**BÀI TẬP CÁ NHÂN NHẬP MÔN CÔNG NGHỆ PHẦN MỀM**

**THÔNG TIN**

Họ và tên: Đinh Quang Hưng

Mã sinh viên: B22DCCN407

Nhóm học phần: 09

Nhóm bài tập: 03

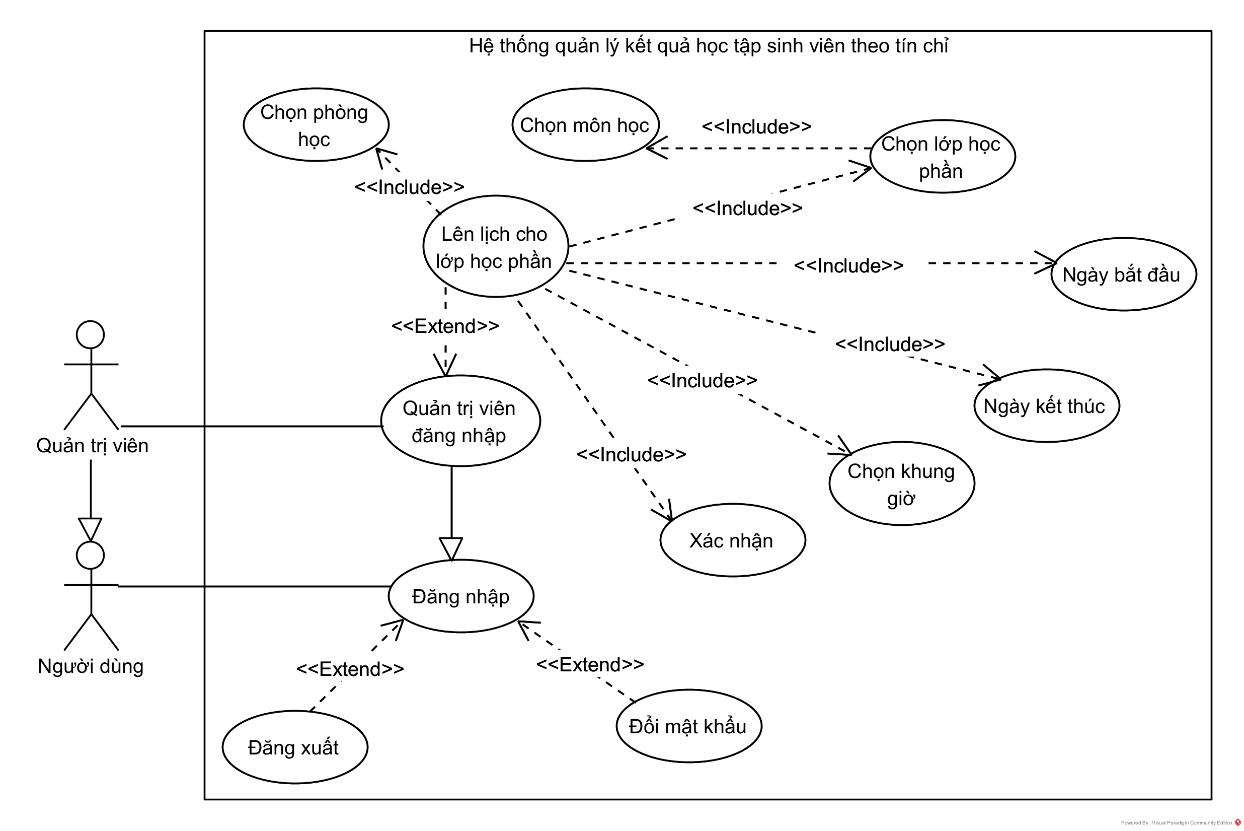
Bài tập nhóm: Hệ thống quản lý kết quả học tập sinh viên theo tín chỉ

Bài tập cá nhân: Modul ***“Lên lịch cho lớp học phần”***

Giảng viên hướng dẫn: Th.S Ngô Tiến Đức

**BÀI LÀM**

1. Vẽ và mô tả các sơ đồ use case chi tiết cho modul ***“Lên lịch cho lớp học phần”***

****

**Mô tả Use Case**

+ Chọn môn học: UC này cho phép Quản trị viên chọn một môn học từ danh sách để hệ thống cập nhật danh sách lớp học phần tương ứng.

+Chọn lớp học phần: UC này cho phép Quản trị viên chọn một lớp học phần từ danh sách lớp học phần đã được cập nhật.

+ Chọn phòng học: UC này cho phép Quản trị viên chọn một phòng học từ danh sách các phòng học

+ Chọn khung giờ: UC này cho phép Quản trị viên chọn một khung giờ trong tuần từ danh sách để xác định thời gian học.

+ Ngày bắt đầu: UC này cho phép Quản trị viên chọn ngày bắt đầu áp dụng lịch học.

+ Ngày kết thúc: UC này cho phép Quản trị viên chọn ngày kết thúc áp dụng lịch học.

+ Xác nhận : UC này cho phép Quản trị viên xác nhận thông tin đã chọn để hệ thống lưu lịch học vào CSDL

1. Viết các scenario chuẩn và scenario ngoại lệ cho các use case

### 2.1. Scenario chuẩn

1. Quản trị viên A chọn chức năng "Lên lịch học cho lớp học phần" từ menu
2. Giao diện lên lịch học cho lớp học phần hiện ra với các ô sổ chọn môn học, lớp học phần, phòng học, khung giờ và ô nhập ngày bắt đầu, ngày kết thúc cùng nút xác nhận
3. Quản trị viên A chọn môn “Nhập môn công nghệ phần mềm”.
4. Hệ thống cập nhật danh sách lớp học phần là D22-001 D22-002 D22-003 của môn học “Nhập môn công nghệ phần mềm”.
5. Quản trị viên A chọn lớp học phần là D22-001 từ ô sổ xuống. Quản trị viên A chọn phòng học 301-A2 từ danh sách sổ xuống. Quản trị viên A nhập ngày bắt đầu là 2025-02-01 và ngày kết thúc 2025-02-01, chọn khung giờ từ ô sổ xuống là Thứ 6 7:00-7:50 . Quản trị viên A nhấn nút xác nhận.
6. Hệ thống hiển thị thông báo lên lịch học thành công.
7. Quản trị viên A click nút OK
8. Hệ thống quay về giao diện lên lịch học cho lớp học phần của người quản trị viên

