



“Salemthar Information System : A Centralized Church Digital Information System and Platform”

Theme: Urban Governance and Civic technology

Team Name: LCNA

Team ID : HSC | MZ | 120652012

Institution : NIELIT (Aizawl)

State - Mizoram

TABLE OF CONTENTS

Sl. No	Contents
1.	Project Management Plan
2.	Requirements Specification
3.	Analysis
4.	Design
5.	Implementation
6.	Test Documentation
7.	Glossary
9.	Version Index
10.	References
11.	Progress reports
12.	Conclusion

Project Management Plan – Salemthar Information System

a. Project Overview

The Salemthar Information System is a centralized digital platform developed for TKP Salem Unit, Lunglei, Mizoram, under BCM Salem. The purpose of the system is to provide seamless access to church-related updates, sermons, articles, notices, community stories, and gallery posts. By digitizing church information, Salemthar enhances communication, accessibility, transparency, and spiritual engagement among church members.

The project aims to replace scattered manual communication methods with a unified, easy-to-use web-based information system. It ensures that all updates, publications, and resources remain available anytime, anywhere—especially for students, remote users, and members living outside Lunglei. The system also helps archive sermons, reports, and church activities to preserve valuable records for future generations.

b. Project Scope and Objectives

Project Scope

The Salemthar Information System covers the full development life cycle, which includes:

- Gathering needs from church leaders and members
- Designing the website architecture, UI/UX, and content structure
- Implementing core modules such as
 - Latest Articles
 - Sermon Insights
 - Community Stories
 - Events & Notices
 - Gallery / FOD Posts
- Developing an admin panel for content management
- Ensuring mobile responsiveness for easy access
- Creating a digital archive for past sermons, reports, and articles
- Conducting thorough testing and delivering complete documentation

The system will be deployed as a live, fully functional platform accessible to the entire TKP Salem Unit community.

Project Objectives

1. Provide timely, accurate, and organized updates on sermons, events, and community notices.
2. Digitally centralize all articles and publications to reduce dependence on printed materials.

3. Offer a user-friendly platform for members, students, and followers to access church information anytime.
4. Ensure transparent communication within the local church.
5. Create a secure digital archive for sermons, articles, photos, and church reports.
6. Enable mobile and remote access to church information.
7. Support spiritual growth by offering inspiring sermons, stories, and gallery insights.

Team Deliverables

The project team must produce:

- Requirements specification documentation
 - System design diagrams (DFD, ERD, architecture diagrams)
 - A fully functional web application (frontend + backend + database)
 - Admin dashboard for content uploads
 - Testing report and bug fixes
 - Final project documentation
-

c. Assignment of Roles and Responsibilities

1. Project Manager

Responsibilities:

- Oversee entire development
- Coordinate team communication
- Track progress and ensure timely completion
- Review documentation and maintain project schedule

2. System Analyst

Responsibilities:

- Identify user needs (sermons updates, gallery posts, notices, etc.)
- Define system requirements
- Communicate with church representatives
- Prepare SRS documentation

3. UI/UX & System Designer

Responsibilities:

- Design layout for homepage, content sections, and gallery
- Create wireframes, prototypes, and database schema
- Ensure ease of use and accessibility

4. Developer / Programmer

Responsibilities:

- Build the website (HTML, CSS, JS, PHP, MySQL)
- Create modules for Articles, Sermons, Notices, Gallery
- Build admin dashboard
- Ensure responsiveness and performance optimization

5. Tester / QA Engineer

Responsibilities:

- Test all functionalities (content upload, viewing, navigation)
- Ensure mobile compatibility
- Identify bugs and verify fixes
- Validate user experience and reliability

6. Content Manager (Optional)

Responsibilities:

- Upload sermons, articles, notices, and gallery images
- Organize categories and archives
- Ensure content accuracy

d. Project Schedule

The project follows a Waterfall Model, where each phase is completed before moving to the next.

Phase	Activities	Duration	Expected Completion
1. Requirement Analysis	Identify user needs, define modules, draft SRS	1 week	Week 1
2. System Design	UI layout, DFD, ER diagram, database design	1 week	Week 2
3. Development	Build frontend + backend, admin panel, gallery system	2–3 weeks	Week 5
4. Testing	Functionality testing, bug fixes, optimization	1 week	Week 6
5. Documentation	Prepare full project report, screenshots, formatting	1 week	Week 7
6. Deployment	Deploy Salemthar system online	Week 7	Week 7

Project Milestones

- M1: Requirements finalized — End of Week 1
- M2: System Design completed — End of Week 2
- M3: Core Development completed — End of Week 5
- M4: Testing completed — End of Week 6
- M5: Final Project Documentation submitted — End of Week 7
- M6: Live Deployment of Salemthar — Week 7

REQUIREMENT SPECIFICATIONS

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the functional and non-functional requirements of the Salemthar Information System, a centralized digital platform developed for the TKP Salem Unit, Lunglei, under BCM Salem.

The document provides a clear description of system features, constraints, user needs, and operational behavior, ensuring developers and stakeholders have a common understanding of the project.

1.2 Scope

The Salemthar Information System aims to:

- Provide church members with timely updates on sermons, events, and notices.
- Digitally centralize and archive articles, publications, and historical church records.
- Offer a responsive and user-friendly platform accessible via mobile and desktop.
- Allow administrators to upload and manage content through a secure backend.
- Showcase sermons, community stories, events, and gallery photos clearly and attractively.

The system does not include:

- Online payment services
- Social networking features
- Member registration or login (unless added later as a future enhancement)

1.3 Intended Users

- General Church Members – to view sermons, notices, events, and gallery posts.
- Students and Followers – to access archived materials and spiritual resources remotely.
- Church Leaders / Admins – to upload and manage content.
- Developers / Technical Personnel – for maintenance and future updates.

