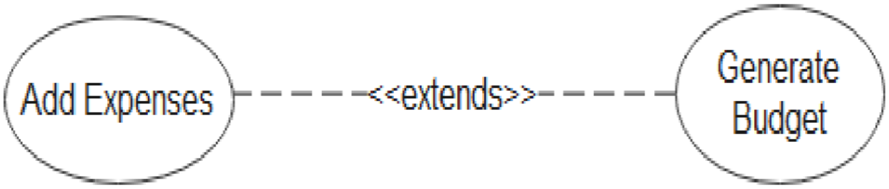


Figure 3.5: Diagram of extends relationship in a use case

iii. Include connections exist between a base use case and an include use case. Because the push notification use case is invoked each time the actor executes the register use case, the connection between them is one of inclusion. The relationship is depicted in Figure 3.6 by a dashed line pointing to the included use case and labeled with '<<include>>'.

iv. Relationships of generalization exist between a generic and a particular use case.

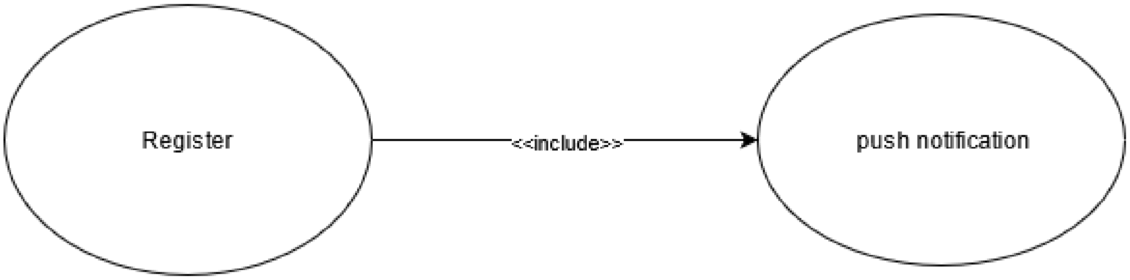


Figure 3.6: Diagram of include relationship in use case

Figure 3.7 is based on the principles we talked about earlier. It shows the proposed system's use case diagrams.

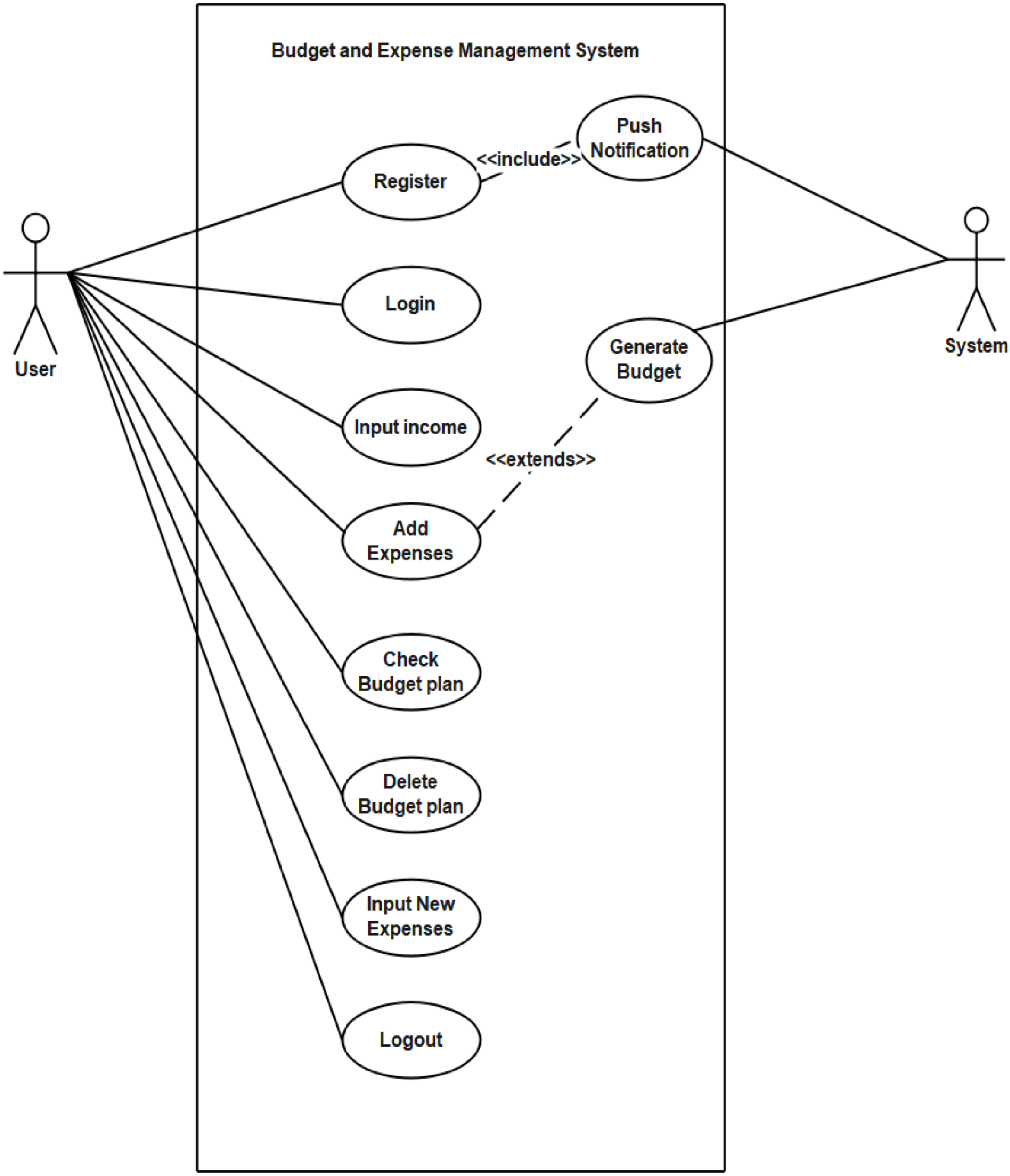


Figure 3.7: Use Case Diagram for Budget and Expense Management System

3.3.4 Activity Diagram

The activity diagram is a graphical representation of the workflow. Because it focuses on the execution and flow of a system's operations rather than its implementation, it's called an object-oriented flowchart. For behavioral learning, activity diagrams are made up of behaviors and activities that connect to technology. The process of User (Author) and System operations is depicted in the diagram below.

