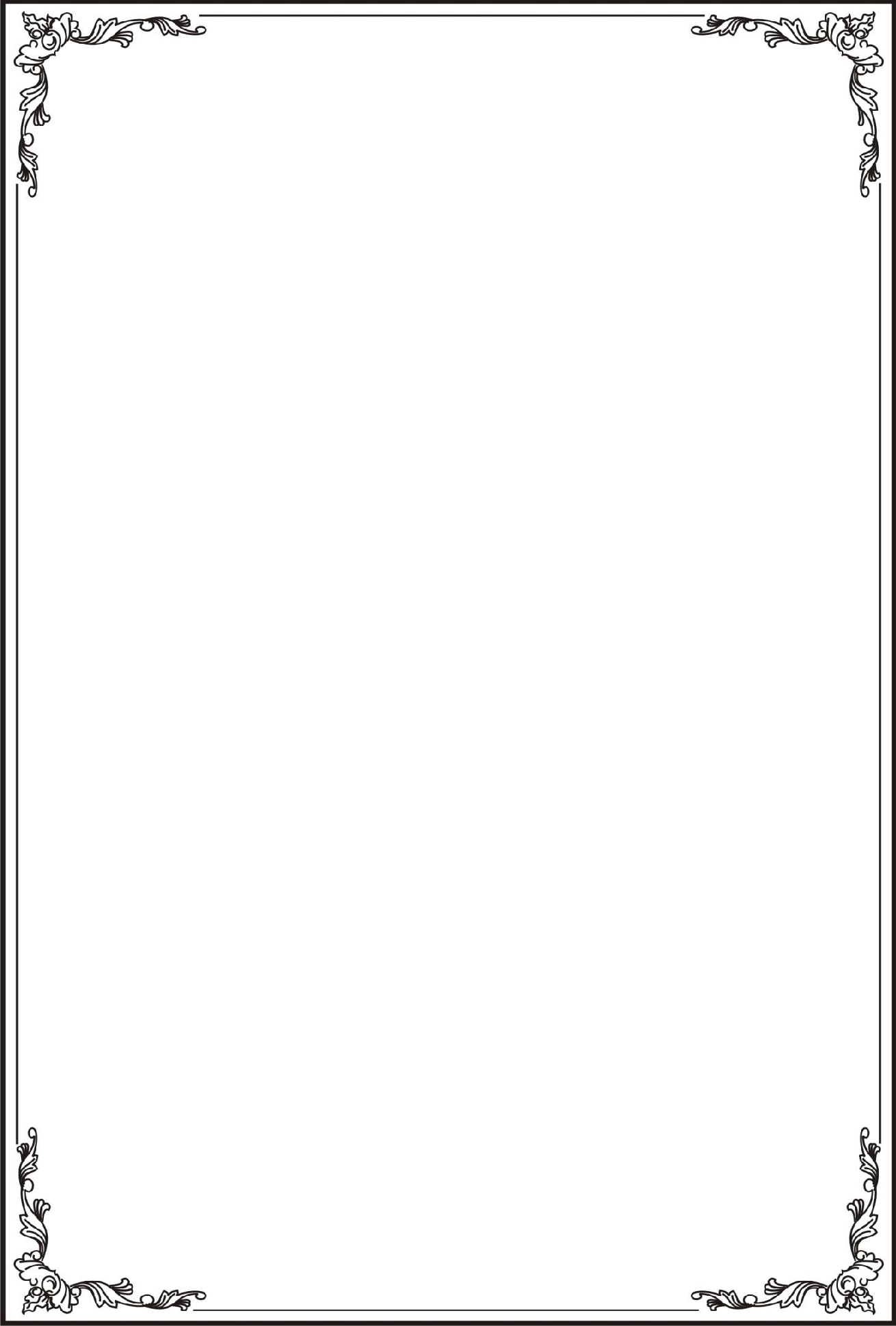
BỘ GIÁO DỤC VÀ ĐÀO TẠO



**TRƯỜNG ĐẠI HỌC THĂNG LONG**

**A red and blue logo

Description automatically generated**

**CHUYÊN ĐỀ TỐT NGHIỆP**

**ỨNG DỤNG ĐĂNG KÝ HỌC**

|  |  |
| --- | --- |
| **GIÁO VIÊN HƯỚNG DẪN** | **NHÓM SINH VIÊN THỰC HIỆN** |
| NGUYỄN THỊ HUYỀN CHÂU | A36192 NGUYỄN VĂN TRỌNG  A35380 VŨ VIỆT HOÀNG  A35102 NGÔ XUÂN HOÀNG  A35069 CỒ ĐỨC NGỌC MINH |

**HÀ NỘI – 2023**

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC THĂNG LONG**

**A red and blue logo

Description automatically generated**

**CHUYÊN ĐỀ TỐT NGHIỆP**

**ỨNG DỤNG ĐĂNG KÝ HỌC**

|  |  |
| --- | --- |
| **GIÁO VIÊN HƯỚNG DẪN** | **NHÓM SINH VIÊN THỰC HIỆN** |
| NGUYỄN THỊ HUYỀN CHÂU | A36192 NGUYỄN VĂN TRỌNG  A35380 VŨ VIỆT HOÀNG  A35102 NGÔ XUÂN HOÀNG  A35069 CỒ ĐỨC NGỌC MINH |

**HÀ NỘI – 2023**

**LỜI CẢM ƠN**

Lời đầu tiên, nhóm em xin chân thành cảm ơn các thầy giáo, cô giáo khoa Công nghệ Thông tin nói riêng và các thầy giáo, cô giáo của Đại học Thăng Long nói chung đã tận tình giảng dạy, chỉ dẫn nhóm em và luôn tạo điều kiện tốt nhất cho chúng em xuyên suốt quá trình học tập và rèn luyện đạo đức trong thời gian theo học tại trường.

Nhóm chúng em xin gửi lời biết ơn sâu sắc tới cô Nguyễn Thị Huyền Châu đã tạo mọi điều kiện thuận lợi và định hướng cho nhóm em trong suốt quá trình thực hiện đề tài. Nhóm em xin cảm ơn thầy đã cho chúng nhóm em những kiến thức, lời khuyên quý báu, sẵn sàng giải đáp những thắc mắc trong suốt quá trình thực hiện chuyên đề. Những lời khuyên và chỉ dẫn của cô đã giúp nhóm có thêm nhiều kiến thức và kinh nghiệm không chỉ trong việc thực hiện đồ án mà còn ứng dụng rất nhiều vào lĩnh việc nghiên cứu khoa học.

Cuối cùng, nhóm em xin chân thành cảm ơn gia đình và bạn bè, đã luôn tạo điều kiện, quan tâm, giúp đỡ, động viên nhóm em trong suốt quá trình học tập và hoàn thành khoá luận tốt nghiệp.

Mặc dù nhóm em đã cố gắng thực hiện đề tài nhưng do thời gian và trình độ có hạn, đồ án này không thể tránh được những thiếu sót. Nhóm em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các thầy cô để nhóm em có điều kiện bổ sung, nâng cao ý thức của mình, phục vụ tốt hơn cho công việc trong tương lai.

*Nhóm em xin chân thành cảm ơn!*

*Hà Nội, ngày 23 tháng 07 năm 2023*

**NHÓM SINH VIÊN**

**LỜI CAM ĐOAN**

Chúng tôi xin cam đoan về số liệu sử dụng trong khóa luận là số liệu thật, kết quả nghiên cứu trình bày trong khóa luận là sản phẩm của chúng tôi, không sao chép, copy của người khác, các nội dung trích dẫn đều ghi đầy đủ thông tin nguồn, tuân thủ qui định của Luật Bản quyền.

Chúng tôi xin chịu trách nhiệm trước nhà trường và pháp luật nếu có bất kỳ hành vi vi phạm pháp luật nào được phát hiện.

**NHÓM SINH VIÊN**

**MỤC LỤC**

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 14](#_Toc142351268)

[1.1. Sơ lược về cơ sở dữ liệu sử dụng 14](#_Toc142351269)

[1.2. Kiến trúc công nghệ chủ đạo. 14](#_Toc142351270)

[1.2.1. REST API [8,9] 15](#_Toc142351271)

[1.2.2. Giới thiệu về API 16](#_Toc142351272)

[1.3. Mô tả kiến trúc tổng thể 17](#_Toc142351273)

[1.4. Công nghệ và ngôn ngữ sử dụng 17](#_Toc142351274)

[1.4.1. MySQL 18](#_Toc142351275)

[1.4.2. Python 19](#_Toc142351276)

[1.4.3. JWT [10] 20](#_Toc142351277)

[1.4.4. React Native 21](#_Toc142351278)

[CHƯƠNG 2. phân tích và đặc tả chức năng 23](#_Toc142351279)

[2.1. Mô tả bài toán 23](#_Toc142351280)

[2.1.1. Thực trạng 23](#_Toc142351281)

[2.1.2. Mô tả hệ thống 23](#_Toc142351282)

[2.2. Xác định yêu cầu 23](#_Toc142351283)

[2.2.2. Xác định và mô tả các use case sử dụng 24](#_Toc142351284)

[2.2.3. Biểu đồ Usecase tổng quát 25](#_Toc142351285)

[2.2.4. Biểu đồ Use Case phân rã 26](#_Toc142351286)

[2.3. Đặc tả chức năng 33](#_Toc142351287)

[2.3.1. Đăng nhập 33](#_Toc142351288)

[2.3.2. Đăng xuất 38](#_Toc142351289)

[2.3.3. Thời khóa biểu toàn trường 41](#_Toc142351290)

[2.3.4. Lịch thi lại toàn trường 45](#_Toc142351291)

[2.3.5. Chương trình đào tạo 48](#_Toc142351292)

[2.3.6. Thông tin cá nhân 50](#_Toc142351293)

[2.3.7. Đăng ký học 52](#_Toc142351294)

[2.3.8. Thời khóa biểu 57](#_Toc142351295)

[2.3.9. Phiếu báo thu tiền 61](#_Toc142351296)

[2.3.10. Bảng điểm 65](#_Toc142351297)

[2.3.11. Lịch thi chính thức 67](#_Toc142351298)

[2.3.12. Lịch thi dự kiến 70](#_Toc142351299)

[2.3.13. Phiếu báo điểm 73](#_Toc142351300)

[CHƯƠNG 3. phân tích, thiết kế chức năng và cơ sở dữ liệu 77](#_Toc142351301)

[3.1. Sơ đồ trình tự 77](#_Toc142351302)

[3.1.1. Sơ đồ trình tự chung 77](#_Toc142351303)

[3.1.2. Sơ đồ tuần tự đăng nhập 78](#_Toc142351304)

[3.1.3. Sơ đồ tuần tự TKB toàn trường 79](#_Toc142351305)

[3.1.4. Sơ đồ tuần tự đăng ký học 80](#_Toc142351306)

[3.1.5. Sơ đồ tuần tự thời khóa biểu cá nhân 81](#_Toc142351307)

[3.1.6. Sơ đồ tuần tự bảng điểm 82](#_Toc142351308)

[3.1.7. Sơ đồ tuần tự phiếu báo thu tiền 83](#_Toc142351309)

[3.2. Thiết kế dữ liệu 83](#_Toc142351310)

[3.2.1. Quy tắc nghiệp vụ 83](#_Toc142351311)

[3.2.2. Biểu đồ thiết kế 84](#_Toc142351312)

[3.3. Các lớp cơ bản 88](#_Toc142351313)

[3.4. Quan hệ giữa các lớp 93](#_Toc142351314)

[3.5. Biểu đồ lớp thiết kế 97](#_Toc142351315)

[CHƯƠNG 4. CÀI ĐẶT VÀ TRIỂN KHAI 98](#_Toc142351316)

[4.1. Cài đặt môi trường 98](#_Toc142351317)

[4.1.1. FastAPI 98](#_Toc142351318)

[4.1.2. React Native 98](#_Toc142351319)

[4.2. Hướng dẫn sử dụng 98](#_Toc142351320)

[4.2.1. Đăng ký học 98](#_Toc142351321)

[4.2.2. Đóng học phí 103](#_Toc142351322)

[CHƯƠNG 5. Kết luận 109](#_Toc142351323)

[5.1. Kết quả đạt được 109](#_Toc142351324)

[5.1.1. Ứng dụng làm được những gì 109](#_Toc142351325)

[5.1.2. Kiến thức đã học: 109](#_Toc142351326)

[5.1.3. Kiến thức về nghiệp vụ 109](#_Toc142351327)

[5.1.4. Môi trường làm việc 110](#_Toc142351328)

[5.2. Hướng phát triển 110](#_Toc142351329)

[TÀI LIỆU THAM KHẢO 111](#_Toc142351330)

**DANH MỤC MINH HỌA**

[Bảng 2.1. Bảng danh sách các tác nhân của hệ thống 24](#_heading=h.3whwml4)

[Bảng 2.2. Bảng danh sách các use case sử dụng 25](#_heading=h.qsh70q)

[Hình 1.1. Sơ đồ kiến trúc tổng thể 14](#_heading=h.3znysh7)

[Hình 1.2. REST API 15](#_heading=h.tyjcwt)

[Hình 1.3. API 16](#_heading=h.1t3h5sf)

[Hình 1.4. Open API 16](#_heading=h.4d34og8)

[Hình 1.5. Công nghệ MySQL 18](#_heading=h.26in1rg)

[Hình 1.6. Công nghệ Python 19](#_heading=h.35nkun2)

[Hình 1.7. JWT 20](#_heading=h.44sinio)

[Hình 1.8. Công nghệ React Native 21](#_heading=h.z337ya)

[Hình 2.1. Biểu đồ Usecase tổng quát 25](#_heading=h.1pxezwc)

[Hình 2.2. Biểu đồ phân rã UC đăng nhập 26](#_heading=h.2p2csry)

[Hình 2.3. Biểu đồ phân rã UC đăng xuất 26](#_heading=h.147n2zr)

[Hình 2.4. Biểu đồ phân rã UC thời khóa biểu toàn trường 27](#_heading=h.3o7alnk)

[Hình 2.5. Biểu đồ phân rã UC lịch thi lại toàn trường 27](#_heading=h.23ckvvd)

[Hình 2.6. Biểu đồ phân rã UC chương trình đào tạo 28](#_heading=h.ihv636)

[Hình 2.7. Biểu đồ phân rã UC thông tin cá nhân 28](#_heading=h.32hioqz)

[Hình 2.8. Biểu đồ phân rã UC đăng ký học 29](#_heading=h.1hmsyys)

[Hình 2.9. Biểu đồ phân rã UC thời khóa biểu 30](#_heading=h.41mghml)

[Hình 2.10. Biểu đồ phân rã UC phiếu báo thu tiền 31](#_heading=h.2grqrue)

[Hình 2.11. Biểu đồ phân rã UC bảng điểm 31](#_heading=h.vx1227)

[Hình 2.12. Biểu đồ phân rã UC lịch thi chính thức 32](#_heading=h.3fwokq0)

[Hình 2.13. Biểu đồ phân rã UC lịch thi dự kiến 32](#_heading=h.1v1yuxt)

[Hình 2.14. Biểu đồ phân rã UC phiếu báo điểm 33](#_heading=h.4f1mdlm)

[Hình 2.15. Giao diện đăng nhập 35](#_heading=h.3tbugp1)

[Hình 2.16. Giao diện đăng nhập sai mật khẩu 36](#_heading=h.28h4qwu)

[Hình 2.17. Giao diện đăng nhập khi không nhập tài khoản 37](#_heading=h.nmf14n)

[Hình 2.18. Giao diện đăng nhập khi không nhập mật khẩu 38](#_heading=h.37m2jsg)

[Hình 2.19. Giao diện tài khoản 40](#_heading=h.46r0co2)

[Hình 2.20. Giao diện đăng xuất 41](#_heading=h.2lwamvv)

[Hình 2.21. Giao diện chọn học kỳ của thời khóa biểu toàn trường 43](#_heading=h.3l18frh)

[Hình 2.22. Giao diện thời khóa biểu toàn trường 44](#_heading=h.206ipza)

[Hình 2.23. Giao diện khi nhấn một môn trong thời khóa biểu toàn trường để hiện thông tin chi tiết 45](#_heading=h.4k668n3)

[Hình 2.24. Giao diện chọn học kỳ của lịch thi lại toàn trường 47](#_heading=h.1egqt2p)

[Hình 2.25. Giao diện lịch thi lại toàn trường 48](#_heading=h.3ygebqi)

[Hình 2.26. Giao diện chương trình đào tạo 50](#_heading=h.sqyw64)

[Hình 2.27. Giao diện thông tin cá nhân 52](#_heading=h.1rvwp1q)

[Hình 2.28. Giao diện Đăng ký học 54](#_heading=h.2r0uhxc)

[Hình 2.29. Giao diện đăng ký thành công 55](#_heading=h.1664s55)

[Hình 2.30. Giao diện hủy đăng ký 56](#_heading=h.3q5sasy)

[Hình 2.31. Giao diện trùng lịch 57](#_heading=h.25b2l0r)

[Hình 2.32. Giao diện hiện danh sách học kỳ của màn hình thời khóa biểu 59](#_heading=h.34g0dwd)

[Hình 2.33. Giao diện thời khóa biểu 60](#_heading=h.1jlao46)

[Hình 2.34. Giao diện chi tiết môn học trong màn hình thời khóa biểu 61](#_heading=h.43ky6rz)

[Hình 2.35. Giao diện chọn học kỳ Phiếu báo thu tiền 63](#_heading=h.xvir7l)

[Hình 2.36. Giao diện Phiếu báo thu tiền 64](#_heading=h.3hv69ve)

[Hình 2.37. Giao diện sao chép số tài khoản trường 65](#_heading=h.1x0gk37)

[Hình 2.38. Giao diện Bảng điểm 67](#_heading=h.2w5ecyt)

[Hình 2.39. Giao diện chọn học kỳ Lịch thi chính thức 69](#_heading=h.3vac5uf)

[Hình 2.40. Giao diện lịch thi chính thức 70](#_heading=h.2afmg28)

[Hình 2.41. Giao diện chọn học kỳ Lịch thi dự kiến 72](#_heading=h.39kk8xu)

[Hình 2.42. Giao diện lịch thi dự kiến 73](#_heading=h.1opuj5n)

[Hình 2.43. Giao diện chọn học kỳ của Phiếu báo điểm 75](#_heading=h.2nusc19)

[Hình 2.44. Giao diện phiếu báo điểm 76](#_heading=h.1302m92)

[Hình 3.1. Sơ đồ trình tự chung 77](#_heading=h.319y80a)

[Hình 3.2. Sơ đồ tuần tự đăng nhập 78](#_heading=h.40ew0vw)

[Hình 3.3. Sơ đồ tuần tự TKB toàn trường 79](#_heading=h.upglbi)

[Hình 3.4. Sơ đồ tuần tự đăng ký học 80](#_heading=h.1tuee74)

[Hình 3.5. Sơ đồ tuần tự thời khóa biểu 81](#_heading=h.2szc72q)

[Hình 3.6. Sơ đồ tuần tự bảng điểm 82](#_heading=h.3s49zyc)

[Hình 3.7. Sơ đồ tuần tự phiếu báo thu tiền 83](#_heading=h.meukdy)

[Hình 3.8. Biểu đồ thiết kế dạng Chen 86](#_heading=h.1ljsd9k)

[Hình 3.9. Biểu đồ thiết kế dạng Crow 89](#_heading=h.45jfvxd)

[Hình 3.10. Quan hệ giữa các lớp 97](#_heading=h.zu0gcz)

[Hình 3.11. Biểu đồ lớp thiết kế 98](#_heading=h.1yyy98l)

[Hình 4.1. Giao diện trang chủ 100](#_heading=h.3bj1y38)

[Hình 4.2. Giao diện đăng ký học 101](#_heading=h.1qoc8b1)

[Hình 4.3. Giao diện đăng ký thành công 102](#_heading=h.4anzqyu)

[Hình 4.4. Giao diện trùng lịch 103](#_heading=h.2pta16n)

[Hình 4.5. Giao diện hủy đăng ký 104](#_heading=h.14ykbeg)

[Hình 4.6. Giao diện trang chủ 105](#_heading=h.243i4a2)

[Hình 4.7. Giao diện chọn học kỳ thu tiền 106](#_heading=h.j8sehv)

[Hình 4.8. Giao diện Phiếu báo thu tiền 107](#_heading=h.338fx5o)

[Hình 4.9. Giao diện chi tiết khoản thu từng môn 108](#_heading=h.1idq7dh)

[Hình 4.10. Giao diện sao chép số tài khoản trường 109](#_heading=h.42ddq1a)

**DANH MỤC THUẬT NGỮ SỬ DỤNG**

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Ý nghĩa** |
| 1 | API | Application Programming Interface |
| 2 | REST API | REpresentational State Transfer Application Programming Interface |
| 3 | OPEN API | Open Application Programming Interface |
| 4 | MySQL | Structured Query Language |
| 5 | JWT | JSON Web Token |
| 6 | UC | Use Case |
| 7 | XML | Extensible Markup Language |
| 8 | URL | Uniform Resource Locator |
| 9 | JSON | JavaScript Object Notation |
| 10 | HTTP | Hypertext Transfer Protocol |

**BẢNG PHÂN CÔNG CÔNG VIỆC**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thành viên** | **Công việc** | **Hoàn thành** | **Điểm** |
| 1 | **Nguyễn Văn Trọng** | - Front end web  - Ứng dụng mobile  - Leader | 100% | 10 |
| 2 | **Vũ Việt Hoàng** | - Back end  - Cài đặt thuật toán | 100% | 10 |
| 3 | **Cổ Đức Ngọc Minh** | - Viết tài liệu phân tích  - Viết tài liệu đặc tả  - Viết tài liệu thiết kế CSDL  - Vẽ sơ đồ | 100% | 10 |
| 4 | **Ngô Xuân Hoàng** | - Viết lời giới thiệu  - Làm slide  - Viết tài liệu công nghệ sử dụng | 100% | 10 |

**LỜI GIỚI THIỆU**

Ngày nay, công nghệ thông tin đã có những bước phát triển mạnh mẽ theo cả chiều rộng và sâu. Những thiết bị điện tử không còn là một thứ phương tiện quý hiếm mà đang ngày càng trở thành một công cụ làm việc và giải trí thông dụng của con người ở mọi lĩnh vực khác nhau trong cuộc sống.

Hiện nay các trường đại học trên toàn thế giới không ngừng đầu tư và cải thiện các phương pháp học tập cũng như các sản phẩm nhằm tạo cho sinh viên những điều kiện học tốt nhất. Thông qua các sản phẩm và công nghệ hiện nay , chúng ta dễ dàng nhận ra tầm quan trọng và tính tất yếu của internet. Với những thao tác đơn giản trên thiết bị có nối mạng Internet bạn sẽ có tận tay những gì mình cần mà không phải mất nhiều thời gian. Ngay cả việc đăng ký học cũng vậy, sản phẩm giúp cho sinh viên dễ dàng chủ động trong việc đăng ký môn học, tìm kiếm thông tin môn học, giáo viên và thời khóa biểu, lịch thi,.... từ đó sinh viên có thể nắm rõ được thông tin cần thiết và chủ động trong quá trình học tập tại trường dễ dàng hơn.

Phần mềm đăng ký học cho phép sinh viên sử dụng các tính năng “tìm kiếm” để tìm các môn học, lịch học, lịch thi, giáo viên phụ trách.... Mỗi sinh viên phải có quyền truy cập vào Internet và một tài khoản sinh viên hợp lệ do nhà trường cung cấp để có thể sử dụng phần mềm đăng ký học. Những sản phẩm xuất hiện trực tiếp trên phần mềm sẽ là tất cả những thông tin do nhà trường cung cấp về giáo viên phụ trách môn học, thông tin môn học, lịch học, ngoài ra còn có lịch thi, phòng thi và bảng điểm để sinh viên nắm rõ kết quả học tập của mình.

Cấu trúc chuyên đề:

* Chương 1: Cơ sở lý thuyết: các công nghệ, kỹ thuật được áp dụng vào trong ứng dụng;
* Chương 2: Phân tích và đặc tả chức năng: bao gồm đặt vấn đề cùng với mô tả các tác nhân và use case, rồi sau đó đưa ra các đặc tả chức năng;
* Chương 3: Phân tích và thiết kế dữ liệu bao gồm các sơ đồ tuần tự cho chức năng và các lược đồ cho cơ sở dữ liệu;
* Chương 4: Cài đặt và triển khai gồm có mô tả cách thiết lập môi trường, cài đặt ứng dụng và một kịch bản hướng dẫn sử dụng bằng cách chụp màn hình;
* Chương 5: Kết luận đưa ra kết quả đạt được sau quá trình làm, về những điều học được, những thứ còn thiếu sót và những định hướng phát triển trong tương lai.

# CƠ SỞ LÝ THUYẾT

## Sơ lược về cơ sở dữ liệu sử dụng

Những công nghệ mà nhóm em ứng dụng để làm ra sản phẩm đăng ký học bao gồm: FastAPI, Python, MySQL, JWT, React Native. Là những công nghệ được ứng dụng phổ biến để thiết kế và phát triển phần mềm. Công nghệ này giúp việc thu thập, lưu trữ thông tin người sử dụng một cách khoa học, dễ dàng và chính xác. Tạo ra giao diện dễ sử dụng, thuận tiện và phù hợp với nhiều thiết bị khác nhau, đảm bảo cung cấp các chức năng xem, tìm kiếm và chỉnh sửa khoa học.

## Kiến trúc công nghệ chủ đạo.

A diagram of a diagram of a system

Description automatically generated

Hình 1.1. Sơ đồ kiến trúc tổng thể

Cách thức hoạt động của sơ đồ:

* React Native là Framework của JavaScript cho nền tảng mobile, framework này hỗ trợ cả 2 nền tảng IOS và Android nên dễ cập nhật và sử dụng;
* Python dùng để code backend của ứng dụng, ngôn ngữ python đã được học trong chương trình nhà trường, dễ hiểu và được sử dụng phổ biến;
* MySQL: là cơ sở dữ liệu dùng được đồng thời cả Windows và MacOS. Câu truy vấn giống với SQL server nên sinh viên học MS SQL sẽ dễ nắm bắt hơn, và MySQL còn được hỗ trợ trên Google Cloud;
* API là Google Cloud.

### REST API [8,9]

A diagram of a computer process

Description automatically generated

Hình 1.2. REST API

REST API (còn được biết với tên gọi RESTful API) là một giao diện lập trình ứng dụng (API) mà tuân thủ các ràng buộc và quy ước kiến trúc REST được sử dụng trong việc giao tiếp giữa client và server.

* Hai thành phần trong REST API:
* API (Application Programming Interface) là tập hợp các quy tắc và cơ chế mà một ứng dụng hay một thành phần nào đó có khả năng tương tác với một ứng dụng với thành phần khác. API sẽ trả về những kiểu dữ liệu phổ biến như JSON hoặc XML mà các ứng dụng cần sử dụng đến.
* REST (Representational State Transfer) là một dạng chuyển đổi cấu trúc hay kiểu kiến trúc để viết API. Nó có khả năng tạo ra sự tương tác giữa các máy với nhau thông qua phương thức HTTP đơn giản. Chức năng của REST là quy định sử dụng các phương thức HTTP và định dạng URL cho ứng dụng web.

REST API cung cấp nhiều phương thức để thao tác với dữ liệu, trong đó có 4 phương thức phổ biến nhất:

* **GET**: yêu cầu dữ liệu từ máy chủ.
* **POST**: gửi các thay đổi từ máy khách đến máy chủ khi thêm thông tin vào máy chủ, như tạo một mục mới.
* **PUT**: sửa đổi hoặc thêm vào thông tin hiện có.
* **DELETE**: xóa thông tin hiện có.

### Giới thiệu về API

#### Khái niệm

A blue and black logo

Description automatically generated

*Hình 1.3. API*

API (Application Programming Interface) là một giao diện lập trình ứng dụng mà một hệ thống máy tính hay ứng dụng cung cấp để cho phép các yêu cầu dịch vụ có thể được tạo ra từ các chương trình máy tính khác và cho phép dữ liệu có thể được trao đổi qua lại giữa chúng.

OpenAPI cho phép người dùng tích hợp các phần mềm và ứng dụng thông qua nhiều ngôn ngữ lập trình khác nhau. Open API với khả năng xử lý các ngôn ngữ lập trình cũ sẽ giảm bớt rào cản và hạn chế tối đa rủi ro mất dữ liệu từ những ứng dụng không còn được hỗ trợ.