1.4 Definitions

- Article: Written publication such as devotionals, messages, or reports.
- Sermon Insight: Summary or full sermon text provided by church leaders.
- Gallery / FOD: Photo collections of events, programs, and fellowships.
- Admin Panel: Backend interface for content management.

2. Overall Description

2.1 System Overview

The Salemthar system consists of:

1. Frontend Website – For viewing articles, sermons, notices, gallery posts, and events.
2. Admin Dashboard – For uploading and modifying content.
3. Database – For storing texts, images, dates, and archives.

The system ensures transparency, accessibility, and organized digital management of church information.

2.2 System Environment

- Frontend: HTML, CSS, JS, Tailwindcss
- Backend: PHP
- Database: MySQL
- Hosting Environment: Apache / Localhost / Web Server
- Devices Supported: Mobile phones, tablets, laptops, desktops

2.3 System Constraints

- Requires internet for remote access
 - Admin must log in through a secured area (optional future upgrade)
 - Storage limitations depend on database and hosting capacity
 - Content must follow BCM / TKP guidelines
-

3. Functional Requirements

3.1 Homepage Requirements

The system shall:

- Display the latest articles, sermons, and announcements.
- Show a navigation menu for easy movement between sections.
- Highlight recent event photos and community stories.

3.2 Articles Module

The system shall:

- Allow admins to upload articles with title, date, and content.
- Display all articles in chronological order.
- Allow users to read full articles.

3.3 Sermon Insights Module

The system shall:

- Allow admins to upload sermons with scripture references and descriptions.
- Display sermons categorized for easy reading.
- Provide access to recent and archived sermons.

3.4 Events & Notices Module

The system shall:

- Allow admins to publish event details and notices.
- Show upcoming programs and meeting schedules.
- Archive past notices for future reference.

3.5 Gallery / FOD Module

The system shall:

- Allow uploading multiple images per event.
- Display photos in a modern, responsive grid layout.
- Categorize photos by event name and date.

3.6 Admin Panel

The system shall allow authorized admins to:

- Add, edit, delete articles
- Add, edit, delete sermons
- Add, edit, delete notices/events
- Upload and manage gallery images

Admin Panel Functionalities:

1. Article Management
 2. Sermon Management
 3. Notice / Event Management
 4. Gallery Upload (with multiple image support)
 5. Logout (if login module implemented)
-

4. Non-Functional Requirements

4.1 Performance Requirements

- Website must load within 3 seconds on a stable internet connection.
- Database queries must execute efficiently with optimized indexing.
- The system must support at least 5000+ records smoothly.

4.2 Usability Requirements

- Design must be clean, simple, and easy for all age groups.
- Font styles must be readable with appropriate spacing.
- Mobile responsiveness is mandatory.

4.3 Security Requirements

- Admin access must be protected using login credentials.
- Image and document uploads should be sanitized.
- Database must prevent SQL injection.

4.4 Reliability Requirements

- System should work 24/7 without interruption.
- Backup of database should be taken weekly.
- System should recover from errors without data loss.

4.5 Maintainability Requirements

- Code must be modular and commented.
- The system must allow easy updates and new feature integration.

4.6 Portability Requirements

- Should run on any modern browser (Chrome, Edge, Firefox).
- Should run on both Windows and Android devices.

5. Future Enhancements

The system can be expanded with:

- Push notifications for events and notices
- Multi-language support (Mizo/English)

6. Conclusion

This SRS defines the complete requirements of the Salemthar Information System, ensuring a structured approach to development. The system will improve communication, information accessibility, and digital archiving for TKP Salem Unit, supporting both current and future needs of the church community.

ANALYSIS

3.1 Introduction and Problem Description

The TKP Salem Unit is one of the active church units under BCM Salem, Lunglei, Mizoram. The church regularly produces sermons, articles, notices, announcements, photos, and various forms of communication meant for members and the wider Christian community. However, these materials are often scattered across physical records, WhatsApp groups, paper notices, or manually shared documents. As a result, members frequently miss updates, older messages become difficult to retrieve, and there is no centralized digital platform to organize and archive the content.

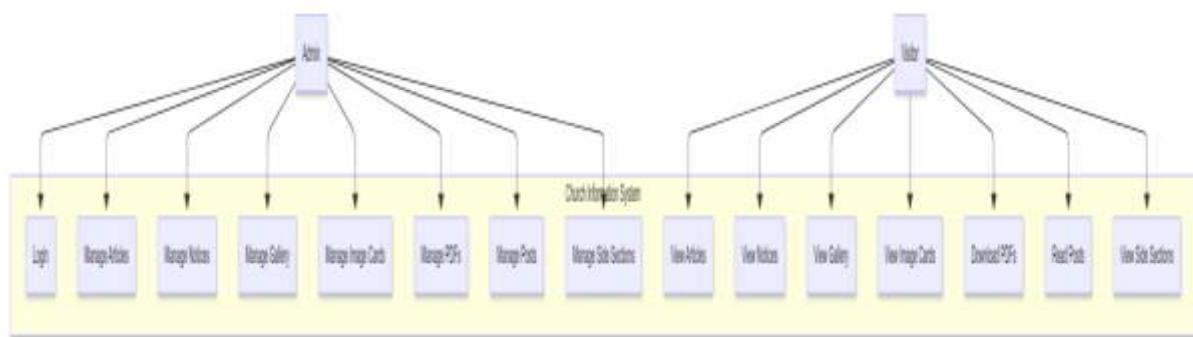
Salemthar, the proposed centralized digital information system, was created to solve these challenges by providing a structured and accessible platform for information distribution. This system digitizes articles, sermons, notices, photos, and other communication materials, enabling the Editorial Team of the church to upload content efficiently while users get easy access through a clean interface. It improves transparency, accessibility, record-keeping, and spiritual engagement.