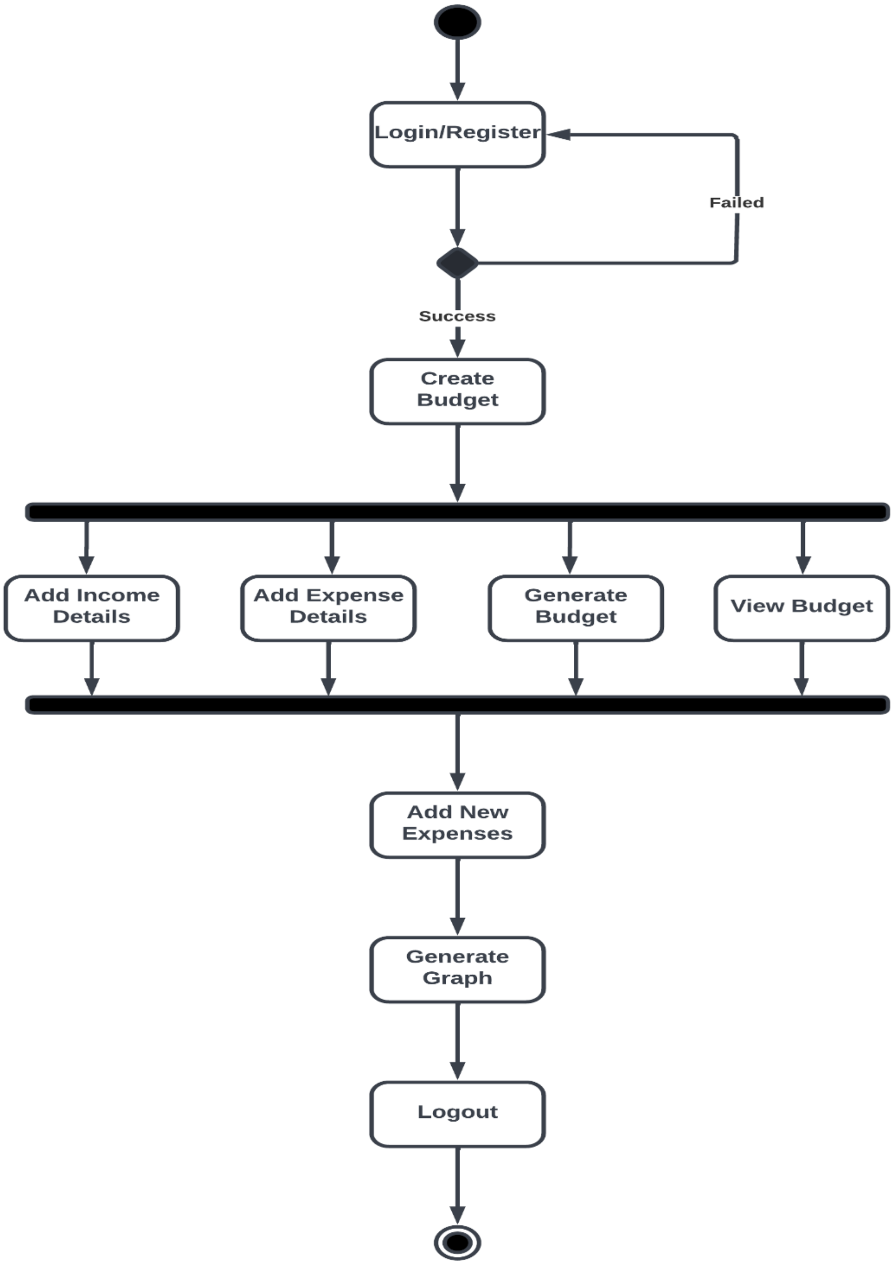


Figure 3.8: Activity Diagram for Budget and Expense Management System.

3.3.5 Sequence Diagram

Sequence diagrams are visual representations of how activities are completed. They depict the relationship between items in terms of collaboration. They are time-centered and use the vertical hub of the graph to discuss the time that messages are sent to highlight the desire for interaction.

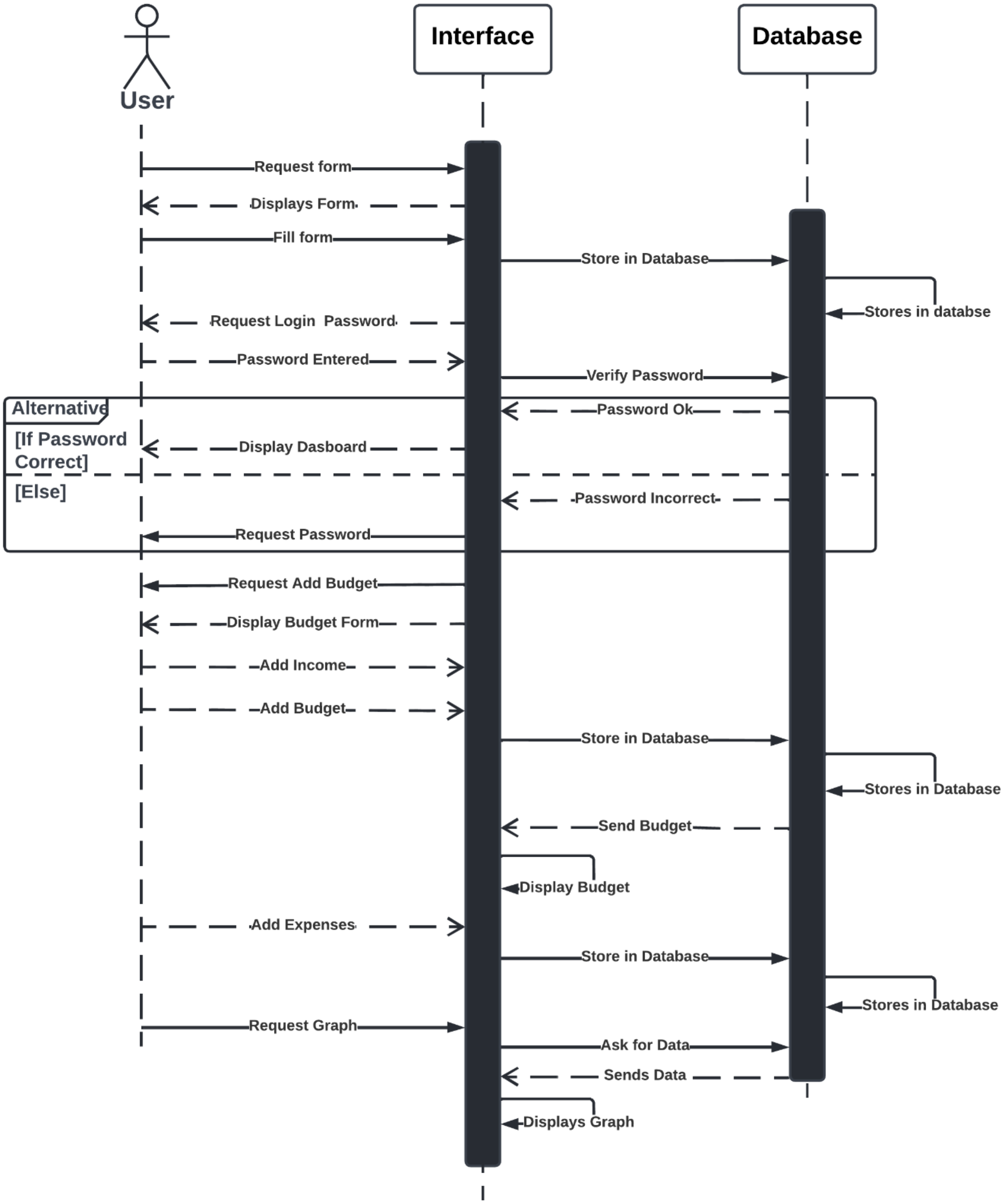
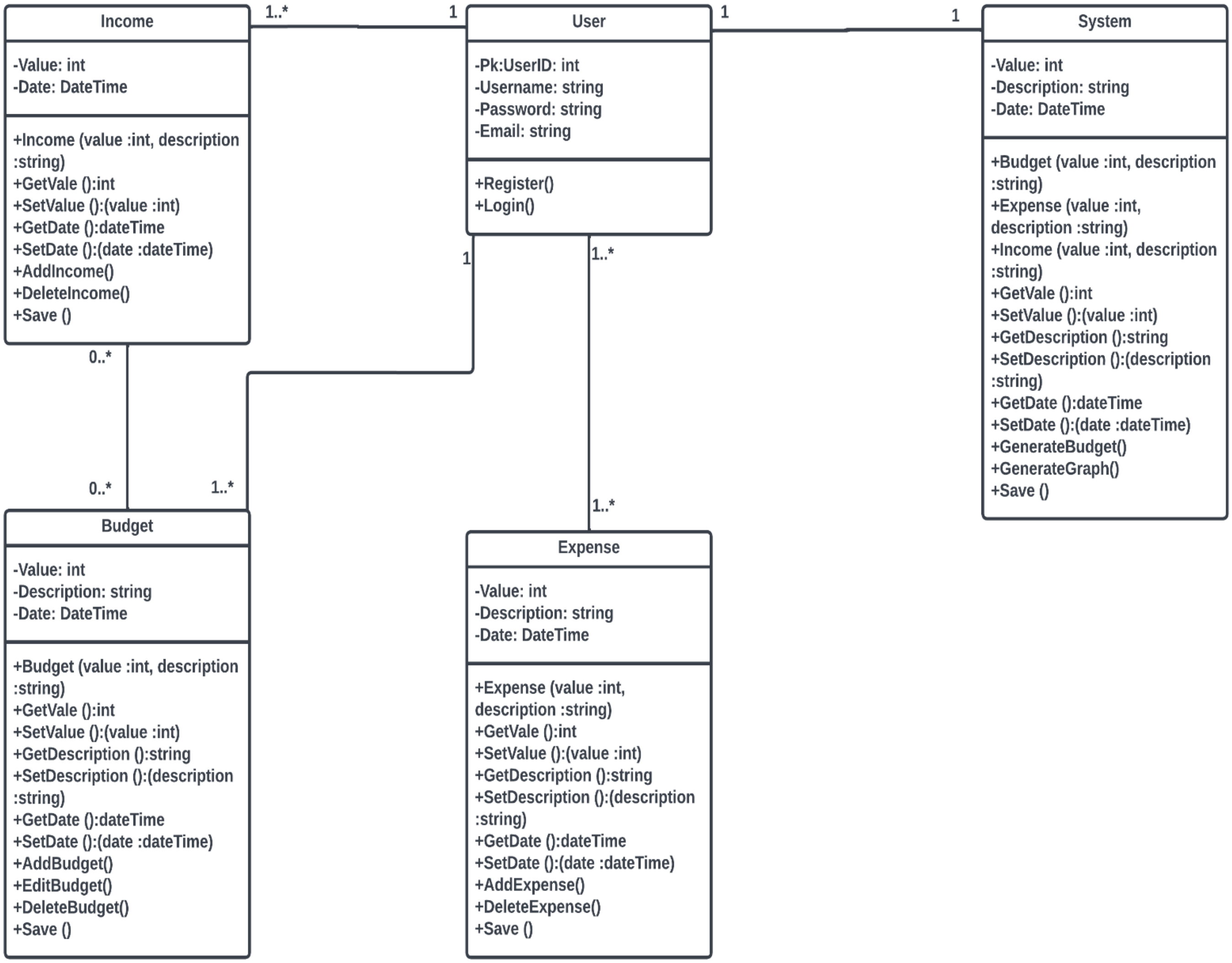


