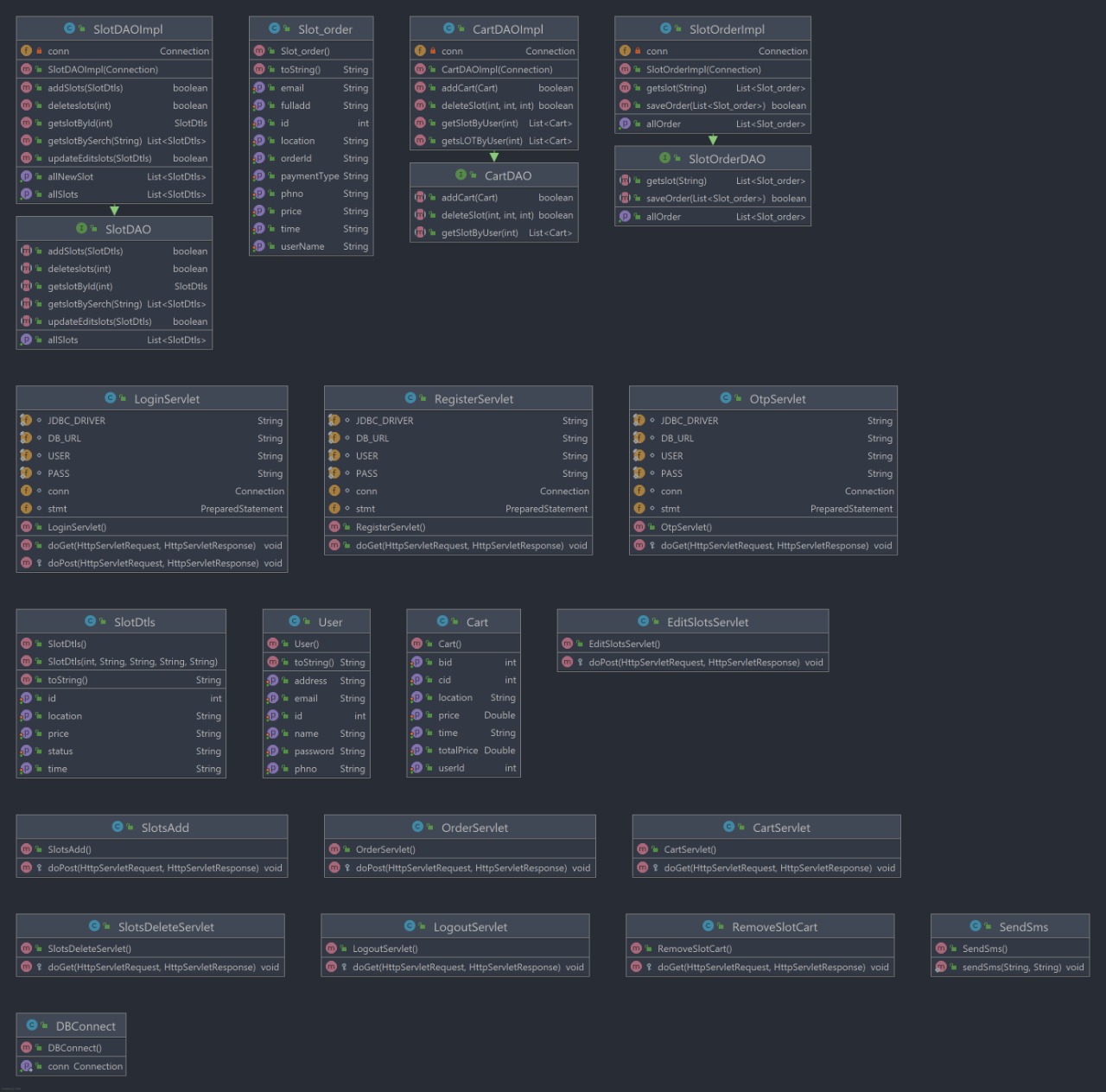
# Group Members:-

1. Moksh Papneja
2. Mohit duggal
3. Manik Arneja
4. Afrin Karim

# Class Diagram:-



# Project Description:-

# Packages

1. com.admin.servlet
2. com.dao
3. com.db
4. com.entity
5. com.servlet
6. com.user.servlet

## 1. com.admin.servlet

This package contains classes for admin controls. The Admin’s end of the program includes editing details, adding or removing services or data and even deleting all or some part of the pre-existing data.

