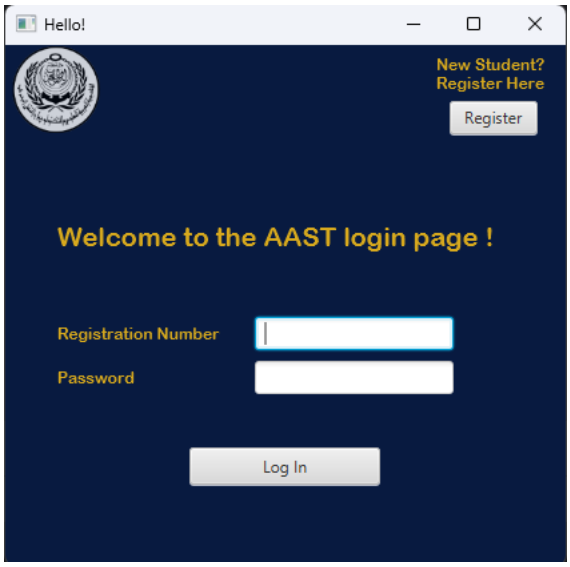
UniversityPortal

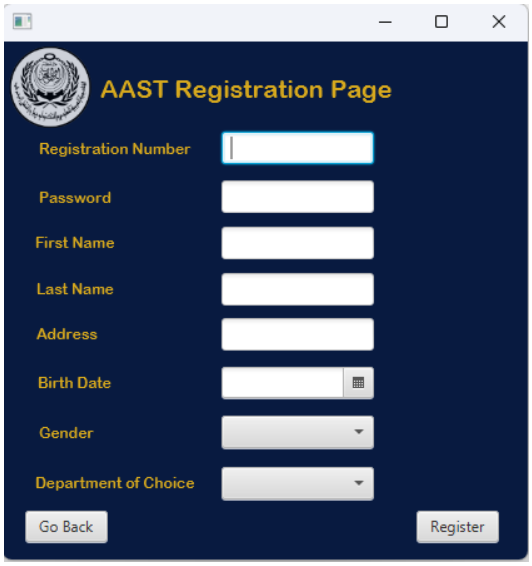
Mohamed Elsayed Taha 221002266
Abdelrahman Ahmed Nabil 221000309

