**AKENTEN APPIAH-MENKA**

**UNIVERSITY OF SKILLS TRAINING AND ENTREPRENEURIAL**

**DEVELOPMENT**

**FACULTY OF APPLIED SCIENCES AND MATHEMATICS EDUCATION**

**DEPARTMENT OF INFORMATION TECHNOLOGY EDUCATION**

**A RESEARCH PROJECT ON THE DEVELOPMENT OF A**

**CHURCH MANAGEMENT SYSTEM FOR THE PRESBYTERIAN CHURCH OF**

**GHANA - ASCENSION CONGREGATION, KWAHU-ATIBIE**

BY

MICHAEL ACHEAMPONG

PRINCE ASIAMAH BOATENG

JERRY MUSAH

AUGUST, 2023

**AKENTEN APPIAH-MENKA**

**UNIVERSITY OF SKILLS TRAINING AND ENTREPRENEURIAL**

**DEVELOPMENT**

**FACULTY OF APPLIED SCIENCES AND MATHEMATICS EDUCATION**

**DEPARTMENT OF INFORMATION TECHNOLOGY EDUCATION**

A RESEARCH PROJECT ON THE DEVELOPMENT OF A

CHURCH MANAGEMENT SYSTEM FOR THE PRESBYTERIAN CHURCH OF

GHANA - ASCENSION CONGREGATION, KWAHU-ATIBIE

BY

MICHAEL ACHEAMPONG -5191040516

PRINCE ASIAMAH BOATENG -5191040503

JERRY MUSAH -5191040515

AUGUST, 2023

# DECLARATION

**Candidate’s Declaration**

We hereby declare that this project work is the result of our original research and that no part of it has been presented to the Akenten Appiah-Menka University of Skills Training and Entrepreneurial Development, or elsewhere.

Candidate’s Signature ………………… Index Number: 5191040503

Candidate’s Signature ………………… Index Number: 5191040515

Candidate’s Signature ………………… Index Number: 5191040516

**Supervisor’s Declaration**

I hereby declare that the preparation and presentation of this project work were supervised in accordance with guidelines on supervision of project works laid down by the Akenten Appiah-Menka University of Skills Training and Entrepreneurial Development.

Name of Supervisor: Dr. Joshua Dagadu

Signature ……………….………………… Date….………...………

ACKNOWLEDGEMENT

First and foremost, we would like to extend our heartfelt gratitude to our project work supervisor, Dr. Joshua Dagadu, for his invaluable guidance, continuous support and expert insight throughout the entire research process. His mentorship and constructive feedback played a crucial role in shaping and refining our project work.

We are incredibly grateful to the respondents who willingly participated in this research study. Their valuable insights and contributions were instrumental in gathering empirical data, allowing us to analyze and draw meaningful conclusions. The willingness of these participants to share their time and experiences is deeply appreciated.

Finally, we are indebted to our families and friends for their unwavering support, encouragement, and understanding during this challenging and rewarding journey. Their belief in our abilities and constant motivation served as the driving force behind the successful completion of this project.

# ABSTRACT

The Church Management System (CMS) is a software solution designed to streamline and optimize administrative and organizational tasks within the Presbyterian Church Ghana - Ascension Congregation, Kwahu-Atibie. A Church Management System aims to enhance the efficiency and effectiveness of church operations by offering a centralized platform for managing member information, event scheduling, financial transactions, and communication. This system provides a user-friendly interface that empowers church administrators to effortlessly maintain and update member records, ensuring accurate demographic data and seamless communication. Financial management functionalities enable precise tracking of donations, and offerings, promoting transparency and accountability, and aiding in responsible financial stewardship. CMS serves as a communication hub, enabling messages and announcements to be disseminated to the entire congregation, fostering strong community connections.

Overall, the Church Management System presents a cohesive solution that supports the day-to-day operations of religious organizations, fostering greater efficiency, member engagement, financial transparency, and communication. Through its multifaceted features, CMS empowers churches to focus on their core mission while efficiently managing administrative tasks.

TABLE OF CONTENTS

Contents

[DECLARATION 3](#_Toc144634608)

[ABSTRACT 5](#_Toc144634609)

[LIST OF TABLES 9](#_Toc144634610)

[LIST OF FIGURES 9](#_Toc144634611)

[Chapter One Introduction 10](#_Toc144634612)

[Background of the Project 10](#_Toc144634613)

[Problem Statement 11](#_Toc144634614)

[Aims and Objectives of the Project 12](#_Toc144634615)

[Significance of the Project 13](#_Toc144634616)

[Scope of Project 13](#_Toc144634617)

[Organization of Project Report 13](#_Toc144634618)

[Chapter Summary 14](#_Toc144634619)

[Chapter Two Introduction 15](#_Toc144634620)

[Conceptual and Theoretical Framework 15](#_Toc144634621)

[Technology Acceptance Model (TAM) 15](#_Toc144634622)

[Empirical Review 19](#_Toc144634624)

[Information Management System 20](#_Toc144634625)

[Challenges with Manual Information Management 20](#_Toc144634626)

[Church Management Software 21](#_Toc144634627)

[Features of Church Management System 22](#_Toc144634628)

[Benefits of using Church Management System 23](#_Toc144634629)

[Summary 25](#_Toc144634630)

[Chapter Three Introduction 26](#_Toc144634631)

[Research Design 26](#_Toc144634632)

[Research Setting 27](#_Toc144634633)

[Data Collection Instrument 27](#_Toc144634634)

[Observation 27](#_Toc144634635)

[Interview 28](#_Toc144634636)

[Ethical Consideration 28](#_Toc144634637)

[Description of Proposed System 29](#_Toc144634638)

[System Development 30](#_Toc144634639)

[Tools…………………………………………………………………………………….30](#_Toc144634640)

[Development Process 31](#_Toc144634641)

S[ystem Development Life Cycle (SDLC) 33](#_Toc144634643)

[Requirements 34](#_Toc144634644)

[Design and Development 34](#_Toc144634645)

[Validation (Testing) 38](#_Toc144634651)

[Implementation 39](#_Toc144634652)

[Summary 40](#_Toc144634653)

Chapter Four [Introduction 41](#_Toc144634654)

[System Demonstration 41](#_Toc144634655)

[Experimental Setup 43](#_Toc144634656)

[Software and Hardware Requirements 43](#_Toc144634657)

[Experimentation 44](#_Toc144634658)

[System Evaluation 45](#_Toc144634659)

[Summary 53](#_Toc144634668)

Chapter Five [Introduction 54](#_Toc144634669)

[Conclusion 54](#_Toc144634671)

[Recommendations 55](#_Toc144634672)

[Summary 56](#_Toc144634673)

[REFERENCES 57](#_Toc144634674)

[APPENDIX 59](#_Toc144634675)

# LIST OF TABLES

Table 4.1: Funtional Evaluation…………………………………………………………….

# LIST OF FIGURES

[Figure 2.1: Technology Acceptance Model (TAM) 18](#_Toc144634013)

[Figure 3.1 Incremental Model 33](#_Toc144634014)

[Figure 3.2: Use case diagram for proposed system 35](#_Toc144634015)

[Figure 3.3 ER Diagram for proposed system 36](#_Toc144634016)

[Figure 3.4: Context Diagram 37](#_Toc144634017)

[Figure 3.5: Data flow diagram (Level1) 37](#_Toc144634018)

[Figure 3.6: Data flow diagram (Level 2) 38](#_Toc144634019)

[Figure 4.1: Admin log-in 49](#_Toc144634020)

[Figure 4.2: Admin dashboard 49](#_Toc144634021)

[Figure 4.3: Admin member list panel 50](#_Toc144634022)

[Figure 4.4: Admin registration panel 50](#_Toc144634023)

[Figure 4.5: Admin events manager panel 51](#_Toc144634024)

[Figure 4.6: Member log in page 51](#_Toc144634025)

[Figure 4.7: Member dashboard 52](#_Toc144634026)

[Figure 4.8: member tithe/giving panel 52](#_Toc144634027)

**CHAPTER ONE**

## **Chapter Introduction**

This introductory gives an overview of chapter one and it entails the background to the project, statement of the problem, aims of the project as well as project objectives. The significance of the project, the project’s scope, the organization of the project and finally a summary of the entire chapter have all been outlined in detail.

## **Background of the Project**

The advent of the computer and other attendant compliments like the internet has led congregations to use computer technologies to enhance and promote traditional ministries: worship, fellowship, pastoral care, education, mission and community outreach, evangelism and communications. Its significant benefits make it fit for ministry. Ngozi (2011)

Prior to the advent of technology, organizations and groups such as religious bodies were largely faced with problems of collecting, organizing, storing, distributing and retrieving data. Presently, many churches are faced with the challenge of keeping proper, secure and accurate records of their members, finances, and activities among others. Kathambi (2021). The main reason for this phenomenon in most churches is their continued use of the manual paper-to-pen system.

In spite of the significance of technology in our world today with particular reference to religious bodies, there are churches that are still stuck to the traditional pen-to-paper approach when it comes to managing the various data they work with. In our interactions with some church leaders of PCG Ascension Congregation on the need for church management software, we gathered that they were interested in having such a system in place for use in their churches. They bemoaned the numerous challenges confronting them in the manual system they are currently using. A presbyter said*; “we were saddled with a serious problem of retrieving critical information when our offices were flooded and till now, we have not been able to do so. It is frustrating”.*

The main idea behind the development of this CMS is to automate and digitize the data and activities within the church. This project aims to build an efficient, effective and secure computerized system to replace the existing manual system.

## **Problem Statement**

Religious institutions like any other organization collects and manipulates data of their members for various uses in their administrative work. Mostly, the bio-data of members are taken irrespective of the age or status. It is the expectation of religious leaders to see their membership grow and to assess the impact they have had on people and society at large. As membership keeps on increasing the issue of administrative work which includes data management becomes a vital concern to leadership.

Even though the use of technology cannot be downplayed in this era of advancement in all fields, some churches are reluctant to embrace and incorporate technology in their activities. This could partly be attributed to the negative perception people have about technology usage as well as the cost involved in procuring, installing and maintaining technological systems. The current manual system of information management in some churches comes with several disadvantages.

In our interactions with the churches we visited, some church clerks and administrators alluded to the fact that writing records manually is tedious, time-consuming and highly susceptible to errors. In addition, updating records becomes cumbersome or nearly impossible whenever the need arises. Thus, productivity is negatively impacted. For instance, an administrator for an orthodox church we visited opined that *“a member of our congregation upon bereavement discovered that his tithe payment records in his membership card did not tally with that of the roll book of the church resulting in an altercation between him and the leaders”.*

There are instances where pages of books get torn or missing owing to excessive flipping when one intends to retrieve or even add new information to existing records. In some cases, critical data are at the mercy of pests at their storage. These given instances undoubtedly pose serious threats to the integrity, accuracy and security of available data.

It is on the background of these findings that a church management software needs to be developed to enhance productivity and efficiency of church’s administrative tasks. The development and deployment of Church Management Software (CMS) for use in churches will help facilitate automation of their activities. It would ensure accurate, safe, and easy access to records of the church. The system is projected to be of an invaluable solution to the dilemma faced by some churches with regard to record keeping.

## **Aims and Objectives of the Project**

The aim of the church management system is to help in restructuring daily operations with integrations to assist with membership databases, management, finances, scheduling events, reporting and communications.

More specifically, this project seeks to:

1. automate records of congregants in a paperless community.
2. serve as a tool to disseminate information to congregants.
3. track and update financial records i.e., fulfilment of financial obligations such as tithes and offerings for members.
4. create a database for easy access and retrieval of data on members.
5. provide security to the system through an authentication mechanism and ensure the integrity of data within the system.

## **Significance of the Project**

This system will provide safe storage and retrieval of data. It will protect against data loss and ensure data integrity is maintained. The CMS will automate all manual processes involved in eliciting, organizing, processing and storing information. It will save time and improve productivity. Upon successful completion and deployment, the system will be available for use at all times.

## **Scope of Project**

The system’s main users will be church agents (clergy) or administrators and congregants. Users will be able to use the system only after creating user accounts. The

Church Management Software will allow registration of all congregants in the church. The system will in addition create a backup of data. Both financial and non-financial records will be maintained by the system. The intended system would be tested and deployed for use in the Ascension congregation of the Presbyterian Church of Ghana at Kwahu-Atibie in the Eastern region.

## **Organization of Project Report**

This project would be outlined in five chapters beginning with chapter one which encompasses a brief introductory to the chapter, background of the project, problem statement, aims and objectives of the project, the project’s significance, project scope as well as the organization of the project report and a chapter summary.

Chapter two of this study would comprise of an extensive literature review on the development and use of Church Management Software in churches.

The third chapter covers the methodology adopted in the development of the system. This methodology entails the research design, data collection instrument, ethical consideration, system development tools, and the system development life cycle (SDLC).

In chapter four, the focus would mainly be on the system demonstration and system evaluation. Finally, chapter five would caption the conclusion of the overall project, a summary of key findings from the project and proffer some recommendations for further academic studies on the system developed.

## **1.8 Chapter Summary**

This introductory chapter captures primarily the background of the project, statement of the problem, aim and objectives of the project, the significance of the project and scope of project. The next chapter takes a look at the literature review regarding the topic under study.

**CHAPTER TWO**

**LITERATURE REVIEW**

## **2.1 Chapter Introduction**

This chapter focuses on the conceptual and theoretical framework used for the study, the empirical review relevant to the topic under consideration and closes with a brief summary of the entire chapter.

## **2.2 Conceptual and Theoretical Framework**

A theoretical framework is a single formal theory. When a study is designed around a theoretical framework, the theory is the primary means in which the research problem is understood and investigated. Although theoretical frameworks tend to be used mostly in quantitative studies, it can also be deployed in qualitative research. On the other hand, a conceptual framework includes one or more formal theories (in part or whole) as well as other concepts and empirical findings from the literature. It is used to show relationships among these ideas and how they relate to the research study. Conceptual frameworks are commonly seen in qualitative research in the social and behavioural sciences, for example, because often one theory cannot fully address the phenomena being studied.