Figure 3.9: Sequence Diagram for Budget and Expense Management System

3.3.6 Class Diagram

A class diagram is a type of structural unified modeling language diagram used to describe the system's structure concerning its properties, groups, functions, and connections that exist within the class objects. It represents a system in an object-oriented view, which makes it simpler to relate to the real world. The diagram below displays the class diagram representation of the web app for the budget and expense management system.



Income
-Value: int
-Date: DateTime
+Income (value :int, description :string)
+GetVale ():int
+SetValue ():(value :int)
+GetDate ():dateTime
+SetDate ():(date :dateTime)
+AddIncome()
+DeleteIncome()
+Save ()
1..*
1
User
-Pk:UserID: int
-Username: string
-Password: string
-Email: string
+Register()
+Login()
1
1
System
-Value: int
-Description: string
-Date: DateTime
+Budget (value :int, description :string)
+Expense (value :int, description :string)
+Income (value :int, description :string)
+GetVale ():int
+SetValue ():(value :int)
+GetDescription ():string
+SetDescription ():(description :string)
+GetDate ():dateTime
+SetDate ():(date :dateTime)
+GenerateBudget()
+GenerateGraph()
+Save ()
1
1..*
0..*
0..*
1..*
1..*
Budget
-Value: int
-Description: string
-Date: DateTime
+Budget (value :int, description :string)
+GetVale ():int
+SetValue ():(value :int)
+GetDescription ():string
+SetDescription ():(description :string)
+GetDate ():dateTime
+SetDate ():(date :dateTime)
+AddBudget()
+EditBudget()
+DeleteBudget()
+Save ()
Expense
-Value: int
-Description: string
-Date: DateTime
+Expense (value :int, description :string)
+GetVale ():int
+SetValue ():(value :int)
+GetDescription ():string
+SetDescription ():(description :string)
+GetDate ():dateTime
+SetDate ():(date :dateTime)
+AddExpense()
+DeleteExpense()
+Save ()

Figure 3.10: Class Diagram for Budget and Expense Management System

3.4 SYSTEM ARCHITECTURE

Any system's architecture refers to the fundamental design considerations that support the system. It provides a basic understanding of how the various system components operate independently and interrelate to produce the system's end outcome. The suggested system is a web application, and its architecture describes the relationship between web applications, middleware, and database systems. With a normal web application, two subprograms operate concurrently.

This refers to the code that resides on the server and processes HTTP requests. Among the languages used for developing server-side code are Java, JavaScript, Python, PHP, and Ruby, among others. The development team is responsible for determining how the server code will interact with the program code. In addition to generating the page requested by the user, worker-side programming is responsible for storing various types of data, including user inputs and profiles. Typically, end users never see this.

Client-side code refers to the code on the browser that responds to user input. Combining HTML, JavaScript, and CSS is used to develop client-side programming. This, unlike server-side code, may be viewed and edited by the end user. It provides a reaction to the user's inputs. Client-side programs cannot directly read server-side files; it can only interact over HTTP.

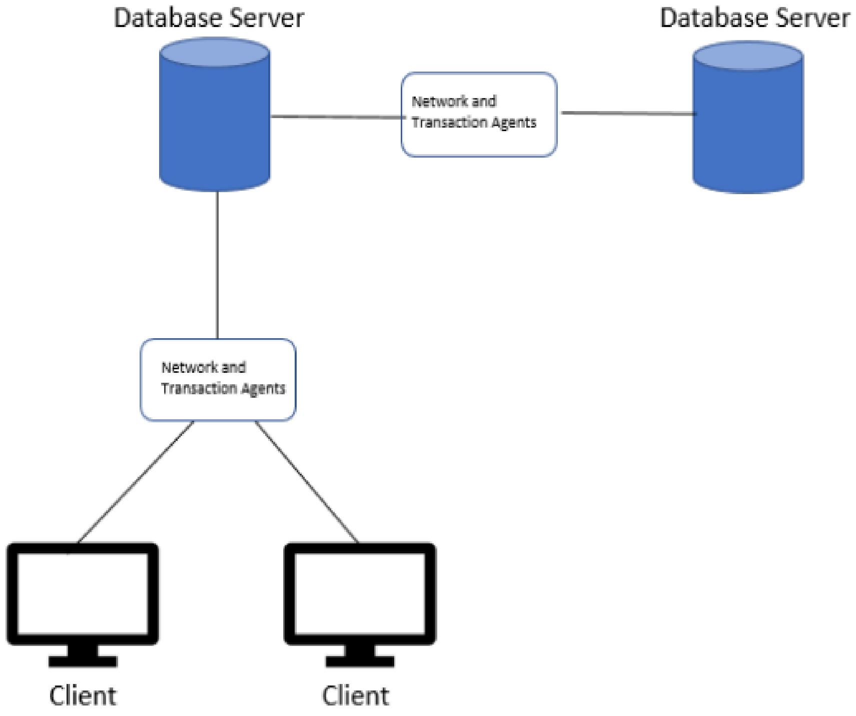

Database Server
Database Server
Network and Transaction Agents
Network and Transaction Agents
Client
Client

Figure 3.11: Client – Server Architecture

3.4.1 Web Application Architecture

A web application is an application that users may access using a web browser or a client-specific application. The application sends HTTP requests for specific URLs that refer to assets to a web worker. The employee transmits and receives HTML pages that the software can show to the consumer. The worker-side logic is the central component of every web application. There might be several separate levels inside the application. Figure 3.12 depicts a typical web application architecture, with shared components arranged by discrete areas of concern.

