A Web-based Responsive Sacred Heart Parish Church Management System: An Integrated Operations Utilizing Constraint Satisfaction Algorithm and Data Visualization

Rex Baldos
BS in Information Technology
University of Mindanao
Matina, Davao City, 8000
r.baldos.526055@umindanao.edu.ph

Johnritz Lirazan
BS in Information Technology
University of Mindanao
Matina, Davao City, 8000
j.lirazan.531490@umindanao.edu.ph

Bostjan Zymmer Rogero
BS in Information Technology
University of Mindanao
Matina, Davao City, 8000
b.rogero.527335@umindanao.edu.ph

Somerset Elcid R. Siang, MIS
University of Mindanao
Matina, Davao City, 8000
somersetelcid_siang@umindanao.edu.ph

Categories and Subject Descriptors
CCS → Human-centered computing → Human-computer interaction (HCI) → Interaction paradigms

General Terms
Design, Management, Algorithms, Documentation, and Reliability

Keywords
Integrated Operations; Digital Transformation; Administrative Efficiency; Event Coordination

1. INTRODUCTION

1.1 Project Context

Nowadays technology arrives in all aspects of human lives even in transactions, medicine, education, and business. Computer-based systems are more applicable than traditional processes or paper-based systems because they are more efficient and consistent. Keeping records is important in a variety of organizations. Adopting automation in church operations can revolutionize the way churches manage their operations, making processes more efficient tasks such as membership management, communication, event scheduling, and sacrament requests. A church management System is used to provide the facility to automate the official functionalities of a church. The system also uses technologies to track, organize, automate, and synchronize the interaction process. The system enables users from across the region to utilize church facilities and ensure easy functioning. Automating church operations is a step toward a more efficient and effective ministry [1]. Phillips emphasizes another critical advantage of integrating technology into church activities: expanding reach [2].

The research will be conducted at the Sacred Heart Parish Church in Sacred Heart of Jesus Parish, Dahican, Mati City. The church is the target beneficiary, and the system is designed to address this community's specific needs and challenges. Sacred Heart Parish Church serves many members who require efficient handling of sacrament requests and event coordination for church events.

In the validation phase, the parish priest of Sacred Heart Parish Church stated that handling an average of 20-30 requests per day (including baptisms, weddings, confirmations, first communions, anointing of the sick, funerals, and burials, and blessings) with paper consent forms is quite challenging. The church staff responsible for these requests must manually review and approve the paper consent forms, leading to slow response times. According to a study by Tech in Churches, challenges and opportunities experienced by church staff in making them shift from traditional to online church [3]. Moreover, reports from the Unitarian Universalist Association mention that church records are so important. Church or Parishioners members generally agree that saving church records is a good thing [4]. By implementing automated systems, churches can streamline data collection, storage, and retrieval, improving efficiency and reducing errors.

According to the parish priest and church staff, several challenges were identified in scheduling sacrament events: parishioners often requested scheduling without having all the requirements ready. With limited priests in the church, certain sacraments had to be scheduled around specific dates. Also, handling schedules and manually keeping track of requirements made scheduling more complicated.

The current scheduling procedures at the parish are as follows: Confirmation schedules are arranged by the parish, particularly during the parish's feast day. The parish arranges the schedule for baptisms and weddings, but parishioners can request a specific date for an additional fee. The scheduling for the anointing of the sick and burials is flexible and based on the needs of the lay faithful. Before scheduling weddings, confirmations, and baptisms, parishioners must attend a seminar and complete all required documents. Blessings are typically conducted after Mass.

Scheduling algorithms are crucial to effectively and efficiently managing limited resources [5]. Autoscaling is an essential feature in Kubernetes scheduling because it automatically adjusts the resources allocated to pods based on current demand. This capability ensures efficient resource utilization, improved

performance, cost savings, and high availability of applications. Auto-rescaling and scheduling are related in that auto-rescaling can ensure that there are always enough resources available to handle the scheduled tasks [6]. The advantage of formulating the problem as a Constraint Satisfaction Algorithm (CSA) due to the commonly disparate structure of the problem is the ability to work out any problem as a mathematical object and to solve any problem algorithmically. In this way, a purely theoretical solution to the problem becomes applicable to almost anything. We can understand CSA as trying to find such a solution from the information we have about the problem so that all the conditions (constraints) of the problem are met [7]. A (finite domain) constraint satisfaction problem (CSA) can be expressed in the following Form. Given a set of variables, together with a finite set of possible values that can be assigned to each variable and a list of constraints, find values of the variables that satisfy every constraint. Many problems in operational research (OR) fall within this general framework. For example, in a timetabling problem where examinations must be assigned to various periods, constraints such as ensuring no student has overlapping exams and accommodating room sizes must be met. Additionally, production scheduling problems require that jobs processed on machines meet specific deadlines without overlap. Optimization problems can also be expressed as a sequence of CSA by setting a threshold value on the objective function value and adjusting it to find the optimal solution [8].

It can also be difficult for church staff to keep track of all the sacrament requests, including pending, approved, or completed. Without a clear visual representation of the data, the church staff of Sacred Heart Parish struggles to identify who is pending and approved in requests. Struggles to identify the trends in requests for which they can't allocate resources, such as priests, and facilities, properly.

The Sacred Heart Parish Church located in Sacred Heart of Jesus Parish, Dahican, Mati City still uses traditional processes which is a paper-based system (paper forms) in managing sacrament requests like Baptism, Marriage, Blessing, etc; and event scheduling. Based on the survey conducted among parishioners, visiting to follow up on their requests in church multiple times is time-consuming. That's why integrating technology into their church operations is very important. The study aims to transition paper-based systems to digital processes to improve efficiency in their church operations. With technology arriving in all aspects of human lives, it's surprising that most churches still rely on traditional processes. This shows the urgent need to modernize church management systems.

With the recent increase in technological advancements, the church has an opportunity to embrace technology for worship, reaching out to the world, and office work [9]. As churches grow, the need for efficient organization and data management becomes increasingly crucial. Church management systems (CMS) address this need by providing a centralized platform for various church operations [10]. Implementing data storage in Database Management Systems (DBMS) within a CMS offers several advantages. Oluwade *et al.* highlights eliminating data redundancy, ensuring consistent and accurate information [10].

Furthermore, a centralized data repository allows for easy access and retrieval, streamlining various processes. Additionally, church management systems can be equipped with integrated security measures to safeguard sensitive member data and control access levels. The ability for parishioners to view information and submit requests online through a web-based CMS represents a significant improvement. This functionality simplifies registration, reduces administrative burdens on church officials, and empowers members with more excellent self-service options [10]. Sabuncuoglu et al. concluded that online scheduling based on the assignment rule is more substantial than offline scheduling [11].

