A blue and gold logo

Description automatically generated

Software Requirements Specification

for

Chambers of the Burning Ashes System (CBAS)

Requirements for Version 1.0.0

Prepared by

Kyle Philippe M. Santos

Jacob De Villa

Janson Pagharion

David Naldo

Ramon Surara

Team Phishda

02-12-2024

**Table of Contents**

[Revision History (may not be needed) 2](#_Toc496642903)

[1.Introduction 2](#_Toc753575485)

[1.1 Purpose 3](#_Toc1520927481)

[1.2 Document Conventions 3](#_Toc334151312)

[1.3 Intended Audience and Reading Suggestions 3](#_Toc369737351)

[1.4 Project Scope 4](#_Toc87645696)

[2.Overall Description 5](#_Toc1657244200)

[2.1 Product Perspective 5](#_Toc3184425)

[2.2 Product Features 6](#_Toc1604185607)

[2.3 User Classes and Characteristics 7](#_Toc406544423)

[2.4 Fully Dressed Use Cases 8](#_Toc1447281096)

[2.5 Operating Environment 14](#_Toc279520080)

[2.6 Design and Implementation Constraints 15](#_Toc246540906)

[2.7 User Documentation 15](#_Toc2083100625)

[2.8 Assumptions and Dependencies 15](#_Toc1626222211)

[3. External Interface Requirements 16](#_Toc12659142)

[3.1 User Interface 16](#_Toc1215776936)

[3.1.1 User Interfaces 16](#_Toc2116043404)

[3.2 Hardware Interface 24](#_Toc903069951)

[3.3 Software Interface: 25](#_Toc989856281)

[3.4 Communication Interface 25](#_Toc914799775)

[4.System Features 25](#_Toc125197119)

[4.1 Columbary Reservation System 25](#_Toc1594453721)

[4.1.1 Description and Priority 25](#_Toc1257690645)

[4.1.2 Stimulus/Response Sequences 25](#_Toc930467744)

[4.2 Information Retrieval System 26](#_Toc943428433)

[4.2.1 Description and Priority 26](#_Toc1500431352)

[4.2.2 Stimulus/Response Sequences 26](#_Toc240846560)

[4.3 Administrative Review System 26](#_Toc658014434)

[4.3.1 Description and Priority 27](#_Toc45259795)

[4.3.2 Stimulus/Response Sequences 27](#_Toc812454855)

[4.4 Dashboard Management 27](#_Toc1125490956)

[4.4.1 Description and Priority 27](#_Toc623895043)

[4.4.2 Stimulus/Response Sequences 27](#_Toc1019606870)

[4.5.1 Description and Priority 28](#_Toc152124881)

[4.5.2 Stimulus/Response Sequences 28](#_Toc1318347830)

[5.Other Nonfunctional Requirements 29](#_Toc1836488919)

[5.1 Performance Requirements 29](#_Toc2138949870)

[5.2 Safety Requirements 29](#_Toc1720040653)

[5.3 Security Requirements 29](#_Toc8280121)

[5.4 Software Quality Attributes 29](#_Toc1371943191)

[5.5 Business Rules 30](#_Toc557584009)

[6. Other Requirements 30](#_Toc1054095064)

[6.1 Target Metrics (Objectives) 30](#_Toc876938237)

[Appendix A: Glossary 31](#_Toc1446211193)

[Appendix B: Analysis Models 31](#_Toc1608813817)

[Appendix C: GitHub Repository 40](#_Toc2025363154)

# Revision History (may not be needed)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Reason For Changes | Version |
|  |  |  |  |
|  |  |  |  |

# 1.Introduction

## 1.1 Purpose

The document outlines the software requirements for the Chambers of The Burning Ashes System (CBAS). CBAS contains two programs:

* CBAS Server – Server for the CBAS that also contains the database of the Columbary customers and its tenants.
* CBAS Program – It is a program mostly hosted on a website to help with the Parish Staff and Customers when accessing the columbary records or visiting the website in general.

CBAS aims to create a system where all records of the columbary are now digitalized, to reduce the reliance on paper-based records for the parish.

## 1.2 Document Conventions

The following conventions are used throughout this Software Requirements Specification (SRS) document:

* **Headings and Titles**: Headings and titles are in bold font and are numbered for easy reference.
* **Section Numbers**: Section numbers are in the numbering format, wherein the main sections are integers while subsections are floating numbers that have a decimal according to which section they are in.
* **Figure and Table Numbers**: Figures and tables are numbered consecutively throughout the document, with captions below each figure and above each table.
* **Terminology**: Definitions of technical terms and acronyms are provided in the Glossary (Appendix A).
* **Font Styles**: Times New Roman is the only font style that was used for this document.
  + **Bold**: Used for headings, titles, and emphasis.
  + **Regular**: Used for body text.
* **Abbreviations and Acronyms**: Abbreviations and acronyms are defined at their first occurrence and listed in the Glossary

## 1.3 Intended Audience and Reading Suggestions

This document is intended for any individual user ,developer, tester, project manager or documentation writer that needs to understand the basic system architecture and its specifications.

Here are the potential uses for each one of the reader types:

1. Parish Administrator: When new requirements or changes are needed, the Parish Administrator should refer to this document to ensure proper updates. Modifications must maintain the integrity of the original requirements and communicate these changes accurately to subsequent development phases.
2. Parish Staff: Reviews the diagrams and specifications in this document to verify that the software fulfills all necessary requirements. Their feedback ensures that the developed solution aligns with their functional needs.
3. Developer: Uses this document as a reference to confirm that the software’s implementation aligns with the specified requirements and provides a guide for further development or debugging.
4. Tester: Compares the documented requirements against the executable program to confirm accuracy and compliance with the initial specifications.
5. Customers**:** This document is for customers when making use of the CBAS application, this SRS can server as a guide for their reference.

Overview

1. **Introduction: Provide** an overview of the application, describe the document structure and point the individual objectives.
2. **Overall Description: Provide** the specification of the system model, the classes model, the main constraints and the list any assumed factors that used within this document.
3. **System Features: Provide** the analysis of the requirements by feature.
4. **External Interface Requirements: Provide** the visualization of the program and the requirements that are related with hardware, software and networking.
5. **Other Nonfunctional Requirements: Provide** some other constraints that apply to factors such as performance, safety and security.