### 2.2. Scenario Ngoại Lệ

#### *Trường hợp 1*

* 1. Hệ thống thông báo phòng 301-A2 đã được sử dụng
     1. Quản trị viên A chọn phòng 401-A2 và ấn nút xác nhận
     2. Hệ thống hiển thị thông báo lên lịch học thành công
     3. Quản trị viên A click nút OK
     4. Hệ thống quay về giao diện lên lịch học cho lớp học phần

#### *Trường hợp 2*

3.1 Quản trị viên A không chọn môn học và click nút xác nhận

3.1.1 Thông báo yêu cầu nhập đầy đủ thông tin hiện ra.

3.1.2 A click vào nút OK

3.1.3 Hệ thống quay lại giao diện lên lịch học cho lớp học phần

#### *Trường hợp 3*

* 1. Quản trị viên A chọn lớp học phần là D22-001 từ ô sổ xuống. Quản trị viên A chọn phòng học 301-A2 từ danh sách sổ xuống. Quản trị viên A nhập ngày bắt đầu là 20-01-2025 và ngày kết thúc 06-01-2025, chọn khung giờ từ ô sổ xuống là Thứ 6 7:00-7:50 . Quản trị viên A nhấn nút xác nhận.
     1. Hệ thống hiển thị thông báo ngày không đúng định dạng yyyy-MM-dd!
     2. Quản trị viên A sửa thành ngày bắt đầu 2025-01-20 và ngày kết thúc thành 2025-01-06 và click nút Xác nhận
     3. Hệ thống hiển thị thông báo lên lịch học thành công
     4. Quản trị viên A click nút OK
     5. Hệ thống quay về giao diện lên lịch học cho lớp học phần

1. Trích các lớp thực thể và vẽ sơ đồ lớp thực thể liên quan đến modul

## Trích lớp thực thể:

## Lịch học - > lớp Schedule, lớp học phần->ClassSection, môn học -> Course, phòng học -> lớp Room, khung giờ -> lớp Time

## Xác định các thuộc tính:

## + Lớp Schedule: ngày bắt đầu, ngày kết thúc

## + Lớp ClassSection: tên lớp,nhóm, khóa,kì học, năm học, ghi chú

## + Lớp Course: tên môn học, số tín chỉ, loại môn, mô tả

## + Lớp Time: thứ trong tuần, giờ bắt đầu, giờ kết thúc

## + Lớp Room: tên phòng, loại phòng, trạng thái

## Quan hệ giữa các lớp thực thể

## + Một Course có thể nhiều ClassSection học, một ClassSection phải phụ thuộc vào một Course -> quan hệ giữa Course và ClassSection là composition 1 – n

## 

## + Một ClassSection có thể có nhiều Schedule, một Schedule thuộc một ClassSection→ quan hệ giữa ClassSection và Schedule là aggregation 1 – n