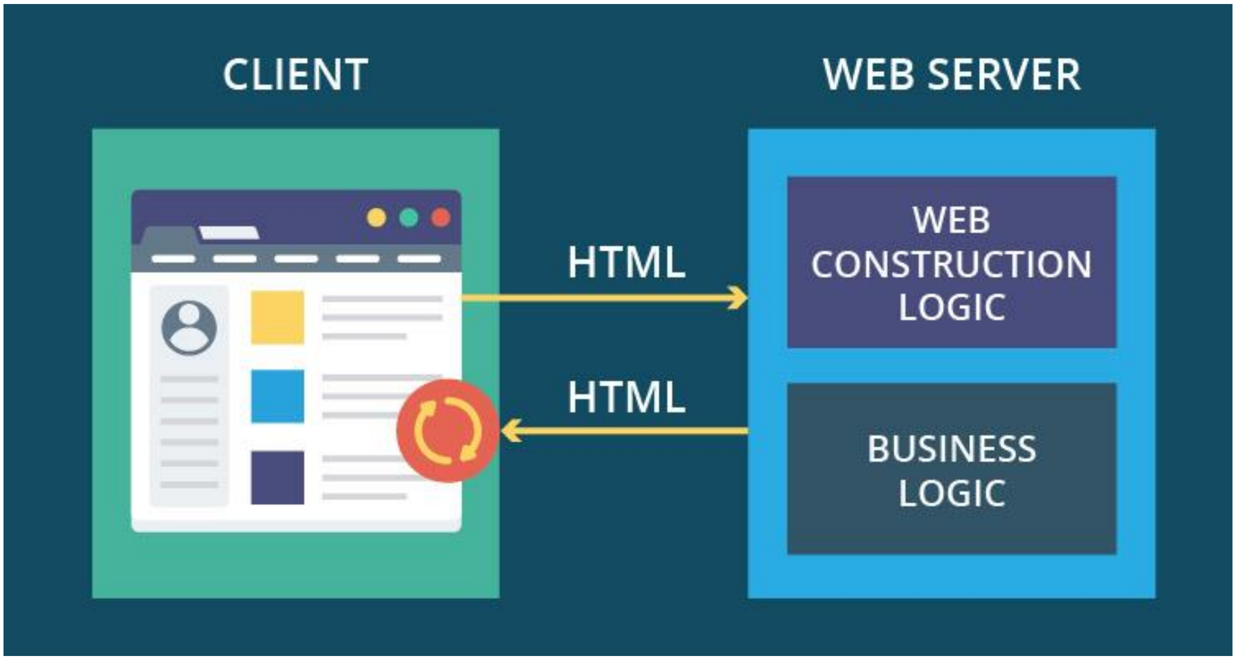


Figure 3.12: Web-Based System Architecture

- **Types of Web-based System Architecture**
  1. Single-Page Applications (SPAs) - Rather than loading new pages from the server, single page web applications contemplate a unique communication via mechanism for providing updated material to the present page.
  2. Micro services - These are lightweight services that perform a single function. The Micro Services Architecture system provides various options that enable engineers to improve efficiency and accelerate the deployment lifecycle.
  3. Server-less Architectures - In this type of web application architecture, an application developer consults with an external cloud framework administrations provider for server and infrastructure management.

3.4.2 Database Schema

A database is a collection of interrelated data, and the software that manages and regulates access to the database is known as a Database Management System (DBMS). This system is also responsible for storing, retrieving, and altering data. Data will be stored locally using MySQL database atlas(Marouf, 2019). This must be maintained small enough to prevent taking an excessive amount of memory, which might negatively affect performance and the availability of resources for other applications, but large enough to store all required data.

Table 3.41: Registration table

COLUMN	DATATYPE	DESCRIPTION
Id	INT	Identity
username	VARCHAR	User’s Unique Name
Email	VARCHAR	User’s Email Address
Phone number	VARCHAR	User’s Phone Number
Password	VARCHAR	User’s Password

Table 3.42: Budget Table

COLUMN	DATATYPE	KEY
Budgetid	INT	Primary Key
Budget Name	VARCHAR	
Income	VARCHAR	
Start Date.	DATETIME	
End Date	DATETIME	

Table 3.43: Admin Table

COLUMN	DATATYPE	KEY
Userid	INT	Primary Key
Username	VARCHAR	
Phone number	VARCHAR	
Email	VARCHAR	
Date registered	DATETIME	

CHAPTER FOUR
SYSTEM IMPLEMENTATION

4.1 INTRODUCTION

System implementation includes all processes involved in the system's efficient operation in its environment, such as operating, testing, systems integration, etc. This chapter addresses and demonstrates the implementation procedures, tools, and techniques for the proposed system. It contains all the software components and programming languages being used to complete the proposed system.

4.2 SYSTEM REQUIREMENTS

The system requirements must be met for the proposed system to function. Before the system is evaluated, it is essential to have a few measures in place to guarantee its smooth operation. Web applications typically have trouble with browsers, particularly obsolete ones. However, the hardware and software specifications outlined in this section should be satisfied for a more seamless platform experience.

4.2.1 HARDWARE REQUIREMENT

Table 4.21: Hardware Requirements Table

Component	Minimum Requirement	Recommended
Processor	Intel(R) Core (TM) i3-5005k CPU @ 2.0GHz 2.60 GHz	Intel(R) Core (TM) i5-8265U CPU @ 1.60GHz 1.80 GHz
RAM	4.00GB	8.00GB or higher
System Type	64-bit Operating system, x64 based processor	64-bit Operating system, x86 based processor
Disk Space	20GB	500GB or higher

4.2.2 SOFTWARE REQUIREMENTS

The system's software requirements can be divided into development software requirements and client software requirements. Development Software Requirements are the tools required to properly create and deploy the system for client interaction. The Client Software Requirements consist of the tools utilized by the client to execute the program. They are the output of the development tools.

Table 4.2: Software Requirements Table.

Component	**Requirement**
Operating System	Windows 10, 11
Database Management System	MySQL
Programming Languages	PHP, HTML, CSS, JavaScript.
Web Server	Xampp Apache Web Server
Integrated Development environment	Visual Studio Code, Notepad++
Internet Browser	Google Chrome, Microsoft Edge

4.3 IMPLEMENTATION TOOLS

The different parts of the web-based budget and expense management system were put together with the following tools:

4.3.1 FRONTEND

The suggested system's frontend is developed with JavaScript, HTML, CSS, and Bootstrap.

i. JavaScript: This is a declarative, efficient, and flexible programming language for constructing user interfaces; it enables the creation of large user interfaces from small code fragments known as components. Therefore, it is an excellent tool for front-end development that is very dynamic and responsive. There are many APIs in JavaScript libraries that make it easier for developers to create user interfaces. JavaScript is excellent for client-side rendering, which makes web application interactions more fluid; the entire page does not refresh for each interaction, and client-side rendering also offers rich, animated micro interactions. But client-side rendering has some problems. It's not good for Search Engine Optimization (SEO), and the first time you visit a website, it might take a while for a page to load because JavaScript must load before the browser runs the code and displays the information.

ii. HTML: Hypertext Markup Language (HTML) is the main language used to display Web pages on the Internet. Web pages are made up of HTML, which is used by a Web browser to show text, images, and other information. HTML represents the Web's architecture and content. HTML5 is the most recent version of HTML, including more characteristics, features, and tags.

iii. CSS: Denoting Cascading Style Sheet (CSS), it is used to organize the layout of websites by deploying falling templates. In addition, they are used to define font styles, table sizes, and other elements of Web pages that were previously defined in the HTML of the page. The most recent version of the CSS standard is CSS3. CSS3 is the third level of development in the CSS standard, so you cannot just refer to it as a reference to CSS's newest feature. CSS3 has more features than CSS, such as new sections, drop shadows, rounded corners, multiple backgrounds, animations, and transparency.

iv. jQuery: jQuery is a lightweight JavaScript framework that makes it easier to handle events, navigate HTML pages, interact with Ajax, and make animations, among other things. This can speed up the development of online web applications.

v. Bootstrap: Bootstrap is simply CSS at its foundation, but it has been constructed with a versatile pre-processor that provides far more power and flexibility than standard CSS. We get a lot of new features, like nested declarations, variables, mixings, operations, and color functions. However, because bootstrap is just pure CSS when it's made with Less, I get two very important benefits.

