Patient Management System

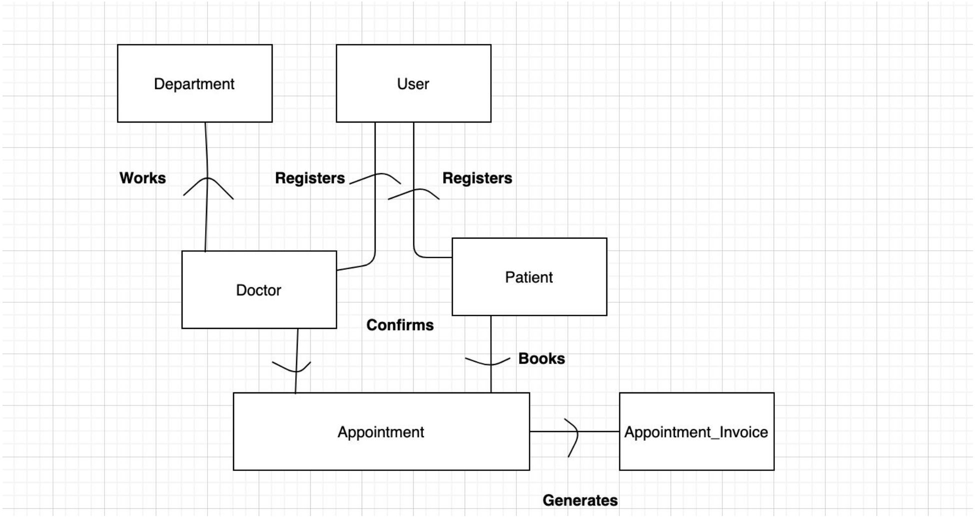
What is the project?

In this project, a user can register as a patient and book appointments with the doctors. A user has the flexibility to create, update, delete, and read appointments. Moreover, the user can also create, update, read, and delete his profile in the system. We offer the user a search menu to search through the database for finding his required doctor.