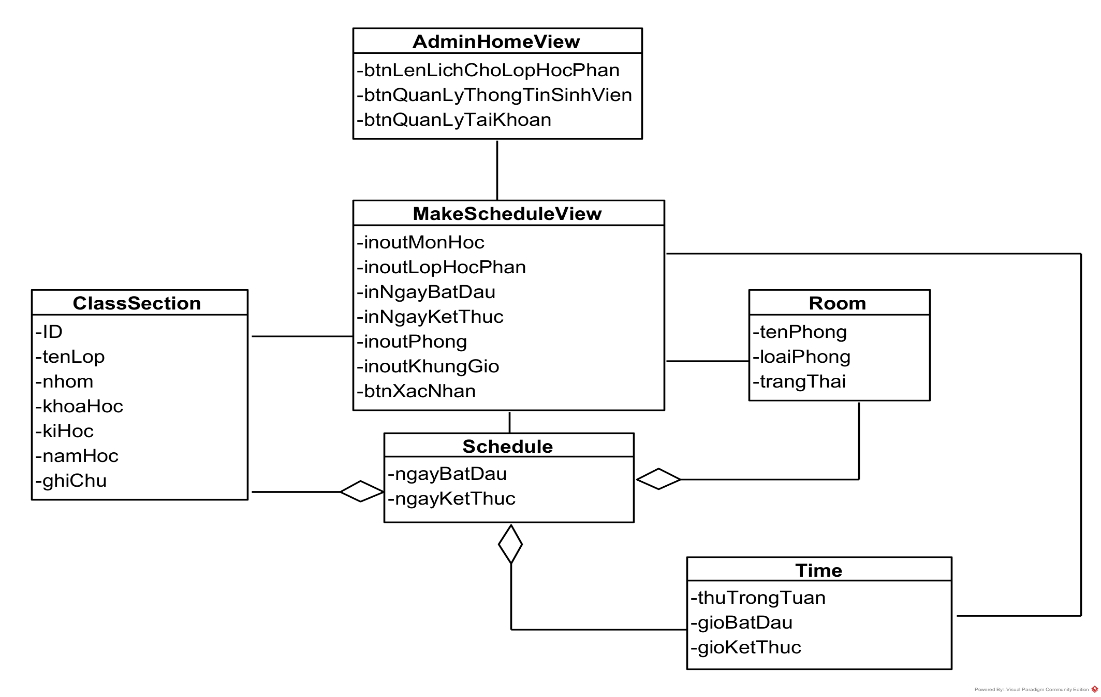
## 

## Sơ đồ lớp thực thể

## 

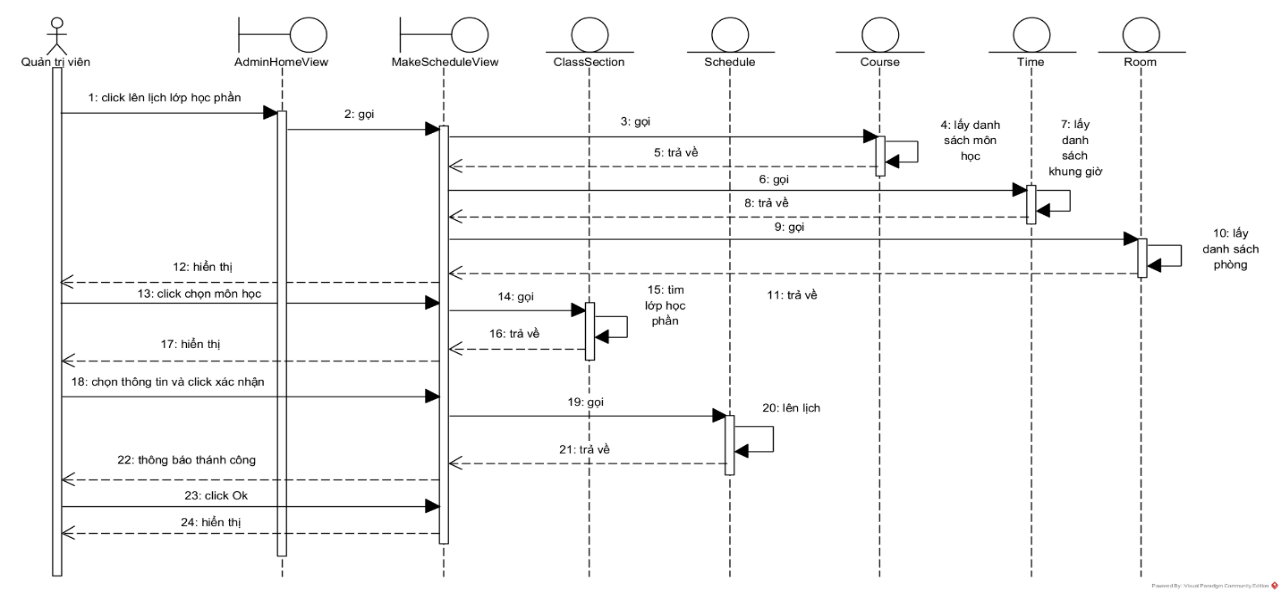
## Vẽ sơ đồ lớp phân tích và biểu đồ tuần tự

* Sơ đồ lớp phân tích cho modul ***“Lên lịch cho lớp học phần”***

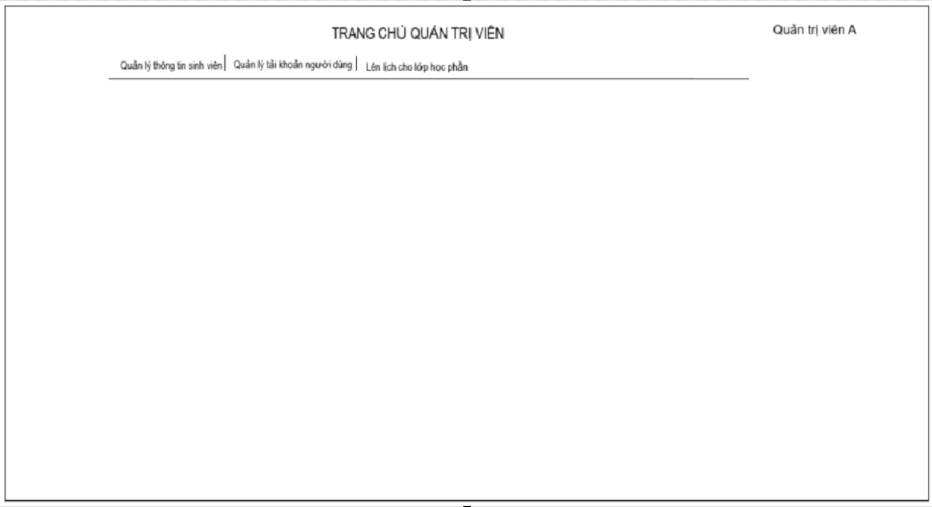


* Biểu đồ lớp tuần tự

1. Quản trị viên A chọn chức năng "Lên lịch học cho lớp học phần" từ AdminHomeView
2. Lớp AdminHomeView gọi lớp MakeScheduleView yêu cầu hiển thị
3. Lớp MakeScheduleView gọi lớp Course yêu cầu xử lý
4. Lớp Course gọi phương thức lấy danh sách môn học
5. Lớp Course trả lại kết quả cho lớp MakeScheduleView
6. Lớp MakeScheduleView gọi lớp Time yêu cầu xử lý
7. Lớp Course gọi phương thức lấy các môn học
8. Lớp Course trả lại kết quả cho lớp MakeScheduleView
9. Lớp MakeScheduleView gọi lớp Course yêu cầu xử lý
10. Lớp Course gọi phương thức lấy các môn học
11. Lớp Course trả lại kết quả cho lớp MakeScheduleView
12. Lớp MakeScheduleView hiển thị cho quản trị viên
13. Quản trị viên chọn môn học từ danh sách sổ xuống
14. Lớp MakeScheduleView gọi lớp ClassSection yêu cầu xử lý
15. Lớp ClassSection gọi phương thức lấy các lớp học phần của môn học
16. Lớp ClassSection trả lại kết quả cho lớp MakeScheduleView
17. Lớp MakeScheuleView hiển thị lớp học phần của môn học
18. Quản trị viên chọn các thông tin từ danh sách sổ xuống, nhập thông tin và click nút xác nhận
19. Lớp MakeScheduleView gọi lớp Schedule yêu cầu xử lý
20. Lớp Schedule gọi phương thức lên lịch
21. Lớp Schedule trả lại kết quả cho lớp MakeScheduleView
22. Lớp MakeScheduleView hiện thông báo thành công
23. Quản trị viên click vào nút OK của thông báo
24. Lớp MakeScheduleView hiển thị cho quản trị viên