First, importing Bootstrap is as simple as inserting it into the code and proceeding. Second, once compiled, Bootstrap comprises only CSS, meaning it does not include any unnecessary pictures, Flash, or JavaScript. CSS is the last remaining option for your web development needs. Bootstrap typically functions by offering a unified solution for the most basic, everyday interface chores encountered by developers. It has become one of the most popular methods for launching new applications and websites on Twitter. It is incredibly comprehensive, secure, and adaptable to various distinct design requirements. Moreover, bootstrap's popularity is growing exponentially.

4.3.2 BACKEND

These are the server-side tools used to carry out the system's functionality.

i. Apache: Apache is the most generally used Web server on Linux computers. Web servers are used to serve Web pages requested by client computers. Clients commonly request and view Web sites using Web browser software such as Firefox, Opera, Chromium, or Internet Explorer. Apache's appeal in the Web hosting sector is partly because it is open source and free to use. Apache can host static websites as well as dynamic websites that employ server-side scripting languages such as PHP, Python, or Perl. Support for these and other languages is done using modules, or installation packages that are added to the regular Apache installation. Apache also works with extra modules that add more security options, tools for managing files, and other features.

ii. PHP: PHP (Hypertext Preprocessor) is a programming language used to create dynamic and interactive webpages. It was one of the first server-side programming languages that could be integrated into HTML, making it easy to add functionality to web sites without calling other files for data. One of the primary reasons PHP has become so popular is that it is reasonably easy to learn. Most people could produce a web page using a single PHP file in a reasonably short amount of time, even without considerable expertise or experience in web programming. PHP facilitates safe connections to practically any database type. This gives developers more freedom to choose the database that works best for the application they are building.

iii. MySQL: Oracle develops and supports the open-source SQL relational database management system MySQL. A database is merely an organized collection of data that is arranged for easy access. When it comes to storing data in a database, there are several methods available. MySQL utilizes a technique known as a relational database. In a relational database, data is divided into numerous distinct storage spaces, called tables, as opposed to being stored in a single massive storage unit. Using a "key," you may link the data from these two tables so that you can edit and mix the data from other tables as needed. MySQL is a relational database system that also employs the client-server architecture. The server component is where your data sits. To obtain this information, however, you must submit a request. Here is where the client comes in; the client sends a request to the database server for the required data.

4.4 SOFTWARE DEVELOPMENT METHODOLOGY

The application was designed using the agile development methodology, which is an iterative approach to software design and development in which concurrent requirements engineering and design tasks are undertaken. It consists of three principal phases, which are:

i. Specification: The process of obtaining system requirements and studying system users to determine their wants and the issues they want the system to solve for them.
ii. Development: The development phase builds systems and software components to address the concerns identified in the specification process.
iii. Validation: This is the process of verifying that the system and its components fulfill the user's requirements.

In this case, this way of developing software is used because it encourages iteration and constant improvement, as system iterations make it easier to see what needs to be done.

4.5. SYSTEM DEVELOPMENT PROCESS

System development is the process of defining, designing, testing, and implementing new software. Typically, there are six steps to the procedure, which are:

i. Requirement Analysis: The objective of requirement analysis is to identify the cause of the problem and, as a consequence, to rectify the system. In this phase, the system is broken down to figure out the problem. Project goals are looked at, what needs to be built is analyzed, and efforts are made to get users involved so that real results can be made.
ii. Design: As seen by the result, the new system is comprised of modules or subsystems. The criteria provided in the approved requirements document serve as the first input throughout the development process. The results of interviews, workshops, and prototypes will be a set of one or more design components for each need.
iii. Development and Testing: Here, the real code is written, and all modules are inspected for errors, bugs, and compatibility.
iv. Implementation: During the implementation phase, the software is turned into software configuration items that have been built, tested, and integrated. These items are now ready for software acceptance testing.

v. Documentation: Software documentation consists of text or graphics that accompany or are integrated into software. What the documentation implies about different people in various roles varies.

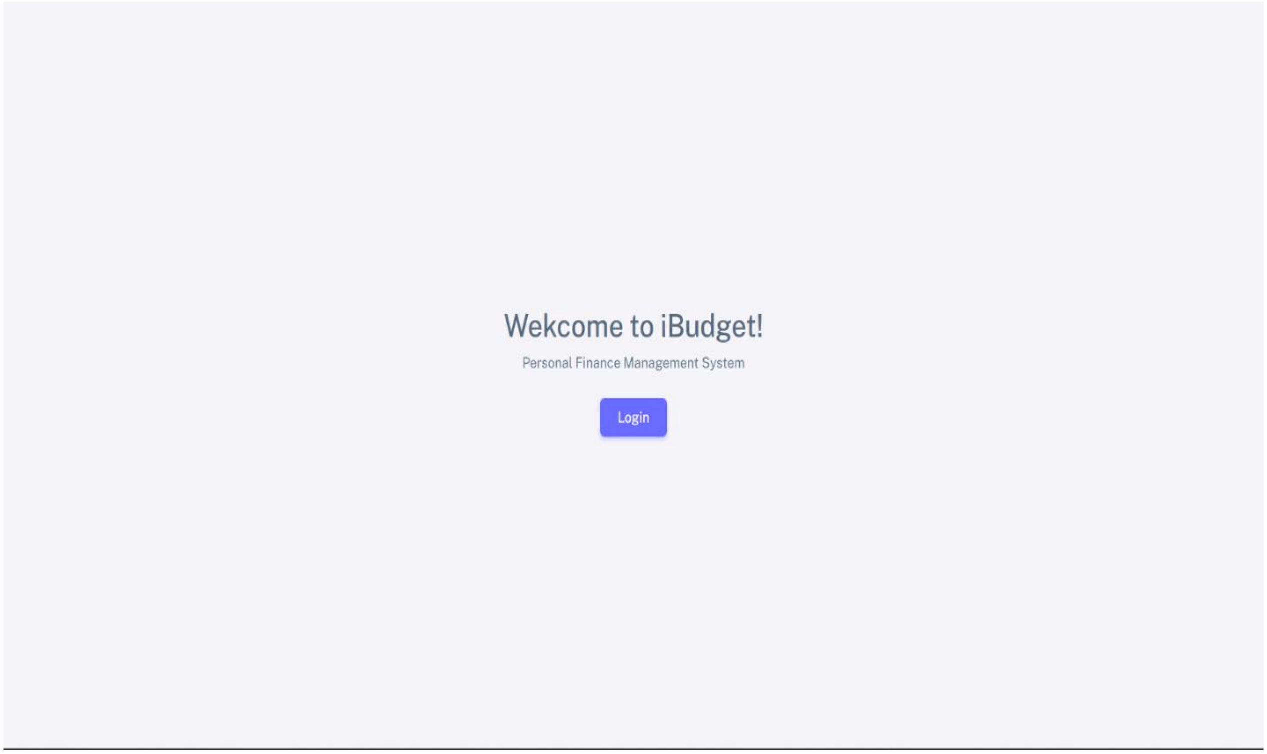
vi. Evaluation: Some businesses do not consider evaluation to be part of the SDLC, while others regard it as a continuation of the maintenance phase, sometimes known as post-implementation review. Here, both the system and the process are evaluated.

4.6 PROGRAMS AND MODULES

The system has been developed to be user-friendly, with straightforward navigation that supports user involvement. The following list describes the pages in the web application:

4.6.1 THE HOMEPAGE

When a user runs the web application, the index page is the first page that appears as shown in Figure 4.1 below. Because the web applications run on a local server, users can only access them after installing XAMPP Server, putting the application in the htdocs folder, and typing localhost/ibudget/php/index.php.



