**Development of a Finance Management System**

**By**

**Wanjoku Sobulachi (211212137)**

**Abubakar Kamaludeen Barde (211212045)**

**Nathaniel Asemokhai (211212067)**

**Peace Ohuabunwa**

**Project Supervisor: Mr. Austin Olom Ogar**

**Department of Software Engineering**

**Faculty of Computing**

**Nile University of Nigeria, Abuja**

**October, 2024**

**Table of Contents**

[**CHAPTER ONE:** 4](#_Toc181606207)

[**INTRODUCTION** 4](#_Toc181606208)

[**1.1** **Background to the study** 4](#_Toc181606209)

[**1.2** **Statement of the Problem** 5](#_Toc181606210)

[**1.3** **Aim and objectives** 6](#_Toc181606211)

[**1.4** **Significance of the study** 6](#_Toc181606212)

[**CHAPTER TWO:** 8](#_Toc181606213)

[**LITERATURE REVIEW** 8](#_Toc181606214)

[**2.1** **Introduction** 8](#_Toc181606215)

[**2.2** **The Role of Automation in Financial Management** 8](#_Toc181606216)

[**2.3 Data Visualization and Financial Analysis** 9](#_Toc181606217)

[**2.4 Using Django for Financial Management Applications** 9](#_Toc181606218)

[**2.5 Agile Methodology in Developing Financial Software** 10](#_Toc181606219)

[**2.6 Responding to the Demands of Small and Medium Enterprises (SMEs).** 11](#_Toc181606220)

[**2.7 Gap in the Literature** 11](#_Toc181606221)

[**CHAPTER THREE:** 12](#_Toc181606222)

[**METHODOLOGY** 12](#_Toc181606223)

[**3.1** **Introduction** 12](#_Toc181606224)

[**3.2** **Agile methodology** 12](#_Toc181606225)

[**3.3 System Requirements** 14](#_Toc181606226)

[**3.4** **System Design** 15](#_Toc181606227)

[**3.5 Development Tools & Technologies** 15](#_Toc181606228)

[***3.5.1 Frontend:*** 15](#_Toc181606229)

[***3.5.2 Backend:*** 16](#_Toc181606230)

[***3.5.3 Data Visualization:*** 16](#_Toc181606231)

[***3.5.4 Data Management:*** 16](#_Toc181606232)

[***3.5.5 Security:*** 17](#_Toc181606233)

[**3.6** **Project timeline** 17](#_Toc181606234)

[**Conclusion** 17](#_Toc181606235)

[**References** 18](#_Toc181606236)

*Abstract*

*Financial management practices nowadays often rely on manual systems or spreadsheets which are inefficient and error prone. The goal of this project is to develop a financial management software designed to streamline financial processes like budgeting, financial forecasting etc. The aim of this project is to reduce these issues by making an automated solution that enhances data accuracy, automates tedious tasks, and enables real time financial analysis and data visualization. The methodology that will be used in the project is the Agile methodology, in order to ensure iterative enhancements and the integration of user feedback. A mix of modern software development technologies like Django for the backend and MySQL for database management, the software will also integrate data visualization libraries such as Matplotlib and Seaborn to show deeper insights into financial trends. When the software is completed, it is expected that it improves efficiency, minimizes errors, and assists users in making well informed financial decisions with accurate and real time data insights. This solution will be scalable and secure.*

# **CHAPTER ONE:**

# **INTRODUCTION**

## **Background to the study**

The landscape of the financial management industry is developing an increasing need for systems with increased accuracy and automation. This purpose of this document is to propose the development of a software tailored for users that will automate budgeting, financial forecasting, data visualization and also give tips on finances to the users. The software will also help the software will also help finance professionals with tasks like performing complex calculations, managing large datasets, and generate report in real time, thereby enhancing decision making efficiency and accuracy.

## **Statement of the Problem**

The traditional ways of managing finances using spreadsheets or manually storing the data are time consuming and prone to errors they also don’t have advanced features like real time data analysis. The goal of this project is to solve those inefficient challenges by developing a software that integrates key financial activities like budgeting and reporting.

Challenges like:

* **Time-Consuming Data Entry:** One of the most obvious disadvantages of manual accounting is the amount of time it requires. This basic process of recording financial transactions, whether in ledgers or spreadsheets, may be tedious and time-consuming.
* **Increased Risk of Human Error:** Paper-based accounting systems are prone to human mistake, which increases as the volume of financial data increases.
* **Difficulty in Tracking Changes:** Tracking changes or modifications in a paper-based accounting system can be time-consuming and labor-intensive.