The parish priest mentioned that as a church leader, they believe that implementing technology in church operations can enhance management and make it easier to serve the needs of all individuals requiring the church's services. Technology facilitates improved information sharing and communication. Churches notify members about forthcoming events, sermons, and other significant announcements using websites, email newsletters, and social media [12]. Customers are well-informed and plan their attendance correctly [12]. Incorporating technology into conventional church services improves the audiovisual experience, expedites administrative work, encourages community involvement, and improves communication, all of which contribute to a more fulfilling and inclusive worship experience. Technology encourages communication and participation from the community during services. Congregants use interactive apps or digital platforms to submit prayer requests, participate in live polls or surveys, and engage in virtual debates, all of which contribute to a stronger feeling of community and active service involvement [12].

The researchers evaluated existing church management systems to understand better the unique features of the Sacred Heart Parish Church Management System.

Table 1. Comparison

Features	CC-IMS	Excellerate	ParishSOFT
Sacrament Request			✓
Feedback Mechanisms		✓	
Document Submissions			
Data Encryption			
Scheduling		✓	✓
Data Visualization	✓		

The researchers evaluated the previous study the CC-IMS or Immaculate Conception Parish og Banganga which focuses on descriptive analytics and clustering algorithms which is different from the proposed study. The Excellerate is a Church Management Software that tracks all of the scattered details in one central system it automates visitor follow-up letters/emails/texts [13]. The ParishSOFT is a software that helps parishes of all sizes reduce their administrative burdens and streamline productivity. The software includes family and member records, a parishioner portal, online giving, and volunteer management [14].

1.2 PURPOSE AND DESCRIPTION

This study targets having a single platform to be used for a number of events in the church for operational efficiency. This move will foster interaction between the members and church staff more

conveniently and efficiently. By automating tasks such as these, processing time can be minimized, delays prevented and waiting periods decreased. Parishioners can send their forms online rather than visiting the church every other day for the same process. To manage various sacraments and church events, the system will incorporate a scheduling feature. This feature aims at making the scheduling process simpler for all sacraments while accommodating preferred dates for certain sacraments thereby ensuring that it operates in an organized and efficient manner. Implementing this system has immense global implications and aligns with the first CCE (College of Computing Education) research agenda which focuses on Technology & Information Management in the Environment to improve Sacred Heart Parish Church's operations [15]. Additionally, through an interface that simplifies jobs like monitoring resources or managing funds, the system makes administrative duties easy for staff working at churches to carry out.

The Sacred Heart Parish Church Management System project includes developing sacrament request forms, document submissions, event scheduling, securing data, notifications, profile management, calendar viewing, and data visualization. This system automates administrative tasks for church staff, providing an easy way to monitor and manage resources. It ensures timely and proper records processing, keeps information secure, and makes data retrieval easy. This database allows parishioners to ensure efficient logistical support during major Christian event planning processes. This study addresses the church's challenges by improving scheduling capabilities and creating community engagement, leading to improved customer satisfaction through well-timed services.

1.3 OBJECTIVES

1.3.1 General Objectives

The researchers aim to develop the Sacred Heart Parish Church Management System: Integrated Operations Utilizing Constraint Satisfaction Algorithm and Data Visualization.

1.3.2 Specific Objectives

To attain the general objectives, the researchers will develop a web-based application to specifically:

1.3.1.1 To develop an online sacrament request feature allowing users to request sacraments (baptism, wedding, confirmation, first communion, anointing of the sick, funerals and burials, and blessings) and submit the required documents electronically, using HTML, MySQL, CSS, JavaScript, and PHP to streamline the request process and ensure efficient handling of necessary documentation.

1.3.1.2 Notify the user of sacrament requests' approval and status updates using WebSockets to ensure clear communication and efficient request processing.

1.3.1.3 To facilitate scheduling using a Constraint Satisfaction Algorithm for church events.

1.3.1.4 To integrate data visualizations using Chart.js, providing clear insights into church activities and sacrament request statistics, helping in informed decision-making.

1.3.1.5 To secure user data by implementing robust security measures such as encryption and authentication, ensuring that personal and sensitive information is protected from unauthorized access and breaches.

1.4 SCOPE AND LIMITATIONS

This study is focused on developing and implementing the Sacred Heart Parish Church Management System. The main goal of this system is to establish an efficient operational procedure. The implementation will target parishioners and staff at the Sacred Heart Parish Church in Mati. The project is expected to be finished the project in 9 months following the process of agile. The researchers only address the problems mentioned by the church staff, priests, and parishioners.

Several constraints and limitations must be considered. The performance of the system depends on the availability and reliability of internet connectivity which may change with time to succeed in implementing this system, parishioners and staff need to be prepared to adapt themselves to new technologies, thereby requiring good training and support.

2. METHODOLOGY

The researchers will be using Agile method for this project to have a fast pace, complete output by using step by step process to have a successful development.

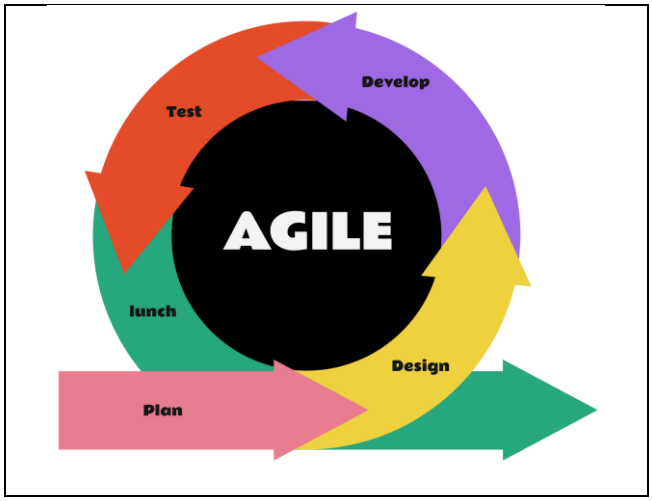

Figure 1. Agile Methodology

Phase 1. Planning: In this phase, data gathering and data analysis are included. The researchers will go to the church at Mati to conduct interviews with the priests and church staff and an online survey for the Parishioners to identify and verify the issues they face; they will also analyze the collected information from the interviews and surveys to identify critical insights. This will help the researchers understand their needs and develop solutions to address these problems effectively.

Phase 2. Design: During the design phase, the researchers will focus on creating user-friendly interfaces to make sure that the parishioners and staff find the system easy to use. Continuous feedback and prototyping will be key activities in this phase.

Phase 3. Development: During the development phase, the programmers will start to develop the system they will use the prototypes and example sacrament paper forms provided by the

church as a guide to ensure the system meets the church's specific requirements and workflows.

Phase 4. Testing: During the testing phase, the researchers will go to the church at Mati to test the developed system in their office to determine if the intended functions are correct in the system.

Phase 5. Implementation: In the final phase, the researchers will deploy the full system functionality at the Sacred Heart Parish Church in Mati. The researchers will provide a user manual and train the church staff on how to use the system properly, transferring all the data to the system, and ensuring that the system is fully operational and ready.

2.1 DATA GATHERING

The planning process started the agile methodology. In order to have a wide range of opinions, the researchers engaged in a comprehensive collection of information. The researchers interview the parish priest at Sacred Heart Parish Church for them to understand the specific needs and issues affecting their church. This enabled them to have more definitive answers than simply making use of a survey and it gave them greater insight into the issues as well as possible solutions.

