**COVER PAGE**

A logo of a university

Description automatically generated

EEC 521/CIS 534: Software Engineering

Professor: Dr. Yongjian Fu

|  |  |  |
| --- | --- | --- |
| **Team Members** | | |
| Team Member 1 | LIKITH REKAPALLI | CSU ID: 2845684 |
| Team Member 2 | KRISHNA SAI SALLA | CSU ID: 2887720 |

**SOFTWARE REQUIREMENTS SPECIFICATION**

**1.0 Introduction**

This document describes the data, functional, and behavioral requirements for the "Health Care Hospital System," a web-based application designed to manage patient and staff information, invoices, and room details. The system will provide an intuitive interface for hospital administrators and staff to efficiently handle critical hospital operations.

**1.1 Goals and Objectives**

The goal of the "Health Care Hospital System" is to streamline hospital management by:

* Managing patient and staff information (registration, search, update, delete).
* Generating and managing invoices for outpatient (OPD) and admitted patients.
* Monitoring room availability and assigning rooms.

The objective is to create a secure, user-friendly, and efficient system to support hospital operations while ensuring data integrity and compliance with healthcare standards.

**1.2 Statement of Scope**

The "Health Care Hospital System" is a web application that facilitates hospital operations. Key functionalities include:

* Inputs: Patient and staff data, invoice data, and room allocation data.
* Processing: CRUD operations on patient and staff data, generating invoices, and managing room availability.
* Outputs: Summary reports on the admin dashboard, updated patient and staff information, generated invoices, and real-time room availability.

The system will be developed using PHP and MySQL for the backend and HTML, CSS, JavaScript, Bootstrap, and AJAX for the front end.

**1.3 Software Context**

The "Health Care Hospital System" is a stand-alone system designed to operate in a hospital environment. It supports the hospital’s business objectives by providing an organized platform to manage patients, staff, and hospital resources. The system will be deployed using local server environments like WAMP or XAMPP, with cross-browser compatibility.

**1.4 Major Constraints**

* Time Constraints: The project must be completed by the end of the academic term.
* Technical Constraints: Compatibility with all major browsers (Opera, Chrome, Mozilla, IE8) is required.
* Resource Constraints: The project has limited access to paid tools for testing and security.

**2.0 Usage Scenario**

The system is primarily used by hospital administrators and staff to perform daily operations. This section details the profiles of the users and presents the different use cases for the system.

**2.1 User Profiles**

The system will have the following user categories:

Administrators: Full access to manage patient, staff, invoices, and room data.

Staff Members: Limited access to patient records and room availability for day-to-day operations.

**2.2 Use-Cases**

* **Admin logs in**: Admin enters credentials and accesses the dashboard.
* **Manage Patients**: Admin/staff can add, edit, delete, or search for patient information.
* **Manage Staff**: Admin can add new staff members, edit their details, and track active staff.
* **Generate Invoices**: Admin/staff can generate, update, and manage invoices for outpatient and admitted patients.
* **Manage Rooms**: Admin can view room availability, assign rooms, and update room status (cleaning schedules, doctor assignments).

**2.3 Special Usage Considerations**

* **Role-Based Access Control**: Certain features (like managing staff data) are accessible only to administrators.
* **Cross-Browser Compatibility**: Special care will be taken to ensure that modern web technologies work across multiple browser versions.

**3.0 Data Model and Description**

This section describes the data structures managed by the software, including data objects, relationships, and a high-level ERD.

**3.1 Data Description**

The software manages the following primary data types:

**3.1.1 Data Objects**

* Patient: Attributes include patient ID, name, age, contact info, history, and room ID.
* Staff: Attributes include staff ID, name, role, department, and shift details.
* Invoice: Attributes include invoice ID, patient ID, services, total amount, and payment status.
* Room: Attributes include room ID, status (available/occupied), assigned patient, and assigned doctor.