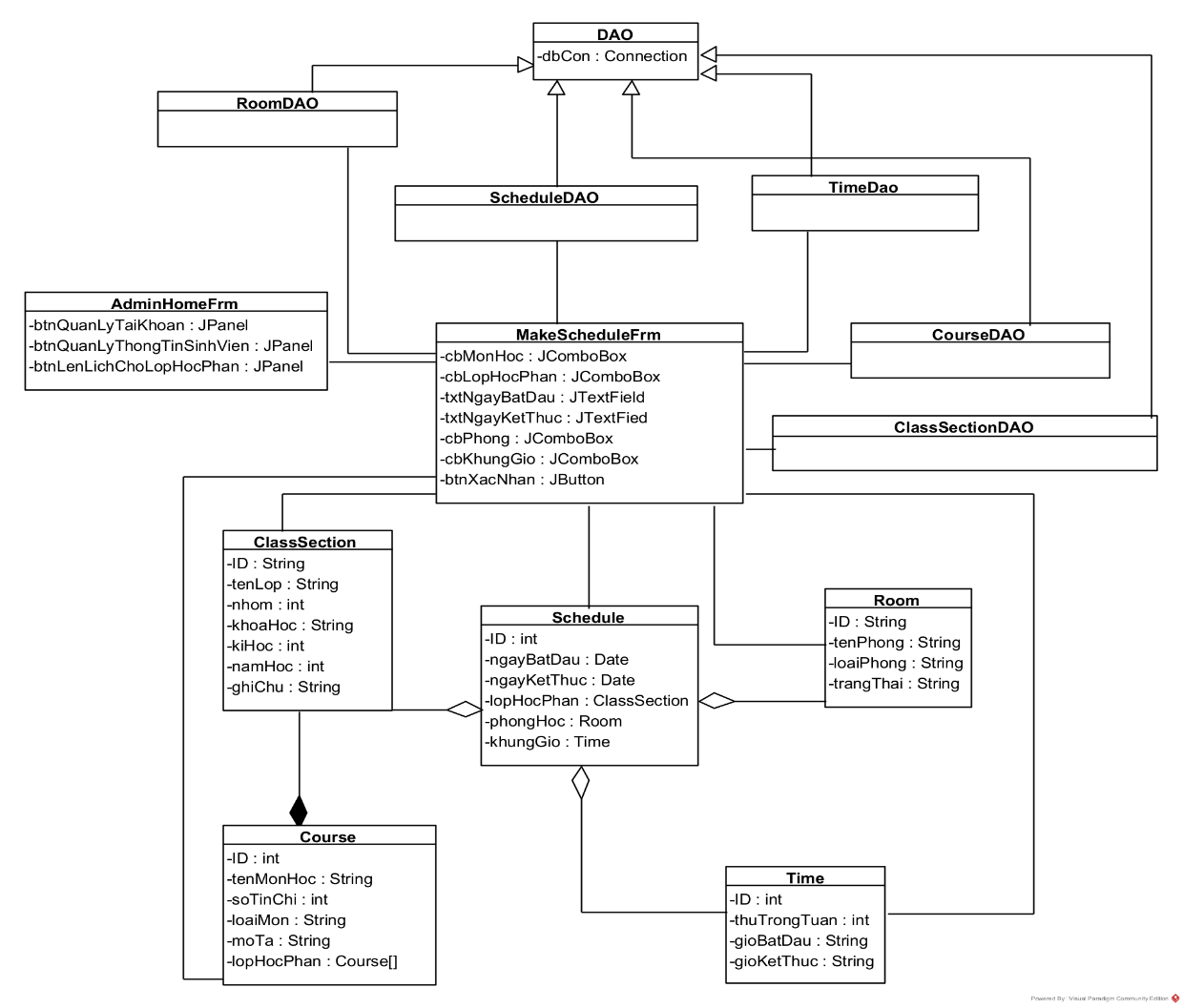
## Thiết kế các giao diện cho modul





## Thiết kế tĩnh

* Thêm thuộc tính cho các lớp giao diện và lớp điều khiển dạng DAO



* Định nghĩa các phương thức cho modul

+ AdminHomeFrm phải có các phương thức hiển thị form(), xử lý nút lên lịch cho lớp học phần

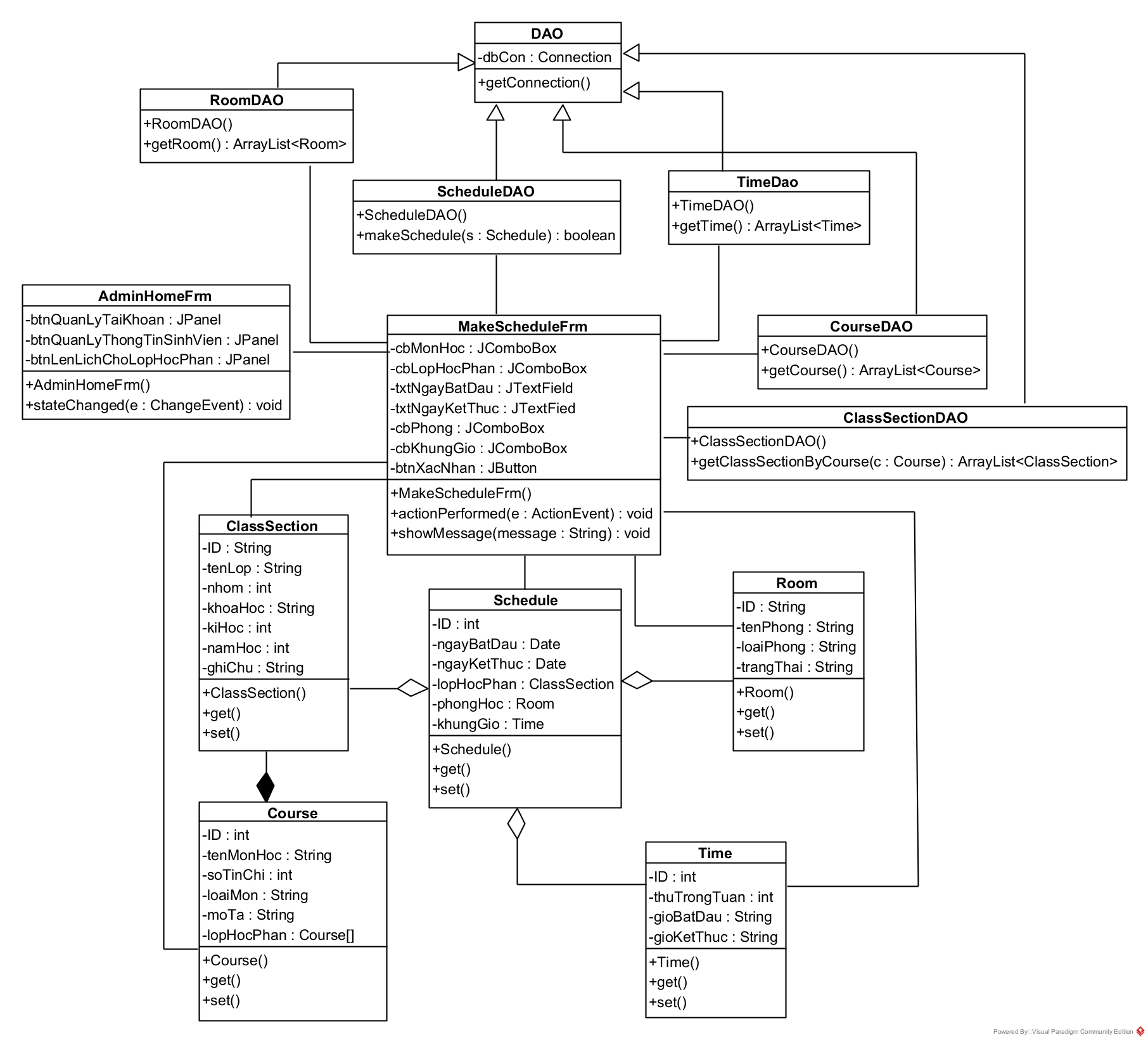
+ Lớp MakeScheduleFrm phải có các phương thức hiển thị form (hàm khởi tạo), xử lý nút Xác nhận, thông báo thành công