## 1.4 Project Scope

The Chambers of the Burning Ashes System is designed to modernize St. Alphonsus Mary de Liguori Parish's operations through an easy-to-use and comprehensive web application of the parishioners. It will facilitate simplified management of columbarium services that will be affected by duplication or lack of records, inefficiency in record retrieval processes, and having no centralized facility to record customers and vault information. The project will also involve digitizing vault and columbarium niche records for accurate tracking of availability, payment status, and contract validity. A secure online portal will be developed to give the customers their columbarium information and transaction history, with encryption and authentication protocols safeguarding sensitive data.

It will automatically process and store the necessary documents used in the acquisition of columbarium niches, ensuring regulatory compliance and general accuracy. It will also have a back-up solution with redundancy and secure cloud storage to avoid data loss and quick recovery in case of a failure in the system. The project will incorporate user training and documentation, providing for the smooth integration of the system by church staff to transition into modern practices. In the end, this will enhance operational efficiency, boost customer experience, and help in offering scalable, secure solutions that meet the needs that are increasing in the columbarium services provided for the parish.

Its main objective is to set up a local and remote connection environment that allows users like Parish staff and administrator to manage columbaries using a server that customers can connect having limited access to the server and can avail for an available columbary. Furthermore, to set up a local server where the application can run locally and implement a backup solution for the database into the server. The goal is to make it possible for the Parish Staff to provide services to their customers and help the Parish administrator in approving letters of intent through the internet.

# 2.Overall Description

## 2.1 Product Perspective

The Chambers of the Burning Ashes System (CBAS) is a custom-built solution designed specifically for St. Alphonsus Mary de Liguori Parish to address inefficiencies in managing their columbarium services. It replaces the outdated manual record-keeping process with a centralized digital platform that ensures accuracy, reliability, and security.

CBAS is composed of two main components:

1. CBAS Server:

Serves as the central repository for columbarium records and customer data.

Provides secure access for staff to manage records and transactions.

1. CBAS Program:

A user-friendly web-based application that allows parish staff to manage records and customers to view their columbarium information.

1. Features role-based access controls to distinguish between staff and customer privileges.

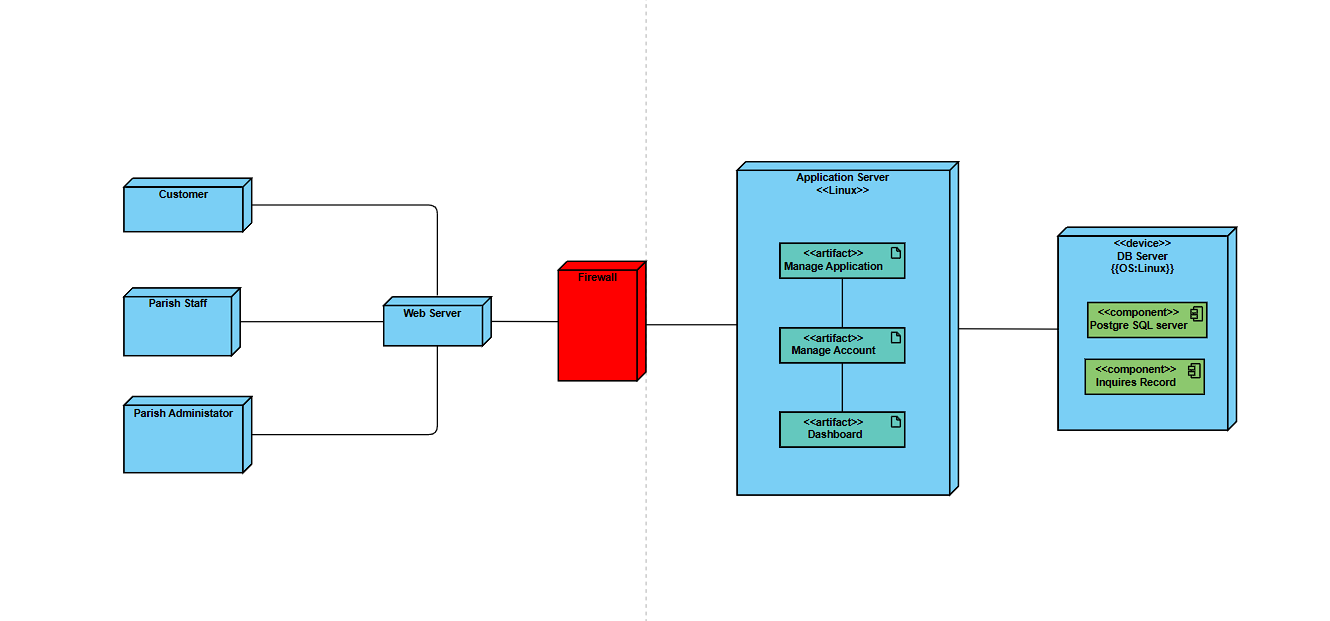
The system integrates with modern cloud-based storage solutions (e.g., Nextcloud) and utilizes technologies like Django (Python framework) and MySQL for database management. Analytics and reporting are supported by Python libraries such as Matplotlib.  


Figure 1: Deployment Diagram

## 2.2 Product Features

CBAS offers the following key features:

* Digital Record Management:
  + Secure storage and retrieval of columbarium tenant and vault information.
  + Automated tracking of payment statuses and contract validity.
* User Access Portal:
  + Customers can view their transaction history and niche information through a secure online portal.
* Backup and Recovery:
  + Redundant backups using Nextcloud for disaster recovery and data protection.
* Report Generation:
  + Automated reports on sales trends, columbarium availability, and tenant records for operational analysis.
* Analytics and Insights:
  + Built-in analytics tools powered by Python libraries to assist in decision-making and trend analysis.
* System Security:
  + Implements encryption protocols to protect sensitive data.
  + Role-based authentication for secure user access.

