CSE222/505

DATA STRUCTURES AND ALGORITHMS

GROUP – 14

PROJECT PROPOSAL

Hospital Management System

Contents

1 - Group Members

- Burak Yıldırım – 1901042609
- Ömer Faruk Sayar – 171044038
- Şeyda Nur Demir – 121044042
- Mete Gonca – 161044075
- Merve Dur – 141044022
- Çağla Şahin – 171044050
- Buğra Sabri Tezcan – 171044012
- Cem Bozkurt – 1801042090

2 - Problem Definition

One of the main problems with daily routine of hospital management is scheduling patient appointments. With increasing number of patients and their appointment requests, it is hard to plan these routine.

As a solution to this problem, we are planning to develop a hospital management system. With this management system, patients will be able to schedule appointments and doctors will see these requests. Also, patients can schedule appointments via Registiry Officers, face to face.

Manager is the managing role that has ability to access and edit each entry on the management system.

3 - Users of the System

- Manager

- Doctor

- Receptionist

- Patient

4 - Requirements in details

Functional Requirements:

• Doctors can accept patients, see their appointments, see patients' medical informations, see patients' previous appointments, make appointments to patients, set patients' current status (e.g., discharged, inline, in appointment, admitted, has appointment) and see admitted patients.

• Patients can get an appointment, cancel an appointment, seetheir appointments and see their prescription.

• Receptionists can open accounts for patients, confirm appointments of patients and cancel appointments.

• Manager can add/remove doctors, add/remove admissions clerks, add beds to dorm, see list ofhospital staffs and see the current situation in dorm.

Non-Functional Requirements:

• Scalability: We will use various data structures such as arrays, queues, maps, lists, etc. to hold big data.

• Portability: We will use Java Language, java code can be executed on several platforms, for example, Windows, Linux,Sun Solaris, Mac / OS, etc. The Java code is compiled by the compiler and converted into bytecode. This bytecode is a platform independent code.

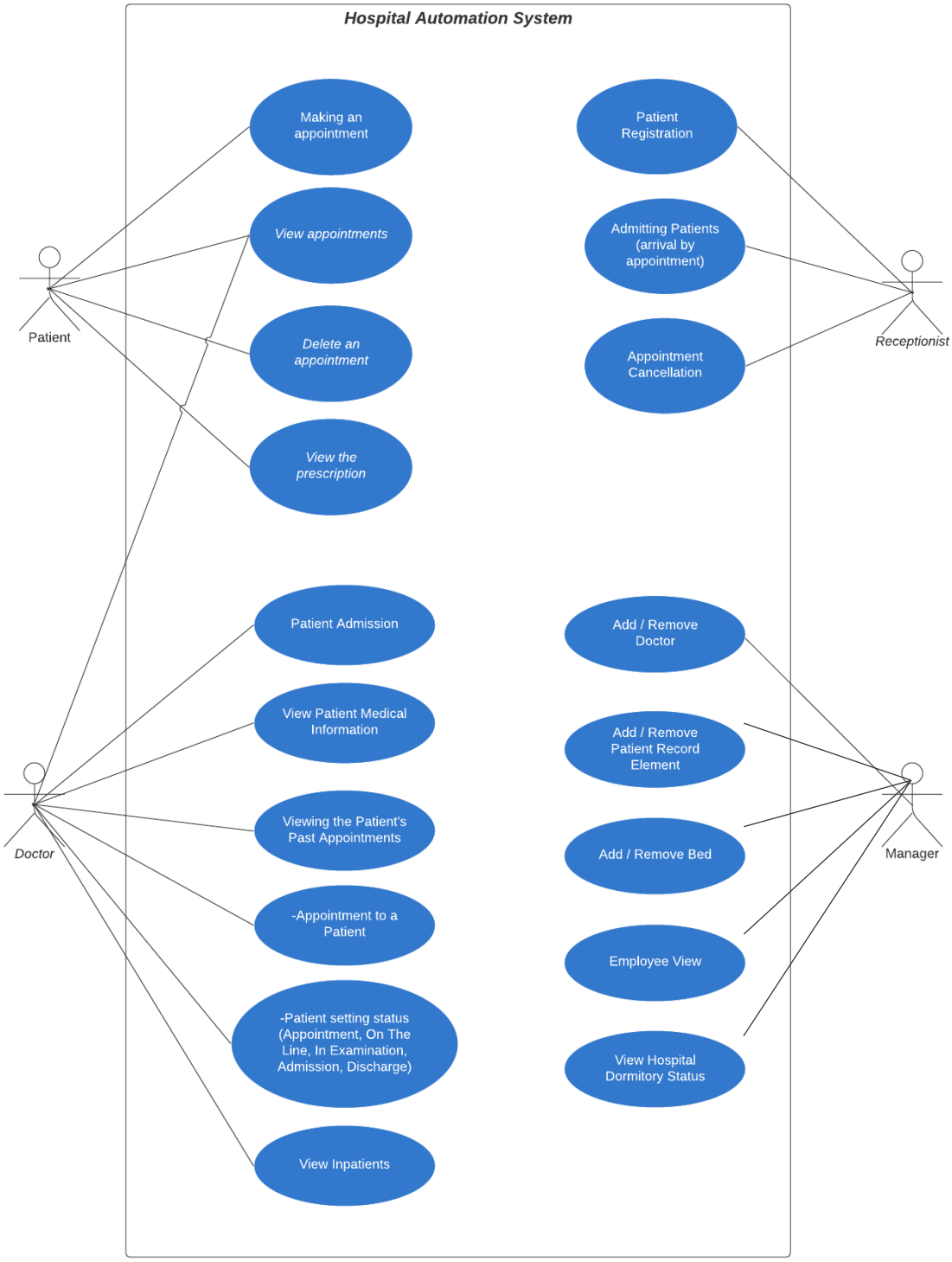
• Security: Each system user will have mail and password to login. We will use hashing to keep the passwords.

