



Bilkent University
Department of Computer Engineering

CS 353

Database Management Systems Term Project

Project Proposal Patient Medical Treatment Tracking System

Group 7:

Kasymbek Tashbaev 21603136

Babanazar Gutlygeldiyev 21503402

Alina Zhumasheva 21503347

Islomiddin Sodiqov 21503402

Teaching assistant: **Arif Usta**

1. INTRODUCTION	3
2. PROJECT DESCRIPTION	3
3. REQUIREMENTS	5
3.1. Functional Requirements	5
3.1.1 Patient	5
3.1.2 Doctor	5
3.1.3 Pharmacist	6
3.2. Non-Functional Requirements	6
3.2.1 Authentication	6
3.2.2 Reliability	6
3.2.3 Security	6
3.2.4 Responsiveness	6
3.2.5 Performance	7
3.2.6 Capacity	7
3.3. Pseudo Requirements	7
4. LIMITATIONS	7
5. ENTITY RELATIONSHIP DIAGRAM	8
6. CONCLUSION	9
7. WEBSITE	9

1. INTRODUCTION

This is the report for our project about Patient Medical Treatment Tracking System (**PMTTS**). The report contains information about the project description, functional, non-functional, pseudo requirements, limitations and finally, Entity Relational (E/R) diagram of our project.

Project Description section contains information about the main topic of our project and about the properties of the system. Then, report discusses functional, non-functional and pseudo requirements requirements of the Patient Medical Treatment Tracking System.

Functional properties and behaviour of the system will be discussed in a Functional Requirements part of this report. Non-functional requirements section describes general characteristics of the **PMTTS**, such as authentication & security, responsiveness, reliability and performance. Last part of the Requirements section is about the pseudo requirements of the system.

Finally, the report continues with defining the limitations of application system and finishes with providing Entity Relation diagram (E/R), describing conceptual design of the database system for our project.

2. PROJECT DESCRIPTION

PMTTS is a web-based system application for tracking medical treatments of patients at certain hospitals. The system will be designed to be used by patients, doctors of different hospitals and pharmacists. System includes information about the hospitals and doctors working there. Doctors will be able to set or change hospitals where they work, set their schedule of working hours and available slots for appointments. Doctors will have different expertises and can treat different diseases. Patients can book or cancel a medical examination from a certain doctor. After the examination, patients can view their diagnosis and a list of prescribed drugs. Patients then can buy drugs from a pharmacist, and if the drug is not available, patient will be able to buy similar drugs having same ingredients. The security of payment and maintenance of the data are very crucial factors in this database system so that the users will not get into any unwanted situations.

Shortly, **PMTTS** will be a web-application system that will help to ease maintaining the interactions between patients, doctors and pharmacists.

2.1 Why web-application system needs a database.

The database is going to provide us with everything we need to know in order to provide certain functionalities to users of the **PMTTS**. We will use the database system to get the relevant information about the hospitals, doctors, patients, pharmacist, drugs and diseases. We will update the database system by creating new data entries or updating the current ones. The system provides with information about the doctors working in a particular hospital. The database will keep information about the patient and the date of the examination he/she booked from certain doctor, and then information about the diagnosis and drugs prescribed by a doctor to a patient will be updated. Database will have information about the drugs each pharmacist has and will be updated when patients will buy drugs from a particular pharmacist and when pharmacists will have new drugs to sell. The database system will relate the information with the different entities such that the system will work much more efficiently. Since every entity of the system is related with one another, when the system is updated, all of these entities will automatically be updated without much effort as well. These are the reasons why we need to use a database for our system.

2.2 How Do We Use Database in This Project

Database will be the foundation of our system. It will store all the data that is related to **PMTTS**. It will return data from tables with given database queries. Database queries will do operations on the database such as update, insert, join and others. Database will be updated frequently and manipulated to fulfil user's needs. Moreover, it will store and protect private information about users. Database will provide detailed information about all actors and entities taking part in the system.