### **2.2.1 Technology Acceptance Model (TAM)**

The theoretical framework around which this research was conducted is the Technology Acceptance Model (TAM) proposed by Davis (1989). This model is an information systems theory that models how users come to accept and use a technology. The actual system use is the end-point where people use the technology.

The primary objective of TAM was to shed light on the processes underpinning the acceptance of technology, in order to predict the behaviour of and provide a theoretical explanation for the successful implementation of technology. The practical objective of TAM was to inform practitioners about measures that they might take prior to the implementation of systems. To fulfil the objectives of the theory, several steps were carried out (Davis, 1989; Davis, 1993). Davis embarked on the development of the model of technology acceptance by framing the processes mediating the relationship between Information Systems (IS) characteristics

(External factors) and actual system use. The model was based on the Theory of Reasoned Action by Ajzen and Fishbein, which provided a psychological perspective on human behaviour and was missing in the (IS) literature at that time (Davis, 1989; Davis, 1993).

The second step was to identify and define variables and validate measures that would highly correlate with system use. Based on prior empirical literature on human behaviour and the management of information systems, multi-item scales for perceived ease of use and perceived usefulness were developed, pre-tested, and validated in several studies. It was hypothesised that the two constructs were fundamental determinants of user acceptance, due to evidence in previous research (e.g. (Johnson & Payne, 1985; Payne, 1982; Robey, 1979). The research suggested that an individual's decision to perform a behaviour is the result of the analysis of the benefit that they expect to receive from the behaviour compared to the effort/costs they put in to perform the behaviour (Johnson & Payne, 1985; Payne, 1982). This means that the use of the information system is determined by an evaluation of the trade-off between the perceived usefulness of the system and the perceived difficulty of using it (Davis, 1989).

Perceived usefulness was defined as the individual's perception of the extent to which the use of a given technology improves performance. The conceptualisation of this construct stemmed from Bandura’s concept of outcome judgement, which refers to an individual's expectation of a positive outcome triggering behaviour (Bandura, 1982). Perceived usefulness was operationalised based on evidence confirming the effect of system performance expectancy on system usage (Robey, 1979).

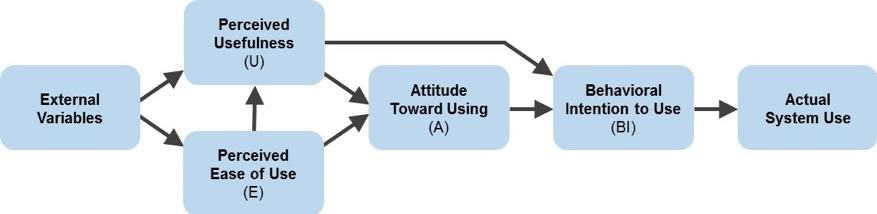
Perceived ease of use was defined as the degree to which a person believes that using a particular system is free of effort (Davis, 1989). This construct derived from the self-efficacy concept, which refers to a situation-specific belief about how well someone can execute actions for the prospective task (Davis, 1989; Bandura, 1982). It was suggested that self-efficacy had a predictive role in decision-making about technology use (Hill, Smith & Mann, 1987). Also, perceived ease of use shared a similarity with the complexity factor theorised in the innovation diffusion literature as a barrier to innovation adoption. It was defined as the degree to which individuals find the innovation difficult to understand and use (Mahajan, 2010).

The validity and reliability of the constructs were assessed by testing the contingency of the self-reported usage of IS on the two proposed factors in the organisational context. The developed scales showed excellent psychometric properties. The model was further validated, by confirming significant relationships between perceived usefulness, perceived ease of use, intention and use behaviour (Davis, 1989).

According to TAM, technology acceptance is a three-stage process, whereby external factors (system design features) trigger cognitive responses (perceived ease of use and perceived usefulness), which, in turn, form an affective response (attitude toward using technology/intention), influencing use behaviour (Davis, 1989; Davis, 1993). TAM represents the behaviour, as the outcome predicted by perceived ease of use, perceived usefulness and behavioural intention (Figure 1). Perceived ease of use and perceived usefulness capture the expectations of positive behavioural outcomes and the belief that behaviour will not be labour consuming (Davis, 1989).

According to a follow-up study, behavioural intention can be substituted by the attitude toward behaviour (Davis, 1993), which is an affective evaluation of the potential consequences of the behaviour (Ajzen, 2011). The higher the affective response, the higher is the likelihood that the behaviour will take place. The effect of perceived usefulness on actual use can be direct, which underscores the importance of the variable in predicting behaviour. Although perceived ease of use does not affect use behaviour directly, it underpins the effect of perceived usefulness (Davis, 1993). The model implies that if an application is expected to be easy to use, the more likely it is that it will be considered useful for the user and the more likely it is that this will stimulate the acceptance of the technology (Davis, 1989; Davis, 1993).

The development of the model and measures for technology acceptance have made significant theoretical contributions and have had a great practical value. The application of the model for testing IS usability has made it possible to evaluate the motivation of users to adopt a range of technologies (Marikyan & Papagiannidis, 2023), which had not been done before due to a lack of validated subjective measures. The development of constructs which had a strong and significant correlation with use behaviour made it possible to understand the cognitive and affective factors mediating the effect of system characteristics on technology acceptance (Davis, 1989).



# Figure 2.1: Technology Acceptance Model (TAM)

There have been some extensions to the TAM theory over the course of time namely; TAM2 and TAM3. TAM2 consisted of five additional exogenous variables and two moderators. The new constructs and moderators incorporated in TAM2 were: subjective norm, image, job relevance, output quality, result demonstrability, experience and voluntariness. TAM3 also introduces three new moderation effects of experience on the relationships between a) computer anxiety and perceived ease of use, b) perceived ease of use and perceived usefulness, and c) perceived ease of use and intention to use. The effect of experience on perceived ease of use was not tested when developing TAM2, although this perception is weakened when people attain hands-on experience and knowledge about the system (Marikyan & Papagiannidis, 2023).

From a practical point of view, TAM is useful for vendors to estimate the potential demand or stock supplies of new information technology products (Davis, 1989). Practitioners can use TAM to facilitate the acceptance of technology. By understanding the degree to which technology is useful and easy to operate by consumers, they can design consumer-oriented IT products (Davis, 1989). In addition, the understanding of the antecedents of perceived usefulness and perceived ease of use, proposed by TAM2 and TAM3, can help managers make informed decisions about the strategies on technology implementation in organisations. The models can be applied to guide the development of pre-implementation (actions leading to the actual roll-out of a system) and post-implementation interventions (actions following the actual deployment of the system) to address acceptance rates (Venkatesh et al, 2012).

Some criticisms raised against TAM research point to its methodological issues, some limitations in the theory’s applications and the focus on the aspects of systems’ utilisation that diverted attention from other important factors and relationships (Marikyan & Papagiannidis, 2023). Nonetheless, the limitations cannot overshadow the contributions of the theory. TAM has been shown to be theoretically resilient and to have a strong predictive power to assess individuals’ intention to use for almost three decades. TAM became the first theory explaining why individuals use information systems, which was once badly needed for IS research and practice (Goodhue, 2007).

## **2.3 Empirical Review**

Churches have been an integral part of communities for centuries, providing spiritual guidance and support to their members. With the growth of technology and the internet, many churches have recognized the need for more effective management systems to handle the various aspects of their operations. Church management systems (CMS) have become an essential tool for many churches in managing their operations. This literature review examines the existing research on church management systems, including the features and benefits.

### **2.3.1 Information Management System**

An information management system (IMS) refers to any framework of software that facilitates the collection, storage, organization, and distribution of information. Information management encompasses a series of organizational activities about acquiring, storing, and distributing information to stakeholders. (Indeed Editorial Team, 2023). Information, in this case, is any detail that allows a business to make a decision. For example, the number of sales that a business records over a week. The information in question can be in different formats, including pictures, audio, and video. An excellent management system allows information to be accessed in a timely and accurate fashion so that key stakeholders can use it to make informed decisions about the business. Religious organizations have been in need of IMS to automate their administrative tasks to save time, promote efficiency and reduce costs.

### **2.3.2 Challenges with Manual Information Management**

The existing manual approach in keeping and managing member information in most churches is confronted with several mishaps. In Ghana, there are churches operating the traditional pento-paper system in managing information on members. The consequences associated cannot be downplayed. Bolanke (2022), highlighted a number of shortcomings with the manual system.

The key findings from Bolanke includes;

1. Manual system is highly prone to human errors;
2. The manual system is time-consuming;
3. It lacks real-time insights. it difficult to spot trends or issues early and make the necessary changes to keep business on track. This also limits decision-making ability, as one does not have access to the most up-to-date information;
4. Difficulty in tracking changes manually;
5. The high incidence of inaccurate data;
6. Security risks which could easily lead to data breach and can have devastating consequences; and
7. Difficulty in scaling and high costs.

### **2.3.3 Church Management Software**

In the words of Allan (2023), church management software (CMS) is a type of computer software specially designed to help churches and religious groups manage, automate and organize their daily operations. Beyond that short definition, this type of software leverages computing technology to handle tasks and activities of churches such as managing membership databases, emails and communication, worship presentation programs, community and religious events, fundraising, finances, and report generation, among others.

CMS includes solutions with a wide range of functionalities related to running a religious organization. The feature set can include everything from member relationship management and reporting to event planning and donor management. These solutions help churches manage the most vital portions of their operations, improving processes and allowing them to build better connections with their members. The typical users of CMS are religious leaders. These solutions can be essential to users, who can use a single solution for nearly all of their operational needs.

Bolanke (2022) established that while some church management products are all-in-one management solutions, others have specific use cases involving certain aspects of church operations. For example, some products focus on managing finances for a church, including staff payroll, accounting, and community donations. Other products might only focus on managing people and membership data as well as for presentation purposes to curate multimedia content.

To maximize the benefits of a CMS, several best practices have been identified. According to Allan (2023), one best practice is to involve stakeholders in the selection and implementation of the CMS, to ensure that the system meets the specific needs of the church. Another best practice is to ensure that the CMS is user-friendly and easy to use, to encourage adoption by church administrators and members. Church administrators should also ensure that the CMS is regularly updated and maintained to address any security or performance issues.

Allan (2023), summed up the functions of church management software into four general categories namely: (i) streamline common administrative tasks; (ii) track and manage resources; (iii) foster communication and; (iv) monitor growth of the congregation.

### **Features of Church Management System**

Although some church management suites may have more functionalities than others, there are features that are common to all. Allan (2023), posited some of these common features as follows:

i. Membership management. Keeps updated information on members, volunteers and visitors such as contact numbers, addresses and activity involvement. ii. Contributions/donor management. Provide secure collection and management of individual and major donations as well as pledges and upcoming donations.

1. Financial accounting. Enables managing, tracking and reporting of finances like revenues and expenditures.
2. Communication tools. Facilitates community-wide communication and outreach activities to increase member participation.
3. Events management. Organizes and manages event activity and planning, payment process support, event listing search, tracking and reporting, and prevents double booking.
4. Check-in and child security management. Tracks member and visitor attendance at events such as seminars, camps and retreats. Many CMS provide child check-in functionality to keep track of children in specific church locations and events.

### **Benefits of using Church Management System**

Church management software brings with it a number of productivity and optimization functions that efficiently assists any church or growing faith-based community. As a church grows, so does the load of tasks and processes that has to be properly managed to keep the church operating smoothly. Allan (2023), outlines key benefits of implementing the usage of

CMS as follows:

* Centralized database - having all contact information, member details, records of collection, financial reports and other important files and documents in one secured and easily accessible location is the first order of business of organizing things. So much time and resources are saved by having one information repository.
* Reduced exposure to fraud - accountability especially when it comes to church finances is strengthened. Fraud and embezzlement of funds can happen in the church. Accounting and reporting features in a CMS can be set with mandatory double entry procedures and traceable audit trail. An administrator can also enforce internal controls such as multi-level passwords or authentication protocols and have a log to monitor users and changes made.
* Church-centric platform – availability of features specially designed to address the unique functional needs of a church. A business software simply will not do because of the distinct nature of church goals, activities and operation. One study pointed out that 70% to 80% of newly formed faith-based communities fail after the first year, and that even established churches lose some of their members after five to seven years. A major reason for this is the lack of tools or resources to manage the many diverse administration and growth aspects of churches. CMS fills in this shortage.
* Management dashboard - a full-featured management dashboard provides a unified way to have all important church information in a single location. No more looking for data and records from various applications, storage and systems. There exists the convenience of a centralized database that can be readily accessible especially if it is a cloud-based platform.
* Statistics and reporting - eliminate guesswork in managing church operations and have reports and statistics right at your fingertips. One can view financial reports and get membership insights and determine where growth factors or where bottlenecks lie so

you can work on improving them. These provide relevant insights and guide in making informed decisions.

* Events management - big church and community events can be difficult to plan, schedule and put together which is why CMS software feature tools to facilitate booking, monitoring of assets, and tracking of contributions. There are also communication features to easily inform and update members of any changes in schedule, activities and programs.
* Check-in management - this nifty functionality is an immense benefit when it comes to large gatherings and events where every member including children must be accounted for. A check-in system ensures that events are well organized, and that participants are physically present and well documented, reducing instances of missing or lost members.
* Contribution management - donations, contributions and collections go a long way in maintaining and growing a church, hence it is essential that everything that goes into the church coffers are accurately tracked and monitored. Fortunately, most CMS carry features to make it easy to keep tabs of donations and generate updated reports of collections.