+ Lớp RoomDAO,CourseDAO,TimeDAO, ClassSectionDAO phải có phương thức lấy thông tin từ cơ sở dữ liệu

+ Lớp ScheduleDAO phải có phương thức lưu thông tin lịch vào cơ sở dữ liệu

+ Lớp Room, Time,Schedule,Course, ClassSection phải có các phương thức đóng gói thông tin đối tượng (hàm khởi tạo) và các phương thức get/set

* Sơ đồ lớp chi tiết theo mô hình MVC thuần



## Thiết kế động: Vẽ sơ đồ tuần tự thiết kế

## Quản trị viên click nút lên lịch từ AdminHomeFrm

## AdminHomeFrm kích hoạt phương thức stateChanged() để yêu cầu lớp MakeScheduleFrm hiển thị

## Phương thức stateChanged() gọi lớp MakeScheduleFrm

## Hàm khởi tạo MakeScheduleFrm() được kích hoạt

## Lớp MakeScheduleFrm gọi phương thức getCourse() của lớp CourseDAO

## Phương thức getCourse() lấy thông tin các môn học từ cơ sở dữ liệu

## Phương thức getCourse() trả về danh sách các môn học cho lớp MakeScheduleFrm

## Lớp MakeScheduleFrm gọi phương thức getTime() của lớp TimeDAO

## Phương thức getTime() lấy thông tin các khung giờ từ cơ sở dữ liệu

## Phương thức getTime() trả về danh sách các khung giờ cho lớp MakeScheduleFrm

## Lớp MakeScheduleFrm gọi phương thức getRoom() của lớp RoomDAO

## Phương thức getRoom() lấy thông tin các phòng từ cơ sở dữ liệu

## Phương thức getRoom() trả về danh sách các phòng cho lớp MakeScheduleFrm

## Giao diện MakeScheduleFrm hiển thị cho quản trị viên

## Quản trị viên chọn môn học từ MakeScheduleFrm

## MakeScheduleFrm kích hoạt phương thức actionPerformed() để gọi phương thức getClassSectionByCourse() của lớp ClassSectionDAO

## Phương thức actionPerformed() gọi phương thức getClassSectionByCourse()

## Phương thức getClassSectionByCourse() lấy dữ liệu từ cơ sở dữ liệu

## Phương thức getClassSectionByCourse() trả quyển điều khiển về cho phương thức actionPerformed()

## Phương thức actionPerformed() hiển thị dữ liệu lớp học phần của môn học

## Quản trị viên chọn các thông tin từ ô sổ và nhập ngày bắt đầu, kết thúc và click nút xác nhận

## Lớp MakeScheduleFrm kích hoạt phương thức actionPerformed() để gọi lớp Schedule yêu cầu khởi tạo đối tượng Schedule

## Phương thức actionPerformed() gọi lớp Schedule

## Lớp Schedule gọi hàm set để đóng gói đối tượng từ MakeScheduleFrm thành một thực thể có kiểu dữ liệu Schedule

## Thực thể được trả về cho phương thức actionPerformed()

## Phương thức actionPerformed() gọi phương thức makeSchedule() của lớp RoomDAO

## Phương thức makeSchedule() cập nhật thông tin vào mới vào CSDL

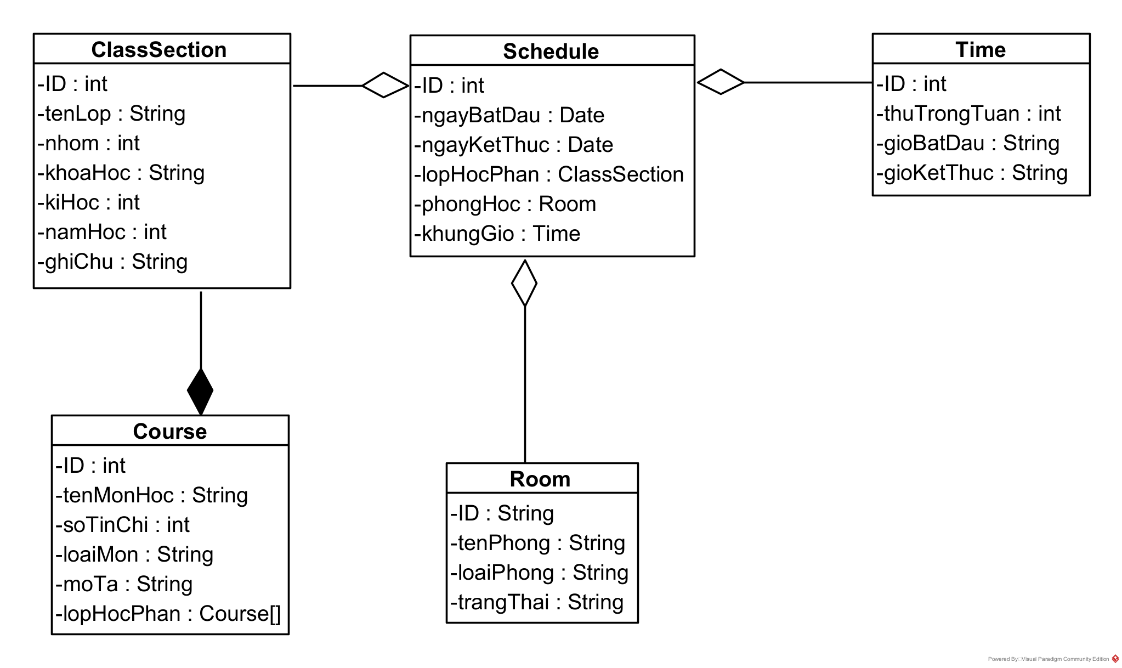
## Phương thức makeSchedule () trả quyền điều khiển về lại cho phương thức actionPerformed()