• Maintainability: We will follow java code name conventions, we will add javadoc documentations for all classes, methods, variables etc. and we will write tests.

• Flexibility: This system can be used in any hospital.

• Reliablility/Robustness: We will do exception handling gracefully and check unexpected inputs.

5 - Use Case Diagrams

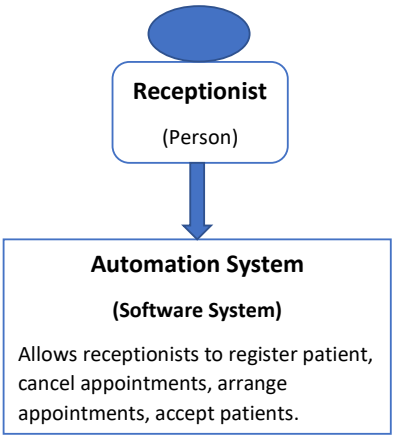


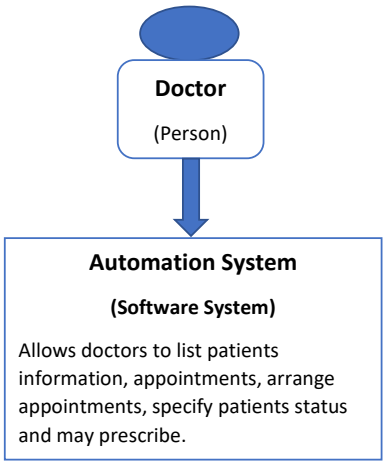
6 - The C4 Model of the System

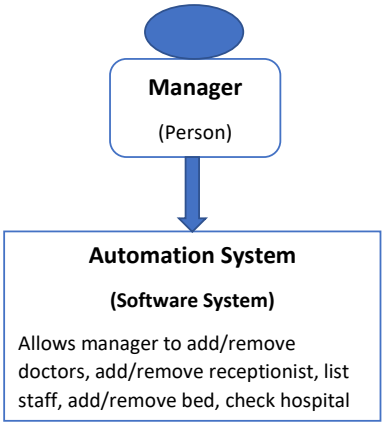
Level 1: System Context Diagram

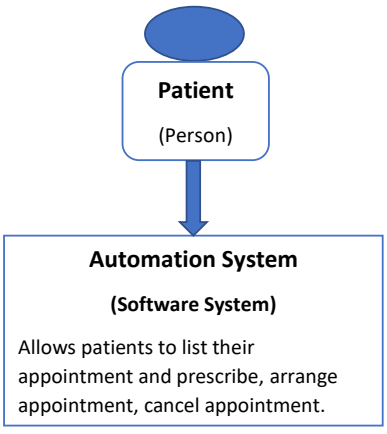
This is a system context diagram for a hospital automation system. It shows the people who use it.

Doctors of the hospital use the automation system list patients information, appointments, arrange appointments, specify patents status and may prescribe. Patients of the hospital list their appointment and prescribe, arrange appointment, cancel appointment. Manager of the hospital adds/removes doctors, adds/removes receptionist, list staff, adds/removes bed, checks hospital. Receptionist of the hospital register patient, cancel appointments, arrange appointments, accept patients.









Level 2: Container Diagram

This is the container diagram for a hospital automation system. It shows the high-level shape of the software architecture and how responsibilities are distributed across it. It also shows the major technology choices and how the containers communicate with one another.

Hospital automation system is a terminal application, uses Java for backend, also stores needed informations in a Json file locally.

If it is needed, this automation system can provide a graphical user interface.