fthu

TRƯỜNG ĐẠI HỌC THĂNG LONG

**KHOA TOÁN TIN**

-----🙞🙜🕮🙞🙜-----

**BÀI TẬP LỚN**

**XÂY DỰNG PHẦN MỀM ĐĂNG KÝ HỌC**

**Giáo viên hướng dẫn Sinh viên thực hiện**

**-**TS. Mai Thúy Nga -A28631 – Hoàng Thanh Sơn

-A28998 – Vũ Tuấn Anh

-A28671 – Tạ Tuấn Hoàng

-A28449 – Đinh Xuân Thắng

-A23610 -Nguyễn Phùng Hưng

**Hà Nội - 2019**

TRƯỜNG ĐẠI HỌC THĂNG LONG

**KHOA TOÁN TIN**

-----🙞🙜🕮🙞🙜-----

**BÀI TẬP LỚN**

**XÂY DỰNG PHẦN MỀM ĐĂNG KÝ HỌC**

**Giáo viên hướng dẫn Sinh viên thực hiện**

**-**TS. Mai Thúy Nga -A28631 – Hoàng Thanh Sơn

-A28998 – Vũ Tuấn Anh

-A28671 – Tạ Tuấn Hoàng

-A28449 – Đinh Xuân Thắng

-A23610 -Nguyễn Phùng Hưng

**Hà Nội - 2019**

**LỜI CÁM ƠN**

Đầu tiên, chúng em xin chân thành cảm ơn các thầy cô giáo trong khoa Toán – Tin nói riêng và các thầy cô của trường Đại học Thăng Long nói chung đã cùng đồng hành và trang bị cho chúng em từ những kiến thức cơ bản nhất trong những năm học vừa qua.

Dự án này được hoàn thành không chỉ với sự cố gắng, tích cực của cả nhóm mà còn có giúp đỡ của cô Mai Thúy Nga, giảng viên của bộ môn Toán – Tin, trong suốt quá trình xây dựng và phát triển dự án. Chúng em xin chân thành cảm ơn về sự hướng dẫn liên tục và hỗ trợ nhiệt tình của cô.

Sau cùng chúng em cũng muốn gửi lời cảm ơn đến các bạn của các nhóm kỳ trước. Với những kinh nghiệm và tài liệu của các bạn, nhóm bọn em đã rút ngắn được thời gian xây dựng án.

Trong quá trình lên kế hoạch, xây dựng và phát triển dự án, dù cả nhóm đã hoạt động rất tích cực, rất cố gắng để đưa ra được 1 phần mềm đăng ký học hoàn thiện nhất có thể nhưng do thiếu kinh nghiệm, thời gian có hạn nên dự án sẽ có những thiếu sót không mong muốn hoặc nhóm không lường trước được. Nhóm em rất mong nhận được những ý kiến quý báu của cô cũng như của các bạn để có dự án có thể hoàn thiện hơn nữa cũng như bổ sung kiến thức, gia tăng chất lượng của các dự án tiếp theo.

Chúng em xin chân thành cảm ơn!

Hà Nội, ngày 22 tháng 3 năm 2020

*Sinh viên thực hiện*

Hoàng Thanh Sơn

Vũ Tuấn Anh

Tạ Tuấn Hoàng

Đinh Xuân Thắng

Nguyễn Phùng Hưng

**Bảng Phân Công Công Việc**

|  |  |  |
| --- | --- | --- |
| **Họ và tên** | **Nhiệm vụ** | **Đánh giá** |
| Tạ Tuấn Hoàng | Phụ trách về Front-End web(Sinh viên), thuyết trình, hỗ trợ làm slide, CSDL, báo cáo:  + Hoàn thiện sơ đồ use-case, doc thiết kế CSDL,  Chương 6  + Hỗ trợ thiết kế CSDL, thuật toán, API  + Chỉnh sửa hoặc làm slide | 95% |
| Vũ Tuấn Anh | Phụ trách về Back-end: Server, CSDL, thuật toán xử lý, các API,…  + Thiết kế thuật toán, CSDL, API  + Xây dựng back-end  + Xây dựng các chức năng quản lý tài khoản, đăng ký học, xem lịch dạy,… | 95% |
| Đinh Xuân Thắng | Phụ trách về Front-End Mobile: Client Mobile:  + Hoàn thiện chương 7 (mobile)  + Hoàn thành front-end các chức năng trên mobile:   * Xem thông tin cá nhân * Bảng điểm * Thời khóa biểu toàn trường * Đăng nhập, đăng xuất * Xem lịch dạy(giáo viên)   + Hỗ trợ thiết kế CSDL, chỉnh sửa dữ liệu nhập vào | 95% |
| Hoàng Thanh Sơn | Phụ trách về báo cáo, phân tích và thiết kế dữ liệu:  + Quản lý chính về doc, hoàn thiện chương 1 chương 2, chương 3, phần doc thiết kế API  + hỗ trợ phần doc thuật toán  + phân tích và thiết kế dữ liệu | 95% |
| Nguyễn Phùng Hưng | Phụ trách về làm slide, hỗ trợ Front-End client mobile(Giáo viên)  + Làm slide 1 số tuần  + kiểm thử giao diện web | 50% |

**MỤC LỤC**

[CHƯƠNG 1. TỔNG QUAN về phần mềm quản lý khoa học 2](#_Toc34761458)

[1.1. Mô tả bài toán 2](#_Toc34761459)

[1.1.1. Thực trạng 2](#_Toc34761460)

[1.1.2. Phân tích bài toán 2](#_Toc34761461)

[1.1.2.1 Chuẩn bị dữ liệu cho việc đăng kí học 2](#_Toc34761462)

[1.1.2.2 Thực hiện đăng ký học 2](#_Toc34761463)

[1.1.2.3 Sử dụng kết quả đăng ký học 3](#_Toc34761464)

[1.2. Yêu cầu nghiệp vụ 3](#_Toc34761465)

[1.3. Sơ đồ Usecase các chức năng chính 4](#_Toc34761466)

[1.3.1. Sơ đồ Usecase 4](#_Toc34761467)

[1.3.2. Các tác nhân hệ thống 5](#_Toc34761468)

[1.3.3. Các Use-case 5](#_Toc34761469)

[1.3.4. Bảng ánh xạ yêu cầu nghiệp vụ với các use-case: 6](#_Toc34761470)

[1.4. Kế hoạch thực hiện 7](#_Toc34761471)

[CHƯƠNG 2. kiến trúc tổng thể của hệ thống 8](#_Toc34761472)

[2.1. Sơ đồ kiến trúc tổng thể 8](#_Toc34761473)

[2.2. Database 8](#_Toc34761474)

[2.3. Server 10](#_Toc34761475)

[Phần mềm đăng ký học dùng Nodejs để tạo server 10](#_Toc34761476)

[2.4. Web Client 11](#_Toc34761477)

[CHƯƠNG 3. ĐẶC TẢ CÁC CHỨC NĂNG 14](#_Toc34761478)

[3.1. UC#01 – Quản lý dữ liệu 14](#_Toc34761479)

[3.2. UC#02 – Đăng nhập 16](#_Toc34761480)

[3.3. UC#03 – Quản lý tài khoản người dùng 18](#_Toc34761481)

[3.4. UC#04 – Quản lý thời gian đăng ký học 22](#_Toc34761482)

[3.5. UC#05 - Quản lý chuyên ngành 25](#_Toc34761483)

[3.6. UC#06 – Quản lý học phần 27](#_Toc34761484)

[3.7. UC#07 – Quản lý sinh viên 29](#_Toc34761485)

[3.8. UC#08 – Quản lý giáo viên 32](#_Toc34761486)

[3.9. UC#09 – Ép cứng lớp học phần 35](#_Toc34761487)

[3.10. UC#10- Đăng ký lớp học phần 37](#_Toc34761488)

[3.11. UC#11 - Xem điểm 40](#_Toc34761489)

[3.12. UC#12 - Xem lịch học 42](#_Toc34761490)

[3.13. UC#13 - Xem lịch giảng dạy 43](#_Toc34761491)

[3.14. Thống kê 47](#_Toc34761492)

[CHƯƠNG 4. Thuật Toán đăng ký học 48](#_Toc34761493)

[4.1. Ý tưởng thuật toán 48](#_Toc34761494)

[4.2. Chuẩn bị dữ liệu 48](#_Toc34761495)

[4.3. Thuật toán 48](#_Toc34761496)

[4.4. Sơ đồ thuật toán DKH 50](#_Toc34761497)

[CHƯƠNG 5. Thiết kế CSDL và Api 51](#_Toc34761498)

[5.1. Thiết kế CSDL 51](#_Toc34761499)

[5.1.1. Sơ đồ kiến trúc CSDL 51](#_Toc34761500)

[5.1.2. Bảng dữ liệu 52](#_Toc34761501)

[5.1.2.1 App-config\_ Lưu niên khóa, kỳ, năm 52](#_Toc34761502)

[5.1.2.2 Class-room \_ Danh sách phòng học 52](#_Toc34761503)

[5.1.2.3 Class \_ Danh sách lớp học phần 52](#_Toc34761504)

[5.1.2.4 Common-user-info \_ Danh sách thông tin người dùng của admin và phòng đào tạo 53](#_Toc34761505)

[5.1.2.5 Division \_ Danh sách bộ môn 53](#_Toc34761506)

[5.1.2.6 Dpt-ins-info \_ Thông tin của giảng viên và bộ môn 53](#_Toc34761507)

[5.1.2.7 Educate-program \_ Chương trình học 53](#_Toc34761508)

[5.1.2.8 Registration-event \_ Đợt đăng ký học 53](#_Toc34761509)

[5.1.2.9 Result\_Bảng điểm sinh viên 54](#_Toc34761510)

[5.1.2.10 Schedule\_Thời khóa biểu sinh viên 54](#_Toc34761511)

[5.1.2.11 School-schedule-items\_Thời khóa biểu toàn trường 54](#_Toc34761512)

[5.1.2.12 Shift\_Danh sách ca học 55](#_Toc34761513)

[5.1.2.13 Speciality\_Chuyên ngành 55](#_Toc34761514)

[5.1.2.14 Student-info\_Thông tin sinh viên 55](#_Toc34761515)

[5.1.2.15 Subject\_Danh sách môn học 55](#_Toc34761516)

[5.1.2.16 User\_Danh sách tài khoản 56](#_Toc34761517)

[5.2. Thiết kế API 56](#_Toc34761518)

[5.2.1. Giới thiệu chung về API 56](#_Toc34761519)

[5.2.2. Các API được sử dụng trong phần mềm Đăng ký học 57](#_Toc34761520)

[**5.2.2.1** **Chương trình học** 57](#_Toc34761521)

[**5.2.2.2** **Đăng ký học** 57](#_Toc34761522)

[**5.2.2.3** **Bảng điểm** 59](#_Toc34761523)

[**5.2.2.4** **Thời khóa biểu** 59](#_Toc34761524)

[**5.2.2.5** **Thời khóa biểu toàn trường** 60](#_Toc34761525)

[**5.2.2.6** **Tài khoản người dùng** 61](#_Toc34761526)

[CHƯƠNG 6. Thiết kế và cài đặt các chức năng 62](#_Toc34761527)

[6.1. Thiết kế các chức năng 62](#_Toc34761528)

[6.1.1. Cấu trúc chung 62](#_Toc34761529)

[6.1.2. Thiết kế chức năng đăng nhập: 63](#_Toc34761530)

[6.1.2.1 Sơ đồ lớp phân tích: 63](#_Toc34761531)

[6.1.2.2 Sơ đồ lớp trình tự: 64](#_Toc34761532)

[6.1.3. Thiết kế chức năng đăng ký học: 65](#_Toc34761533)

[6.1.3.1 Sơ đồ lớp phân tích: 65](#_Toc34761534)

[6.1.3.2 Sơ đồ lớp trình tự: 66](#_Toc34761535)

[6.2. Cài đặt các chức năng 67](#_Toc34761536)

[6.2.1. Cấu trúc thư mục mã nguồn web 67](#_Toc34761537)

[6.2.1.1 Cấu trúc chung của server 67](#_Toc34761538)

[6.2.1.2 Cấu trúc chung của client 68](#_Toc34761539)

[6.2.2. Cấu trúc thư mục mã nguồn mobile 69](#_Toc34761540)

[CHƯƠNG 7. Thiết kế, phân tích và cài đặt chức năng trên di động 73](#_Toc34761541)

[7.1. Đặc tả chức năng 73](#_Toc34761542)

[7.1.1. UC#1 - Đăng nhập 73](#_Toc34761543)

[7.1.2. UC#02 - Xem thông tin cá nhân 75](#_Toc34761544)

[7.1.3. UC#03 - Xem bảng điểm 78](#_Toc34761545)

[7.1.4. UC#04 - Xem lịch học(dạy) 81](#_Toc34761546)

[7.1.5. UC#05 - Xem thời khóa biểu toàn trường 84](#_Toc34761547)

[7.1.6. UC#06 - Đăng xuất 87](#_Toc34761548)

[7.2. Cấu trúc thư mục mã nguồn 90](#_Toc34761549)

[CHƯƠNG 8. KIỂM THỬ 92](#_Toc34761550)

[8.1. Giới thiệu chung 92](#_Toc34761551)

[8.1.1. Tổng quan về phần mềm và quy trình phát triển phần mềm 92](#_Toc34761552)

[8.1.1.1 Tổng quan về phần mềm 92](#_Toc34761553)

[8.1.1.2 Quy trình phát triển phần mềm 92](#_Toc34761554)

[8.1.2. Kiểm thử phần mềm 92](#_Toc34761555)

[8.1.3. Tại sao phải kiểm thử 93](#_Toc34761556)

[8.1.4. Ý nghĩa của kiểm thử 93](#_Toc34761557)

[8.1.5. Mục tiêu của kiểm thử phần mềm 93](#_Toc34761558)

[8.1.6. Nguyên tắc kiểm thử 94](#_Toc34761559)

[8.1.7. Quy trình kiểm thử 94](#_Toc34761560)

[8.2. Kiểm thử cho từng module : 95](#_Toc34761561)

[8.2.1. Kiểm thử UC Đăng ký học : 95](#_Toc34761562)

[tài liệu tham khảo 100](#_Toc34761563)

**DANH MỤC HÌNH ẢNH, SƠ ĐỒ**

[Hình 1.1 Sơ đồ Usecase 4](#_Toc32746214)

[Hình 2.1 Sơ đồ kiến trúc tổng thể 8](#_Toc32746215)

[Hình 3.1 Giao diện import dữ liệu 15](#_Toc32746216)

**DANH MỤC TỪ VIẾT TẮT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Từ viết tắt** | | **Ý nghĩa** | |
| 1. | ĐKHP | | Đăng ký học phần | |
| 2. | CSDL | | Cơ sở dữ liệu | |
| 3. | UC | | Use-case | |
| 4. | TC | Test case | |

# TỔNG QUAN về phần mềm quản lý khoa học

## Mô tả bài toán

### Thực trạng

Đăng ký học phần là một phần thiết yếu của mỗi sinh viên trước một kì học mới. Để hỗ trợ sinh viên có thể ĐKHP một cách đơn giản, hiệu quả, phần mềm đăng kí học đã được ra đời và trở thành cốt lõi trong quản lý đào tạo theo hệ thống tín chỉ tại các trường đại học. Để có một phần mềm giúp ĐKHP tốt, phục vụ được lượng lớn truy cập và cân bằng tải là một vấn đề khá khó khăn.

Với vai trò là người phát triển phần mềm, nhóm xây dựng một phần mềm đăng kí học tại trường đại học Thăng Long dành cho sinh viên dễ dàng tham gia ĐKHP cũng như dành cho Phòng đào tạo và bộ môn có thể quản lý và hỗ trợ việc ĐKHP của sinh viên trong trường một cách tối ưu. Thêm nữa, sau khi đăng kí học, sinh viên có thể xem thời khóa biểu và kết quả học tập của mình dựa theo việc ĐKHP của mình mỗi học kì cũng như giáo viên có thể theo dõi lịch dạy và theo dõi sinh viên trong lớp học của mình. Mỗi người tham gia phần mềm đều được cấp sẵn tài khoản (gồm user và password) với quyền được phân bởi admin của hệ thống.

Phần mềm được phát triển trên 2 nền tảng là web và mobile. Trên nền tảng web, người dùng có thể sử dụng toàn bộ chức năng. Trên mobile người dùng chỉ dùng được một số chức năng

### Phân tích bài toán

#### Chuẩn bị dữ liệu cho việc đăng kí học

* Thời khóa biểu toàn trường
* Chương trình học của các ngành
* Bảng điểm của sinh viên

#### Thực hiện đăng ký học

Chức năng chính cùa phần mềm là thực hiện đăng ký học:

* Phòng đào tạo import thời khóa biểu toàn trường để lưu vào cơ sở dữ liệu.
* Phòng đào tạo có thể xem lại dữ liệu đã import
* Sinh viên vào phần mềm đăng ký học chọn những môn học mình muốn và có thể đăng ký.
* Ngoài ra, phòng đạo tạo hoặc bộ môn có thể hỗ trợ ép cứng môn học cho sinh viên theo nhu cầu.
* Đăng kí học có thể xem học phí tương ứng với các lớp học phần.
* Sinh viên có thể xem kết quả của mình để đăng ký lớp học phần hợp lý do có nhiều học phần cần điều kiện tiên quyết.

#### Sử dụng kết quả đăng ký học

* Sau khi kết thúc thời gian đăng ký học, sinh viên sẽ có thời khóa biểu cá nhân để theo dõi học tập.
* Giáo viên có thể truy cập phần mềm để theo dõi lịch giảng dạy và danh sách sinh viên thuộc lớp của mình.
* Phòng đào tạo quản lý việc dạy và học của giáo viên, sinh viên trong kì dựa vào kết quả ĐKHP của sinh viên.

## Yêu cầu nghiệp vụ

* **BR1**: Quản lý dữ liệu
* **BR2**: Đăng nhập
* **BR3**: Quản lý tài khoản người dùng
* **BR4**: Đăng ký học:

+ **BR4.1**: Quản lý thời gian đăng ký học

+ **BR4.2**: Quản lý chuyên ngành

+ **BR4.3**: Quản lý học phần

+ **BR4.4**: Quản lý sinh viên

+ **BR4.5**: Quản lý giáo viên

+ **BR4.6:** Ép cứng lớp học phần

+ **BR4.7**: Đăng ký lớp học phần

+ **BR4.8:** Xem điểm

* **BR5:** Thời khóa biểu

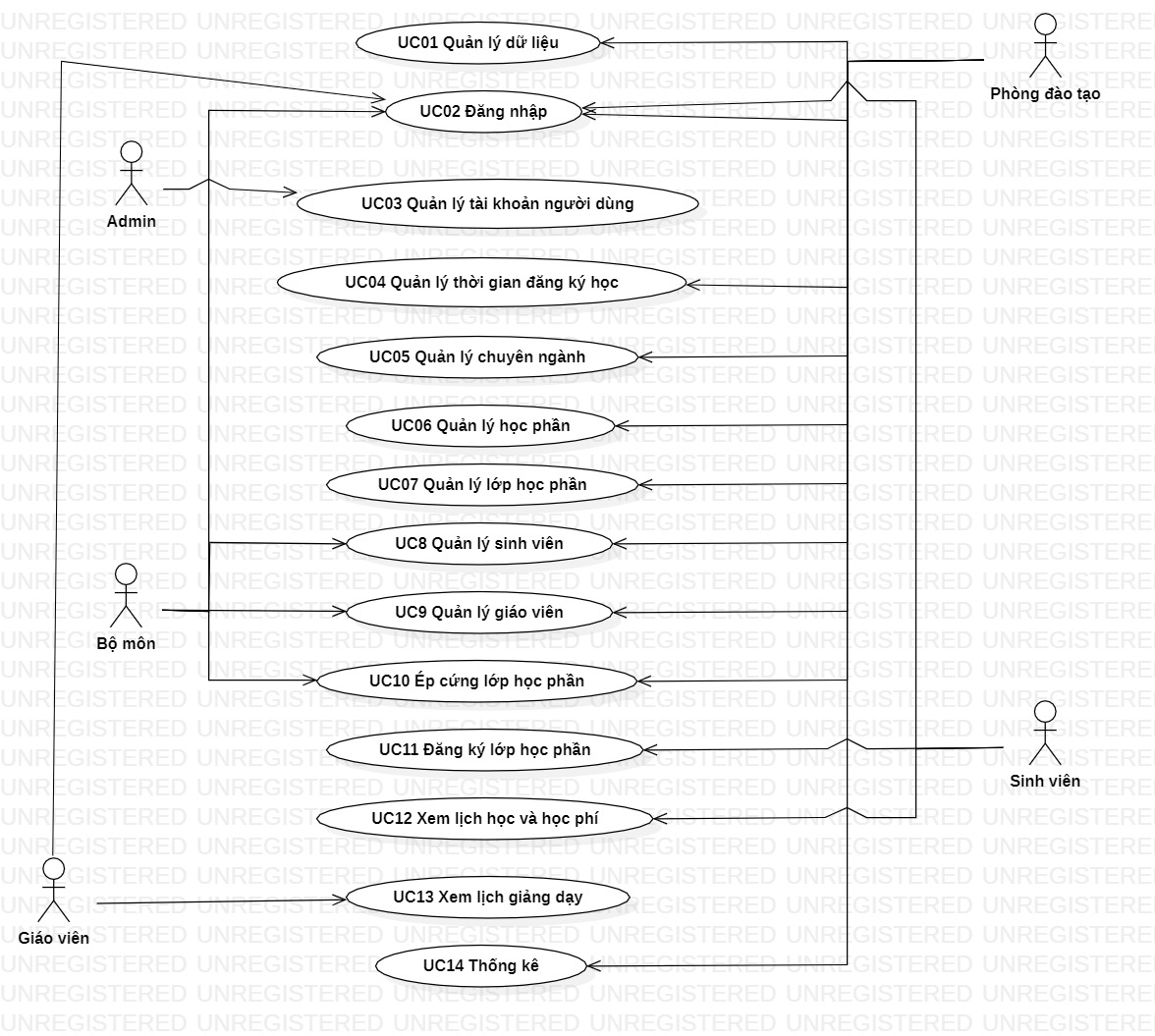
**+ BR5.1**: Xem lịch học và học phí

+ **BR5.2**: Xem lịch giảng dạy

* **BR6**: Thống kê

## Sơ đồ Usecase các chức năng chính

### Sơ đồ Usecase

****

Hình . Sơ đồ Usecase

### Các tác nhân hệ thống

* **Người quản trị**: Là tác nhân chịu trách chịu quản lý tài khoản người dùng
* **Phòng đào tạo**: Là tác nhân chịu trách nhiệm về việc đăng ký học phần của sinh viên, quản lý tất cả các dữ liệu của phần mềm, hỗ trợ ép cứng cũng như hủy lớp.
* **Sinh viên**: là tác nhân sử dụng phần mềm để ĐKHP cho mình và theo dõi kết quả học tập cũng như thời khóa biểu cá nhân
* **Giáo viên**: là tác nhân tham gia hệ thống để xem thông tin giảng dạy của mình
* **Bộ môn**: là tác nhân hỗ trợ ép cứng môn học cho sinh viên khoa mình

### Các Use-case

**UC #01 – *Quản lý dữ liệu***: Phòng đào tạo có thể import và xem lại các dữ liệu như thời khóa biểu toàn trường, bảng điểm sinh viên và chương trình học

**UC #02 – *Đăng nhập***: Hệ thống cho phép các tác nhân (Người quản trị, phòng đào tạo, sinh viên, giáo viên, trưởng bộ môn) đăng nhập vào hệ thống để thực hiện các chức năng của phần mềm trong quyền hạn.