Classes in the package:

1. EditSlotsServlet.java
2. SlotsAdd.java
3. SlotsDeleteServlet.java

The names are self explanatory.

## 2. com.dao

//dao is for data access object.

This package involves data manipulation and implementation around the slots and orders.

Classes in the package:

1. SlotDAO: This is an interface, that has parameters from SlotDtls and makes use of its implementation to store slot parameters(Id,price,location,status) in the database.
2. SlotDAOImpl.java : This class has functions that were pre-defined in the slotdao interface.
3. Slotorderdao.java : This class has functions that include orders at the user’s end and slots at the admin’s end.
4. Slotorderimpl.java : This class implements orders, slots and checks up with the class Slotorder.java to determine availability of slots and other feasibilities during execution. This also uses Dbconnect.java class to access the backend for checking if a particular slot is available.

## 3. com.db

//db is for database

This package involves the backend database of the project. It is implemented in all other packages for database connectivity.

Classes in the package:

1. Dbconnect.java

## 4. com.entity

//entity is for various entities or entries of the data entered or received throughout the run of the web application

Classes in the package:

1. Slotorder.java : Takes/Receives order details. Basically, all the parameters of the order.
2. SlotDtls.java : Contains the details of orders like Order ID, location, price, etc. It works with the class slotdaoimpl.java which further coordinates with the database package.
3. User.java : User details/entities like Name, Phone number, E-mail, etc.

## 5. com.servlet

//for all the APIs and servlets implementation throughout

Classes in the package:

1. Sendsms.java : Backend for sending SMS through an external API Fast2sms.
2. OTPservlet.java : Generates an OTP for the API to forward it to the user, stores a copy in the database too so further authentication of the OTP could take place at the user’s end (to check if the code entered matches the OTP)
3. Loginservlet.java : This class receives all the login credentials and coordinates with the database to authenticate the login. Google Authentication is also linked to this class.
4. Registerservlet.java : Checks all the details with the other classes of the entity package and coordinates with the login servlet to register the user for a first time login. After the approval, the user data is stored in the database with the help of Dbconnect.java class.

## 

## 6. com.user.servlet

//for modification to the pre-existing data and some external features at the user’s end.

Classes in the package:

1. Cartservlet.java : For Storing the items in the cart, net booking details and other details of the order/cart.
2. Logoutservlet.java : Temporarily inactivates or restricts the other classes (except the login servlet class) to access the user data so as to log-out the user. The user needs to authenticate the login again after they decide to log out of the application.
3. Removeslotcartservlet.java : Functionality: Removes the slot after pressing the ‘minus’ button in the cart .

# Programming Languages Used:

1. JAVA (throughout the project)
2. HTML and CSS (for front-end, No framework was used)
3. Javascript (for the Google Auth. Part)
4. JSP and servlets (for back-end functionalities, API integration)
5. MySQL (for database management)
6. JDBC (for Java-database connectivity)
7. Web.xml (for the Welcome page)

# Front-end files

* add\_slots.jsp
* all\_slots.jsp
* alICss.jsp
* booking.jsp
* edit\_slot.jsp
* home.jsp
* navbar.jsp
* style.css
* all\_new \_slots.jsp
* checkout.jsp
* edit\_profile.jsp
* helpline.jsp
* Home.html
* Index.jsp
* Login.html
* order\_success.jsp
* order.jsp
* Register.html
* search.jsp
* view\_slots.jsp

# Library files

* servlet-api.jar
* mysql-connector-java-5.1.49.jar
* jstl-1.2.jar
* activation-1.1.jar

Screenshots of UI:

