

WeCare Hospital Management System



**SAN JOSÉ STATE
UNIVERSITY**

Hospital Management System Proposal

Team #19: Nicholas Hsiao, Huynh Phan, Diana Sok

September 5, 2019

Instructor: Dr. Mike-Wu

Table of Contents

Table of Contents

1. Project Overview
 - 1.1. Stakeholders
 - 1.2. Relevance
2. System Environment
3. Functional Requirements
4. Non-Functional Requirements

1. Project Overview

For this database application project, we will be developing a hospital management system with both user and administrative functions. Users (patients) will be able to register as a new user and fill in a medical profile or continue as a registered user and have access to their medical profile and appointments. Users will also be able to schedule and update appointments. Administrators (doctors and nurses) will have read and write access patient medical records and be able to confirm, view and update all upcoming appointments. This project will be a great opportunity to build a real-world full stack application.

1.1 Stakeholders

Table 1. Stakeholders, Interest, and Potential

Stakeholder	Interest	Potential
Patients	<ul style="list-style-type: none">• Difficulty in scheduling appointments• Difficulty in retrieving updated and current medical profile	<ul style="list-style-type: none">• Ease in scheduling• Improved communication• Quick access to up-to-date information• Improved customer experience
Doctors/Nurses	<ul style="list-style-type: none">• Limited time per patient• Delay in information exchange	<ul style="list-style-type: none">• Quick access to a patient's medical history• Ease of access to schedule
Project Team	<ul style="list-style-type: none">• Fulfill product expectations of clients within technical reason	<ul style="list-style-type: none">• Creation of hospital management system

Table 2. Stakeholders, Concerns, and Strategy

Stakeholder	Concerns	Strategy
Patients	<ul style="list-style-type: none">• Scheduling conflicts• Accuracy of medical profile• Data breaches	<ul style="list-style-type: none">• Ensure only authorized accounts can modify data• Prevent overlapping appointments• Defined appointment time slots

Project Proposal

Doctors/Nurses	<ul style="list-style-type: none">• Ensuring patient medical profile cannot be accessed or modified by unauthorized figures• Scheduling conflicts• Data breaches• Legal issues associated with data breaches	<ul style="list-style-type: none">• Ensure only authorized accounts can modify data• Transaction logging• Prevent overlapping appointments• Defined appointment time slots
Project Team	<ul style="list-style-type: none">• Ensuring timely delivery of product• Data breaches• Legal issues associated with data breaches	<ul style="list-style-type: none">• Clear communication with clients of features provided and features out of scope• Constant team communication via frequent updates and reviews• Attempt to provide proper security measures

1.2 Relevance

The value of this hospital management system presents itself in improving the patient experience and lightening the workload of administrators (doctors and nurses). In providing a system in which patients and administrators have quick access to up-to-date data in a *single* destination, both parties communicate better and save time. The appointment-scheduling system to be provided will not only make it easier for patients to receive the appropriate medical attention but will also provide important patient data for use by administrators.

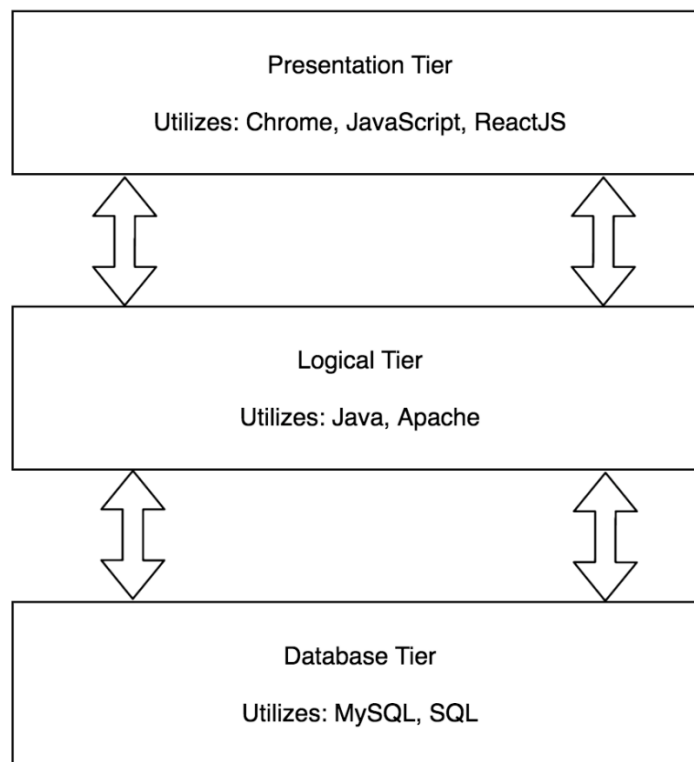
2. System Environment

Our application will follow the three-tier architecture structure. For client browser we will be using Google Chrome. The front-end and user interface will be built using JavaScript and ReactJS as the framework. Our project will use Apache and Java for hosting the web server. Finally, we will be using SQL and the MySQL RDBMS for our database tier.

We will be hosting the Apache server on a desktop PC. The specifications are listed below:

Project Proposal

Component	Specification
Central Processing Unit	AMD Ryzen 5 1400 Quad-Core Processor (8 CPUs), ~3.2 GHz
Memory	12288MB (12GB) DDR4 2400 MHz
Operating System	Windows 10 Home 64-bit
Systems Manufacturer	Micro-Star International Co., Ltd
Download Speed	298.70 Mb/s
Upload Speed	11.87 Mb/s
Ping	10 ms



3. Functional Requirements

1. Patient Functional Requirements
 - a. Create Account
 - i. Application should allow first time users to register an account and fill out their personal information and medical profile
 - b. Log In/Log out
 - i. Application should allow existing users to log in after providing valid credentials
 - c. Access personal information and medical profile

Project Proposal

- i. Users should be able to have read write access to their personal information and medical profile
 - d. Schedule appointments
 - i. Users should be able to schedule appointments with a doctor based on availability
 - e. View/Update appointments
 - i. Users should be able to view their upcoming appointments and make changes as necessary
 - f.
- 2. Administrative (Doctors/Nurses) Functional Requirements
 - a. Log in/Log out
 - i. System administrators should be able to log in to the system after providing valid credentials
 - b. Access patient medical profiles
 - i. Administrators should have read and write access to all patient medical profiles
 - c. View/Update appointments
 - i. Administrators should be able to view all upcoming appointments and make changes

4. Non-Functional Requirements

- 1. Execution qualities (Qualities which are observable during operation)
 - a. Security
 - i. There will be no broken authentication or broken access control points through which admin privileges are given to non-admin users
 - ii. Patients will not be able to access restricted data
 - b. Privacy
 - i. Patients cannot view other patients' data
 - ii. Patients cannot view any of the doctor's private data, such as their patient list
 - c. Performance
 - i. The database program shall execute in timely fashion, returning queries in a reasonable amount of time
 - d. Constancy
 - i. The program will not need to be executed separately or restarted in a single user's session as well as a series of concurrent users of variable permission levels. The program will run in a perpetual state throughout its use.
- 2. Evolution qualities
 - a. Documentation

Project Proposal

- i. The system will feature an organization of code with descriptions such that each component can be easily understood as a constituent of the system
- b. Testability
 - i. The system source code will follow the 3-tier architecture. Using that to the advantage of self-testing, the divided infrastructure will enable the project team to easily identify the point of vulnerability or error as one of the three classifications once error/exception handling is implemented