The system follows a simple admin–user structure:

- Editorial Team (Admin role) uploads and manages all digital content.
- Users/Visitors can view content, read articles, browse galleries, download PDFs, and stay updated.

3.2 Use Case Diagram

- Below is the UML 2.0 use case diagram representing interactions between the actors and the system.



UML 2.0 Use case diagram

Use Case 1: Login

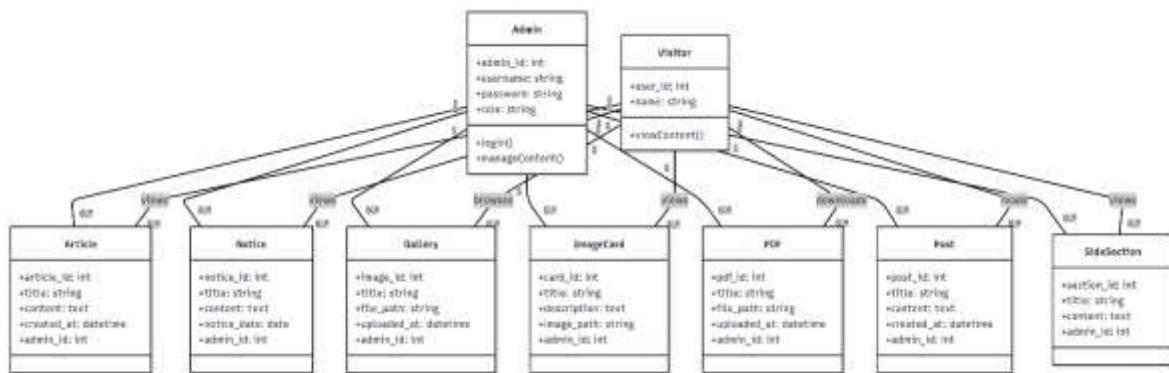
Item	Description
Actor	Admin (Editorial Team)
Description	Admin logs in to access the management dashboard
Precondition	Admin must have an account
Postcondition	Admin gains access to admin panel
Main Flow	Enter username → Enter password → System verifies → Login success

Use Case: Manage Articles

Field	Description
Actor	Admin
Goal	Upload, update, or delete articles
Description	Admin creates new article content for users
Precondition	Admin must be authenticated
Postcondition	Article stored or updated in database

Use Case 2: View Articles

Item	Description
Actor	User
Goal	Read articles and sermons
Description	Users access articles published by the Editorial Team
Precondition	Articles must exist
Postcondition	Article content displayed



UML 2.0 Class diagram

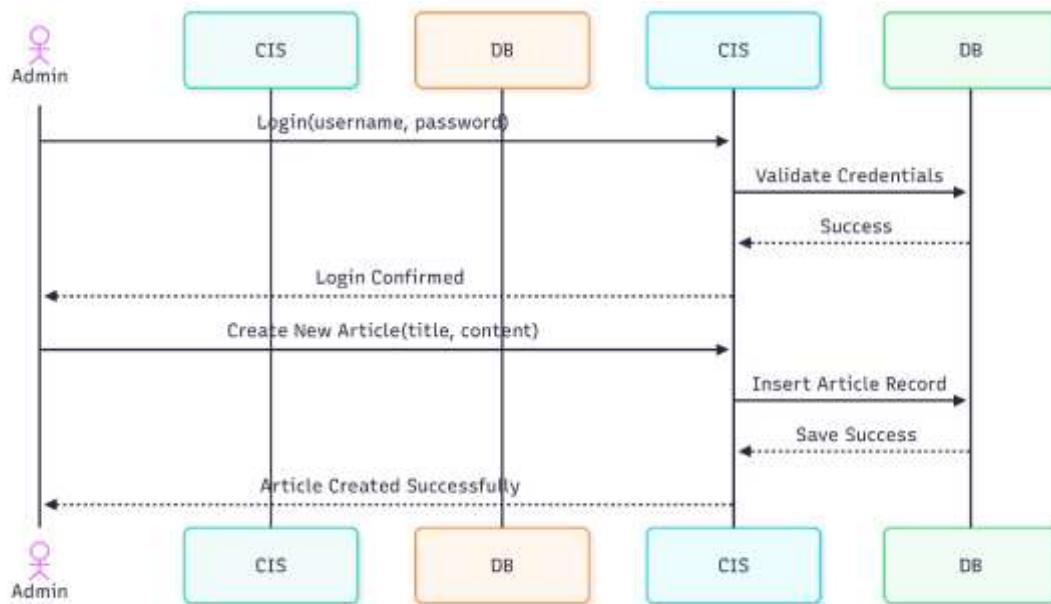


Figure: UML 2.0 Sequence diagram

Entity Relationship Diagram Interpretation

- One Admin can create many Articles, Notices, Posts, PDFs, etc.
- Each content type holds a foreign key referencing the Admin who created it.
- Users only read/view these entities; they do not write into the database.

ER DIAGRAM FOR CIS:

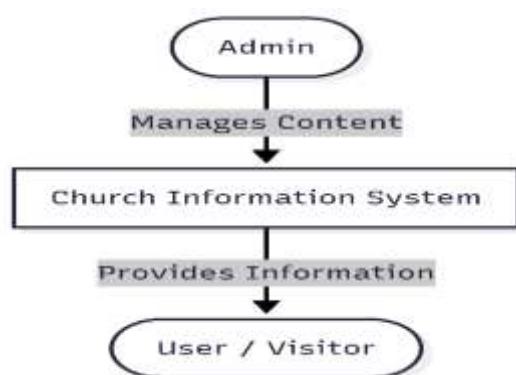
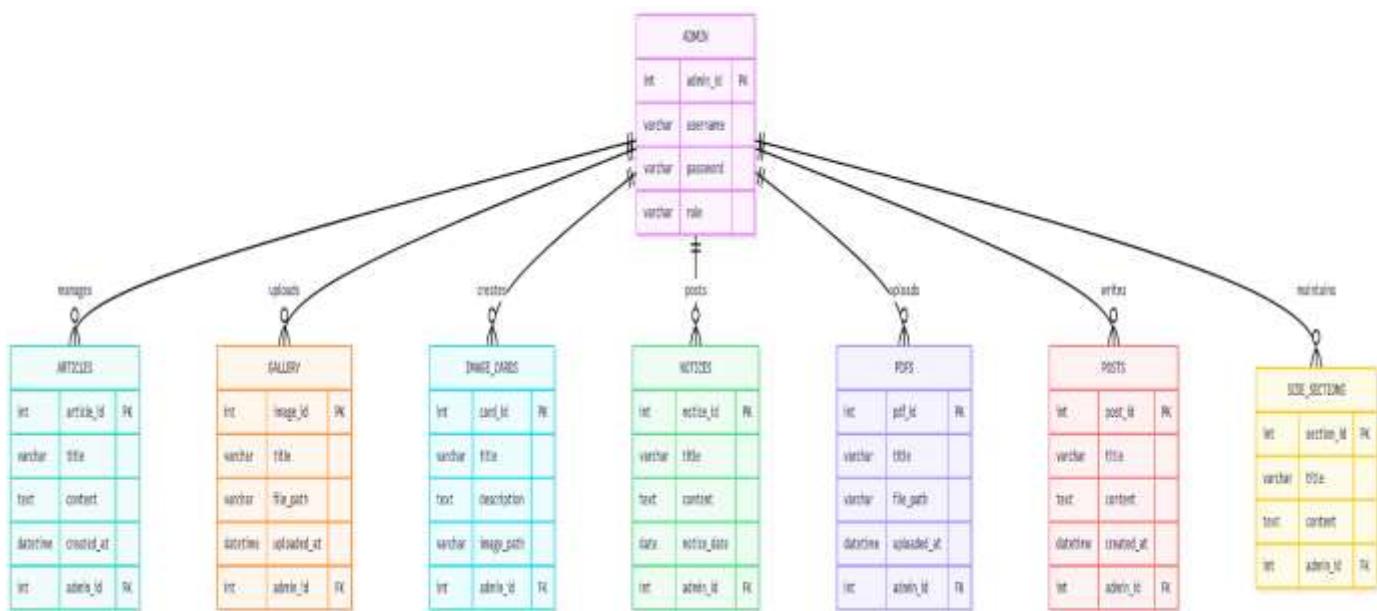


