**CHAPTER FOUR**

**IMPLEMENTATION, TESTING, AND RESULTS**

**4.1 Introduction**

This chapter presents the implementation of the Computerized Church Database and Communication System, the testing strategies adopted, and the results obtained. The purpose of this phase was to map the logical design developed in Chapter Three onto a physical platform, integrate the different system modules, and test the system for functionality, security, and user satisfaction.

**4.2 Mapping Logical Design onto the Physical Platform (System Development)**

The logical models designed in the previous chapter were translated into a physical implementation using ( tools e.g., PHP, MySQL, and HTML/CSS/JavaScript). The physical design involved:

* Creating the church database structure (members, departments, tithes, offerings, events, communication records).
* Developing the graphical user interface (GUI) for data entry, search, and reporting.
* Implementing communication modules for SMS/email notifications.
* Establishing security features such as login authentication and role-based access.

**4.3 System Modules Implementation**

The system was implemented in modular form to ensure flexibility and ease of testing. Major modules include:

1. **User Interface (UI) Module:** Provides interactive forms for member registration, updating records, and accessing reports.
2. **Database Module (DB):** Stores member information, financial records, and event schedules in a structured format.
3. **Communication Module:** Allows sending of SMS or email alerts to members regarding meetings, contributions, and church programs.
4. **Authentication Module:** Provides secure login, role-based access (e.g., admin, pastor, secretary), and password protection.

**4.4 System Modules Integration**

The individual modules were integrated to form a single, cohesive system. The UI was linked to the database through queries, and the communication module was connected to the database records to automate sending notifications. Integration testing confirmed that data entered via the interface was properly stored in the database and could be retrieved for reporting or communication.

**4.5 Testing**

**4.5.1 Testing Plan**

The system was tested using unit testing, integration testing, and user acceptance testing. The testing plan involved:

* Selecting church staff and members as testers.
* Providing test data (sample member records, contributions, and events).
* Documenting errors and recommendations.

**4.5.2 Verification Testing**

Verification testing ensured that the system was built according to the specifications defined during design. For example, all required fields (Name, Age, Department, Phone Number) were available and functioned correctly.

**4.5.3 Validation Testing**

Validation testing confirmed that the system met the actual needs of the church. Church staff were able to register members, record contributions, generate reports, and send notifications as expected.

**4.5.4 System Security Testing**

The login authentication was tested to ensure only authorized users could access sensitive data. Invalid login attempts were denied, and password encryption was verified.

**4.5.5 Recommendations Made by Testers**

Testers suggested the following improvements:

* Adding a search filter for retrieving member records quickly.
* Including a backup feature for the database.
* Providing an option for bulk SMS to multiple members.

**4.5.6 Responses to Recommendations from Testing**

In response to these recommendations:

* A search filter by name, department, and phone number was implemented.
* A database backup function was added using scheduled exports.
* Bulk SMS functionality was integrated into the communication module.

**4.6 Summary**

This chapter has presented the implementation process, system integration, and testing of the Computerized Church Database and Communication System. The results of testing demonstrated that the system was functional, secure, and aligned with the requirements of the church. Recommendations from testers were addressed to improve system usability and reliability.