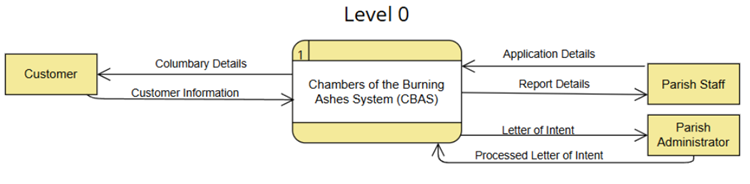


Figure 2: Level 0 Data Flow Diagram/Context Diagram

## User Classes and Characteristics

Table 1: User Classes and Characteristics

|  |  |
| --- | --- |
| Roles | Description |
| Parish Office Staff | The main user of the new system that is being  developed |
| Parish Administrator | The one who oversees the management of  columbaries and check them sometimes. |
| Customer | The possible consumers of the columbary  services offered by the Parish |

## 2.4 Fully Dressed Use Cases

Table 2: Send Letter of Intent

|  |  |
| --- | --- |
| Use Case Name | Send letter of intent |
| Use Case Number | UC-01 |
| Actors | Customer, Parish Administrator |
| Description | This use case presents how the customer sends the letter of intent to the parish. |
| Pre-Conditions | * The columbary map and pricing information are up to date. * The parish CBA systems are operational and accessible. * Customer wants to avail a columbary. |
| Post Conditions | * The customer successfully views available columbaries and their prices. * A letter of intent with the customer’s information is generated and sent to the parish administrator. * The parish administrator receives a notification of letter of intent in the CBA system from the customer. * The Parish administrator receives an email to notify t |
| Main Scenario | 1. Customer visits the system. 2. Customer Navigates to the Columbary section of the system. 3. Customer clicks “avail now” button. 4. Customer view available columbaries in the columbary map. 5. Customer chooses columbary location. 6. Customer views the pricing and sections of each columbary. 7. Customer chooses a columbary and clicks “Avail now”. 8. Terms & conditions will appear before proceeding. 9. Customer reads the terms & conditions 10. Customer fills in customer information such as email, Cellphone number, Name, and address 11. Customer clicks on the “send” button to send a letter of intent to the Parish. |

Table 3: Retrieve Columbary Information

|  |  |
| --- | --- |
| Use Case Name | Retrieve Columbary Information |
| Use Case Number | UC-02 |
| Actors | Customer, Parish Staff |
| Description | This use case describes how the system retrieves information relevant to the customer. |
| Pre-Conditions | * The customer has previously applied for and purchased a columbary and provided a phone number and email address during the application process. * The customer has access to the internet and a valid email account. * The CBA system website is operational, and the customer’s columbary information is stored and accessible within the system. * Customer receives and inputs the correct one-time pin. |
| Post Conditions | * The customer successfully retrieves their columbary information via email, including any relevant documents. * The system logs the retrieval request and verification process for auditing purposes. * The system ensures that the customer’s information is securely transmitted and accessed. |
| Main Scenario | 1. Customers navigates to the retrieve columbary tab. 2. Customer clicks the retrieve information button. 3. Customer inputs their email address or phone number. 4. The system will generate an OTP and send it to the customer. 5. Customer receives their OTP. 6. Customer inputs their OTP code into the website. 7. The customer receives an email containing minimal columbary information. |

Table 4: Review Letter of Intent

|  |  |
| --- | --- |
| Use Case Name | Review letter of intent |
| Use Case Number | UC-03 |
| Actors | Parish Administrator, Customer, Parish Staff |
| Description | This use case describes how the parish administrator reviews the letter of intent sent by the customer and decides either to approve or deny it. |
| Pre-Conditions | * The Parish Administrator logged in to the CBA system. * The Parish admin navigated to the inquiries tab. * A letter of intent has been sent by the customer. |
| Post Conditions | * The inquiry has been reviewed. * A decision has been made. * Customer Receives letter of approval/denial through email. * The approved letter of intent is sent to the Parish Staff. |
| Main Scenario | 1. The parish administrator navigates to the inquiries tab. 2. Parish administrator chooses a letter of intent 3. The parish administrator reviews the letter of intent. 4. The parish administrator approves the letter of intent. 5. The system sends a letter of approval to the customer’s email. 6. The system notifies and sends the approved letter of intent to the Parish Staff. |
| Alternate Scenario | 1. The parish administrator navigates to the LOI (letter of intent) tab. 2. The parish administrator reviews the letter of intent. 3. The parish administrator denied the letter of intent. 4. The system sends a letter of denial to the customer’s email. |

Table 5: Manage Dashboard

|  |  |
| --- | --- |
| Use Case Name | Manage Dashboard |
| Use Case Number | UC-04 |
| Actors | Parish Staff |
| Description | This use case describes how the parish staff views necessary customer and columbary information and how they manage the CBA systems dashboard and AI assistant. |
| Pre-Conditions | * The parish staff is logged into the CBA system. * The CBA system is fully operational. * All records are viewable. * Columbary information is viewable. * CBA system AI assistant is functional. |
| Post Conditions | * PA |
| Main Scenario | 1. Parish Staff logs in to the system 2. Parish views the dashboard 3. Parish Staff views the columbary map 4. Parish Staff can view the available columbaries 5. Parish Staff navigates the mic icon and clicks on it. 6. AI assistant is now opened and ready for prompts 7. Parish staff inputs his prompt 8. Ai assistant answers the query. |

Table 6: Manage Records

|  |  |
| --- | --- |
| Use Case Name | Manage Records |
| Use Case Number | UC-05 customer |
| Actors | Parish Staff |
| Description | This use case describes how the parish staff opens the CBA system to update, create and delete customer and columbary records. |
| Pre-Conditions | * The parish staff member has valid login credentials (username and password). * The CBA system is operational and accessible. * The parish staff’s access permissions are properly configured in the system. * Parish staff is Logged-in in the system |
| Post Conditions | * The parish staff perform their duties, including managing customer records. * The Parish staff can add, edit and delete records. * Parish staff can scan new applications |
| Main Scenario | 1. The parish staff logs in to the system 2. The parish staff has three options add, edit and delete 3. The parish staff clicks on the “add” button to add new records. 4. The parish staff clicks on an existing columbary record and clicks on edit. 5. The Parish staff opens an existing columbary record and clicks on delete. 6. The parish staff clicks on the “Save button” |

Table 7: Manage Accounts

|  |  |
| --- | --- |
| Use Case Name | Manage Accounts |
| Use Case Number | UC-06 |
| Actors | Parish Staff, Parish Administrator |
| Description | This use case describes how the parish staff manage parish accounts on the CBA system. |
| Pre-Conditions | * The parish staff is logged into the CBA system with the necessary permissions to manage accounts. * The parish staff has navigated to the main dashboard of the CBA system. * The parish admin can be granted permissions to the system |
| Post Conditions | * The parish staff deletes, updates, manages permissions or creates an account for the CBA system. |
| Main Scenario | 1. The parish staff navigates to the accounts tab. 2. The parish Staff can then create, update, manage permissions, and delete accounts through the CBA System. 3. Parish staff manage access permissions to the Parish Administrator. |

## 2.5 Operating Environment

CBAS operates in the following environment:

* Server Requirements:
  + Ubuntu or Windows Server operating system.
  + MySQL database for managing records.
  + AWS for backup storage.
* Client Requirements:
  + Modern web browser (e.g., Chrome, Firefox, Edge, Safari).
  + Internet connection for accessing the online portal.
* Technologies Used:
  + Django (Python framework) for backend development.
  + Matplotlib and other Python libraries for analytics.
  + HTML/CSS and JavaScript for the frontend.

## 2.6 Design and Implementation Constraints

The Chambers of the Burning Ashes System (CBAS) is developed using the **Django framework** with **MySQL** for database management and integrates **AWS** for backup storage. It is optimized for modern server and client hardware, with the following requirements:

1. Minimum System Requirements:
   * Server:
     + OS: Kubuntu Linux Distributable
     + RAM: 16 GB | CPU: Ryzen 5/Intel i5 Processor | Storage: 500GB + 500GB SSD RAID 0 Configuration
     + Network: Stable internet connection
   * Client:
     + Modern web browser (e.g., Chrome, Firefox, Brave)
     + 16 GB RAM and Ryzen 5/Intel i5 Processor
2. **Installation**:  
   A guided installer handles system setup, including Django configuration and Nextcloud backup integration.
3. **Language Support**:  
   Default language is English; future updates may support localization (e.g., Filipino, Spanish).
4. **Network Protocols**:  
   Uses **HTTP/HTTPS** for secure communication and **TCP/IP** for server-client connections.
5. **Warranty**:  
   CBAS is provided "AS IS" with no warranty. Regular backups are recommended to prevent data loss.

## 2.7 User Documentation

Comprehensive user manuals and guides will be provided to facilitate the system's adoption:

1. **Parish Staff Guide**: Instructions for managing records, generating reports, and using the backup system.
2. **Customer Guide**: Steps to access columbarium records and view transaction history.
3. **Technical Documentation**: Detailed information for developers and testers, including API references and database schema.

## 2.8 Assumptions and Dependencies

The Chambers of the Burning Ashes System (CBAS) relies on the Django framework for backend development, offering a robust set of tools for building web applications with a focus on security and scalability. The system also integrates MySQL for database management and AWS for cloud storage. These technologies ensure seamless data handling and secure cloud backup solutions.

For better understanding, this document assumes the reader is more interested in the system's functionality rather than the technical details of how Django, MySQL, and AWS are implemented or how the code is written and structured.

# 3. External Interface Requirements

## 3.1 User Interface

This section will cover the external interface requirements for the Chambers of the Burning Ashes System (CBAS). This section will include the user interface, hardware interface, and the interactions the users will have with the system.

### 3.1.1 User Interfaces

The web interface is designed using vanilla HTML, JavaScript, and CSS. It will be using the Django framework. The following is a prototype of the web application:

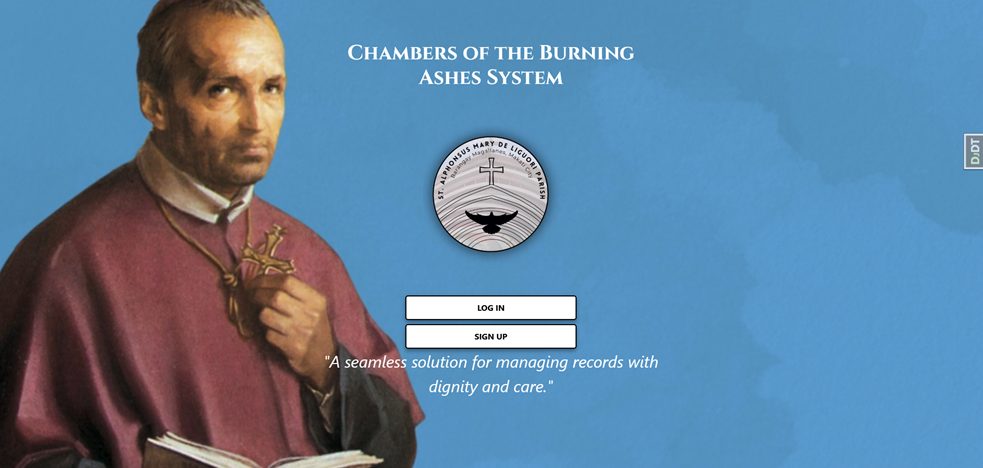


Figure 3: Login Page

CBAS Login Page for the Parish Staff and Parish Administrator to login into the system.

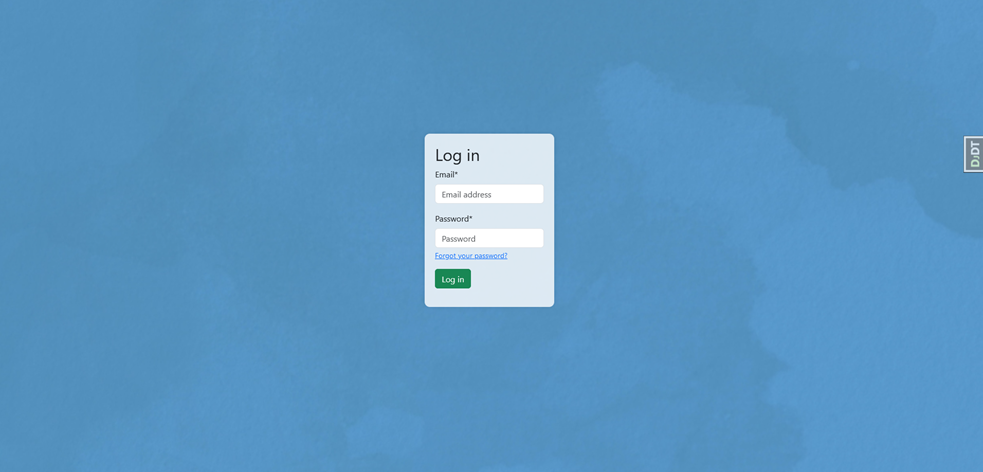


Figure 4: Login Details Page

CBAS login details page for the Parish Staff and Administrator to input their email and/or username and password to successfully login to the system.



