Method Name : Login button	class Name: Login User	ID: 01

Clients : Users

Associated Use case : Login booking

Description of Responsibilities: Username and password information should retrieve ~~ded~~ user from data base and they should not be null.

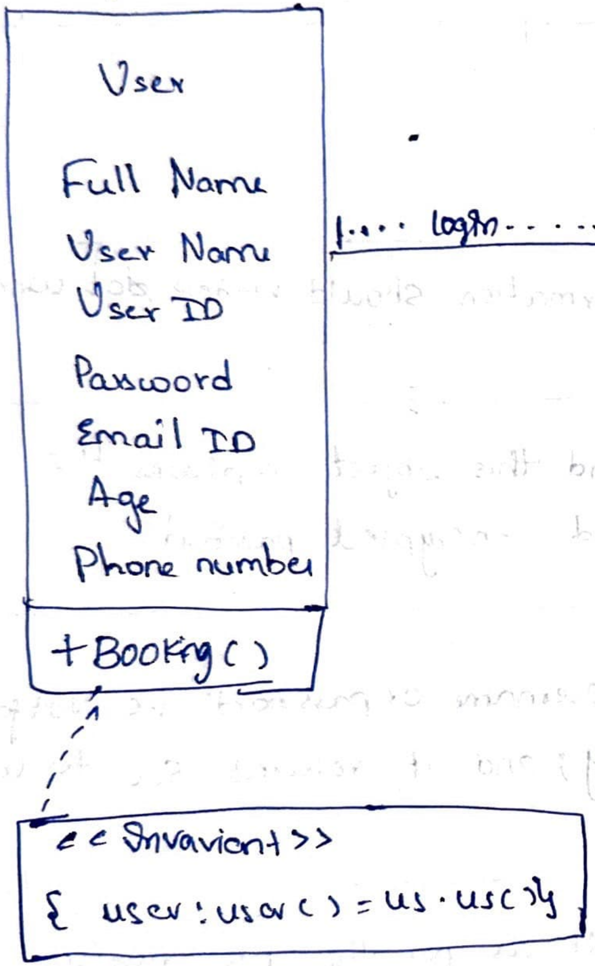
Arguments Received: As a response, it receives an object and this object contains the user information like the user name and encrypted password.

Type of value Returned: It will return either "Incorrect Username or password" i.e Notfound or OK which is "Logined successfully; and it returns 500 to 400 status accordingly.

Pre-Condition : User name and the password must not be null as per the pre-condition.

Post-Condition :- To verify if the respective user is existing in the database, we must check if i=0;
In the above condition, i refers to the number of rows returned from the database for the given Username and User password.

Class diagram



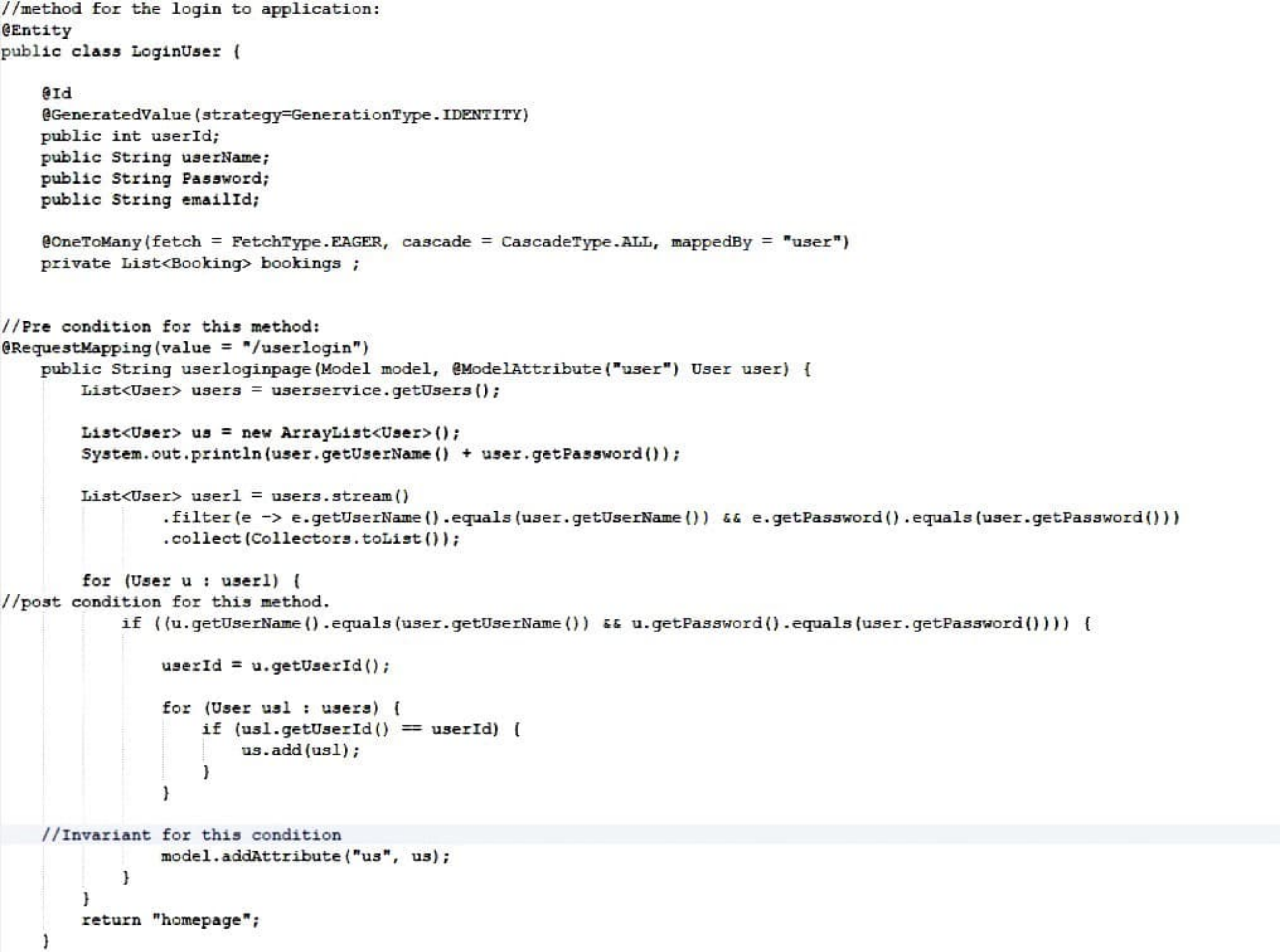
Team Members

Subhiksha Ravichandran (G02495764)

Venkata Sai Kavya Kannedari (G02494113)

Sahithi Reddy Kalijavedu (G02492753)

Sai Sujith Areti (G02494574)



```java
//method for the login to application:
@Entity
public class LoginUser {

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    public int userId;
    public String userName;
    public String Password;
    public String emailId;

    @OneToMany(fetch = FetchType.EAGER, cascade = CascadeType.ALL, mappedBy = "user")
    private List<Booking> bookings ;


//Pre condition for this method:
@RequestMapping(value = "/userlogin")
    public String userloginpage(Model model, @ModelAttribute("user") User user) {
        List<User> users = userservice.getUsers();

        List<User> us = new ArrayList<User>();
        System.out.println(user.getUserName() + user.getPassword());

        List<User> userl = users.stream()
                .filter(e -> e.getUserName().equals(user.getUserName()) && e.getPassword().equals(user.getPassword()))
                .collect(Collectors.toList());

        for (User u : userl) {
//post condition for this method.
            if ((u.getUserName().equals(user.getUserName()) && u.getPassword().equals(user.getPassword()))) {

                userId = u.getUserId();

                for (User usl : users) {
                    if (usl.getUserId() == userId) {
                        us.add(usl);
                    }
                }

    //Invariant for this condition
                model.addAttribute("us", us);
            }
        }
        return "homepage";
    }
```