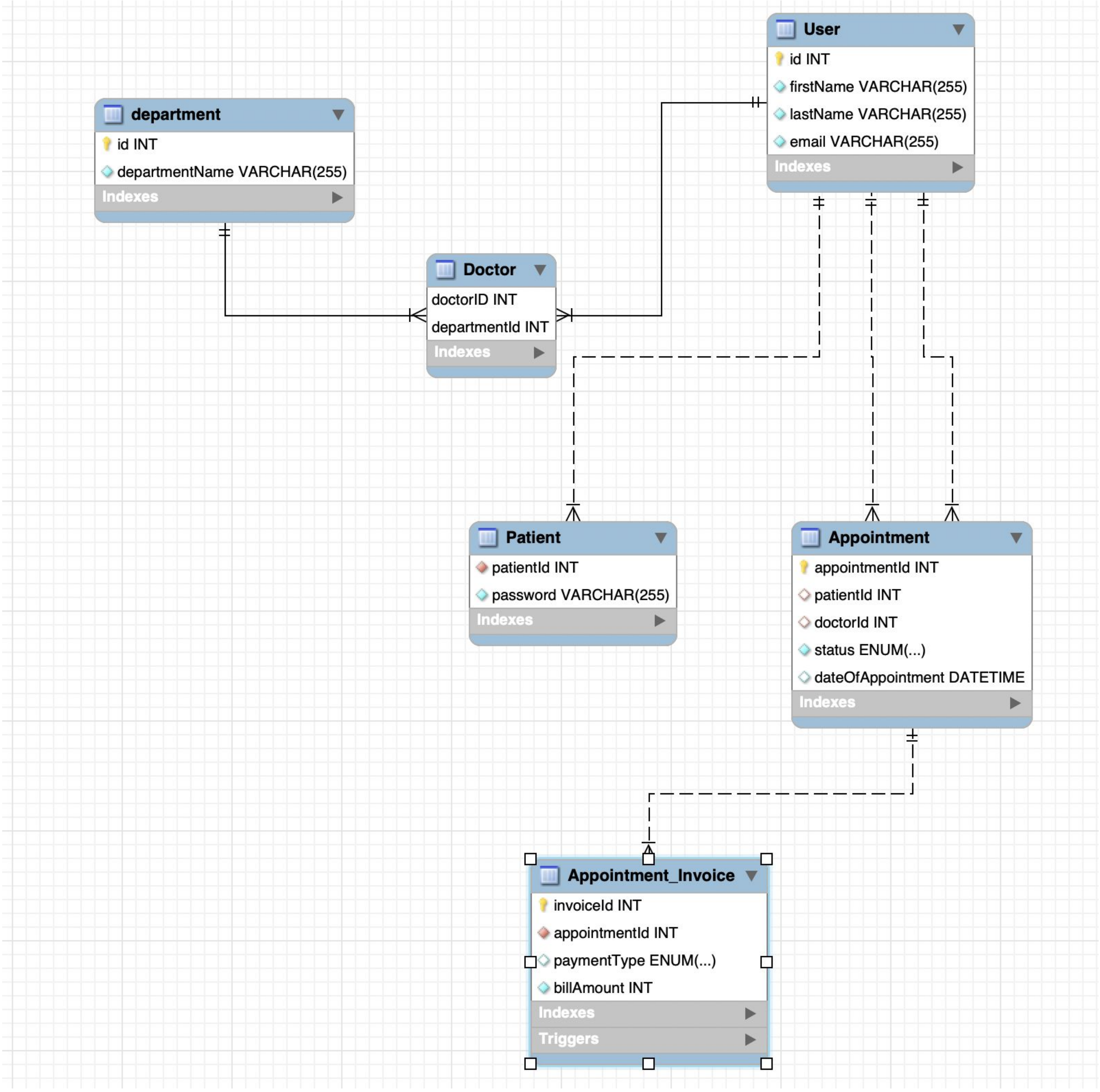
Technical Specifications

The project consisted of the front end, backend, database, and cache layer between the database and backend. The application is an MVC application. To implement the front we are using Javascript, HTML, CSS, for styling, layout, and validation. Moreover, we are using Thymeleaf so that the controller can interact with the view of the application. In addition to that, the backend is written in spring-boot which is part of the spring framework in java. This has helped to make the application faster, avoid boiler code at the server-side, and focus more on the data model than code implementation. Moreover, I have used Redis as a cache to store patient details so the patient can access the system faster. The cache is updated as soon as the database is updated, so it can be in the sink. Further, I have a MySql database which is the main storage for all the tables and data.

Conceptual Design



UML



User activity description

A user can interact with the system using a UI. After the application has started the user can perform the following operations:

http://localhost:9000/Login : Login into the system

http://localhost:9000/SignUp : Sign Up as a patient

http://localhost:9000/Home : Find the pending details

http://localhost:9000/ViewDoctors : Search doctors to book an appointment

http://localhost:9000/ViewAllAppointments: Update/delete/generate an invoice of an appointment

http://localhost:9000/UpdatePatient: Update account details like the first name, last name, email

http://localhost:9000/DeleteAccount: Delete a user account

http://localhost:9000/ViewDoctorsResult: Search all the available doctors after searching from ViewDoctors's page

http://localhost:9000/BookAppointment/{doctorId}: Book an appointment with doctor for a date where {doctorId} is doctor id

http://localhost:9000/EditAppointmentTime/{appointmentId}: Edit an appointment's date with a doctor is {appointmentId} appointment id

http://localhost:9000/DeleteAppointment/{appointmentId}: Delete an appointment

Lessons Learned

This project helped me to learn more about procedure, triggers, functions and design. It motivated me to study Redis cache as well, and find the use case of it in real applications. I gained a lot of insight into how to implement MVC architecture with database functionality using Thymeleaf in the front end. It also learned how to use database tables as data models in application code. I also realised that there can be alternative designs such a user can have enum for patient and doctor, but this design would not be efficient as the patient would not have department id, and later on, in future doctor can have a lot of attributes which patient won't have. Thus we should prefer to make separate tables. Moreover, I was at design confusion when I had to implement payment for all appointments, but then I realised that only completed/confirmed appointments payment can be done. In addition to that, I also learned that I should use a procedure and functions to reduce network load instead of sending a lot of queries for a small functionality and decrease the execution time. In sum, I learned a lot about design, use of cache and increasing efficiency of an application.