Figure 5: Home Page & Dashboard

The Parish Staff and Parish Administrator Home page showcases the different available pages and features of the system and the dashboard in which a daily report will be shown to the user.

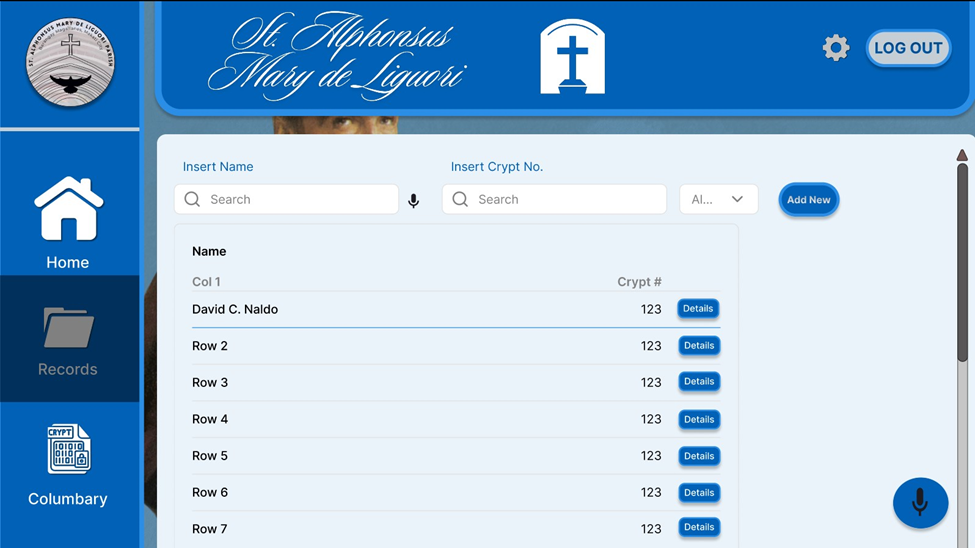


Figure 6: Records Page

Records Page for the Parish Staff to add, delete, and edit customer records within the system, and for the Parish Administrator to view customer records.



Figure 7: Customer Record Page

Customer record page in which all the available files, status, inurnment date and all necessary information are in display for the Parish Staff and Parish Administrator.

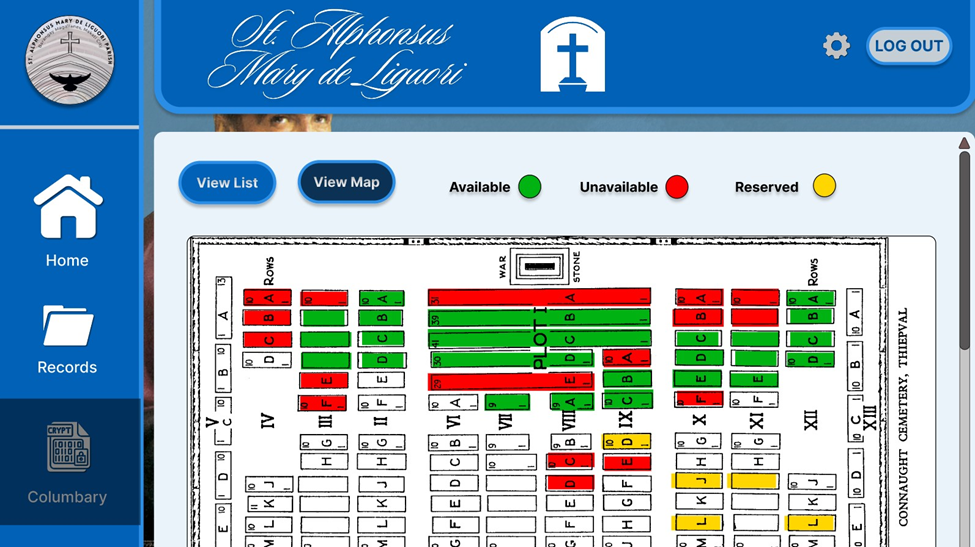


Figure 8: Columbary Map Page

Columbary Map Page for the Parish Staff and Administrator to see the availability of crypts to see the reserved, available, and unavailable spots.

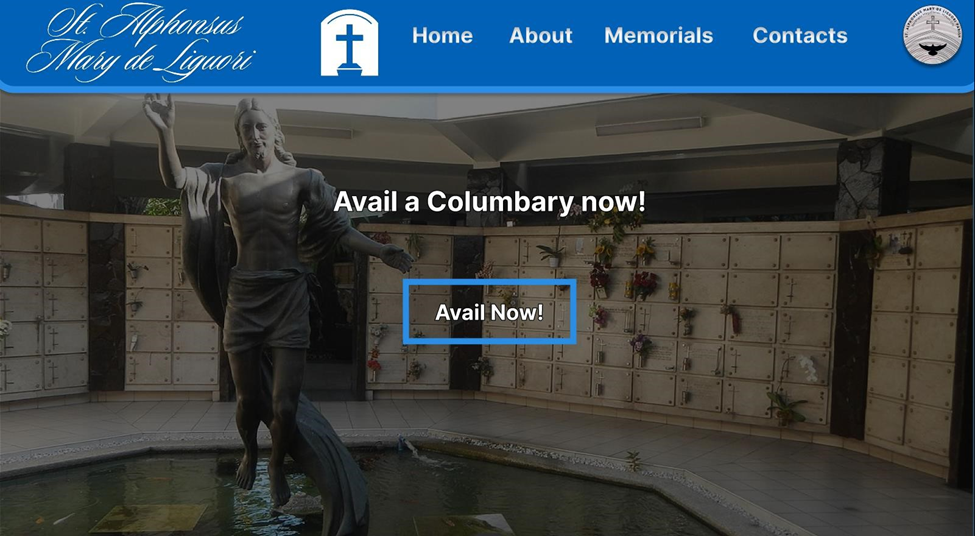


Figure 9: Customer Home Page

Customer home page which will be the first page the customer sees. The customer home page displays all the options to avail a crypt, contact the parish, and see the memorials page.