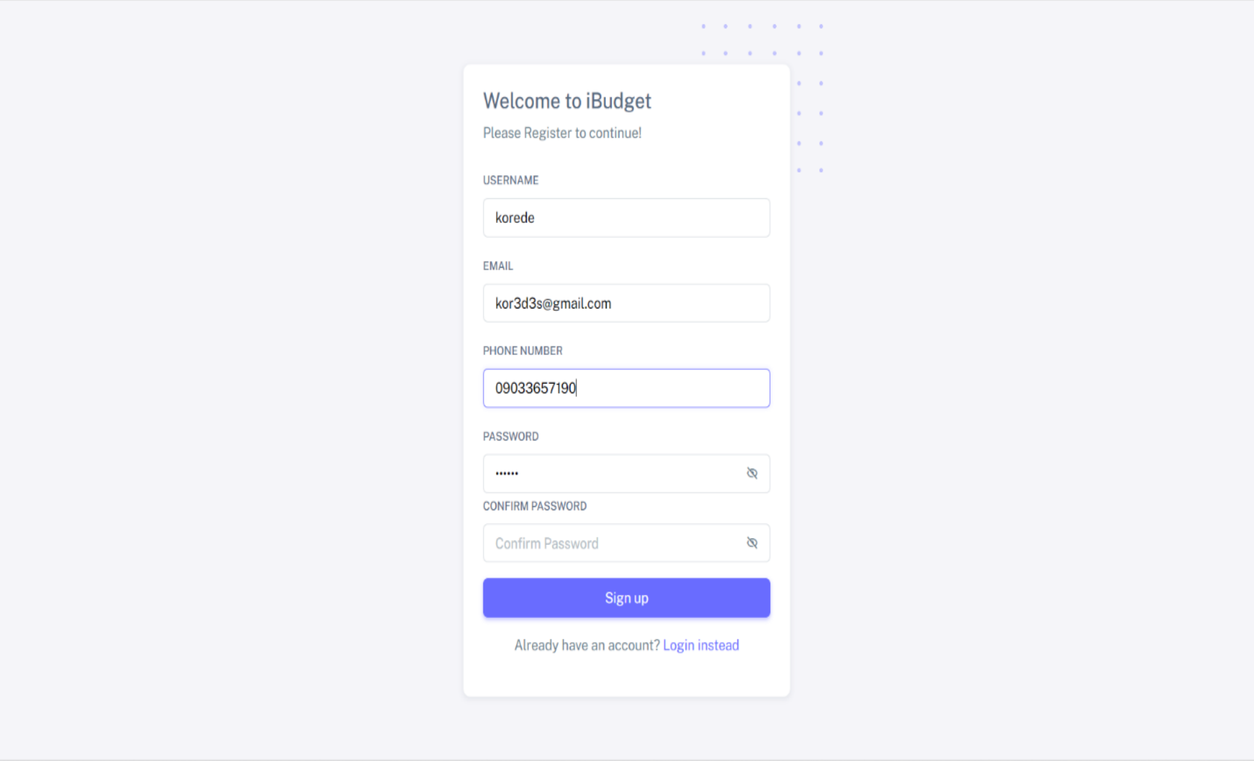
Wekcome to iBudget!

Personal Finance Management System

Figure 4.1: Landing Page

4.6.2 THE SIGN-UP AND LOGIN PAGE

The signup page, as seen in Figure 4.2 below, is where users register. The user's information is collected on the Registration Page, which includes a username, email address, phone number, and password. Before a user can register on the platform, all records must be filled in. After reregistering, users will be directed to the login page. The login page is only available after logging out of the system and only from the homepage. Administrator and user logins are the two types of logins available. Only one system administrator is allowed.



Welcome to iBudget

Please Register to continue!

USERNAME: korede

EMAIL: kor3d3s@gmail.com

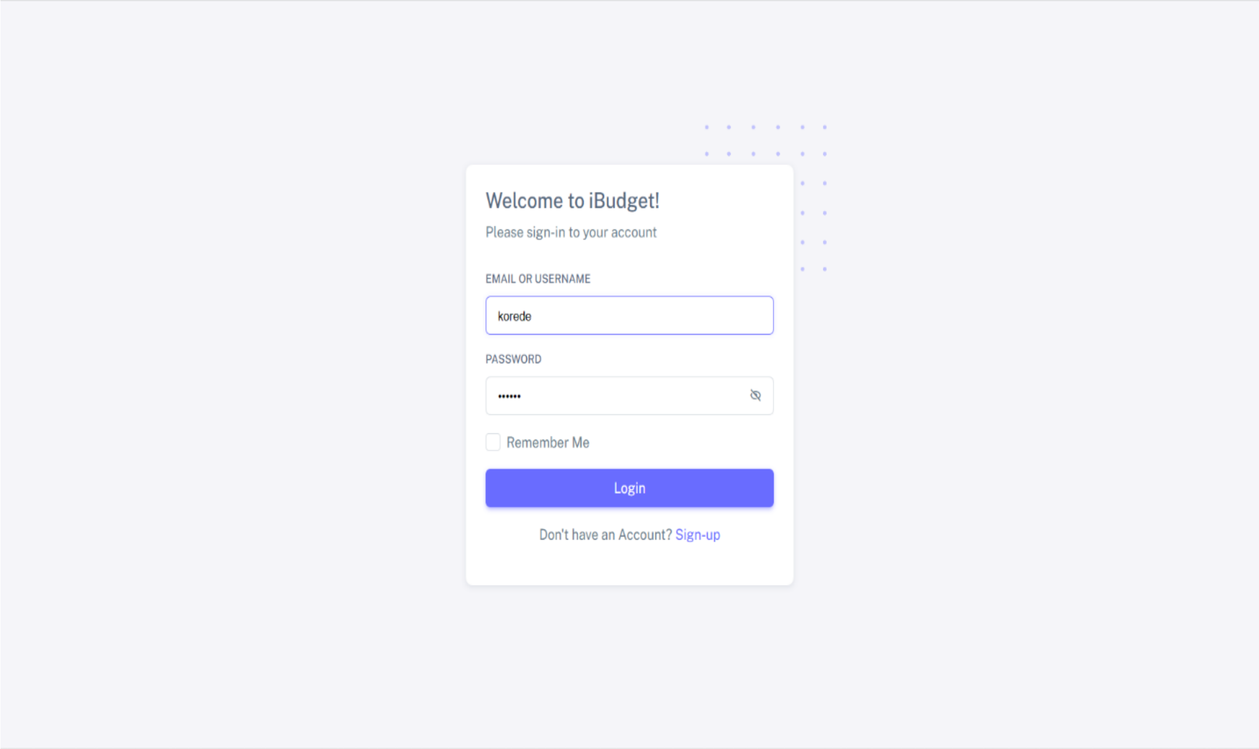
PHONE NUMBER: 09033657190

PASSWORD: ••••••

CONFIRM PASSWORD: Confirm Password

Figure 4.2: Signup Page

After registering on the system and saving all of their information in a MySQL database, the user is redirected back to the login page to properly log in as a registered user. For the time being, the user will not be able to receive any confirmation emails because the system is not hosted on the web. When a user enters their information on the login page, it is compared to the information stored in the database. If the information is correct, the user will be directed to their own personal dashboard. The user can log in using their email address or their username. This is depicted in Figure 4.3 below.



Welcome to iBudget!

