IT 314 SOFTWARE ENGINEERING

LAB 6 : DOMAIN ANALYSIS MODELLING

HEALTH CENTRE MANAGEMENT

GROUP: 12

Domain analysis modelling

Boundary, Entity and control objects

Boundary objects:

- Authentication interface

  It is located at the boundary of the system where the user interacts with the system and uses it to login or signup.

- Payment gateway and interface

  It is used by the user to make payments for booking appointments/lab tests or ordering medicines.

- Android interfaces

  These interfaces allow the user to interact with the system while performing the desired functionality.

Entity objects:

- Patient profile and medical history

  It is an entity object as it contains data about real world objects i.e personal and medical details of the patient and is present at the end of the path from the user through boundary and control objects.

- Medical report

  It contains the data of the result of the lab test performed on a patient by a lab assistant.

- Appointments

  It contains the information about the scheduled consultation of a user with a doctor with the exact date and time.

- Medicine order

  It contains the information of the medicines ordered by the user with the bill date and amount.

- Doctor's schedule

  It contains all the information about all the appointments of a doctor within a specific time period.

- Payment history

  It contains the data about all the payments made by a user in the past.

- Notifications

  It contains information about some reminders for the user. Notifications are sent for all important activities like

reminding about upcoming appointments, medicine deliveries, lab reports and many more.

.

- Prescription

  It contains data generated by the doctor for a patient about the medicines that the patient must take and the lab reports that he/she must perform after consulting him/her.

Control objects:

- Patient Data Manager

  It is a control object as it controls the data flow between boundary interfaces and their respective entity objects related to patients' medical and personal data.