Despite the benefits of using CMS, several challenges have been identified in implementing and using these systems. A study by Bolanke (2022), identified several challenges facing churches in adopting CMS, including the lack of resources, resistance to change, and the complexity of some systems. Churches may also face challenges in selecting the right CMS that meets their specific needs and budget. Another challenge is ensuring data security and privacy, especially in light of the increasing incidence of cyber-attacks.

## **Summary**

The chapter discussed the Technology Acceptance Model which was used as the theoretical framework for this project. A detailed discussion on the significance, extensions and criticisms of TAM were presented. The empirical review captured the challenges with using manual information management, the meaning of Church Management Software, its features and benefits as outlined by other authors in previous research studies relevant to the topic.

**CHAPTER THREE**

**METHODOLOGY**

## **3.1 Introduction**

This chapter entails a detailed description of the research approach, the data collection instruments used, the system development life cycle, the proposed system input and output, users and system development tools.

## **3.2 Research Design**

Research design refers to the overall plan, structure or strategy that guides a research project, from its conception to the final data analysis. A good research design serves as the blueprint for the researcher to collect and analyze data while ensuring consistency, reliability, and validity throughout the study. This research project is based on the action research method.

Action research is a research method that aims to simultaneously investigate and solve an issue. (Tegan, 2023). In other words, as its name suggests, action research conducts research and takes action at the same time. Action research prioritizes reflection and bridges the gap between theory and practice. Action research is focused on solving a problem or informing individual and community-based knowledge in a way that impacts teaching, learning, and other related processes. It is less focused on contributing theoretical input, instead producing actionable input. Action research was selected for this research because it provides an immediate and actionable path forward for solving problems. In addition, action research can be used with quantitative, as well as, qualitative data and the possibility to gain in-depth knowledge about a problem.

## **3.3 Research Setting**

The Presbyterian Church of Ghana – Ascension Congregation is located at Kwahu-Atibie on the Kwahu Ridge in the Eastern Region of Ghana. The church is predominantly youthful and is recognized as one of the vibrant churches in the area. Both the administrative and finance departments of the church are the key focus for this research. The church operates a manual system with regards to information management. Members are registered after filling a membership form before their details are entered into the roll book by the administrator. The roll book serves as the main record resource for keeping personal and financial records on members of the church. Tithe payments are recorded in the membership cards of congregants and each member is issued with a receipt upon payment. Weekly offerings and donations are recorded in the church’s weekly analysis book. All announcements/upcoming events are

entailed in the announcement log book. Besides these, important certificates such as baptismal, confirmation and marriage certs are recorded, issued and stored manually. The church’s over dependence on the manual system is a of a significant concern to not only its leadership but the entire congregation since all vital information are deemed to be at risk.

## **3.4 Data Collection Instrument**

The term data collection instrument refers to tools or devices used to gather relevant data for the purposes of carrying out a research activity. Common data collection tools in research include, questionnaire, interviews, documentary analysis and observation. Observation and interviews were the tools deployed to elicit data for this research.

### **3.4.1 Observation**

Sometimes researchers will go to a certain place to watch and examine what is happening in that setting and document what they see. This qualitative method of gathering data is referred to as observation. In an observational study, researchers study how participants perform certain behaviours or activities without instructing them on what methods or behaviours to choose. Observation method may involve watching, listening, reading, and recording behaviour and characteristics of a phenomenon. Advantages of the observation method include direct access to research phenomena, high levels of flexibility in terms of application and generating a permanent record of phenomena to be referred to later.

### **3.4.2 Interview**

An interview is a conversation for gathering information. A research interview involves an interviewer, who coordinates the process of the conversation and asks questions, and an interviewee, who responds to those questions. Interviews can be conducted face-to-face or over the telephone. The internet is also emerging as a tool for interviewing. Interviews are an appropriate method when there is a need to collect in-depth information on people’s opinions, thoughts, experiences, and feelings. Interviews are useful when the topic of inquiry relates to issues that require complex questioning and considerable probing. Face-to-face interviews are suitable when the target population can communicate through face-to-face conversations better than they can communicate through writing or phone conversations. Interviews can be structured or unstructured depending on the discretion of the researcher and the nature of the study under consideration. In a structured interview, the interviewer asks a set of standard, predetermined questions about particular topics, in a specific order. However, in an unstructured interview, the interviewer has no specific guidelines, restrictions, predetermined questions, or list of options. The interviewer asks a few broad questions to engage the respondent in an open, informal, and spontaneous discussion.

## **3.5 Ethical Consideration**

Ethical considerations in research are a set of principles that guide your research designs and practices. These principles include voluntary participation, informed consent, anonymity, confidentiality, potential for harm, and results communication. In the development and implementation of this project, the researcher recognizes the paramount importance of upholding data privacy and security as fundamental ethical principles. The researcher commits to safeguarding the personal information and sensitive data of individuals, respecting their rights to privacy and confidentiality. In addressing the ethical considerations, the following measures have been adopted;

* Informed consent will be obtained from the appropriate parties before collecting, storing, or processing their personal data.
* Robust security measures will be implemented to protect data against unauthorized access, loss, disclosure or alteration. To this end, data encryption, access controls and other industry-standard security practices to ensure data integrity and confidentiality.
* User control and transparency options will be provided to control and manage personal user data. Users will have the ability to review, modify, or delete their information, and we will provide transparent mechanisms for accessing, correcting, or withdrawing consent for data processing.
* Ethical use of data. Data collected shall be used for legitimate purposes defined within the project scope. Any potential secondary use or sharing of data will be conducted with explicit consent or in compliance with relevant regulations.

## **3.6 Description of Proposed System**

The proposed system is designed with the goal of eliminating the identified overheads associated with the manual system. Specifically, the proposed system will exhibit the following features and functionalities:

1. create a database to securely store data and for retrieval.
2. Users of the system will be a church administrator and members of the church on successful creation of user accounts.
3. Only an admin can register a member, create a user account, add a new admin and give admin level access.
4. Announcements, events and devotionals shall be posted by system admin.
5. An admin can generate and print records.
6. Registered members with user accounts can access the system with valid credentials.
7. A user when logged in can give donations, offerings and pay tithe.
8. Users can read devotionals, announcements and view upcoming events of the church.
9. Furthermore, a user can update his/her username and password on the user dashboard.

## **3.7 System Development**

### **3.7.1 Tools**

The following tools were deployed in developing the proposed system:

**HTML**: HTML is the standard markup language for Web site creation as well as interfaces. These websites can then be viewed by anyone else connected to the Internet. It is relatively easy to learn, with the basics being accessible to most people in one sitting; and quite powerful in what it allows you to create.

**PHP**: PHP: Hyper-Text Pre-processor is a server-side scripting language usually written in an HTML context. Unlike an ordinary HTML page, a PHP script is not sent directly to a client by the server; instead, it is parsed by the PHP binary or module. HTML elements in the script are left alone, but PHP code is interpreted and executed. PHP code in a script can query databases, create images, read and write files, and talk to remote servers— the possibilities are endless.

**Cascading Style Sheet (CSS)**: Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML or XML. It is a simple mechanism for adding style (e.g., fonts, colors, spacing) to web documents.

**SQL**: SQL is a database computer language designed for storing, manipulating and retrieving data in a relational database. SQL stands for Structured Query Language.

SQL is the standard language for Relational Database Systems. All the Relational

Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, Postgres, and SQL Server use SQL as their standard database language.

**Sublime Text**: Sublime Text is a shareware cross-platform source code editor. It natively supports many programming languages and mark-up languages. Users can expand its functionality with plugins, typically community-built and maintained under free-software licences.

**XAMPP**: XAMPP is a free and open-source cross-platform web server solution stack package consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes

transitioning from a local test server to a live server possible.

### **3.7.2 Development Process**

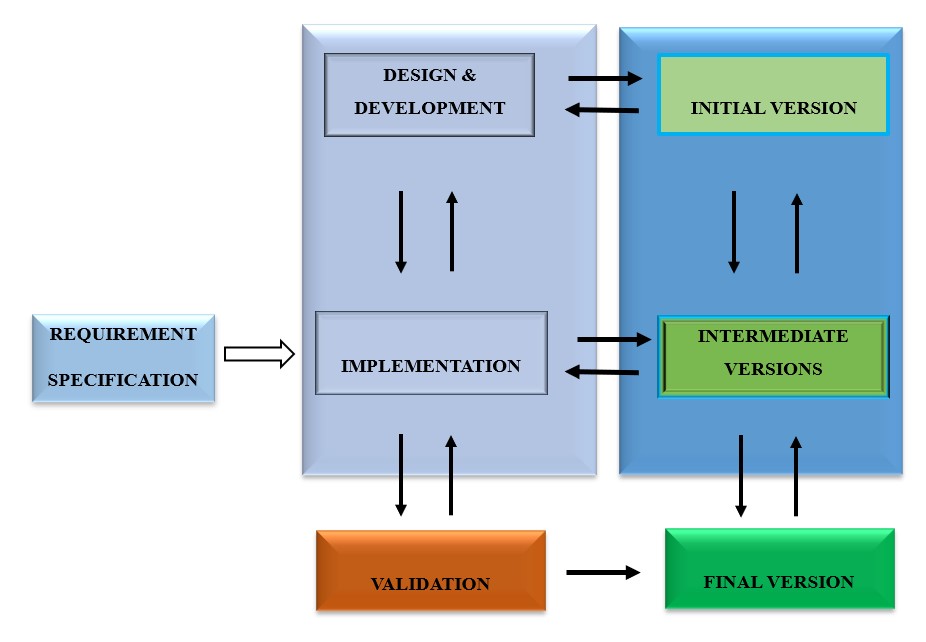
INCREMENTAL DEVELOPMENT

Incremental development is based on the idea of developing an initial implementation, exposing this to user comment and evolving it through several versions until an adequate system has been developed. Incremental software development, a fundamental part of agile approaches, is most suitable for businesses, e-commerce, large groups, and personal systems. Each increment or version of the system incorporates some of the functionalities needed by the customer. The Incremental model interleaves some major process activities. Requirement specification, design and development, validation (testing) and implementation process activities are performed concurrently rather than separately, with rapid feedback across each activity.

By developing the software incrementally, it is cheaper and easier to make changes in the software as it is being developed. Each rollout of the versions or increments of the system tries to factor in the major functionalities needed by the customer. This approach generally incorporates the most required and basic functionalities of the software that the customer urgently needs. By doing this, the user is exposed to the system at an early stage of the development process so that if the current prototype does not meet the user’s requirement, then the current version will be revised and possibly new functionalities incorporated. Some merits of the incremental model are;

* A more rapid delivery and deployment of useful software to the customer is possible, even if all of the functionality has not been included.
* It is easier to get customer feedback on the development work that has been done.
* The cost of accommodating changing customer requirements is reduced.

One criticism of the Incremental model is that one can develop a system incrementally and expose it to customers for comment, without actually delivering it and deploying it in the customer’s environment. Incremental delivery and deployment mean that the software is used in real, operational processes. This is not always possible as experimenting with new software can disrupt normal business processes.



# Figure 3.1 Incremental Model

## **3.8 System Development Life Cycle (SDLC)**

SDLC is a methodology used in software engineering and system engineering to guide the development process of software systems and information systems (Sommerville, I.2016). SDLC is a structured approach used to plan, design, implement and maintain information systems within an organization. It has been considered to be the basis for most methodologies (Davis et al., 1988; Orr, 1989). Adhering to a properly-defined methodology enables a project to provide better estimates, deliver stable systems, create a clear understanding of the task ahead, and identify pitfalls earlier, allowing for ample time to make adjustments. The Incremental Development is the SDLC methodology used in this case.

### **3.8.1 Requirements**

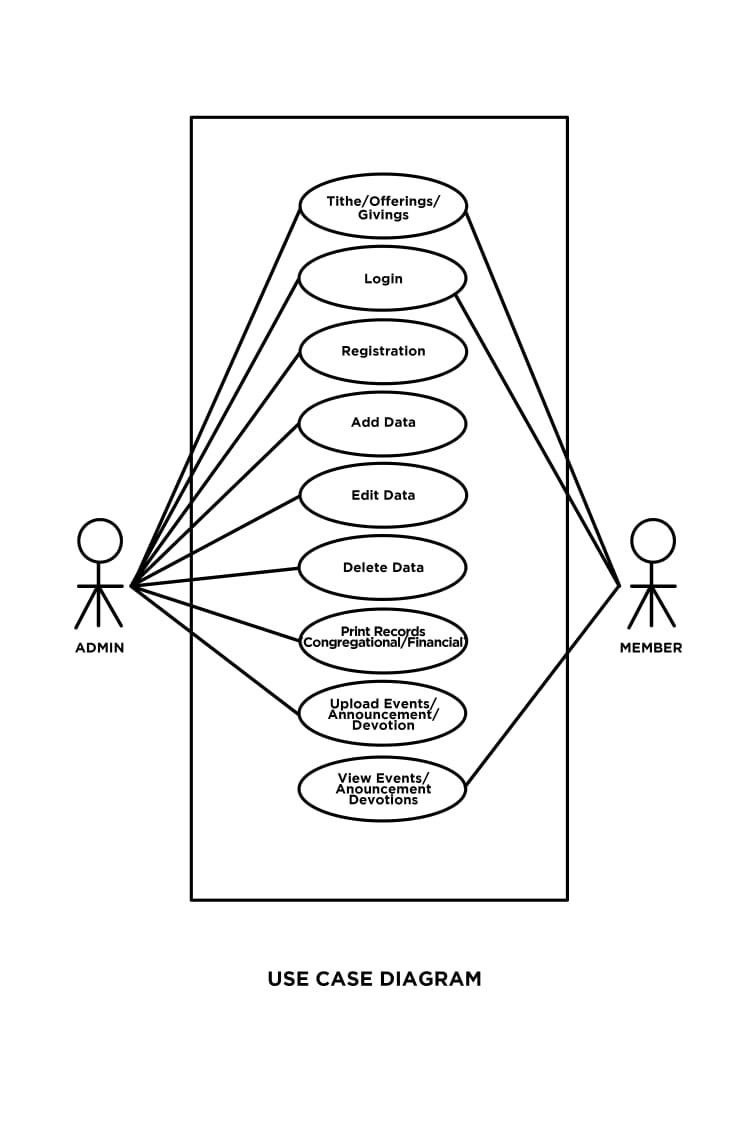
At this phase, all requirements for the proposed system will be obtained through initial observation of the as-is (paper-based) system upon regular visits to the church during services over a time period, and further conducting interviews to elicit requirements from potential users. This is the phase where we will determine the user requirements in consultation with the potential users and then deduce the functional requirements from it. The next step is to analyze these requirements and determine which ones are the critical features that need to be designed and implemented first and expose them to the user for criticism or which features will form part of the first version to be implemented.