## Phương thức actionPerformed() hiện thông báo thành công cho Quản trị viên

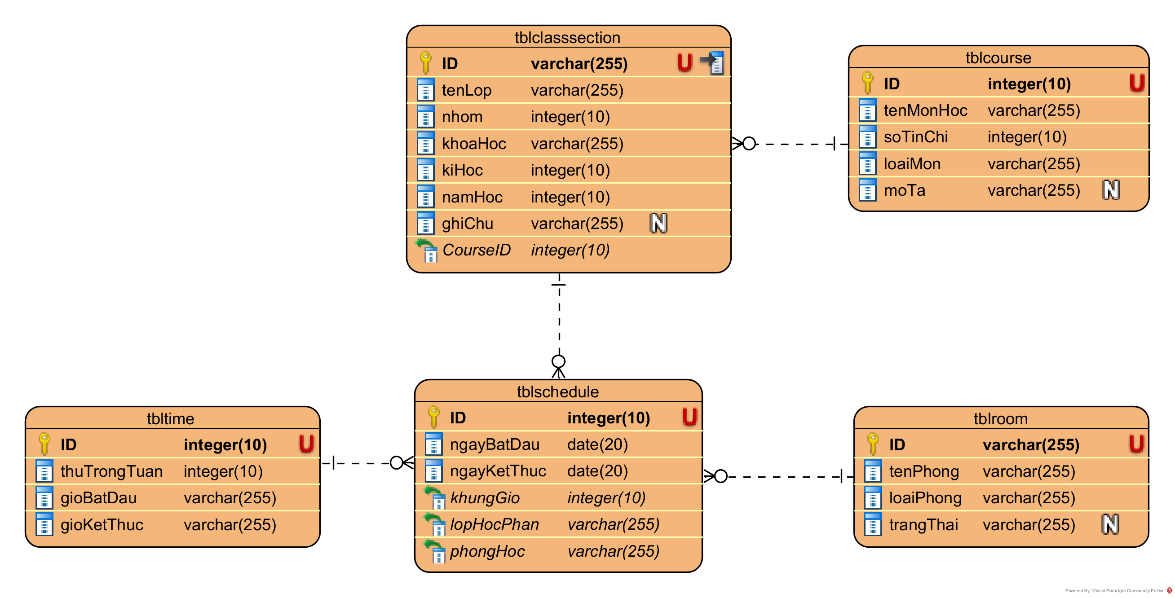
## 

## Vẽ sơ đồ lớp và sơ đồ CSDL

Sơ đồ lớp



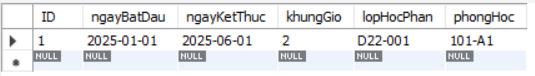
Sơ đồ CSDL



## Viết các test case

* Test case cho chức năng lên một lịch cho lớp học phần - Trường hợp lịch chưa có trong CSDL

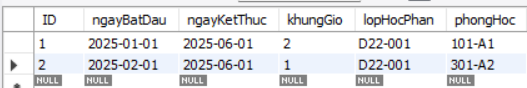
Bước 1: Dữ liệu hiện có trong CSDL



Bước 2:

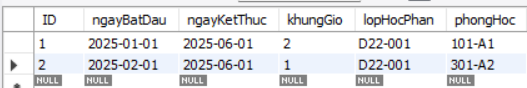
|  |  |
| --- | --- |
| Các bước thực hiện | Kết quả mong đợi |
| 1. Click Lên lịch cho lớp học phần | Giao diện lên lịch hiện ra gồm các ô sổ xuống : Môn học, Lớp học phần, Phòng, Khung giờ và các ô nhập : Ngày bắt đầu, Ngày kết thúc |
| 1. Chọn và nhập thông tin lịch:  * Môn học = Nhập môn công nghệ phần mềm * Lớp học phần = D22-001 * Phòng = 301-A2 * Khung giờ = Thứ 6 7:00-7:50 * Ngày bắt đầu= 2025-02-01 * Ngày kết thúc=2025-06-01   Và click nút xác nhận | Thông báo hiện lên: “Thêm lịch mới thành công” và nút OK |
| 1. Click vào nút OK của thông báo | Quay về giao diện lên lịch |

Bước 3: Kết quả mong đợi trong CSDL



* Test case cho chức năng lên một lịch cho lớp học phần - Trường hợp lịch bị trùng với lịch khác trong CSDL

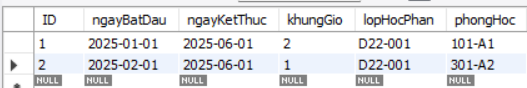
Bước 1: Dữ liệu hiện có trong CSDL



Bước 2:

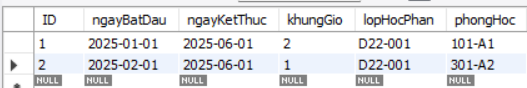
|  |  |
| --- | --- |
| 1. Click Lên lịch cho lớp học phần | Giao diện lên lịch hiện ra gồm các ô sổ xuống : Môn học, Lớp học phần, Phòng, Khung giờ và các ô nhập : Ngày bắt đầu, Ngày kết thúc |
| 1. Chọn và nhập thông tin lịch:  * Môn học = Nhập môn công nghệ phần mềm * Lớp học phần = D22-001 * Phòng = 301-A2 * Khung giờ = Thứ 6 7:00-7:50 * Ngày bắt đầu= 2025-02-01 * Ngày kết thúc=2025-06-01   Và click nút xác nhận | Thông báo hiện lên: “lịch bị trùng”, lênh lịch không thành công và nút OK |
| 1. Click vào nút OK của thông báo | Quay về giao diện lên lịch |

Bước 3: Kết quả mong đợi trong CSDL



* Test case cho chức năng lên một lịch cho lớp học phần - Trường hợp phòng được dùng trong khung giờ đã chọn

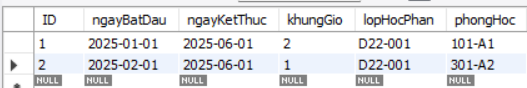
Bước 1: Dữ liệu hiện có trong CSDL



Bước 2:

|  |  |
| --- | --- |
| 1. Click Lên lịch cho lớp học phần | Giao diện lên lịch hiện ra gồm các ô sổ xuống : Môn học, Lớp học phần, Phòng, Khung giờ và các ô nhập : Ngày bắt đầu, Ngày kết thúc |
| 1. Chọn và nhập thông tin lịch:  * Môn học = Kỹ năng làm việc nhóm * Lớp học phần = D22-004 * Phòng = 301-A2 * Khung giờ = Thứ 6 7:00-7:50 * Ngày bắt đầu= 2025-03-26 * Ngày kết thúc=2025-04-30   Và click nút xác nhận | Thông báo hiện lên: “Phòng đã được sử dụng trong khung giờ”, lênh lịch không thành công và nút OK |
| 1. Click vào nút OK của thông báo | Quay về giao diện lên lịch |