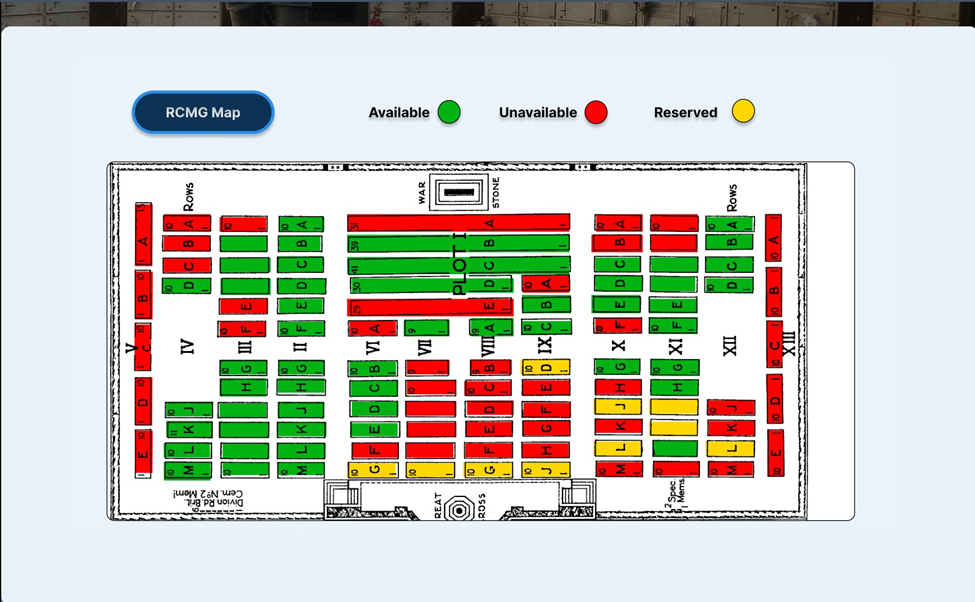


Figure 10: Customer Columbary Map Page

The web interface is designed using vanilla HTML, JavaScript, and CSS. It will be using the Django framework. The following is a prototype of the web application:

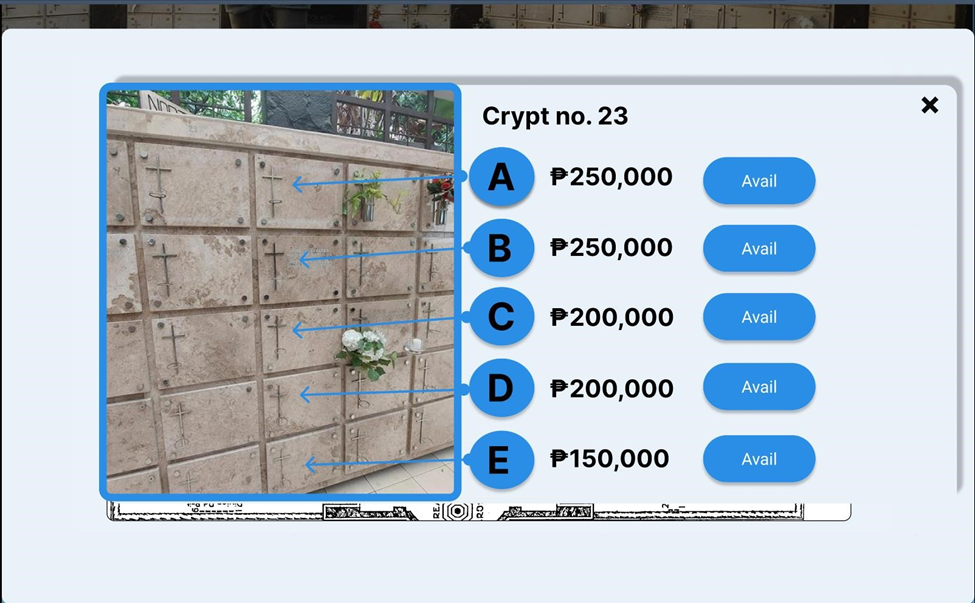


Figure 11: Vault Pricing Page

After choosing an available spot in the columbary map, the customer will be redirected to the crypt prices to continue to availing process.

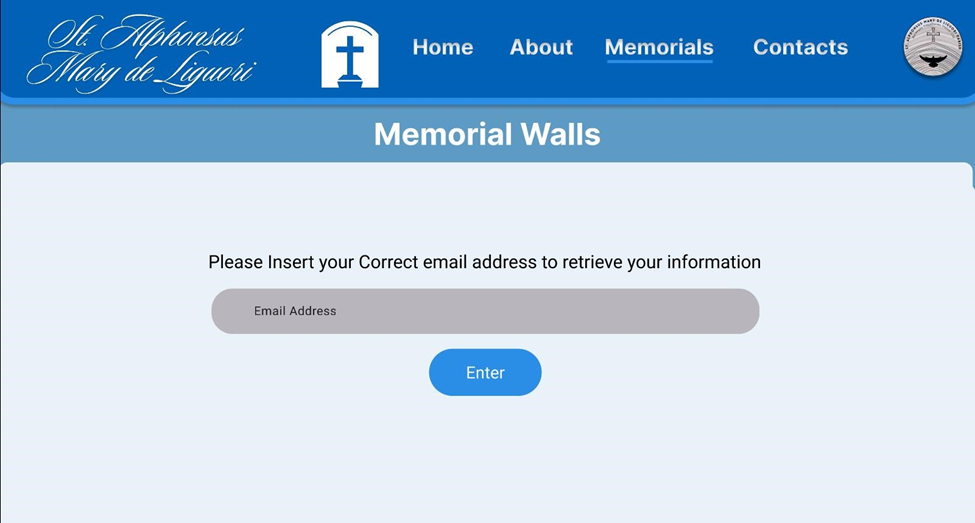


Figure 12: Customer Records Retrieval Page

Customer records retrieval page for the customer to get their records from the parish via online.



