BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC THĂNG LONG**



**CHUYÊN ĐỀ TỐT NGHIỆP**

**TÊN ĐỀ TÀI: XÂY DỰNG PHẦN MỀM ĐĂNG KÝ HỌC**

**GIÁO VIÊN HƯỚNG DẪN: SINH VIÊN THỰC HIỆN:**

**MAI THÚY NGA A29378 – ĐẶNG ANH TÚ**

**A27957 – ĐỖ LÊ PHI LONG**

**NGÀNH : KHOA HỌC MÁY TÍNH +**

**HỆ THỐNG THÔNG TIN**

*Hà Nội, 2020*

**Lời cảm ơn**

Bọn em chân thành cảm ơn các thầy cô giáo trong khoa Toán – Tin những người đã dạy dỗ, trang bị cho bọn em nhưng kiến thức cơ bản, cần thiết để bọn em thực hiện chuyên đề.

Đặc biệt bọn em xin cảm ơn sâu sắc tới cô Mai Thúy Nga, người đã hướng dẫn, chỉ bảo, động viên và trực tiếp giúp đỡ bọn em trong suốt quá trình thực hiện chuyên đề tốt nghiệp

Cuối cùng bọn em xin kính gửi lời chúc tới toàn thể các thầy cô trong khóa Toán- Tin nói riêng cũng như toàn thể các thầy cô giảng dậy tại trường Đại học Thăng Long luôn luôn khỏe mạnh, tràn đầy nhiệt huyết để tiếp tục giảng dạy, đào tạo và truyền đạt các kiến thức cho các thế hệ sinh viên tiếp theo

Hà Nội, 2020

Sinh viên thực hiện:

Đặng Anh Tú – Đỗ Lê Phi Long

**MỤC LỤC**

[CHƯƠNG 1. Tổng quan hệ thống 5](#_Toc501811301)

[1.1. Đặt vấn đề 5](#_Toc501811302)

[1.2. Các thành phần cơ bản của hệ thống 5](#_Toc501811303)

[1.2.1. Phân tích yêu cầu nghiệp vụ 5](#_Toc501811304)

[1.3. Sơ đồ tổng quan của hệ thống 6](#_Toc501811305)

[1.3.1. Dành cho …. 7](#_Toc501811306)

[1.3.2. Dành cho …. 7](#_Toc501811307)

[1.4. Ánh xạ yêu cầu nghiệp vụ và sơ đồ chức năng 7](#_Toc501811308)

[CHƯƠNG 2. Giới thiệu công nghệ sử dụng 9](#_Toc501811309)

[CHƯƠNG 3. Đặc tả hệ thống 10](#_Toc501811310)

[3.1. Phân hệ người sử dụng 10](#_Toc501811311)

[UC #0001 10](#_Toc501811312)

[CHƯƠNG 4. Thiết kế hệ thống 13](#_Toc501811313)

[4.1. Kiến trúc tổng thể 13](#_Toc501811314)

[4.2. Thiết kế cơ sở dữ liệu 13](#_Toc501811315)

[4.2.1. Chi tết bảng…. 13](#_Toc501811316)

[CHƯƠNG 5. Cài đặt hệ thống 14](#_Toc501811317)

[5.1. Môi trường cài đặt 14](#_Toc501811318)

[5.2. Cấu trúc thư mục 14](#_Toc501811319)

[5.2.1. Server 14](#_Toc501811320)

[5.2.2. Client 14](#_Toc501811321)

…

**Danh mục hình ảnh**

**DANH MỤC TỪ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Ý nghĩa** |
| 1. | CSDL | Cơ sở dữ liệu |
| 2. | UC | Use-case |
| 3. |  |  |

LỜI GIỚI THIỆU

Năm 2020 với sự bùng nổ mạnh mẽ về khoa học công nghệ, các ứng dụng tin học đang và vẫn tiếp tục áp dụng trong hầu hết trong các lĩnh vực đời sống. Việc ứng dụng tin học trong quản lý nhằm nâng cao chất lượng, hiệu suất công việc. Hiện nay Đại học Thăng Long đã ứng dụng rất nhiều công nghệ thông tin nhằm giải quyết các bài toán trong lĩnh vực quản lý tại trường. Ví dụ như hệ thống đăng ký học, quản lý thời khóa biểu,…. Vận dụng những kiến thức tại trường học, giáo viên đã dạy bọn em quyết định làm chuyên đề : **HỆ THỐNG ĐĂNG KÝ HỌC** để hiểu sâu hơn về nghiệp vụ, công nghệ của hệ thống đăng ký học đại học thăng long nói riêng và các trường đại học trên toàn quốc nói chung