### **3.8.2 Design and Development**

The design phase is the description of the structure of the software to be implemented, the data models and the structures that will be used by the system, the interfaces between the system components and sometimes the algorithm that will be used. All these will be done iteratively as the process goes on. Key components of the system design are the user interface design (UI) and the database design, the various tables and fields of the database.

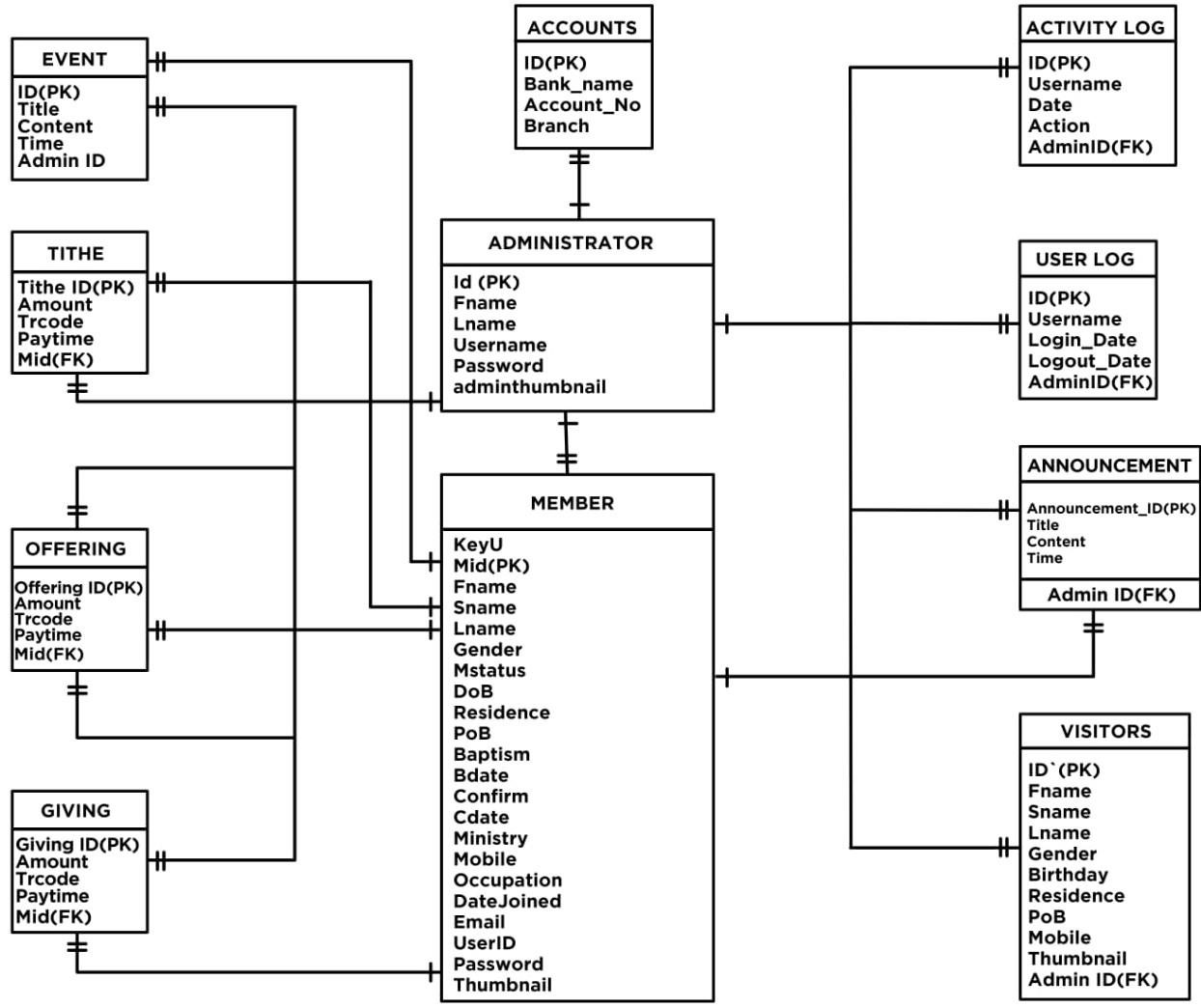
Most software interfaces with another software system. These include the database application which is independent of the User Interface, the platform or Operating System on which the system will run and many more. A detailed design of the proposed system using use case diagram and E-R diagram/Database Schema respectively.

Use case: use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behaviour (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (i. e. use case diagram). A key concept of use case modelling is that it helps us design a system from the end user's perspective.



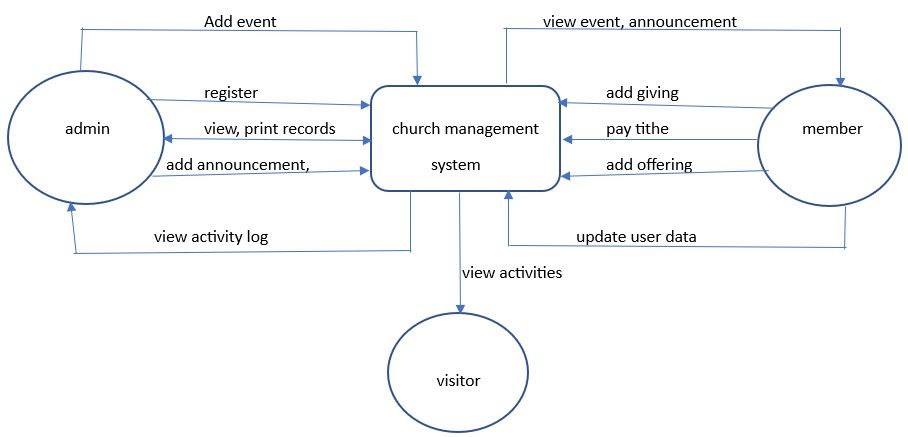
# Figure 3.2: Use case diagram for proposed system

Entity Relationship Diagram: An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. It develops a conceptual design for the database. It also develops a very simple and easy to design view of data. In ER modeling, the database structure is portrayed as a diagram called an entity-relationship diagram.

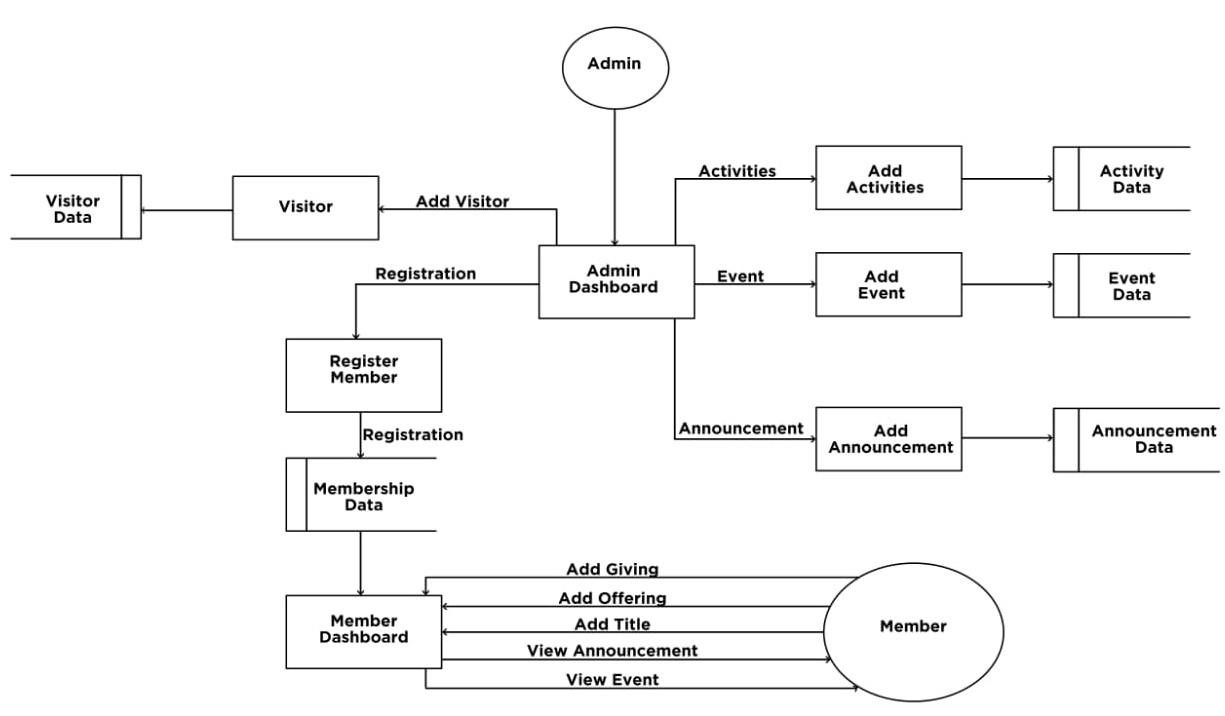


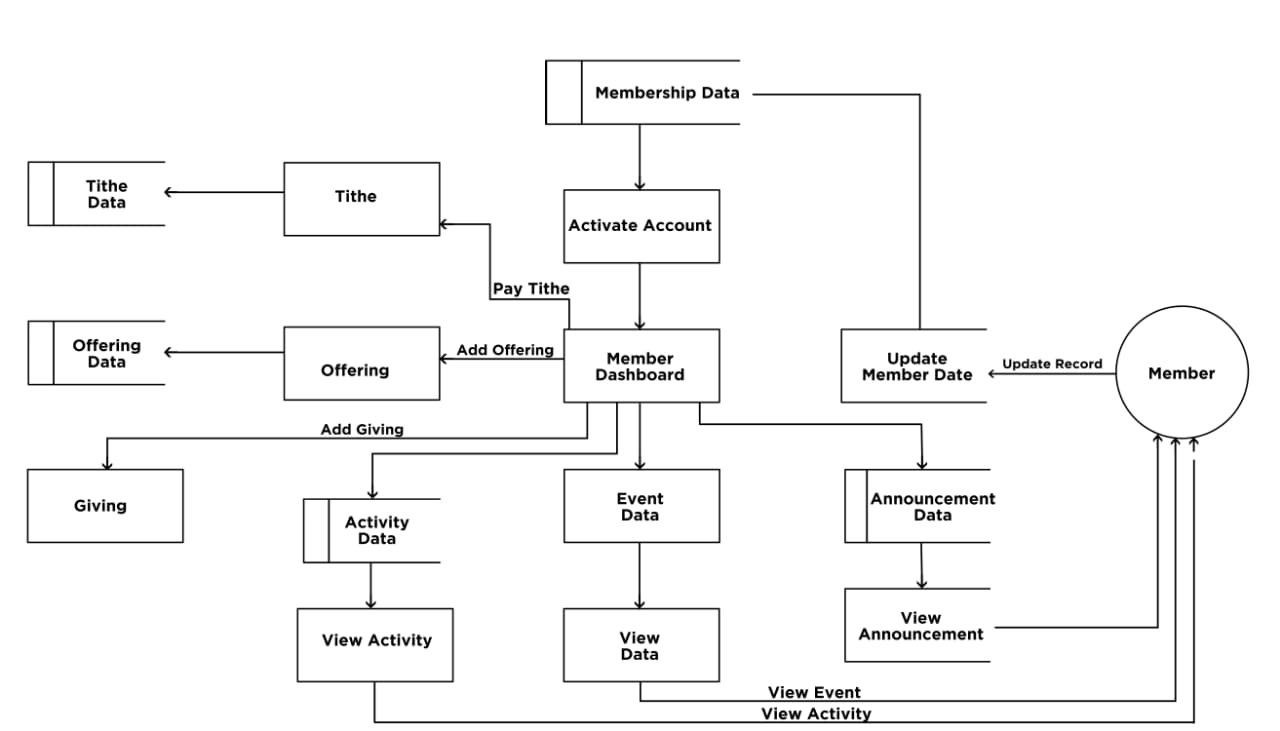
# Figure 3.3 ER Diagram for proposed system

Data Flow Diagram (DFD): A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled.



# Figure 3.4: Context Diagram

Figure 3.5: Data flow diagram (Level1)



# Figure 3.6: Data flow diagram (Level 2)

### **3.8.3 Validation (Testing)**

At this phase, the performance of the system will be tested to ascertain whether it conforms to both its specification and also meets the user’s expectations. The various component of the system will be tested individually (unit testing) to see their performance. An integration testing will then be conducted on the system’s performance when various units are integrated. The next step of validation after this will be to input some simulated data into the system to see that the data processing of the system is devoid of any logical error and that it generates the required results. Final testing will be done by the user using real-life data. This is known as verification of the system by the users. Users will be allowed to use the system and report any failure or malfunction to the developers and may request for new functionalities to be added to the next version of the system. This is done taking into consideration scope or feature creep. Once the system is validated and verified as fit for business, it will be deployed for use. This version may not incorporate all the functionalities of the system as these will be developed in increments.