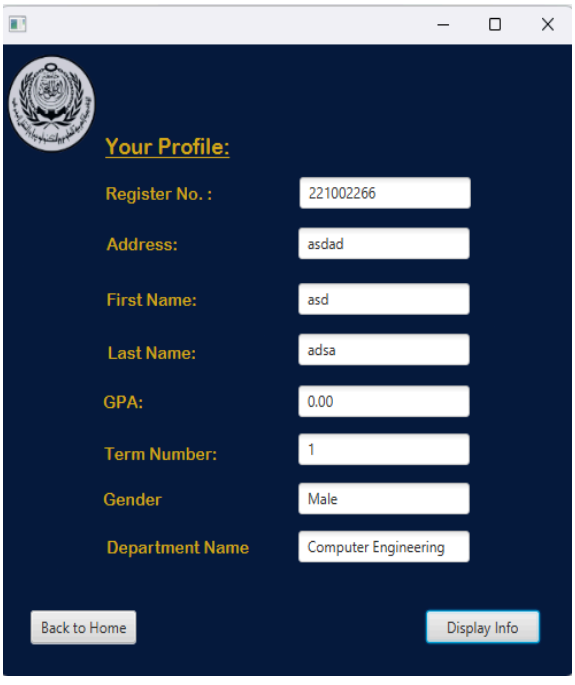
Laila Kariem Maged 221010923

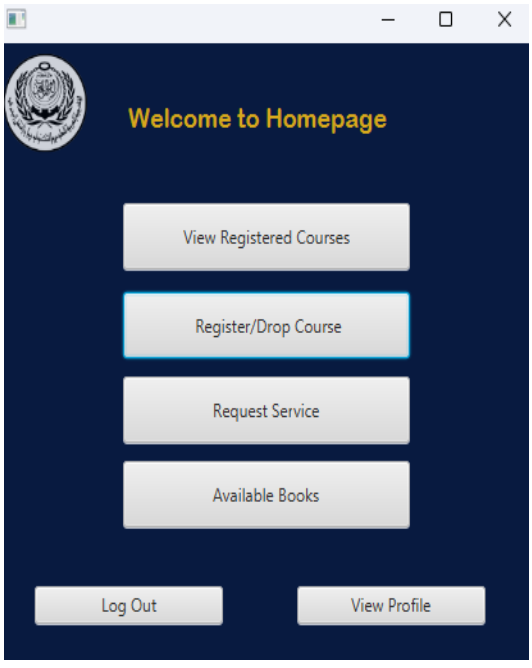
Student Portal Application

This application improves the student experience by providing a platform for various academic and campus services. Students can easily register for courses, drop courses, and track their enrollment status. They can view their grades and communicate with assigned teaching assistants and professors. Additionally, the app allows students to access university services, such as booking reservations for transportation (bus service) or medical appointments at the campus clinic.

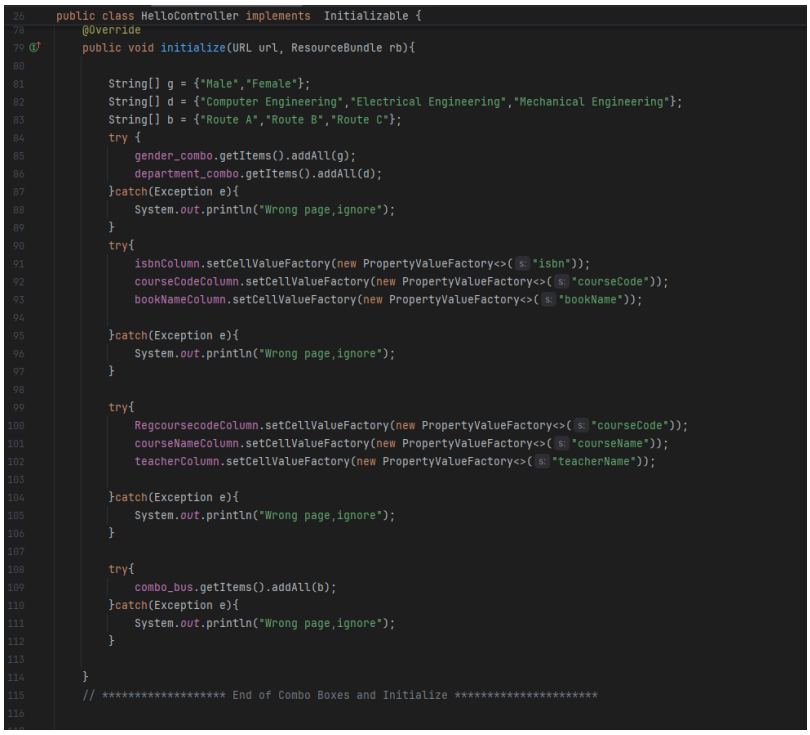








Example of Interface And Design Patterns:



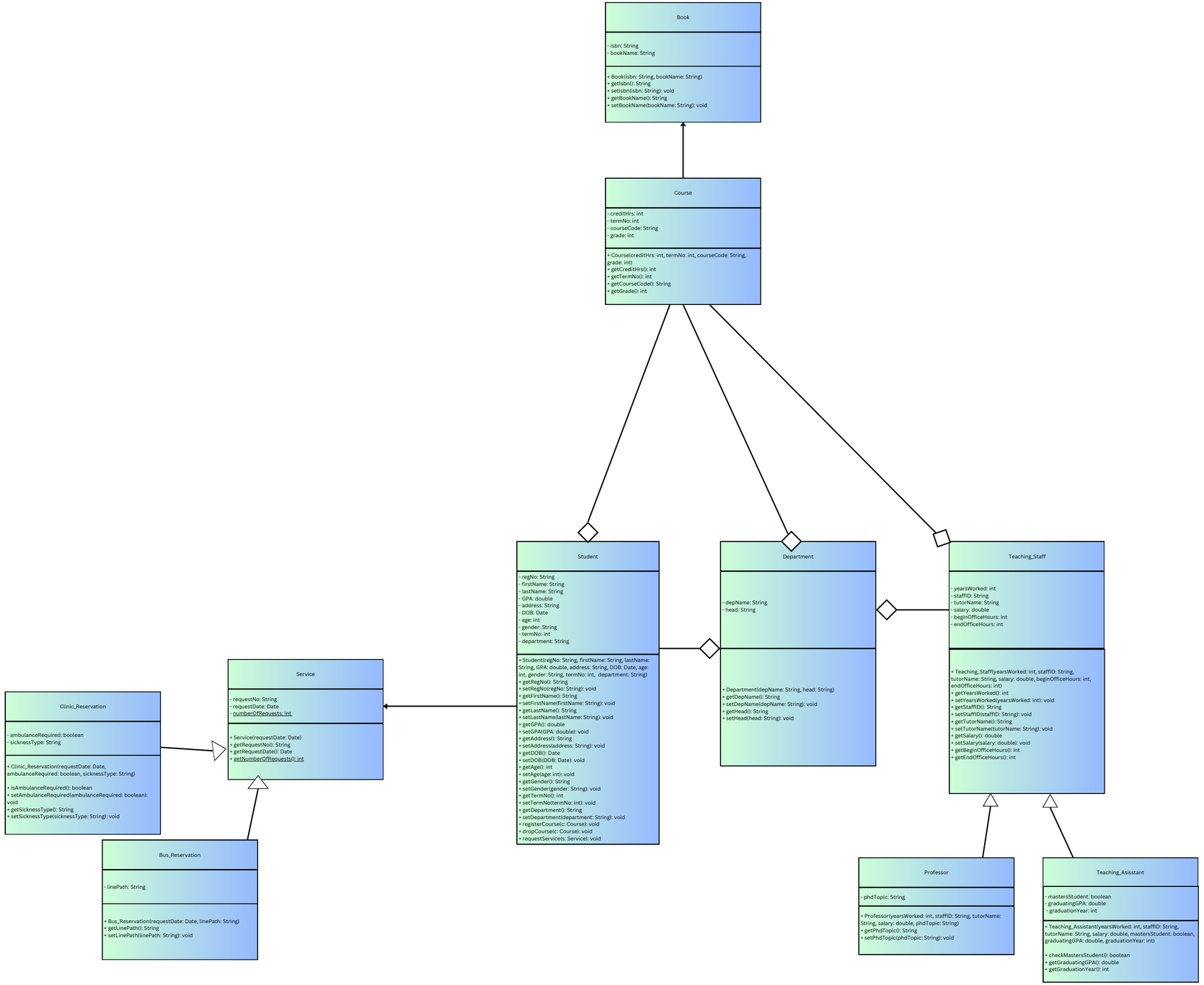
```
public class HelloController implements  Initializable {
    @Override
    public void initialize(URL url, ResourceBundle rb){

        String[] g = {"Male","Female"};
        String[] d = {"Computer Engineering","Electrical Engineering","Mechanical Engineering"};
        String[] b = {"Route A","Route B","Route C"};
        try {
            gender_combo.getItems().addAll(g);
            department_combo.getItems().addAll(d);
        }catch(Exception e){
            System.out.println("Wrong page,ignore");
        }
        try{
            isbnColumn.setCellValueFactory(new PropertyValueFactory<>( s: "isbn"));
            courseCodeColumn.setCellValueFactory(new PropertyValueFactory<>( s: "courseCode"));
            bookNameColumn.setCellValueFactory(new PropertyValueFactory<>( s: "bookName"));

        }catch(Exception e){
            System.out.println("Wrong page,ignore");
        }

        try{
            RegcoursecodeColumn.setCellValueFactory(new PropertyValueFactory<>( s: "courseCode"));
            courseNameColumn.setCellValueFactory(new PropertyValueFactory<>( s: "courseName"));
            teacherColumn.setCellValueFactory(new PropertyValueFactory<>( s: "teacherName"));

        }catch(Exception e){
            System.out.println("Wrong page,ignore");
        }

        try{
            combo_bus.getItems().addAll(b);
        }catch(Exception e){
            System.out.println("Wrong page,ignore");
        }

    }
    // ****************** End of Combo Boxes and Initialize *********************
```

Explanation:

Here is a snippet of the controller code, which contains all necessary methods for event handling and any "active" elements that are variable to change in the FXML pages, such as text fields, buttons etc..