Báo cáo này tổng kết lại quy trình phát triển phần mềm, nội dung báo cáo gồm:

**Chương 1: Tổng quan về phần mềm**

Giới thiệu bài toán hệ thống cần giải quyết và quy trình nghiệp vụ của hệ thống. Chương này đưa ra các sơ đồ chức năng, mô tả các tác nhân tham gia của hệ thống

**Chương 2: Kiến trúc tổng thể - công nghệ hệ thống sử dụng**

Chương này đưa ra kiến trúc tổng thể của hệ thống kèm theo mô tả và tất cả các công nghệ được áp dụng vào hệ thống

**Chương 3: Đặc tả chức năng**

Chương này đặc tả tất cả các chức năng và giao diện của hệ thống

**Chương 4: Thiết kế hệ thống**

Chương này đưa ra các sơ đồ kiến trúc CSDL, cấu trúc các bảng dữ liệu.

**Chương 5: Cài đặt hệ thống**

Chương này đưa ra môi trường cài đặt hệ thống, cấu trúc thư mục mã nguồn và kết quả

# Tổng quan hệ thống

## Mô tả Bài toán

Việc đăng ký học của sinh viên là một việc bắt buộc và rất quan trọng trước khi bắt đầu mỗi kỳ học, đó là quyền lợi và trách nhiệm của sinh viên trong quá trình học tập của mình. Và để áp dụng công nghệ thông tin vào quản lý giáo dục cho việc quản lý được nhanh chóng và chính xác nhất, bọn em xây dựng hệ thống phần mềm quản lý đăng ký học trường Đại học Thăng Long. Hệ thống cần đáp ứng:

Thứ nhất, hệ thống đảm bảo tính chính xác và nhanh chóng trong quy trình đăng ký học của sinh viên.

Thứ hai, giúp giảng viên hỗ trợ việc đăng ký học cho sinh viên được tốt nhất, và quản lý được sinh viên trong lớp của mình giảng dạy.

Thứ ba, công việc quản lý sinh viên, giảng viên, môn học, ngành học, lớp học,…dễ dàng cho các nhân viên phòng đào tạo.

Hệ thống được phát triển trên hai nền tảng web và mobile. Trên nền tảng web, người dùng có thể thực hiện được toàn bộ các chức năng của hệ thống. Còn trên nền tảng mobile, người dùng có thể đăng nhập, xem thời khóa biểu cá nhân, thời khóa biểu toàn trường và đăng ký học.

## Phân tích bài toán

### Chuẩn bị dữ liệu:

* Dữ liệu TKB: ca học, phòng học, kỳ học, môn học, dữ liệu về giảng viên, sinh viên và để tổng kết, chương trình học của ngành, thời khóa biểu toàn trường, dữ liệu các bộ môn
* Dữ liệu bảng điểm: danh sách bảng điểm sinh viên
* Dữ liệu chương trình học: danh sách các trương trình học của từng ngành trong trường

### Đăng ký học

* Thiết lập thời gian đăng ký học: Trước mỗi kỳ học mới bắt đầu, quản trị viên sẽ là người thiết lập thời gian đăng ký học cho sinh viên. Thời gian được đăng ký học trong khung giờ nhất định để không bị quá tải và công bằng cho mọi sinh viên.
* Đăng ký học: Sinh viên trong trường đều có mỗi tải khoản riêng để đăng ký lịch học phù hợp với bản thân. Sinh viên đủ điều kiện tiên quyết thì các môn học được hiển thị trong phần đăng ký học, sinh viên có thể đăng ký các môn học để không bị trùng lịch học với môn khác của học kỳ đó.
* Ép cứng: Chức năng dành cho giảng viên và quản trị viên để hỗ trợ sinh viên ép cứng môn học khi không đủ điều kiện tiên quyết.
* Theo dõi đăng ký học:
* Đối với Giảng viên, giảng viên có thể theo dõi được tình trạng lớp mình được phụ trách giảng dạy: xem danh sách sinh viên của lớp đó.
* Quản trị viên có thể theo dõi các lớp được mở trong học kỳ tới, xem danh sách sinh viên và giảng viên trong trường của từng lớp học, từng ngành học,.... Ngoài ra quản trị viên còn có thể hủy lớp nếu số lượng sinh viên không đạt đủ 20 sinh viên; hoặc gộp lớp nếu 2 lớp cùng giờ, cùng môn, ít sinh viên đăng ký học,...