## **Aim and objectives**

The aim of this project is to develop a finance management system that streamlines and increase the efficiency of financial processes like budgeting, financial forecasting etc.

The main objectives of this project are:

* To conduct a feasibility study on the financial management system
* To design a financial management system for budgeting, data visualization, and financial analysis
* To implement the designed financial management system and add all the features outlined
* To test the financial management system and fix any bugs found

## **Significance of the study**

This study has a lot of value to both academic research and practical applications in the fields of financial management and software engineering. The goal of this study is to reduce the gap between traditional financial processes and modern technology-oriented solutions. The significance of this study can be seen from several perspectives:

1. **For finance managers:** The software designed and developed in this project will provide finance managers with an efficient and accurate, automated tool for budgeting, forecasting, reporting and data visualization. This will substantially reduce the time spent on manual data entry and analysis, this allows for quicker and more accurate financial decision making. By integrating real time financial visualization, this software will empower financial managers to analyze trends, identify financial risks, and to make strategic decisions with confidence.
2. **For organizations:** Small to medium sized organizations normally lack access to advanced financial management tools due to the cost. This software will offer a cost effective and scalable solution, enhancing financial management and operational efficiency. By automating key financial activities, organizations will be able to reduce errors, improve data accuracy, and optimize resource allocation, thereby leading to improved profits and financial health.
3. **For the Software engineering community:** This project contributes to the development of domain specific software that displays the effective use of Python frameworks in building scalable and robust financial software. It demonstrated the application of Agile methodologies in the software development life cycle showing insights into how iterative development and continuous feedback can result in high quality software solution tailored to user needs.

# **CHAPTER TWO:**

# **LITERATURE REVIEW**

## **Introduction**

Over the last few decades, there have been notable improvements in the financial management systems. The older techniques, which mostly employed excel sheets and physical data entry practices, were common though they are inefficient and highly dependent on human input particularly when large quantities of information are concerned (Schnitt, 2023) In this literature review, we seek to analyze the development of financial management systems and software, the role of processes like automation and data visualization, and advantages of adopting suitable structures like Django in developing high-end financial applications.

## **The Role of Automation in Financial Management**

Automation in financial systems is vital for achieving a desired level of efficiency and precision. Holovaty and Kaplan-Moss (2009) explain that one of the merits of automating processes is the elimination of routine tasks making it possible for finance practitioners to engage in other high level critical thinking aspects rather than spending time transcribing data. Management of finance includes transaction processing, which involves enatic transaction classification, budget management, and real-time operational data that is crucial for effective reporting and projections (Nagle, 2023).

Moreover, there is a growing inclination towards integrating machine learning and artificial intelligence engines in financial management systems. The implementation of such techniques provides solutions such as predictive analytics, which helps organizations to analyze past expenditures for patterns that inform how much will be spent in certain periods and enabling forward looking actions (Brealey, Myers & Allen, 2019). Django REST Framework for html web development has helped in achieving automated processes that are scalable and secured enabling interaction of both the front-end applications and back-end applications without any friction (Vincent, 2020).

## **2.3 Data Visualization and Financial Analysis**

Data visualization has an essential component of today's financial management software. Dumais (2021) states that monetary visuals allow complex database analyses to be much easier to accept and read trends for the audience. ‘Fat’ libraries such as Matplotlib and Seaborn, which are associated with the Python programming language, have been used ever since being able to provide interactive and advance visualizations in financial data, enhancing the comprehension of the audiences (Gross, 2018).

Gross (2018) also asserts that data visualization assists finance managers in assessing the financial status of an organization almost instantly and performing benchmarking to analyze previous performance figures. When embedded in financial management applications, these visualizing tools aid effective managing of financials and help users to easily spot risks and opportunities for expansion.

## **2.4 Using Django for Financial Management Applications**

Django, a high-level web framework written in Python, is popular among financial solution developers because of its incredible security and scalability (Vincent, 2020). Lutz (2013) notes the importance of the MVT architecture in Django, as it promotes performance in working with several sets, which is critical in applications whose services entail a lot of data storage and retrieval, as in finance. Moreover, the coupling of the framework with Django REST, which enables easy construction of APIs, ensures that there is connectivity of the financial management system with other applications or platforms easily (Holovaty & Kaplan-Moss, 2009).