A diagram of a software development process

Description automatically generated

*Hình 1.4. Open API*

FastAPI là một framework phát triển ứng dụng web nhanh chóng và hiệu quả với Python. Nó được xây dựng dựa trên tiêu chuẩn OpenAPI và ASGI, giúp viết các ứng dụng web chất lượng cao với mã nguồn mở. FastAPI có khả năng xử lý đồng thời, hỗ trợ validation và serialization tự động đối với các dữ liệu đầu vào và đầu ra. Nó cung cấp khả năng tự tạo tài liệu API, sử dụng các công cụ như Swagger UI, và hỗ trợ autocomplete trong các trình biên dịch. FastAPI cũng tương thích với các công nghệ phổ biến khác của Python như Pedantic, SQLAlchemy, OAuth, JWT và nhiều công nghệ khác.

#### Những đặc điểm nổi bật của API

* API sử dụng mã nguồn mở, dùng được với mọi client hỗ trợ XML, JSON.
* API có khả năng đáp ứng đầy đủ các thành phần HTTP: URI, request/response headers, caching, versioning, content format…
* Mô hình web API dùng để hỗ trợ MVC như: unit test, injection, ioc container, model binder, action result, filter, routing, controller. Ngoài ra, nó cũng hỗ trợ RESTful đầy đủ các phương thức như: GET, POST, PUT, DELETE các dữ liệu.

## Mô tả kiến trúc tổng thể

Hệ thống này được chia làm 2 phần: Client và Server. Mỗi phần đều có những chức năng và đặc điểm riêng.

**Client:**

* Đây là nơi giúp người dùng có thể nhìn thấy, nhằm mục đích tương tác, giao tiếp với Server qua giao diện người dùng.
* Công nghệ sử dụng: React Native.
* Ứng dụng sử dụng: VS Studio Code, Xcode, Android Studio.
* Thiết bị sử dụng:
* MacBook Pro M1 2020: sử dụng để code React Native.
* Iphone 13 Pro Max: sử dụng để debug ứng dụng.
* Laptop window: sử dụng để code API.

**Server:**

* Server là nơi trực tiếp xử lý mọi request của client và trả lại những thông tin nếu cần thiết.
* Các endpoints (APIs) của server sẽ được thiết kế theo phong cách RESTful, nhằm mục đích tạo nơi giao tiếp giữa các những server và giao diện người dùng.
* Server sử dụng: Google Cloud
* Công nghệ sử dụng: Python, FastAPI, MySQL.

## Công nghệ và ngôn ngữ sử dụng

### MySQL

A logo with a dolphin

Description automatically generated

*Hình 1.5. Công nghệ MySQL*

* MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở, được phát triển bởi Oracle Corporation. Nó được sử dụng phổ biến trong việc lưu trữ và truy xuất dữ liệu trong các ứng dụng web. MySQL hỗ trợ ngôn ngữ truy vấn SQL để truy vấn và quản lý cơ sở dữ liệu. Nó cung cấp tính năng bảo mật, kiểm soát đồng thời, và khả năng mở rộng linh hoạt giúp cho việc xử lý và truy xuất dữ liệu trở nên hiệu quả.
* Ưu điểm của Mysql:
* Tốc độ xử lý nhanh: MySQL được thiết kế để tối ưu hiệu suất xử lý truy vấn, cho phép xử lý một lượng lớn dữ liệu một cách nhanh chóng.
* Độ tin cậy cao: MySQL có tính năng khôi phục dữ liệu trong trường hợp xảy ra sự cố như mất điện, crash hệ thống hoặc lỗi phần cứng.
* Hỗ trợ cho nhiều hệ điều hành: MySQL có sẵn cho hầu hết các hệ điều hành phổ biến như Windows, Linux và macOS.
* Bảo mật cao: MySQL cung cấp các tính năng bảo mật như mã hóa dữ liệu, kiểm soát quyền truy cập và giám sát hoạt động của người dùng.
* Dễ sử dụng và linh hoạt: MySQL có giao diện quản lý đơn giản và dễ sử dụng, cho phép người dùng tùy chỉnh cấu hình và thực hiện các truy vấn phức tạp.
* Hỗ trợ cho các ứng dụng phổ biến: MySQL được tích hợp sẵn trong nhiều ngôn ngữ lập trình và các ứng dụng phát triển web như PHP, Python, Java và Ruby.
* Tuy nhiên, cần lưu ý rằng các ưu điểm này có thể thay đổi tuỳ theo mục đích sử dụng và môi trường triển khai của từng dự án.

### Python

A yellow and grey logo

Description automatically generated

*Hình 1.6. Công nghệ Python*

Python là một ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học. Các nhà phát triển sử dụng Python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau. Phần mềm Python được tải xuống miễn phí, tích hợp tốt với tất cả các loại hệ thống và tăng tốc độ phát triển.

* Lợi ích python mang lại:
* Dễ dàng đọc và hiểu một chương trình Python vì ngôn ngữ này có cú pháp cơ bản giống tiếng Anh.
* Python giúp cải thiện năng suất làm việc của các nhà phát triển vì so với những ngôn ngữ khác, họ có thể sử dụng ít dòng mã hơn để viết một chương trình Python.
* Python có một thư viện tiêu chuẩn lớn, chứa nhiều dòng mã có thể tái sử dụng cho hầu hết mọi tác vụ.
* Cộng đồng Python tích cực hoạt động bao gồm hàng triệu nhà phát triển nhiệt tình hỗ trợ trên toàn thế giới.
* Trên Internet có rất nhiều tài nguyên hữu ích nếu bạn muốn học Python. Ví dụ: bạn có thể dễ dàng tìm thấy video, chỉ dẫn, tài liệu và hướng dẫn dành cho nhà phát triển.
* Python có thể được sử dụng trên nhiều hệ điều hành máy tính khác nhau, chẳng hạn như Windows, macOS, Linux và Unix.
* Đặc điểm của python:
* Là một ngôn ngữ thông dịch
* Là một ngôn ngữ dễ sử dụng
* Là một ngôn ngữ linh hoạt
* Là một ngôn ngữ lập trình hướng đối tượng
* Là một ngôn ngữ cấp cao

### JWT [10]

A logo with a black background

Description automatically generated

*Hình 1.7. JWT*

JWT là viết tắt của JSON Web Token. Nó là một tiêu chuẩn mở và là một phương tiện nhỏ gọn, an toàn với URL để trình bày các khiếu nại một cách an toàn giữa hai bên. Mã thông báo này được ký điện tử để đảm bảo tính toàn vẹn của thông tin mà nó mang theo. JWT thường được sử dụng cho mục đích xác thực và ủy quyền trong các ứng dụng web và API.

JWT bao gồm ba phần: tiêu đề, tải trọng và chữ ký. Tiêu đề thường chứa siêu dữ liệu về mã thông báo, chẳng hạn như loại mã thông báo (JWT) và thuật toán được sử dụng để ký mã thông báo. Tải trọng chứa các xác nhận quyền sở hữu hoặc tuyên bố về người dùng hoặc thực thể, chẳng hạn như ID người dùng, vai trò hoặc thời gian hết hạn. Chữ ký được tạo bằng cách kết hợp tiêu đề được mã hóa, tải trọng được mã hóa và khóa bí mật chỉ được biết bởi máy chủ. Nó được sử dụng để xác minh rằng mã thông báo không bị giả mạo.

Nhìn chung, JWT cung cấp một cách an toàn và hiệu quả để truyền và xác minh khiếu nại giữa các bên, khiến nó được sử dụng rộng rãi trong các ứng dụng web và API hiện đại.

**Lý do sử dụng:** JWT là một cơ chế để xác minh chủ sở hữu của một số dữ liệu JSON. Đó là một chuỗi được mã hóa, an toàn cho URL, có thể chứa lượng dữ liệu không giới hạn (không giống như cookie) và được ký bằng mật mã. Khi một máy chủ nhận được JWT, nó có thể đảm bảo dữ liệu chứa trong đó đáng tin cậy vì nó được ký bởi nguồn. Không người trung gian nào có thể sửa đổi JWT sau khi nó được gửi. Điều quan trọng cần lưu ý là JWT đảm bảo quyền sở hữu dữ liệu chứ không đảm bảo mã hóa. Dữ liệu JSON bạn lưu trữ trong JWT có thể được nhìn thấy bởi bất kỳ ai chặn mã thông báo vì nó chỉ được tuần tự hóa, không được mã hóa.

### React Native

A blue and white logo

Description automatically generated

*Hình 1.8. Công nghệ React Native*

* React Native là các đoạn code đã được viết sẵn (framework) do công ty công nghệ Facebook phát triển. Các lập trình viên React Native là người sử dụng những framework này để phát triển nên các hệ thống, nền tảng ứng dụng trên các hệ điều hành như IOS và Android. Ngôn ngữ lập trình được sử dụng nhiều nhất là Javascript.
* Sự ra đời của React Native đã giải quyết được bài toán về hiệu năng và sự phức tạp khi trước đó người ta phải dùng nhiều loại ngôn ngữ native cho mỗi nền tảng di động. Chính vì thế lập trình React Native sẽ giúp tiết kiệm được phần lớn thời gian và công sức khi thiết kế và xây dựng nên một ứng dụng đa nền tảng. Javascript phù hợp với rất nhiều nền tảng khác nhau.
* Ưu điểm của React Native:
* Không phải dùng nhiều native nhờ đó đơn giản hóa quá trình xây dựng nền tảng
* Rút ngắn thời gian khi phát triển ứng dụng
* Tối thiểu hoá chi phí cho doanh nghiệp
* Khả năng tái sử dụng code lên đến 80%
* Mang đến trải nghiệm người dùng chất lượng hơn
* Không đòi hỏi kiến thức nền quá chuyên sâu, bất kỳ ai đam mê IT cũng có thể học lập trình React Native.

# phân tích và đặc tả chức năng

## Mô tả bài toán

### Thực trạng

Hiện nay, có nhiều ứng dụng đăng ký học trên di động được phát triển bởi các trường đại học, trung tâm đào tạo, hoặc các công ty cung cấp dịch vụ giáo dục. Các ứng dụng này đều cung cấp khả năng đăng ký môn học, xem thời khóa biểu, tra cứu thông tin về các khóa học, và thậm chí cả việc nộp học phí.

Ứng dụng đăng ký học trên điện thoại di động giúp sinh viên và học viên tiết kiệm thời gian và công sức, vì họ có thể đăng ký học và quản lý lịch trình học tập mọi lúc, mọi nơi thông qua điện thoại di động của mình. Điều này giúp người dùng linh hoạt hơn trong việc quản lý thời gian và tạo điều kiện thuận lợi cho việc học tập.

Đa số các ứng dụng đăng ký học trên mobile được thiết kế với giao diện thân thiện, dễ sử dụng và dễ tương tác. Người dùng có thể dễ dàng chọn môn học, kiểm tra số lượng chỗ còn lại, và tiến hành đăng ký chỉ bằng một vài thao tác đơn giản.

Mặc dù ứng dụng đăng ký học trên di động đã có sự phát triển tích cực, nhưng vẫn còn một số trường hợp nơi hệ thống truy cập Internet không ổn định hoặc người dùng không quen sử dụng công nghệ. Điều này khiến cho việc tiếp cận ứng dụng trở nên khó khăn và đòi hỏi sự hỗ trợ và đào tạo từ các tổ chức giáo dục để nâng cao tính phổ biến.

### Mô tả hệ thống

Hệ thống đăng ký học là hệ thống giúp sinh viên chủ động hơn trong quá trình sắp xếp thời gian học, môn học theo nguyện vọng của bản thân .

## Xác định yêu cầu

#### Xác định và mô tả các tác nhân

| **STT** | **Tác nhân** | **Mô tả** |
| --- | --- | --- |
| 1 | Sinh viên | Đây là actor đại diện cho sinh viên là những người có nhu cầu sử dụng app. Khi truy cập vào app, sinh viên có thể đăng ký học, đăng ký thi lại, xem thời khóa biểu, chương trình đào tạo, bảng điểm, phiếu báo thu tiền,... |

*Bảng 2.1. Bảng danh sách các tác nhân của hệ thống*

### Xác định và mô tả các use case sử dụng

| **Use Case** | **Chức năng** |
| --- | --- |
| UC#01: Đăng nhập | Người dùng thực hiện chức năng đăng nhập để sử dụng hệ thống.   * Đối với sinh viên, cần đăng nhập để có thể đăng ký môn học, đăng ký thi lại và nắm rõ thông tin lịch học, giáo viên và kết quả học tập. |
| UC#02: Đăng xuất | Chức năng này cho phép người dùng thoát hẳn khỏi trạng thái đăng nhập hệ thống, người dùng sau khi đã đăng xuất sẽ không thể thực hiện chức năng nào nữa, mà bắt buộc phải đăng nhập lại khi muốn sử dụng hệ thống. |
| UC#03: Thời khóa biểu toàn trường | Chức năng này cho phép người dùng xem danh sách các môn học toàn trường dựa theo học kỳ |
| UC#04: Lịch thi lại toàn trường | Chức năng này cho phép người dùng xem danh sách các môn thi lại toàn trường dựa theo năm học đã học |
| UC#05: Chương trình đào tạo | Chức năng này cho phép người dùng xem danh sách các môn học toàn trường mà sinh viên sẽ học |
| UC#06: Thông tin cá nhân | Chức năng này cho phép người dùng xem được thông tin của bản thân |
| UC#07: Đăng ký học | Chức năng này cho phép sinh viên biết được thời gian đăng ký học, cho phép đăng ký các môn đủ điều kiện |
| UC#8: Thời khóa biểu cá nhân | Chức năng này cho phép sinh viên xem thời khóa biểu các môn học đã đăng ký trong từng học kỳ |
| UC#9: Phiếu báo thu tiền | Chức năng này cho phép sinh viên xem thông tin về học phí trong từng học kỳ |
| UC#10: Bảng điểm | Chức năng này cho phép sinh viên xem thông tin về các học phần |
| UC#11: Lịch thi chính thức | Chức năng này cho phép sinh viên xem thông tin về các môn sẽ thi trong học kỳ |
| UC#12: Lịch thi dự kiến | Chức năng này cho phép sinh viên xem thông tin dự kiến về các môn sẽ thi trong học kỳ |
| UC#13: Phiếu báo điểm | Chức năng này cho phép sinh viên xem thông tin về điểm của các học phần |