The researchers also made an online survey using Google Forms. This helped to gather large amount of data from many people within a short period of time. They therefore targeted individuals who had enough knowledge about the church and were actively involved. It asked questions about existing challenges that face staff and members of churches and their views on advantages associated with application of church management systems. By using Google forms, they could organize and look at what they found out. By conducting interviews in addition to surveys, these researchers obtained an excellent understanding of what was expected from such a system.

2.2 ANALYSIS

This part will now focus on systematic analysis required in development of church management system following agile methodology. This involves evaluating the tools needed for system development and identifying functional and non-functional requirements. The analysis will facilitate the creation of a system that aligns with user expectations and operational demands.

2.2.1 User Requirements Definition

The researchers specify the user's requirements that are considered necessary for continuous use of the online portal:

2.2.1.1 Users will create an account on the web portal to access online sacrament request forms and fill in the necessary details.

2.2.1.2 Users will submit necessary documents based on the type of event they are requesting.

2.2.1.3 Church staff can oversee and approve sacrament requests. Users will be notified once their requests are approved.

2.2.1.4 Users can update their information, view event calendars, and check recent requests.

2.2.2 Hardware and Software Requirements

The researcher requires the following hardware specifications and software requirements for the system.

Tables 2 and 3 show the system's minimum hardware and software requirements for installation and operation. They also show the recommended hardware specifications for running the application smoothly, especially for the processor, RAM, storage, screen resolution, and internet connection.

Table 2. Hardware Requirements

Hardware	Requirements
RAM	8GB (minimum) or 16GB (recommended
CPU	Corei5 10th Gen or Higher
RAM	4 GB
Internet/ Data	10-15 Mbps or higher
Storage	SSD 500GB or higher

Table 3. Software Requirements

Software	Requirements
OS	Windows 10 or higher
MySQL	Used for managing the database, it is responsible for storing, retrieving, and organizing all the data related to the church management system.
Visual Studio Code	Used for coding editor programming both front end and back end
Web Browser (Google Chrome, MS Edge, Firefox etc.)	Used for running the application and testing during the development and after implementation

2.2.3 Technical Tools

2.2.3.1 *Visual Studio Code*

Visual Studio Code (VS Code) is an integrated development environment (IDE) created by Microsoft. Its lightweight and profoundly customizable nature makes VS Code an amazing choice for coding and investigating the Sacred Heart Parish Church Management System. The IDE offers strong highlights such as IntelliSense for brilliant code completion, coordinates Git for adaptation control, and a wide determination of expansions, which streamline the advancement handle. Moreover, VS Code's compatibility with numerous stages (Windows, macOS, and Linux) advances adaptability and cultivates collaboration among designers working on different working frameworks.

2.2.3.2 *MySQL Database*

MySQL could be a capable social database administration framework utilized to oversee and store organized information for the church administration framework. Its Unwavering quality and tall execution make it appropriate for dealing with complex information and overseeing parishioner records, holy observance plans, and occasion subtle elements. MySQL's bolster for SQL (Structured Query Language) empowers effective questioning and

information control, guaranteeing that the framework can effectively handle expansive volumes of information. Its open-source nature too permits for taken a toll successful improvement and sending.

2.2.3.3 *Chart.js*

Chart.js is a JavaScript library for making intelligently and responsive charts inside the church administration framework. It gives different chart sorts, such as bar, line, pie, and radar charts, permitting viable information visualization of church exercises and part measurements. Chart.js is known for its straightforwardness and ease of integration. It guarantees the advancement group can rapidly execute energetic charts to upgrade the client interface and give clear, visual bits of knowledge into the church's information.

2.2.3.4 *PHP*

PHP is a server-side scripting dialect utilized in creating the Sacred Heart Parish Church Management System. Its capacity to be implanted inside HTML and its compatibility with different databases make PHP a basic instrument for making energetic web pages and applications. PHP encourages the advancement of intuitive highlights such as online holy observance demands, profile administration, and record entries. Its broad library back and ease of utilize empower the effective and successful advancement of a feature-rich church administration framework.

2.2.3.5 *JavaScript*

Javascript is utilized within the church management system to form energetic and intuitive web substance. This programming language is fundamental for client-side scripting, empowering the control of web page components, dealing with client interactions, validating input information, and asynchronously communicating with servers. Within the church management system setting, JavaScript upgrades client involvement by empowering highlights such as frame entries, genuine time upgrades, and intuitive calendars. Its flexibility and far reaching back over all major web browsers make it a basic component of cutting-edge web improvement for the framework.

2.2.5.1 Non-Functional Requirements

2.2.5.1.1 Performance

The system should have a large storage of data to handle the submitted documents.

2.2.5.1.2 Usability

The interface should be user-friendly so that any user can use the system, especially for people who may not be tech-savvy.

2.2.5.1.3 Compatibility

The system should always be accessible and available to users; it can run on any web-supported device if it has internet.

2.2.5 Feasibility Study

This section explains the project's viability testing and analysis. Researchers must evaluate and validate the application's features, usability, performance, and viability. This can only be accomplished by conducting testing and feasibility studies. Furthermore, this would be used to assess the overall efficiency of the application.

2.2.5.1 *Technical Feasibility*

The researchers ensured that they had the necessary tools to develop the system such as laptops, computers, and the internet. They also ensure that they have skills in specific programming languages such as PHP, JavaScript, and relevant web development. The church also has the necessary tools, in their office they have computers and an internet. Overall, all the necessary tools and skills are there to develop the system.

2.2.5.2 *Operational Feasibility*

The study shows that the system will improve the church's operations by transitioning paper-based system to a digital process. According to the parish priest, he believes that implementing technology in church operations can enhance management and make it easier to serve the needs of all individuals requiring the church's services. The operational advantages of the system include improved coordination of church activities, streamlined sacrament requests, and timely updates and notifications to members.

2.2.5.3 *Schedule Feasibility*

The researchers have planned the project timeline to ensure each phase is completed on time. The estimated duration of the project is from June to December. The planning phase is scheduled for June when the researcher will gather data from Parishioners and church staff about their encountered problems. In the design phase, also in June after the planning, the researchers will make a user-friendly interface using Figma prototypes. The development phase will be from June to September when one of the research programmers will start to develop or code the system. Testing is planned for September to October, the research will test it in the office of the church if there will be a revision of the system functionality. For full system functionality, the researchers must finalize the system it should be a full-blown system. The implementation phase is in December where the researchers will deploy the full-blown system in the Sacred Heart parish church in Mati.