Future work

We can plan to use the same database functionality for doctor appointment management. We add a login for a doctor, so the doctor can manage their appointments with minimal code change in application and database side. We can also extend the cache to store details for the doctor. Moreover, we have lab appointments for a patient and each lab appointment will have a test for which a patient takes the appointment. We use cache to store details of commonly used lab tests, so data can be fetched faster to the user.

README

What is the need to run the project?

In this project, we are making a patient application system where we are building the application front end using HTML, CSS, and thymeleaf. In addition to that, the backend is built using java's pring framework with databases being Redis and MySQL. To avoid installing dependencies/various jar files we are using maven to build the project. All the libraries and software needed to build the project is mentioned below:

- Java (version: 8 to 11)
- MySQL server (version: any)
- Redis (version: above 2)
- Eclipse (version: any)
- MySQL workbench (version: any)

Below are details of how to install all software and libraries on MAC Os.

How to download and install Java?

1)Open the terminal in your system

2) Verify that java is present by entering the command: java -version

if the version number is not between 8 to 11, then uninstall that version by entering the following commands, else skip all the installation points:

```
cd /Library/Java/JavaVirtualMachines
```

Enter the system password if the commands prompt you to enter

```
sudo rm -rf /Library/Java/JavaVirtualMachines/jdk[version].jdk
```

```
sudo rm -rf /Library/PreferencePanes/JavaControlPanel.prefPane
```

```
sudo rm -rf /Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin
```

```
sudo rm -rf ~/Library/Application\ Support/Oracle/Java
```

Now we will install java 8.

3)To install java 8. Open the website *https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html*

4) Go to the file whose description says macOS x64

5)Click on the download link and then accept the user agreement to begin the download

5)After the download a dmg file will present in your download directory

6)Open the dmg file, and double click the package icon, and proceed further by entering the system password if it asks.

7)After java has been installed, use the following command to know java home path:

```
/usr/libexec/java_home -v1.8
```

Store the path in some text file. We will use this as a reference later to set our environment variable. The path may look like this '/Library/Java/JavaVirtualMachines/jdk1.8.0_261.jdk/Contents/Home'

8) Enter: vim .bash_profile to open the environment file where we insert the path to java home. The line that we enter in the file would be:

```
export JAVA_HOME=<JAVA_HOME_PATH>
```

In my case <JAVA_HOME_PATH> was *'/Library/Java/JavaVirtualMachines/jdk1.8.0_261.jdk/Contents/Home'*

So I inserted *JAVA_HOME=/Library/Java/JavaVirtualMachines/jdk1.8.0_261.jdk/Contents/Home* in the file.

9)After writing to the file exit from that file. Now enter the command: source ~/. bash_profile to reflect the changes.

10) Now you check java is installed by entering: *java -version*

you will get *java version "1.8.0_261"*

How to download and install Eclipse?

1) First verify that if you eclipse installed, search it in the search icon present on the top right corner of your laptop screen, if you can find it, then skip all below points till point 5.

2) To download Eclipse, go to *http://www.eclipse.org/downloads/*

3) Click on download 'X86-64' and download the dmg file

4) Go to the downloaded directory and open the dmg file; you would find an eclipse icon on the desktop when you open dmg. Open Finder, select applications and then drag the eclipse icon from the desktop the application folder in finder. Agree on the agreement and proceed, and installation is done.

5) You can now launch the eclipse by searching it from the search icon.

6) Pin it to the dock so that you can launch it directly from the desktop.

How to install and download Redis?

1)First check if Redis is installed by entering this command in terminal:

$ redis-server -v. If the version is something returned like this 'Redis server v=6.0.9 sha=00000000:0 malloc=libc bits=64 build=8b085194059a6df3', then skip all points, else follow the points to install and download redis.

2) Go to the Desktop folder and create a folder 'Redis'.

3) Open terminal and go to that folder by entering: cd Desktop/Redis

4) Enter the below command in terminal

```
$ wget https://download.redis.io/releases/redis-6.0.9.tar.gz
```

If the above command doesn't work then enter

```
$ curl https://download.redis.io/releases/redis-6.0.9.tar.gz

$ tar xzf redis-6.0.9.tar.gz

$ cd redis-6.0.9

$ make

$ make test
```

make test check whether redis is all test that confirms the installation

4)To start the server enter: *src/redis-server*

5)To start the client for performing operations in it enter:

```
$ src/redis-cli
```

So any anytime you want to start redis server Open terminal and enter:

```
cd Desktop/Redis/src/redis-server
```

Similarly, we start redis client by

```
cd Desktop/Redis/src/redis-cli
```

How to download and install the Mysql server?