*Bảng 2.2. Bảng danh sách các use case sử dụng*

### Biểu đồ Usecase tổng quát

A screenshot of a computer

Description automatically generated

*Hình 2.1. Biểu đồ Usecase tổng quát*

### Biểu đồ Use Case phân rã

#### Phân rã Use Case đăng nhập

A screen shot of a diagram

Description automatically generated

*Hình 2.2. Biểu đồ phân rã UC đăng nhập*

#### Phân rã Use Case đăng xuất

A diagram of a diagram

Description automatically generated

*Hình 2.3. Biểu đồ phân rã UC đăng xuất*

#### Phân rã Usecase thời khóa biểu toàn trường

A diagram of a diagram

Description automatically generated

*Hình 2.4. Biểu đồ phân rã UC thời khóa biểu toàn trường*

#### Phân rã Use Case lịch thi lại toàn trường

A diagram of a diagram

Description automatically generated

*Hình 2.5. Biểu đồ phân rã UC lịch thi lại toàn trường*

#### Phân rã Use Case chương trình đào tạo

A diagram of a diagram

Description automatically generated

*Hình 2.6. Biểu đồ phân rã UC chương trình đào tạo*

#### Phân rã Use Case thông tin cá nhân

A diagram of a diagram

Description automatically generated

*Hình 2.7. Biểu đồ phân rã UC thông tin cá nhân*

#### Phân rã Use Case đăng ký học

A diagram of a diagram

Description automatically generated

*Hình 2.8. Biểu đồ phân rã UC đăng ký học*

#### Phân rã Use Case thời khóa biểu

A diagram of a diagram

Description automatically generated

*Hình 2.9. Biểu đồ phân rã UC thời khóa biểu*

#### Phân rã Use Case phiếu báo thu tiền

A diagram of a diagram

Description automatically generated

*Hình 2.10. Biểu đồ phân rã UC phiếu báo thu tiền*

#### Phân rã Use Case bảng điểm

A diagram of a diagram

Description automatically generated

*Hình 2.11. Biểu đồ phân rã UC bảng điểm*

#### Phân rã Use Case lịch thi chính thức

A diagram of a diagram

Description automatically generated

*Hình 2.12. Biểu đồ phân rã UC lịch thi chính thức*

#### Phân rã Use Case lịch thi dự kiến

A diagram of a diagram

Description automatically generated

*Hình 2.13. Biểu đồ phân rã UC lịch thi dự kiến*

#### Phân rã Use Case phiếu báo điểm

A diagram of a diagram

Description automatically generated

*Hình 2.14. Biểu đồ phân rã UC phiếu báo điểm*

## Đặc tả chức năng

### Đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Đăng nhập** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng đăng nhập vào tài khoản | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Người dùng đăng nhập vào hệ thống | |
| **Hậu điều kiện** | **Thành công** | Đăng nhập thành công | |
| **Lỗi** | Đăng nhập không thành công | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn Đăng nhập để sử dụng các chức năng của ứng dụng.   * Hệ thống hiển thị bảng thông tin Đăng nhập gồm: Tên đăng nhập, mật khẩu * Nhập tên đăng nhập, mật khẩu * Nhấn nút Đăng nhập * Sau khi đăng nhập thành công, hệ thống hiển thị tài khoản của người dùng | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Nếu người dùng nhập sai tên mật khẩu: hệ thống sẽ yêu cầu điền lại thông tin. * Nếu người dùng không nhập tên đăng nhập hoặc mật khẩu: hệ thống sẽ yêu cầu điền đầy đủ thông tin. | | | |
| **Giao diện minh họa** | | | |
| *A white sign with blue and red text  Description automatically generated*  *Hình 2.15. Giao diện đăng nhập*  A screenshot of a phone  Description automatically generated  *Hình 2.16. Giao diện đăng nhập sai mật khẩu*  A screenshot of a phone  Description automatically generated  *Hình 2.17. Giao diện đăng nhập khi không nhập tài khoản*  A screenshot of a cell phone  Description automatically generated  *Hình 2.18. Giao diện đăng nhập khi không nhập mật khẩu* | | | |

### Đăng xuất

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#02** | | **Đăng xuất** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng đăng xuất khỏi tài khoản | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Vào mục Tài khoản, nhấn vào nút Đăng xuất | |
| **Hậu điều kiện** | **Thành công** | Người dùng đăng xuất khỏi hệ thống và quay lại màn hình đăng nhập. | |
| **Lỗi** | Không có lỗi | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn Đăng xuất khỏi tài khoản   * Vào mục Tài khoản * Chọn dòng Đăng xuất * Hiện thông báo xác nhận đăng xuất * Nhấn nút Đăng xuất * Sau khi đăng xuất thành công, hệ thống đăng xuất khỏi tài khoản của người dùng. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Giao diện minh họa** | | | |
| *A screenshot of a phone  Description automatically generated*  *Hình 2.19. Giao diện tài khoản*  *A screenshot of a chat  Description automatically generated*  *Hình 2.20. Giao diện đăng xuất* | | | |

### Thời khóa biểu toàn trường

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#03** | | **Thời khóa biểu toàn trường** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Ứng dụng hiển thị thời khóa biểu tổng quát cho toàn bộ trường hoặc các khoa/ngành trong trường.  Sinh viên có thể dễ dàng truy cập và xem thời khóa biểu của mình từ ứng dụng. | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Thời khóa biểu toàn trường | |
| **Hậu điều kiện** | **Thành công** | Hiển thị danh sách chi tiết các môn học toàn trường | |
| **Lỗi** | Không hiện danh sách | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn xem Thời khóa biểu toàn trường.   * Tại mục Trang chủ * Chọn Thời khóa biểu toàn trường. * Chọn học kỳ * Nhập mã môn, tên môn tìm kiếm * Kiểm tra thông tin. Nếu hợp lệ khi nhấn vào môn học hệ thống sẽ hiển thị lên thông tin chi tiết gồm Mã môn, Tên môn, Tên lớp, Thứ, Ca, Phòng học, Giáo viên. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Hệ thống không hiển thị thời khóa biểu. * Hệ thống không hiển thị năm học và học kỳ. * Không hiện mã môn, tên môn tìm kiếm. * Không hiển thị thông tin môn học. | | | |
| **Giao diện minh họa** | | | |
| A screenshot of a phone  Description automatically generated  *Hình 2.21. Giao diện chọn học kỳ của thời khóa biểu toàn trường*  *A screenshot of a phone  Description automatically generated*  *Hình 2.22. Giao diện thời khóa biểu toàn trường*  A screenshot of a phone  Description automatically generated  *Hình 2.23. Giao diện khi nhấn một môn trong thời khóa biểu toàn trường để hiện thông tin chi tiết* | | | |

### Lịch thi lại toàn trường

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#04** | | **Lịch thi lại toàn trường** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng xem lịch thi lại của toàn trường | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Lịch thi lại toàn trường | |
| **Hậu điều kiện** | **Thành công** | Hiển thị danh sách chi tiết các môn thi lại | |
| **Lỗi** | Không hiện danh sách | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn xem Lịch thi lại toàn trường.   * Tại mục Trang chủ * Chọn Lịch thi lại toàn trường. * Chọn học kỳ * Nhập mã môn, tên môn tìm kiếm * Kiểm tra thông tin. Nếu hợp lệ khi nhấn vào môn học hệ thống sẽ hiển thị lên thông tin chi tiết gồm Mã môn, Tên môn, Ca thi, Phòng thi, Ngày thi. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Hệ thống không hiển thị lịch thi. * Hệ thống không hiển thị năm học và học kỳ. * Không hiện mã môn, tên môn tìm kiếm. * Không hiển thị thông tin môn học. | | | |
| **Giao diện minh họa** | | | |
| A screenshot of a phone  Description automatically generated  *Hình 2.24. Giao diện chọn học kỳ của lịch thi lại toàn trường*  *A screenshot of a cell phone  Description automatically generated*  *Hình 2.25. Giao diện lịch thi lại toàn trường* | | | |

### Chương trình đào tạo

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#05** | | **Chương trình đào tạo** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng xem chương trình đào tạo của toàn trường | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Chương trình đào tạo | |
| **Hậu điều kiện** | **Thành công** | Hiển thị danh sách chi tiết các môn học | |
| **Lỗi** | Không hiện danh sách | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn xem Chương trình đào tạo.   * Tại mục Trang chủ * Chọn Chương trình đào tạo. * Chọn học kỳ * Kiểm tra thông tin. Nếu hợp lệ khi nhấn vào môn học hệ thống sẽ hiển thị lên thông tin chi tiết gồm Mã môn, Tên môn, Học phần tiên quyết, Tín chỉ tiên quyết, Số tín chỉ. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Hệ thống không hiển thị danh sách ngành và chương trình đào tạo * Không hiển thị thông tin môn học. | | | |
| **Giao diện minh họa** | | | |
| *A screenshot of a cell phone  Description automatically generated*  *Hình 2.26. Giao diện chương trình đào tạo* | | | |

### Thông tin cá nhân

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#06** | | **Thông tin cá nhân** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng xem thông tin cá nhân của mình | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Thông tin cá nhân | |
| **Hậu điều kiện** | **Thành công** | Hiển thị danh sách Thông tin cá nhân | |
| **Lỗi** | Không hiện danh sách | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn xem Thông tin cá nhân   * Vào mục Tài khoản * Chọn chức năng Thông tin cá nhân * Sau khi thành công, hệ thống hiển thị thông tin tài khoản của người dùng | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Hệ thống không hiển thị thông tin cá nhân * Sửa thông tin cá nhân | | | |
| **Giao diện minh họa** | | | |
| *A screenshot of a phone  Description automatically generated*  *Hình 2.27. Giao diện thông tin cá nhân* | | | |

### Đăng ký học

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#07** | | **Đăng ký học** | **Độ phức tạp: khó** |
| **Mô tả** | | Sinh viên đăng nhập vào hệ thống để đăng ký các môn được mở | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Đăng ký học | |
| **Hậu điều kiện** | **Thành công** | * Hiển thị thời gian đăng ký học. * Khi đến giờ được đăng ký học hiển thị danh sách các môn có thể đăng ký, hiển thị các lớp có thể học của môn đó. | |
| **Lỗi** | * Đăng ký không thành công. * Hủy môn đã đăng ký không thành công. | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn Đăng ký học.   * Tại mục Trang chủ * Chọn Đăng ký học. * Khi đến giờ được đăng ký học hiển thị danh sách các môn có thể đăng ký, hiển thị các lớp có thể học của môn đó. * Người dùng tích vào ô checkbox của môn học để đăng ký. * Người dùng có thể hủy môn học khi hủy tích của các ô checkbox đã chọn. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Không hiện môn đăng ký * Hủy môn đã đăng ký không thành công * Hết giờ đăng ký học:Không thể đăng ký nữa * Đăng ký không thành công: * Lớp của môn đăng ký đã đủ. * Trùng thời gian học với môn khác. | | | |
| **Giao diện minh họa** | | | |
| *A screenshot of a chat  Description automatically generated*  *Hình 2.28. Giao diện Đăng ký học*    *Hình 2.29. Giao diện đăng ký thành công*    *Hình 2.30. Giao diện hủy đăng ký*    *Hình 2.31. Giao diện trùng lịch* | | | |

### Thời khóa biểu

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#8** | | **Thời khóa biểu** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng xem thời khóa biểu | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Thời khóa biểu, chọn học kỳ | |
| **Hậu điều kiện** | **Thành công** | Hiển thị danh sách các môn học và thông tin của môn học đó | |
| **Lỗi** | Không hiển thị thời khóa biểu | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn xem Thời khóa biểu.   * Vào mục Thời khóa biểu * Chọn học kỳ * Hệ thống sẽ hiển thị lên thông tin chi tiết các môn học mà sinh viên đã đăng ký gồm Tên lớp, Thứ, Ca, Phòng học. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Không hiện danh sách học kỳ * Không hiện các môn đã đăng ký | | | |
| **Giao diện minh họa** | | | |
| *A screenshot of a cell phone  Description automatically generated*  *Hình 2.32. Giao diện hiện danh sách học kỳ của màn hình thời khóa biểu*    *Hình 2.33. Giao diện thời khóa biểu*  A screenshot of a phone  Description automatically generated  *Hình 2.34. Giao diện chi tiết môn học trong màn hình thời khóa biểu* | | | |

### Phiếu báo thu tiền

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#9** | | **Phiếu báo thu tiền** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép sinh viên xem Phiếu báo thu tiền | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Phiếu báo thu tiền, chọn học phí | |
| **Hậu điều kiện** | **Thành công** | Hiển thị danh sách hóa đơn điện tử và thông tin của hóa đơn đó | |
| **Lỗi** | Không hiển thị danh sách | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn xem Phiếu báo thu tiền.   * Tại mục Trang chủ * Chọn Phiếu báo thu tiền. * Chọn học kỳ * Hệ thống sẽ hiển thị lên thông tin chi tiết các hóa đơn điện tử của sinh viên gồm Tổng tiền học, Tình trạng, Chi tiết học phí các môn | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Không hiện danh sách hóa đơn * Tính sai học phí * Tình trạng chưa thu dù đã nộp học phí | | | |
| **Giao diện minh họa** | | | |
| *Hình 2.35. Giao diện chọn học kỳ Phiếu báo thu tiền*    *Hình 2.36. Giao diện Phiếu báo thu tiền*  *A screenshot of a phone  Description automatically generated*  *Hình 2.37. Giao diện sao chép số tài khoản trường* | | | |

### Bảng điểm

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#10** | | **Bảng điểm** | **Độ phức tạp:**  **đơn giản** |
| **Mô tả** | | Cho phép sinh viên xem Bảng điểm | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Bảng điểm | |
| **Hậu điều kiện** | **Thành công** | Hiển thị danh sách tất cả các học phần, số tín tích lũy, điểm trung bình, số tín đang đợi điểm | |
| **Lỗi** | Không hiển thị hết các môn đã học | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi tác nhân muốn xem bảng điểm của mình.   * Tại mục Trang chủ * Chọn Bảng điểm * Hệ thống sẽ hiển thị danh sách tất cả các học phần, số tín, điểm | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Không hiển thị môn đã học * Hiển thị sai thông tin các môn * Tính sai số tín tích lũy và điểm trung bình | | | |
| **Giao diện minh họa** | | | |
| *A screenshot of a phone  Description automatically generated*  *Hình 2.38. Giao diện Bảng điểm* | | | |

### Lịch thi chính thức

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#011** | | **Lịch thi chính thức** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Sinh viên đăng nhập vào hệ thống có thể xem lịch thi cho từng học kỳ | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Lịch thi chính thức, chọn Học kỳ | |
| **Hậu điều kiện** | **Thành công** | Hiển thị thời gian, địa điểm thi cho từng môn đang học trong học kỳ | |
| **Lỗi** | Hiển thị thiếu hoặc không hiển thị lịch thi các môn đang học  Hiển thị sai lịch thi | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi tác nhân muốn xem lịch thi.   * Chọn học kỳ * Hệ thống truy xuất và hiển thị thông tin về lịch thi gồm: Tên học phần, Ca thi/Mã phòng thi, Ngày thi, Tình trạng | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Không hiện danh sách học kỳ * Hiển thị thiếu hoặc không hiển thị lịch thi các môn đang học * Hiển thị sai lịch thi các môn đang học | | | |
| **Giao diện minh họa** | | | |
| *Hình 2.39. Giao diện chọn học kỳ Lịch thi chính thức*    *Hình 2.40. Giao diện lịch thi chính thức* | | | |

### Lịch thi dự kiến

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#012** | | **Lịch thi dự kiến** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Sinh viên đăng nhập vào hệ thống có thể xem lịch thi dự kiến cho từng học kỳ | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Lịch thi dự kiến, chọn Học kỳ | |
| **Hậu điều kiện** | **Thành công** | Hiển thị thời gian, địa điểm thi dự kiến cho từng môn đang học trong học kỳ | |
| **Lỗi** | * Hiển thị thiếu hoặc không hiển thị lịch thi các môn * Hiển thị sai lịch thi | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi tác nhân muốn xem lịch thi.   * Chọn học kỳ * Hệ thống truy xuất và hiển thị thông tin về lịch thi dự kiến gồm: Mã môn, Tên môn, Ca thi, Ngày thi | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Hiển thị thiếu hoặc không hiển thị lịch thi dự kiến * Hiển thị sai lịch thi dự kiến | | | |
| **Giao diện minh họa** | | | |
| *Hình 2.41. Giao diện chọn học kỳ Lịch thi dự kiến*    *Hình 2.42. Giao diện lịch thi dự kiến* | | | |

### Phiếu báo điểm

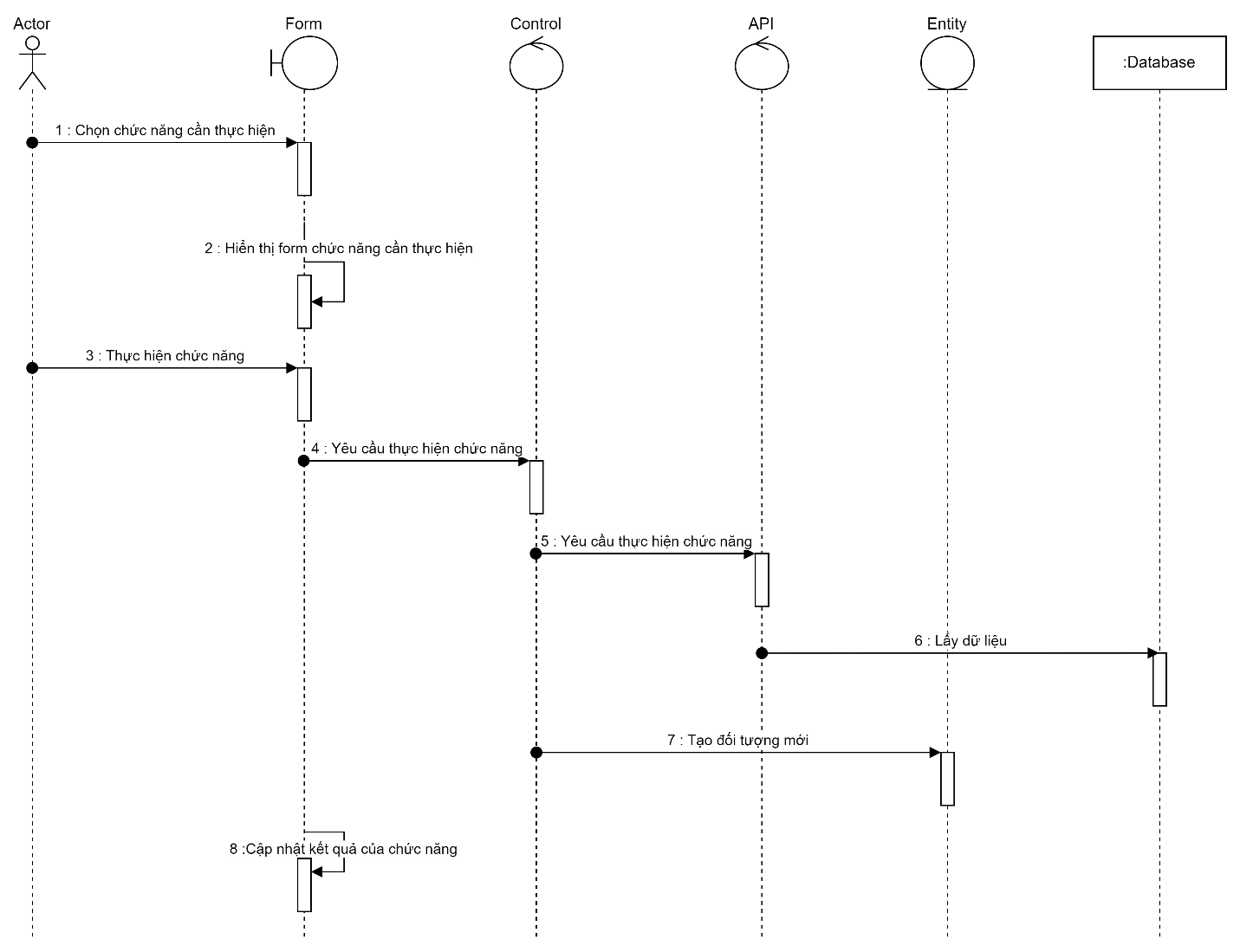
|  |  |  |  |
| --- | --- | --- | --- |
| **UC#013** | | **Phiếu báo điểm** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép sinh viên xem Phiếu báo điểm | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống, chọn chức năng Phiếu báo điểm, chọn học kỳ | |
| **Hậu điều kiện** | **Thành công** | Hiển thị danh sách các học phần, Điểm QT, Điểm cộng, Điểm KN1, Điểm KN2, Kết quả | |
| **Lỗi** | * Không hiển thị hết các môn đã học * Không hiển thị điểm các môn | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi tác nhân muốn xem bản Phiếu báo điểm của mình.   * Chọn học kỳ * Hệ thống sẽ hiển thị danh sách các học phần, Điểm QT, Điểm cộng, Điểm KN1, Điểm KN2, Kết quả | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Không hiển thị môn đã học * Không hiển thị điểm các môn * Tính sai điểm tổng | | | |
| **Giao diện minh họa** | | | |
| *Hình 2.43. Giao diện chọn học kỳ của Phiếu báo điểm*  *A screenshot of a phone  Description automatically generated*  *Hình 2.44. Giao diện phiếu báo điểm* | | | |

# phân tích, thiết kế chức năng và cơ sở dữ liệu

## Sơ đồ trình tự

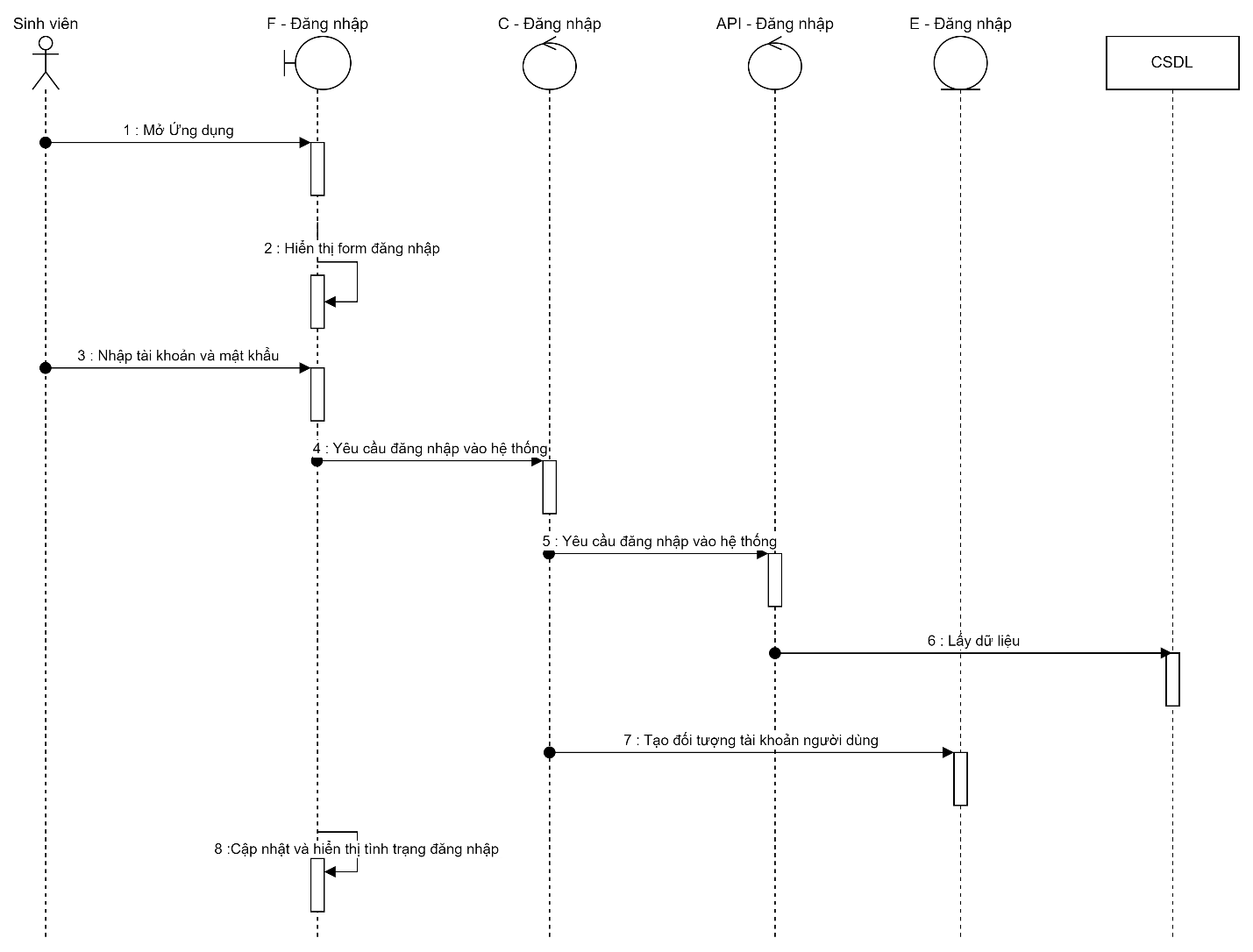
### Sơ đồ trình tự chung

* **Actor**: Tác nhân tham gia sử dụng hệ thống.
* **Form**: Lớp trung gian thể hiện sự tương tác giữa hệ thống và những gì bên ngoài hệ thống. Ở đây là giao diện giữa người dùng và hệ thống.
* **Control**: Lớp điều khiển thực hiện chức năng chính của UC.
* **Entity**: Lớp thực thể chứa những thông tin tồn tại và được lưu trữ lâu dài trong hệ thống.
* **API**: Lớp điều khiển thực hiện việc nhận/trả các API.
* **Database**: Cơ sở dữ liệu.



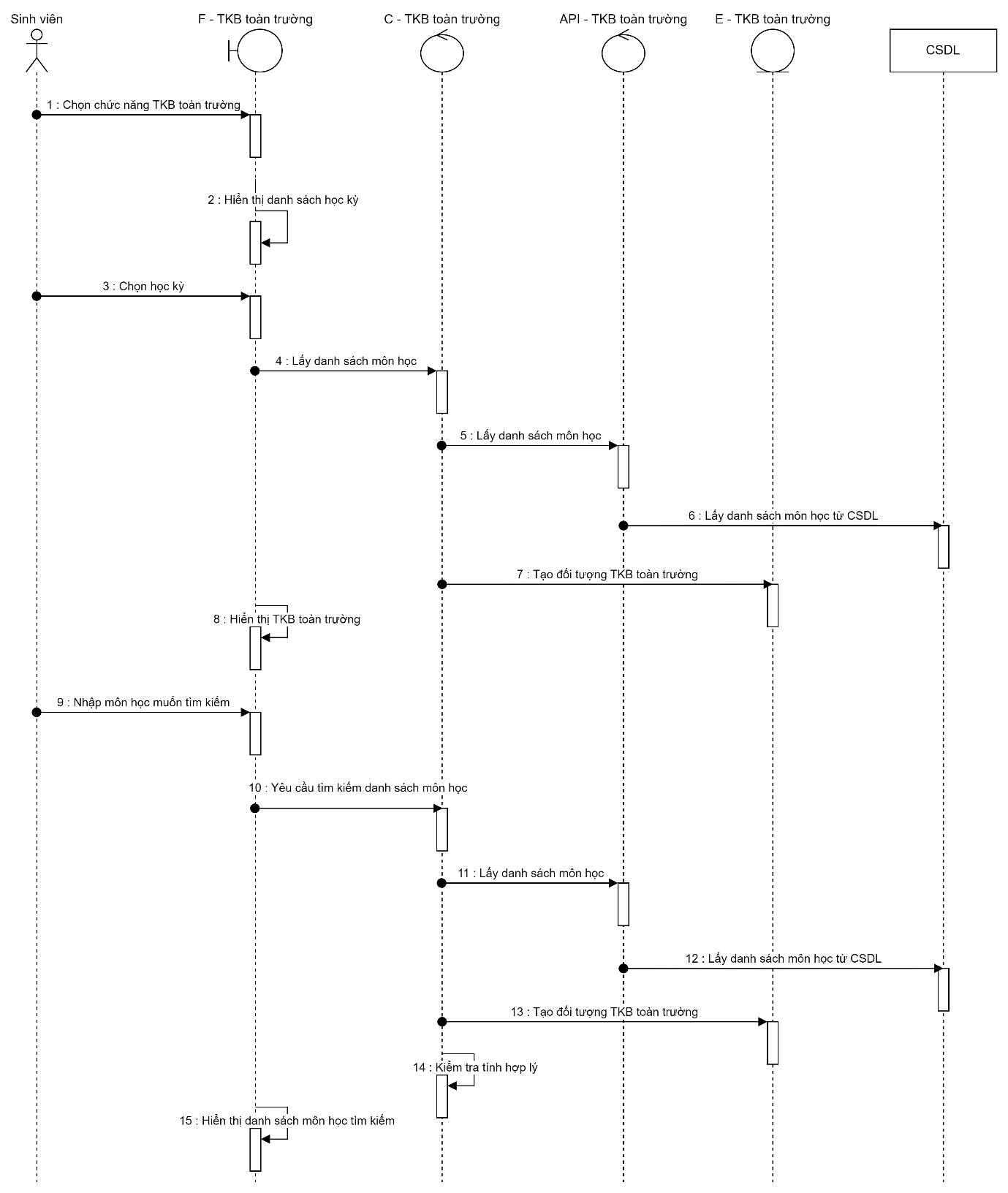
*Hình 3.1. Sơ đồ trình tự chung*

### Sơ đồ tuần tự đăng nhập



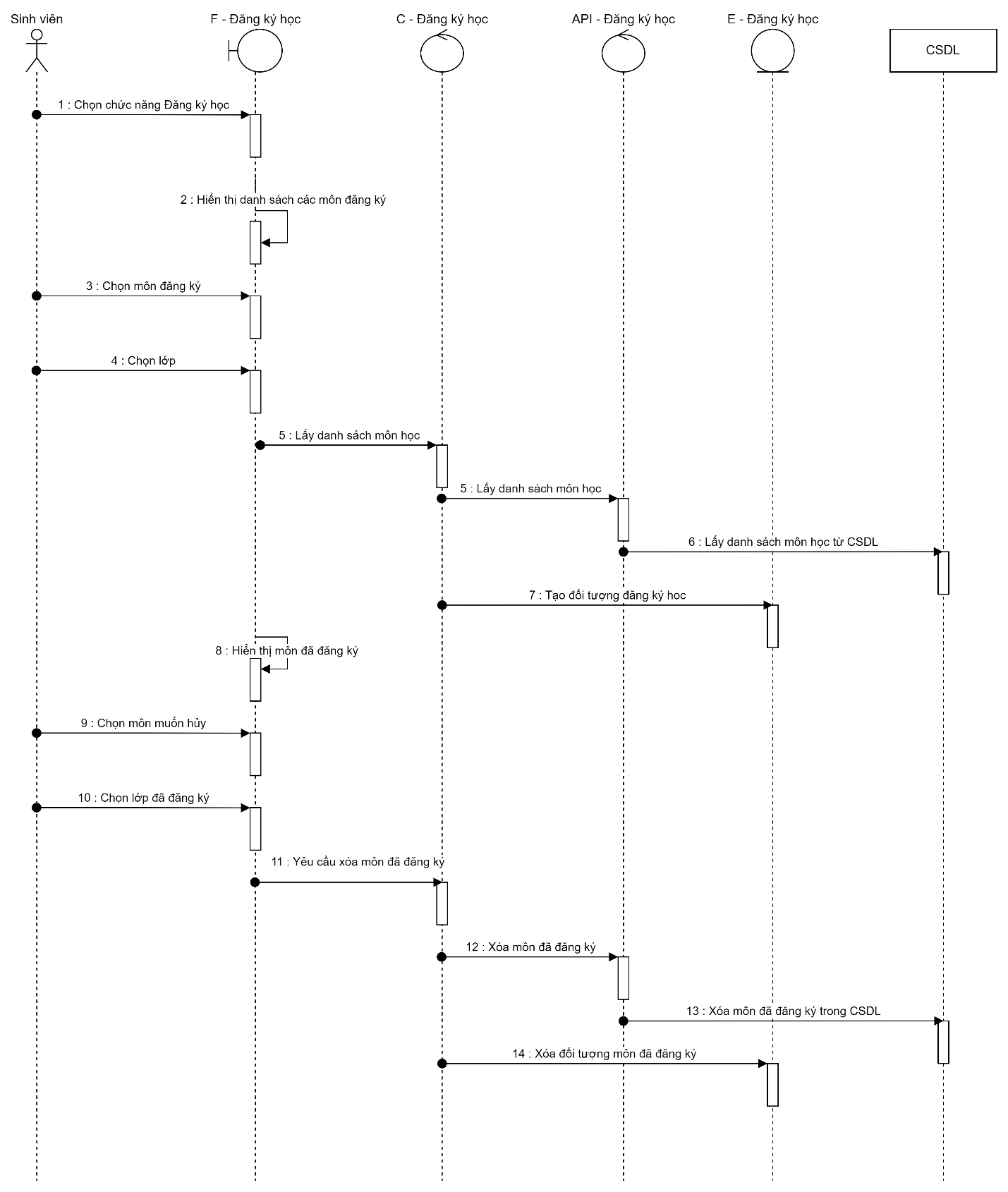
*Hình 3.2. Sơ đồ tuần tự đăng nhập*

### Sơ đồ tuần tự TKB toàn trường



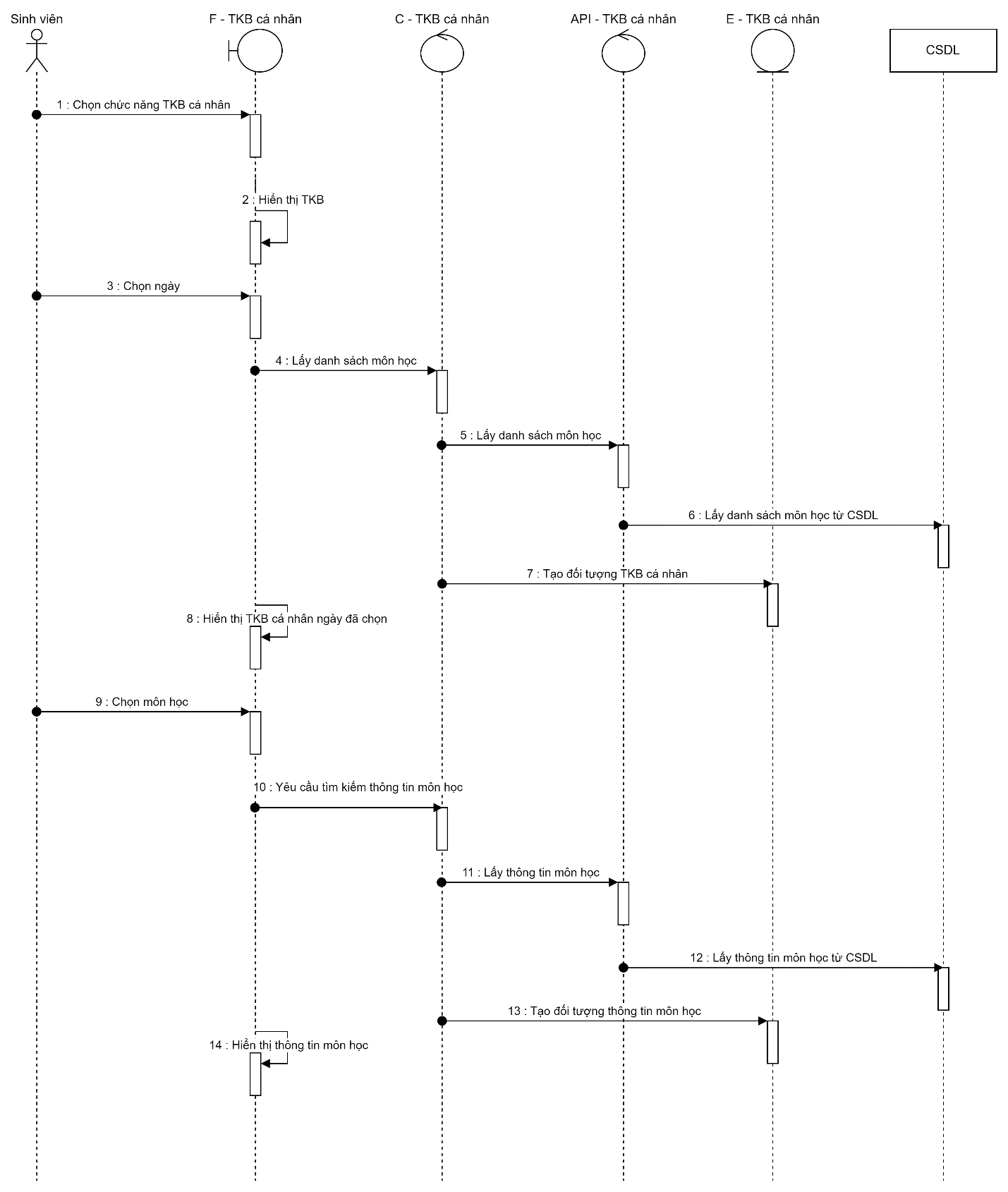
*Hình 3.3. Sơ đồ tuần tự TKB toàn trường*

### Sơ đồ tuần tự đăng ký học



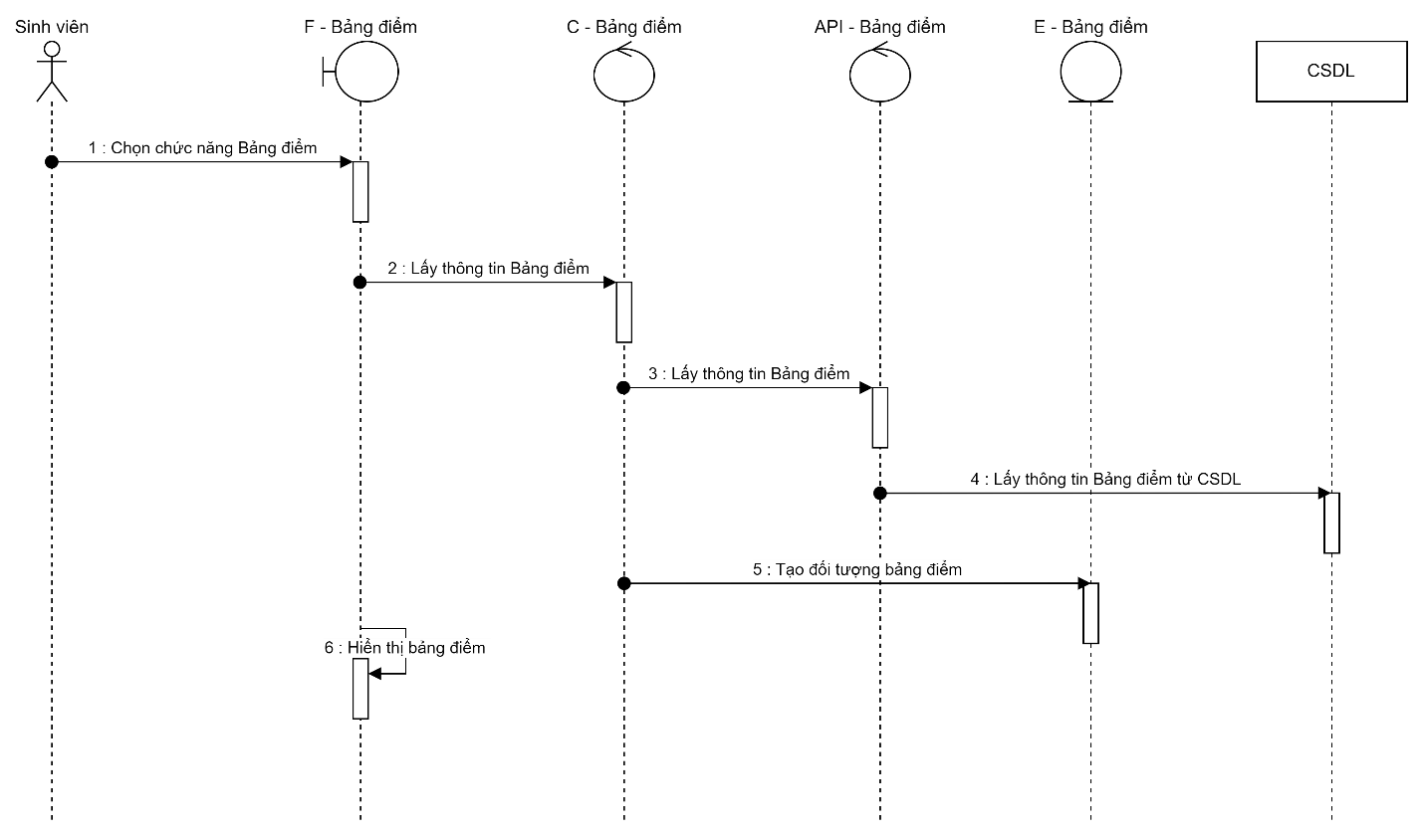
*Hình 3.4. Sơ đồ tuần tự đăng ký học*

### Sơ đồ tuần tự thời khóa biểu cá nhân



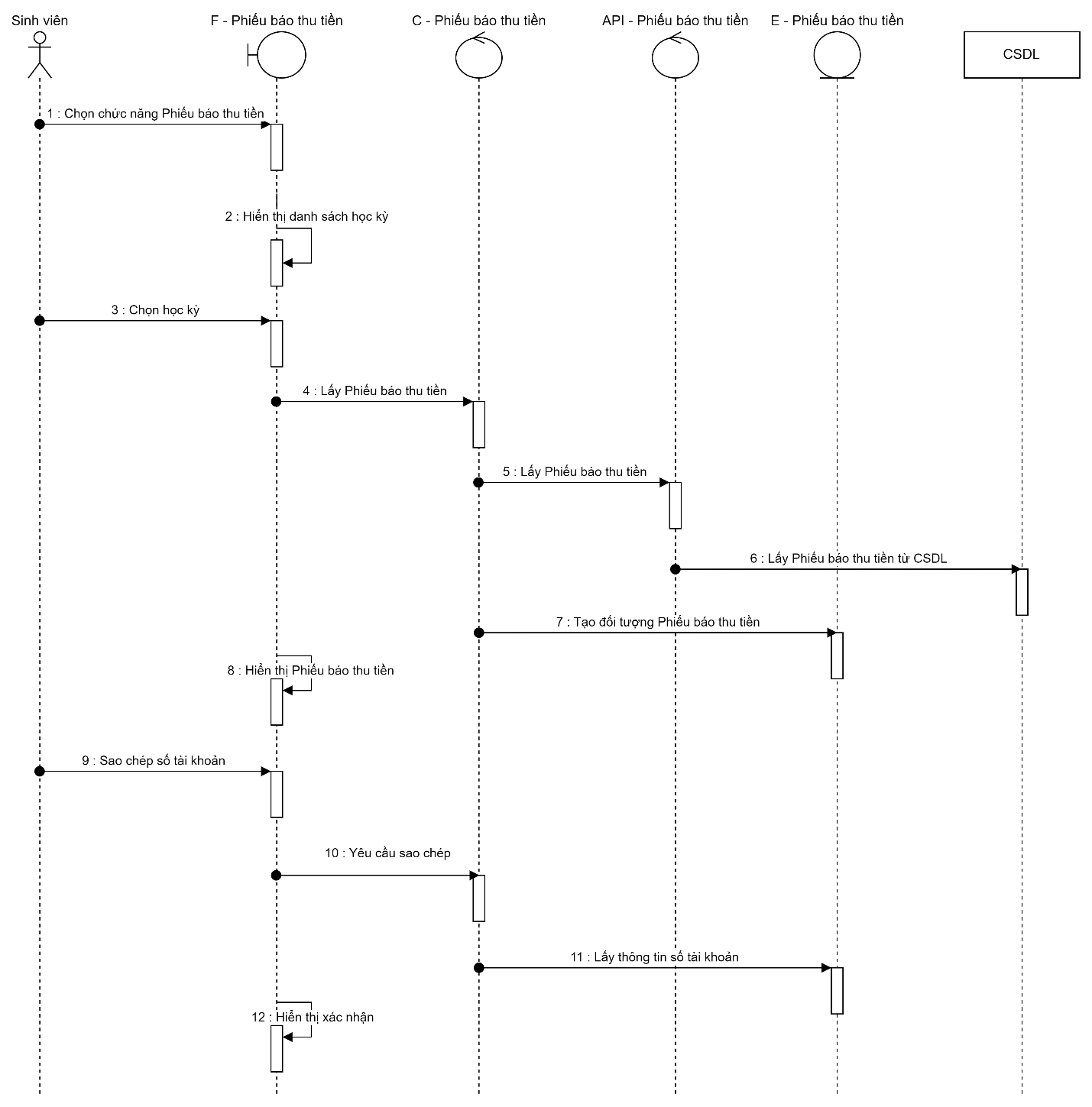
*Hình 3.5. Sơ đồ tuần tự thời khóa biểu*