Bước 3: Kết quả mong đợi trong CSDL



## Cài đặt theo kiến trúc đã thiết kế và demo chương trình

Ngôn ngữ lập trình: Java

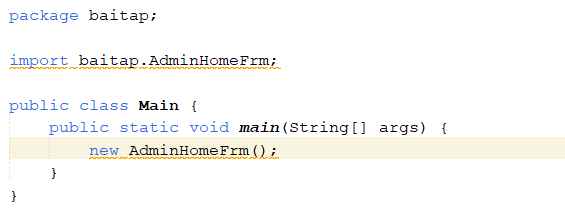
• CSDL: MySQL

• Trước khi chạy code:

• Bật MySQL Server

• Thêm driver JDBC của MySQL vào library của project

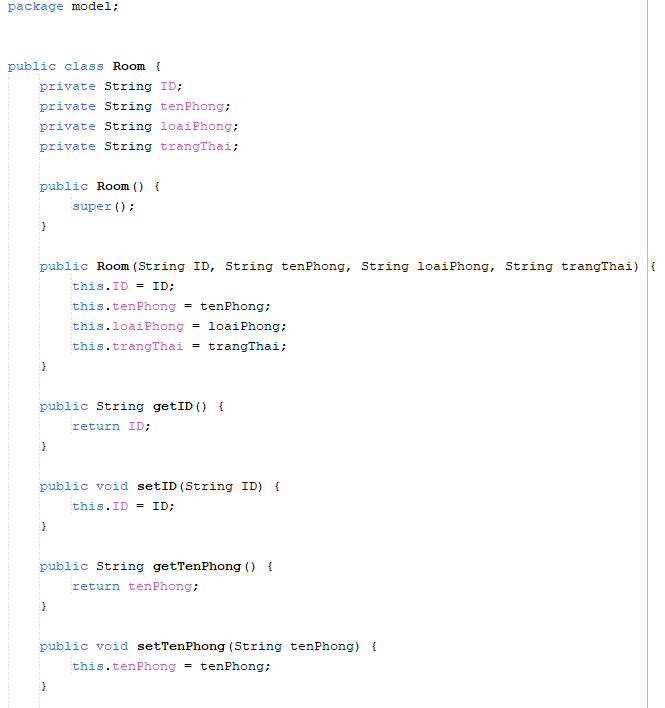
Lớp Main:



Lớp DAO:



Lớp Room:



Lớp RoomDAO



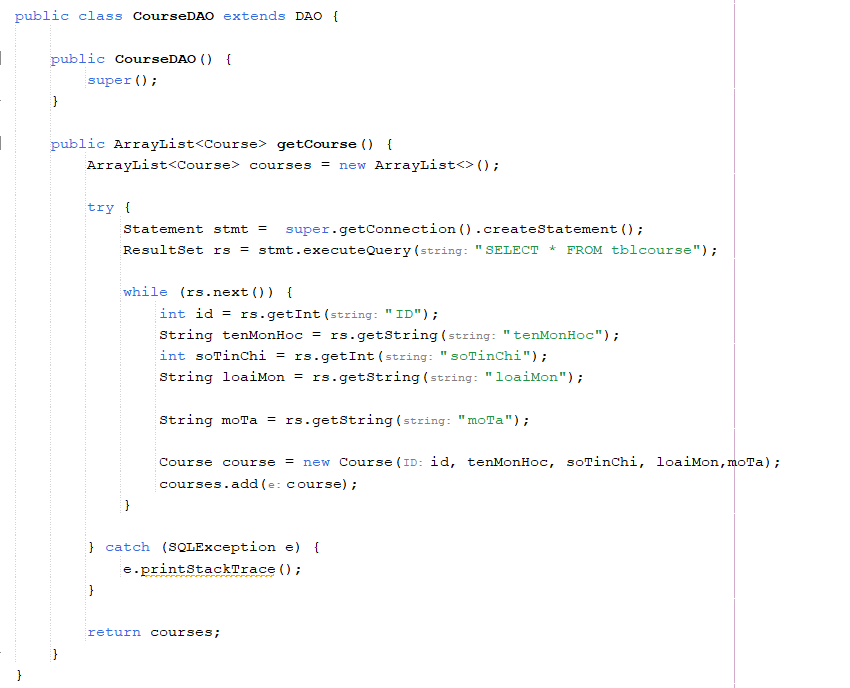
Lớp ScheduleDAO:



Lớp TimeDAO:



Lớp CourseDAO



Lớp Schedule:



Lớp AdminHomeFrm:

package baitap;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

import javax.swing.event.ChangeEvent;

import javax.swing.event.ChangeListener;

public class AdminHomeFrm extends JFrame implements ChangeListener {

private JTabbedPane tabbedPane;

private JPanel btnlenLichChoLopHocPhan, btnQuanLyTaiKhoan,btnQuanLyThongTinSinhVien;

private JLabel tenQuanTriVien, trangChu;

public AdminHomeFrm() {

super("Quản trị viên-MVC");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(1200, 700);

setLocationRelativeTo(null); // Center screen

// Khởi tạo các component

tenQuanTriVien = new JLabel("Quản trị viên A");

trangChu = new JLabel("TRANG CHỦ QUẢN TRỊ VIÊN",

SwingConstants.CENTER);

tabbedPane = new JTabbedPane();

btnlenLichChoLopHocPhan = new JPanel();

btnQuanLyTaiKhoan = new JPanel();

btnQuanLyThongTinSinhVien = new JPanel();

// Thêm tab

tabbedPane.addTab("Quản lý tài khoản", btnQuanLyTaiKhoan);

tabbedPane.addTab("Quản lý thông tin sinh viên", btnQuanLyThongTinSinhVien);

tabbedPane.addTab("Lên lịch cho lớp học phần", btnlenLichChoLopHocPhan);

// Tạo panel chứa phần tiêu đề

JPanel topPanel = new JPanel(new BorderLayout());

topPanel.setBorder(BorderFactory.createEmptyBorder(10, 10, 10, 10));

topPanel.add(trangChu, BorderLayout.CENTER);

topPanel.add(tenQuanTriVien, BorderLayout.EAST);

tabbedPane.addChangeListener(this);