**UC #03** – ***Quản lý tài khoản người dùng*:** Chức năng cho phép người quản trị có thể quản lý được tất cả các tài khoản tham gia hệ thống với các thao tác như thêm, sửa, xóa tài khoản và phân quyền tài khoản

**UC #04** – ***Quản lý thời gian đăng kí học***: Chức năng cho phép phòng đào tạo có thể thêm sửa xóa thời gian đăng kí học của sinh viên theo kì học.

**UC #05** – ***Quản lý chuyên ngành***: Chức năng cho phép phòng đạo tạo có thể thêm sửa xóa các chuyên ngành trong trường.

**UC #06** – ***Quản lý học phần***: Chức năng cho phép phòng đạo tạo có thể thêm sửa xóa thông tin các học phần.

**UC #07** – ***Quản lý sinh viên*** – Chức năng cho phép phòng đào tạo và bộ môn có thể xem cũng như thêm sửa xóa các thông tin của sinh viên.

**UC #08** – ***Quản lý giáo viên*** – Chức năng cho phép phòng đào tạo cũng như bộ môn theo dõi, them, sửa xóa thông tin của giáo viên.

**UC #09** – ***Ép cứng lớp học phần***: Chức năng này cho phép bộ môn có thể ép cứng lớp học phần thuộc bộ môn của mình cho sinh viên và phòng đào tạo có thể hỗ trợ ép sinh viên ở bất cứ khoa nào.

**UC #10** – ***Đăng ký lớp học phần***: Chức năng này cho phép sinh viên có thể đăng ký các lớp học phần mình mong muốn cũng như bỏ những lớp học phần đã đăng ký nhưng k muốn học nữa trong thời gian phòng đào tạo cho phép

**UC #11**– ***Xem điểm***: Chức năng cho phép sinh viên có thể kiểm tra kết quả học tập của mình

**UC #12** – ***Xem lịch học***: Chức năng cho phép sinh viên có thể xem lại thời khóa biểu của cá nhân sau khi đăng kí học.

**UC #13** – ***Xem lịch giảng dạy***: Chức năng cho phép giáo viên có thể xem lịch giảng dạy của mình.

**UC #14** – ***Thống kê***: Khi phòng đào tạo cần những thông tin dưới dạng các danh sách, có thể sử dụng chức năng này, phần mềm sẽ tạo báo cáo.

### Bảng ánh xạ yêu cầu nghiệp vụ với các use-case:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **BR** | | | **Mô tả** | | **UC** | |
| **BR1: Quản lý dữ liệu** | | | | | | |
| BR1 | Quản lý dữ liệu | | | UC #01 | | |
| **BR2: Đăng nhập** | | | | | | |
| BR2 | | | Đăng nhập | | UC #02 | |
| **BR3: Người dùng** | | | | | | |
| BR3 | | | Quản lý tài khoản người dùng | | UC #03 | |
| **BR4: Đăng ký học** | | | | | | |
| BR4.1 | | | Quản lý thời gian đăng ký học | | UC #04 | |
| BR4.2 | | | Quản lý chuyên ngành | | UC #05 | |
| BR4.3 | | | Quản lý học phần | | UC #06 | |
| BR4.5 | | | Quản lý sinh viên | | UC #07 | |
| BR4.6 | | | Quản lý giáo viên | | UC #08 | |
| BR4.7 | | | Ép cứng lớp học phần | | UC #09 | |
| BR4.8 | | | Đăng ký lớp học phần | | UC #10 | |
| BR4.9 | | | Xem điểm | | UC #11 | |
| **BR5 : Thời khóa biểu** | | | | | | |
| BR5.1 | | Xem lịch học và học phí | | | | UC#12 |
| BR5.2 | | Xem lịch giảng dạy | | | | UC#13 |
| **BR6 : Thống kê** | | | | | | |
| BR6 | | Thống kê | | | | UC#14 |

Bảng 1.1 Bảng ánh xạ yêu cầu nghiệp vụ với các use-case

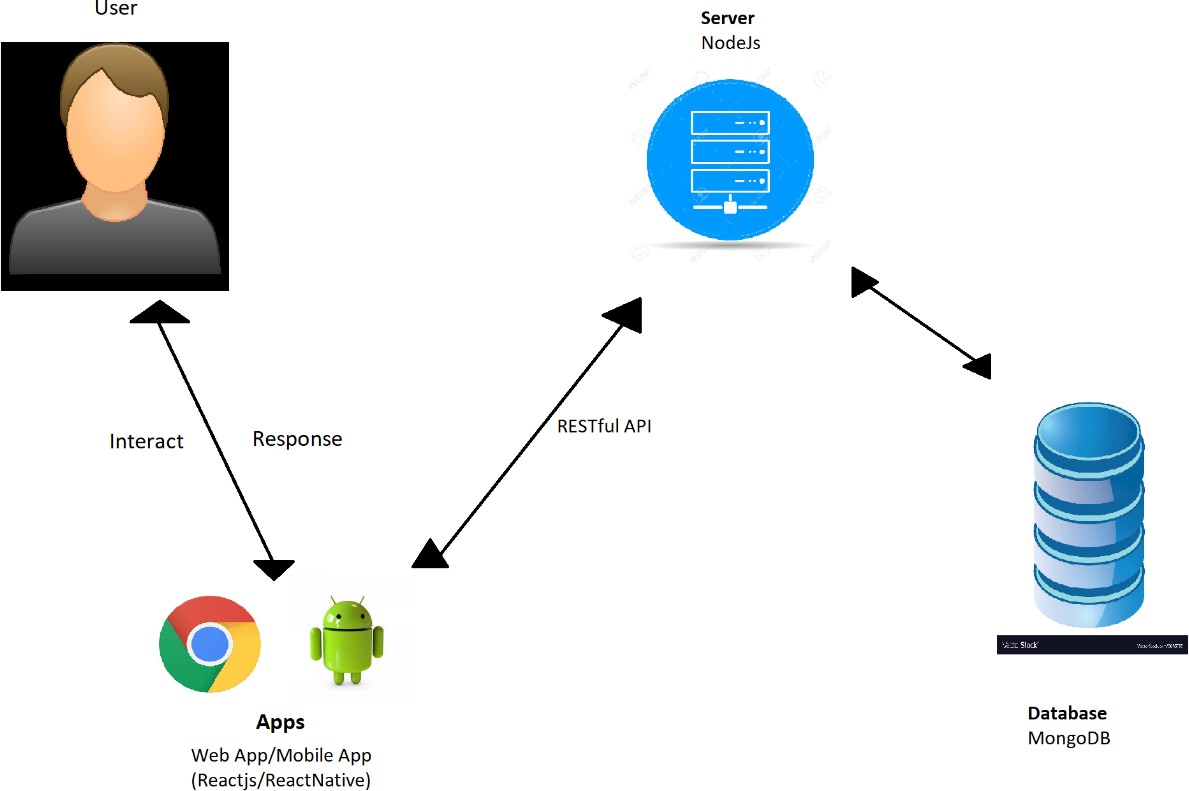
## Kế hoạch thực hiện

Thời gian thực hiện: ~3 tháng – 102 ngày (14/12/2019 – 25/03/2020)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Task Name** | **Duration** | **Start** | **Finish** |
| 1 | Phân tích yêu cầu nghiệp vụ | 12 days | 14/12/2019 | 26/12/2019 |
| 2 | Thiết kế CSDL | 10 days | 26/12/2019 | 05/01/2020 |
| 3 | Thiết kế giao diện cho người dùng | 20 days | 05/01/2020 | 25/01/2020 |
| 4 | Lập trình và hoàn thiện tài liệu | 46 days | 25/01/2020 | 11/03/2020 |
| 5 | Kiểm thử | 14 days | 11/03/2020 | 25/03/2020 |

# kiến trúc tổng thể của hệ thống

## Sơ đồ kiến trúc tổng thể



Hình . Sơ đồ kiến trúc tổng thể

Hệ thống được chia làm 2 phần: Client và server với các chức năng và đặc điểm khác nhau. Client (Web app và Mobile App) là nơi duy nhất người dùng có thể nhìn thấy và tương tác với ứng dụng, sau khi nhận yêu cầu từ người dùng, client sẽ tương tác với server. Server là nơi trực tiếp xử lý mọi yêu cầu của người dùng và trả lại những thông tin cần thiết.

## Database

Phần mềm đăng ký học dùng MongoDB để tạo cơ sở dữ liệu

* ***NoSQL***



NoSQL là thuật ngữ chung cho các hệ CSDL không sử dụng mô hình dữ liệu quan hệ. NoSQL đặc biệt nhấn mạnh đến mô hình lưu trữ cặp giá trị - khóa và hệ thống lưu trữ phân tán.

NoSQL được giới thiệu lần đầu năm 1988 gọi chung cho các hệ CSDL quan hệ nguồn mở nhỏ không sử dụng SQL để truy vấn.

CSDl NoSQL ra đời nhằm giảm thiểu tối đa các phép tính toán, tác vụ đọc-ghi liên quan kết hợp với xử lý theo lô (batch processing) đảm bảo được yêu cầu xử lý dữ liệu của các dịch vụ mạng xã hội. Hệ CSDL này có thể lưu trữ, xử lý từ lượng nhỏ đến hang petabytes dữ liệu với khả năng chịu tải, chịu lỗi cao nhưng chỉ đòi hỏi về tài nguyên phần cứng.

Một số đặc điểm của CSDL NoSQL:

* Phi quan hệ (hay không ràng buộc): không sử dụng các ràng buộc dữ liệu cho nhất quán dữ liệu
* Lưu trữ phân tán: mô hình lưu trữ phân tán các tập tin hoặc dữ liệu ra nhiều máy khác nhau dưới sự kiểm soát của phần mềm
* Nhất quán cuối: tính nhất quán của dữ liệu không cần phải đảm bảo ngay tức khắc sau mỗi phép ghi.
* Triển khai đơn giản, dễ dàng mở rộng

NoSQL sử dụng cho các ứng dụng cực lớn haowjc nhỏ. Thiết kế NoSQL chấp nhận tính nhất quán yếu và có thể không dung đến “transaction”. NoSQL thích hợp cho các mô hình lưu trữ dữ liệu có tính đặc thù như object

* ***MongoDB***



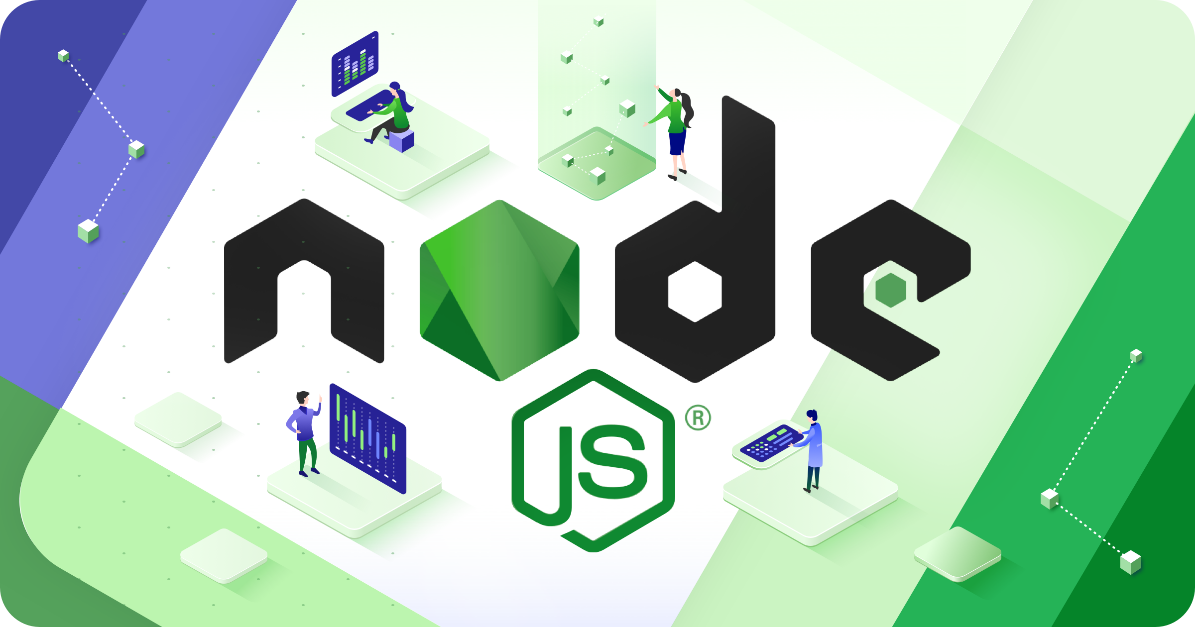
MongoDB là một chương trình CSDL mã nguồn mở được thiết kế theo kiểu hướng đối tượng trong đó các bảng được cấu trúc một cách linh hoạt cho phép các dữ liệu lưu trên bảng không cần phải tuân theo một dạng cấu trúc nhất định nào. Do vậy, MongoDB có thể được dùng để lưu trữ các dữ liệu có cấu trúc phức tạp, đa dạng và không cố định.

**Ưu điểm khi sử dụng MongoDB:**

* Ít Schema hơn: MongoDB là một CSDL dựa trên Document, trong đó một Collection giữ các Document khác nhau. Số trường, nội dung và kích cỡ của Document này có thể khác với Document khác
* Cấu trúc của một đối tượng là rõ ràng
* Không có các join phức tạp
* Khả năng truy vấn sâu hơn. MongoDB hỗ trợ các truy vấn động trên các Document bởi sử dụng một ngôn ngữ truy vấn dựa trên Document mà mạnh mẽ như SQL.
* MongoDB dễ dàng để mở rộng
* Việc chuyển đổi/ánh xạ của các đối tượng ứng dụng đến các đối tượng CSDL là không cần thiết
* Sử dụng bộ nhớ nội tại để lưu trữ phần công việc, giúp truy cập dữ liệu nhanh hơn.

## Server

## Phần mềm đăng ký học dùng Nodejs để tạo server



Nodejs là một nền tảng phát triển độc lập được xây dựng ở trên Javascript Runtime của Chrome mà chúng ta có thể xây dựng được các ứng dụng mạng một cách nhanh chóng và dễ dàng mở rộng.

Phần core bên dưới của NodeJS được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao.

NodeJS tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực.

NodeJS áp dụng cho các sản phẩm có lượng truy cập lớn, cần mở rộng nhanh, đổi mới công nghệ.

Lý do để sử dụng Nodejs:

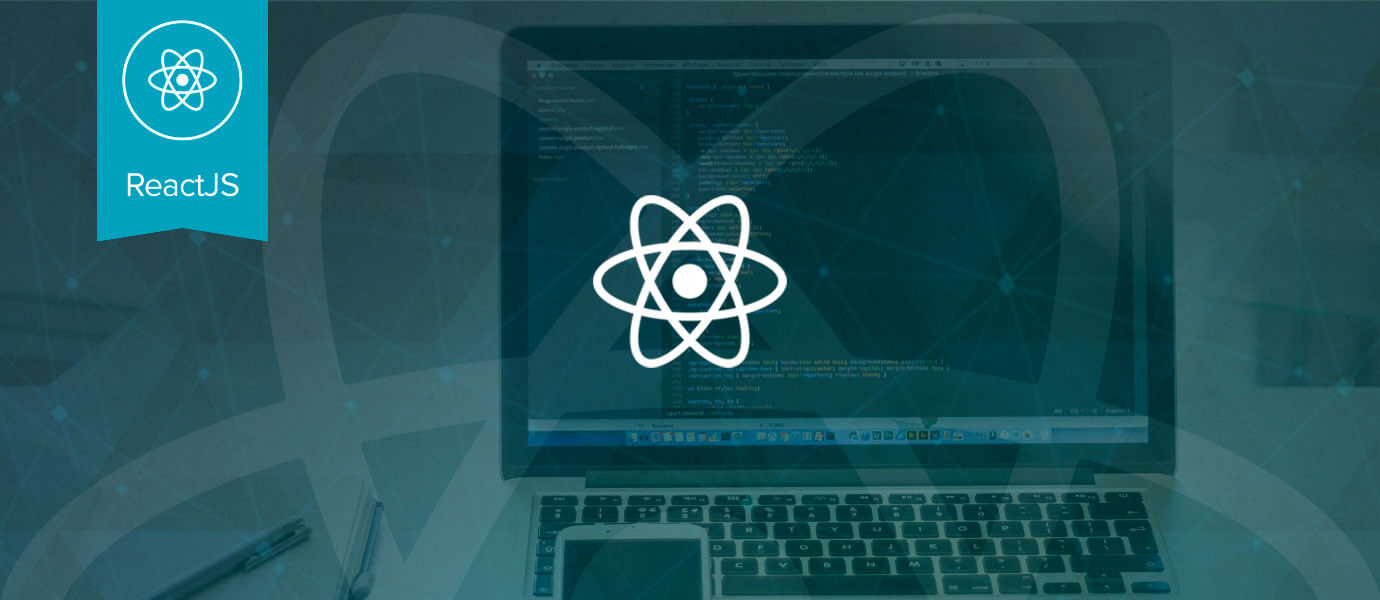
* Các ứng dụng Nodejs được viết bằng javascript, là một ngôn ngữ khá thông dụng.
* NodeJS chạy đa nền tảng phía server, sử dụng kiến trúc hướng sự kiện Even-driven, cơ chế non-blocking I/O làm cho nó nhẹ và hiệu quả.
* Có thể chạy ứng dụng Nodejs ở bất kì đâu trên máy Window-Mac-Linux, cộng đồng Nodejs rất lớn và miễn phí.
* Các ứng dụng Nodejs đáp ứng tốt thời gian thực và chạy đa nền tảng, đa thiết bị.

Nodejs đang được sử dụng bởi một số công ty lớn như Amazon, Ebay, Microsoft…v…v…

## Web Client

Phần mềm đăng ký học sử dụng reactjs cho web app và react native cho mobile app.

* ***Reactjs***



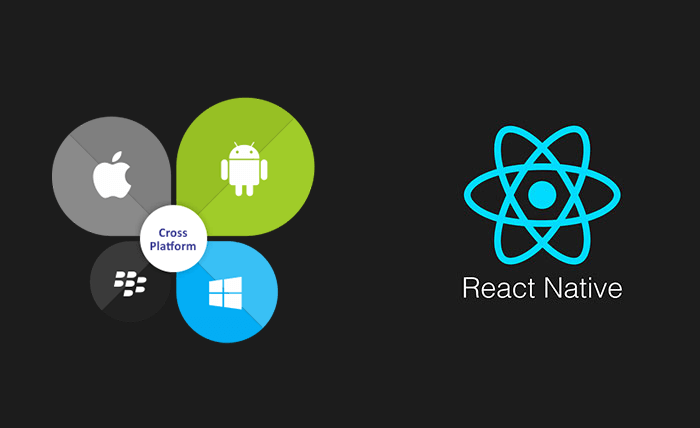
Reactjs là một thư viện trong đó có chứa nhiều Javascript mã nguồn mở và cha đẻ của Reactjs là Facebook. Mục đích của việc tạo ra Reactjs là để tạo ra những ứng dụng website hấp dẫn với tốc độ nhanh và hiệu quả cao với những mã hóa tối thiểu.

Nhìn chung tất cả các tính năng hay sức mạnh của Reactjs thường xuất phát tự việc tập trung vào các phần riêng lẻ. Chính vì điểm này, khi làm việc trên web thay vì nó sẽ làm việc trên toàn bộ ứng dụng thì reactjs cho phép Developer có thể phá vỡ giao diện của người dùng từ một cách phức tạp và biến nó thành các phần đơn giản hơn nhiều lần

Reactjs chỉ là view Library hỗ trợ xây dựng những thành phần UI khi có tính tương tac cao, có trạng thái và sử dụng lại bất cứ lúc nào. Là nơi xây dựng xung quanh các component, khi hoạt động, react hoạt động trên cả Client và render trên cả server để 2 phần này có thể kết nối với nhau.