Figure 13: Add New Records Page

For the records page, if the Parish Staff wants to add a new columbary record, an add new button is present which then the Parish staff has the option to either manually add files or to use an OCR to scan customer documents.

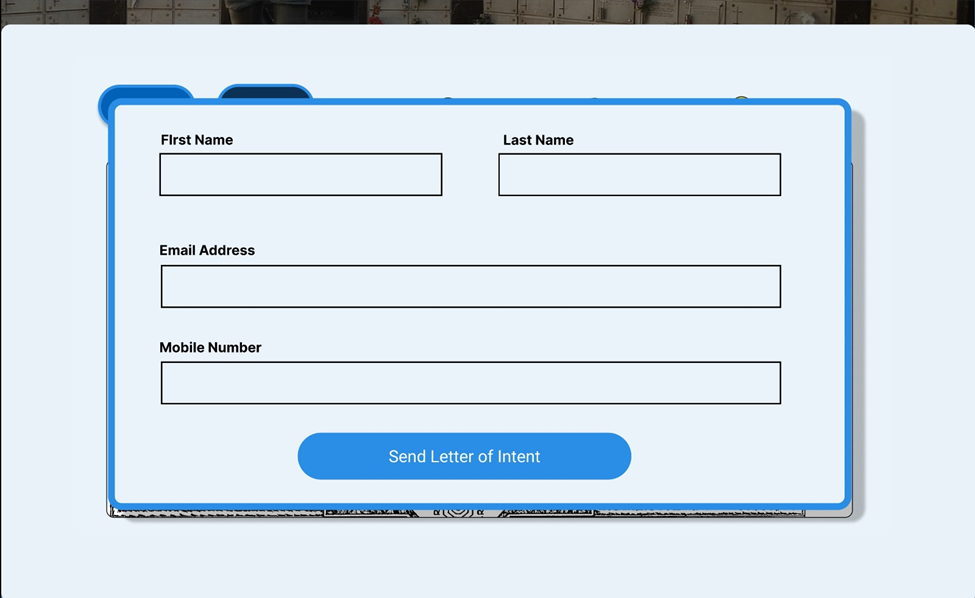


Figure 14: Letter of Intent Application Form

After choosing a crypt the customer tis then met with the letter of intent form, the customer must fill in the Letter of Intent form containing their Full name, Email Address and Mobile Number to finish the availing process.

### 3.2 Hardware Interface

The system will make use of the following hardware components:

* Desktop PC
  + The client will need a PC to run the web application on. This is specifically required by the client for the usage of the system. The whole Desktop PC package will include:
    - Monitor
    - UPS
    - Keyboard
    - Mouse
* Mini server
  + This mini server will be used to run the web application through a local cloud hosting service. The whole mini server will include:
    - Monitor
    - Keyboard
    - Mouse
* Scanner/Printer

1. For the OCR and handling of physical documents, the system will need a scanner for the digitalization of the customer documents for the columbarium.

### Software Interface:

* Database:
  + MySQL Server will be utilized to store all the systems data such as customer records, columbary information, letter of intent and many other more information.
* Operating Systems:
  + Linux server. The purpose of this is to host the application server locally.

## Communication Interface

* Communication Standards
  + HTTPS for all data transmissions.
  + UTF-8-character encoding for all transmitted data.
* Web Browser Interface
  + The system will support modern web browsers when accessing the application.
  + HTTPS/HTTP Protocols are used to ensure security and compatibility.

# 4.System Features

## 4.1 Columbary Reservation System

### 4.1.1 Description and Priority

1. Description: Enables customers to browse, select and initiate reservation of columbary units through a digital interface with interactive mapping and pricing information

* Priority: High

### 4.1.2 Stimulus/Response Sequences

* Stimulus: Customer selects "avail now" button
  + Response: System displays detailed pricing and unit information
* Stimulus: Customer submits letter of intent
  + Response: System validates customer information and sends confirmation
* Functional Requirements:
  + REQ-1: The system shall provide an interactive map interface showing available columbary units
  + REQ-2: The system shall validate customer information before submission
  + REQ-3: The system shall generate and send automated confirmations upon successful submission
  + REQ-4: The system shall notify parish administrator of new letters of intent

## 4.2 Information Retrieval System

### 4.2.1 Description and Priority

* Description: Provides secure access for customers to retrieve their columbary information through verified channels
* Priority: High

### 4.2.2 Stimulus/Response Sequences

* Stimulus: Customer requests information retrieval
  + Response: System prompts for email/phone verification
* Stimulus: Customer submits verification credentials
  + Response: System generates and sends OTP to verified contact
* Functional Requirements:
  + REQ-1: The system shall implement two-factor authentication for information access
  + REQ-2: The system shall verify customer identity before information disclosure
  + REQ-3: The system shall maintain audit logs of all information requests
  + REQ-4: The system shall only send minimal required information via secure channels

## 4.3 Administrative Review System

### 4.3.1 Description and Priority

* Description: Provides parish administrators with tools to review, approve, or deny letters of intent and manage the approval workflow
* Priority: High

### 4.3.2 Stimulus/Response Sequences

* Stimulus: Administrator accesses inquiries tab
  + Response: System displays queue of pending letters of intent
* Stimulus: Administrator reviews and makes decision on letter of intent
  + Response: System processes decision and generates appropriate notifications
* Functional Requirements:
  + REQ-1: The system shall provide a dedicated interface for reviewing letters of intent
  + REQ-2: The system shall automate email notifications based on administrator decisions
  + REQ-3: The system shall maintain complete records of all decisions and communications
  + REQ-4: The system shall forward approved applications to parish staff

## 4.4 Dashboard Management

### 4.4.1 Description and Priority

* Description: Provides comprehensive system monitoring and management interface with AI assistance for staff operations
* Priority: Medium

### 4.4.2 Stimulus/Response Sequences

* Stimulus: Staff accesses dashboard
  + Response: System displays operational overview and status
* Stimulus: Staff activates AI assistant
  + Response: System enables voice command functionality and awaits input
* Functional Requirements:
  + REQ-1: The system shall provide real-time monitoring of columbary availability
  + REQ-2: The system shall integrate AI assistant for staff queries
  + REQ-3: The system shall maintain updated status of all system components
  + REQ-4: The system shall provide access to all management functions through a single interface