There’s more to it than that, as Holovaty and Kaplan-Moss (2009) researched and discussed and claimed a number of functionalities related to Django which include user management and role management for securing the software against unauthorized usage. This is reinforced in other works which discuss how Django-Allauth was able to help secure finance-based applications with sensitive user information from hackers.

## **2.5 Agile Methodology in Developing Financial Software**

Among the various software development methodologies, the Agile development approach is the one that has gained popularity especially for the development of Financial Management Information System (FMIS) due to the fact that users can always provide suggestions after every stage of development (Kochhar, 2011). With the stages of the development life cycle divided into shorter periods known as sprints, Agile makes sure that the programmers did not stay frozen as changes in financial requirements came, thus creating more agile software solutions that met the real-world problems (Vincent, 2020).

Moreover, according to recent research by Nagle (2023), the combination of Agile development with Django for instance, serves to widen the scope of financial software thanks to multiple foster iterations and updates. In addition, since Agile encourages client interaction and the conducting of tests during development, there is a very low chance of errors after delivery which also expresses the effectiveness of Agile in developing efficient financial management systems.

## **2.6 Responding to the Demands of Small and Medium Enterprises (SMEs).**

Financial management software is most cases expensive or too sophisticated for the SMEs which leads to very little uptake rates (Atrill & McLaney, 2020). In this context, Ohlhorst (2012) discusses rebalancing the development priorities to create low-cost, flexible solutions that do address the financial requirements of SMEs, while not omitting features such as budgeting, reporting, and data visualization. This project seeks to fill this niche by utilizing technologies and frameworks…, which are mainly open source, such as Django, to design reasonably priced and feature-rich financial management systems that are easily adaptable by mid-sized enterprises.

## **2.7 Gap in the Literature**

The conventional model of overseeing financial resources is prone to errors and inefficiencies owing to the overdependence on manual input and fixing of data in databases (Kochhar, 2011). In addition, Atrill & McLaney (2020) states that there is more than one constitutive definition available for a MANUAL accounting system: it is 'time-consuming' and 'outdated'. Moreover, there are no advanced analytical capabilities available since the information is historical in nature and therefore limits the ability to act on current financial status (Dumais, 2021).

Once again, Schnitt (2023) discusses the need for the automation of the financial data processes as a means of error reduction and enhancement of operational speed. There are also risks associated with financial information when there is no computerized system, which could impede decision making processes, therefore underscoring the importance of financial systems that promote data handling and analysis using almost real time data.

# **CHAPTER THREE:**

# **METHODOLOGY**

## **Introduction**

The methodology section describes the well organized process employed in the designing, constructing and launching the specific financial management software, as well as ensuring that each stage conforms to the relevant software engineering principles and the project aims. In order to mitigate the challenges that finance managers face such as the precision of data, analysis in real time, and security issues, the development process will be based on the Agile methodology that enhances works in progress, constant feedback and accommodates flexibility to changes.

## **Agile methodology**

The Agile methodology is the most effective method for the creation of a financial management system for the following reasons:

• Iterative development and regular feedback: Financial management systems are often complex and need a high caliber of accuracy and security. Agile methodology’s iterative development style allows for frequent testing and validation of features in order to catch errors and usability issues early on.

With the Agile methodology, developers can work in bursts of usually 1 or 2 weeks in order to release small increments of the software, this allows the team to get used to feedback from end users, stakeholders, or in some cases regulatory changes without waiting for the release of the finished product.

• Flexibility: Financial software must follow the regulations of the industries it applies to, and these regulations change over time. Different organizations may have special needs that depend on the structure of finances or reporting requirements.

Agile allows one to make changes if any new regulatory requirements or business needs are arising in an ongoing project; it is thus easier to implement those changes without interfering with the whole development process.

• Team-oriented and Communicative: Financial Management Systems demand teamwork across different contributors. This ensures that the system is not only functional but also very secure.