Lợi ích khi sử dụng Reactjs:

* Tạo cho nó một dom ảo, là nơi các component được tồn tại. Giúp cải thiện hiệu suất làm việc.
* Việc viêt các đoạn code JS trở nên dễ dàng hơn vì sử dụng cú pháp đặc biệt là JSX cho phép trộn giữa code HTML và Javascipt. Ngoài ra còn có thể đem đoạn code thêm vào trong hàm render mà không cần phải thực hiện việc nối chuỗi.
* Có nhiều công cụ phát triển
* ***ReactNative***



ReactNative là một framework do công ty công nghệ nổi tiếng Facebook phát triển nhằm mục đích giải quyết bài toán hiệu năng của Hybrid (sự kết hợp giữa ứng dụng Web và ứng dụng mobile) và bài toán chi phí khi mà phải viết nhiều loại ngôn ngữ native cho từng nền tảng di động.

Chúng ta sẽ xây dựng được ứng dụng Native một cách đa nền tảng (multi-platform) chứ không phải là một “mobile web app”, không phải là “HTML5 app”, và cũng không phải là một “hybrid app” hay cũng không chỉ xây dựng trên iOS hay Android mà chúng ta xây dựng và chạy được cả hai hệ sinh thái.

Một điểm hay ho nữa là giảm chi phí recompile của Native bằng cách sử dụng Hot-Loading tức là bạn không cần phải xây dựng lại ứng dụng từ đầu nên việc chỉnh sửa diễn ra rất nhanh chóng. Giúp cho lập trình viên có thể thấy được những chỉnh sửa của họ một cách nhanh chóng trực quan, không còn phải bỏ quá nhiều thời gian trong việc xây dựng và chạy ứng dụng nữa.

Chúng ta chỉ cần sử dụng JS để phát triển được một ứng dụng di động hoàn chỉnh, đồng thời giải quyết được các vấn đề mà Native App(các ứng dụng được phát triển và build trên chính những công cụ hỗ trợ mà nhà phát triển Ios và Android cung cấp cho người lập trình ở đây là XCode và Android) gặp phải đã nêu ở trên. Và rồi còn cả kết hợp với code native như Swift, Java, v.v.

Ưu điểm khi sử dụng React Native:

* Hiệu quả về mặt thời gian khi muốn phát triển một ứng dụng nhanh chóng
* Hiệu năng tương đối ổn định
* Cộng đồng phát triển mạnh
* Tiết kiệm chi phí.
* Ứng dụng tin cậy và ổn định
* Xây dựng cho nhiều hệ điều hành khác nhau với ít native code nhất.

# ĐẶC TẢ CÁC CHỨC NĂNG

## UC#01 – Quản lý dữ liệu

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Quản lý dữ liệu** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép phòng đào tạo quản lý các dữ liệu như bảng điểm sinh viên, chương trình học và thời khóa biểu toàn trường | |
| **Tác nhân** | | Phòng đào tạo | |
| **Tiền điều kiện** | | Phòng đào tạo phải đăng nhập vào hệ thống bằng tài khoản của mình | |
| **Hậu điều kiện** | **Thành công** | Bảng điểm sinh viên, chương trình học, thời khóa biểu toàn trường được phòng đào tạo import vào cơ sở dữ liệu và có thể xem lại. | |
| **Lỗi** | Không thêm hoặc xem được dữ liệu | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi phòng đào tạo muốn import dữ liệu hoặc xem lại dữ liệu   * Usecase này bắt đầu khi phòng đào tạo muốn import dữ liệu vào cơ sở dữ liệu * Phòng đạo tạo chọn chức năng import dữ liệu trong thanh sidebar. * Hệ thống hiển thị form chọn kiểu dữ liệu * Phòng đào tạo chọn loại dữ liệu muốn import lên (bảng điểm, thời khóa biểu toàn trường hoặc chương trình học). * Hệ thống hiển thị form “tải dữ liệu lên” * Phòng đào tạo chọn “tải lên” và chọn file cần tải lên từ nguồn dữ liệu của mình và chọn tiếp theo. Nếu chọn sai dữ liệu thì thực hiện luồng A. * Hệ thống thống kê lại dữ liệu chuẩn bị lưu và hiển thị form xác nhận dữ liệu để phòng đào tạo xác nhận lại dữ liệu. * Phòng đào tạo xác nhận và dữ liệu sẽ được hệ thống lưu vào CSDL. * Luồng con: xem dữ liệu * Sau khi đã import dữ liệu, hệ thống hiển thị form chọn dữ liệu cần xem trong thanh sidebar * Phòng đào tạo chọn dữ liệu cần xem * Hệ thống hiển thị dữ liệu | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Luồng A: Dữ liệu không hợp lý.** | | | |
| * Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng chọn lại dữ liệu * Phòng đào tạo chọn lại dữ liệu và chọn “tiếp theo”, nếu không muốn chọn dữ liệu nữa thì chọn “trở về” | | | |
| **Giao diện minh họa** | | | |
| Hình . Giao diện import dữ liệu    Hình 3.2 Xem dữ liệu | | | |

## UC#02 – Đăng nhập

## 

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Đăng nhập** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép người dùng đăng nhập vào hệ thống để sử dụng các chức năng của hệ thống. | |
| **Tác nhân** | | Người quản trị, phòng đào tạo, giáo viên, sinh viên, bộ môn | |
| **Tiền điều kiện** | | Người quản trị, phòng đào tạo, giáo viên, sinh viên, bộ môn phải đăng nhập vào hệ thống | |
| **Hậu điều kiện** | **Thành công** | Đăng nhập thành công vào hệ thống | |
| **Lỗi** | Đăng nhập không thành công | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi người dùng muốn đăng nhập để sử dụng các chức năng của hệ thống   * Usecase này bắt đầu khi tác nhân muốn đăng nhập vào hệ thống. * Hệ thống yêu cầu tác nhân nhập tài khoản và mật khẩu đăng nhập. * Tác nhân nhập tài khoản và mật khẩu đăng nhập của mình. * Hệ thống xác nhận tài khoản và mật khẩu đăng nhập có hợp lệ không? Nếu không hợp lệ thì thực hiện luồng A. * Hệ thống ghi lại quá trình đăng nhập. * Luồng con: đổi mật khẩu * Tác nhân mở form đổi mật khẩu. * Hệ thống hiển thị form đổi mật khẩu và yêu cầu nhập thông tin thay đổi mật khẩu. * Tác nhân nhập thông tin thay đổi mật khẩu bao gồm: mật khẩu cũ, mật khẩu mới, xác nhận mật khẩu mới. Sau đó, nhấn nút “đồng ý”.   Kiểm tra thông tin. Nếu hợp lệ, hệ thống thông báo thay đổi mật khẩu thành công. Nếu không hợp lệ thì thực hiện luồng A1.  **Đăng xuất khỏi hệ thống:**  Chức năng này bắt đầu khi người dùng muốn rời khỏi hệ thống.   * Người dùng chọn đăng xuất trên trang website. * Hệ thống đăng xuất người dùng ra khỏi hệ thống.   Sau khi người dùng chọn đăng xuất. Hệ thống sẽ hiển thị ra trang đăng nhập. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Luồng A: Nhập sai tài khoản/ mật khẩu đăng nhập** | | | |
| * Hệ thống hiển thị một thông báo lỗi * Người sử dụng có thể chọn hoặc là đăng nhập lại hoặc là hủy bỏ đăng nhập, khi đó usecase này sẽ kết thúc. | | | |
| **Luồng A1: Nhập thông tin không hợp lệ** | | | |
| * Hệ thống hiển thị thông báo “tài khoản không tồn tại” và yêu cầu nhập lại thông tin. * Người sử dụng có thể chọn nhập lại thông tin hoặc hủy bỏ thay đổi mật khẩu, khi đó usecase này kết thúc. | | | |
| **Giao diện minh họa** | | | |
| Hình .3 Giao diện đăng nhập | | | |

## UC#03 – Quản lý tài khoản người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Quản lý tài khoản người dùng** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép người quản trị quản lý tất cả tài khoản của các tác nhân tham gia vào hệ thống. Người quản trị có thể xem danh sách các tài khoản cũng như thông tin chi tiết của tài khoản, ngoài ra còn có thể thêm, sửa, xóa các tài khoản nếu cần thiết. | |
| **Tác nhân** | | Người quản trị | |
| **Tiền điều kiện** | | Người quản trị đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Thông tin tài khoản và danh sách tài khoản được cập nhật vào hệ thống | |
| **Lỗi** | Thông tin các tài khoản không thay đổi | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi người quản trị muốn quản lý tài khoản người dùng   * Usecase này bắt đầu khi người quản trị muốn quản lý tài khoản người dùng. * Người quản trị chọn chức năng quản lý tài khoản trong thanh sidebar. * Hệ thống hiển thị danh sách tài khoản người dùng theo form: * STT * Họ tên * Tên đăng nhập * Quyền * Ngày sinh * Người quản trị có thể lựa chọn thanh công cụ “Thêm” để thêm mới tài khoản, luồng con “Thêm tài khoản” được thực hiện. * Người quản trị có thể lựa chọn thanh công cụ “Sửa” để sửa thông tin tài khoản, luồng con “Sửa thông tin tài khoản” được thực hiện. * Người quản trị có thể lựa chọn thanh công cự ‘xóa” để xóa đi tài khoản, luồng con “xóa tài khoản” được thực hiện. * Luồng con: Thêm tài khoản * Hệ thống hiển thị form với các thông tin: * Tên đăng nhập * Mật khẩu * Quyền * Mã định danh * Họ và tên * Ngày sinh * Người quản trị nhập thông tin theo form (bao gồm các thông tin tài khoản kèm theo thông tin ứng với từng quyền) * Hệ thống thêm mới tài khoản và cập nhật lại danh sách tài khoản. * Luồng con: Sửa thông tin tài khoản * Hệ thống hiển thị form với các thông tin * Tên đăng nhập * Mật khẩu * Quyền * Mã định danh * Họ và tên * Ngày sinh * Người quản trị sửa lại các thông tin theo form cho sẵn (bao gồm các thông tin tài khoản kèm theo thông tin ứng với từng quyền) * Hệ thống cập nhật lại thông tin tài khoản và danh sách tài khoản sau khi người dùng xác nhận sửa. * Luồng con – Xóa tài khoản: * Hệ thống hiển thị thông báo cho người quản trị:” Xác nhận xóa tài khoản người dùng”. * Người quản trị chọn “xác nhận” để xóa tài khoản người dùng. * Người quản trị chọn “hủy” để hủy bỏ yêu cầu. * Hệ thống cập nhật lại danh sách tài khỏan người dùng. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .4 Giao diện xem danh sách tài khoản    Hình 3.5 Giao diện thêm mới tài khoản    Hình 3.6 Giao diện xóa tài khoản | | | |

## UC#04 – Quản lý thời gian đăng ký học

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Quản lý thời gian đăng ký học** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép phòng đào tạo có thể quản lý các đợt đăng ký học như xem thông tin các đợt đăng ký học, ngoài ra còn có thể thêm, sửa xóa các đợt đăng ký học nếu cần thiết. | |
| **Tác nhân** | | Phòng đào tạo | |
| **Tiền điều kiện** | | Phòng đào tạo đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Thông tin các đợt đăng ký học được lưu vào cơ sở dữ liệu | |
| **Lỗi** | Thông tin các đợt đăng ký học không thay đổi | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi phòng đào tạo muốn xem các đợt đăng ký học   * Usecase này bắt đầu khi phòng đào tạo muốn xem các đợt đăng ký học * Người quản trị chọn chức năng quản lý đợt đăng ký. * Hệ thống hiển thị danh sách các đợt đăng ký học với các thông tin: * STT * Kì * Nhóm * Năm học * Số đợt nhỏ * Trạng thái * Phòng đào tạo có thể lựa chọn thanh công cụ “Tạo đợt đăng ký” để thêm mới đợt đăng ký học, luồng con “Thêm mới đợt đăng ký học” sẽ được thực hiện. * Phòng đào tạo có thể lựa chọn thanh công cụ “Sửa” để sửa thông tin đợt đăng ký học, luồng con “Sửa thông tin đợt đăng ký học” được thực hiện. * Phòng đào tạo có thể lựa chọn thanh công cụ ‘xóa” để xóa đợt đăng ký học, luồng con “xóa đợt đăng ký học” được thực hiện. * Luồng con: Thêm mới đợt đăng ký * Hệ thống hiển thị form với các thông tin: * Năm học * Học kì * Nhóm * Số đợt nhỏ gồm các thông tin con như thời gian bắt đầu, thời gian kết thúc, thời gian delay. * Phòng đào tạo điền thông tin theo form yêu cầu (bao gồm các thông tin kèm theo theo đợt đăng ký nhỏ) * Hệ thống thêm mới đợt đăng ký học và cập nhật lại danh sách đợt đăng ký học * Luồng con: Sửa thông tin đợt đăng ký học * Người dùng click chuột vào đợt đăng ký học muốn sửa * Hệ thống hiển thị thông tin chi tiết đợt đăng ký học theo form gồm: * Năm học * Học kì * Nhóm * Số đợt nhỏ gồm các thông tin con như thời gian bắt đầu, thời gian kết thúc, thời gian delay. * Phòng đào tạo sửa lại các thông tin theo form cho sẵn (bao gồm các thông tin kèm theo đợt đăng ký nhỏ) * Hệ thống cập nhật lại thông tin đợt đăng ký và danh sách đợt đăng ký sau khi phòng đào tạo xác nhận sửa. * Luồng con – Xóa đợt đăng ký: * Phòng đào tạo click chuột vào đợt đăng ký học muốn xóa * Hệ thống hiển thị thông tin chi tiết đợt đăng ký học * Phòng đào tạo chọn “xóa bỏ” để yêu cầu xóa đợt đăng ký học * Hệ thống xác nhận yêu cầu của phòng đào tạo rồi xóa đợt đăng ký học và cập nhật lại danh sách đợt đăng ký | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| C:\Users\b360\Downloads\82162816_1539622302868693_2426878825452273664_n.png  Hình .7 Giao diện quản lý đợt đăng ký | | | |

## UC#05 - Quản lý chuyên ngành

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Quản lý chuyên ngành** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép phòng đào tào có thể quản lý các chuyên ngành có trong trường, ngoài ra phòng đào tạo còn có thể thêm, sửa xóa các chuyên ngành nếu cần thiết. | |
| **Tác nhân** | | Phòng đào tạo | |
| **Tiền điều kiện** | | Phòng đào tạo đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Thông tin các chuyên ngành được thay đổi và lưu vào CSDL | |
| **Lỗi** | Thông tin các chuyên ngành được giữ nguyên. | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi phòng đào tạo xem thông tin các chuyên ngành ;   * Usecase này bắt đầu khi phòng đào tạo muốn xem thông tin các chuyên ngành * Phòng đào tạo chọn chức năng quản lý chuyên ngành ở thanh sidebar. * Hệ thống hiển thị danh sách các chuyên ngành hiện có của trường. * Phòng đào tạo có thể click chuột vào chuyên ngành để xem thông tin chi tiết của chuyên ngành * Phòng đào tạo có thể lựa chọn thanh công cụ “Tạo mới chuyên ngành” để thêm mới chuyên ngành, luồng con “Thêm mới chuyên ngành” sẽ được thực hiện. * Phòng đào tạo có thể lựa chọn thanh công cụ “Sửa” để sửa thông tin chuyên ngành, luồng con “Sửa thông tin chuyên ngành” được thực hiện. * Phòng đào tạo có thể lựa chọn thanh công cụ ‘xóa” để xóa chuyên ngành, luồng con “xóa chuyên ngành” được thực hiện. * Luồng con: Thêm mới chuyên ngành * Hệ thống hiển thị form với các thông tin: * Tên chuyên ngành * Chương trình học * Phòng đào tạo điền thông tin theo form yêu cầu, trong task chương trình học, phòng đào tạo chọn công cụ thêm mới học phần-luồng con “thêm học phần” sẽ được thực hiện. * Hệ thống thêm mới chuyên ngành và cập nhật lại danh sách chuyên ngành * Luồng con: Sửa thông tin chuyên ngành * Người dùng click chuột vào chuyên ngành muốn sửa * Hệ thống hiển thị thông tin chi tiết học phần theo form gồm: * Tên chuyên ngành * Chương trình học gồm danh sách các học phần * Phòng đào tạo sửa lại tên chuyên ngành, thêm học phần hoặc xóa học phần-luồng con “xóa học phần” sẽ được thực hiện tùy theo yêu cầu * Hệ thống cập nhật lại thông tin đợt đăng ký và danh sách đợt đăng ký sau khi phòng đào tạo xác nhận sửa. * Luồng con – Xóa chuyên ngành: * Phòng đào tạo click chuột vào chuyên ngành học muốn xóa * Hệ thống hiển thị thông tin chi tiết chuyên ngành học * Phòng đào tạo chọn “xóa bỏ” để yêu cầu xóa chuyên ngành học. * Hệ thống xác nhận yêu cầu của phòng đào tạo rồi xóa chuyên ngành học và cập nhật lại danh sách chuyên ngành. * Luồng con: thêm mới học phần: * Hệ thống hiển thị danh sách các học phần có sẵn * Phòng đào tạo click chọn học phần muốn thêm để thêm vào chương trình học * Hệ thống cập nhật chương trình học. * Luồng con: xóa học phần: * Hệ thống hiển thị danh sách các học phần * Phòng đào tạo chọn “xóa” để yêu cầu xóa các lớp học phần * Hệ thống xác nhận lại yêu cầu của phòng đào tạo và cập nhật lại chương trình học. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .8 Giao diện xem danh sách chuyên ngành  Hình 3.9 giao diện thêm mới chuyên ngành  Hình 3.10 Giao diện xóa chuyên ngành | | | |

## UC#06 – Quản lý học phần

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Quản lý học phần** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép phòng đào tào có thể quản lý các học phần có trong trường, ngoài ra phòng đào tạo còn có thể thêm, sửa xóa các học phần nếu cần thiết | |
| **Tác nhân** | | Phòng đào tạo | |
| **Tiền điều kiện** | | Phòng đào tạo đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Thông tin các học phần được thay đổi và lưu vào CSDL | |
| **Lỗi** | Thông tin các học phần được giữ nguyên. | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi phòng đào tạo xem thông tin các học phần ;   * Usecase này bắt đầu khi phòng đào tạo muốn xem thông tin các học phần * Phòng đào tạo chọn chức năng quản lý học phần ở thanh sidebar. * Hệ thống hiển thị danh sách các học phần hiện có của trường với các thông tin: * STT * Mã Môn * Tên Môn * Học phần tiên quyết * TC tiên quyết * Số TC * Phòng đào tạo có thể lựa chọn thanh công cụ “Tạo mới học phần” để thêm mới học phần, luồng con “Thêm mới học phần” sẽ được thực hiện. * Phòng đào tạo có thể lựa chọn thanh công cụ “Sửa” để sửa thông tin học phần, luồng con “Sửa thông tin học phần” được thực hiện. * Phòng đào tạo có thể lựa chọn thanh công cụ ‘xóa” để xóa học phần, luồng con “xóa học phần” được thực hiện. * Luồng con: Thêm mới chuyên ngành * Hệ thống hiển thị form với các thông tin: * STT * Mã Môn * Tên Môn * Học phần tiên quyết * TC tiên quyết * Số TC * Danh sách các lớp học phần * Phòng đào tạo điền thông tin theo form yêu cầu, chọn “xác nhận’ * Hệ thống thêm mới học phần và cập nhật lại danh sách học phần * Luồng con: Sửa thông tin học phần * Người dùng click chuột vào học phần muốn sửa * Hệ thống hiển thị thông tin chi tiết học phần theo form gồm: * STT * Mã môn * Tên Môn * Học phần tiên quyết * TC tiên quyết * Số TC * Danh sách các lớp học phần. * Phòng đào tạo sửa lại các thông tin trong form mẫu, chọn “xác nhận” * Hệ thống cập nhật lại danh sách các học phần * Luồng con – Xóa học phần: * Phòng đào tạo click chuột vào học phần muốn xóa * Hệ thống hiển thị thông tin chi tiết học phần * Phòng đào tạo chọn “xóa bỏ” để yêu cầu xóa học phần. * Hệ thống xác nhận yêu cầu của phòng đào tạo rồi xóa học phần và cập nhật lại danh sách học phần * Luồng con: thêm mới lớp học phần: * Hệ thống hiển thị form gồm các thông tin: * Tên lớp * Số lượng sinh viên tối thiểu * Số lượng sinh viên tối đa * Phòng đào tạo điền thông tin vào form rồi chọn xác nhận * Hệ thống cập nhật danh sách lớp học phần * Luồng con: sửa lớp học phần * Hệ thống hiển thị form gồm * Tên lớp * Số lượng sinh viên tối thiểu * Số lượng sinh viên tối đa * Phòng đào tạo thay đổi thông tin và chọn “xác nhận”. * Hệ thống cập nhật lại danh sách lớp học phần. * Luồng con: xóa lớp học phần * Phòng đào tạo click chuột vào lớp học phần muốn xóa * Hệ thống hiển thị thông tin lớp * Phòng đào tạo chọn "xóa bỏ” * Hệ thống xác nhạn lại yêu cầu của phòng đào tạo và cập nhật lại danh sách lớp học phần | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .11 Giao diện xem danh sách học phần    Hình 3.12 Giao diện thêm mới học phần    Hình 3.13 Giao diện xóa học phần | | | |