Table 4. Project Duration

Project Phase	2024						
	Jun	Jul	Aug	Sept	Oct	Nov	Dec
Planning							
Design							
Develop							
Test							
Full System							
Implement							

2.3 SYSTEM DESIGN

2.3.1 Conceptual Design

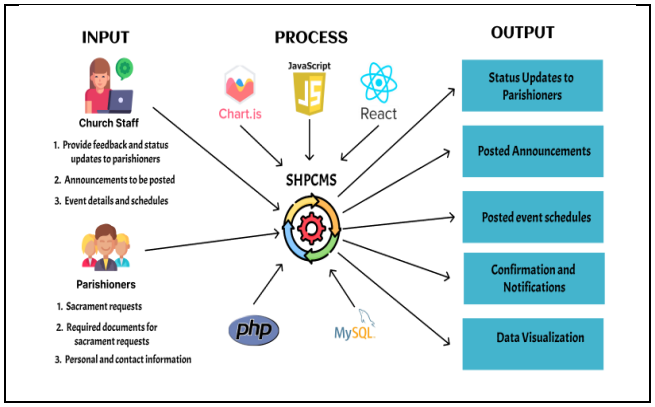

Figure 2. Input-Process-Output

2.3.1.1 *Input*

The Church staff are the admin users of the system who will manage the sacrament request. The system will allow the admin to provide feedback about whether their request is correct or incorrect the system automatically updates the status of the request pending, approved, and rejected.

The Parishioners are the system users who can request sacraments through online forms they are required to input personal details and attach any necessary documents, such as identification, birth certificates, baptismal certificates, confirmation certificates, and others based on the sacrament they request.

2.3.1.2 *Process*

The system will allow the admin to review the submitted requests by the users. In scheduling the events in the church, the system will utilize Constraint Satisfaction Algorithm (CSA). Finding a solution that meets a set of constraints is the goal of the constraint satisfaction Algorithm (CSA) [15]. By defining variables (such as dates and times for various sacraments) Variables in a CSA are the objects that must have values assigned to them to satisfy a particular set of constraints [16]. Domains (possible dates and times) The range of potential values a variable can have is represented by domains [16]. The constraints are (specific rules for scheduling) Constraints: The guidelines that control how variables relate to one another are known as constraints. Constraints in a CSA define the ranges of possible values for variables [16].

To secure the user's data, data encryption is implemented this is to protect the submitted requests by the users which include information and documents stored in the database. Also, Access control is implemented in the system to prevent authorized users view different sensitive information in the system.

2.3.1.3 *Output*

The users will be notified about the status of their request, whether it is approved, if there are any necessary documents needed to resubmit, or if it is rejected. Notifications will pop up on the web interface using WebSocket technology.

It will also have a data visualization component in the system that helps give insight into the activities of the church and the sacrament requests. Graphical representations may be in charts and graphs representing trends of the sacrament requests and upcoming events. Dashboards will be provided for staff to view summarized data and detailed reports who are the users who are pending approved, and rejected.

The constraint satisfaction algorithm for scheduling is hence conflict-free and transparent for all activities and sacraments. This calendar satisfies all the rules and needs, like the dates available and prerequisites required, avoiding double-booking or overlapping of events. The plan is open to both the admin and the users. It ensures everything is organized and ready in advance. Constraint Satisfaction Algorithm scheduling a baptism defines variables: baptism date, time slot, prerequisites—required documents, like birth certificates, and seminars that are to be conducted. Then it specifies the domains for these variables, listing possible dates and available time slots, such as August 10 and August 17, or 10:00 AM and 2:00 PM, respectively. These are the constraints applied to ensure that the date and time picked are not in conflict with other church activities like weddings or funerals. It checks to see if parishioners have completed all the necessary documents and seminars before scheduling. All documents would have been reviewed and checked on whether the stipulated constraints adhere to them so that the baptism schedule would be well organized and free of conflicts.

It keeps all documents related to the sacrament request securely in an encrypted database, with attachments of documents such as identification, birth certificates, baptismal certificates, and confirmation certificates, in addition to personal information. The system prevents users from accessing information beyond their authorized scope or other information they can only view information.

2.4 DATA MODELS

The researchers will present data models, user interaction, and flow in this section. It shows how data moves through the web application system and is saved in the database. It also includes the entire system's design, architecture, and functionality.

2.4.1 Use-Case Diagram

The use-case diagram will illustrate the roles of parishioners and church staff, highlighting the specific actions they can perform within the system.

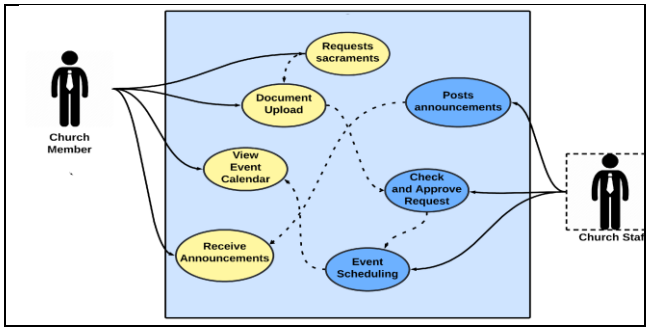

Figure 3. Use-case Diagram

2.4.2 Entity Relationship Diagram

The entity relationship diagram visually represents the connections between entities and their fields within the system database.

Figure 4. ERD Diagram

2.4.3 Data Dictionary

Table 5: Users Table: The Users table includes a unique UserID (primary key), user details, account creation timestamp, and last update timestamp.

Table 5. Users Table

Fields	Type	Description
UserID(PK)	int	Unique identifier of user
FirstName	varchar	First name of user
LastName	varchar	Last name of user
Email	varchar	Email address of user
Password	varchar	Password of user
Address	varchar	Address of the user
Created_at	Timestamp	Date and time the account was created.

Table 6: The Baptism Request table includes a unique RequesterID (primary key), details of the person to be baptized, the name of the father, mother, and godparents, status, processed date, with requester ID serving as the primary key, approved by and serviced by as the foreign keys.

Table 6. Baptismal Request

Fields	Type	Description
RequesterID(PK)	int	Unique Identifier for each baptism request
FullName	varchar	Full name of the person to be baptized
PlaceOfBirth	varchar	Place of birth of the person to be baptized
FatherName	varchar	Full name of the father of the person to be baptized
FatherBPlace	varchar	Address of the father of the person who will be baptized
MotherMNe	varchar	The maiden name of the mother of the person who will be baptized
MotherBPlace	varchar	Address of the mother of the person who will be baptized
DMarriage	varchar	Date of marriage by a priest of the parents of the person who will be baptized
MCertNo	int	Marriage certificate number of parents of the person who will be baptized
PMarriage	varchar	The church where the wedding took place
BaptismalDate	varchar	The date of the ceremony
GodParentName	varchar	Full name of the godparent
GodParentAddress	varchar	Address of the godparent
ApprovedBy(FK)	varchar	The name of the person who approved the request.
ServicedBy(FK)	varchar	Identifier of the person or entity that provided the service or handled the request

Table 7: The blessing request table includes a unique request ID (primary key), request date, preferred blessing date, blessing type, blessing place, full name, requester contact, status, and processed date, approved by and service by as the foreign keys.