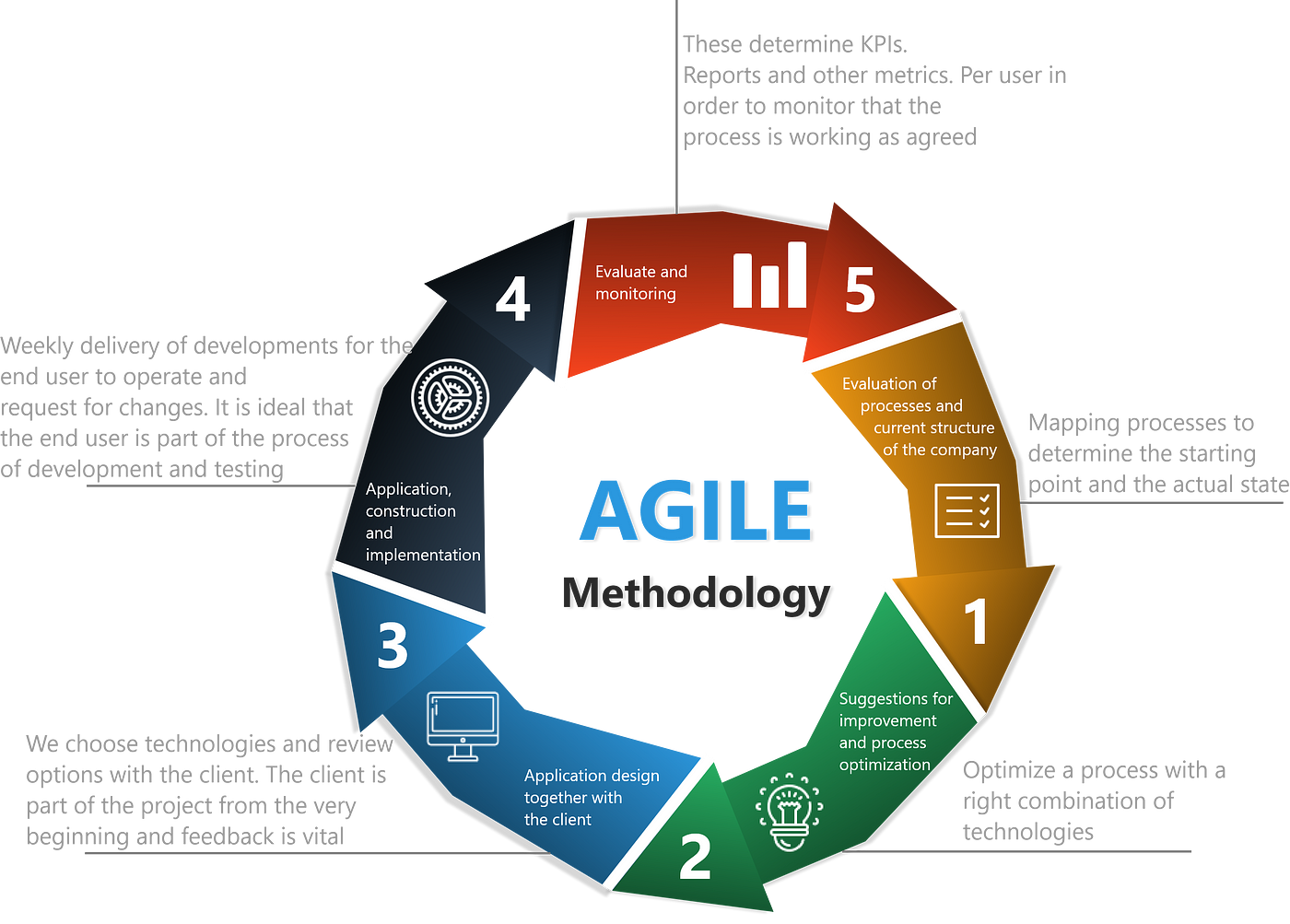
Agile methodology allows for frequent communication and continuous collaboration that builds a mutual understanding of the needs, thus allowing ongoing feedback by the users and stakeholders in improving the ultimate product.

• Risk management: Financial Management Systems work with sensitive data and they must prioritize data integrity, security and compliance to industry regulations before all else. The regular testing approach of the Agile Methodology helps in identifying potential security or data integrity issues in the development lifecycle.

Regular testing greatly reduces the probability of major failures, security breaches, or inconsistencies in data by finding out and resolving problems as they set in, rather than at the end of the project.

It provides flexibility, iterative feedback, and it also focuses on teamwork and communication-all of which are important in creating robust, secure, user-focused financial management systems that respond to changing requirements and continuous user feedback.