## UC#07 – Quản lý sinh viên

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Quản lý sinh viên** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Chức năng cho phép phòng đào tạo, bộ môn có thê quản lý thông tin của sinh viên như xem, thêm mới, sửa xóa, tìm kiếm | |
| **Tác nhân** | | Phòng đào tạo, bộ môn. | |
| **Tiền điều kiện** | | Phòng đào tạo, bộ môn đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Thông tin sinh viên được thay đổi và lưu vào hệ thống | |
| **Lỗi** | Thông tin sinh viên không thay đổi | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi phòng đào tạo muốn quản lý sinh viên trong trường.   * Usecase này bắt đầu khi phòng đào tạo hoặc bộ môn muốn quản lý thông tin sinh viên trong trường. * Phòng đào tạo, bộ môn chọn chức năng quản lý sinh viên trong thanh sidebar. * Hệ thống hiển thị danh sách sinh viên theo form: * STT * Mã sinh viên * Họ tên * Lớp chuyên ngành * Phòng đạo tạo có thể xem chi tiết thông tin của sinh viên bằng cách click chuột vào sinh viên muốn xem * Hệ thống sẽ hiển thị thông tin chi tiết bao gồm: * Họ tên * Mã sinh viên * Ngày sinh * Lớp * Điện thoại * Email sinh viên * Địa chỉ * Bảng điểm sinh viên * Thời khóa biểu * Phòng đào tạo, bộ môn có thể lựa chọn thanh công cụ “Thêm” để thêm mới sinh viên, luồng con “Thêm sinh viên” được thực hiện. * Phòng đào tạo, bộ môn có thể lựa chọn thanh công cụ “Sửa” để sửa thông tin tài khoản, luồng con “Sửa thông tin sinh viên” được thực hiện. * Phòng đào tạo, bộ môn có thể lựa chọn thanh công cự ‘xóa” để xóa sinh viên, luồng con “xóa sinh viên” được thực hiện. * Phòng đào tạo, bộ môn có thể tìm kiếm thông tin sinh viên bằng cách chọn chức năng “tìm kiếm”-luồng con “tìm kiếm sinh viên” được thực hiên. * Luồng con: Thêm sinh viên * Hệ thống hiển thị form với các thông tin: * Họ tên * Mã sinh viên * Ngày sinh * Lớp * Điên thoại * Email sinh viên * Địa chỉ * Bảng điểm * Thời khóa biểu * Phòng đào tạo hoặc bộ môn nhập thông tin theo form. * Hệ thống thêm mới sinh viên và cập nhật lại danh sách sinh viên. * Luồng con: Sửa thông tin sinh viên * Hệ thống hiển thị form với các thông tin * Họ tên * Mã sinh viên * Ngày sinh * Lớp * Điện thoại * Email sinh viên * Địa chỉ * Bảng điểm * Thời khóa biểu * Phòng đào tạo, bộ môn sửa lại các thông tin theo form cho sẵn. * Hệ thống cập nhật lại thông tin giáo viên và danh sách giáo viên sau khi người dùng xác nhận sửa. * Luồng con – Xóa sinh viên: * Hệ thống hiển thị thông tin chi tiết sinh viên * Phòng đào tạo hoặc bộ môn chọn “xóa bỏ” để yêu cầu xóa sinh viên. * Hệ thống yêu cầu xác nhận lại yêu cầu * Người quản trị chọn “xác nhận” để xóa sinh viên. * Người quản trị chọn “hủy” để hủy bỏ yêu cầu. * Hệ thống cập nhật lại danh sách sinh viên. * Luồng con- Tìm kiếm sinh viên: * Phòng đào tạo, bộ môn nhập từ khóa vào thanh tìm kiếm * Hệ thống sẽ tìm kiếm trong cơ sở dữ liệu và đưa ra những sinh viên phù hợp từ khóa * Phòng đào tạo, bộ môn click vào để xem thông tin chi tiết. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .14 Giao diện xem danh sách sinh viên    Hình 3.15 Giao diện thêm mới sinh viên    Hình 3.16 Giao diện xóa sinh viên    Hình 3.17 Giao diên tìm kiếm sinh viên | | | |

## UC#08 – Quản lý giáo viên

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Quản lý giáo viên** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Chức năng cho phép phòng đào tạo, bộ môn có thê quản lý thông tin của giáo viên như xem, thêm mới, sửa xóa, tìm kiếm | |
| **Tác nhân** | | Phòng đào tạo, bộ môn. | |
| **Tiền điều kiện** | | Phòng đào tạo, bộ môn đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Thông tin giáo viên được thay đổi và lưu vào hệ thống | |
| **Lỗi** | Thông tin giáo viên được giữ nguyên | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi phòng đạo tạo, bộ môn muốn quản lý giáo viên trong trường.   * Usecase này bắt đầu khi phòng đào tạo hoặc bộ môn muốn quản lý thông tin giáo viên trong trường. * Phòng đào tạo, bộ môn chọn chức năng quản lý giáo viên trong thanh sidebar. * Hệ thống hiển thị danh sách giáo viên theo form: * STT * Mã giáo viên * Họ tên * Bộ môn * Phòng đạo tạo có thể xem chi tiết thông tin của sinh viên bằng cách click chuột vào giáo viên muốn xem * Hệ thống sẽ hiển thị thông tin chi tiết bao gồm: * Họ tên * Mã giáo viên * Bộ môn * Ngày sinh * Điện thoại * Email giáo viên * Địa chỉ * Lịch giảng dạy * Phòng đào tạo, bộ môn có thể lựa chọn thanh công cụ “Thêm” để thêm mới giáo viên, luồng con “Thêm giáo viên” được thực hiện. * Phòng đào tạo, bộ môn có thể lựa chọn thanh công cụ “Sửa” để sửa thông tin tài khoản, luồng con “Sửa thông tin giáo viên” được thực hiện. * Phòng đào tạo, bộ môn có thể lựa chọn thanh công cự ‘xóa” để xóa giáo viên, luồng con “xóa giáo viên” được thực hiện. * Phòng đào tạo, bộ môn có thể tìm kiếm thông tin giáo viên bằng cách chọn chức năng “tìm kiếm”-luồng con “tìm kiếm giáo viên” được thực hiên. * Luồng con: Thêm giáo viên * Hệ thống hiển thị form với các thông tin: * Họ tên * Mã giáo viên * Ngày sinh * Bộ môn * Điên thoại * Email giáo viên * Địa chỉ * Lịch giảng dạy * Phòng đào tạo hoặc bộ môn nhập thông tin theo form. * Hệ thống thêm mới giáo viên và cập nhật lại danh sách giáo viên. * Luồng con: Sửa thông tin giáo viên * Hệ thống hiển thị form với các thông tin * Họ tên * Mã giáo viên * Ngày sinh * Bộ môn * Điện thoại * Email giáo viên * Địa chỉ * Lịch giảng dạy * Phòng đào tạo, bộ môn sửa lại các thông tin theo form cho sẵn. * Hệ thống cập nhật lại thông tin giáo viên và danh sách giáo viên sau khi phòng đạo tạo, bộ môn xác nhận sửa. * Luồng con – Xóa giáo viên: * Hệ thống hiển thị thông tin chi tiết giáo viên * Phòng đào tạo hoặc bộ môn chọn “xóa bỏ” để yêu cầu xóa giáo viên. * Hệ thống yêu cầu xác nhận lại yêu cầu * Người quản trị chọn “xác nhận” để xóa giáo viên. * Người quản trị chọn “hủy” để hủy bỏ yêu cầu. * Hệ thống cập nhật lại danh sách giáo viên. * Luồng con- Tìm kiếm giáo viên: * Phòng đào tạo, bộ môn nhập từ khóa vào thanh tìm kiếm * Hệ thống sẽ tìm kiếm trong cơ sở dữ liệu và đưa ra những giáo viên phù hợp từ khóa * Phòng đào tạo, bộ môn click vào để xem thông tin chi tiết. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .18 Giao diện xem danh sách giáo viên    Hình 3.19 Giao diện xem chi tiết thông tin giáo viên    Hình 3.20 Giao diện thêm mới giáo viên    Hình 3.21 Giao diện xóa giáo viên    Hình 3.22 Giao diện tìm kiếm giáo viên | | | |

## UC#09 – Ép cứng lớp học phần

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Ép cứng lớp học phần** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Chức năng cho phép phòng đào tạo, bộ môn có thể ép sinh viên vào một lớp học phần trong thời gian đăng ký học hoặc trước thời gian đăng ký học một thời gian quy định. | |
| **Tác nhân** | | Phòng đào tạo, bộ môn. | |
| **Tiền điều kiện** | | Phòng đào tạo, bộ môn đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Lớp học phần được thêm vào thời khóa biểu của sinh viên được chọn | |
| **Lỗi** | Thời khóa biểu được giữ nguyên | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case này bắt đầu khi phòng đào tạo, bộ môn muốn ép cứng lớp học phần vào thời khóa biểu của sinh viên   * Use case này bắt đầu khi phòng đào tạo hoặc bộ môn click vào chức năng ép cứng ở thanh sidebar * Hệ thống hiển thị form “chọn sinh viên ép cứng” * Phòng đào tạo hoặc bộ môn nhập vào form từ khóa và chọn những sinh viên muốn ép cứng môn học. * Hệ thống hiển thị những học phần có thể ép cứng cho sinh viên (tất cả học phần trong chương trình học của sinh viên mà sinh viên chưa học đối với phòng đào tạo, những học phần thuộc bộ môn đối với bộ môn). * Phòng đào tạo hoặc bộ môn click vào tên học phần muốn ép cứng. * Hệ thống hiển thị các lớp học phần thuộc học phần đó * Phòng đào tạo hoặc bộ môn click vào lớp học muốn ép cứng cho sinh viên * Hệ thống thêm lớp học phần vào thời khóa biểu của sinh viên và sinh viên không thể tự xóa lớp học phần này. * Phòng đạo tạo hoặc bộ môn có thể click vào lớp học phần (trong thời khóa biểu hoặc trong phần lớp học phần đăng ký) để xóa lớp học phần – luồng con “xóa lớp học phần” sẽ bắt đầu. * Luồng con – xóa lớp học phần: Phòng đào tạo sẽ xóa được tất cả các lớp học phần, bộ môn chỉ có thể xóa lớp học phần thuộc bộ môn đó. * Phòng đào tạo hoặc bộ môn click vào lớp học phần muốn hủy đang hiển thị trên thời khóa biểu của sinh viên. * Hệ thống sẽ xóa lớp học phần khỏi thời khóa biểu của sinh viên. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .23 Giao diện ép cứng lớp học phần    Hình .24 Giao diện xóa lớp học phần | | | |

## UC#10- Đăng ký lớp học phần

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Đăng ký lớp học phần** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Chức năng cho phép sinh viên có thể đăng ký lớp học phần phù hợp để học trong kì tiếp theo trong thời gian quy định của phòng đào tạo. | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Sinh viên đăng nhập vào hệ thống | |
| **Hậu điều kiện** | **Thành công** | Sinh viên có thời khóa biểu kì tới của mình | |
| **Lỗi** | Sinh viên không có thời khóa biểu kì tới | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi sinh viên muốn đăng ký lớp học phần.   * Use case này bắt đầu khi sinh viên click vào chức năng đăng ký học trong thanh sidebar * Hệ thống hiển thị thông báo đợt đăng ký học trong thời gian cho phép hiển thị (thời gian delay) dành cho sinh viên gồm thời gian bắt đầu, thời gian kết thúc, đối tượng được đăng ký học và thời khóa biểu gồm danh sách môn đăng ký được. * Nếu sinh viên không trong thời gian đăng ký học thì hệ thống thông báo “Bạn không phải là đối tượng được đăng ký học” * Sinh viên click vào đợt đăng ký học mà hệ thống hiển thị nếu sinh viên đang trong thời gian đăng ký học mà hệ thống thông báo * Hệ thống hiển thị danh sách các học phần mà sinh viên có thể đăng ký học * Sinh viên click vào học phần - luồng con “đăng ký lớp học phần” sẽ được thực hiện * Sinh viên có thể bỏ lớp đã đăng ký – luồng con “xóa lớp học phần” được thực hiện. * Luồng con: đăng ký lớp học phần * Hệ thống hiển thị danh sách các lớp học phần bao gồm các thông tin: * Tên * Thứ * Ca * Số lượng người hiện tại * Sinh viên click vào lớp học phần muốn đăng ký. * Hệ thống thông báo thành công và thêm lớp học phần vào thời khóa biểu của sinh viên * Hệ thống cập nhật lại danh sách lớp học phần * Nếu không thành công, luồng con A được thực hiện * Luồng con: xóa lớp học phần * Hệ thống hiển thị danh sách các lớp học phần * Sinh viên click vào lớp học phần đã đăng ký * Hệ thống cập nhật lại danh sách lớp học phần * Hệ thống xóa lớp học phần khỏi thời khóa biểu | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Luồng A:   * Hệ thống thông báo 1 trong lỗi sau * “Lớp đã đầy” nếu lớp học phần đã đủ số sinh viên * “Trùng thời gian học” nếu trong thới khóa biểu có lớp học phần trùng thời gian với lớp đăng ký * “Quá tín chỉ cho phép” nếu tổng số tín đăng ký + tín nợ >=20 | | | |
| **Giao diện minh họa** | | | |
| C:\Users\b360\Downloads\82358636_794307934378763_4737343133577117696_n.png  Hình .25 Giao diện hiển thị danh sách học phần đăng ký  C:\Users\b360\Downloads\82181981_596821747815671_2401834244084072448_n.png  Hình 3.26 Giao diện hiển thị các lớp học phần  C:\Users\b360\Downloads\81762332_575946476295673_5946070720218923008_n.png  Hình 3.27 Giao diện hiển thị các môn đã đăng ký  C:\Users\b360\Downloads\81778368_602575097210602_6823161256048852992_n.png  Hình 3.28 Giao diện hiển thị lỗi | | | |

## UC#11 - Xem điểm

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Xem điểm** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Chức năng cho phép sinh viên có thể xem và theo dõi kết quả học tập của mình. | |
| **Tác nhân** | | Sinh viên. | |
| **Tiền điều kiện** | | Sinh viên đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Không có | |
| **Lỗi** | Không có | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi sinh viên muốn xem kết quả học tập của mình   * Use case bắt đầu khi sinh viên click vào chức năng “Bảng điểm” ở thanh sidebar * Hệ thống hiển thị ra các thông tin gồm có: * Thông tin cá nhân sinh viên gồm tên, mã sinh viên, ngày sinh, lớp. * Tổng số tín chỉ tích lũy của sinh viên * Trung bình chung tích lũy điểm của sinh viên. * Tổng số tín chỉ đang đợi kết quả. * Bảng điểm chi tiết ngành học chính của sinh viên bao gồm các thông tin mã học phần, tên học phần, số tín chỉ và điểm. * Sinh viên chọn ngành học, chọn ngành học thứ 2 của mình (nếu có): * Hệ thống hiển thị thông tin bảng điểm ngành học phụ của sinh viên. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .29 Giao diện xem điểm | | | |

## UC#12 - Xem lịch học

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Xem lịch học** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Chức năng cho phép sinh viên có thể xem thời khóa biểu từng kì học của mình | |
| **Tác nhân** | | Sinh viên. | |
| **Tiền điều kiện** | | Sinh viên đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | Không có | |
| **Lỗi** | Không có | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi sinh viên muốn xem thời khóa biểu của mình   * Use case bắt đầu khi sinh viên click vào chức năng “Thời khóa biểu” ở thanh sidebar * Hệ thống hiển thị thông tin gồm có: * Năm học mặc định hiện tại của sinh viên * Học kì mặc định hiện tại của sinh viên * Thời khóa biểu mặc định học kì hiện tại của sinh viên * Học phí tương ứng của học kì * Thời gian các ca học * Sinh viên click vào năm học và học kì để chọn thời khóa biểu kì học cần xem * Hệ thống hiển thị thông tin gồm có: * Năm học đã chọn * Học kì đã chọn * Thời khóa biểu của kì học năm học đã chọn * Học phí của kì đó * Thời gian các các ca học | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .30 Giao diện thời khóa biểu | | | |

## UC#13 - Xem lịch giảng dạy

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Xem lịch giảng dạy** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Chức năng cho phép giáo viên có thể xem lịch giảng dạy từng kì học của mình | |
| **Tác nhân** | | Giáo viên | |
| **Tiền điều kiện** | | Giáo viên đăng nhập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** | In thành công danh sách sinh viên của lớp | |
| **Lỗi** | Không có | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi giảng viên muốn xem lịch giảng dạy của mình   * Use case bắt đầu khi giảng viên click vào chức năng “Lịch giảng dạy” ở thanh sidebar * Hệ thống hiển thị thông tin gồm có: * Năm học mặc định hiện tại * Học kì mặc định hiện tại * Lịch giảng dạy * Thời gian các ca học * Giảng viên click vào năm học và học kì để chọn lịch giảng dạy kì học cần xem * Hệ thống hiển thị thông tin gồm có: * Năm học đã chọn * Học kì đã chọn * Lịch giảng dạy của kì đã chọn * Thời gian các các ca học * Giảng viên chọn vào lớp trong lịch giảng dạy để xem danh sách sinh viên lớp đấy – luồng con “xem danh sách sinh viên” được thực hiện. * Luồng con – xem danh sách sinh viên: * Hệ thống hiển thị danh sách sinh viên lớp giảng viên đã chọn là một bảng gồm các thông tin: * Số thứ tự * Mã sinh viên * Họ và tên * Lớp * Giáo viên chọn “đóng” để thoát khỏi giao diện xem danh sách sinh viên. * Giáo viên chọn “in danh sách” để in ra danh sách sinh viên, luông con “in danh sách sinh viên sẽ được thực hiện” * Luồng con – in danh sách sinh viên: * Hệ thống hiển thị form gồm các tùy chỉnh cho việc in tùy theo từng trình duyệt web * Giáo viên tùy chỉnh và chọn “in” để in * Hệ thống kết nối với máy in và máy in sẽ cho ra danh sách sinh viên | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình .31 Giao diện xem lịch giảng dạy    Hình 3.32 Giao diện xem danh sách sinh viên    Hình 3.33 Giao diện in danh sách sinh viên | | | |

## Thống kê

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Thống kê** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Chức năng này cho phép các tác nhân tham gia hệ thống có thể xem được các thông tin một cách tổng quan từ trang chủ của mình | |
| **Tác nhân** | | Tất cả các tác nhân | |
| **Tiền điều kiện** | | Cần đăng nhập vào hệ thống | |
| **Hậu điều kiện** | **Thành công** | Hiển thị các thông tin cần thiết | |
| **Lỗi** | Không có | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| * Chức năng bắt đầu khi người dùng đăng nhập vào hệ thống. * Hệ thống hiển thị trên trang chủ thông tin tổng quang dựa theo quyền truy cập của tài khoản ngay trên trang chủ của giao diện của tài khoản đó. * Luôn có thanh công cụ tìm kiếm cho tác nhân trong từng chức năng để tác nhân có thể xem nhanh các thông tin một cách nhanh chóng. * Tác nhân nhập thông tin trên thanh tìm kiếm, hệ thống sẽ hiển thị ra danh sách các thông tin liên quan đến từ khóa. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| Không có | | | |
| **Giao diện minh họa** | | | |
| Hình 3.34 . Thống kê lượng sinh viên theo tín chỉ đăng ký | | | |

# Thuật Toán đăng ký học

## Ý tưởng thuật toán

Trước mỗi kỳ học của từng nhóm, phòng đào tạo sẽ tạo ra các đợt đăng ký học cho các sinh viên, tối đa 3 đợt. Trong khoảng thời gian này, các sinh viên này sẽ có thể tham gia đăng ký học, danh sách các môn hiển thị cho sinh viên đăng ký sẽ dựa theo các môn học có lớp học phần được mở trong kỳ(theo thời khóa biểu toàn trường), các môn học này có trong chương trình học của từng sinh viên và dựa theo tiến trình học của từng sinh viên đó. Các sinh viên có thể chọn lớp học phần phù hợp trong từng môn để tạo thành TKB cho kỳ học sau.

Sau khi kết thúc quá trình đăng ký học, SV vẫn có thể xem lại TKB của bản thân cũng như học phí của kỳ học.

Các giáo viên cũng có thể sử dụng phần mềm để xem lịch giảng dạy của bản thân.

## Chuẩn bị dữ liệu

* Lấy danh sách thời khóa biểu toàn trường của kỳ học tới của nhóm cần tạo đợt đăng ký học
* Lấy thông tin về tiến trình học của các sinh viên trong nhóm(bảng điểm)
* Lấy chương trình học của từng ngành học

## Thuật toán

a) Thuật toán Đăng ký học

1. Hệ thống tìm đợt đăng ký học tương ứng với các thông tin sau của sinh viên:

- gồm nhóm sinh viên

- kì học, niên khóa

- năm học hiện tại

1.1 Nếu không tìm thấy đợt đăng ký học nào. Hệ thống báo “bạn không trong thời gian đăng ký học”.

1.2 Nếu tìm thấy, hệ thống sẽ kiểm tra thời gian tính từ hiện tại đến thời gian bắt đầu của đợt đăng ký học nhỏ gần nhất (một đợt đăng ký lớn có thể bao gồm nhiều nhất 3 đợt đăng ký nhỏ), khi đó:

1.2.1 Nếu khoảng thời gian này nhỏ hơn hoặc bằng thời gian delay (thời gian giữa thời điểm hiện tại và thời gian bắt đầu đợt đăng ký học gần nhất), hệ thống thông báo sắp tới đợt đăng ký học cho sinh viên.

1.2.2 Nếu thời gian hiện tại vượt qua thời gian kết thúc của các đợt đăng ký nhỏ, hệ thống thông báo “đã quá thời gian đăng ký”.

1.2.3 Nếu đang trong thời gian đăng ký, hệ thống tìm trong thời khóa biểu toàn trường các lớp học phần có đặc điểm sau:

* Phù hợp với năm học, kì học, nhóm học của sinh viên
* Nằm trong chương trình học của sinh viên và không nằm trong bảng điểm của sinh viên (nếu nằm trong bảng điểm thì có điểm là “-1”).
* Học phần chứa lớp học phần không có điều kiện tiên quyết (là mảng các mã môn), nếu có, điều kiện tiên quyết nằm trong bảng điểm của sinh viên và điểm >5.
* Tín chỉ tiên quyết <= tổng số tín chỉ tích lũy trong bảng điểm của sinh viên.