Table 7. Blessing Request

Fields	Type	Description
RequestID	int	Unique identifier for each funeral and burial request
RequestDate	varchar	Date when the funeral and burial request was made
PreferredBlessingDate	varchar	Preferred date for the blessing
BlessingType	varchar	Type of blessing requested (e.g., house, car, personal)
BlessingPlace	varchar	Place where the blessing is requested
FullName	varchar	Full name of the person making the request
RequesterContact	varchar	Contact information of the requester (phone/email)
Status	varchar	Current status of the request (e.g., Pending, Approved)

ProcessedDate	varchar	Date when the request was processed
ApprovedBy(FK)	varchar	The name of the person who approved the request.
ServiceBy(FK)	varchar	Identifier of the person or entity that provided the service or handled the request

Table 8: The Anointing of the Sick Request table includes a unique request ID (primary key), request date, requester contact, patient's details, location of anointing, priest preference, status, processed date, approved by and service by as the foreign keys

Table 8. Anointing of the sick request

Fields	Type	Description
RequestID(PK)	int	Unique identifier for each anointing of the sick request
RequestDate	varchar	Date when the Anointing of the Sick request was made
RequesterContact	varchar	Contact information of the requester (phone/email)
PatientFullName	varchar	Full name of the person to receive Anointing of the Sick
PatientAddress	varchar	Address of the person to receive anointing of the Sick
PatientCondition	varchar	Current medical condition of the patient
Sex	varchar	Gender of the person to receive anointing of the sick
PreferredDate	varchar	Preferred date for the Anointing of the Sick
PreferredTime	varchar	Preferred time for the anointing of the sick
LAnnointing	varchar	Location of anointing
Age	int	Age of the person to receive the anointing of the sick
PriestPreference	varchar	Name of the preferred priest to perform the Anointing
Status	varchar	Status of the request (e.g., Pending, Approved)
ProcessedDate	varchar	Date when the request was processed
ApprovedBy(FK)	varchar	The name of the person who approved the request.

Table 9: The Wedding Request table includes a unique requester ID (primary key), request date, request time, groom's details, bride's details, status, processed date, approved by and service by as the foreign keys.

Table 9. Wedding Request

Fields	Type	Description
RequesterID(PK)	int	Unique identifier for each wedding request
RequestDate	varchar	Date when the wedding request will be made
RequestTime	varchar	Preferred time of the wedding
GroomFullName	varchar	Full name of the groom
GroomAddress	varchar	Address of the groom
Groomphone	int	Phone number of the groom
BrideEmail	varchar	Email address of the bride
ApprovedBy(FK)	varchar	The name of the person who approved the request.
ServiceBy(FK)	varchar	Identifier of the person or entity that provided the service or handled the request
Status	varchar	Current status of the request (e.g., Pending, Approved)

Table 10: The Confirmation Request table includes a unique request ID (primary key), RequestDate, details of the person to be confirmed, name of the father and mother, preferred date, status, processed date, approved by and service by as the foreign keys.

Table 10. Confirmation Request

Fields	Type	Description
RequestID(PK)	int	Unique identifier for each confirmation request
RequestDate	timestamp	Date when the confirmation request was made
FullName	varchar	Full name of the person to be confirmed
DateOfBirth	varchar	Date of birth of the person to be confirmed
PlaceOfBirth	varchar	Place of birth of the person to be confirmed
Phone	int	Contact Number of the person to be confirmed
FatherFullName	varchar	Full name of the father of the person to be confirmed
MotherFullName	varchar	Full name of the mother of the person to be confirmed
Gender	varchar	Gender of the person to be confirmed
Age	int	Age of the person to be confirmed
FatherFullName	varchar	Full name of the father of the person to be confirmed
MotherFullName	varchar	Full name of the mother of the person to be confirmed
Gender	varchar	Gender of the person to be confirmed
Age	int	Age of the person to be confirmed
PreferredCDate	varchar	Preferred Confirmation Date
Status	varchar	Status of the request (e.g., Pending, Approved)
ProcessedDate	varchar	Date when the request was processed
ApprovedBy(FK)	varchar	The name of the person who approved the request.
ServiceBy(FK)	varchar	Identifier of the person or entity that provided the service or handled the request

Table 11: The First Communion Request table includes a unique request ID (primary key), request date, details of the person to receive first communion, details of the mother and father, status, processed date, approved by and service by as the foreign keys.

Table 11. First Communion Request

Fields	Type	Description
RequestID(PK)	int	Unique identifier for each communion request
RequestDate	varchar	Date when the communion request was made
PreferredDate	varchar	Preferred date of the requester of the first communion
MinisterName	varchar	The name of the minister who will handle the first communion.
FullName	varchar	The full name of the person who will receive the first communion.
FatherName	varchar	The name of the father of the person who will receive the first communion.
MotherName	varchar	The name of the mother of the person who will receive the first communion.
Sponsors	varchar	The name of the sponsor of the first communion.
Residence	varchar	The residence of the person who will receive the first communion.
BookNo	int	Book number
Year	int	Year of the request
RecordNo	int	Record number

ApprovedBy(FK)	varchar	The name of the person who approved the request.
ServiceBy(FK)	varchar	Identifier of the person or entity that provided the service or handled the request
ProccesedDate	varchar	The date when is the request was processed
Status	varchar	Current status of the feedback (e.g., Pending, Reviewed)

Table 12: The Funeral and Burial Request table includes a unique request ID (primary key), request date, preferred funeral date, deceased person's details, the name of the spouse if married, name of mother and father, confirmation details, last rites details, status, processed date, approved by and service by as the foreign keys.

Table 12. Funeral and Burial Request

Fields	Type	Description
RequestID(PK)	int	Unique identifier for each funeral and burial request
RequestDate	varchar	Date when the funeral and burial request was made
PreferredFuneralDate	varchar	Preferred date for the funeral service
DeceasedFullName	varchar	Full name of the deceased
DeceasedDateOfBirth	varchar	Date of birth of the deceased
DeceasedDateOfDeath	varchar	Date of death of the deceased
CStatus	varchar	Civil status if single, married, widowed, divorced
DFatherName	varchar	Name of father of the deceased person
DMotherName	varchar	Name of the mother of the deceased person
CDeath	varchar	Cause of death of the deceased person.
BMassTime	varchar	Date and time of the funeral and burial mass.
BLocation	varchar	The location of the burial.
SpouseName	varchar	The name of the spouse of the deceased person
SacReceived	varchar	The sacraments received of the deceased person.
FAssistant	varchar	Name of the assistant who will officiate the funeral.
PriestName	varchar	The name of the priest who will handle the mass.
ProcessedDate	varchar	Date when the request was processed
ApprovedBy(FK)	varchar	The name of the person who approved the request.
ServiceBy(FK)	varchar	Identifier of the person or entity that provided the service or handled the request
Status	varchar	Current status of the feedback (e.g., Pending, Reviewed)

Table 13: The feedback table includes a unique feedback ID (primary key), feedback date, user name, user contact, feedback type, feedback text, status, and processed date

Table 13. Feedback

Fields	Type	Description
FeedbackID(PK)	int	Unique identifier for each feedback
FeedbackDate	varchar	Date when the feedback was submitted
UserName	varchar	Full name of the person giving feedback
UserContact	varchar	Contact information of the user (phone/email)
FeedbackType	varchar	Type of feedback (e.g., suggestion, complaint, praise)
FeedbackText	varchar	Detailed feedback from the user
Status	varchar	Current status of the feedback (e.g., Pending, Reviewed)
ProcessedDate	varchar	Date when the feedback was processed

Table 14: The Announcement table includes a unique AnnouncementID (primary key), UserID(identifier of the user making the announcement), Title, Content, and Announcement Dates, with AnnouncementID as the primary key and no identified foreign keys.