In summary, Agile methodology provides flexibility, iterative feedback and it also focuses on teamwork and communication which are essential for creating robust, secure and user focused financial management systems that can adopt to changing requirements and continuous user feedback.



## **3.3 System Requirements**

Analyzing existing financial software to identify gaps and areas that need improvement. The hardware and software needed to run the financial management software application effectively are very significant in ensuring the software achieves its objectives, caters to the user’s purpose, and incorporates the best strategies and practices in safety, growth, and handling of data. This section presents the functional as well as non-functional requirements which discusses the key elements that ought to be incorporated in order to build a usable, effective and secure application.

## **System Design**

The financial management software system design is about putting in place a solid framework that will warrant efficiency, extensibility, safety, and ease of use. This design will be both high and low level in nature explaining the physical arrangements of structural components, their data movement, communication of modules and the systems consist technologies. The following are steps to be taken in designing the system:

* Create system architecture using UML diagrams, use case diagrams and Class diagrams.
* Define the database schema for storing financial data securely and efficiently
* Design the UI/UX with FIGMA

## **3.5 Development Tools & Technologies**

Selection of appropriate development tools and technologies is the key to a robust, efficient, and scalable financial management software solution. This project will be supported by Python-based frameworks and libraries that can facilitate rapid development with security features and complex data processing for various financial management tasks. Python, being flexible, readable, and having huge community support, is among the leading programming languages applied in the financial industry because of its big libraries related to data analysis and visualization.

### ***3.5.1 Frontend:***

**• Django Templates:** The default and inherent templating engine of Django will be utilized in order to create the user interface for developing dynamic web pages which are integrated with backend data.

### ***3.5.2 Backend:***

* **Framework-Django:** A high-level Python web framework following the pattern of Model-View-Template (MVT architecture), Django will be used to manage the core logics of the application, handle the queries at the database, and finally, handling authentication and security of the users.
* **Django REST Framework:** It will be used for the construction of APIs that disseminate financial data to the frontend or other recognized external applications. Apart from that, it is advantageous during the incorporation of third-party financial tools or data sources.
* **Plaid:** This financial data aggregator will be used to provide APIs for securely accessing bank account data while complying with financial data protection standards.

### ***3.5.3 Data Visualization:***

Python's libraries, **Matplotlib and Seaborn**, will be used to show interactive financial charts, graphs, and different visualization elements that can enable finance managers to interpret trends and performance on time and effectively.