## Yêu cầu nghiệp vụ

* BR#: Đăng nhập/ Đăng xuất
* BR#: Đăng nhập
* BR#: Đăng xuất
* BR: Đăng ký học
* BR: Thiết lập thời gian đăng ký học
* BR: Đăng ký học
* BR: Ép cứng
* BR: Theo dõi ĐKH
* BR: Kết quả đăng ký học
* BR: Xem danh sách giảng viên giảng dạy
* BR: Xem danh sách Sinh viên của lớp học
* BR: Xem danh sách lớp học
* BR: Xem thời khóa biểu cá nhân
* BR: Xem thời khóa biểu giảng dạy
* BR: Quản lý thông tin
* BR: Quản lý thông tin người dùng
* BR: Quản lý thông tin cá nhân
* BR: Xây dựng thời khóa biểu
* BR: Quản lý tài khoản
* BR: Phân quyền
* BR: Quản lý TKB trường
* BR: Quản lý TKB khoa
* BR: Quản lý phòng học
* BR: Quản lý môn học
* BR: Quản lý khối giờ học
* BR: Quản lý ngành học
* BR: Phân công giảng dạy

## Sơ đồ tổng quan của hệ thống

Hình 1: Sơ đồ Use Case tổng quan của hệ thống

### Các tác nhân tham gia hệ thống

* **Sinh viên**: là các sinh viên đang theo học trong trường. Mỗi sinh viên đều được cấp một tài khoản đăng nhập để thực hiện đăng ký học cũng như xem thời khóa biểu cá nhân hoặc toàn trường
* **Trưởng bộ môn:** là các giáo viên đang giảng dạy tại trường. Mỗi giáo viên cũng được cập một tài khoản đăng nhập để có thể thực hiện các nghiệp vụ như xem danh sách sinh viên lớp mình giảng dạy, xem lịch dạy,phân công giảng dậy,....giáo viên có thể hỗ trợ ép cứng thời khóa biểu cho sinh viên (khi sinh viên chưa đủ điều kiện đăng ký) trong phạm vi lớp học phần do giáo viên đó phụ trách
* **Giáo viên:** là các giáo viên đang giảng dậy tại trường. Mỗi giáo viên đều được cấp một tài khoản đăng nhập để thực hiện việc theo dõi tkb dạy học.
* **Phòng đào tạo:** là người chịu trách nhiệm cho việc quản lý người dùng, phân quyền, có thể hỗ trợ ép cứng cho bất kỳ sinh viên của ngành nào. Quản trị viên cũng có thể hủy lớp khi số lượng sinh viên của lớp không đủ……..