Table 14. Announcement

Fields	Type	Description
AnnouncementID(PK)	int	Unique identifier for each Announcement
UserID	varchar	Identifier of the user making the announcement
Title	varchar	Title of the announcement
Content	varchar	Content of the announcement
AnnouncementDates	varchar	Dates on which the announcement is active

Table 15: Scheduling table includes a unique ScheduleID (primary key).

Table 15. Scheduling

Fields	Type	Description
ScheduleID(PK)	int	Unique identifier for each schedule entry
SacramentType	varchar	Type of sacrament requested (e.g., Baptism, Confirmation, Marriage)
RequestID	varchar	Identifier of the related sacrament request
RequesterName	varchar	Full name of the person who made the request
RequesterContact	varchar	Contact information of the requester (phone/email)
EventDate	varchar	Date of the sacrament event
StartTime	varchar	Start time of the event
EndTime	varchar	End time of the sacrament event
Status	varchar	Current status of the schedule (e.g., Scheduled, Completed)
CreatedDate	varchar	Date when the schedule was created

2.4.4 Architecture Design

This architectural design shows how the church management system works. Church staff and parishioners use computers to access the system via the Internet. They interact through a web interface, which connects to a MySQL database.

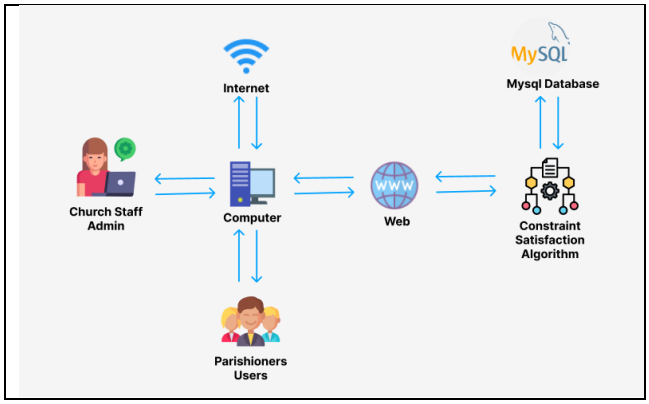

Figure 5. Architecture Design

2.4.5 Sample Prototypes

The figure number shows the different pages and forms for the users in our church management system. Each figure represents a specific page that the users will interact with.

Figure 6. Home Page

The figure 6 shows the main landing page where users can access various features.

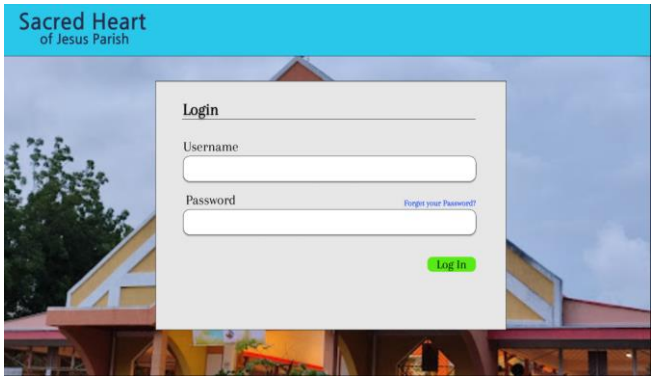

Figure 7. Login Page

The figure 7 shows where users enter their credentials to access the system.

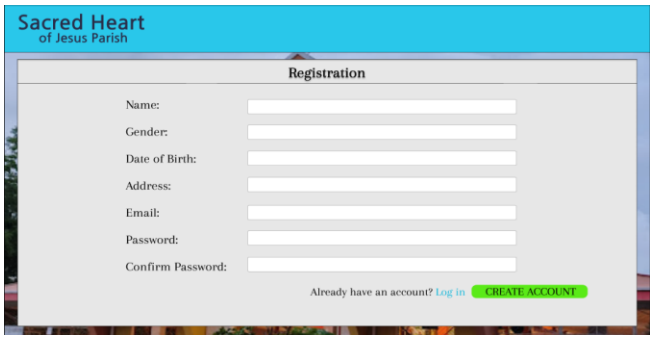

Figure 8. Registration Page

The figure 8 shows the page where new users can create an account.

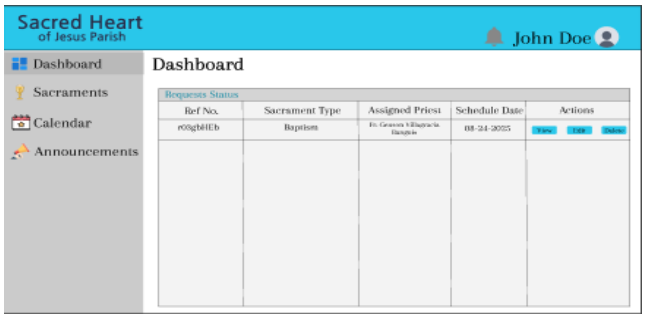

Figure 9. Dashboard User Page

The figure 9 shows the page displaying the status of the user's recent requests.

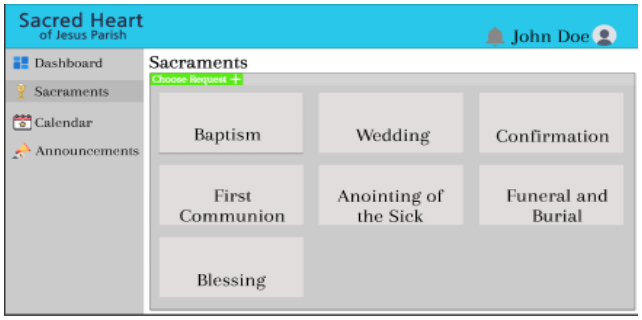

Figure 10. Choose Sacraments Page

The figure 10 shows the page where users select the sacrament they want to request.

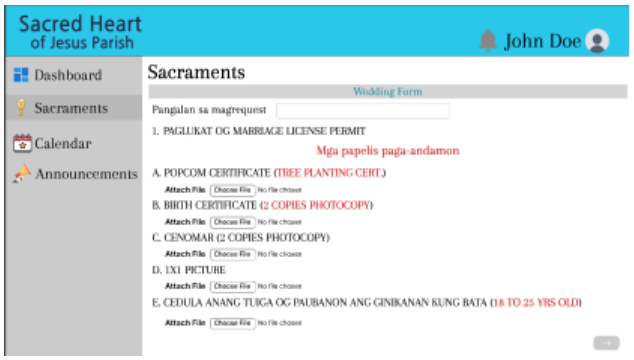

Figure 11. Wedding Form

The figure 11 shows the form used to request a wedding sacrament.