2. Sau đó, hệ thống gộp dữ liệu lớp học phần, nhóm lớp học phần thỏa mãn các điều kiện trên theo học phần, tạo thành 1 danh sách các môn có thể đăng ký cho SV

3. SV chọn các môn phù hợp với mong muốn và tiến trình học của bản thân để tạo thành TKB cho mình (các môn học không thể có thời gian học của các lớp học phần trùng nhau)

b) Thuật toán tạo nhóm lớp học phần

1. Với những học phần trong phần import thời khóa biểu toàn trường, xét tên từng lớp học phần thuộc học phần.

2.1 Tách tên thành từng phần theo ví dụ: CAUTRUCDULIEU.2.1\_BT

* Phần tên: CAUTRUCDULIEU
* Phần phiên bản: .2.1
* Phần kiểu lớp: có 2 kiểu bài tập(BT) hoặc lí thuyết(LT). ở đây là BT

2.2 Trong phần phiên bản, tổng hợp các kí tự sau kí tự “.” thành 1 mảng (mảng phiên bản), sau đó sắp xếp các lớp học phần theo độ dài của mảng phiên bản từ cao xuống thấp.

2.3 Tạo 1 đối tượng lưu tạm thời kí hiệu là x

2.4 Duyệt các lớp học phần theo thứ tự sắp xếp:

* Nếu phiên bản nào chưa nằm trong x, tạo 1 lớp đăng kí mới.
* Nếu phiên bản nào nằm trong x rồi, nhét lớp học phần vào lớp đăng ký đã tồn tại.
* Với những lớp học phần có mảng phiên bản ngắn hơn, kiểm tra phần phiên bản tương ứng có thuộc phần bắt đầu của phần phiên bản thuộc lớp đăng ký không, nếu có cho vào lớp đăng ký, nếu không có, tạo 1 lớp mới.

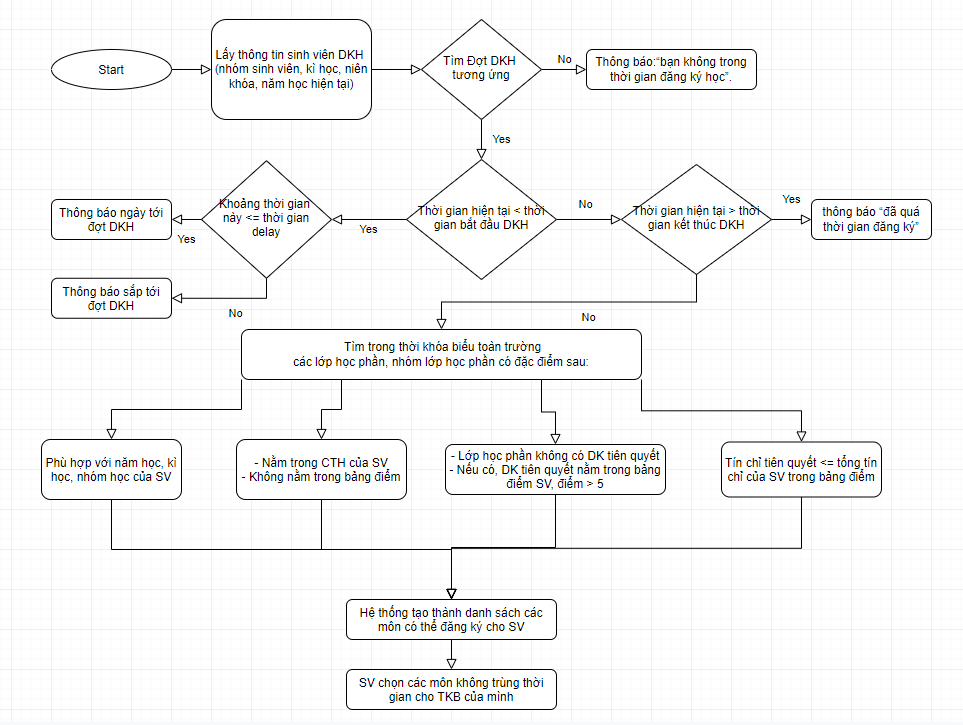
c) Thuật toán tạo nhóm sinh viên

- Nếu Sinh viên là niên khóa mới nhất -> nhóm 1

- Nếu không phải niên khóa mới nhất và thuộc ngành quản lý – kinh tế thì thuộc nhóm 3

- Còn lại là nhóm 2

## Sơ đồ thuật toán DKH

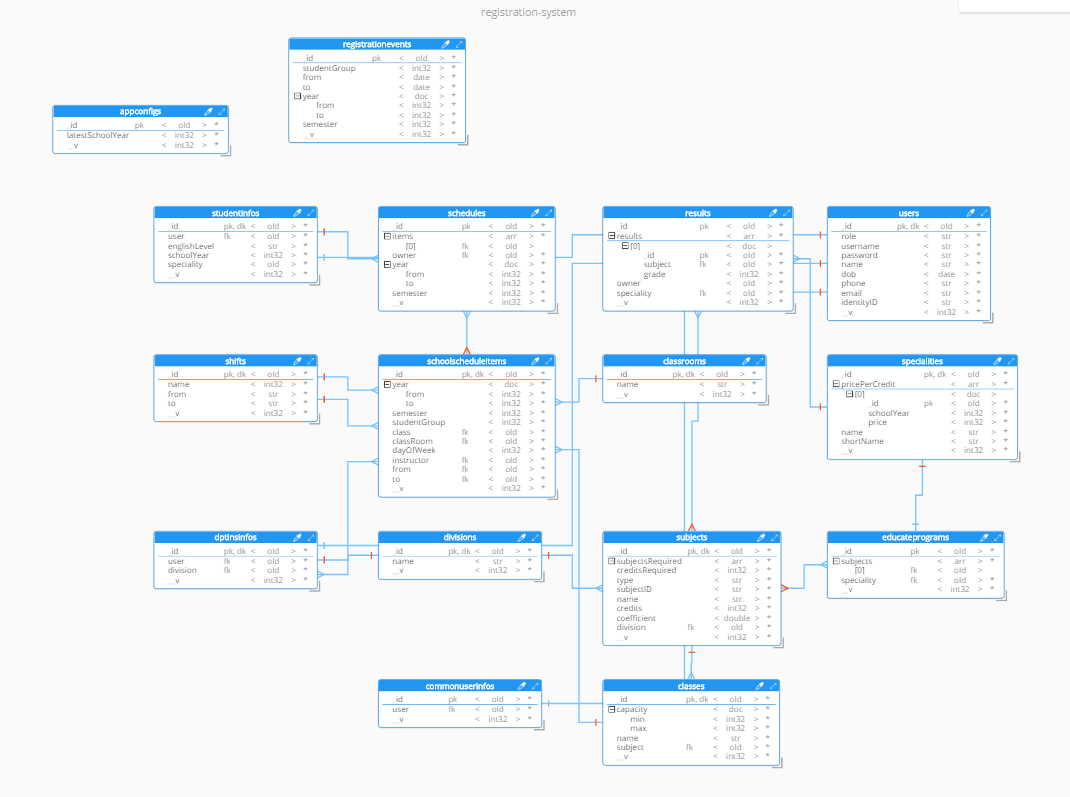


Hình 4.1 Sơ đồ thuật toán đăng ký học

# Thiết kế CSDL và Api

## Thiết kế CSDL

### Sơ đồ kiến trúc CSDL



Hình 5.1 Sơ đồ kiến trúc CSDL

### Bảng dữ liệu

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Collection** | **Mô tả** |
| 1. | App-config | Lưu niên khóa , kỳ, năm |
| 2. | Class-room | Danh sách phòng học |
| 3. | Class | Danh sách lớp học phần |
| 4. | Common-user-info | Danh sách thông tin người dùng của admin và phòng đào tạo |
| 5. | Division | Danh sách bộ môn |
| 6. | Dpt-ins-info | Thông tin của giảng viên và bộ môn |
| 7. | Educate-program | Chương trình học |
| 8. | Registration-event | Đợt đăng ký học |
| 9. | Result | Bảng điểm sinh viên |
| 10. | Schedule | Thời khóa biểu sinh viên |
| 12. | School-schedule-items | Thời khóa biểu toàn trường |
| 13. | Shift | Danh sách ca học |
| 14. | Speciality | Chuyên ngành |
| 15. | Student-info | Thông tin sinh viên |
| 16. | Subject | Danh sách môn học |
| 17. | User | Danh sách tài khoản |

#### App-config\_ Lưu niên khóa, kỳ, năm

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | latestSchoolYear | Number | Niên khóa hiện tại |
| 2. | currentSemester | Number | Kỳ học hiện tại |
| 3. | currentYear(from-to) | Number | Năm học hiện tại |
| 4. | pricePerCredit | Number | Giá tín chỉ |

#### Class-room \_ Danh sách phòng học

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Name | String | Tên phòng học |

#### Class \_ Danh sách lớp học phần

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Name | String | Tên lớp học phần |
| 2. | Subject | Number | Môn học của lớp học phần |
| 3. | Capacity(min-max) | Number | -min: số lượng sinh viên thấp nhất  -max:số lượng sinh viên tối đa của lớp |

#### Common-user-info \_ Danh sách thông tin người dùng của admin và phòng đào tạo

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | User | ObjectId | Id tài khoản phòng đào tạo  (lấy dữ liệu bảng user) |

#### Division \_ Danh sách bộ môn

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Name | String | Tên bộ môn |

#### Dpt-ins-info \_ Thông tin của giảng viên và bộ môn

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | User | ObjectId | Id tài khoản phòng đào tạo  (lấy dữ liệu bảng user) |
| 2. | Division | ObjectId | Id bộ môn  (lấy dữ liệu bảng division) |

#### Educate-program \_ Chương trình học

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Subjects | ObjectId | Id môn học  (lấy dữ liệu bảng subjects) |
| 2. | Division | ObjectId | Id bộ môn  (lấy dữ liệu bảng division) |

#### Registration-event \_ Đợt đăng ký học

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | studentGroup | Number | Nhóm của sinh viên |
| 2. | Active | Boolean | Có đợt DKH không |
| 3. | childEvents | -from: Date  -delay: String  -to: Date | Thời gian các đợt đăng ký học |
| 4. | Year(from-to) | Number | Năm |
| 5. | Semester | Number | Kỳ |

#### Result\_Bảng điểm sinh viên

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Owner | ObjectId | Id sinh viên  (lấy dữ liệu bảng studentInfo) |
| 2. | Speciality | ObjectId | Id chuyên ngành  (lấy dữ liệu bảng speciality) |
| 3. | Results | -subject:  ObjectId  -grade: Number | Điểm |

#### Schedule\_Thời khóa biểu sinh viên

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | List | ObjectId | Danh sách môn học |
| 2. | Owner | ObjectId | Id sinh viên  (lấy dữ liệu bảng studentInfo) |
| 3. | Active | Boolean | Có đợt DKH không |
| 4. | Year(from-to) | Number | Năm |
| 5. | Semester | Number | Kỳ |

#### School-schedule-items\_Thời khóa biểu toàn trường

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Year(from-to) | Number | Năm |
| 2. | Semester | Number | Kỳ |
| 3. | studentGroup | Number | Nhóm của sinh viên |
| 4. | Class | ObjectId | Id lớp  (lấy dữ liệu bảng Class) |
| 5. | classRoom | ObjectId | Id phòng học  (lấy dữ liệu bảng ClassRoom) |
| 6. | dayOfWeek | Number | Thứ mấy trong tuần |
| 7. | Instructor | ObjectId | Id giáo viên  (lấy dữ liệu bảng DptInsInfo) |
| 8. | From | ObjectId | Id ca  (lấy dữ liệu bảng Shift) |
| 9. | To | ObjectId | Id ca  (lấy dữ liệu bảng Shift) |

#### Shift\_Danh sách ca học

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Name | Number | Ca học |
| 2. | From | String | Giờ bắt đầu |
| 3. | To | String | Giờ kết thúc |

#### Speciality\_Chuyên ngành

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Name | String | Tên chuyên ngành |
| 2. | shortName | String | Mã ngành |
| 3. | Department | String | Bộ môn |

#### Student-info\_Thông tin sinh viên

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | User | ObjectId | Id tài khoản sinh viên  (lấy dữ liệu bảng user) |
| 2. | englishLevel | String | Mức Tiếng Anh |
| 3. | schoolYear | Number | Niên khóa |
| 4. | Speciality | ObjectId | Id chuyên ngành  (lấy dữ liệu bảng Speciality) |
| 5. | credits | Number | Số tín chỉ |

#### Subject\_Danh sách môn học

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | subjectID | String | mã môn học |
| 2. | Name | String | tên môn học |
| 3. | credits | Number | Số tín chỉ |
| 4. | Coefficient | Number | ? |
| 5. | subjectsRequired | String | Điều kiện tiên quyết(môn) |
| 6. | creditsRequired | Number | Điều kiện tiên quyết(số tín chỉ) |
| 7. | Division | ObjectId | Id bộ môn  (lấy dữ liệu bảng Division) |

#### User\_Danh sách tài khoản

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** | **Mô tả** |
| 1. | Username | String | Tên tài khoản |
| 2. | Password | String | Mật khẩu |
| 3. | Role | String | Vai trò |
| 4. | identityID | String | Mã |
| 5. | Name | String | tên |
| 6. | Dob | Date | Ngày sinh |
| 7. | Phone | String | Số điện thoại |
| 8. | Email | String | Email |

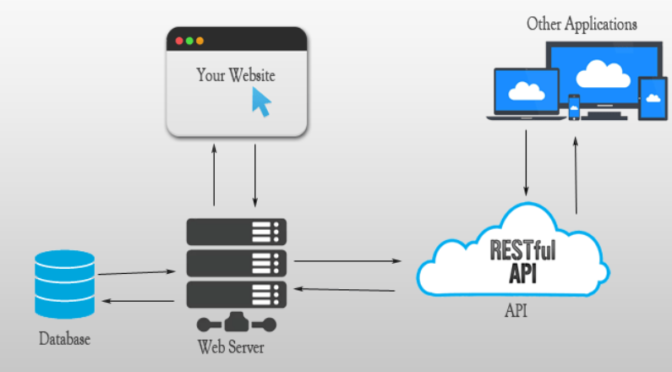
## Thiết kế API

### Giới thiệu chung về API

API là từ viết tắt của Application Programming Interface (giao diện lập trình ứng dụng), là phương thức kết nối giữa các thư viện và ứng dụng khác nhau với mục đích cuối là giúp ứng dụng của người dùng có thể truy cập đến một tệp các hàm thực hiện chức năng như tương tác hoặc trao đổi thông tin với nhau.

API không phải là một ngôn ngữ lập trình. Các hàm API cũng tương tự như các hàm thông thường khác.

API đã ra đời từ lâu, được ứng dụng ở nhiều loại ứng dụng và phần mềm khác nhau. Thế hệ mới nhất của API, Web API có thể dùng cho mọi hệ thống (hệ điều hành, cơ sở dữ liệu, hệ thống nền web hay còn gọi là web-based system, thư viện ứng dụng,..) thậm chí cả phần cứng máy tính.



### Các API được sử dụng trong phần mềm Đăng ký học

* + - 1. ***Chương trình học***

|  |  |  |
| --- | --- | --- |
| **METHOD** | **URI** | **Mô** **tả** |
| GET | /api/edu-programs/ | Lấy ra tất cả các chương trình học đang có trong cơ sở dữ liệu. |
| GET | /api/edu-program/ | Lấy ra một chương trình học dựa theo tham số là :   * SpecialityID (mã chuyên ngành) |

* + - 1. ***Đăng ký học***

|  |  |  |
| --- | --- | --- |
| **METHOD** | **URI** | **Mô** **tả** |
| GET | /api/registration-event/overview/ | Lấy ra số lượng sinh viên tham gia đợt đăng ký dựa vào tham số :   * StudentGroup(nhóm sinh viên) |
| GET | /api/registration-event/full-overview/semester/:semester/year/:year | Lấy ra số lượng sinh viên có số tín chỉ chia làm các mốc sau :   * >=15 tín chỉ * >=12 tín chỉ * >=10 tín chỉ * <10 tín chỉ   Các tham số :   * Semester(kì học) * Year(Năm học) |
| DETELE | /api/registration-event/:rID/ | Xóa đợt đăng ký học có id là rID |
| POST | /api/ registration-event /create/ | Tạo đợt đăng ký mới |
| GET | /api/registration-event/all/ | Lấy ra toàn bộ đợt đăng ký học đang có trong cơ sở dữ liệu  Tham số là :   * Year(năm) * Semester(kì) * Studentgroup(nhóm) |
| GET | /api/registration-event/force-registration/subjects/ | Lấy tất cả các lớp học phần trong thời khóa biểu toàn trường dựa theo một số điều kiện để ép cứng.  Tham số có thể là 1 trong 2 :   * InstructerID * PdtID |
| GET | /api/registration-event/subjects/ | Lấy tất cả các lớp học phần trong thời khóa biểu toàn trường dựa theo một số điều kiện để sinh viên đăng kí.  Tham số là :   * StudentID |
| GET | /api/registration-event/:rID/ | Lấy ra toàn bộ thông tin một đợt đăng ký có ID là rID |
| PUT | /api/registration-event/:rID/ | Cập nhật thông tin một đợt đăng ký có ID là rID |
| GET | /api//registration-event/semester/:semester/year/:year/subject-info/ | Lấy thông tin chi tiết của 1 nhóm lớp học phần bao gồm số lượng sinh viên đăng ký vào lớp, thông tin các lớp thuộc nhóm lớp)  Tham số là :   * Semester * Year * SubjectID |

* + - 1. ***Bảng điểm***

|  |  |  |
| --- | --- | --- |
| **METHOD** | **URI** | **Mô** **tả** |
| POST | /api/result/upload/ | Biến đổi dữ liệu thô thành dữ liệu chuẩn. |
| POST | /api/result/import/ | Gửi dữ liệu chuẩn và lưu vào database |
| GET | /api/result/student/:studentID/ | Lấy bảng điểm của sinh viên dựa theo ID của sinh viên.  Tham số là:   * StudentID * SpecialityID(mã chuyên ngành) |

* + - 1. ***Thời khóa biểu***

|  |  |  |
| --- | --- | --- |
| **METHOD** | **URI** | **Mô** **tả** |
| PUT | /api/schedule/student/:studentID/semester/:semester/year/:year/toggle-register/ | Đăng ký vào một nhóm lớp học phần.  Tham số là:   * StudentID * Semester * Year |
| GET | /api/ schedule/student/:studentID/semester/:semester/year/:year/ | Lấy ra thời khóa biểu của một sinh viên dựa theo ID của sinh viên  Tham số là :   * StudentID * Semester * Year |

* + - 1. ***Thời khóa biểu toàn trường***

|  |  |  |
| --- | --- | --- |
| **METHOD** | **URI** | **Mô** **tả** |
| POST | /api//school-schedule/upload/ | Biến đổi dữ liệu thô thành dữ liệu chuẩn |
| POST | /api/school-schedule/import/ | Gửi dữ liệu chuẩn và lưu vào database |
| GET | /api/school-schedule/all/ | Lấy ra tất cả dữ liệu trong thời khóa biểu toàn trường |
| POST | /api/school-schedule/lessons/ | Lấy ra tất cả các nhóm lớp học phần dựa vào một lớp học phần. |
| GET | /api/school-schedule/division/:divisonID/ | Lấy ra những lớp học phần trong thời khóa biểu toàn trường theo một bộ môn.  Tham số là :   * DivisionID * Year * Semester * Keyword |
| PUT | /api/school-schedule/disable-items/ | Để hủy nhóm lớp học phần. |
| GET | /api/school-schedule/instructor-schedule/:instructorID/ | Lấy ra lịch giảng dạy của một giáo viên  Tham số là:   * InstructerID * Semester * Year |
| GET | /api/shift/overview/ | Lấy ra thông tin cơ bản(tên,id và thời gian) về tất cả các kì học |

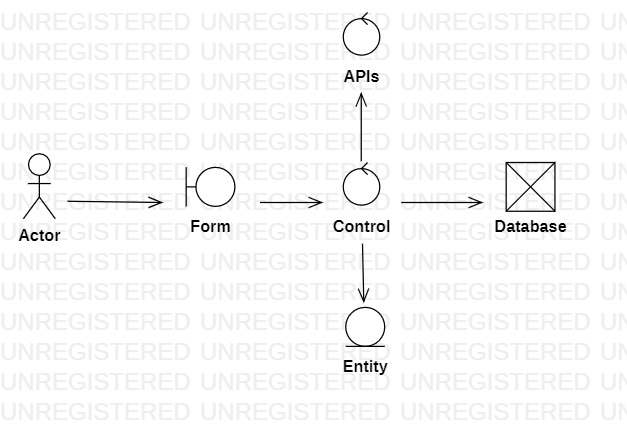
* + - 1. ***Tài khoản người dùng***

|  |  |  |
| --- | --- | --- |
| **METHOD** | **URI** | **Mô** **tả** |
| GET | /api/auth / | Phụ trách các chức năng như : kiểm tra mật khẩu, lấy thông tin người dùng đã đăng nhập. |
| POST | /api/login/ | Dùng để đăng nhập |
| GET | /api/user/all/ | Lấy ra tất cả người dùng có trong cơ sở dữ liệu |
| PUT | /api/user/:acountID/ | Cập nhật thông tin người dùng có id là acoountID |
| GET | /api/user/:acountID/ | Lấy thông tin chi tiết của người dùng có id là acountID |
| DETELE | /api/acount/:acountID/ | Xóa tài khoản người dùng có id là acountID |

# Thiết kế và cài đặt các chức năng

## Thiết kế các chức năng

### Cấu trúc chung



Hình 6.1: Cấu trúc lớp chung của chức năng trên web

- Actor: Tác nhân tham gia hệ thống

- Form: Form là lớp trung gian, nơi tương tác giữa tác nhân và hệ thống (Là giao diện giữa người dùng và hệ thống).