Figure: DFD Level O – Context diagram

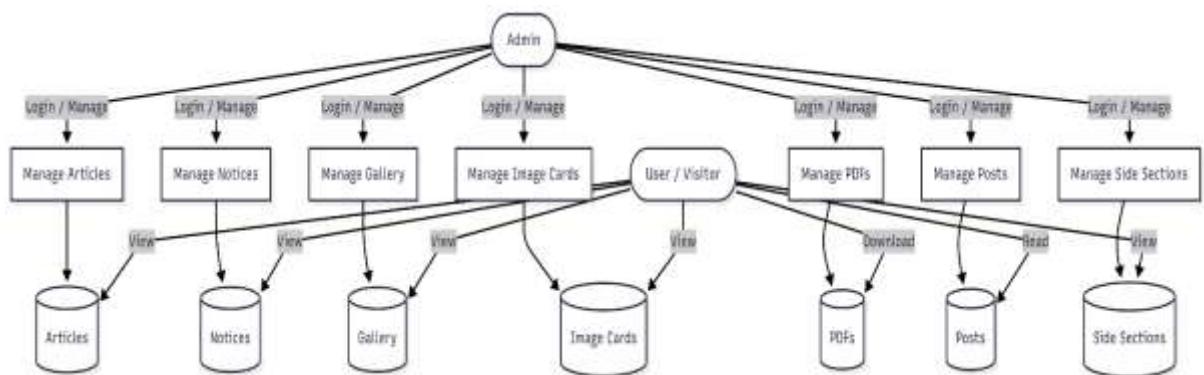


Figure: DFD Level One – Expanded System

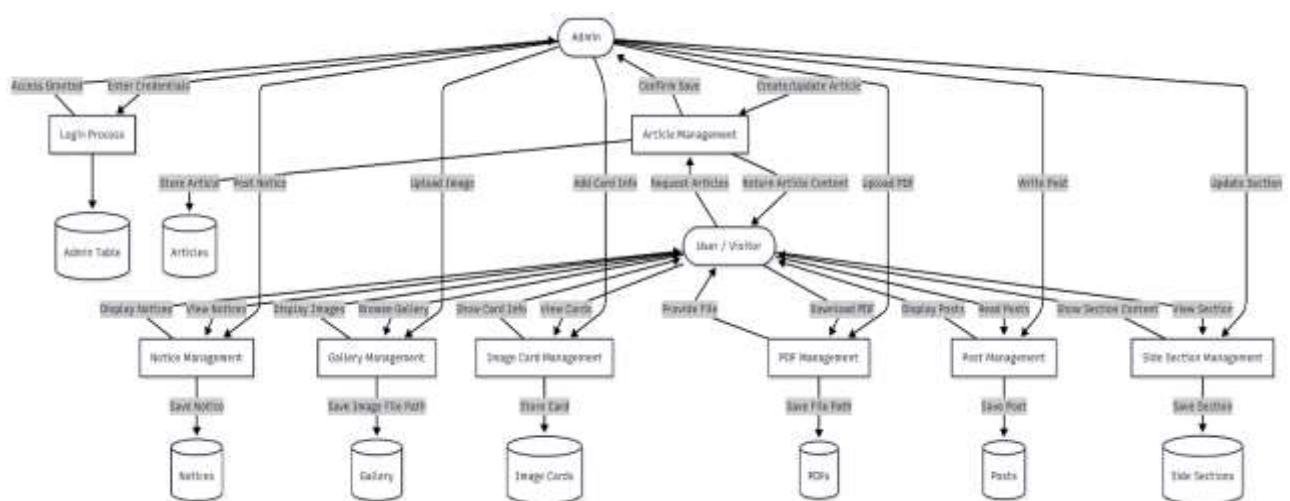


Figure: DFD Level Two – Detailed diagram

DESIGN

User Interface (UI) Design

The user interface of the Salemthar Information System is designed to be clean, organized, and accessible to a wide range of users, including church members, elders, and students. The layout follows a structured hierarchy where content is grouped into meaningful sections to improve readability and navigation. The wireframe provides a clear visual representation of the intended UI and serves as the foundation for the final design.

At the top of the interface, the system displays the Church Logo and a Date & Time picker, which allows users to easily filter or reference content based on specific dates. Below this header area, the layout is divided into two major content columns. On the left side is the Article Section, which is scrollable, allowing users to browse multiple articles without overwhelming the main screen. This ensures accessibility and efficient use of space. On the right side, two important panels—Preacher List and Sunday School Report—present essential information that users may frequently check. These short content cards make it easy to access summarized updates quickly.

The middle part of the interface contains Admin-Based Post Sections, where each admin's posts appear in scrollable containers. This feature is particularly important because different editorial members may handle different types of content. The scrollable functionality ensures that posts do not overcrowd the page, maintaining a balanced and user-friendly layout. At the bottom, a larger Notice/Content Post panel provides space for important announcements or updates that need wider visibility. This ensures that critical church information—such as meeting notices, service changes, or event reminders—is prominently displayed for all visitors.

Overall, the UI design prioritizes clarity, simplicity, and structured content flow. Scrollable panels reduce clutter, while categorized sections make navigation intuitive. The design ensures that church members can easily access sermons, reports, articles, notices, and other important materials, making the Salemthar platform an effective and user-friendly digital information system.

Final Home Design layout

The screenshot shows the homepage of the SALEMTHAR website. At the top center is the church's logo, which is a red circular emblem with the letters 'KTP' in the center, surrounded by text in Burmese and English. Below the logo, the word "SALEMTHAR" is written in large, bold, black capital letters. Underneath "SALEMTHAR" is a horizontal navigation bar containing three items: a date selector ("2025-11-15") with a calendar icon, a green button labeled "Download PDF", and a green button labeled "Articles / Sermon". The main content area is divided into several sections. On the left, there is a section titled "Sermon / Articles" with the sub-section "Tarlanna:" below it. Both sections contain the message "No articles for selected date". To the right of these, there is a section titled "Hriattur pawimawhte:" with a sub-section "Thupuan / Hriattirna / Sawmna" below it. This section also contains the message "No posts for selected date". Further down, there is a section titled "FOD Report / Post" with the message "No FOD Report / Post for selected date". The entire page has a light gray background with white rectangular boxes for each content section.

Article page layout:

CHANCHIN THA HRIL HI RINGTUTE TIH TUR A NI

Posted on: 2025-10-05

Isaia 6:1-10, Marka 16:15-18

Chanchin tha chu Lal Isua Kristaa Pathian hnathawh zawng zawng hi a ni a, Lal Isuan chhandamna hna a thawh chanchin hi a ni kan ti thei bawk ang. Bible chuan kawng hrangin a sawifiah a, Pathian Ram Chanchin Tha te, Chhandamna Chanchin Tha te, Khawngaihna Chanchin Tha te, Remna Chanchin Tha te, Sual ngaihdamna Chanchin Tha te, Tihthianglimna te. Nunna Thu te leh a dangle pawh. Mihringin thinlung taka Isua Krista chu Lal leh Chhandamtua a pawmin piangthar a lo ni a, thil siam thar a lo ni bawk a; a sual a sim a, sual ngaihdamna a chang a, chatuana nunna neiñ a lo awm ta a ni tih hÉ chanchin tha kan hril thin, kan la hril zei tur chu a ni.

1. Tumah kan bång thei lor Kan Thurin No 9-nain ringtute tih tur a sawi zingah, "Khawvel puma Krista ram tizau tura beih' tih hi a tel avangin chanchin tha hril hi ringtu zawng zawng tih tur leh tih makmawh a ni a, tumah kan bång thei lo. Krista chhandamna chu a thlawna kan hmuh avangin a thlawna vçkin kan pek chhuah a ngai a. Nitin thlarau bo tam tak an boral avangin hmanhmawh a ngai bawk. Krista chhandamna chang inti si, chanchin tha hril ngai pawimawh lo chu ringtu tak a ni loveng. Tirikhoh Paula chuan, "Tih mäkmawh ka chunga inngat a ni si; chanchin tha hril suh ila, ka chung a pik dawn si," a ti bawk a. Chhandamna lut zia hretu tan chuan engmah sawi chhawn châkna neih loh chu a inhmeñ lo hië; "Sawi lovin kan awm thei lo," ti ve tur kan ni zawk.