### Sơ đồ tuần tự bảng điểm



*Hình 3.6. Sơ đồ tuần tự bảng điểm*

### Sơ đồ tuần tự phiếu báo thu tiền



*Hình 3.7. Sơ đồ tuần tự phiếu báo thu tiền*

## Thiết kế dữ liệu

### Quy tắc nghiệp vụ

* Mỗi sinh viên, giáo viên là một người dùng;
* Mỗi người dùng chỉ có một ảnh đại diện;
* Mỗi sinh viên học một niên khóa xác định;
* Mỗi niên khóa có nhiều kỳ;
* Mỗi kỳ mở nhiều lớp;
* Một khoa gồm có nhiều ngành;
* Mỗi sinh viên theo học ít nhất một ngành;
* Mỗi ngành có rất nhiều học phần, một học phần có thể có trong nhiều ngành;
* Mỗi học phần sẽ có nhiều kỳ thi;
* Một sinh viên phải tham gia nhiều kỳ thi, mỗi kỳ thi sẽ có nhiều sinh viên;
* Một học phần sẽ phân ra nhiều khóa học, mỗi khóa học sẽ có nhiều lớp;
* Một khóa học có thể có nhiều giáo viên, một giáo viên có thể dạy nhiều khóa học;
* Một sinh viên có thể đạt được nhiều đầu điểm, mỗi học phần sẽ có nhiều điểm.

### Biểu đồ thiết kế

#### Dạng Chen

A diagram of a computer program

Description automatically generated

A diagram of a diagram

Description automatically generated

A diagram of a student branch

Description automatically generated

A diagram of different types of geometric shapes

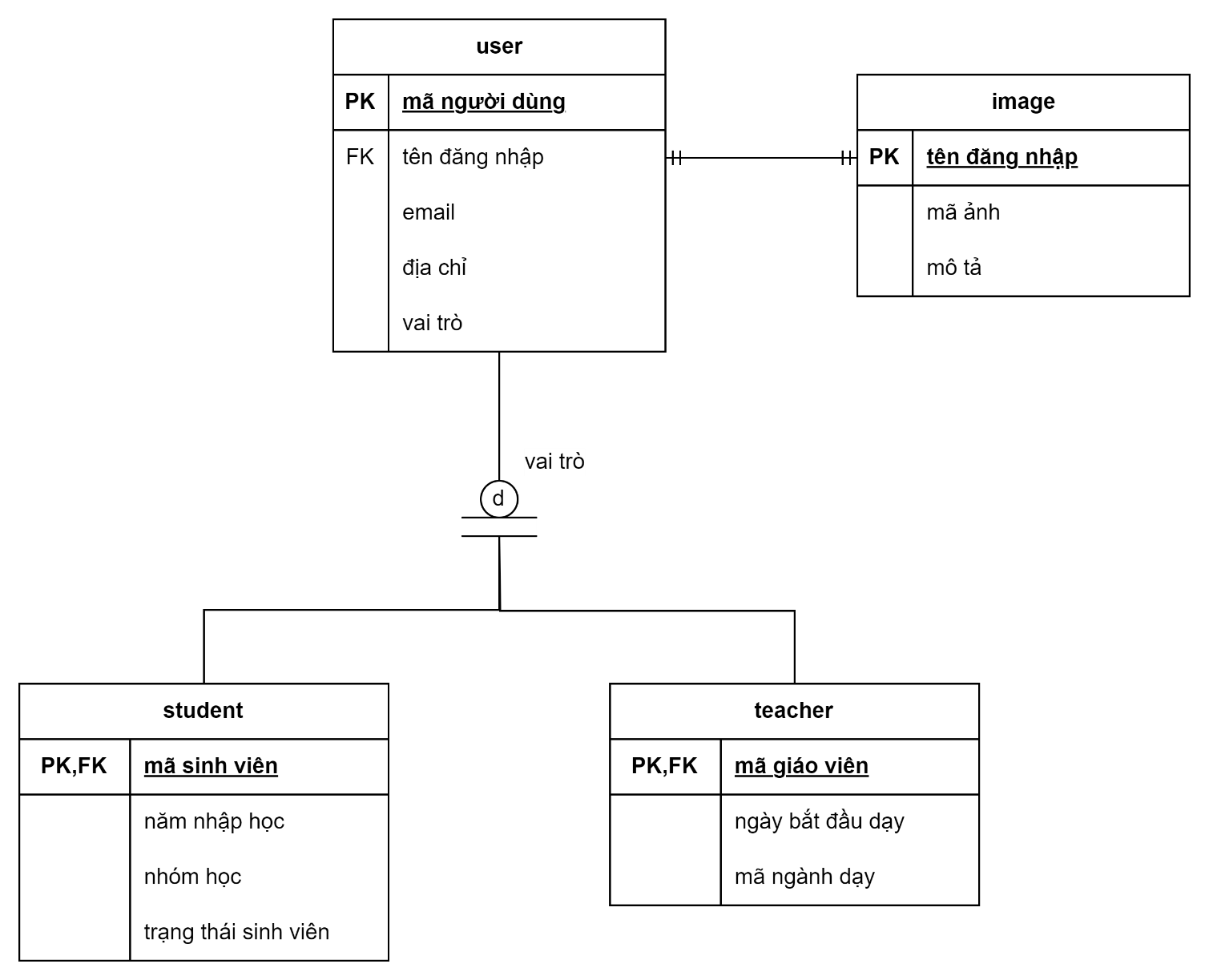
Description automatically generated

A diagram of a course

Description automatically generated

*Hình 3.8. Biểu đồ thiết kế dạng Chen*

#### Dạng Crow



A black background with white text

Description automatically generated

A black screen with white text

Description automatically generated

A black screen with white text

Description automatically generated

A black screen with white text

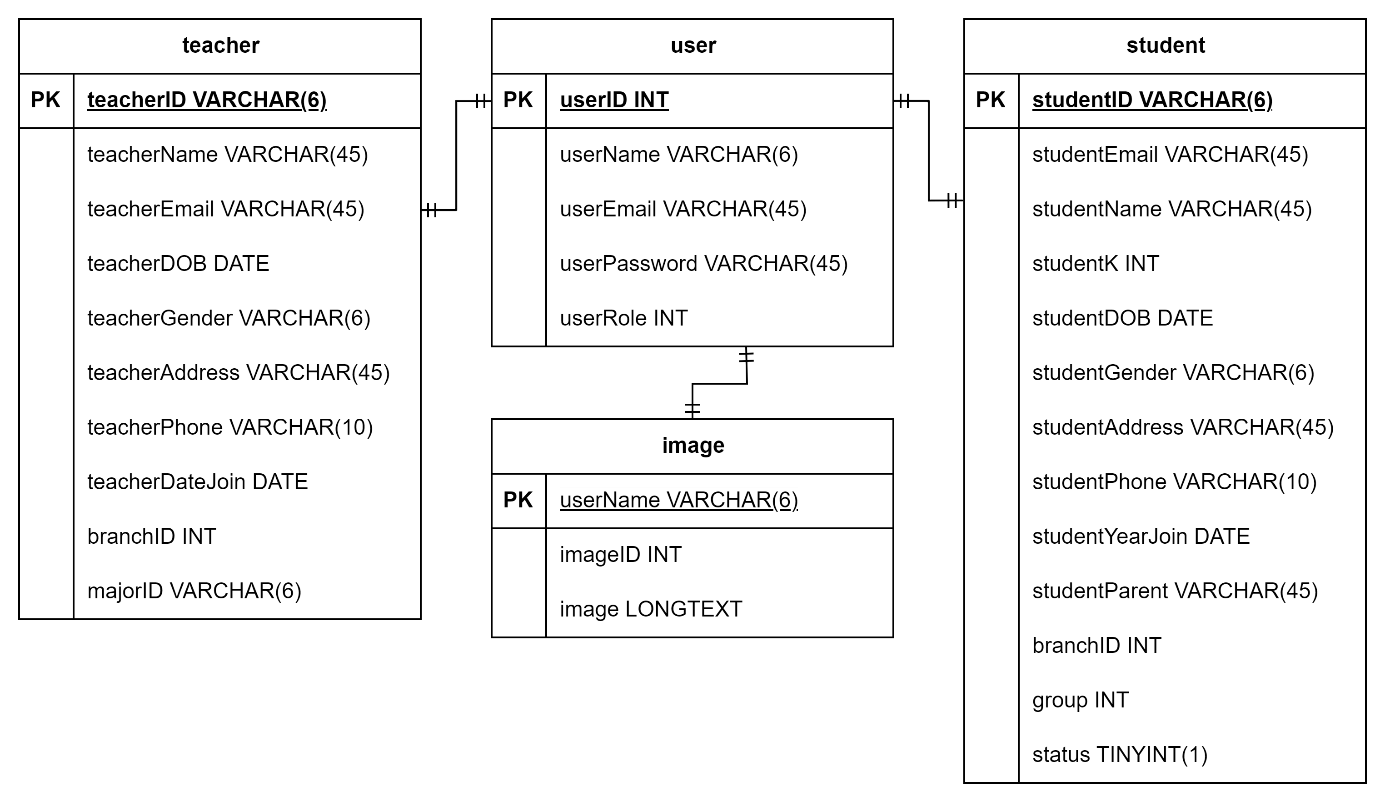
Description automatically generated

*Hình 3.9. Biểu đồ thiết kế dạng Crow*

## Các lớp cơ bản

| **Tên lớp** | **Mô tả** | |
| --- | --- | --- |
| A black background with white text  Description automatically generated | Định nghĩa | Chứa thuộc tính cơ bản của các đối tượng là người dùng hệ thống. |
| Thuộc tính | * userID: mã người dùng * userName: tên đăng nhập * userEmail: email người dùng * userPassword: mật khẩu đăng nhập * userRole: vai trò người dùng |
| A black background with white text  Description automatically generated | Định nghĩa | Hình ảnh đại diện của người dùng |
| Thuộc tính | * imageID: mã hình ảnh * userName: tên đăng nhập * image: mô tả hình ảnh |
| A black background with a black square  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về sinh viên |
| Thuộc tính | * studentID: mã sinh viên * studentEmail: email sinh viên * studentName: tên sinh viên * studentK: khóa * studentDOB: ngày sinh * studentGender: giới tính * studentAddress: địa chỉ * studentPhone: số điện thoại * studentYearJoin: năm nhập học * studentParent: phụ huynh * branchID: mã ngành học * group: nhóm học * status: trạng thái sinh viên (1: bình thường; 0: cảnh cáo; -1: đuổi học) |
| A black screen with white text  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về giáo viên |
| Thuộc tính | * teacherID: mã giáo viên * teacherName: tên giáo viên * teacherEmail: email giáo viên * teacherDOB: ngày sinh * teacherGender: giới tính * teacherAddress: địa chỉ * teacherPhone: số điện thoại * teacherDateJoin: ngày bắt đầu dạy * branchID: mã ngành dạy * majorID: mã khoa |
| A black background with white text  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về khóa học |
| Thuộc tính | * courseID: mã khóa học * subjectID: mã học phần * className: tên lớp học * courseDate: ngày học * courseShiftStart: ca học bắt đầu * courseShiftEnd: ca học kết thúc * courseRoom: phòng học * teacherID: mã giáo viên dạy |
| A black background with white text  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về học phần |
| Thuộc tính | * subjectID: mã học phần * subjectName: tên học phần * subjectCredit: số tín chỉ * Coefficient: hệ số |
| A black background with white text  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về niên khóa |
| Thuộc tính | * yearID: mã năm * yearStart: năm bắt đầu * yearEnd: năm kết thúc |
| A black background with a black square  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về nhóm học |
| Thuộc tính | * groupID: mã nhóm * groupName: tên nhóm * groupYear: năm học của nhóm * groupTerm: học kỳ của nhóm |
| A black background with white text  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về điểm |
| Thuộc tính | * gradeID: mã điểm * studentID: mã sinh viên * termID: mã học kỳ * subjectID: mã học phần * progressGrade: điểm quá trình * examGrade1: điểm kiểm tra 1 * examGrade2: điểm kiểm tra 2 * finalGrade: điểm cuối kỳ * status: trạng thái điểm (1: đăng ký; 0: chưa đăng ký; 2: xác nhận đăng ký) |
| A black background with white text  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về học kỳ |
| Thuộc tính | * id: kỳ * termID: mã kỳ * termName: tên kỳ * termStart: bắt đầu kỳ * termEnd: kết thúc kỳ * groupID: mã nhóm * yearID: mã năm |
| A black background with a black square  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về ngành |
| Thuộc tính | * branchID: mã ngành * branchName: tên ngành * majorID: mã khoa * groupEnd: phân loại nhóm từ năm 3 |
| A black background with a black square  Description automatically generated | Định nghĩa | Là lớp liên kết giữa lớp branch và lớp subject, chứa các phương thức gọi tới thuộc tính của cả 2 lớp branch, subject |
| Thuộc tính | * id: mã * subjectID: mã học phần * branchID: mã ngành |
| A black background with white text  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về kỳ thi |
| Thuộc tính | * examID: mã kỳ thi * subjectID: mã học phần * examShiftStart: ca thi bắt đầu * examShiftEnd: ca thi kết thúc * examDate: ngày thi * termID: kỳ học |
| A black background with a black square  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về lớp |
| Thuộc tính | * classID: mã lớp * courseID: mã môn * studentID: mã sinh viên * termID: mã kỳ * status: trạng thái lớp (1: đăng ký; 0: chưa đăng ký; 2: xác nhận) |
| A black background with a black square  Description automatically generated | Định nghĩa | Có thuộc tính là các thông tin về khoa |
| Thuộc tính | * majorID: mã khoa * majorName: tên khoa * majorELO: điều kiện ra trường |
| A black rectangular object with white text  Description automatically generated | Định nghĩa | Là lớp liên kết giữa lớp student và lớp exam, chứa các phương thức gọi tới thuộc tính của cả 2 lớp student, exam |
| Thuộc tính | * id: mã * studentID: mã sinh viên * examID: mã kỳ thi * status: trạng thái cấm thi (1: đăng ký; 0: chưa đăng ký; 2: xác nhận) |