- Entity: Entity (lớp thực thể) chứa thông tin của các thực thể tham gia hệ thống

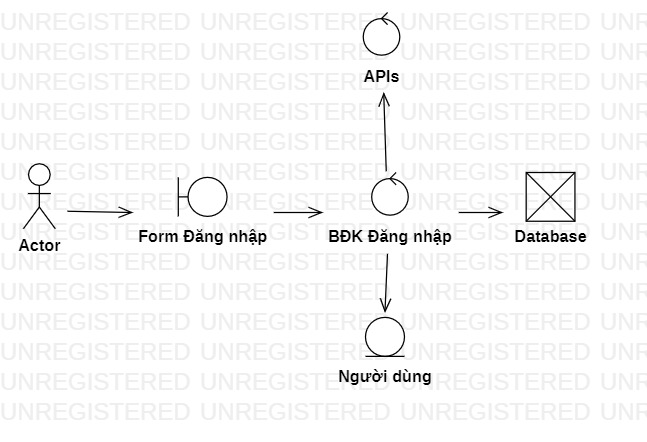
- Control: Control (lớp điều khiển) điều hướng hoạt động của các chức năng trong Usecase

- APIs: Lớp điều khiển thực hiện việc nhận/trả các API.

- Database: cơ sở dữ liệu.

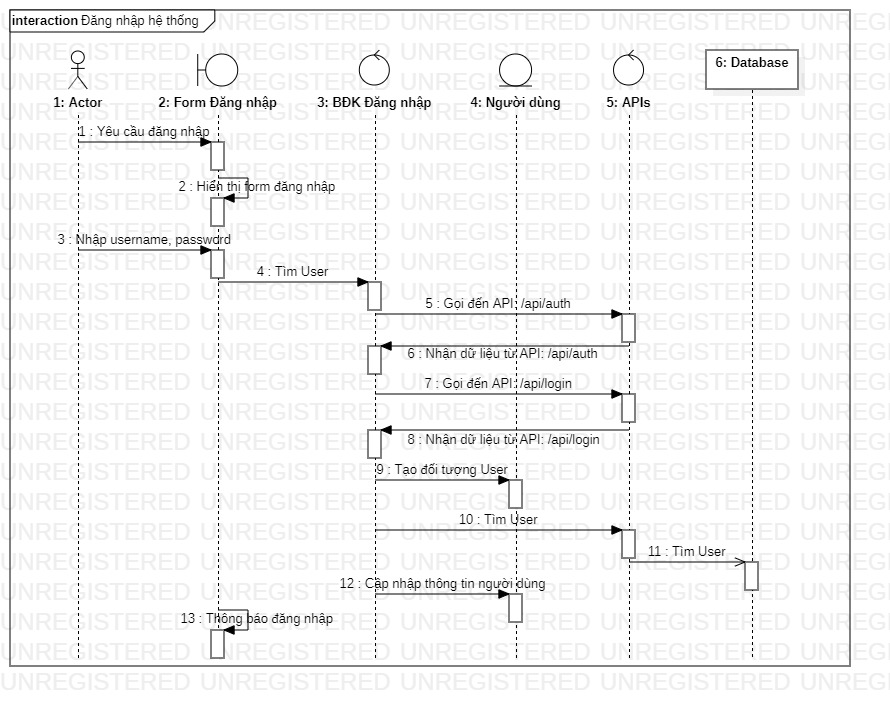
### Thiết kế chức năng đăng nhập:

#### Sơ đồ lớp phân tích:



Hình 6.2: Sơ đồ lớp phân tích đăng nhập

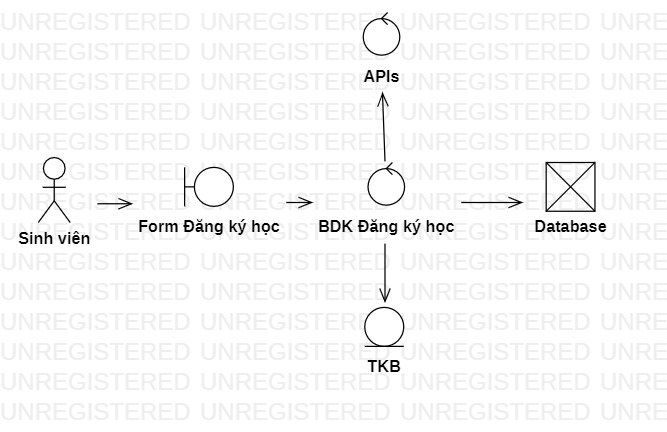
#### Sơ đồ lớp trình tự:



Hình 6.3: Sơ đồ lớp trình tự đăng nhập

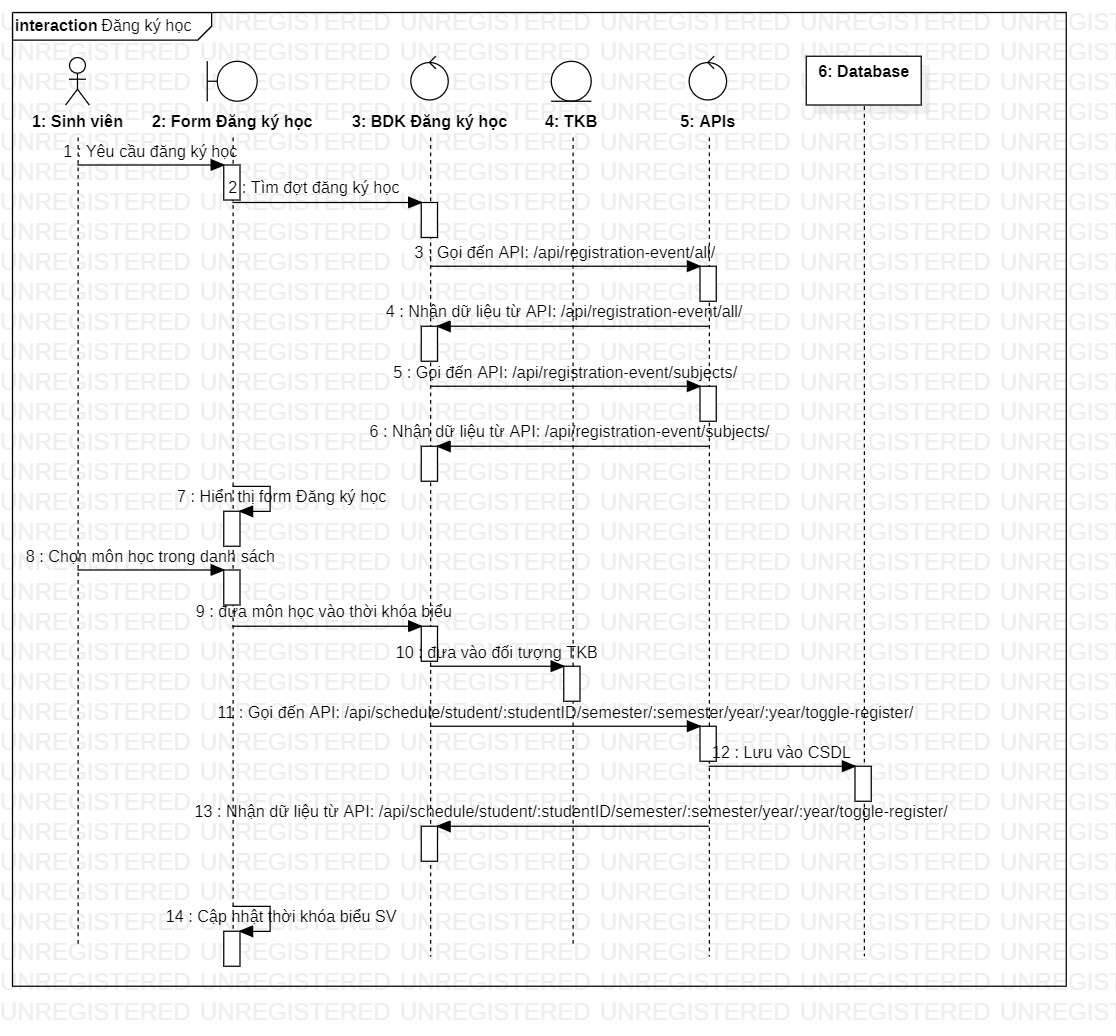
### Thiết kế chức năng đăng ký học:

#### Sơ đồ lớp phân tích:



Hình 6.4: Sơ đồ lớp phân tích đăng ký học

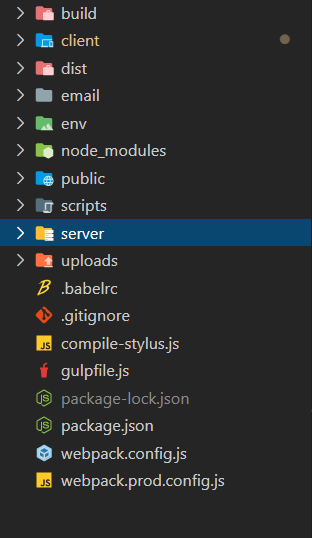
#### Sơ đồ lớp trình tự:



Hình 6.5: Sơ đồ lớp trình tự đăng ký học

## Cài đặt các chức năng

### Cấu trúc thư mục mã nguồn web

Mã nguồn của dự án DKH có 2 phần chính cần được chú ý: server, client (tạo nên phần website của dự án). Mỗi phần chứa nhiều file, dễ bị trùng lặp nên mỗi phần được để trong một thư mục để việc quản lý và phát triển dự án diễn ra nhanh chóng và thuận lợi hơn.

Hình 6.6 Cấu trúc chung thư mục mã nguồn

#### Cấu trúc chung của server

Hình 6.7 Cấu trúc chung thư mục server

Mỗi file cần thiết được phần tán đến các thư mục riêng biệt(các thư mục độc lập như config, controller, db,…) dựa trên các chức năng và mục đích rõ ràng. Do đó để có thể hiểu được luồn hoạt động của server, ta cần đi chi tiết vào từng phần:

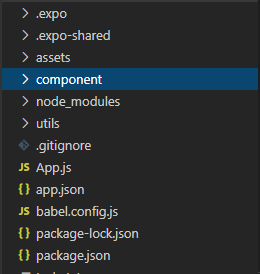
* config: chứa các cài đặt cơ bản về cơ sơ dữ liệu, khung máy chủ,… khi khởi động server
* controllers: chứa các file điều hướng api cho việc giao tiếp giữa client và server
* db: chứa các thông tin của CSDL
* scripts: chứa các dữ liệu đã được thêm vào trước đó với mongoose
* utils: một số thành phần hỗ trợ việc xây dựng và phát triển phần mềm

#### Cấu trúc chung của client

Hình 6.8 Cấu trúc chung thư mục client

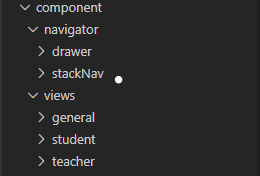
* Api: các điều hướng từ client lên server
* Common: các component(thành phần) nhỏ được build từ trước, được thêm vào để tăng tốc xây dựng phần mềm
* Const: các dữ liệu ban đầu đã được khóa cứng
* React : nơi chứa các file về thiết kế, bố cục, giao diện website, hướng di chuyển (router) của các thành phần
* Style: một số file giao diện cơ bản.

### Cấu trúc thư mục mã nguồn mobile



Hình 6.9 Cấu trúc chung thư mục mã nguồn mobile

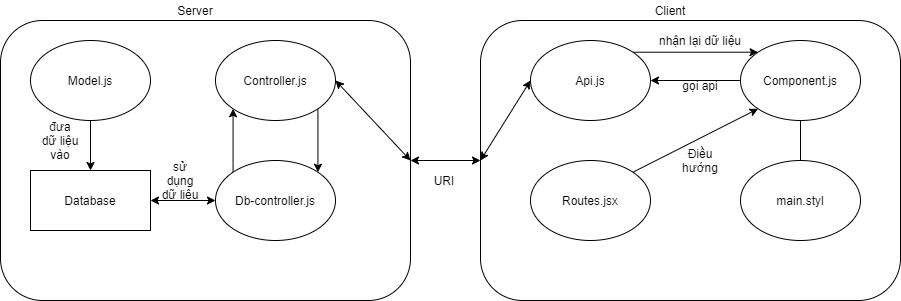
* Asset: các tài nguyên cho giao diện như logo, splash gif…
* Component: nơi chứa các component để render giao diện ra màn hình & các điều hướng
* Node\_modules: nơi chứa các thư viện dùng để xây dựng chương trình
* Utils: một số chức năng sử dụng để xây dựng chương trình



Hình 6.10 Cấu trúc trong thư mục component

* Navigator: nơi chứa các điều hướng
  + Drawer: là giao diện của sidebar dùng để điều hướng
  + stackNav: là nơi quản lý các điều hướng đến các component khác
* views: nơi chứa giao diện của người dùng
  + general: là nơi chứa những component giao diện có thể dùng chung như: đăng nhập, loading…
  + student: là những component giao diện dành riêng cho học sinh
  + teacher: là những component giao diện dành riêng cho giáo viên

### Cấu trúc chung của 1 chức năng



Hình 6.11 Sơ đồ cấu trúc chung 1 chức năng

#### Server

* Model.js: Khai báo cấu trúc dữ liệu của từng tài nguyên
* Db-controller.js: xử lý các vấn đề như lấy, tạo,… của từng tài nguyên
* Controller.js: nhận các URL từ client, xử lý với các db-controller rồi truyền lại

#### Client

* Api.js : điều hướng với các URL lên server
* Routes.jsx: điều hướng các luân chuyển giữa các component của react
* Component.jsx: các phần của giao diện, được ghép với nhau tạo thành giao diện hoàn chỉnh
* Main.styl: điều chỉnh vẻ ngoài của giao diện

### Cấu trúc chức năng đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **File mã nguồn** | **Ý nghĩa** | **Ánh xạ lớp phân tích** |
| 1 | client/api/common/user-api.js | Phụ trách các chức năng như:  -Kiểm tra tài khoản và mật khẩu để login  -Lấy các thông tin người dùng, token  -Gửi email quên mật khẩu | User(API) |
| 2 | client/react/routes/guest-routes/login/login.jsx | -Xây dựng cấu trúc của view hiển thị cho người dùng  -Các function kiểm tra thông tin login, kiểm soát các lỗi khi nhập tài khoản, lấy lại mật khẩu cũ và đưa người dùng vào trang chủ hoặc nơi cuối cùng trước khi mất token khi đăng nhập thành công. | User(Form đăng nhập) |
| 3 | client/react/routes/guest-routes/login/login.styl | Định dạng các style dành cho view |
| 4 | Server/controllers/user.js | Nơi tiếp nhận các URL của user-api, xử lý với các function và gửi lại các data | User(BDK đăng nhập) |
| 5 | Server/db/db-controllers/user.js | Nơi chứa các function của user, thực hiện các chức năng như lấy role của user, đăng nhập,… |

### Cấu trúc chức năng đăng ký học

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **File mã nguồn** | **Ý nghĩa** | **Ánh xạ lớp phân tích** |
| 1 | client/api/common/registration-event-api.js | Phụ trách các chức năng như:  -Lấy danh sách các môn học có thể đăng ký cho từng sinh viên  -Lấy thông tin(các lớp học phần) về từng môn học | Registration event (API) |
| 2 | client/api/common/schedule-api.js | Phụ trách các chức năng như:  -Đăng ký lớp học phần vào thời khóa biểu của từng sinh viên | Schedule(API) |
| 3 | client/react/routes/authen-routes/registration-route/registration-route.js | -Xây dựng giao diện đăng ký học chung hiển thị cho người dùng  -Các function kết nối, kiểm tra thời gian đợt đăng ký học, các môn học trong đợt đăng ký học, đăng ký lớp học phần, hiển thị thời khóa biểu đang đăng ký … | Registration route(Form DKH) |
| 4 | client/react/routes/authen-routes/registration-route/registration-detail.js | Giao diện danh sách các lớp học phần khi nhấn vào từng môn |
| 5 | client/react/routes/authen-routes/registration-route/subject-registerable-list/subject-registerable-list.jsx | Giao diện danh sách các môn học có thể đăng ký trong đợt đăng ký học |
| 6 | Server/controllers/registration-event.js | Nơi tiếp nhận các URL của registration-api, xử lý với các function và gửi lại các data | Registration route (BDK DKH) |
| 7 | Server/controllers/shedule.js | Nơi tiếp nhận các URL của shedule-api, xử lý với các function và gửi lại các data |
| 8 | Server/db/db-controllers/ registration-event.js | Nơi chứa các function của registration event, thực hiện các chức năng như tạo đợt đăng ký học,… |
| 9 | Server/db/db-controllers/ shedule.js | Nơi chứa các function của shedule, thực hiện các chức năng như lấy thời khóa biểu sinh viên, đăng ký vào thời khóa biểu |  |

# Thiết kế, phân tích và cài đặt chức năng trên di động

## Đặc tả chức năng

### UC#1 - Đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Đăng nhập** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép người dùng đăng nhập vào hệ thống để sử dụng các chức năng của hệ thống. | |
| **Tác nhân** | | giáo viên, sinh viên | |
| **Tiền điều kiện** | | Có tài khoản trong hệ thống | |
| **Hậu điều kiện** | **Thành công** | Đăng nhập thành công vào hệ thống | |
| **Lỗi** | Đăng nhập không thành công | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi người dùng muốn đăng nhập để sử dụng các chức năng của hệ thống   * Usecase này bắt đầu khi tác nhân muốn đăng nhập vào hệ thống. * Hệ thống yêu cầu tác nhân nhập tài khoản và mật khẩu đăng nhập. * Tác nhân nhập tài khoản và mật khẩu đăng nhập của mình. * Hệ thống xác nhận tài khoản và mật khẩu đăng nhập có hợp lệ không?   + Nếu không hợp lệ thì thực hiện luồng A.   + Nếu hợp lệ thì điều hướng người dùng vào trang thông tin cá nhân * Hệ thống ghi lại quá trình đăng nhập vào local storage. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Luồng A: Nhập sai tài khoản/ mật khẩu đăng nhập** | | | |
| * Hệ thống hiển thị một thông báo lỗi * Người sử dụng có thể chọn hoặc là đăng nhập lại hoặc là hủy bỏ đăng nhập, khi đó usecase này sẽ kết thúc. | | | |
| **Giao diện minh họa** | | | |
| Hình .3 Giao diện đăng nhập | | | |
| **THIẾT KẾ UML** | | | |
| **Sơ đồ lớp phân tích** | | | |
|  | | | |
| **Sơ đồ trình tự** | | | |
|  | | | |

### UC#02 - Xem thông tin cá nhân

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#02** | | **Xem thông tin cá nhân** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép người dùng xem các thông tin cá nhân của mình | |
| **Tác nhân** | | Giáo viên, sinh viên | |
| **Tiền điều kiện** | | Đã phải đăng nhập | |
| **Hậu điều kiện** | **Thành công** | Hệ thống hiển thị thông tin của người dùng | |
| **Lỗi** | Hệ thống không hiển thị thông tin của người dùng | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi người dùng muốn xem các thông tin cá nhân   * Use case bắt đầu khi sinh viên click vào chức năng “thông tin cá nhân” ở thanh sidebar * Hệ thống hiển thị ra các thông tin gồm có: * Đối với tài khoản có vai trò là sinh viên:   + - * Tên đầy đủ       * Ngày sinh       * Số điện thoại       * E-mail       * Khóa       * Trình độ tiếng anh       * Chuyên ngành       * Số tín chỉ tích lũy * Đối với tài khoản có vai trò là giáo viên.   + - * Tên đầy đủ       * Ngày sinh       * Số điện thoại       * E-mail       * Chuyên ngành | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Giao diện minh họa** | | | |
| Hình .3 Giao diện thông tin cá nhân | | | |
| **THIẾT KẾ UML** | | | |
| **Sơ đồ lớp phân tích** | | | |
|  | | | |
| **Sơ đồ trình tự** | | | |
|  | | | |

### UC#03 - Xem bảng điểm

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#03** | | **Xem bảng điểm** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép người dùng xem các thông tin về bảng điểm của mình của mình | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | Đã phải đăng nhập | |
| **Hậu điều kiện** | **Thành công** | Hệ thống hiển thị thông tin của người dùng | |
| **Lỗi** | Hệ thống không hiển thị thông tin của người dùng | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi người dùng muốn xem các thông tin cá nhân   * Use case bắt đầu khi sinh viên click vào chức năng “bảng điểm” ở thanh sidebar * Hệ thống hiển thị ra các thông tin gồm có:   + - Họ tên     - Mã sinh viên     - Tổng số tín chỉ tích lũy     - Bảng điểm chi tiết từng môn học của sinh viên gồm có:     - Mã học phần     - Tên học phần     - Số tín chỉ     - Điểm | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Giao diện minh họa** | | | |
| Hình .3 Giao diện bảng điểm | | | |
| **THIẾT KẾ UML** | | | |
| **Sơ đồ lớp** | | | |
|  | | | |
| **Sơ đồ trình tự** | | | |
|  | | | |