Please sign-in to your account

EMAIL OR USERNAME

korede

PASSWORD

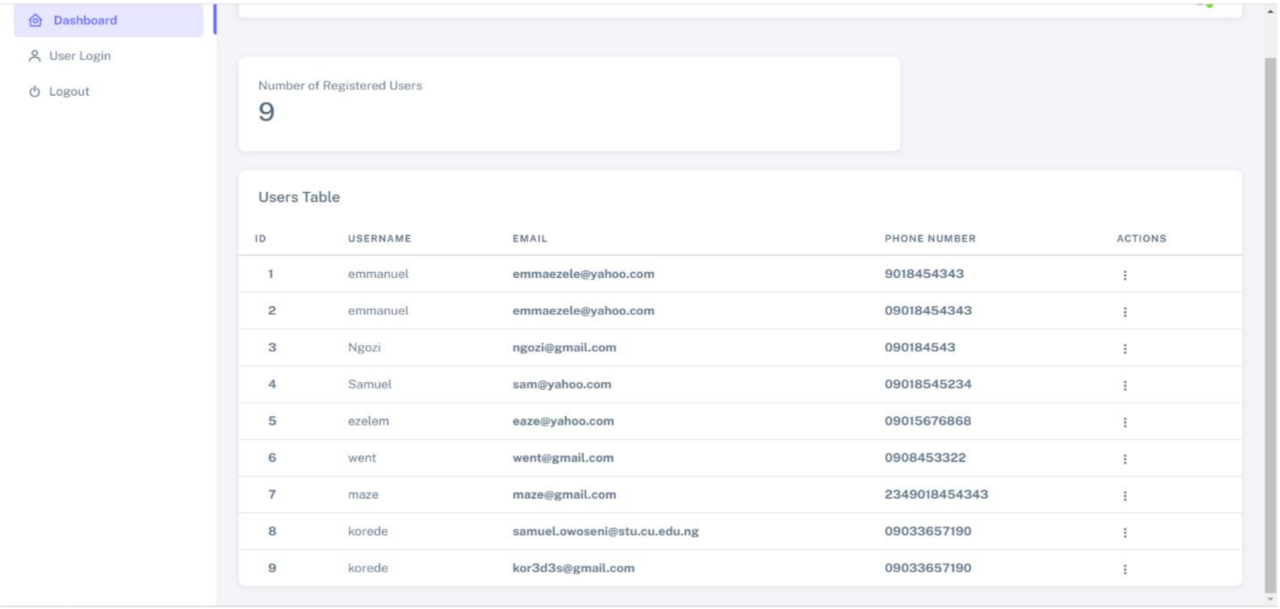
••••••

□ Remember Me

Figure 4.3: Login Page

4.6.3 ADMIN DASHBOARD

In the login portal seen in Figure 4.3 above, the administrator can log in to the control panel using a username and password provided to him or her. The login page consists of a form where the username and password can be entered. The administrator is then taken to a different dashboard than regular users, where he or she can see how many people have signed up and what information they gave when they did so, as shown in Figure 4.4 below.



Number of Registered Users
9

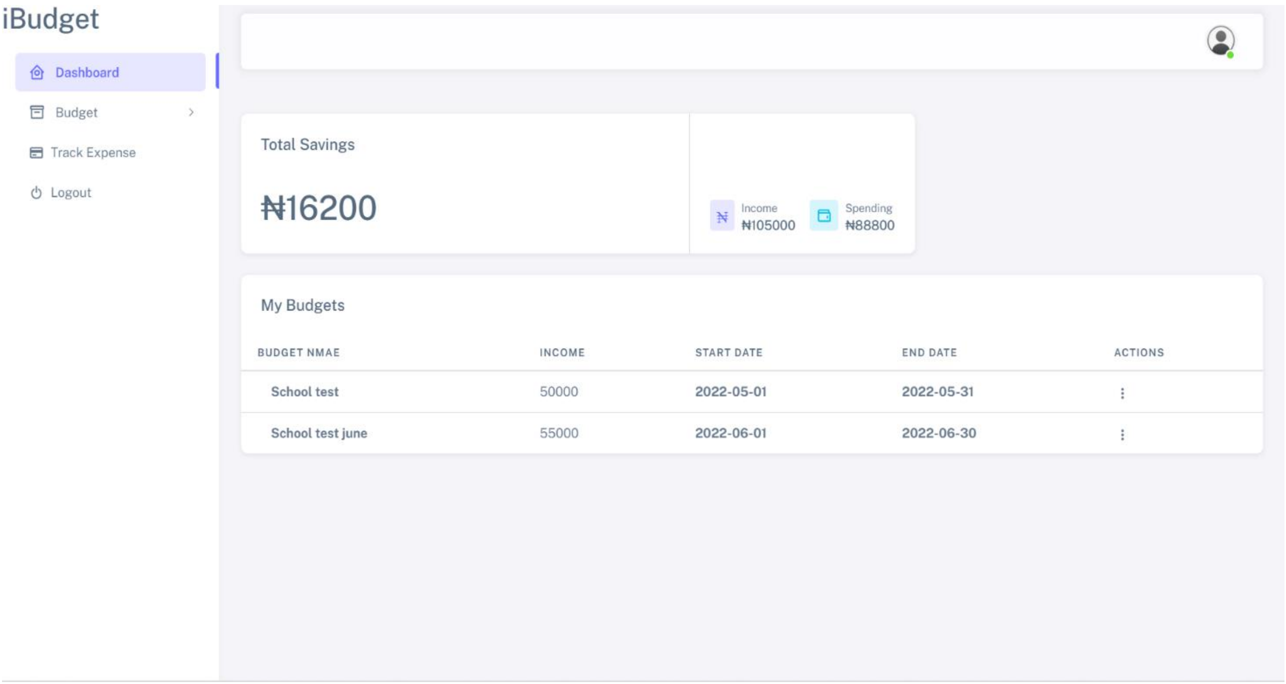
Users Table

ID	USERNAME	EMAIL	PHONE NUMBER	ACTIONS
1	emmanuel	emmaezele@yahoo.com	9018454343	⋮
2	emmanuel	emmaezele@yahoo.com	09018454343	⋮
3	Ngozi	ngozi@gmail.com	090184543	⋮
4	Samuel	sam@yahoo.com	09018545234	⋮
5	ezelem	eaze@yahoo.com	09015676868	⋮
6	went	went@gmail.com	0908453322	⋮
7	maze	maze@gmail.com	2349018454343	⋮
8	korede	samuel.owoseni@stu.cu.edu.ng	09033657190	⋮
9	korede	kor3d3s@gmail.com	09033657190	⋮

Figure 4.4: Administrator Dashboard

4.6.4 EMULATOR DASHBOARD

Users can access the dashboard by entering their username and password on the login page. There is a field for inputting the username and password on the login page. The user is then directed to the dashboard, which displays various data such as total savings, income, expenditures, and a table of generated budgets. This is seen in Figure 4.5 below.