**4.5** **Records Management System**

### 4.5.1 Description and Priority

* Description: Enables comprehensive management of customer and columbary records with full CRUD capabilities
* Priority: High

### Stimulus/Response Sequences

* Stimulus: Staff initiates record management action
  + Response: System presents appropriate interface for requested action
* Stimulus: Staff modifies or creates record
  + Response: System validates and saves changes while logging the action
* Functional Requirements:
  + REQ-1: The system shall provide complete CRUD operations for all records
  + REQ-2: The system shall implement data validation for all record changes
  + REQ-3: The system shall maintain audit trails of all record modifications
  + REQ-4: The system shall support document scanning and digital storage

**4.6 Dashboard Management**

4.6.1 Description and Priority

* Description: Manages system access and permissions for parish staff and administrators
* Priority: High

4.6.2 Stimulus/Response Sequences

* Stimulus: Staff manages user accounts
  + Response: System processes account changes and updates permissions
* Stimulus: User requests account modification
  + Response: System validates request and implements approved changes
* Functional Requirements:
  + REQ-1: The system shall support role-based access control
  + REQ-2: The system shall enforce secure password policies

1. REQ-3: The system shall log all account management activities
2. REQ-4: The system shall support multiple permission levels for different user roles

# 5.Other Nonfunctional Requirements

Here we specify some nonfunctional constraints that the program satisfies in order to be more concrete and stable.

## 5.1 Performance Requirements

**Performance:** The client-facing CBAS web system must deliver seamless performance across devices and browsers, ensuring optimal functionality for desktop users.

The system should process queries from authorized users and retrieve columbarium data within a maximum of two seconds under typical usage conditions.

The server must be capable of supporting up to 200 simultaneous active users without performance degradation.

## 5.2 Safety Requirements

The system must be designed to prevent the permanent deletion of sensitive records. Instead, records will be marked as archived or deactivated, allowing for historical traceability.

Automated backups must be conducted daily to ensure data recovery in case of system failure.

Historical logs for all transactions and modifications will be maintained for auditing and troubleshooting purposes.

## 5.3 Security Requirements

The CBAS system must comply with security best practices to protect sensitive columbarium data. All connections to the system will require secure authentication using multi-factor authentication (MFA).

Role-based access controls (RBAC) will be implemented to ensure that users can only access information pertinent to their roles.

All communications between clients and the server must be encrypted using TLS 1.3 or higher.

Intrusion detection and prevention mechanisms will be implemented to safeguard against unauthorized access or malicious activity.

## 5.4 Software Quality Attributes

The CBAS system must be robust and resilient, capable of handling unexpected inputs or errors without crashing or corrupting data.

The system must provide clear error messages and guidance to users in case of invalid inputs or network issues.

While the current deployment environment is a Linux-based server, the system must be platform-agnostic and capable of being deployed on other operating systems (e.g., Windows) to accommodate future infrastructure changes.

The system must be user-friendly, with an intuitive interface to accommodate users who may not have extensive technical expertise.

The software must be maintainable, adhering to standard coding practices and well-documented to enable future updates or modifications by other developers.

## 5.5 Business Rules

Parish Staff can add, update, or delete records for vaults, and customer details.

Parish Staff and Administrators can generate comprehensive reports for columbarium usage, including occupancy and transaction summaries.

Parish Administrator can accept or reject letter of intents.

Customers can receive a letter of acceptance or denial by the parish of their letter of intent.

Customers can receive the status of their columbarium bookings and manage their account details through their email and phone number.

Clients must receive confirmation emails for successful transactions, including payment receipts and reservation details.

Authorized Parish staff can manage reservations, update payment statuses, and access client inquiries.

System logs must capture all administrative actions, including data changes, user logins, and configuration updates, to ensure accountability and traceability.

# 6. Other Requirements

## 6.1 Target Metrics (Objectives)

Upon its implementation, the project must be able to achieve the following metrics:

1. 80% reduction in columbary information retrieval time (from days to minutes)
2. 95% accuracy in automated information delivery
3. Customer wait time reduced from hours to under 5 minutes for basic inquiries.

# Appendix A: Glossary

Here we list all the terminology that is used throughout the document:

**SRS**: **S**oftware **R**equirements Specification

**Connects:**Links this requirement with another

**Includes:**Has the appropriate constraint in it

**Extends:**Shows or cancels a constraint effect if the conditions are met.

# Appendix B: Analysis Models

A diagram of a document

Description automatically generated

Figure 15: DFD Level 1 Diagram

A diagram of a company

Description automatically generated

Figure 16: DFD Level 2-1 Diagram: Send Letter of Intent

A diagram of a computer

Description automatically generated

Figure 17: DFD Level 2-2 Diagram, Review Letter of Intent

A diagram of a account

Description automatically generated

Figure 18: DFD Level 2-3 Diagram, Manage Accounts

A diagram of a data flow

Description automatically generated

Figure 19: DFD Level 2-4 Diagram, Manage Dashboard

A diagram of a process flow

Description automatically generated

Figure 20: DFD Level 2-5, Manage Applications

A diagram of a data flow

Description automatically generated

Figure 21: DFD Level 2-6, Retrieve Columbary Information

A diagram of a company

Description automatically generated

Figure 22: UC-01 Send Letter of Intent

A diagram of a computer flowchart

Description automatically generated

Figure 23: UC-02 Retrieve Columbary Information

A diagram of a flowchart

Description automatically generated

Figure 24: UC-03 Review Letter of Intent

A diagram of a company

Description automatically generated

Figure 25: UC-04 Manage Dashboard

A diagram of a diagram

Description automatically generated

Figure 26: UC-05 Manage Records

A diagram of a company

Description automatically generated

Figure 27: UC-06 Manage Accounts

A computer screen shot of a computer

Description automatically generated

Figure 28: Entity Relationship Diagram

A diagram of a diagram

Description automatically generated

Figure 29: Use Case Diagram

# Appendix C: GitHub Repository

<https://github.com/APC-SoCIT/APC-2024-2025-T2-10-Chambers-of-the-Burning-Ashes-System>