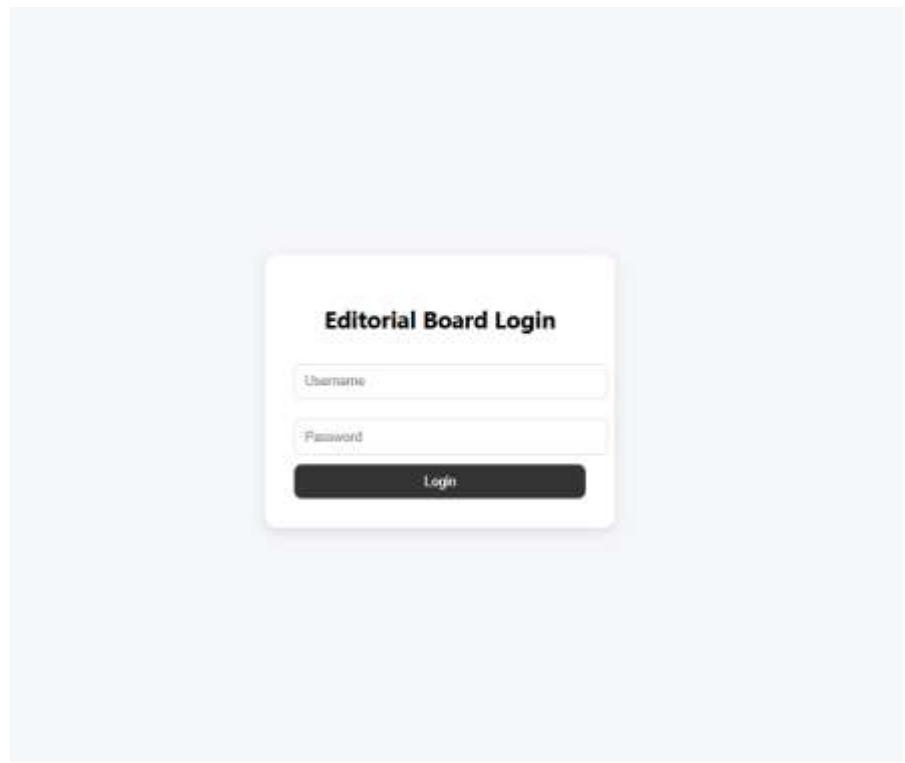
2. Insenso huam a ngai: Lal Isuan, "Ngai teh u, chinghne zinga beram awm angin ka tirh che u hi," a ti a, niuam taka hril tura beisei chi a ni lo. Ringlote hnena chanchin tha hril chu hautak tak a ni. Mahni intum vek a ngai a, mahni intum chung pawhin lawm kan hlawh lova; lawm ahnekin tihduhdah kan tuar zawk thin a ni. Chuvangin, engkim huama thawh chí a ni. Sum leh pa; thei rilru sén a ngai a, hrehawm tuar a ngai a, nunna hial pawh chán a ngaih hun a awm thei. Mahni lam chu insensona tur hril a ni.

3. Thlarau bo pakhat tal man kan tum tur a ni: Mihring hi Pathian anpuia siam kan niñ avangin Pathian mithmuah kan hlu em em a ni. Mi pakhat boral pawh Lalpan a ui em em a, chuvangin, thlarau boral tur chhanchhuak turin ringtu tawh phawt chuan mawphphurhna kan nei a ni. Mahni kan kal chhuak theilo a nih pawhin tawng talin emaw, mite rualin emaw engtin emaw tala kan tan ve a ngai a. Thenawm hnaiah pawh ringlo mi an awm chuan kan hrilh ve thei. Ram hla tak kal kher lo pawhin, kan hmuh phakah leh kan hrilh theiha awm apiang hnenah Isua chanchin tha hi hrilh mawlh mawlh mai tur a ni. Ringtu puitling ve lem chuan thlarau bo pakhat tal man ve tum fiat a tul a ni.

4. Tih theih tåwk tih: Lal Isua, Bethani khua a hmeichhe pakhatin hriak rímtui a thiil khan, "A tih theih tawk a ti a ni," tih a sawi khan, hmeichhe thiil tih chuan a ti lawm a, a chhan pawh a tih theih tawk a tih vang a ni tih kan hria a. Hei hi Isua ringtu apiangin hrerengin, Isua lawm zawng eng noe ni tih kan ngaihtuah a tul. Aman thu min pek kan tihhlawhlii chuan a lawm dawn a ni tih a lang. Hetah hian a hmung mai nilo, theihawp takmeuh chhuah a tul zia a lang bawk. Kan theih chinah chuan tlanchhe lovin kan ti ve mai tur a ni. Mahni kalchhuah emaw, sum leh pala lo tan ve emaw, thildang pawh kan tih ve theih tawk kan tih hi Isua lawm zawng a ni.