### **3.8.4 Implementation**

This phase is critical in the overall system development and deployment process. It involves translating the system design and requirements into a fully functional and operational solution. We will begin with writing the actual codes based on the intended system design, integrating the modules, and perform testing to ensure the system functions as intended. The following tools and technologies will be used to execute this stage:

* The programming language to be used for developing this system is PHP. PHP as a server-side scripting language is widely due to its wide platform support, rich frameworks and libraries, excellent database integration and versatility. It is also known for its simplicity and ease of use.
* HTML and CSS will be used to design the user interface (UI). Key reasons for choosing these tools for the UI design include their cross-platform compatibility, easy integration, and creating visually appealing and interactive user interfaces.
* The database development tool adopted is MySQL. MySQL is widely used for database development due to its combination of factors such as its open-source nature, scalability, reliability, performance, and extensive community support.
* XAMPP will be used to test the application on a web-server. XAMPP is widely used because it provides a convenient and all-in-one solution for setting up a local web development environment with Apache, MySQL, PHP, and Perl, enabling developers to easily develop and test web applications locally before deploying them to a live server.
* This project’s integrated development environment tool selected is Sublime Text 3. It is one of the popular code editors used by developers because it is lightweight, fast, customizable and extensible, and features a command palette and powerful search.

## 3.9 Summary

This chapter focused on the system methodology to be deployed in designing, developing and implementing the system. Action research design was considered for the study. Observation and interview were the data collection tools adopted. The chapter further discussed the system development (i. e. Incremental development), the tools and technologies needed to develop and implement the system. The entire System Development Life Cycle (SDLC) was clearly

outlined.

**CHAPTER FOUR**

**SYSTEM DEMONSTRATION AND EVALUATION**

## **4.1 Introduction**

This chapter entails the system demonstration (experimental setup and experimentation), and the system evaluation (system testing methodologies) of the Church Management System. The functional and non-functional requirements of the system have also been outlined in-depth. The chapter captures a detailed account on how the system was tested in the target organization

(PCG Ascension Congregation, Kwahu-Atibie) and an evaluation of the system’s usability and performance.

## **4.2 System Demonstration**

The CMS is to automate the administrative and some financial tasks for the PCG Ascension Congregation, Kwahu-Atibie. This is a web-based system that works online. It can be accessed on a computer or a smartphone offering its users various functionalities and features.

The system’s architecture comprises of a graphical user-interface (front-end) and linked to the back-end (database). Registering members, updating member records, carrying out financial commitments in the form of tithings, offerings and donations are the key functions of the system. The system administrator(s) and users have access to the system provided there’s internet connectivity and they have the required login credentials.

**How the System Works:**

1. Admin Dashboard
   * + An admin logs in with valid credentials to access the admin dashboardanua.
     + The admin dashboard contains a panel for registering new members, edit and update member records, create event, publish announcement.
     + There is a section for viewing tithes, donations and offerings paid by members on the system.
2. Member Registration
   * After a successful login, the admin can register a member on the registration panel by entering the required data as provided on the system.
   * the admin sends member username and password to the member for first time access after which the member is required to change his/her password after log in.
3. Record Update
   * Admin updates existing records of users by editing or deleting member

profile.

1. Event Publication and Announcement
   * Events and announcement are created and published by the system admin to

all users.

* + Users receive system notification of new events and announcements on log in.
  + The admin also has the privilege to update any publication already sent and users are notified on the system.

1. Member Profile
   * A registered user logs in with valid credentials to access the user dashboard.
   * After a successful login, users update their profile by uploading a passport-sized photograph and change password.
2. Financial Commitments
   * User pays tithe, give offering or donation electronically via an online payment system.
   * Respective payments are updated on the user dashboard.
3. Receipts:

- Admin generates an electronic receipt for all payments and prints out to user.

### **4.2.1 Experimental Setup**

The team at this phase experiments with the entire performance of the system. Focus is placed on the platform compatibility of the system, and both software and hardware requirements for accessing the system. This is followed by a system testing using testing methodologies.

#### **4.2.1.1 Platform Compatibility**

The system was tested on Windows OS to determine its compatibility and it run successfully. The CMS is compatible on Windows version 7 or higher, 1.8 GHz processor speed, and Net framework of 4.0 or newer.

### Software and Hardware Requirements

Minimum hardware specifications for running this CMS are:

* Desktop or laptop computer or smartphone.
* 1.0 GHz processor speed (dual core or higher)
* 4GB RAM
* 100GB ROM
* Printer

To run the system on Windows platform, Apache Xampp Server application needs to be installed to launch the system on a web server.

### 4.2.2 Experimentation

* **System Installation**

System installation was carried out by installing XAMPP server application on a laptop running on Windows 11 that met the hardware and software requirements earlier stated. Upon successful installation, we proceed to run the system in a controlled environment.

* **System Launching**

A web browser is required to run the system online. Any modern web browser can be installed for this purpose in addition to installing XAMPP. A step-by-step guide to run the system has been outlined as follows to explore features and functions of the system:

1. XAMPP Setup

Open XAMPP control panel.

Start Apache and MySQL modules

Ensure server is running with a green highlighted background on the Apache and

MySQL modules.

1. Database Setup

Open web browser. Go to <https://localhost/phpmyadmin>

Create new database for the system. NB: name the database. E.g., cman

Import the system SQL file to set up the respective relations and records.

1. Configuration

Update database connection settings i. e. username, password, database name, host in the configuration files to match XAMPP configuration.

1. System Access:

To access the system, open web browser and enter the URL <http://localhost/cman>

Log in to the system with valid credentials.

Explore system features and functionalities and test some scenarios. This is to ascertain whether or not the system performs its functions as prescribed.

## **4.3 System Evaluation**

**System testing**

System testing is a crucial phase in the software development life cycle where the complete software system is tested as a whole to ensure that all components work together as expected and meet the specified requirements. This type of testing is performed after unit testing and integration testing have been completed.

The primary goal of system testing is to validate that the entire software application, including all its modules, functions, and interactions, is functioning correctly and meeting the intended business and technical requirements. It aims to identify defects, inconsistencies, and issues that might arise due to the interactions between different components or modules.

**Testing Methodologies**

This section outlines the various testing procedures carried out by the team in assessing the overall performance, stability and reliability of the developed Church Management System. The team conducted a systemic unit testing, system testing, integration testing, and acceptance testing.

* **Unit Testing**

Unit testing is a software testing technique that focuses on testing individual units or components of a software application in isolation. The team conducted unit testing on some functions of the system code such as member registration, donation method, and an event class.

Method: White box testing

White box testing is a form of application testing that provides the tester with complete knowledge of the application being tested, including access to source code and design documents.

The team adopted white box testing to test the various units (function, method, class) of the system because it allows testers to inspect and verify the inner workings of a software system— its code, infrastructure, and integrations with external systems.

* **System Testing**

System testing is a type of software testing that evaluates the overall functionality and performance of a complete and fully integrated software solution.

Method: Black box testing

Black-box testing is a software testing methodology that focuses on evaluating the functionality of a software application without having knowledge of its internal code and structure. The team selected black box test to validate that the software meets its specified requirements and functions as expected from a user's perspective.

* **Integration testing**

Integration testing is a software testing technique that focuses on verifying the interactions and data exchange between different components or modules of a software application.

Method: Black box testing

Selecting black box testing for integration was to identify any problems or bugs that may arise when different components are combined and interact with each other devoid of knowledge of any internal code structures by users.

* **Acceptance Testing (Alpha and Beta Tests)**

Acceptance testing, a testing technique performed to determine whether or not the software system has met the requirement specifications.

Method: Gray box and black box testing *Alpha Test:*

A few selected internal users were used for the alpha testing phase. Gray box testing was used to test the system's input and output functionality at a high level.

Gray box testing is a testing technique that combines elements of both white box testing (where the tester has access to the internal code and logic of the software) and black box testing (where the tester doesn't have access to the internal code and focuses on testing the functionality).

Using grey box is useful in uncovering defects that might not be apparent through black box testing alone, while still maintaining some level of realism since the testers are not as familiar with the internal details as the development team.

*Beta Test:*

Beta test proceeded using black box testing. The selected users tested the beta version of the software without knowledge of its underlying code. They interacted with the software as end users would, performing various tasks, using different features, and providing feedback about their experiences.

Black box was chosen to identify issues related to user interface, usability, functionality, and compatibility across different devices or environments in real world scenarios.

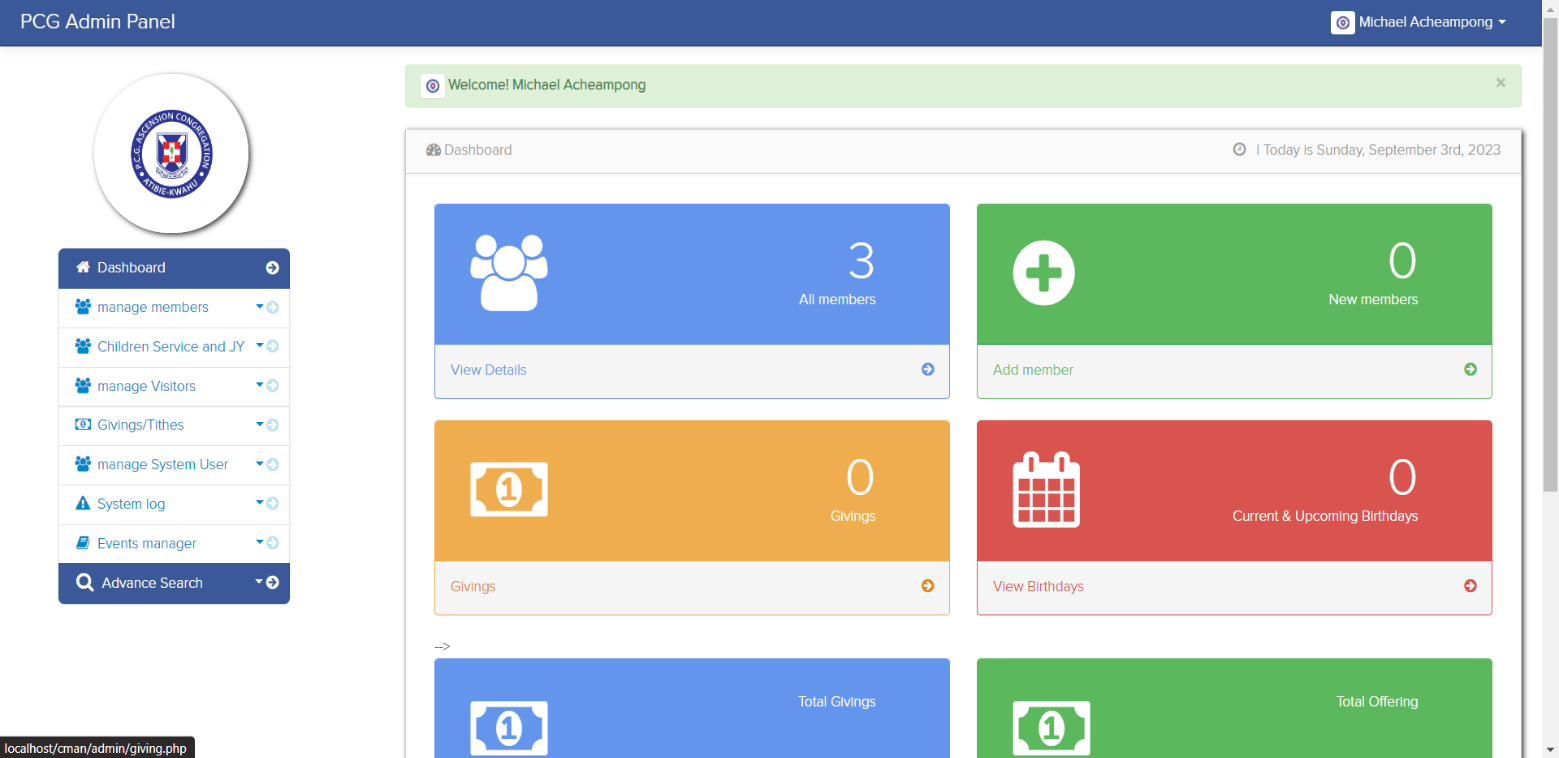
Snapshots of system operations:



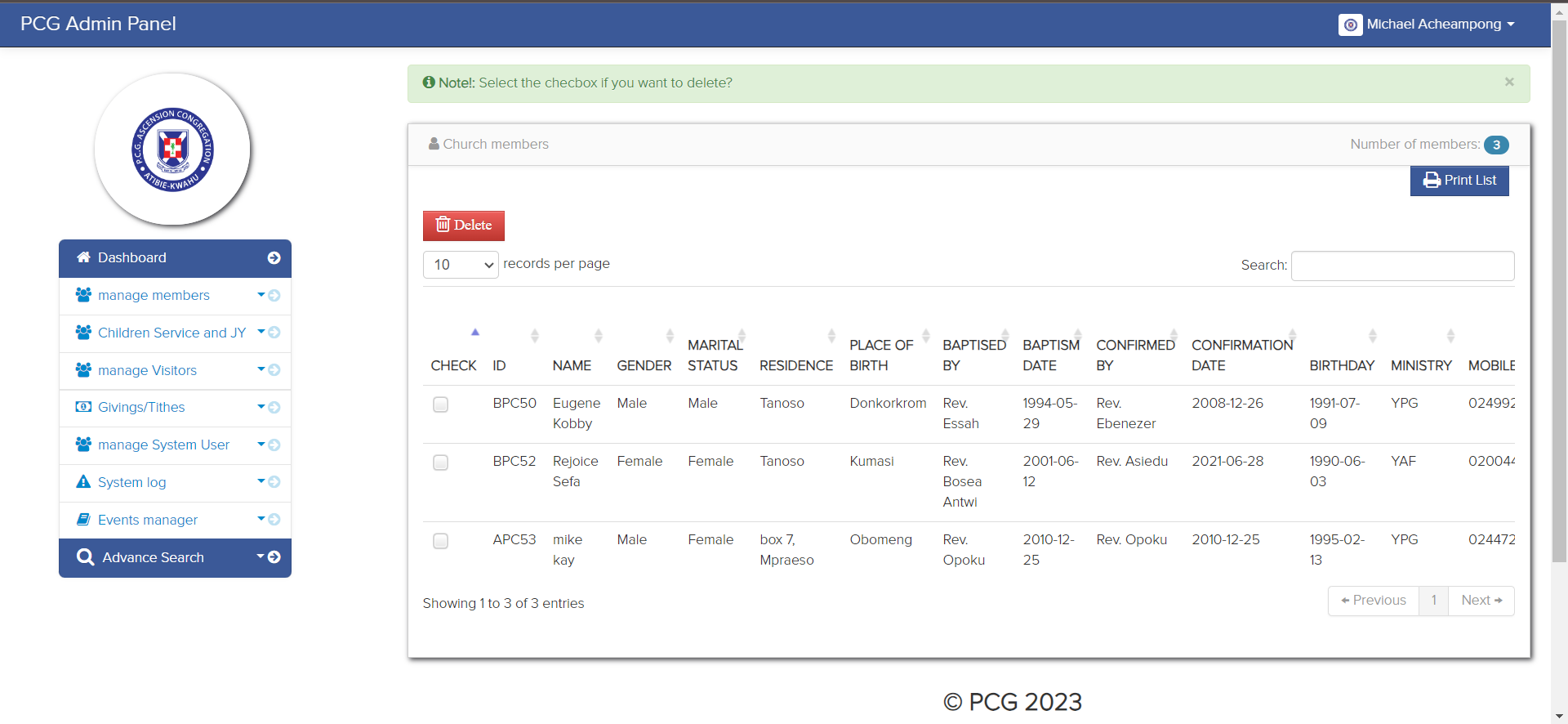
Figure 4.0: System Home page

# 

# Figure 4.1: Admin log-in

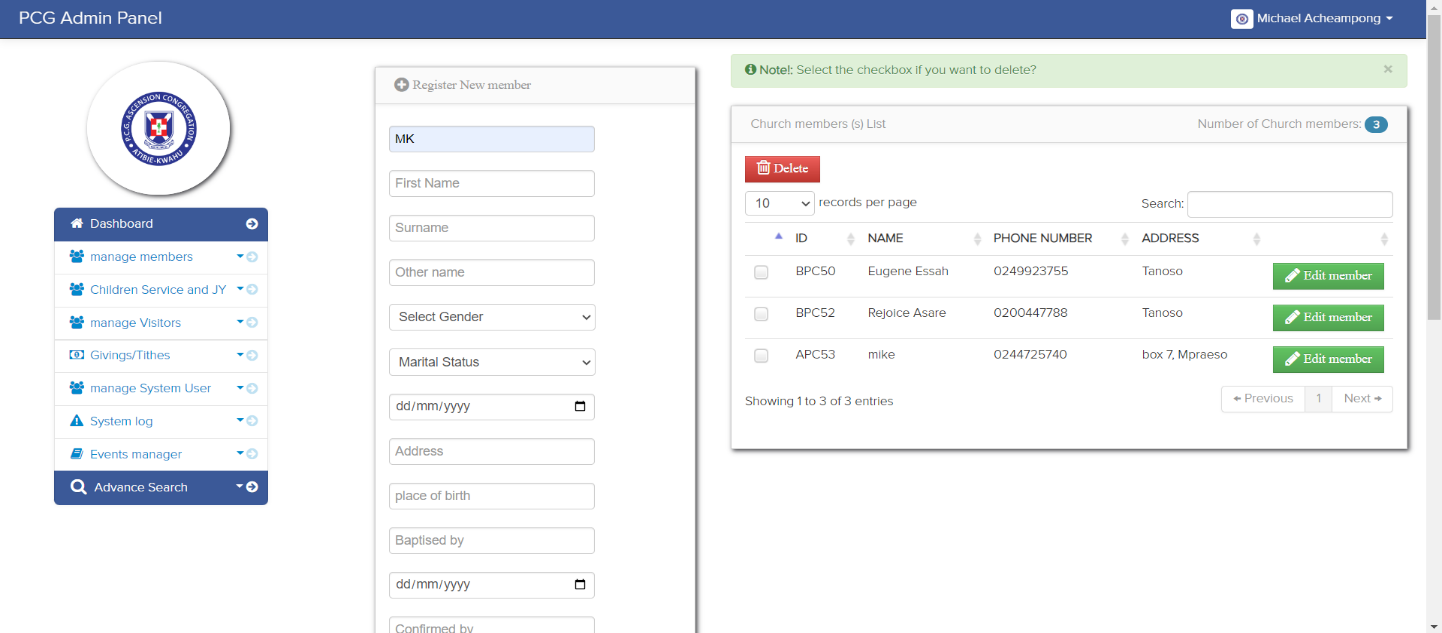


# Figure 4.2: Admin dashboard



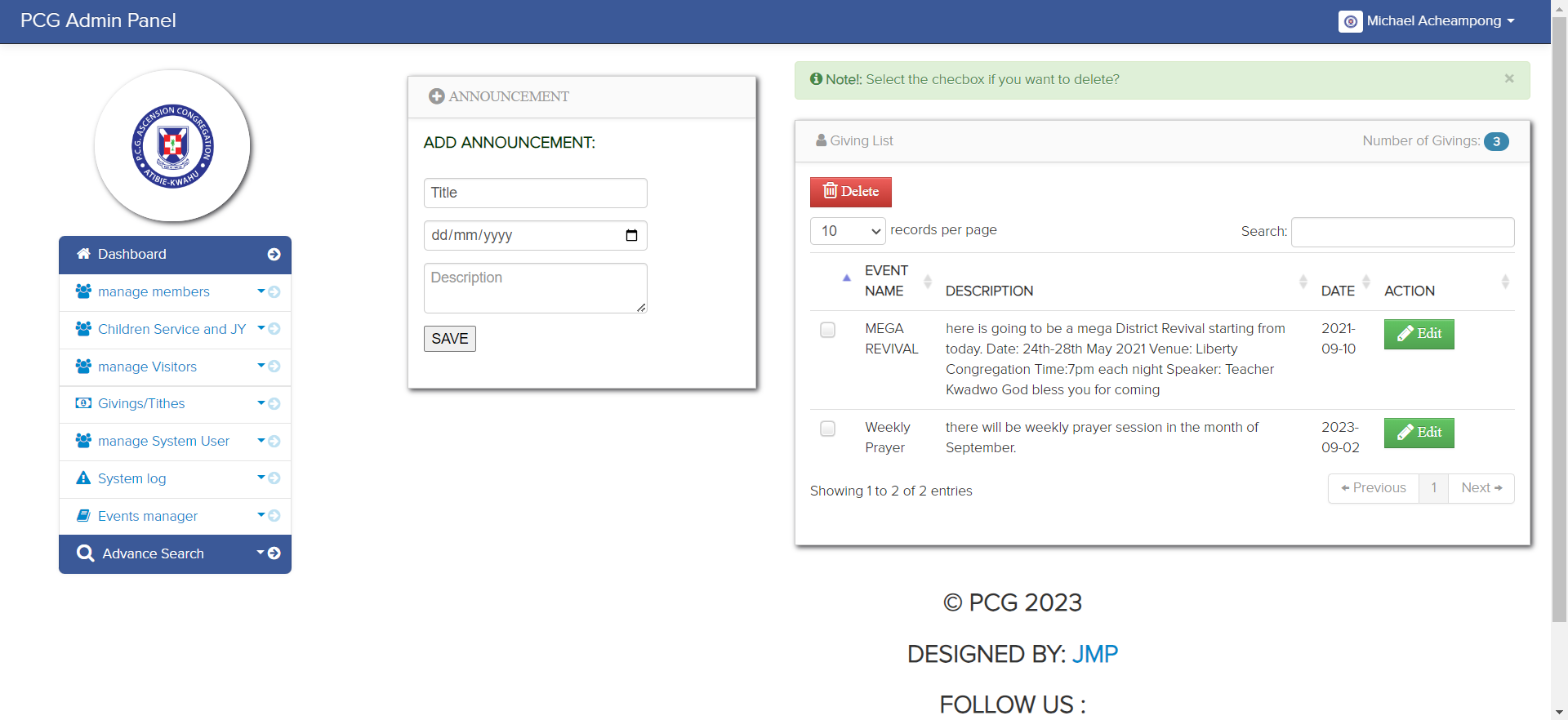
# 

# Figure 4.3: Admin member list panel



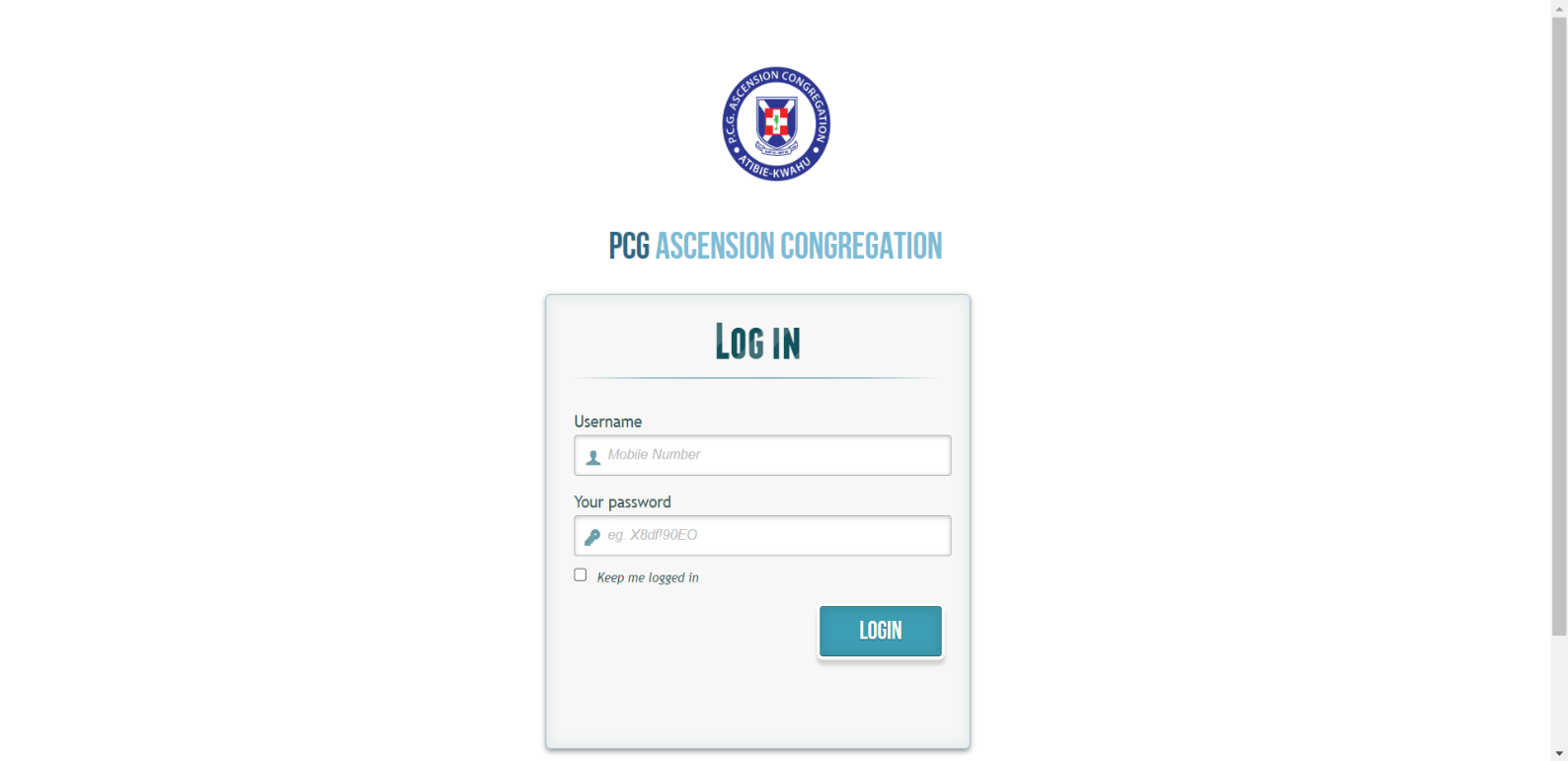
# 

# Figure 4.4: Admin registration panel

****

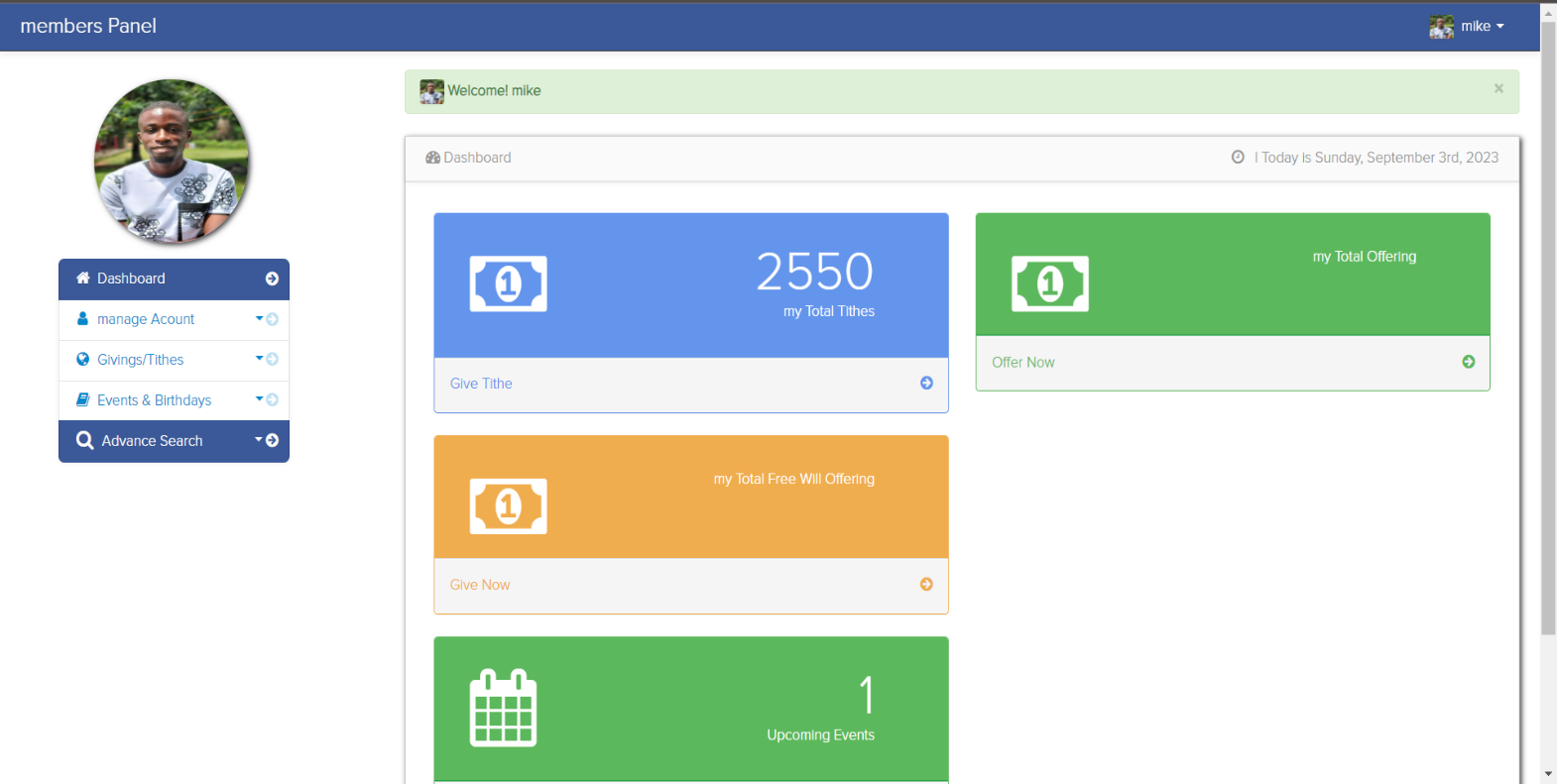
# 

# Figure 4.5: Admin events manager panel



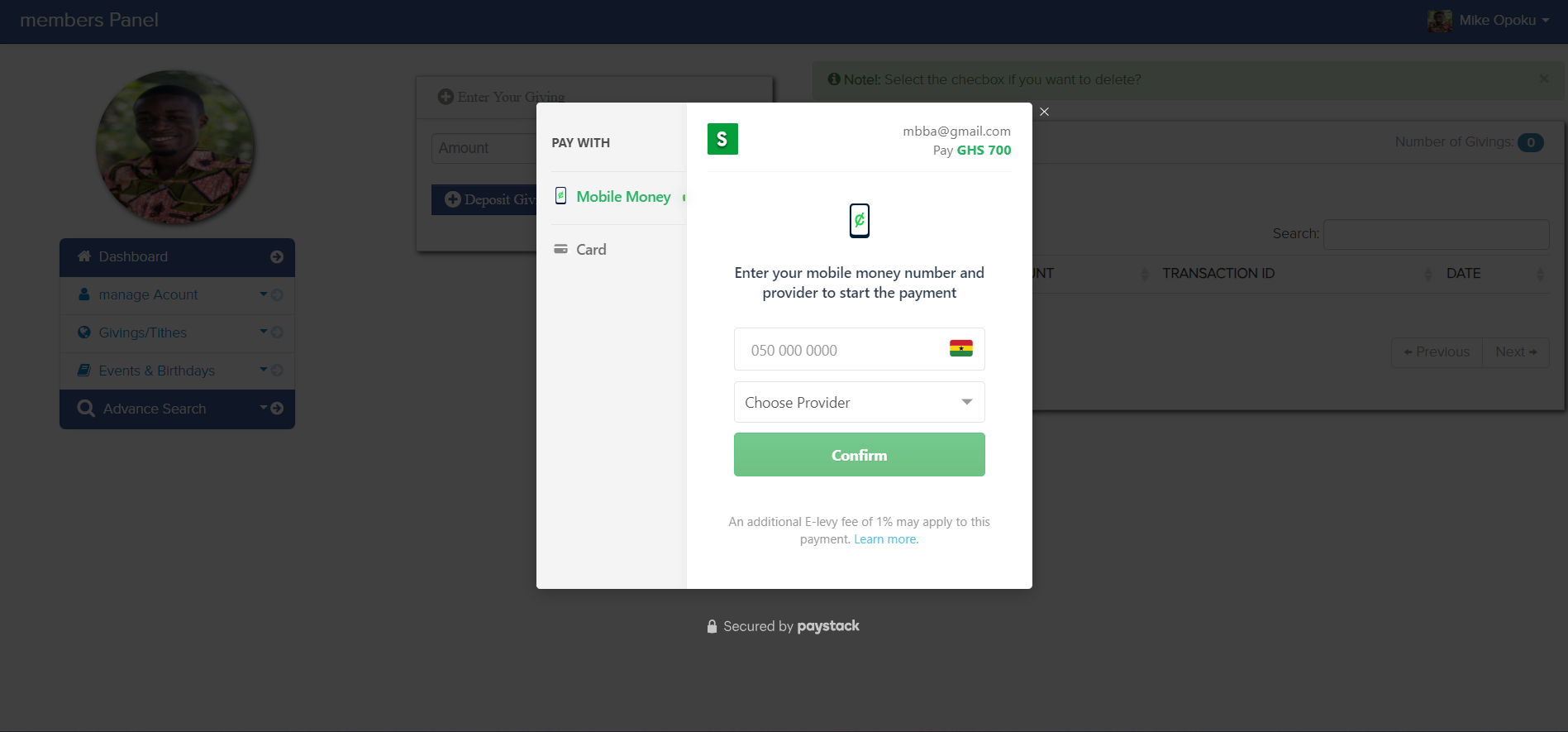
# 

# Figure 4.6: Member log in page

****

# 

# Figure 4.7: Member dashboard



# Figure 4.8: member tithe/giving panel

**4.3.1 Functional Evaluation**

|  |  |  |
| --- | --- | --- |
| **User Requirement** | **System Functionality** | **Remarks** |
| To register member. | Registration of members. | Members successfully registered. |
| To post church events/announcements and notify users. | Posts Events/announcements with notification. | Events/announcements posted  successfully and users notified. |
| To update member records. | Update of member records | Records updated successfully. |
| Track financial payments. | Track and update tithe,  donation, offering payments. | Tracking financial payments  successful. |
| Online payment feature. | Online payment feature for tithes, donations and offerings. | Online payment feature successfully deployed. |

Table 4.1: Functional evaluation

**4.3.2 Non-functional Evaluation**

The system’s non-functional features were evaluated to ensure the system meets quality standard criterion. These non-functional aspects include system security checks, performance, scalability, responsiveness and usability.

## **4.4 Summary**

A detailed report on the system demonstration and evaluation were given in this chapter. The chapter portrays how the team conducted system demonstration to show how the system works. Experimental setup, platform compatibility, simulation, and all the extensive testing phases for the Church Management System have been fully outlined.

**CHAPTER FIVE**

**SUMMARY, CONCLUSION AND RECOMMENDATIONS**

## **5.1 Introduction**

This final chapter gives summary of what the Church Management System is about, conclusion on the project indicating the level of achievement of stated objectives, and recommendations on improving the system’s functionality and overall performance going forward.

## **5.2 Summary**

The Church Management System is an online web-based system developed basically to automate the administrative functions for the Presbyterian Church Ghana – Ascension Congregation at Kwahu-Atibie. The system is designed to handle tasks including registration, update member and administrative records, serve as a tool to communicate activities to members as well as tracking financial commitments of congregants signed on to the system. The overall aim was to fully switch from the manual paper-based system of performing administrative tasks to an automated, digitalized, centralized and secure system to ensure data integrity and security, reduce and or eliminate data redundancy and duplicates. The system was tailored to meet the specific requirements agreed on by the users and the team. The system underwent series of testing during its development and implementation to ascertain its performance in meeting the intended objectives.

## **5.3 Conclusion**

The underlying objectives set for the development and implementation of this Church Management System for the target organization were achieved by the team. The challenges that prevailed hitherto in the manual system have been addressed after its implementation.

Specifically, the following are evidence of the achieved functionalities of the CMS:

1. Member management: this system module ensures seamless registration of new members, editing, and deleting member records by the system admin.
2. Communication: the communication module of the system delivers information on events, announcements, and other church related activities to users on the system via the notification feature.
3. Financial fulfilment: the financial module of the system allows fulfilment of financial obligations by members in terms of tithe, donation and offering payments and keep financial records updated.

## **5.4 Recommendations**

The Church Management System is recommended for adoption and implementation in various churches to enhance effectiveness and efficiency in their administrative activities.

Despite meeting the objectives for which this system was developed, it is worth noting that there are other functionalities to be added to further enhance its capacity and operationalization. The following are some potential features that can be incorporated into the system:

1. An integrated and secured online payment platform with verification which can also generate financial reports.
2. Deployment of system to a cloud server for scalability and enhanced data security as demand increases. iii. Development of a mobile app for the system to provide users on-the-go experience on their mobile devices.
3. An attendance register feature using fingerprint to record member attendance at church gatherings.
4. A repository to store and categorize recorded sermons, making them easily accessible to members.

## **5.5 Summary**

The chapter captured a summary on the development and implementation of the Church Management System, conclusion depicting its key functionalities and evidence on the achievement of stated objectives. The chapter finally ends with a recommendation to potential bodies that might need this system together with additional features that will enhance the system and its operations.

### REFERENCES

Ngozi B., Ossai U. (2021*). The use of information and communication technologies in Nigerian*

*Baptist churches. International Journal of Science and Technology Education Research Vol*. 2(3), pp. 49 – 57 https://academicjournals.org/journal/IJSTER

Kathambi, F. (2020). Counselor: *Digital age poses pastoral challenges to church.* Internet material from http://www.abpnews.com/content/view/3602/53/

Ajzen, I., & Fishbein, M. (1975). *A Bayesian analysis of attribution processes.* Psychological bulletin, 82 (2), 261.

Lee, Younghwa; Kozar, Kenneth A.; and Larsen, Kai R.T. (2003) *"The Technology*

*Acceptance Model: Past, Present, and Future, Communications of the Association for*

*Information Systems*: Vol. 12, Article 50. Available at: http://aisel.aisnet.org/cais/vol12/iss1/50

Bandura, A. (1982). Social Foundations of Thought and Action: A Social-Cognitive Theory

Marikyan, D. & Papagiannidis, S. (2023) *Technology Acceptance Model:* A review. In S. Papagiannidis (Ed), TheoryHub Book. Available at http://open.ncl.ac.uk / ISBN: 9781739604400

Venkatesh, V., Thong, J. Y. L., &amp; Xu, X. (2012). *Consumer acceptance and use of information technology: Extending the Unified Theory of Acceptance and Use of Technology*, MIS Quarterly, 36(1), 157–178.

Indeed Editorial Team, (2023). *What is an Information System? (With Features).* Sourced from https://ca.indeed.com/career-advice/career-development/information-management-system

Allan, J. (2023). *What is Church Management Software? Analysis of Features, Types, Benefits and Pricing.* https://financesonline.com/church-management-software-analysis-features-types-benefitspricing/#benefits