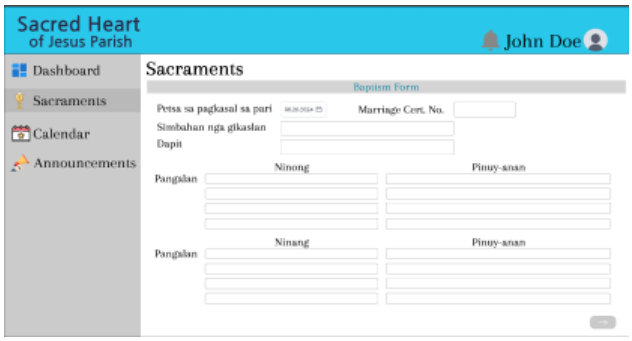

Figure 12. Baptism Form

The figure 12 shows the form used to request a baptism sacrament.

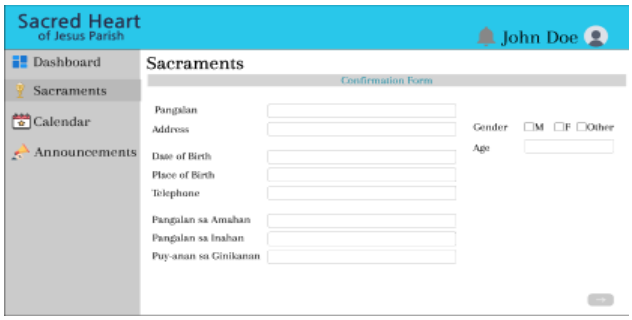

Figure 13. Confirmation Form

The figure 13 shows the form used to request a confirmation sacrament.

Figure 14. First Communion Form

The figure 14 shows the form used to request a first communion sacrament.

Figure 15. Anointing to the sick Form

The figure 15 shows the form used to request an anointing of the sick sacrament.

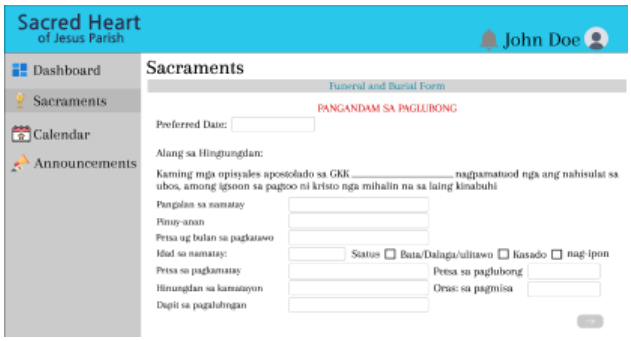

Figure 16. Funeral and Burial Form

The figure 16 shows the form used to request a funeral and burial service.

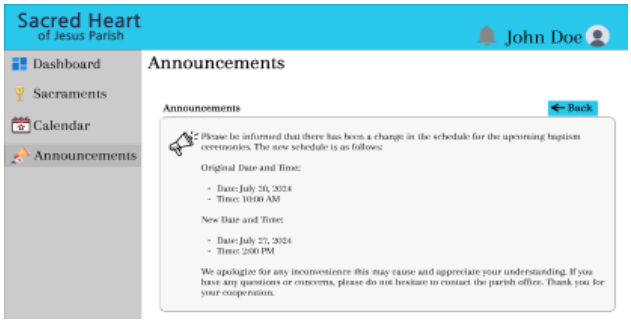

Figure 17. Announcement User Form

The figure 17 shows the form used by users to receive announcements.

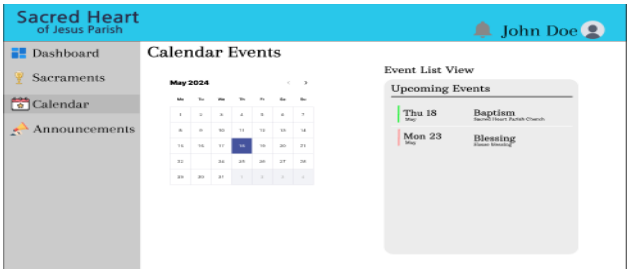

Figure 18. Event calendar Form

The figure 18 shows the page displaying scheduled church events.

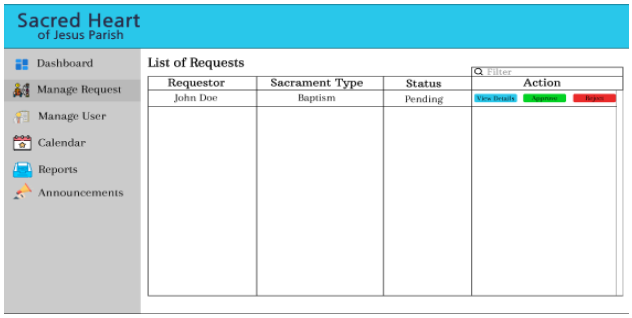

Figure 19. Manage Request Form

The figure 19 shows the form used by the admin to manage sacrament requests.

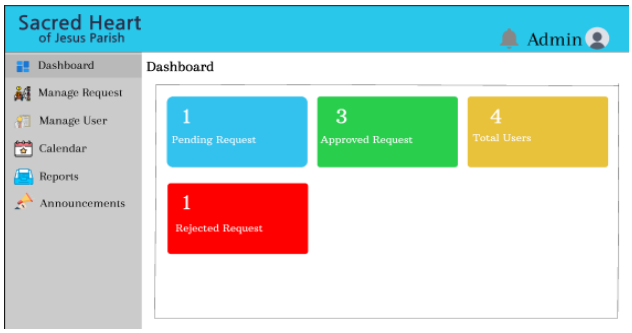

Figure 20. Dashboard Form

The figure 20 shows the dashboard providing an overview of requests and activities.

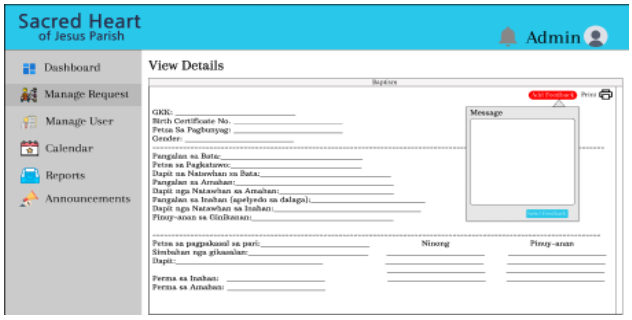

Figure 21. Review Requests Form

The figure 21 shows the form where the admin can review requests and provide feedback to the user if the submitted documents are correct.

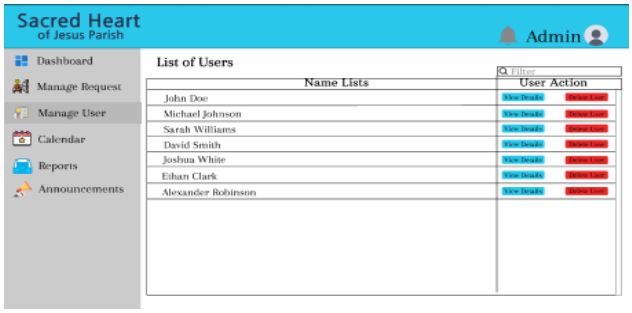

Figure 22. Manage User Form

The figure 22 shows the form used by the admin to manage user accounts.