[← Let leh na](#)

Editorial Login Page:



Editorial Panel Layout:

A screenshot of the editorial panel layout. At the top, it says "Welcome" and has a "Logout" button. Below that is a section titled "Sermon / Articles" with fields for "Title" and "Content". A timestamp "19-11-2025" is shown. A blue "Add Article" button is visible. The main area displays a table with columns "ID", "Title", "Content", "Date", and "Actions". One row in the table shows the title "Isaia 6:1-10, Marka 15:15-18" and the content: "Chanchin tha chu Lal luuk Kristaa Pathian hnathawni zawng zawng hi a ni a. Lal issan chhandamma hna a thaah chanchin hi a ni kan ti thei break ang. Bihle chuan kawng hreng hrangin a sawfieh a. Pathian Ram Chanchin Tha te, Chandamna Chanchin Tha te, Khawngshna Chanchin Tha te, Remna Chanchin Tha te, Suu ngalhdamma Chanchin Tha te, Tinthianghama te. Nuna Tha te leh a dangle pawh. Misringin thinfung taka luuk Krista chu Lal leh Chhandaantua a paesmin prangthar a lo ni a, thi siem ther a lo ni bawk a; a suul o sm a, suul ngahlema a chang a, chatuana numna nelin a lo zwem ta a ni sh hE chanchin tha kan hil thin, kao la hil zei tur chu a hi..". Below the table is a "Display Mode" section with "Latest Post" and "Update Mode" buttons.

Implementation

This section describes the implementation details of the Salemthar Church Information System, covering coding standards, code documentation practices, validation methods, error-handling routines, and system configuration used during development. It also highlights updates made during the coding process.

a. Coding Standards Employed

To ensure consistency, readability, and maintainability, the following coding standards were used:

Backend (PHP)

- Followed PSR-1 and PSR-12 coding conventions.
- Used clear indentation (4 spaces per level).
- Variable and function names follow camelCase.
- Class names follow PascalCase.
- SQL queries completed using prepared statements for security.
- Modular coding with separate files for configuration, database connection, and individual modules.

Frontend (HTML, CSS, JavaScript)

- HTML5 semantic elements for better structure (<header>, <section>, <nav>, <footer>).
- CSS organized into component-based classes.
- JavaScript functions kept small and purpose-specific.
- External CSS/JS files linked separately for modularity.

Database (MySQL)

- Tables named using uppercase identifiers (e.g., ARTICLES, NOTICES).
- Primary keys use auto-increment integers.
- Foreign key relationships named logically (e.g., admin_id).

b. Comments and Description of Coding Segments

Throughout the system, clear and meaningful comments were added to improve understanding:

Example Segments:

```
// Fetch all articles for the article section
$stmt = $conn->prepare("SELECT title, content, created_at FROM ARTICLES ORDER BY
created_at DESC");
$stmt->execute();
```

Comments explain purpose, logic, and expected values, helping future developers maintain the system easily.

c. Error Handling Mechanisms

The system integrates robust error-handling mechanisms:

Backend Error Handling (PHP)

- Wrapped database operations inside try–catch blocks.
- Used custom error messages instead of exposing server details.
- Logged critical errors into a secure log file.

Database Errors

- Invalid queries prevented via prepared statements.
- Foreign key failures handled with fallback user-friendly messages.

File Upload Errors (Gallery & PDFs)

- Checked file size, extension, and upload path before saving.
- Provided detailed messages such as “Invalid file format” or “Upload failed”.

Front-end Error Handling

- JavaScript alerts for missing inputs.
- Validation errors displayed near form fields.

d. Validation Checks

To ensure integrity and system reliability, multiple layers of validation were implemented:

Client-Side Validation

- HTML5 validation (required, maxlength, pattern).
- JavaScript validation for fields like title length, empty content, date picker, etc.

Server-Side Validation

- Sanitized inputs using mysqli_real_escape_string() or prepared statements.
- Checked maximum file size for PDF and image upload.
- Ensured titles, content, and dates are not empty.
- Verified admin permissions before allowing content uploads.

Database-Level Validation

- Constraints: NOT NULL, PRIMARY KEY, FOREIGN KEY.
- Enforced valid relationships between admin and their uploads.

e. Execution Instructions and Machine/System Configuration Used

System Requirements

- Server: Apache (XAMPP/WAMP/LAMP recommended)
- PHP Version: 7.4 or above
- MySQL Version: 5.7 or above
- Browser: Chrome, Firefox, or Edge

f. Notice of Code Changed / Added

During development, several improvements and additions were made:

Enhancements

- Replaced raw queries with secure prepared statements.
- Added scrollable containers in the Article Section and Post Section.
- Improved the user interface to match the final wireframe layout.
- Added additional tables for:
 - SIDE_SECTIONS
 - IMAGE_CARDS
- Added role-handling logic for future multi-admin support.

Bug Fixes

- Resolved issue where long content overflowed layout sections.
- Fixed PDF upload bug related to incorrect file path handling.
- Corrected date format inconsistencies in Notice module.

New Features Added

- Date & time picker integrated for filtering content.
- Separate admin content categories (Admin 1, Admin 2).
- Dedicated notice panel with increased visibility.
- Improved login authentication with session validation.

TEST DOCUMENTATION

This section describes the testing procedures carried out to ensure the Salemthar Information System meets all functional and non-functional requirements. It includes test plans, test cases, execution logs, expected vs. actual outputs, and final results.

Test Plan

The test plan outlines the overall testing strategy used to validate system behavior. Testing was conducted to verify that:

- All functional requirements (articles, sermons, notices, gallery, user login, admin uploads) operate correctly.
- The system works smoothly across devices (responsiveness).
- Inputs are validated and errors are handled gracefully.
- Uploaded content is stored, retrieved, and displayed correctly.
- Roles and permissions function correctly (Editorial Team, Admin).

Testing Methods Used:

- Unit Testing – Individual functions such as login, upload, view, delete.
- Integration Testing – Verifying modules work together (e.g., upload → database → display).
- System Testing – Testing the complete flow end-to-end.
- User Acceptance Testing (UAT) – Conducted by sample users from TKP Salem Unit.