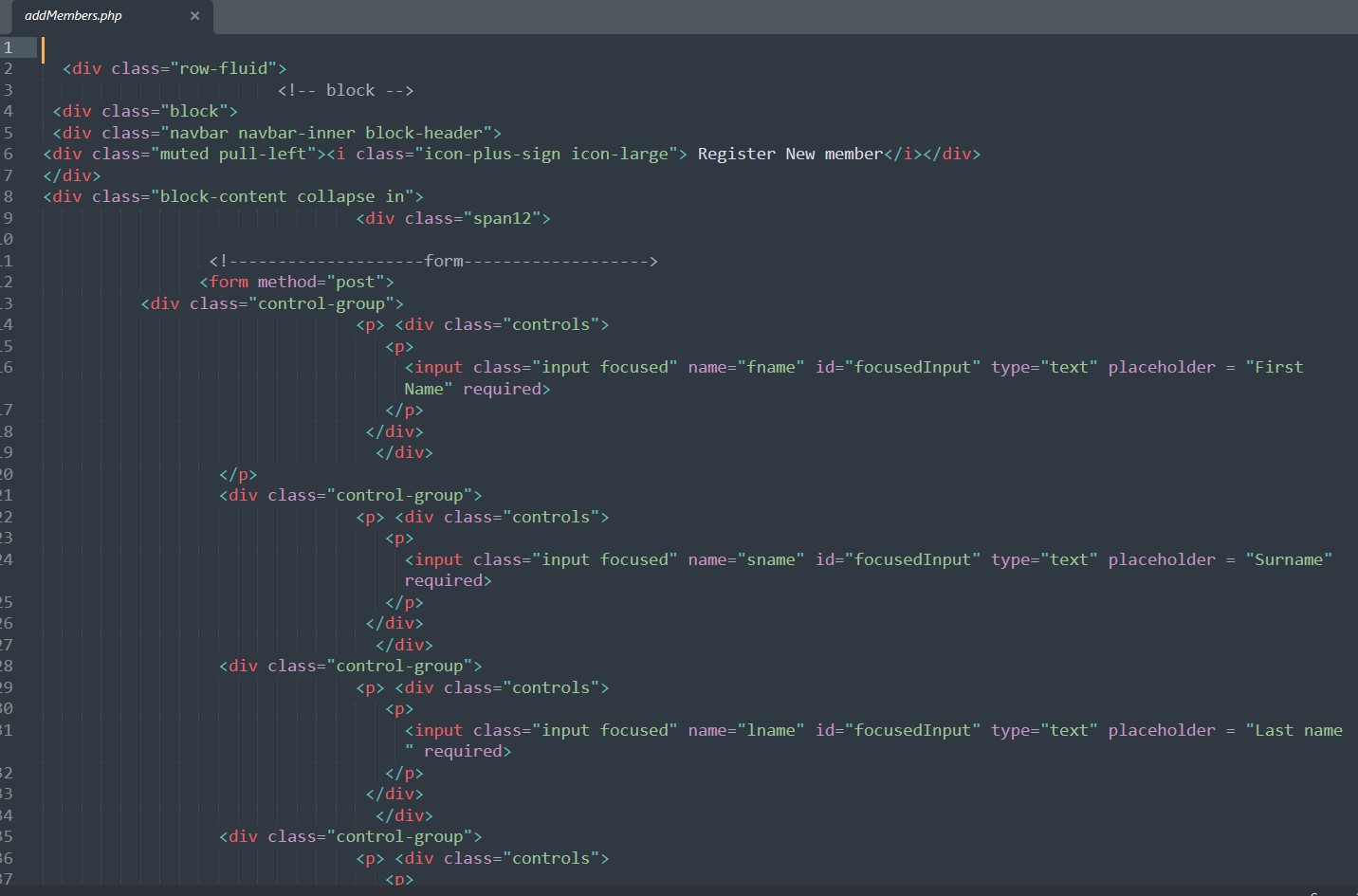
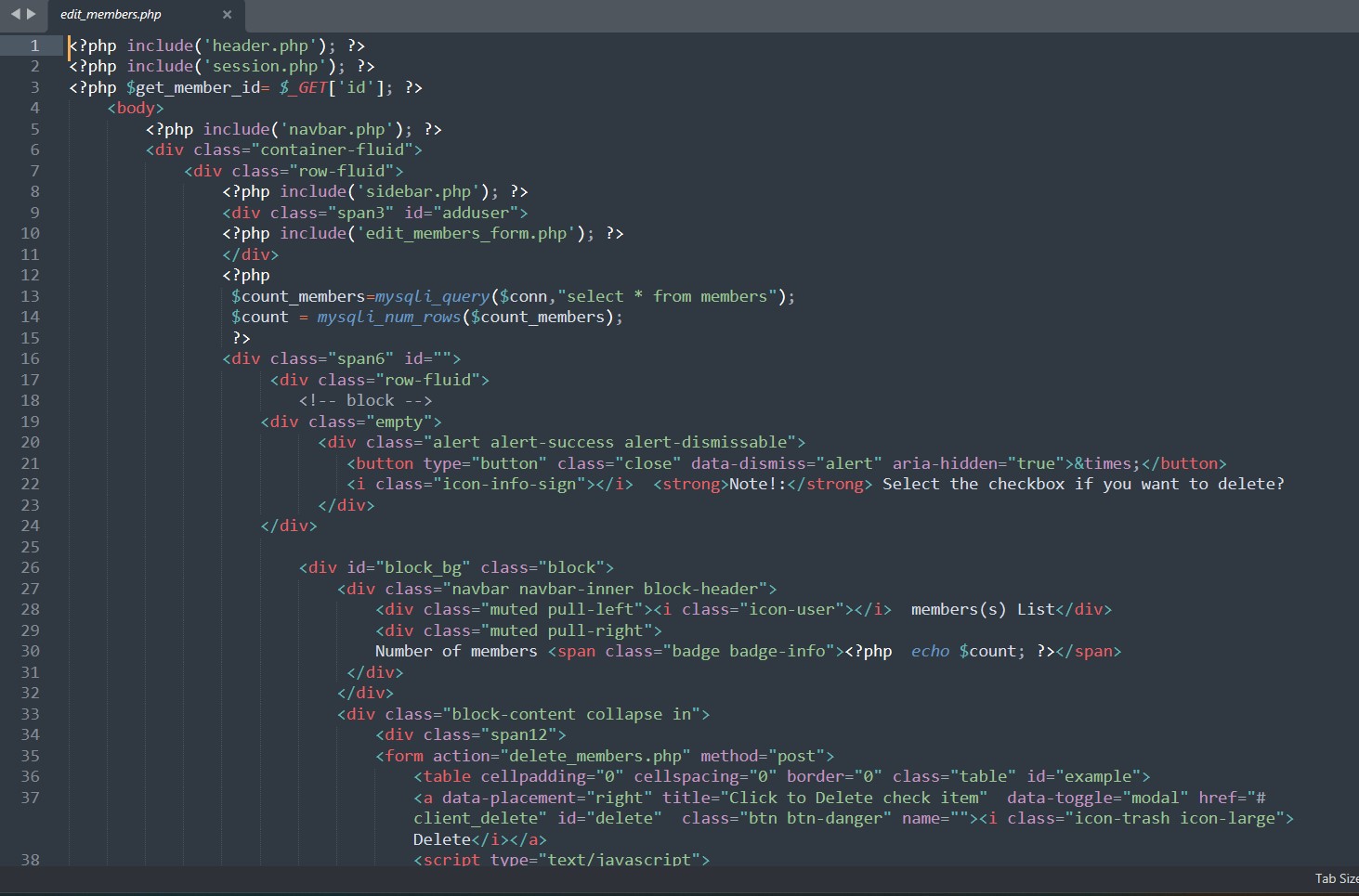
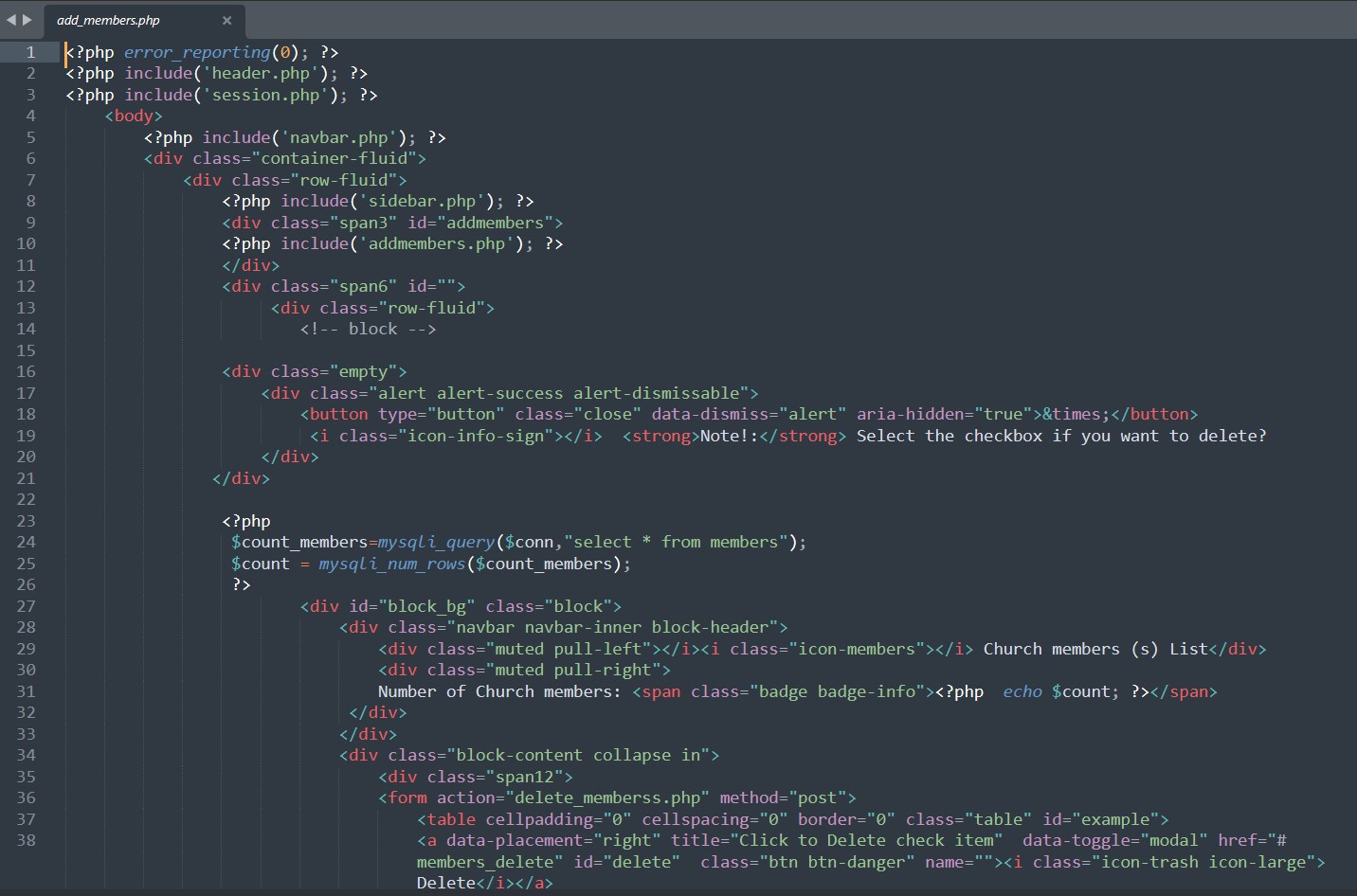
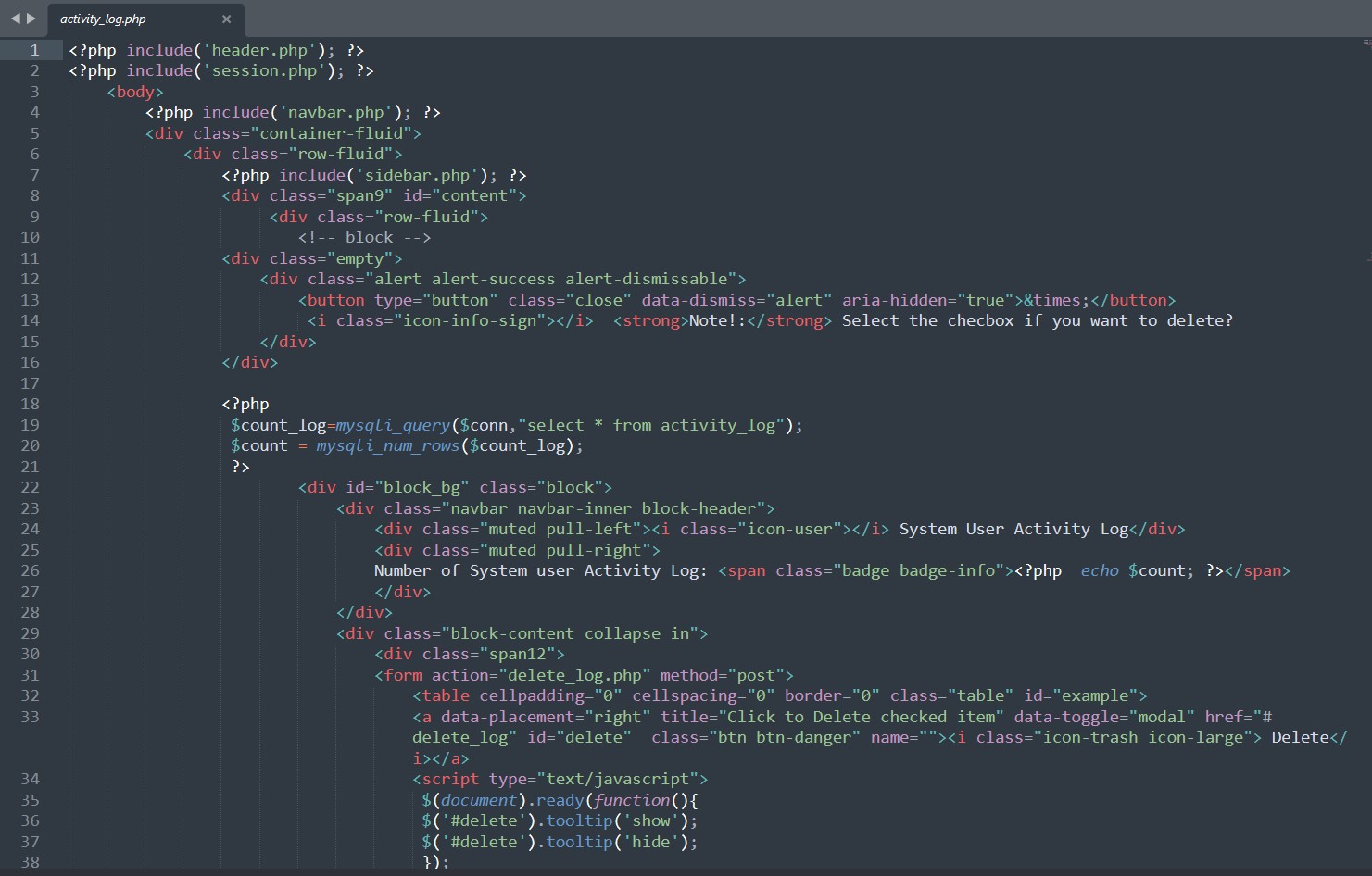
Bolanke F, (2022). *Church Sustainability: Exploring Online Technologies through the Lens of*

*System Dynamics,* retrieved from

[https://www.researchgate.net/publication/362536821\_Church\_Sustainability\_Exploring\_Onli ne\_Technologies\_through\_the\_Lens\_of\_System\_Dynamics](https://www.researchgate.net/publication/362536821_Church_Sustainability_Exploring_Online_Technologies_through_the_Lens_of_System_Dynamics)

### APPENDIX

**Sample codes**



**MySQL sample code**