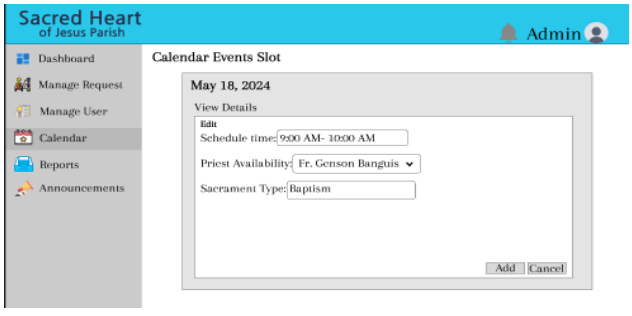

Figure 23. Add Time Slot Form

The figure 23 shows the form where the admin can add available time slots for sacraments.

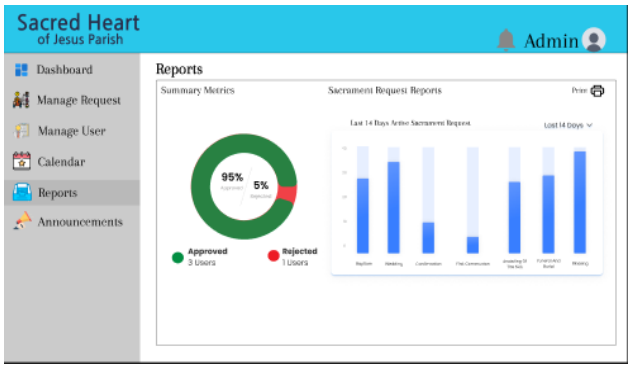

Figure 24. Report Form

The figure 24 shows the form used to generate sacrament request reports.

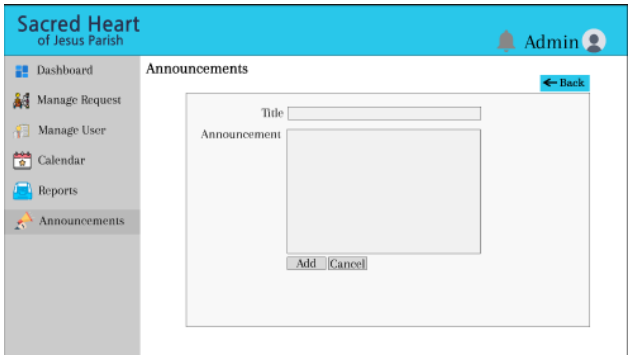

Figure 25. Announcement Form

The figure 25 shows the form used by the admin to make announcements.

3. REFERENCES

[1] A. Hunter, "Automating Church Operations: A Guide to Streamlined Ministries - Ministry Vitals," Ministry Vitals, Sep. 08, 2023. https://ministryvitals.com/automating-church-operations/

[2] A. Phillips, "The Role of Technology in Church Growth and Outreach: Enhancing Community Engagement (2024)," The Witness, May 21, 2024. https://thewitness.org/the-role-of-technology-in-church-growth-and-outreach/

[3] "Tech in churches," Tech in Churches. https://www.techinchurches.org/

[4] K. Parker, "Archiving church records: What items to keep and how to catalogue them," UUA.org, Dec. 15, 2023. https://www.uua.org/leaderlab/archiving-church-records

[5] A. Mamane, M. Fattah, M. E. Ghazi, M. E. Bekkali, Y. Balboul, and S. Mazer, "Scheduling Algorithms for 5G Networks and Beyond: Classification and Survey," *IEEE Access*, vol. 10, pp. 51643-51661, 2022, doi: 10.1109/ACCESS.2022.3174579.

[6] K. Senjab, S. Abbas, N. Ahmed, and A. U. R. Khan, "A survey of Kubernetes scheduling algorithms," Journal of Cloud Computing, vol. 12, no. 1, Jun. 2023, doi: 10.1186/s13677-023-00471-1.

[7] V. Ďuriš, "Algorithmic Verification of Constraint Satisfaction Method on Timetable Problem," Mathematics and Statistics, vol. 8, no. 6, pp. 728-739, 2020. doi: 10.13189/ms.2020.080614.

[8] S. C. Brailsford, C. N. Potts, and B. M. Smith, "Constraint satisfaction problems: Algorithms and applications," *European Journal of Operational Research*, vol. 119, no. 3, pp. 557–581, Dec. 1999, doi: 10.1016/s0377-2217(98)00364-6.

[9] "View of technological advancement and the task of the church in the virtual world." https://www.shahidihub.org/shahidihub/index.php/ijtrs/article/view/139/50

[10] B. Oluwade, F. P. Abifarin, and T. Aro, "Online Church Management System," in Proceedings of the 2022 Conference of the Society for the Advancement of ICT Comparative Knowledge (SOCTHADICKconf'22), 2022, p. 199. [Online]. Available: https://www.researchgate.net/profile/Bamidele-Oluwade/publication/375959307_PROCEEDINGS_OF_THE_2022_CONFERENCE_OF_THE_SOCIETY_FOR_THE_ADVANCEMENT_OF_ICT_COMPARATIVE_KNOWLEDGE_SOCTHADICKconf'22_EDITORS_Bamidele_Oluwade_Fasola_P_Abifarin_and_Taye_Aro_THEME_ICT_as_a_Key_Driving_/links/65654042ce88b87031199ea4/PROCEEDINGS-OF-THE-2022-CONFERENCE-OF-THE-SOCIETY-FOR-THE-ADVANCEMENT-OF-ICT-COMPARATIVE-KNOWLEDGE-SOCTHADICKconf22-EDITORS-Bamidele-Oluwade-Fasola-P-Abifarin-and-Taye-Aro-THEME-ICT-as-a-Key-Drivin.pdf#page=199

[11] Sabuncuoglu Ihsan, Kizilisik Omer Batuhan. Reactive Scheduling in a Dynamic and Stochastic FMS Environment [J]. International Journal of Production Research, 2003, 41(17): 4211-4231.

[12] "Church and Technology: Necessity, types of technology, advantages and disadvantages," Ministry Brands. https://www.ministrybrands.com/church/management/technology/

[13] Excellerate Church Software, "Church Management Software - Excellerate," Excellerate, May 03, 2024. https://www.excellerate.com/

[14] "Catholic Church software for parishes and dioceses | ParishSOFT," *ParishSOFT*, Jun. 10, 2024. https://www.parishsoft.com/

[15] "CCE Research Center - CCE Research Agenda." https://sites.google.com/umindanao.edu.ph/cceresearchcenter/cce-research-agenda?authuser=0

[16] GeeksforGeeks, "Constraint Satisfaction Problems (CSP) in artificial intelligence," GeeksforGeeks, Jun. 06, 2024. https://www.geeksforgeeks.org/constraint-satisfaction-problems-csp-in-artificial-intelligence/