Test Cases

Sample Test Case Table

Test Case ID	Description	Inputs	Expected Output	Actual Output	Status
TC01	Admin Login	Correct username & password	User redirected to Admin Dashboard	Works as expected	Pass
TC02	Admin Login (Incorrect)	Wrong password	“Invalid credentials” error	Error displayed	Pass
TC03	Upload Article	Title, Category, Body, Image	Article saved and visible in Latest Articles	Visible after refresh	Pass
TC04	Upload sermon	Video link, sermon text	Sermon appears in Sermon Insights	Displays correctly	Pass
TC05	View Notices	None	List of notices displayed	Works correctly	Pass
TC06	Gallery Upload	Event images	Photos stored & displayed	Works	Pass
TC07	Search Articles	Keyword	Matching articles displayed	Works	Pass
TC08	Responsive Layout	Mobile screen size	Layout adjusts to mobile	Works	Pass

Test Execution Logs

Below is an example of a structured test log format used during testing:

Sample Log Entry

- Test Date: 14/11/2025
- Test Case: TC03
- Inputs:
 - Title: "Sunday Message – Faithfulness"
 - Category: Sermon
 - Image: sermon.png
- Expected Output: Article appears in "Latest Articles" list.
- Actual Output: Displayed correctly after page reload.
- Result: PASS
- Remarks: No issues encountered.

Logs were maintained for all features to ensure consistency between expected and actual behavior.

6.4 Test Summary Report

Testing confirmed that:

- All system components operated according to requirements.
- No major functional failures occurred.
- Minor UI inconsistencies were corrected during UAT.
- Performance and responsiveness met the desired standards.

The Salemthar Information System is ready for deployment.

Appendix A: Glossary

This section defines terms and acronyms used in the project report to ensure clarity.

A.1 Glossary of Terms

Term	Definition
System	The complete Salemthar digital information platform.
Failure	The inability of a system function to perform as expected.
Verification	The process of checking whether the system meets specified requirements.
Validation	Ensuring the final product satisfies user needs and expectations.
User Interface (UI)	The visible part of the system that users interact with.
Dashboard	The admin control panel for managing content.
Module	A component or section of the system (Articles, Gallery, Notices).
Content Upload	Process of adding articles, pictures, or sermons to the platform.
Responsive Design	Ability of the system to automatically adjust to all screen sizes.

A.2 Acronyms

Acronym Full Form

DFD Data Flow Diagram

UAT User Acceptance Testing

Acronym Full Form

UI	User Interface
UX	User Experience
CRUD	Create, Read, Update, Delete
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
PHP	Hypertext Preprocessor
DBMS	Database Management System
SQL	Structured Query Language

Appendix B: Source Code

This appendix contains the complete source code used in the development of the Salemthar Information System. The source code follows established coding standards and includes proper indentation, inline comments, and modular structures for readability and maintainability.

Contents of the Source Code Folder

1. Front-End Files

- HTML templates
- CSS stylesheets
- JavaScript files for UI interactions

2. Back-End Files (PHP)

- Authentication module
- Upload and content management scripts
- Sermon, article, gallery, and notices handling logic
- Database connection and configuration file
- Error handlers and validators

3. Database Files

- SQL file for generating the database structure
- Table creation statements
- Sample inserted data (admin/editor accounts)

Appendix C: Version Index

This appendix documents all version numbers and revision dates for major sections of the project report. It tracks changes made over the project duration.

Section	Version	Description of Revisions	Date
Project Proposal	0.5	Initial draft	10-10-2025
Requirements Specification	0.6	Added editorial roles, corrected upload flow	15-10-2025
Design Document	0.7	DFD and wireframe updated	20-10-2025
Implementation Section	0.8	Added coding and validation notes	30-10-2025
Testing Documentation	0.9	Test logs updated after UAT	05-11-2025
Final Report	1.0	Combined all sections and corrected layout	15-11-2025

All future updates should continue from the latest version number.

Appendix D: References

This appendix contains all the sources consulted during the development of the Salemthar Information System, including online resources, documentation, images, and code references.

Books / Documents

- "PHP & MySQL Web Development" – Luke Welling & Laura Thomson
- "System Analysis and Design" – Shelly, Cashman & Rosenblatt

Online Resources

- PHP Official Documentation – php.net
- MySQL Documentation – mysql.com
- W3Schools Web Development Tutorials – w3schools.com
- Bootstrap Documentation – getbootstrap.com
- FontAwesome Icons – fontawesome.com

Images & Graphics

- Icons used in UI obtained from FontAwesome (free license).
- Illustrations/wireframes designed manually by project developers.

All sources have been used only for academic and developmental purposes.

Appendix E: Progress Reports

This appendix shows the development timeline documenting the completion of major tasks.

Date	Task Completed	Remarks
05-10-2025	Requirements gathering	Initial meeting with Salem Unit representatives
10-10-2025	System requirement document	Approved
15-10-2025	Wireframes & DFD	Completed
20-10-2025	Database design	Tables finalized
25-10-2025	Basic UI design	Homepage completed
01-11-2025	Admin panel + Upload module	Working version ready
05-11-2025	Full system integration	All modules connected
10-11-2025	Testing and debugging	Minor UI fixes
12-11-2025	Final documentation	Ready for submission
15-11-2025	Full project submission	Completed

Conclusions

The Salemthar Information System successfully digitizes the publications, sermons, notices, and gallery content of the TKP Salem Unit, offering a centralized and accessible platform for members and followers.

The system helps improve transparency, communication, and engagement within the church community while reducing reliance on print media.

Through structured analysis, careful design, and systematic implementation, the project demonstrates how digital tools can enhance information distribution for local organizations.

Future enhancements such as automated analytics, mobile apps, advanced search, and notification systems can further strengthen the platform. Overall, the project achieves its primary objectives and provides a scalable foundation for future improvements.