Bilkent University

Department of Computer Engineering

CS 353 Term Project

Project Proposal

Hospital Database Management System

Section 3

Group 29

Project Group Members

1.	Oğuzhan Angın	21501910
2.	Ahmet Furkan Ahi	21501903
3.	Mehmet Alperen Yalçın	21502273

Supervisor: Arif Usta

Table of Contents

1. INTRODUCTION

This report describes various features of a hospital database system. The report contains the general purpose of the project, functional and non-functional requirements, limits and constraints, and finally, technical details and ER diagram regarding the project's programming.

The report begins with a project description part prepared to give an idea to the reader. It briefly explains why such a project is essential and essential for a hospital. How and for what purpose the database can be used is answered.

The report continues with the requirements part. As part of the requirements, functional, non-functional, and pseudo requirements are listed. Functional requirements are necessary to establish the functions, scope, and features of the project. In this subtitle, system and user permissions, ends and capacities are analyzed and clarified. Non-functional requirements include system authentication & security, user-friendliness, reliability, and capacity subheadings. The technologies to be used within the scope of the project are listed in the pseudo requirements section. The limitations and permissions of the system are specified in the limitations section.

After determining the project requirements, the ER diagram of the project was added to the report to explain that it can be changed later. ER diagram is designed to meet the needs and functions of the system.

2. PROJECT DESCRIPTION

Hospital database management system is a web application to observe and manage entities that are part of the hospital environment by application users such as patients, doctors, and laboratorians. The application handles the data such as patients, doctors, laboratorians, tests, test results, interactions between those, and many other data that are part of the hospital environment. A patient can book an appointment with a particular doctor and share his/her symptoms with the doctor using the system. The

doctor can ask for necessary tests after an appointment to examine the disease further and choose appropriate treatment or drugs from the database after diagnosis.

A variety of information needs to be stored and structured in a hospital environment for users to carry out complicated operations safely and efficiently. For instance, a doctor can assign a test after the examination. A laboratorian carries out the given test, which makes him responsible for that particular test result, or a patient can check his/her prescribed drugs using the system. It is hard to handle such massive data that many people will be using every day without an online database system.

As a system with a high amount of essential data, we will distinguish entities and establish relations between them in a precise manner. To protect the integrity of that critical information, we will be setting strict relation constraints. The application user will be able to take high-level actions using the application according to their respective authorization level. Those operations will then be evaluated and carried out as low-level operations such as display, add, delete, and modify on the data stored in the database.

3. REQUIREMENTS

3.1. Functional Requirements

• Persons must be able to log into the system and enter information such as name, surname, age, and telephone.

• These information should be able to be edited.

• A patient should be able to make an appointment with a doctor for a specific date.

• A doctor should be able to look at the appointments he has.

• A doctor should be able to view the patient's personal characteristics.

• An examination should be carried out with the participation of the doctor and the patient on the date determined by the patient at the appointment.

According to an examination result:

• A doctor should be able to order a test from the patient.

• A doctor should be able to diagnose.

• A laboratory worker should be able to perform the tests of patients

• A test can contain many results. The test should have all of these components, and each component should have a normality interval.

• After seeing the results of the tests, the doctor makes a particular diagnosis by choosing the appropriate disease / s from the database.

• A doctor should be able to choose a treatment according to the diagnosis result.

• A doctor should be able to prescribe according to diagnose results.

• Prescription drugs should be changed to other appropriate alternative drugs.

3.2. Non-Functional Requirements

3.2.1. Security & Authentication

• Access permissions should only be changed by the administrator.

• Users should only be able to log in to the system with their own passwords and use the features provided by the system.

• Access to user information should only be done in the manner of the administrator, and the information should not be shared.

• Password quality must be above a certain standard: at least one uppercase letter, one digit, etc.

3.2.2. User-Friendliness

• The appointment system should have a non-confusing and relatively simple interface.

• The patient should be able to get a better user experience by making the necessary filtering while making an appointment

3.2.3. Reliability

Users may knowingly or unknowingly make mistakes while using the system:

• In order that these errors do not prevent the system from not working, the system should never fail. It should continue working in any case and keep the information it contains for the desired period.

3.2.4. Capacity

• Since the system contains many components in a hospital, the provided big data must be properly hosted.

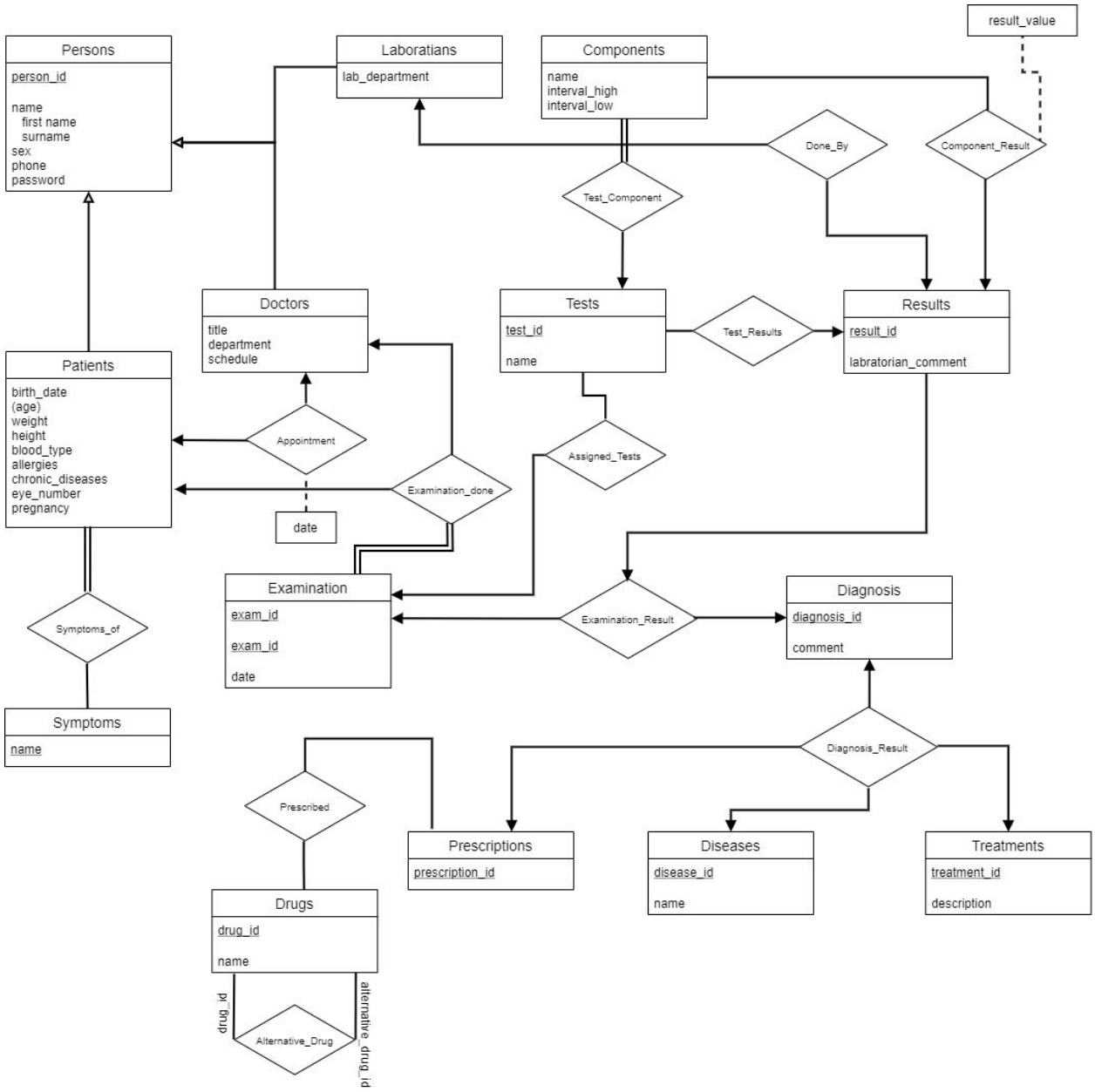
3.3. Constraints

- PostgreSQL will be used as a DBM system for the project.
- HTML, CSS, JavaScript, JQuery, Bootstrap will be used to develop the front-end of the project.
- Django will be used to develop the back-end of the project.

4. LIMITATIONS

- A disease can be diagnosed by only doctors.
- Drugs can be prescribed by only doctors.
- Only laboratorians can do tests.
- Only patients can demand appointments.
- Only doctors can reach patient information.
- Only patients can add allergies and chronic diseases.

5. ENTITY RELATIONSHIP MODEL



6. WEBPAGE ADDRESS

https://alperenya.github.io/CS-353-Project-Group-29/