3. REQUIREMENTS

3.1. Functional Requirements

There are doctors, patients, hospitals, pharmacists, diseases and drugs. A patient can book for a medical examination from a doctor that she/he wishes in a particular hospital. Doctors have certain expertises and treat certain diseases accordingly. After the examination, patient is diagnosed with a disease and is prescribed drug(s). Patients then can buy the prescribed drugs from a particular pharmacist. If a drug is not available, patient should be provided with a similar drug having same active ingredient.

3.1.1 Patient

- Patients can book medical examinations from a doctor that they wish.
- Patients can cancel booked medical examination.
- Patients can get list of hospitals and information about doctors who work in a particular hospital.
- Patients can view result of examination online.
- Patients can get list of pharmacists and drugs that they sell.
- Patients can buy drugs from a particular pharmacist.
- If the drugs that patient wants to purchase are not available in the pharmacy, then other drugs with the same effects should be displayed.

3.1.2 Doctor

- Doctors can set their working graphic.
- Doctors can change their specialisations.
- Doctors can set or change the hospital that they work at.
- Doctors can discover new diseases.

- Doctors can change the description of diseases.

3.1.3 Pharmacist

- Pharmacists can create new drugs.
- Pharmacists can change the description of drugs.
- Pharmacists can add or remove drugs that they offer.
- Pharmacists can change or set the price and discount of drug they sell.

3.2. Non-Functional Requirements

3.2.1 Authentication

- Access permission can be changed only by the system administrator.
- If there are more than three failed attempts to login, **PMTTS** will apply an extra restriction/requirement to the user e.g. two factor authentication.

3.2.2 Reliability

- Since **PMTTS** is the system that works on the lives of people, it should be reliable. In other words, it must have almost no down time.
- It should be functioning 24/7 except scheduled maintenance time.
- During maintenance no data loss is tolerable. So it must be backed up accurately.

3.2.3 Security

- To protect the privacy of users, the login process will require the use of a username and a password.
- If there are more than 100 failed attempts to login from the same user within an hour, **PMTTS** will block “this user” from **PMTTS** for 24 hours.

3.2.4 Responsiveness

- Since **PMTTS** is a system that is going to be used in critical moments, its responsiveness must

be as high as possible. In other words, it should respond to user input in maximum one second.

- **PMTTS** will open and get ready within 15 seconds, once computer is ready.

3.2.5 Performance

- For **PMTTS**, it is crucially important to respond rapidly and accurately. It will perform its operations like search, query, sort etc., in less than 15 seconds.

3.2.6 Capacity

- Since the system is huge, **PMTTS** will be need to sort vast amount data such as patients', doctors', hospitals' and pharmacists' data. Additionally, it will keep drug list with required information. Thus, it needs to have enough capacity of resources e.g. performance and memory capacity.

3.3. Pseudo Requirements

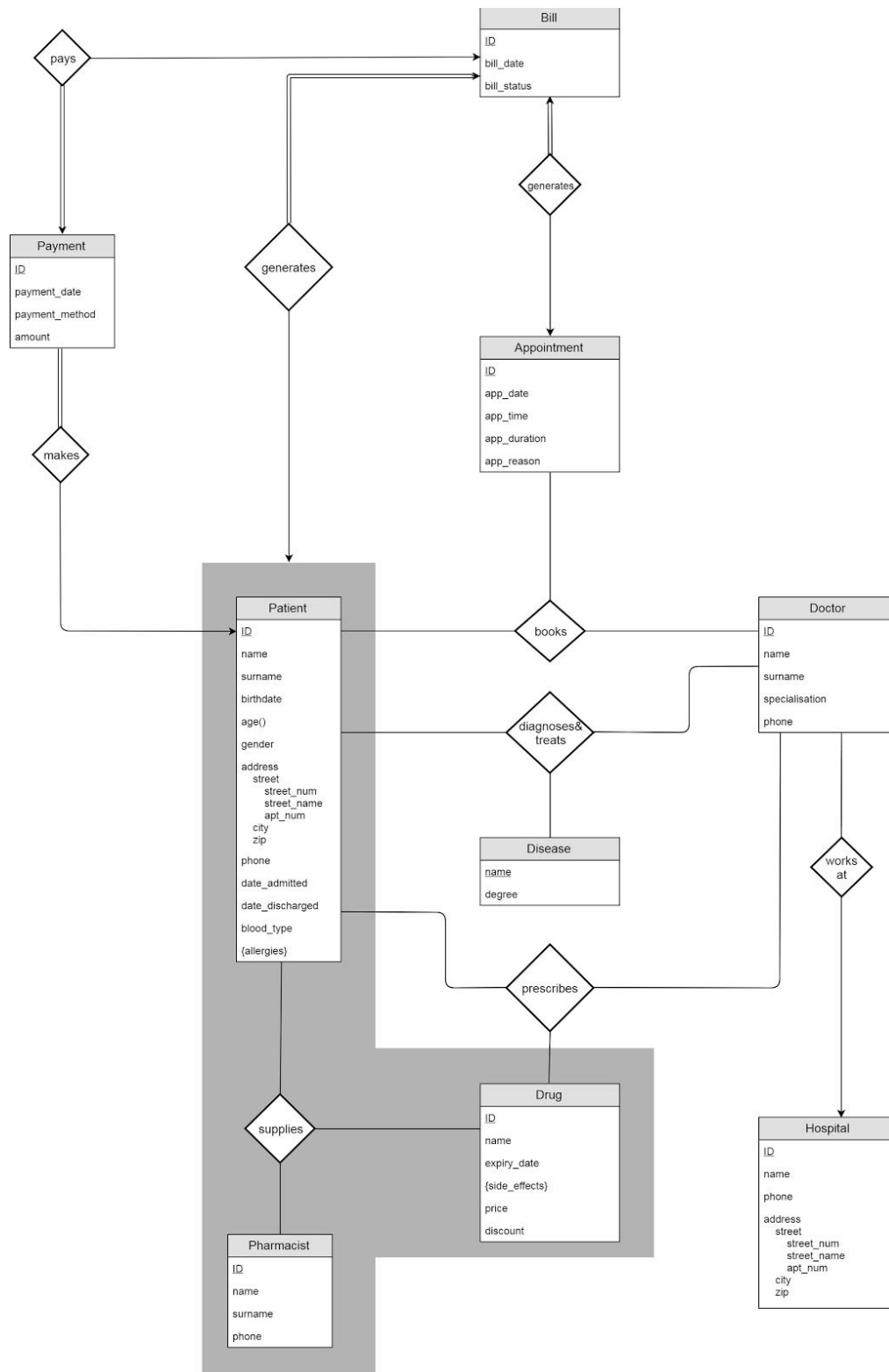
- In front end HTML/CSS, Bootstrap, Javascript will be used.
- In back end Python - with Django framework will be used.
- MySQL will be used for the database system.

4. LIMITATIONS

- Patients can book many many appointments from different doctors. For each appointment bill will be generated for patients to make payment.
- Patients can be prescribed with different diagnosis by doctors and doctors can treat different diseases of patients.
- Each doctor can work only in one hospital.
- Doctors can prescribe many drugs to their patients.
- Pharmacists can supply many different patients with as many drugs as they please.

5. ENTITY RELATIONSHIP DIAGRAM

The provided figure is the entity-relationship diagram of our database.



6. CONCLUSION

Patient Medical Treatment Tracking System will be a web application to track treatments of the patients by doctors, designed to be used by patients and doctors in hospitals and pharmacists. Moreover it will help pharmacists to supply required medicine to patients. The report proposes basic properties and functionalities of such a system by providing functional, non-functional requirements and its limitations. Entity Relational (ER) diagram is used to visualize the high-level design of the system.

7. WEBSITE

Our project website is: <https://babanazar.github.io/CS353-PatientMedicalTreatmentTrackingSystem/>