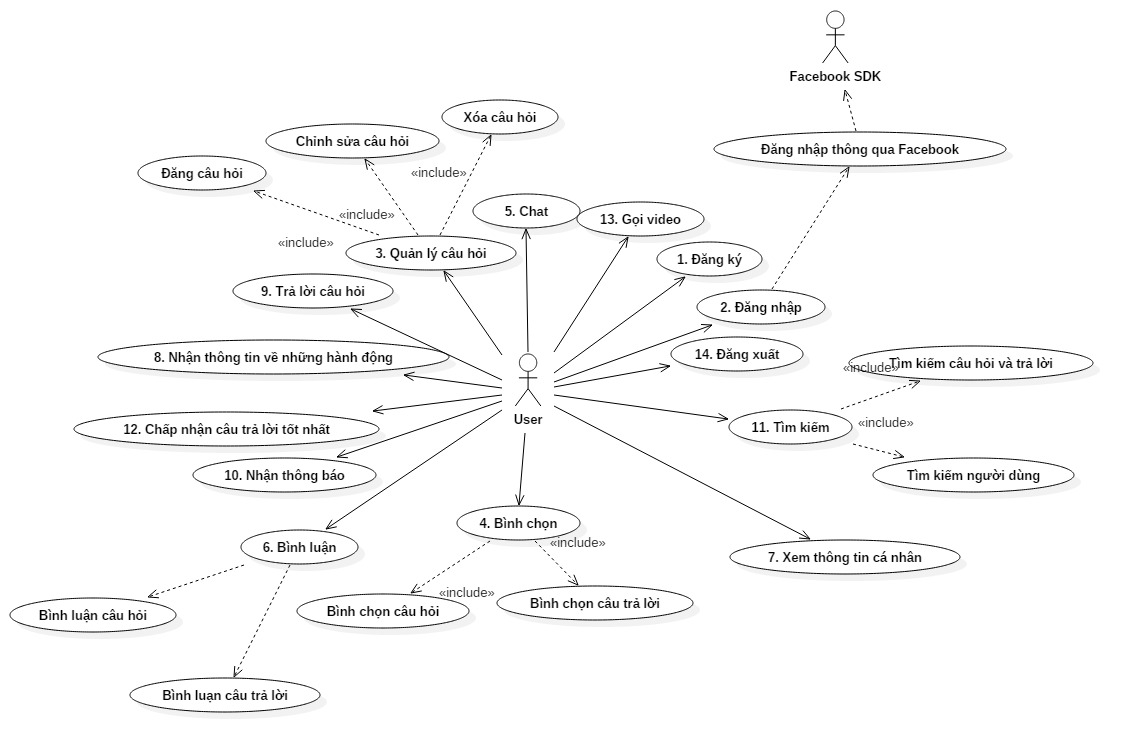
### Các chức năng trên website

* UC: Đăng nhập: Hệ thống cho phép các tác nhân đăng nhập vào hệ thống sau khi điền đầy đủ tên đăng nhập và mật khẩu. Sau khi đăng nhập vào hệ thống, người dùng có thể sử dụng các chức năng của hệ thông tùy vào quyền của tài khoản
* UC: Đăng xuất: Chức năng cho phép người dùng có thể đăng xuất ra khỏi hệ thống sau khi kết thúc phiên làm việc.
* UC: Quản lý CTH: Mỗi ngành học có một chương trình học khác nhau. Chức năng này cho phép Quản trị viên có thể quản lý được CTH có trong trường, các chức năng như import dữ liệu, chỉnh sửa CTH,...
* UC: Quản lý TKB toàn trường: Chức năng này cho phép quản trị viên có thể import dữ liệu thời khóa biểu toàn trường trước mỗi kỳ học vào database. Sau đó người dùng có thể xem được TKB toàn trường đó.
* UC: Quản lý lớp học:
* Xem danh sách lớp học: Quản trị viên có thể xem được toàn bộ danh sách các lớp học trong kỳ
* Xem danh sách Sinh viên của lớp học: Chức năng này cho phép quản trị viên hoặc giảng viên có thể xem được danh sach sinh viên của từng lớp học phần.
* UC: Quản lý thông tin người dùng: Quản trị viên có thể quản lý người dùng trong hệ thống: sinh viên và giảng viên. Quản trị viên có thể thêm quyền cho Giảng viên, sửa xóa thông tin sinh viên, giảng viên; xem danh sách giảng viên và danh sách sinh viên.
* UC: Quản lý thông tin cá nhân: Người dùng có thể xem, sửa, xóa, cập nhật thông tin cá nhân của mình.
* UC: Quản lý bảng điểm: Chức năng tổng hợp danh sách điểm những môn đã học theo từng sinh viên. Quản trị viên có thể xem chi tiết bảng điểm của từng sinh viên, sinh viên có thể xem được danh sách của mình.
* UC: Thống kê: Chức năng cho phép quản trị viên thống kê danh sách lớp mở trong kỳ, thống kê về số lượng sinh viên đăng ký trong kỳ, danh sách các lớp của các môn.
* UC: Quản lý Đăng ký học: Chức năng này cho phép quản trị viên quản lý quá trình đăng ký học của từng kỳ học: thiết lập thời gian đăng ký học, Ép cứng môn học cho sinh viên khi được yêu cầu.
* Thiết lập thời gian đăng ký học: Chức năng này cho phép quản trị viên thiết lập thời gian bắt đầu và kết thúc quá trình đăng ký học cho sinh viên, Quản trị viên cũng có thể mở thêm đợt đăng ký bổ sung sau thời gian đăng ký chính thức.
* Ép cứng: Khi sinh viên chưa đủ điều kiện tiên quyết mà phải học một môn nào đó hoặc không thể tự đăng ký được, thì quản trị viên có thể hỗ trợ sinh viên ép cứng thời khóa biểu, ngoài ra còn có giảng viên phụ trách của bộ môn đó có thể hỗ trợ ép cứng môn học cho sinh sinh viên.
* UC: Đăng ký học: Tác nhân Sinh viên sử dụng chức năng này để thêm, xóa môn học mà mình đủ điều kiện đăng ký trong học kỳ vào thời khóa biểu cả nhân, các môn học đăng ký không được trùng ca học với nhau.
* UC: Xem thời khóa biểu cá nhân: Sau khi đăng ký học thành công, Sinh viên có thể sử dụng chức năng này để kiểm tra các môn học đã đăng ký được và thời gian học của mình.
* UC: Xem thời khóa biểu giảng dạy: Mỗi lớp học được sắp xếp 1 giảng viên, nên giảng viên cần xem thời khóa biểu giảng dạy của mình để không bị nhầm lịch dạy.