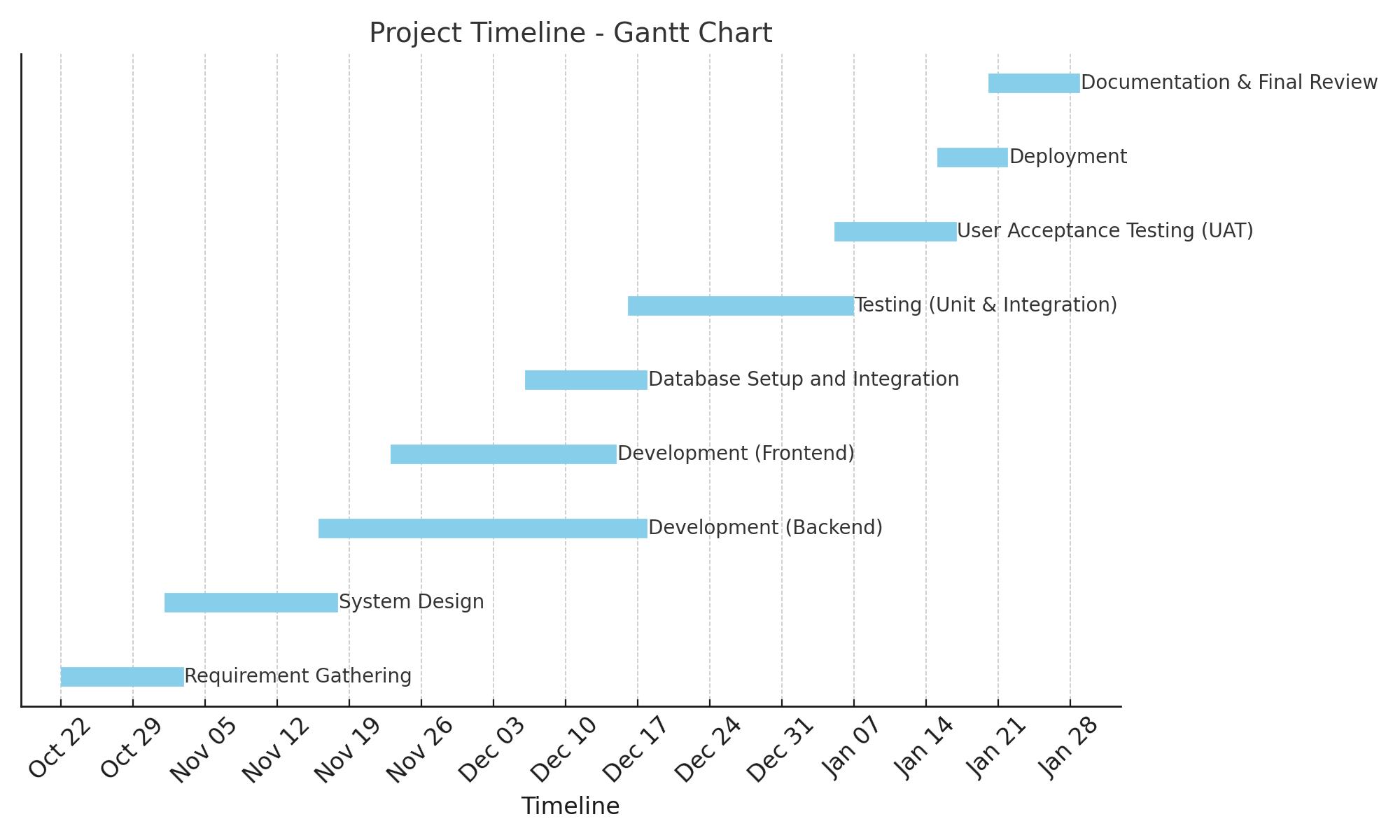
### ***3.5.4 Data Management:***

* + **Pandas:** A Python package that deals with and manipulates volumes of big data in budgeting, forecasting, and financial research.
  + **MySQL:** This is a relational database management system that will be used to store transactional data for reporting and analytics.

### ***3.5.5 Security:***

• **Django-Allauth:** Provides user authentication and authorization through secure login methods. This is particularly pertinent in scenarios involving multi-role users including finance manager, analyst, executive.

## **Project timeline**



## **Conclusion**

This initiative aims to develop software that will help manage, report, and forecast financial data to rectify the inefficiencies that currently exist. This application is targeted at improving the overall efficiency of financial operations, minimizing inaccuracies, and thereby supporting finance managers in decision-making by automating these key activities. It will also bring a new scalable solution particularly targeted for the needs of small and medium-sized enterprises.

## **References**

1. R. A. Brealey, S. C. Myers, and F. Allen, Principles of Corporate Finance, 12th ed., New York, NY, USA: McGraw-Hill Education, 2019.
2. P. Atrill and E. McLaney, Financial Management for Decision Makers, 9th ed., Harlow, UK: Pearson Education, 2020.
3. M. Lutz, Learning Python, 5th ed., Sebastopol, CA, USA: O'Reilly Media, 2013.
4. A. Holovaty and J. Kaplan-Moss, The Definitive Guide to Django: Web Development Done Right, 2nd ed., Berkeley, CA, USA: Apress, 2009.
5. W. S. Vincent, Django for Professionals: Production Websites with Python & Django, 3rd ed., Newton, MA, USA: Leanpub, 2020.
6. F. Ohlhorst, Big Data Analytics: Turning Big Data into Big Money, Hoboken, NJ, USA: Wiley, 2012.
7. S. Kochhar, Practical Django Projects, 2nd ed., Berkeley, CA, USA: Apress, 2011.
8. S. T. Dumais, "Data Visualization Tools for Financial Analysis," Journal of Financial Data Science, vol. 3, no. 2, pp. 45-58, June 2021.
9. R. Gross, Matplotlib for Python Developers, Birmingham, UK: Packt Publishing, 2018.
10. D. Nagle, "Automating Financial Processes with Python: A Comparative Study," International Journal of Financial Computing, vol. 7, no. 1, pp. 14-22, Mar. 2023.
11. Dave Schnitt, “The Challenges of Having a Manual Accounting System” October.2023