Total Savings
₦16200

Income
₦105000

Spending
₦88800

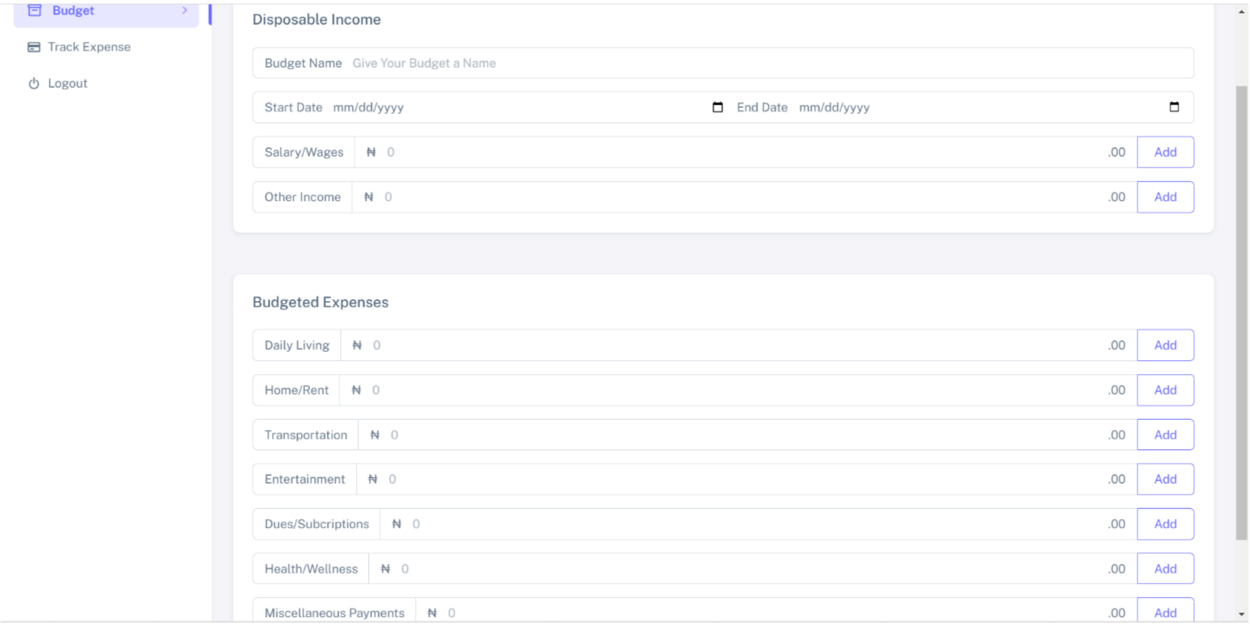
My Budgets

BUDGET NMAE	INCOME	START DATE	END DATE	ACTIONS
School test	50000	2022-05-01	2022-05-31	⋮
School test june	55000	2022-06-01	2022-06-30	⋮

Figure 4.5: User Dashboard

4.6.5. ADD BUDGET PAGE

Figure 4.6 below is the page where the users enter their income and selects a budget period. The user will enter the figures for the various expense categories before submitting. The data will be stored in a database and will be shown in a table on the dashboard. Users can build an unlimited number of budgets.



Disposable Income

Budget Name Give Your Budget a Name

Start Date mm/dd/yyyy End Date mm/dd/yyyy

Salary/Wages ₦ 0 .00 Add

Other Income ₦ 0 .00 Add

Budgeted Expenses

Daily Living ₦ 0 .00 Add

Home/Rent ₦ 0 .00 Add

Transportation ₦ 0 .00 Add

Entertainment ₦ 0 .00 Add

Dues/Subcriptions ₦ 0 .00 Add

Health/Wellness ₦ 0 .00 Add

Miscellaneous Payments ₦ 0 .00 Add

Figure 4.6: Add Budget Page

CHAPTHER FIVE

SUMMARY AND RECOMENDATION

5.1. SUMMARY

The budget and expense management system was created in phases. Using a top-down approach, the emphasis is placed on what comes first, followed by how, and then successive levels of detail. During the course of this project, several challenges that affected the efficiency of the previous manual technique were discovered. After the first phase, these challenges, information needs, and actions were recorded and utilized as the basis for system design. The majority of the design phase was devoted to defining the system's components in accordance with the functional and non-functional requirements. During this phase, well-established software engineering principles and practices were followed without modification. It is anticipated that the successful deployment of this web-based application will reduce a number of system development challenges. This project involved the development of a web application that keeps track of a user's weekly or monthly personal spending and budget. Sticky notes, spreadsheets, and ledgers, which can lead to inaccurate data and confusion when tracking and dividing costs, will no longer be utilized. This web application helps the user better manage their finances, and it also has a number of features that make it easy to use.

5.2. Recommendation

This project emphasizes the need for adopting digital technology into personal budgeting and expense tracking. However, due to project limitations and implementation issues, the system could be enhanced. The following suggestions, as well as extra work, are critical:

For future work on this project, the expense tracking system should be integrated with users' bank accounts for greater effectiveness and efficiency. Additionally, the system should be hosted on a cloud server rather than a local server to ensure that other functionalities such as email validation for each user and budget notifications are implemented. Users will have access to extra functionality that the web app does not currently offer if the system is implemented as a mobile web application rather than just a web app. Users should also be allowed to create their own expense categories. Finally, I would suggest incorporating artificial intelligence into the system to increase its performance and provide consumers with advice and suggestions on how to save money and cut expenses.

5.3. Conclusion

After developing this web application for this project, it will assist its users in managing their everyday expenses. It would direct and inform them of their daily spending. It will be helpful for those who are unhappy with their daily budget management, irritated by the number of expenses and desire to manage their money and those who wish to keep a record of their daily costs, which will be used to adjust their spending habits. In summary, this web application will aid its users in avoiding financial waste. The effectiveness of your financial planning will depend on your ability to use contemporary budgeting, planning, and forecasting approaches inside the finance department.

REFERENCES

Capucao, J. N., Palaoag, T., & Sierra, S. (2020). Web-based System for State Universities and Colleges Budget Office Management. *IOP Conference Series: Materials Science and Engineering, 803*(1), 009-012.

Focardi, S. M., & Fabozzi, F. J. (2020). Climate change and asset management. *The Journal of Portfolio Management*, *46*(3), 95-107.

Hinojosa, D. (2018). Essential Building Blocks for State School Finance Systems and Promising State Practices. School Finance Series. *Learning Policy Institute*. Northern Technical University.

Ma, S. (2016, January). A platform of comprehensive budget management system. In *2016 International Conference on Civil, Transportation and Environment*. *438*, 1284-1287.

Yang, X. F., & Wang, R. R. (2014). The design and implementation of teaching management system based on web. In *Applied Mechanics and Materials*. *543*, 3207-3212.

Althnian, A. (2021). Design of a rule-based personal finance management system based on financial well-being. *International Journal of Advanced Computer Science and Applications*, *12*(1), 182-187.

Jadhav, N. J., Chakor, R. V., Gunjal, T. M., & Pawar, D. D. (2022). Expense tracker. *04*, 1676–1679.

Lee, S. K. (2019). FinTech nudges: Overspending messages and personal finance management. *NYU Stern School of Business*, 555-570.

Lysiak, L., Kushnir, A., Tereshchenko, T., Kachula, S., & Datsenko, V. (2021). Assessment of Financial Sustainability of Local Budgets in the Budget Management System Using Kohonen Maps. *Universal Journal of Accounting and Finance*, *9*(6), 1558-1570.

Marouf, A. Al. (2019). Working as a Web Application Designer and Developer at SoftCareIT Fahad Ibne Fahian ID :*This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering*, *53,* 142-15-3679

Mumpuni, M., & Sukarno, S. (2014). Design and implementation money management web based application for personal and family proposed for CV. X. *Procedia-Social and Behavioral Sciences*, *115*, 444-459.

Singh, H. (2021). Budget and forecasting a comprehensive approach. *Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12*(6). 5095-5103.

Singh, U. P. (2021). Spending Tracker: A Smart Approach to Track Daily Expense. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(6), 5095-5103.

Ugochukwu, S. C., Emedosi, A., Emoh, F. C., & Chinagorom, J. C. (2020). *A Conceptual Model and Medium-Term Strategy Templates for Improving Housing Budget Performance in Anambra State, Nigeria. 21*(3), 51-70.

Withana, P. W. J. C., Uwanthika, G. A. I., & Wijethunge, I. A. (2021). A Novel Personalized Mobile Application for Systematically Monitoring Cash Transactions. *4th International Conference on Electronics, Communication and Aerospace Technology, 11*(5), 780-785.

Awaludin, M. (2018). Penerapan Sistem Piranti Lunak Personal Finance Berbasis Android untuk Peningkatkan Kualitas Ekonomi Individu. *JSI (Jurnal sistem Informasi) Universitas Suryadarma*, *3*(2), 107-114.

Roy, S., Siddiquee, S. M. T., Rahman, M. K., & Al Marouf, A. (2020). A Novel Authentication Method for Password Encryption. In *2020 4th International Conference on Electronics, Communication and Aerospace Technology, 11*(9), 780-788.