One of its functionalities is initializing the value of Combo boxes, through a list containing the possible values. It's important to add it to a try-catch so that another page that doesn't recognize these elements executes this code leading to a NullPointerException that will stop the code entirely

Initialization is also crucial for tables by defining the exact data type the table will take, as well as the values for each column.


Book
- isbn: String
- bookName: String
+ Book(isbn: String, bookName: String)
+ getIsbn(): String
+ setIsbn(isbn: String): void
+ getBookName(): String
+ setBookName(bookName: String): void
Course
- creditHrs: int
- termNo: int
- courseCode: String
- grade: int
+ Course(creditHrs: int, termNo: int, courseCode: String, grade: int)
+ getCreditHrs(): int
+ getTermNo(): int
+ getCourseCode(): String
+ getGrade(): int
Student
- regNo: String
- firstName: String
- lastName: String
- GPA: double
- address: String
- DOB: Date
- age: int
- gender: String
- termNo: int
- department: String
+ Student(regNo: String, firstName: String, lastName: String, GPA: double, address: String, DOB: Date, age: int, gender: String, termNo: int, department: String)
+ getRegNo(): String
+ setRegNo(regNo: String): void
+ getFirstName(): String
+ setFirstName(firstName: String): void
+ getLastName(): String
+ setLastName(lastName: String): void
+ getGPA(): double
+ setGPA(GPA: double): void
+ getAddress(): String
+ setAddress(address: String): void
+ getDOB(): Date
+ setDOB(DOB: Date): void
+ getAge(): int
+ setAge(age: int): void
+ getGender(): String
+ setGender(gender: String): void
+ getTermNo(): int
+ setTermNo(termNo: int): void
+ getDepartment(): String
+ setDepartment(department: String): void
+ registerCourse(c: Course): void
+ dropCourse(c: Course): void
+ requestService(s: Service): void
Department
- depName: String
- head: String
+ Department(depName: String, head: String)
+ getDepName(): String
+ setDepName(depName: String): void
+ getHead(): String
+ setHead(head: String): void
Teaching_Staff
- yearsWorked: int
- staffID: String
- tutorName: String
- salary: double
- beginOfficeHours: int
- endOfficeHours: int
+ Teaching_Staff(yearsWorked: int, staffID: String, tutorName: String, salary: double, beginOfficeHours: int, endOfficeHours: int)
+ getYearsWorked(): int
+ setYearsWorked(yearsWorked: int): void
+ getStaffID(): String
+ setStaffID(staffID: String): void
+ getTutorName(): String
+ setTutorName(tutorName: String): void
+ getSalary(): double
+ setSalary(salary: double): void
+ getBeginOfficeHours(): int
+ getEndOfficeHours(): int
Service
- requestNo: String
- requestDate: Date
- numberOfRequests: int
+ Service(requestDate: Date)
+ getRequestNo(): String
+ getRequestDate(): Date
+ getNumberOfRequests(): int
Clinic_Reservation
- ambulanceRequired: boolean
- sicknessType: String
+ Clinic_Reservation(requestDate: Date, ambulanceRequired: boolean, sicknessType: String)
+ isAmbulanceRequired(): boolean
+ setAmbulanceRequired(ambulanceRequired: boolean): void
+ getSicknessType(): String
+ setSicknessType(sicknessType: String): void
Bus_Reservation
- linePath: String
+ Bus_Reservation(requestDate: Date, linePath: String)
+ getLinePath(): String
+ setLinePath(linePath: String): void
Professor
- phdTopic: String
+ Professor(yearsWorked: int, staffID: String, tutorName: String, salary: double, phdTopic: String)
+ getPhdTopic(): String
+ setPhdTopic(phdTopic: String): void
Teaching_Asisstant
- mastersStudent: boolean
- graduatingGPA: double
- graduationYear: int
+ Teaching_Assistant(yearsWorked: int, staffID: String, tutorName: String, salary: double, mastersStudent: boolean, graduatingGPA: double, graduationYear: int)
+ checkMastersStudent(): boolean
+ getGraduatingGPA(): double
+ getGraduationYear(): int