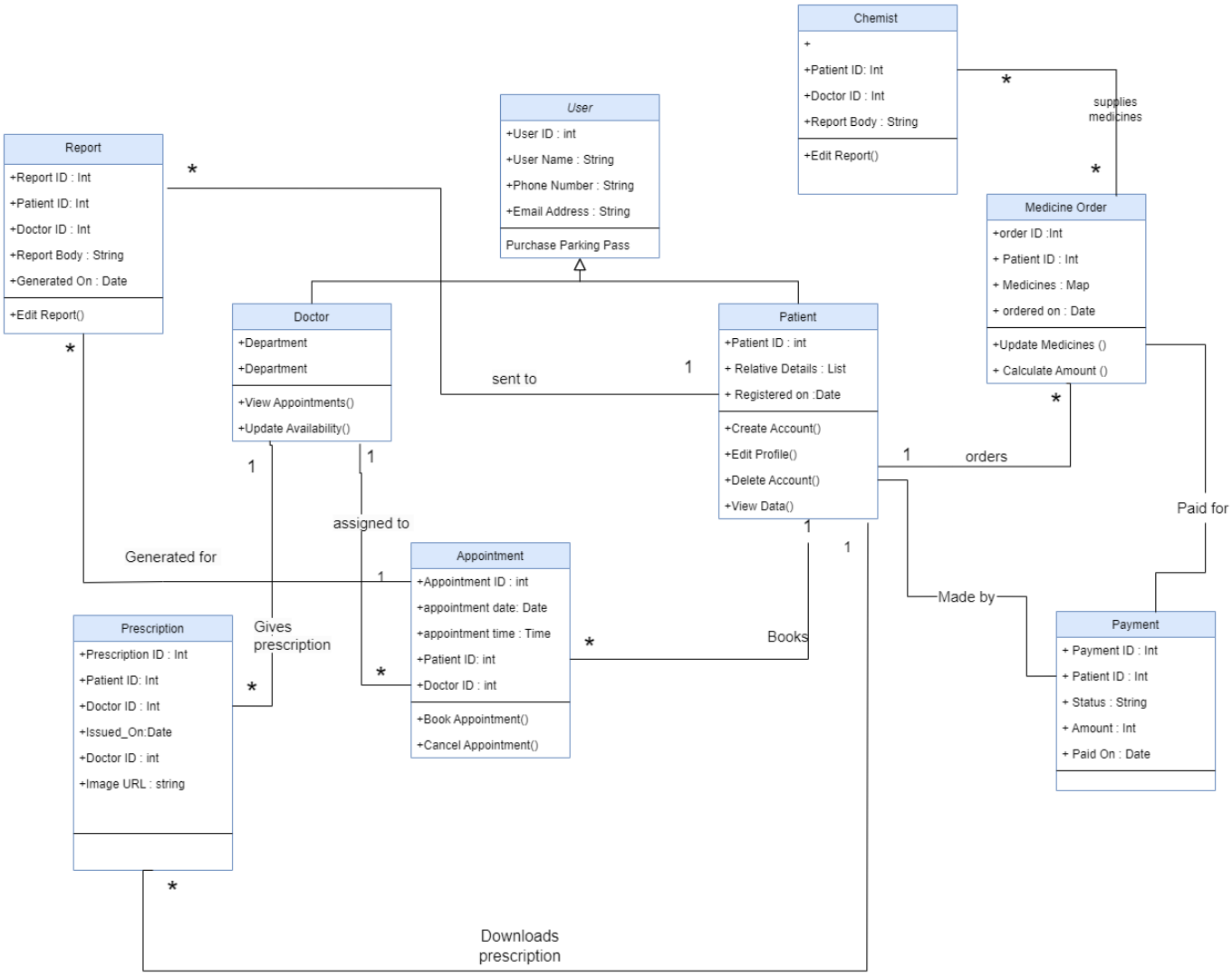
- Appointments interface

  It controls appointments booking by users, and assigns it to doctors and also manages any conflicts between schedules, performs rescheduling and so on.

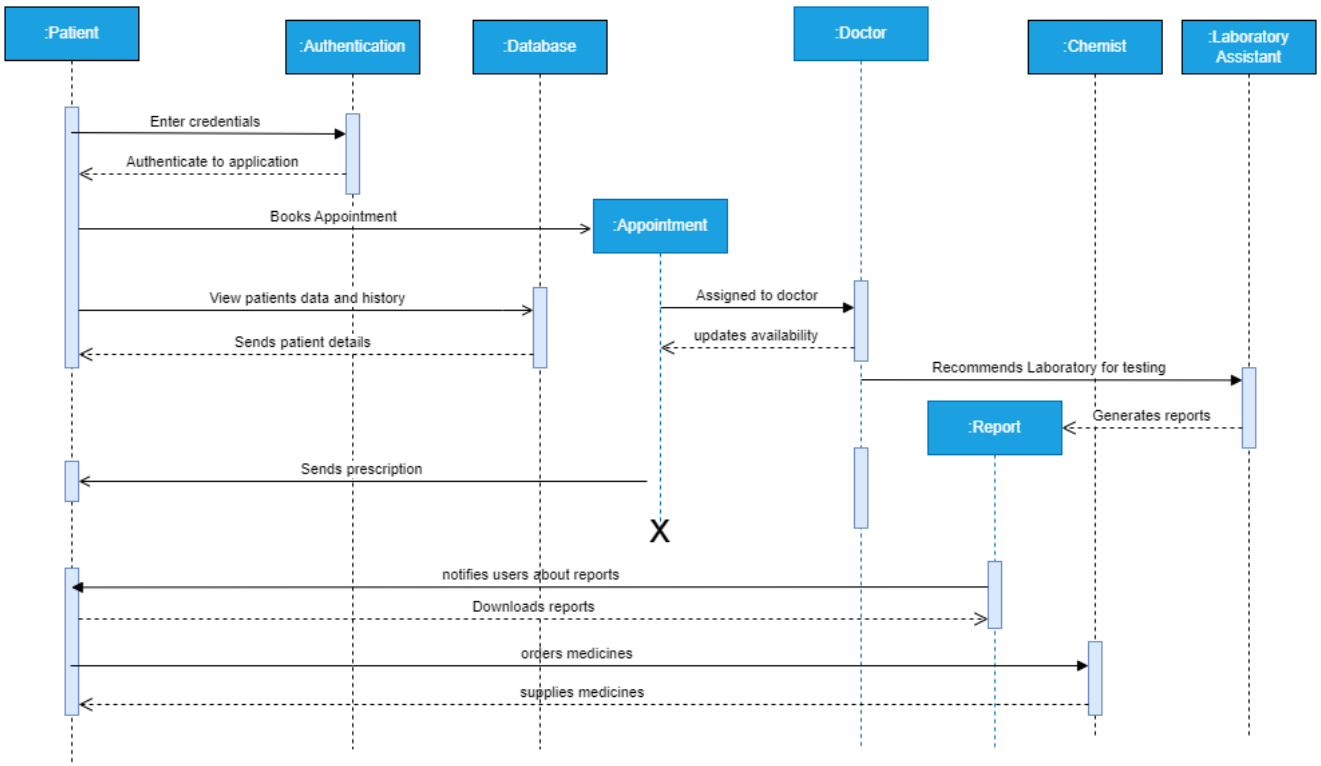
- Payment methods

  It is a control object as it directs a user from either starting to book an appointment/lab test to successfully booking it or searching for medicines from the chemist and placing an order.

Class Diagram



Sequence Diagram



Design goals

1) Reliability:

Reliability is a major concern as low reliability leads to higher risk of errors and failures which potentially harm the patient using the system.

2) Maintainability:

The system should be easy to maintain and manage as it should provide services all around the clock to ensure that emergency services will be serviced.

3) Efficiency

There should not be any delays in providing services to patients as it might lead to poor user experience and patients may dislike the system.

4) User-friendliness

Most patients using the application may not be professional users and can find it difficult if the application is not user friendly.

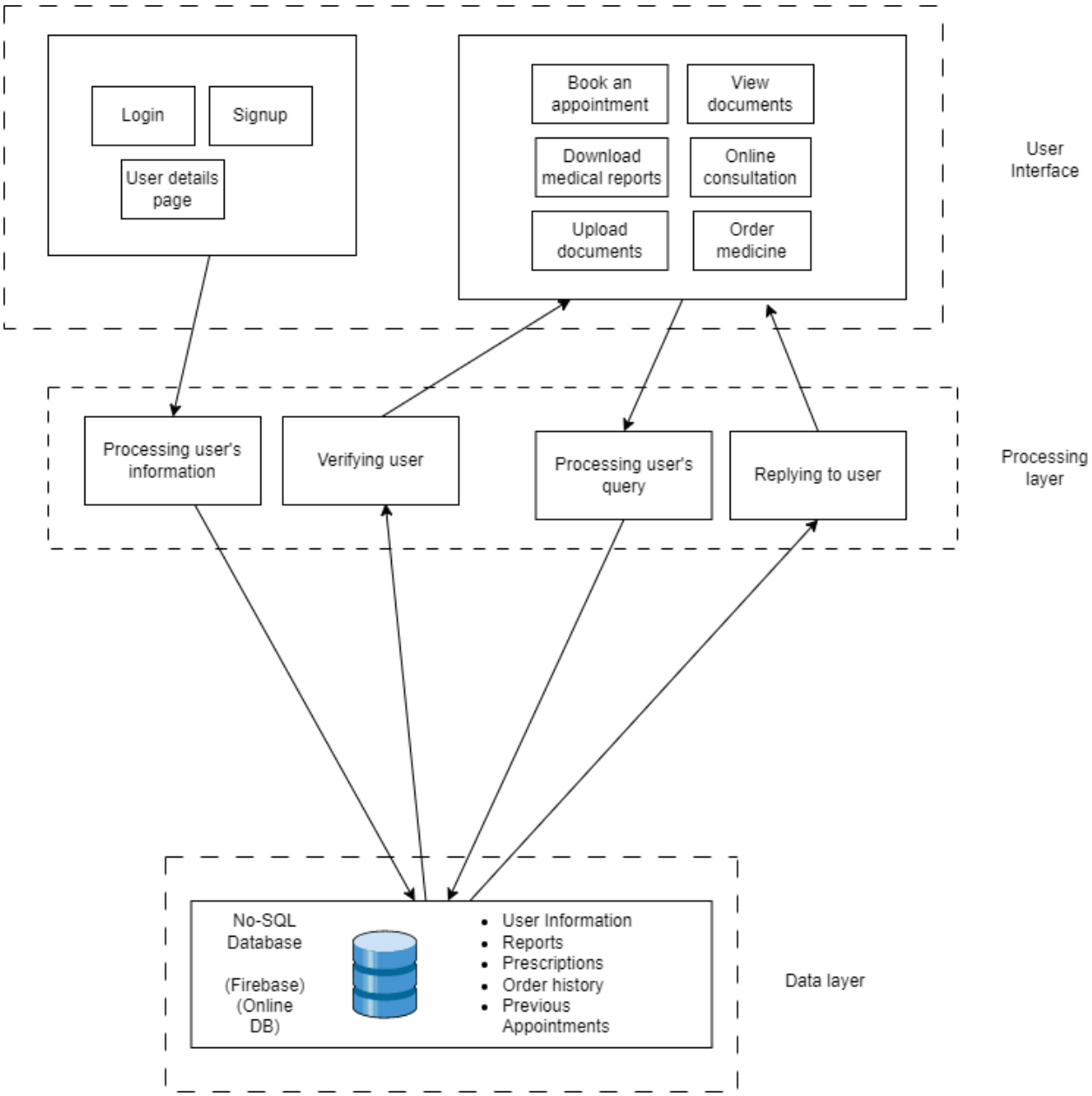
5) Flexibility

The system should be flexible enough to adapt to new changes in future as there will always be some evolving requirements by the users.

6) Fault tolerance

The system should be resistant to any potential failures and in case of failures it needs to be recovered using backups and other methods to ensure that patients data is not lost.

High level system design



Architecture : Layered Architecture

We have used layered architecture consisting of 3 layers.

- The upper layer is the user interface using which the users interact with frontend of the system. This is the only layer that is exposed to users.
- The middle layer is the processing layer which is the backend of the application which processes user queries and sends appropriate responses.
- The lower layer is the data layer which stores the data of the system. It is a No-SQL database provided by firebase used for storing and querying data.