// Layout chính

setLayout(new BorderLayout());

add(topPanel, BorderLayout.NORTH);

add(tabbedPane, BorderLayout.CENTER);

setVisible(true);

}

@Override

public void stateChanged(ChangeEvent e) {

int selectedIndex = tabbedPane.getSelectedIndex();

String selectedTitle = tabbedPane.getTitleAt(selectedIndex);

if (selectedTitle.equals("Lên lịch cho lớp học phần")) {

new MakeScheduleFrm().setVisible(true);

}

}

}

Lớp MakeScheduleFrm:

package baitap;

import controller.\*;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.Date;

import model.\*;

public class MakeScheduleFrm extends JFrame implements ActionListener {

private JLabel trangChu;

private JComboBox cbMonHoc, cbLopHocPhan, cbPhong, cbKhungGio;

private JTextField txtNgayBatDau, txtNgayKetThuc;

private JButton btnXacNhan;

public MakeScheduleFrm() {

super("Lên lịch học");

setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

setSize(1200, 700);

setLocationRelativeTo(null);

// Đặt layout thành null để căn chỉnh thủ công

setLayout(null);

// Khởi tạo các component

trangChu = new JLabel("Lên lịch cho lớp học phần", SwingConstants.CENTER);

cbMonHoc = new JComboBox<>();

cbMonHoc.addItem("");

for (Course course : new CourseDAO().getCourse()) {

cbMonHoc.addItem(course);

}

cbMonHoc.addActionListener(this);

cbLopHocPhan = new JComboBox<>();

cbLopHocPhan.addItem("");

cbPhong = new JComboBox<>();

cbPhong.addItem("");

for (Room room : new RoomDAO().getRoom()) {

cbPhong.addItem(room);

}

cbKhungGio = new JComboBox<>();

cbKhungGio.addItem("");

for (Time time : new TimeDAO().getTime()) {

cbKhungGio.addItem(time);

}

txtNgayBatDau = new JTextField();

txtNgayKetThuc = new JTextField();

btnXacNhan = new JButton("Xác nhận");

int labelWidth = 120;

int inputWidth = 300;

int rowHeight = 30;

int startY = 50;

int spacing = 10;

int startX = getWidth() / 2 - 200;

// Đặt vị trí và kích thước cho các component

trangChu.setBounds(10, 10, getWidth() - 20, 30); // Căn ngang JLabel lblMonHoc = new JLabel("Môn học");

lblMonHoc.setBounds(startX, startY, labelWidth, rowHeight);

cbMonHoc.setBounds(startX + labelWidth + spacing, startY, inputWidth, rowHeight);

add(lblMonHoc);

add(cbMonHoc);

startY += rowHeight + spacing;

JLabel lblLopHocPhan = new JLabel("Lớp học phần");

lblLopHocPhan.setBounds(startX, startY, labelWidth, rowHeight);

cbLopHocPhan.setBounds(startX + labelWidth + spacing, startY, inputWidth, rowHeight);

add(lblLopHocPhan);

add(cbLopHocPhan);

startY += rowHeight + spacing;

JLabel lblPhong = new JLabel("Phòng học");

lblPhong.setBounds(startX, startY, labelWidth, rowHeight);

cbPhong.setBounds(startX + labelWidth + spacing, startY, inputWidth, rowHeight);

add(lblPhong);

add(cbPhong);

startY += rowHeight + spacing;

JLabel lblNgayBatDau = new JLabel("Ngày bắt đầu");

lblNgayBatDau.setBounds(startX, startY, labelWidth, rowHeight);

txtNgayBatDau.setBounds(startX + labelWidth + spacing, startY, inputWidth, rowHeight);

add(lblNgayBatDau);

add(txtNgayBatDau);

startY += rowHeight + spacing;

JLabel lblNgayKetThuc = new JLabel("Ngày kết thúc");

lblNgayKetThuc.setBounds(startX, startY, labelWidth, rowHeight);

txtNgayKetThuc.setBounds(startX + labelWidth + spacing, startY, inputWidth, rowHeight);

add(lblNgayKetThuc);

add(txtNgayKetThuc);

startY += rowHeight + spacing;

JLabel lblKhungGio = new JLabel("Khung giờ");

lblKhungGio.setBounds(startX, startY, labelWidth, rowHeight);

cbKhungGio.setBounds(startX + labelWidth + spacing, startY, inputWidth, rowHeight);

add(lblKhungGio);

add(cbKhungGio);

startY += rowHeight + spacing;

int buttonY = startY + rowHeight + spacing + 10;

btnXacNhan.setBounds(getWidth() / 2 - 75, buttonY, 150, 30);

btnXacNhan.addActionListener(this);

add(btnXacNhan);

add(trangChu);

}

public void showMessage(String message) {

JOptionPane.showMessageDialog(this, message);

}

@Override

public void actionPerformed(ActionEvent e) {

Object object = e.getSource();

if (object == cbMonHoc) {

for (ClassSection classSection : new ClassSectionDAO()

.getClassSectionByCourse((Course) cbMonHoc.getSelectedItem())) {

cbLopHocPhan.addItem(classSection);

}

} else if (object == btnXacNhan) {

if (cbMonHoc.getSelectedIndex() != 0

&& cbKhungGio.getSelectedIndex() != 0

&& cbPhong.getSelectedIndex() != 0

&& cbLopHocPhan.getSelectedIndex() != 0

&& !txtNgayBatDau.getText().equals("")

&& !txtNgayKetThuc.getText().equals("")) {

Schedule schedule = new Schedule();

schedule.setLopHocPhan(((ClassSection) cbLopHocPhan.getSelectedItem()).getID());

schedule.setPhongHoc(((Room) cbPhong.getSelectedItem()).getID());

schedule.setKhungGio(((Time) cbKhungGio.getSelectedItem()).getID());

try {

schedule.setNgayBatDau(Date.valueOf(txtNgayBatDau.getText()));

schedule.setNgayKetThuc(Date.valueOf(txtNgayKetThuc.getText()));

if(new ScheduleDAO().makeSchedule(schedule) == true){

showMessage("Thêm lịch mới thành công");

};

} catch (IllegalArgumentException ex) {

showMessage("Ngày không đúng định dạng yyyy-MM-dd!");

return;

}

} else {

showMessage("Cần nhập đủ thông tin");

}

}

}

}