### UC#04 - Xem lịch học(dạy)

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#04** | | **Xem lịch học(dạy)** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép người dùng xem lịch học(dạy) của mình | |
| **Tác nhân** | | giáo viên, sinh viên | |
| **Tiền điều kiện** | | Đã phải đăng nhập | |
| **Hậu điều kiện** | **Thành công** | Hệ thống hiển thị thông tin của người dùng | |
| **Lỗi** | Hệ thống không hiển thị thông tin của người dùng | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi người dùng muốn xem các thông tin cá nhân   * Use case bắt đầu khi sinh viên click vào chức năng “thời khóa biểu”(nếu là sinh viên) hoặc “lịch dạy”(nếu là giáo viên) ở thanh sidebar * Hệ thống hiển thị ra thời khóa biểu(lịch dạy) của kỳ học và năm học hiện tại của người dùng * Người dùng có thể chọn xem thời khóa biểu của kỳ, năm học khác của mình bằng cách nhấn vào droplist ở trên bảng điểm và chọn kỳ, năm học   + - Nếu có thông tin về thời khóa biểu, hệ thống sẽ hiển thị thời khỏa biểu đó     - Nếu không có thực hiện luồng A | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Luồng A: Khi không có thông tin về lịch học(dạy) của người dùng** | | | |
| Hệ thống sẽ thông báo “Không có dữ liệu thời khóa biểu của kỳ học … và năm học …” | | | |
| **Giao diện minh họa** | | | |
| Hình .3 Giao diện lịch học(dạy) | | | |
| THIẾT KẾ SƠ ĐỒ UML | | | |
| Sơ đồ lớp phân tích | | | |
|  | | | |
| Sơ đồ trình tự | | | |
|  | | | |

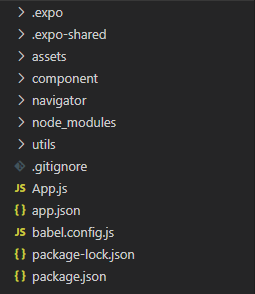
### UC#05 - Xem thời khóa biểu toàn trường

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#05** | | **Xem thời khóa biểu toàn trường** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép người dùng xem thời khóa biểu toàn trường | |
| **Tác nhân** | | giáo viên, sinh viên | |
| **Tiền điều kiện** | | Đã phải đăng nhập | |
| **Hậu điều kiện** | **Thành công** | Hệ thống hiển thị thông tin của người dùng | |
| **Lỗi** | Hệ thống không hiển thị thông tin của người dùng | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi người dùng muốn xem các thông tin cá nhân   * Use case bắt đầu khi sinh viên click vào chức năng “thời khóa biểu toàn trường” ở thanh sidebar * Hệ thống hiển thị ra thời khóa biểu toàn trường của kỳ học và năm học hiện tại * Người dùng có thể chọn xem thời khóa biểu toàn trường của kỳ, năm học và nhóm học khác của mình bằng cách nhấn vào các filter ở trên bảng điểm và chọn kỳ, năm học và nhóm học   + - Nếu có thông tin về thời khóa biểu, hệ thống sẽ hiển thị thời khỏa biểu đó     - Nếu không có thực hiện luồng A | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Luồng A: Khi không có thông tin về thời khóa biểu toàn trường** | | | |
| Hệ thống sẽ thông báo “Không có dữ liệu thời khóa toàn trường của kỳ học … nhóm … năm học …” | | | |
| **Giao diện minh họa** | | | |
| Hình 7.5 Giao diện thời khóa biểu toàn trường | | | |
| THIẾT KẾ UML | | | |
| Sơ đồ lớp phân tích | | | |
|  | | | |
| Sơ đồ trình tự | | | |
|  | | | |

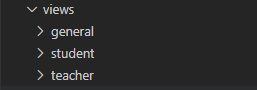
### UC#06 - Đăng xuất

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#06** | | **Đăng xuất** | **Độ phức tạp: trung bình** |
| **Mô tả** | | Cho phép người dùng đăng xuất ra khỏi hệ thống | |
| **Tác nhân** | | giáo viên, sinh viên | |
| **Tiền điều kiện** | | Đã phải đăng nhập | |
| **Hậu điều kiện** | **Thành công** | Hệ thống đăng xuất | |
| **Lỗi** | Hệ thống không đăng xuất | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi người dùng muốn xem các thông tin cá nhân   * Use case bắt đầu khi sinh viên click vào chức năng “Đăng xuất” ở thanh sidebar * Hệ thống sẽ xóa toàn bộ thông tin của người dùng trong local storage và điều hướng người dùng tới màn hình đăng nhập | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Giao diện minh họa** | | | |
| Hình 7.6 Giao diện đăng xuất | | | |
| **THIẾT KẾ UML** | | | |
| **Sơ đồ lớp phân tích** | | | |
|  | | | |
| **Sơ đồ trình tự** | | | |
|  | | | |

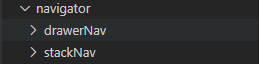
## Cấu trúc thư mục mã nguồn



* Asset: các tài nguyên cho giao diện như logo, splash gif…
* Component: nơi chứa các component để render giao diện ra màn hình & các điều hướng
  + Drawers: nơi chứa giao diện của drawer(sidebar để điều hướng giữa các component)
  + Views: nơi chứa các component giao diện



* + - * General: chứa các component dùng chung
      * Student: chứa các component dành riêng cho học sinh
      * Teacher: chứa các component dành riêng cho giáo viên
* Node\_modules: nơi chứa các thư viện dùng để xây dựng chương trình
* Utils: một số chức năng sử dụng để xây dựng chương trình
* Navigator: nơi chứa các điều hướng



* + drawerNav: là nơi quản lý điều hướng của sidebar
  + stackNav: là nơi quản lý các điều hướng đến các component khác

# KIỂM THỬ

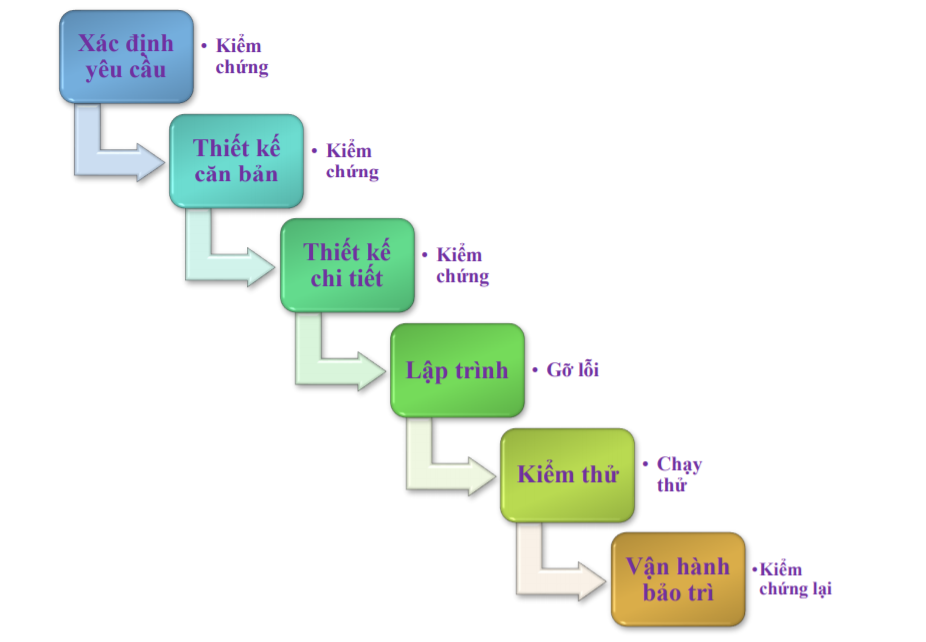
## Giới thiệu chung

### Tổng quan về phần mềm và quy trình phát triển phần mềm

#### Tổng quan về phần mềm

* Phần mềm là những chương trình, ứng dụng, website được viết, cài đặt và thực thi trên môi trường điện toán(computing) như: máy tính(computer), điện thoại di động(mobilephone),...
* Phần mềm trong kiểm thử phần mềm còn mở rộng ra bao gồm các tài liệu (document), dữ liệu(data) phù hợp và liên quan đến hoạt động của phần mềm.

#### Quy trình phát triển phần mềm



### Kiểm thử phần mềm

* Kiểm thử phần mềm(software testing) là hoạt động nhằm tìm kiếm, phát hiện các lỗi của phần mềm.
* Kiểm thử phần mềm là một tiến trình hay một tập hợp các tiến trình được thiết kế để đảm bảo chương trình máy tính thực hiện theo đúng yêu cầu của khách hàng.
* Kiểm thử phần mềm là quá trình thực thi một chương trình với mục đích tìm lỗi (theo The Art of Software Testing)

### Tại sao phải kiểm thử

* Kiểm thử phần mềm để tìm ra lỗi và thỏa mãn một số câu hỏi sau :
* Phần mềm có thực sự làm việc như mong muốn?
* Phần mềm có giải quyết được yêu cầu của người sử dụng?
* Người dùng có thích nó không?
* Phần mềm có tương thích với hệ thống khác của chúng ta không?
* Phần mềm thực thi như thế nào? Nó có sẵn sàng để phát hành?
* ….
* Một số ví dụ của việc sai lầm do phần mềm
* China Airline Airbus A300 bị rơi năm 1994 làm chết 264 người do lỗi của phần mềm
* Năm 1985, máy xạ trị của Canada do lỗi của phần mềm làm chết 3 bệnh nhân.
* Năm 1999, một lỗi phần mềm làm thất bại việc phóng vệ tinh quân sự tiêu tốn 1.2 tỷ USD
* Do đó, kiểm thử là quan trọng bởi vì các lỗi của phần mềm có chi phí sửa đắt đỏ hoặc thậm chí gây nguy hiểm.

### Ý nghĩa của kiểm thử

* Kiểm thử không chỉ cần tìm lỗi phần mềm, mà còn là quá trình kiểm tra và xác minh một phần mềm đã đáp ứng yêu cầu và mong đợi của khách hàng hay chưa.
* Chi phí kiểm thử chiếm : 40% tổng sức phát triển, ≥ 30% tổng thời gian phát triển. Với các phần mềm quan trọng chi phí có thể gấp từ 3 đến 5 lần tổng các chi phí khác cộng lại. Do đó, kiểm thử tốt sẽ :
* Giảm chi phí điều chỉnh và thời gian phát triển
* Tăng độ tin cậy của phần mềm

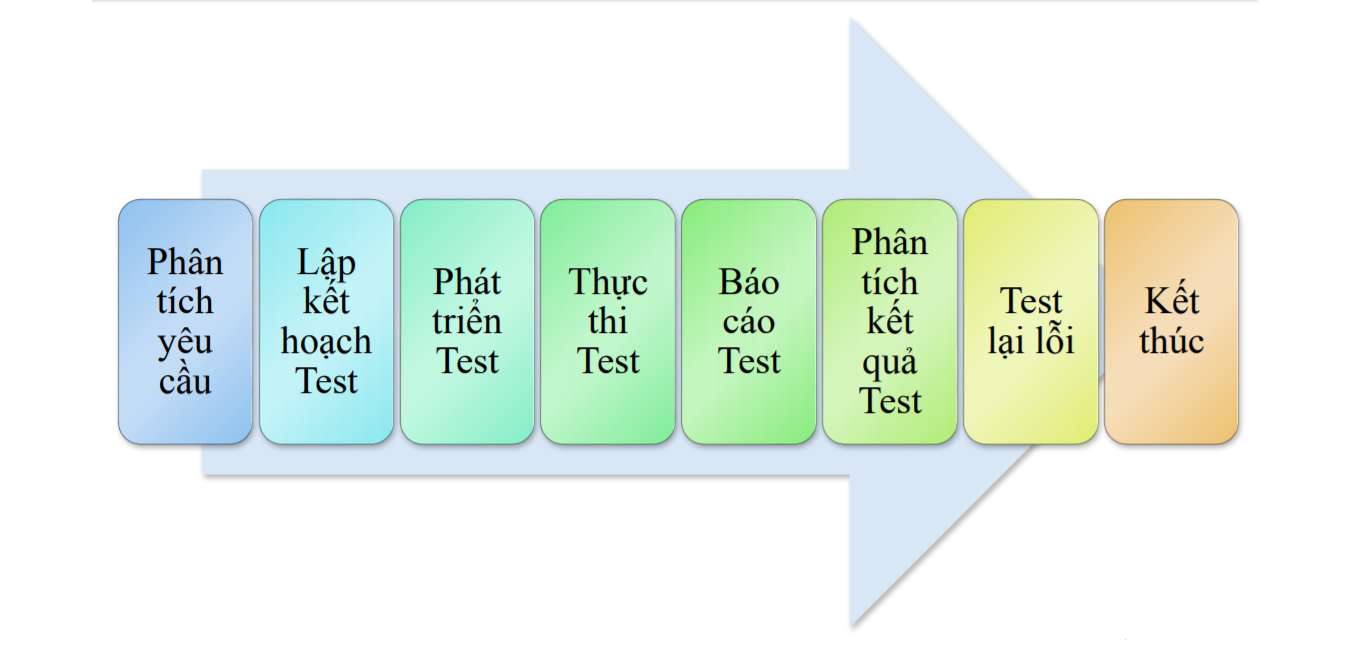
### Mục tiêu của kiểm thử phần mềm

* Phát hiện càng nhiều lỗi càng tốt trong thời gian xác định trước.
* Làm thế nào để nghĩ ra cách ngăn chặn lỗi phát sinh
* Chứng minh rằng sản phầm phần mềm phù hợp với đặc tả của nó.
* Chứng tỏ rằng các tester và các báo cáo tạo ra đúng và hữu dụng.

### Nguyên tắc kiểm thử

* Giống như nguyên tắc sử dụng thuốc trừ sâu, khi chúng ta sử dụng một loại thuốc trừ sâu mãi thì sẽ bị nhờn thuốc nên phải thay đổi loại thuốc khác. Trong kiểm thử phần mềm, khi dùng đi dùng lại một kịch bản kiểm thử thì sẽ đến lúc không thể tìm ra lỗi mới nữa. Chính vì vậy các bộ kịch bản kiểm thử phải được thường xuyên xem xét và cập nhật, phù hợp với từng thành phần khác nhau của phần mềm, mang lại khả năng tìm được lỗi lớn nhất.
* Nguyên tắc :
* Kiểm thử để chỉ ra sự hiện diện của lỗi
* Kiểm thử toàn diện, toàn bộ là không thể.
* Kiểm thử càng sớm càng tốt.
* Phân nhóm lỗi để xác định một số module tập trung lỗi nhiều nhất.
* Nghịch lý thuốc trừ sâu
* Kiểm thử được thực hiện khác nhau trong những bối cảnh khác nhau
* Suy nghĩ “Không có lỗi” là một sai lầm..

### Quy trình kiểm thử



* Phân tích yêu cầu : Xác định phạm vi test
* Lập kế hoạch test : Chiến lược test (test strategy), test plan.
* Phát triển test : Viết test procedure, test scenario, test case, test data và test script.
* Thực thi test : Tester thực thi phần mềm dựa trên test plan và test case.
* Báo cáo test : Tester điền kết quả test vào test case và tạo báo cáo kết quả test.
* Phân tích kết quả test : Phân tích lỗi để quyết định lỗi nào sẽ được sửa và lỗi nào không sửa.
* Test lại lỗi : Sau khi một lỗi được DEV sửa xong , chuyển phần mềm cho tester test lại.
* Kết thúc test : Khi test đã đáp ứng được điều kiện dừng. Từ đó rút ra các bài học kinh nghiệm.

## Kiểm thử cho từng module :

### Kiểm thử UC Đăng ký học :

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Mục tiêu kiểm thử | Mô tả thao tác kiểm thử | Kết quả chờ đợi | Kết quả thực tế | Dữ liệu kiểm thử | Kết quả (P/F) |
| TC\_01.1 : Kiểm thử giao diện chung | | | | | | |
| 1 | Hiển thị màn hình khi không trong thời gian đăng ký học | Tại màn hình hiển thị giao diện đăng ký học, bằng trực quan kiểm tra hiển thị trên màn hình từ trên xuống dưới, từ trái sang phải :  - Alert  màu đỏ :“**Thông báo** : Bạn chưa thuộc đối tượng đăng ký học kì này”  - Icon và dòng text “Học phí tạm tính : 0 VNĐ”  - Text : Thời khóa biểu tạm thời.  - Thời khóa biểu gồm các tiêu đề : ngày( từ thứ 2 đến chủ nhật theo hàng ngang) và ca (từ 1 đến 10 theo hàng dọc)  - Text : Tra cứu thời gian ca học.  - Table : Tra cứu thời gian ca học gồm các ca học :   * ca 1 :7h05 - 8h * ca 2 :8h05-9h00 * ca 3 :9h15-10h10 * ca 4 :10h15-11h10 * ca 5 :11h15-12h10 * ca 6 :13h00-13h55 * ca 7 :14h00-14h55 * ca 8 :15h10-16h05 * ca 9 :16h1017h05 * ca 10:17h10-18h05 * Text : “Đăng ký học không khả dụng lúc này” màu trắng đè lên thời khóa biểu, text :Tra cứu thời gian cac học và table tra cứu thời gian ca học. | Tại màn hình hiển thị giao diện đăng ký học, hiển thị trên màn hình từ trên xuống dưới, từ trái sang phải :  - Alert  màu đỏ :“**Thông báo** : Bạn chưa thuộc đối tượng đăng ký học kì này”  - Icon và dòng text “Học phí tạm tính : 0 VNĐ”  - Text : Thời khóa biểu tạm thời.  - Thời khóa biểu gồm ngày( từ thứ 2 đến chủ nhật theo hàng ngang) và ca (từ 1 đến 10 theo hàng dọc)  - Text : Tra cứu thời gian ca học.  - Table : Tra cứu thời gian ca học gồm các ca học :   * ca 1 :7h05 - 8h * ca 2 :8h05-9h00 * ca 3 :9h15-10h10 * ca 4 :10h15-11h10 * ca 5 :11h15-12h10 * ca 6 :13h00-13h55 * ca 7 :14h00-14h55 * ca 8 :15h10-16h05 * ca 9 :16h1017h05 * ca 10:17h10-18h05   Text : “Đăng ký học không khả dụng lúc này” màu trắng đè lên thời khóa biểu, text: Tra cứu thời gian ca học và table tra cứu thời gian ca học. | Giao diện màn hình hiển thị bằng trực quan kiểm tra giống với kết quả chờ đợi. |  | P |
| 2 | Hiển thị màn hình khi trong thời gian đăng ký học | Tại màn hình hiển thị giao diện đăng ký học, bằng trực quan kiểm tra hiển thị trên màn hình từ trên xuống dưới, từ trái sang phải :  - Text : Danh sách môn đăng ký  - List : các học phần sinh viên có thể đăng ký học được lọc bằng thuật toán.  - Icon và dòng text “Học phí tạm tính : 0 VNĐ”.  - Text : Thời khóa biểu tạm thời.  - Thời khóa biểu gồm các tiêu đề : ngày( từ thứ 2 đến chủ nhật theo hàng ngang) và ca (từ 1 đến 10 theo hàng dọc)  - Text : Tra cứu thời gian ca học.  - Table : Tra cứu thời gian ca học gồm các ca học :   * ca 1 :7h05 - 8h * ca 2 :8h05-9h00 * ca 3 :9h15-10h10 * ca 4 :10h15-11h10 * ca 5 :11h15-12h10 * ca 6 :13h00-13h55 * ca 7 :14h00-14h55 * ca 8 :15h10-16h05 * ca 9 :16h1017h05 * ca 10:17h10-18h05 | Tại màn hình hiển thị giao diện đăng ký học, hiển thị trên màn hình từ trên xuống dưới, từ trái sang phải:  - Text : Danh sách môn đăng ký  - List : các học phần sinh viên có thể đăng ký học được lọc bằng thuật toán.  - Icon và dòng text “Học phí tạm tính : 0 VNĐ”.  - Text : Thời khóa biểu tạm thời.  - Thời khóa biểu gồm các tiêu đề : ngày( từ thứ 2 đến chủ nhật theo hàng ngang) và ca (từ 1 đến 10 theo hàng dọc)  - Text : Tra cứu thời gian ca học.  - Table : Tra cứu thời gian ca học gồm các ca học :   * ca 1 :7h05 - 8h * ca 2 :8h05-9h00 * ca 3 :9h15-10h10 * ca 4 :10h15-11h10 * ca 5 :11h15-12h10 * ca 6 :13h00-13h55 * ca 7 :14h00-14h55 * ca 8 :15h10-16h05 * ca 9 :16h1017h05 * ca 10:17h10-18h05 | Giao diện hiển thị giống như kết quả mong đợi. |  | P |
| TC\_01.2 : Kiểm thử chức năng đăng ký học | | | | | | |
| 1 | Kiểm tra khi click vào học phần bất kì trong list các học phần | Tại giao diện đăng ký học khi sinh viên đang trong thời gian đăng ký học. Di chuyển con trỏ chuột đến vị trí một học phần bất kì và click chuột vào tên học phần đó. | Tại giao diện đăng ký học, hiển thị thêm list các nhóm lớp học phần thuộc lớp học phần đã chọn bên dưới list các học phần và trên text “Học phí tạm tính” | Tại giao diện đăng ký học, hiển thị thêm danh sách các nhóm lớp học phần thuộc học phần đã chọn |  | P |
| 2 | Kiểm tra khi click vào 1 nhóm lớp học phần | Tại giao diện đăng ký học, sau khi chọn 1 học phần, click chuột vào 1 nhóm lớp học phần | Tại giao diện đăng ký học, nhóm lớp học phần đã chọn được thêm vào thời khóa biểu tạm thời và học phí tạm thời thay đổi dựa theo giá tiền học phần | Nhóm lớp học phần đã chọn được thêm vào thời khóa biểu tạm thời và học phí tạm thời tăng lên. |  | P |
| 3 | Kiểm tra khi hover vào tên một nhóm lớp học phần đã đăng ký. | Tại giao diện đăng ký học, sau khi đã có lớp học phần trong thời khóa biểu tạm thời, di chuyển con trỏ chuột hover lên lớp học phần trong thời khóa biểu. | Hiển thị dòng text : “Click để hủy môn” | Không hiển thị gì cả |  | F |
| 4 | Kiểm tra khi click vào lớp học phần đã đăng ký trong thời khóa biểu tạm thời | Tại giao diện đăng ký học, sau khi đã đăng ký lớp học phần, di chuyển con trỏ chuột đến lớp học phần đang hiển thị trong thời khóa biểu tạm thời và click vào. | Lớp học phần được xóa khỏi thời khóa biểu tạm thời và học phí tạm thời giảm xuống kèm theo thông báo : “Đã hủy lớp học phần” . | Lớp học phần được xóa khỏi thời khóa biểu tạm thời và học phí giảm xuống nhưng không có thông báo kèm theo |  | F |
| 5 | Kiểm tra khi click vào lớp học phần trùng thời gian ca học với lớp học phần đã đăng ký | Tại giao diện đăng ký học, sau khi đã đăng ký lớp học phần, chọn một lớp học phần có thời gian ca học trùng với một lớp học phần đã đăng ký. | Hiển thị pop-up Lỗi đăng ký “**Thông báo** :Bạn không thể đăng ký lớp ABC do trùng vào Thứ…Ca… của lớp XYZ” màu đỏ và không thêm vào thời khóa biểu tạm thời | Hiển thị pop-up Lỗi đăng ký “Thông báo : Bạn không thể đăng ký lớp ABC do trùng vào Thứ…Ca… của lớp XYZ” màu đỏ và không thêm vào thời khóa biểu tạm thời |  | P |
|  | | Tên TC | Kiểm tra giao diện chung và chức năng Đăng ký học | Tổng số TC Pass | 5 |  |
| Mã TC | TC\_01 | Tổng số TC fail | 2 |
| Người tạo TC | Hoàng Thanh Sơn | Tổng số TC Not-Tested | 0 |
| Ngày tạo TC | 11/3/2020 | Tổng số TC thực hiện | 7 |

**Nhận xét** : Chức năng đăng ký học là chức năng quan trọng và có độ phức tạp nhất của hệ thống. Nhóm đã cố gắng để hoàn thiện chức năng này và đã khắc phục được các lỗi lớn trong quá trình làm. Những TC fail ở trên đa phần là về thông báo hướng dẫn thao tác không quá quan trọng. Còn lại chức năng đã đáp ứng được yêu cầu của phần mềm.