First, check if your Mysql server is installed by clicking the apple icon at the top and then clicking system preferences. If you can see MySQL icon that means it is installed and you can skip the steps below.

1) Download the latest stable version of MySQL server from *http://dev.mysql.com/downloads/mysql/*

please make sure you select the operating system as Mac and you download the .dmg file.

2) Click on the downloaded .dmg file and unpack it. Click on the MySQL server package from unpacked files.

3) Install MySQL server by clicking on the MySQL package to open up the installer. If you want to install the startup script to automatically start the MySQL server at the time of system startup, you should also install the start-up package of MySQL now. Agree to the user agreement and enter a password for the server. The pass I am using for the project is 'ujjval1234' but you can have a different for your computer and then for database connection in java use your password.

How to download and install MySql Workbench?

Let's see if you have the application present by searching MySQL workbench through the search icon at the top right of the screen. If you can find it, then you can skip all the steps till step 4 else follow the steps:

1) Download the MySQL Workbench for MAC Os and the file is .dmg from the following link, and click download button when you see it available for the operating system :

https://dev.mysql.com/downloads/workbench/

2) You can begin the download without signing up by clicking 'no thanks, begin my download'

3) Search the file in your downloaded directory and click the file and the installation pane will open. Click continue to move forward with the installation.

4) Drag the Icon on the left to the Applications folder to the right to install it.

5) Open the MySQL workbench by searching it through the search icon.

6) Pin it to the dock, so it can be opened from the desktop.

7) Creating a connection. Click on the plus sign next to MySQL Connections. You will be provided with a Connection settings window. Provide the Connection name, connection method should be standard (TCP/IP) and change the port to 3306 as this is the default MySQL port. Now enter the connection and details of the server required for the client to connect. After all the details are entered press 'test connection' that test and create a connection. If the details are incorrect then enter correct details to establish the connection.You should have all the above software and libraries needed to run the project. Let's set up the application and database for it.

Setting Up MySql database

1) Open MySQL workbench which is found on desktop as suggested by the procedure or you can find by searching it on the search icon.

2) Open/Create a connection to the server.

3) Open the MySQL dump file of the database for this project in a text editor.

4) Copy the contents of the file and paste it workbench to execute the query

5) Execute all the query

Setting Up Redis Server:

Start Redis server:

cd Desktop/Redis/src/redis-server

Start Redis client to interact with server:

cd Desktop/Redis/src/redis-cli

Setting Up eclipse for the application

1)Download the application zip file and find it in the download directory

2) copy and paste it Desktop

3) Unzip the file

4) There will be a folder present DbProject1

5) Open eclipse from the desktop or search using the search icon

6) Click on 'file' then click on 'Open projects from file system'

7) In the import source select the unzipped folder directory with that folder

8) Click finish

9) Make sure you have internet connection as Maven will install dependencies for spring-boot, Jdbc, and Redis. Please wait for some time for the Maven to do its job. It may take longer, it is your first time. You can view the status in the bottom right corner of download. If the download has not finished or a problem occurred, then go to the

pom.xml file in the project directory and enter a random letter, then delete it to make the file to its original state. the maven will again check dependencies if installed. Download them if required. The server will start at port 9000 as mentioned in the application.properties file.

10) I have database properties to default values as mentioned in the procedure, but if you have changed it to your wish please open DbRepository.java file in com. example.DbPorject1. In this file, you find a few variable parameters that are final (not editable). You can edit them according to your need. I have also added comments to understand the variable purpose. Let's say you want to change the password mention in the file as you may have a different database password

old variable

private static String mysqlPass="ujjval123"; // mysql password

new variable

private static String mysqlPass="yourPassword"; // mysql password

Remove the old variable in the same line number and add a new variable instead of it. Similarly, you can change other variables according to need. Now go to DbProject1Application java and right-click on it to run as a java application. This will start your project on port 9000. If you send this message on console 'Completed initialization in 1 ms' that means the application has started. All the executed queries will be seen on the console. It will also tell when Redis did its job and when Mysql did it.You can now access the application by typing ***http://localhost:9000/*** in your browser.