**3.1.2 Relationships**

* Patient-Staff Relationship: A patient may be treated by multiple staff members, and staff can manage multiple patients.
* Patient-Invoice Relationship: A patient can have multiple invoices for different treatments.
* Room-Patient Relationship: A room can be assigned to multiple patients over time, but only one patient at a time.

**3.1.3 Complete Data Model**

This section outlines how these objects and their relationships form the overall system structure, ensuring that the hospital management system can effectively manage its operations. The relationships allow for efficient data retrieval and updates across different system modules.

1. **Data Objects and Attributes**
2. **Patient**

* Patient ID: Unique identifier for each patient (Primary Key)
* Name: Full name of the patient
* Age: Age of the patient
* Gender: Gender of the patient
* Contact Information: Phone number or email address
* Medical History: Summary of the patient’s past medical conditions and treatments

1. **Staff**

* Staff ID: Unique identifier for each staff member (Primary Key)
* Name: Full name of the staff member
* Role: Designation/position (e.g., Doctor, Nurse, Admin)
* Department: Department to which the staff member belongs
* Contact Information: Phone number or email address

1. **Invoice**

* Invoice ID: Unique identifier for each invoice (Primary Key)
* Patient ID: References the patient related to this invoice (Foreign Key)
* Invoice Date: Date the invoice was issued
* Charges: Breakdown of charges (e.g., doctor fees, hospital charges, medicines)
* Payment Status: Paid/Unpaid status of the invoice

1. **Room**

* Room ID: Unique identifier for each room (Primary Key)
* Room Number: Number or code assigned to the room
* Doctor Assigned: Reference to the doctor responsible for the room
* Cleaning Schedule: Scheduled cleaning times (Morning and Evening)
* Availability Status: Current status of the room (Available/Occupied)

1. **Relationships Between Data Objects**
   1. **Patient to Invoice (1-to-Many)**

* Relationship: A single patient can have multiple invoices.
* Foreign Key: Patient ID in the Invoice table references Patient ID in the Patient table.
* Example: A patient can receive multiple invoices for various treatments or visits.
  1. **Patient to Room (Many-to-One)**

Relationship: Multiple patients can be assigned to the same room, though not simultaneously.

Foreign Key: Patient details will contain a reference to Room ID, indicating which room the patient was/is in.

* 1. **Staff to Patient (Many-to-Many)**

Relationship: Multiple staff members can manage or treat multiple patients, and each patient can be managed by multiple staff members.

Join Table: A separate table is required to store this many-to-many relationship with the staff information and patient information.

**3.1.4 Data Dictionary**

A comprehensive data dictionary will describe the attributes, types, and constraints of each data object. It will be maintained electronically.

**4.0 Functional Model and Description**

This section describes major functions and software interfaces.

**4.1 Description of Major Functions**

**4.1.1 Requirement 1: Patient Management**

Description: The system must allow CRUD operations on patient data.

**4.1.2 Requirement 2: Staff Management**

Description: The system must allow administrators to manage staff information.

**4.1.3 Requirement 3: Invoice Generation**

Description: The system must allow staff to generate invoices and view billing histories.

**4.1.4 Requirement 4: Room Management**

Description: The system must track room availability and allow room assignments to patients.

**4.2 Software Interface Description**

This section describes how the system interacts with external systems and users.

**4.2.1 External Machine Interfaces**

The system will interface with external printers for printing invoices and reports.

**4.2.2 External System Interfaces**

The system will interface with external email systems for sending notifications about billing or patient updates.

**4.2.3 Human Interface**

The system will provide a user-friendly web interface accessible through modern web browsers. Forms, tables, and dashboard views will be used to interact with the system's functions.

**5.0 Restrictions, Limitations, and Constraints**

* **Security Limitations**: The system must include proper authentication mechanisms to ensure secure access to sensitive data.
* **Browser Compatibility**: The application is designed to be used with modern browsers such as Google Chrome and Firefox but may not fully support older versions of Internet Explorer.
* **Performance**: The system is designed for small to medium hospital environments. As the number of users and data grows, performance issues may arise without optimization.