### Kiểm thử UC Ép cứng lớp học phần

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Mục tiêu kiểm thử | Mô tả thao tác kiểm thử | Kết quả chờ đợi | Kết quả thực tế | Dữ liệu kiểm thử | Kết quả (P/F) |
| TC\_02.1 : Kiểm thử giao diện chung khi đã chọn sinh viên ép cứng | | | | | | |
| 1 | Hiển thị màn hình sau khi chọn sinh viên ép cứng | Tại màn hình hiển thị giao diện chức năng ép cứng sau khi chọn sinh viên, bằng trực quan kiểm tra hiển thị trên màn hình từ trên xuống dưới, từ trái sang phải :  - Text : “Bước 1 : Chọn sinh viên ép cứng”  - Multiple select “chọn sinh viên ép cứng”  - Text : “Bước 2: Ép cứng cho sinh viên”  - Text “Sinh viên đang chọn”  - Select “Sinh viên đang chọn”  - Text : Danh sách môn đăng ký  - List : các học phần có thể ép cứng cho sinh viên  - Icon và dòng text “Học phí tạm tính : VND”  - Thời khóa biểu gồm các tiêu đề : ngày( từ thứ 2 đến chủ nhật theo hàng ngang) và ca (từ 1 đến 10 theo hàng dọc)  - Table : Tra cứu thời gian ca học gồm các ca học :   * ca 1 :7h05 - 8h * ca 2 :8h05-9h00 * ca 3 :9h15-10h10 * ca 4 :10h15-11h10 * ca 5 :11h15-12h10 * ca 6 :13h00-13h55 * ca 7 :14h00-14h55 * ca 8 :15h10-16h05 * ca 9 :16h1017h05 * ca 10:17h10-18h05 | Tại màn hình hiển thị giao diện chức năng ép cứng sau khi chọn sinh viên, hiển thị trên màn hình từ trên xuống dưới, từ trái sang phải :  - Text : “Bước 1 : Chọn sinh viên ép cứng”  - Multiple select “chọn sinh viên ép cứng”  - Text : “Bước 2: Ép cứng cho sinh viên”  - Text “Sinh viên đang chọn”  - Select : “Sinh viên đang chọn”  - Text : Danh sách môn đăng ký  - List : các học phần có thể ép cứng cho sinh viên  - Icon và dòng text “Học phí tạm tính : VND”  - Thời khóa biểu gồm các tiêu đề : ngày( từ thứ 2 đến chủ nhật theo hàng ngang) và ca (từ 1 đến 10 theo hàng dọc)  - Table : Tra cứu thời gian ca học gồm các ca học :   * ca 1 :7h05 - 8h * ca 2 :8h05-9h00 * ca 3 :9h15-10h10 * ca 4 :10h15-11h10 * ca 5 :11h15-12h10 * ca 6 :13h00-13h55 * ca 7 :14h00-14h55 * ca 8 :15h10-16h05 * ca 9 :16h1017h05   ca 10:17h10-18h05 | Giao diện màn hình hiển thị bằng trực quan kiểm tra giống với kết quả chờ đợi. |  | P |
| TC\_02.2 : Kiểm thử chức năng ép cứng | | | | | | |
| 1 | Kiểm tra khi click vào học phần bất kì trong list các học phần | Tại giao diện đăng ký học khi sinh viên đang trong thời gian đăng ký học. Di chuyển con trỏ chuột đến vị trí một học phần bất kì và click chuột vào tên học phần đó. | Tại giao diện đăng ký học, hiển thị thêm list các nhóm lớp học phần thuộc lớp học phần đã chọn bên dưới list các học phần và trên text “Học phí tạm tính” | Tại giao diện đăng ký học, hiển thị thêm danh sách các nhóm lớp học phần thuộc học phần đã chọn |  | P |
| 2 | Kiểm tra khi click vào 1 nhóm lớp học phần | Tại giao diện đăng ký học, sau khi chọn 1 học phần, click chuột vào 1 nhóm lớp học phần | Tại giao diện đăng ký học, nhóm lớp học phần đã chọn được thêm vào thời khóa biểu tạm thời và học phí tạm thời thay đổi dựa theo giá tiền học phần | Nhóm lớp học phần đã chọn được thêm vào thời khóa biểu tạm thời và học phí tạm thời tăng lên. |  | P |
| 3 | Kiểm tra khi hover vào tên một nhóm lớp học phần đã đăng ký. | Tại giao diện đăng ký học, sau khi đã có lớp học phần trong thời khóa biểu tạm thời, di chuyển con trỏ chuột hover lên lớp học phần trong thời khóa biểu. | Hiển thị dòng text : “Click để hủy môn” | Không hiển thị gì cả |  | F |
| 4 | Kiểm tra khi click vào lớp học phần đã đăng ký trong thời khóa biểu tạm thời | Tại giao diện đăng ký học, sau khi đã đăng ký lớp học phần, di chuyển con trỏ chuột đến lớp học phần đang hiển thị trong thời khóa biểu tạm thời và click vào. | Lớp học phần được xóa khỏi thời khóa biểu tạm thời và học phí tạm thời giảm xuống kèm theo thông báo : “Đã hủy lớp học phần” . | Lớp học phần được xóa khỏi thời khóa biểu tạm thời và học phí giảm xuống nhưng không có thông báo kèm theo |  | F |
| 5 | Kiểm tra khi click vào lớp học phần trùng thời gian ca học với lớp học phần đã đăng ký | Tại giao diện đăng ký học, sau khi đã đăng ký lớp học phần, chọn một lớp học phần có thời gian ca học trùng với một lớp học phần đã đăng ký. | Hiển thị pop-up Lỗi đăng ký “**Thông báo** :Bạn không thể đăng ký lớp ABC do trùng vào Thứ…Ca… của lớp XYZ” màu đỏ và không thêm vào thời khóa biểu tạm thời | Hiển thị pop-up Lỗi đăng ký “Thông báo : Bạn không thể đăng ký lớp ABC do trùng vào Thứ…Ca… của lớp XYZ” màu đỏ và không thêm vào thời khóa biểu tạm thời |  | P |
|  | | Tên TC | Kiểm tra giao diện chung và chức năng Ép cứng | Tổng số TC Pass | 4 |  |
| Mã TC | TC\_02 | Tổng số TC fail | 2 |
| Người tạo TC | Hoàng Thanh Sơn | Tổng số TC Not-Tested | 0 |
| Ngày tạo TC | 11/3/2020 | Tổng số TC thực hiện | 6 |

### Kiểm thử UC quản lý tài khoản

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Mục tiêu kiểm thử | | Mô tả thao tác kiểm thử | Kết quả chờ đợi | | Kết quả thực tế | Dữ liệu kiểm thử | Kết quả (P/F) | |
| TC\_03.1 : Kiểm thử giao diện chức năng quản lý tài khoản | | | | | | | | | |
| 1 | Hiển thị màn hình ở trạng thái khởi tạo | | Tại màn hình hiển thị chức năng quản lý tài khoản, bằng trực quan kiểm tra hiển thị trên màn hình từ trên xuống dưới, từ trái sang phải :  - Textbox : “Tìm theo mã định danh, email, họ tên, số điện thoại.”  - Button “Tìm kiếm”  - Text : “Loại tài khoản”  - Select : “ Loại tài khoản”  - Button “Tạo tài khoản”  - Table : “ Tài khoản” gồm các mục STT, Mã định danh, họ tên, Loại tài khoản. | | Tại màn hình hiển thị chức năng quản lý tài khoản, hiển thị từ trên xuống dưới, từ trái sang phải :  - Textbox : “Tìm theo mã định danh, email, họ tên, số điện thoại.”  - Button “Tìm kiếm”  - Text : “Loại tài khoản”  - Select : “ Loại tài khoản”  - Button “Tạo tài khoản”  - Table : “ Tài khoản” gồm STT, Mã định danh, họ tên, Loại tài khoản. | Button “tạo tài khoản” nằm ở trên textbox “tìm kiếm” thay vì nằm ở dưới select “Loại tài khoản |  | **F** | |
| 2 | Kiểm thử giao diện tạo tài khoản | Tại giao diện khởi tạo, click chuột vào button “Tạo tài khoản” và bằng trực quan kiểm tra hiển thị từ trên xuống dưới, ,trái qua phải :  - Text: “Thông tin tài khoản cơ bản” có gạch dưới.  - Text: “Tên đăng nhập”  - Textbox: “Nhập tên đăng nhập”.  - Text: “Mật khẩu”  - Textbox: “Nhập mật khẩu”  - Text: “Họ và tên”.  - Textbox: “Nhập họ và tên”.  - Text; “email”  - Textbox: “Nhập email”  - Text : “Số điện thoại”  - Textbox: “Nhập số điện thoại”  - Tex: “Loại tài khoản”  - Select: “Loại tài khoản”.  - Text: “Mã định danh( Nhân viên)”  - Textbox : “Nhập mã định danh”  - Text : “Ngày sinh”  - Datepicker : “Ngày sinh”  - Button : “Hủy bỏ”  - Button: “Tiếp theo” | | | Tại giao diện “Tạo tài khoản”, hiển thị từ trên xuống dưới, trái qua phải :  - Text: “Thông tin tài khoản cơ bản” có gạch dưới.  - Text: “Tên đăng nhập”  - Textbox: “Nhập tên đăng nhập”.  - Text: “Mật khẩu”  - Textbox: “Nhập mật khẩu”  - Text: “Họ và tên”.  - Textbox: “Nhập họ và tên”.  - Text; “email”  - Textbox: “Nhập email”  - Text : “Số điện thoại”  - Textbox: “Nhập số điện thoại”  - Tex: “Loại tài khoản”  - Select: “Loại tài khoản”.  - Text: “Mã định danh( Nhân viên)”  - Textbox : “Nhập mã định danh”  - Text : “Ngày sinh”  - Datepicker : “Ngày sinh”  - Button : “Hủy bỏ”  - Button: “Tiếp theo” | Kết quả thực tế giống với kết quả mong đợi, không có sai phạm nào. |  | | **P** |
| TC\_02.2 : Kiểm tra validate giao diện tạo tài khoản | | | | | | | | | |
| TC\_02.2.1 : Tên đăng nhập | | | | | | | | | |
| 1 | Kiểm tra bỏ trống dữ liệu | | Tại giao diện tạo tài khoản, bỏ trống trường tên đăng nhập và nhập đầy đủ dữ liệu vào các trường còn lại và click vào button “Tiếp theo”. | Hiển thị báo lỗi : “Tên đăng nhập không được bỏ trống” màu đỏ dưới textbox: “Nhập tên đăng nhập” và không thể click vào button “Tiếp theo”. | | Hiển thị thông báo lỗi : “Tên đăng nhập không được bỏ trống” màu đỏ dưới textbox :Nhập tên đăng nhập” và không thể click vào button : “Tiếp theo” | - Tên đăng nhập :  - Mật khẩu : 123456  - Họ và tên : Hoàng Thanh Sơn  - Email : [sonhoang.hl97@gmail.com](mailto:sonhoang.hl97@gmail.com)  - Số điện thoại: 0386809657  - Loại tài khoản : Sinh viên  - Mã định danh : Nhân viên  - Ngày sinh : 29/9/1997 | P | |
| 2 | Độ dài tối đa | | Nhập vào trường “Tên đăng nhập” 201 kí tự | Không cho nhập kí tự thứ 201 | | Vẫn cho phép nhập | Abcdfabcdf…. | F | |
| 3 | Nhập kí tự đặc biệt | | Nhập vào trường “tên đăng nhập” một số kí tự đặc biệt | Hiển thị thông báo không cho nhập kí tự đặc biệt và không cho phép nhập. | | Vẫn cho phép nhập và không hiển thị thông báo | @#T@#@Y\*&% | F | |
| TC\_02.2.2 : Mật khẩu | | | | | | | | | |
|  |  | |  |  | |  |  |  | |
| TC\_02.2.3 : Họ và tên | | | | | | | | | |
|  |  | |  |  | |  |  |  | |
| TC\_02.2.4 : Email | | | | | | | | | |
|  |  | |  |  | |  |  |  | |
| TC\_02.2.5 :Số điện thoại | | | | | | | |  | |
|  |  | |  |  | |  |  |  | |
| TC\_02.2.6 : Loại tài khoản | | | | | | | | | |
|  |  | |  |  | |  |  |  | |
| TC\_02.3 : Kiểm tra chức năng chính của quản lý tài khoản | | | | | | | | | |
| 1 | Tạo mới tài khoản | | Tại màn hình giao diện quản lý tài khoản click chuột vào button “Tạo tài khoản”, thực hiện thêm mới tài khoản theo từng trường dữ liệu. Sau đó click chọn button “tiếp theo” | Tài khoản mới được thêm vào table “Tài khoản” với các thông tin giống với dữ liệu đã nhập vào. | | Tài khoản với tên đăng nhập a28631, mật khẩu : 123456,…. được thêm vào table “Tài khoản” ở giao diện khởi tạo chức năng quản lý tài khoản. | - Tên đăng nhập : a28631  - Mật khẩu : 123456  - Họ và tên : Hoàng Thanh Sơn  - Email : [sonhoang.hl97@gmail.com](mailto:sonhoang.hl97@gmail.com)  - Số điện thoại: 0386809657  - Loại tài khoản : Sinh viên  - Mã định danh : Nhân viên  - Ngày sinh : 29/9/1997 | P | |
| 2 | Sửa tài khoản | | Tại giao diện khởi tạo chức năng Quản lý tài khoản, click chuột vào tài khoản muốn sửa, nhập lại dữ liệu vào các trường muốn sửa và click vào button “Cập nhật” | Cửa sổ giao diện sửa tài khoản được mở với các trường:  Tại giao diện “Tạo tài khoản”, hiển thị từ trên xuống dưới, trái qua phải :  - Text: “Thông tin tài khoản cơ bản” có gạch dưới.  - Text: “Tên đăng nhập”  - Textbox: “Nhập tên đăng nhập”.  - Text: “Mật khẩu”  - Textbox: “Nhập mật khẩu”  - Text: “Họ và tên”.  - Textbox: “Nhập họ và tên”.  - Text; “email”  - Textbox: “Nhập email”  - Text : “Số điện thoại”  - Textbox: “Nhập số điện thoại”  - Tex: “Loại tài khoản”  - Select: “Loại tài khoản”.  - Text: “Mã định danh( Nhân viên)”  - Textbox : “Nhập mã định danh”  - Text : “Ngày sinh”  - Datepicker : “Ngày sinh”  - Button: “hủy bỏ”  - Button : “Xóa”  - Button : “Cập nhật”  - Sau khi sửa thông tin và click vào button “Cập nhật”, thông tin tài khoản sẽ được cập nhật lại. | | Cửa sổ giao diện được mở với các trường như kết quả mong đợi và thông tin tài khoản được cập nhật lại, số điện thoại chuyển từ 0386809657 thành 0387862256 | - Tên đăng nhập : a28631  - Mật khẩu : 123456  - Họ và tên : Hoàng Thanh Sơn  - Email : [sonhoang.hl97@gmail.com](mailto:sonhoang.hl97@gmail.com)  - Số điện thoại: 0387862256  - Loại tài khoản : Sinh viên  - Mã định danh : Nhân viên  - Ngày sinh : 29/9/1997 | P | |
| 3 | Xóa tài khoản | | Tại giao diện khởi tạo chức năng quản lý học phần, click vào tài khoản muốn xóa. Click vào button “Xóa”. | Cửa sổ giao diện sửa tài khoản được mở với các trường:  Tại giao diện “Tạo tài khoản”, hiển thị từ trên xuống dưới, trái qua phải :  - Text: “Thông tin tài khoản cơ bản” có gạch dưới.  - Text: “Tên đăng nhập”  - Textbox: “Nhập tên đăng nhập”.  - Text: “Mật khẩu”  - Textbox: “Nhập mật khẩu”  - Text: “Họ và tên”.  - Textbox: “Nhập họ và tên”.  - Text; “email”  - Textbox: “Nhập email”  - Text : “Số điện thoại”  - Textbox: “Nhập số điện thoại”  - Tex: “Loại tài khoản”  - Select: “Loại tài khoản”.  - Text: “Mã định danh( Nhân viên)”  - Textbox : “Nhập mã định danh”  - Text : “Ngày sinh”  - Datepicker : “Ngày sinh”  - Button: “hủy bỏ”  - Button : “Xóa”  - Button : “Cập nhật”.  Tài khoản bị xóa khỏi table “tài khoản" sau khi click button “Xóa” | | Cửa sổ giao diện xóa tài khoản dược mở ra với các trường giống như kết quả mong đợi. Sau khi click vào button “Xóa” thì tài khoản bị xóa khỏi table “Tài khoản”. |  | P | |
|  |  | |  |  | |  |  |  | |
|  |  | |  |  | |  |  |  | |
|  |  | |  |  | |  |  |  | |
|  |  | |  |  | |  |  |  | |
| ` |  | |  |  | |  |  |  | |
|  | | | Tên TC | Kiểm tra giao diện chung và chức năng quản lý tài khoản | | Tổng số TC Pass | 5 |  | |
| Mã TC | TC\_03 | | Tổng số TC fail | 3 |
| Người tạo TC | Hoàng Thanh Sơn | | Tổng số TC Not-Tested | 0 |
| Ngày tạo TC | 11/3/2020 | | Tổng số TC thực hiện | 8 |

**Nhận xét** : Chức năng quản lý tài khoản của nhóm về cơ bản đã hoàn thiện, chỉ còn một số lỗi về nhập kí tự không quá ảnh hưởng đến quá trình sử dụng của người dùng.

**KẾT LUẬN**

Dự án của nhóm đã giới thiệu một cách tổng thể về quá trình lên kế hoạch, xây dựng và phát triển phần mềm đăng ký học trường Đại học Thăng Long. Sau quá trình phát triển khoảng 3 tháng, dưới sự hướng dẫn, giúp đỡ của cô và việc hoạt động nhóm tích cực, hiệu quả của các bạn trong nhóm, phần mềm đã có tương đối đầy đủ những tính năng cần thiết của 1 phần mềm đăng ký học cũng như đáp ứng được những yêu cầu nghiệp vụ đã đề ra.

Nhóm đã tích lũy được thêm nhiều kiến thức mới và bổ ích như:

* Tìm hiểu được các nghiệp vụ quản lý và xây dựng phần mềm đăng ký học trường Đại học Thăng Long
* Tăng thêm hiểu biết về các bước lập kế hoạch cũng như phát triển dự án theo nhóm nói chung và với từng thành viên nói chung
* Xây dựng và nâng cao hiểu biết về cách xây dựng phần mềm gồm cả website (với NodeJs, ReactJS, MongoDB, …) và mobile (với: React Native, Javascript, …)
* Cải thiện khả năng làm việc nhóm
* Sử dụng Github để quản lý source code trong quá trình phát triển phần mềm cũng như deploy lên heroku

Cùng với việc áp dụng những kiến thức đã học ở trường, sự hướng dẫn của cô,

tham khảo kinh nghiệm của các nhóm trước và những kiến thức mới đã học được, nhóm đã cố gắng xây dựng và phát triển phần mềm tốt nhất có thể. Tuy nhiên nhóm cũng tin rằng nếu có đầy đủ thời gian phần mềm có thể trở nên hoàn thiện hơn cũng như những dự án tiếp sau có thể giảm thiểu thời gian cũng như gia tăng chất lượng phần mềm.

# tài liệu tham khảo

* Các website tham khảo:
* <https://viblo.asia/p/tong-quan-ve-mongodb-EoDkQoxqGbV>
* <https://viblo.asia/p/mot-cai-nhin-tong-quan-nhat-ve-nodejs-Ljy5VeJ3lra>
* <http://free-php-editor.com/reactjs-la-gi/>
* <https://nordiccoder.com/blog/tong-quan-ve-react-native/>