## Quan hệ giữa các lớp



A black screen with white text

Description automatically generated

A black background with white rectangles

Description automatically generated

A black background with white text

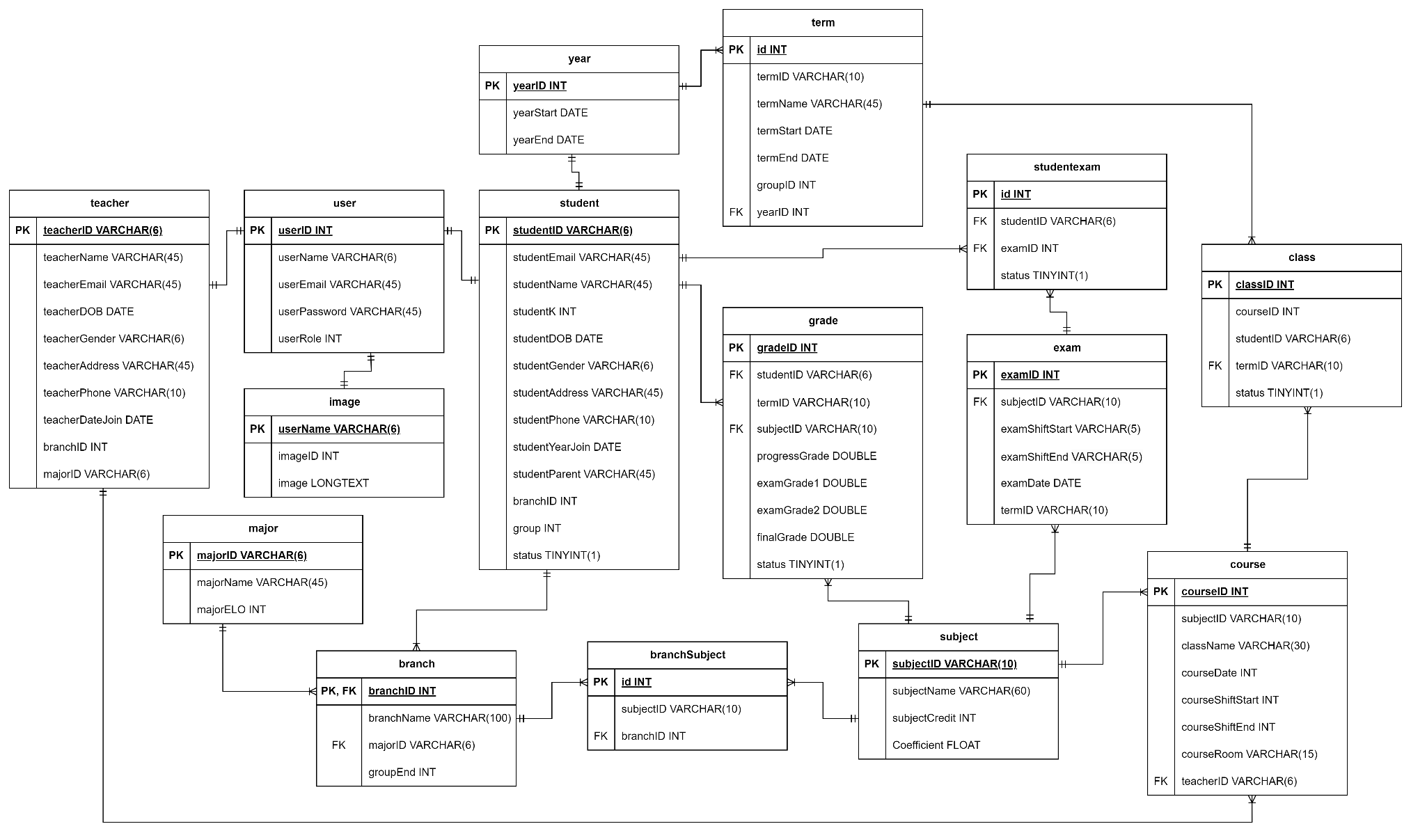
Description automatically generated

A black screen with white text

Description automatically generated

*Hình 3.10. Quan hệ giữa các lớp*

## Biểu đồ lớp thiết kế



*Hình 3.11. Biểu đồ lớp thiết kế*

# CÀI ĐẶT VÀ TRIỂN KHAI

## Cài đặt môi trường

### FastAPI

* Cài đặt Visual Studio Code
* Tải Visual Studio Code
* Cài các extension của Python như Python, Pylance
* Tạo folder chứa project
* Cài đặt python
* Tải python từ web và cài đặt
* Cài đặt FastAPI: Thông thường khi cài đặt VS code sẽ tự động có pip
* Trong VS Code, mở terminal
* Gõ py -m pip install --upgrade pip
* Sau khi update pip, gõ pip install fastapi[all]
* Cài đặt sqlalchemy
* Sau khi update pip, gõ pip install sqlalchemy[all]

### React Native

Cách dễ nhất để bắt đầu với React Native là với Expo Go. Expo là một bộ công cụ và dịch vụ được xây dựng xung quanh React Native và mặc dù nó có nhiều tính năng nhưng tính năng phù hợp nhất với chúng tôi hiện tại là nó có thể giúp bạn viết ứng dụng React Native trong vòng vài phút. Bạn sẽ chỉ cần phiên bản Node.js gần đây và điện thoại hoặc trình giả lập.

* Cài đặt Node.js: <https://nodejs.org/en>
* Cài đặt Expo Go trên điện thoại

## Hướng dẫn sử dụng

### Đăng ký học

Sinh viên đăng nhập vào ứng dụng, tại màn hình trang chủ chọn Đăng ký học:

A screenshot of a phone

Description automatically generated

*Hình 4.1. Giao diện trang chủ*

Ứng dụng sẽ hiển thị ra màn hình Đăng ký học:

A screenshot of a phone

Description automatically generated

*Hình 4.2. Giao diện đăng ký học*

Chọn môn muốn đăng ký rồi tích vào checkbox để xác nhận, sau khi thành công sẽ hiện:

A screenshot of a phone

Description automatically generated

*Hình 4.3. Giao diện đăng ký thành công*

Nếu như trùng lịch với môn đã đăng ký sẽ hiện:

A screenshot of a phone

Description automatically generated

*Hình 4.4. Giao diện trùng lịch*

Khi muốn hủy đăng ký, tích lần nữa vào ô checkbox:

A screenshot of a phone

Description automatically generated

*Hình 4.5. Giao diện hủy đăng ký*

### Đóng học phí

Sinh viên đăng nhập vào ứng dụng, tại màn hình trang chủ chọn Phiếu báo thu tiền:

A screenshot of a phone

Description automatically generated

*Hình 4.6. Giao diện trang chủ*

Sinh viên chọn học kỳ muốn đóng học phí:

A screenshot of a phone

Description automatically generated

*Hình 4.7. Giao diện chọn học kỳ thu tiền*

Ứng dụng sẽ hiển thị ra màn hình Phiếu báo thu tiền:

A screenshot of a phone

Description automatically generated

*Hình 4.8. Giao diện Phiếu báo thu tiền*

Nhấn vào môn học để xem chi tiết khoản thu:

A screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generated

*Hình 4.9. Giao diện chi tiết khoản thu từng môn*

Sinh viên cũng có thể sao chép số tài khoản của trường để chuyển khoản bằng cách nhấn vào biểu tượng bên cạnh số tài khoản:

A screenshot of a phone

Description automatically generated

*Hình 4.10. Giao diện sao chép số tài khoản trường*

# Kết luận

## Kết quả đạt được

### Ứng dụng làm được những gì

* Về phía sinh viên: Ứng dụng đăng ký học giúp cho sinh viên chủ động hơn trong việc đăng ký môn học, lịch thi lại, thi nâng điểm và nắm bắt thông tin về môn học, lịch thi, thời khóa biểu.
* Về phía giáo viên: Cập nhật điểm số, đánh giá của sinh viên lên hệ thống để sinh viên theo dõi và nắm bắt lộ trình học và kết quả học tập của sinh viên.(chưa hoàn thiện)
* Về phía người quản trị: Bảo mật thông tin người dùng, đảm bảo sự an toàn khi sử dụng, không bị lỗi khi sử dụng.(chưa hoàn thiện)

### Kiến thức đã học:

* Phân tích thiết kế: Vẽ biểu đồ lớp, use case, biểu đồ tuần tự,...
* MySQL: Phân tích và thiết kế dữ liệu,quan hệ giữa các lớp,..
* React Native: Thiết kế và tối ưu giao diện đăng ký học trên web mang lên nền tảng app
* Python: Hiểu về hướng đối tượng, truy vấn trên cơ sở dữ liệu và API

### Kiến thức về nghiệp vụ

* Phân tích và thiết kế hệ thống: Nghiên cứu và phân tích yêu cầu của khách hàng, sau đó thiết kế hệ thống phần mềm hoặc phần cứng để đáp ứng các yêu cầu đó.
* Phát triển phần mềm: Viết mã, kiểm tra và triển khai các ứng dụng phần mềm. Các kỹ năng cần thiết bao gồm lập trình, quản lý mã nguồn và kiểm thử.
* Quản lý dự án: Quản lý và điều phối các dự án CNTT. Các nhiệm vụ bao gồm lập kế hoạch, điều phối công việc và giám sát tiến độ của dự án.
* Quản trị hệ thống: Quản lý và duy trì hệ thống mạng, máy chủ và phần cứng CNTT. Nhiệm vụ bao gồm cấu hình, bảo mật và giám sát hệ thống.
* Hỗ trợ người dùng: Giúp đỡ người dùng sử dụng và khắc phục sự cố trong việc sử dụng các ứng dụng và thiết bị CNTT.
* An toàn thông tin: Đảm bảo bảo mật thông tin và bảo vệ khỏi các mối đe dọa mạng và tấn công từ bên ngoài. Các nhiệm vụ bao gồm phân tích rủi ro, triển khai biện pháp bảo mật và xử lý các sự cố bảo mật.

### Môi trường làm việc

* MySQL, Python, Fast API, JWT, React Native

## Hướng phát triển

Phát triển nâng cao những chức năng đã có: Hiện những môn không thuộc chuyên ngành của sinh viên, giới hạn sinh viên theo môn học đăng ký, theo lớp và phân quyền môn học.

Phát triển thêm những chức năng chưa có: Quên mật khẩu, xác thực sinh trắc, làm thẻ sinh viên online, đăng ký thi lại và thi các môn đại cương.

Phát triển phần mềm đăng ký học:

* Lựa chọn công nghệ: Chọn công nghệ phù hợp để xây dựng phần mềm, tính đa dụng, thuận tiện để đáp ứng được phát triển phần mềm lâu dài và nhu cầu của người sử dụng.
* Phát triển và kiểm thử: Bắt đầu phát triển phần mềm theo thiết kế và sử dụng các công cụ và kỹ thuật phát triển phần mềm phù hợp. Kiểm tra phần mềm thường xuyên để đảm bảo tính đúng đắn và hiệu suất.
* Triển khai và triển khai: Đóng gói phần mềm và triển khai nó trên môi trường hoạt động thực tế. Kiểm tra lại để đảm bảo rằng phần mềm hoạt động như mong đợi trong môi trường sản xuất.
* Hỗ trợ và duy trì: Sau khi triển khai, tiếp tục hỗ trợ người dùng, giải quyết sự cố và cải thiện phần mềm dựa trên phản hồi từ người dùng. Đảm bảo rằng phần mềm được duy trì và nâng cấp định kỳ để đáp ứng yêu cầu mới.

Quá trình phát triển phần mềm là một quá trình liên tục và cần sự phối hợp giữa các phần tử trong đội ngũ phát triển.

# TÀI LIỆU THAM KHẢO

1. <https://reactjs.org/docs/getting-started.html>
2. <https://topdev.vn/blog/mo-hinh-3-lop-la-gi/>
3. <https://bizflycloud.vn/tin-tuc/jwt-la-gi-20220524180820505.htm>
4. <https://fptcloud.com/mysql-la-gi/>
5. <https://viblo.asia/p/base64-nhung-dieu-ban-can-biet-3P0lPePp5ox>
6. <https://topdev.vn/blog/react-native/>
7. <https://fastapi.tiangolo.com/tutorial/security/oauth2-jwt/>
8. <https://www.mulesoft.com/resources/api/what-is-an-api>
9. <https://aws.amazon.com/vi/what-is/restful-api/>
10. <https://pyjwt.readthedocs.io/en/stable/>
11. [https://www.jigsawacadnemy.com/blogs/tutorial/expressjs-tutorial/](https://www.jigsawacademy.com/blogs/tutorial/expressjs-tutorial/)
12. <https://www.freecodecamp.org/news/these-are-the-concepts-you-should-know-in-react-js-after-you-learn-the-basics-ee1d2f4b8030/>