### Các chức năng trên mobile

* UCM: Quản lý danh sách
* Xem thời khóa biểu toàn trường: người dùng đăng nhập vào hệ thống đều có thể sử dụng chức năng này.
* Xem Chương trình học: Chức năng này cho phép người dùng có thể xem được danh sách các chương trình học có trong hệ thống, bao gồm mã môn, tên môn, số tín chỉ và điều kiện tiên quyết.
* Xem thời khóa biểu cá nhân: Sau khi đăng ký học thành công, Sinh viên có thể sử dụng chức năng này để kiểm tra các môn học đã đăng ký được và thời gian học của mình.
* Xem Bảng điểm: Kết thúc một kỳ học, điểm của sinh viên được nhập vào hệ thống, hệ thống sẽ xử lý dữ liệu và
* Xem thời khóa biểu giảng dạy: Mỗi lớp học được sắp xếp 1 giảng viên, nên giảng viên cần xem thời khóa biểu giảng dạy của mình để không bị nhầm lịch dạy.
* UCM: Đăng ký học: Tác nhân Sinh viên sử dụng chức năng này để thêm, xóa môn học mà mình đủ điều kiện đăng ký trong học kỳ vào thời khóa biểu cả nhân, các môn học đăng ký không được trùng ca học với nhau.
* UCM: Đăng nhập: Hệ thống cho phép các tác nhân đăng nhập vào hệ thống sau khi điền đầy đủ tên đăng nhập và mật khẩu. Sau khi đăng nhập vào hệ thống, người dùng có thể sử dụng các chức năng của hệ thông tùy vào quyền của tài khoản
* UCM: Đăng xuất: Chức năng cho phép người dùng có thể đăng xuất ra khỏi hệ thống sau khi kết thúc phiên làm việc.
* UCM: Quản lý thông tin cá nhân
* Xem thông tin cá nhân: chức năng này cho phép toàn bộ các tác nhân có thể xem được thông tin cá nhân của mình.
* Cập nhật thông tin cá nhân: chức năng này cho phép toàn bộ các tác nhân có thể cập nhật được thông tin cá nhân của mình.

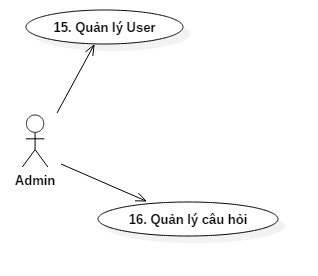
### Dành cho ….



Hình 2: Sơ đồ Uce Case dành cho…

* Các tác nhân tham gia vào hệ thống dành cho người sử dụng
* **User:** ….
* Các chức năng chính của hệ thống trong phân hệ người sử dụng
* **UC#001**

### Dành cho ….



Hình …: Sơ đồ Use Case dành cho …

## Ánh xạ yêu cầu nghiệp vụ và sơ đồ chức năng

….

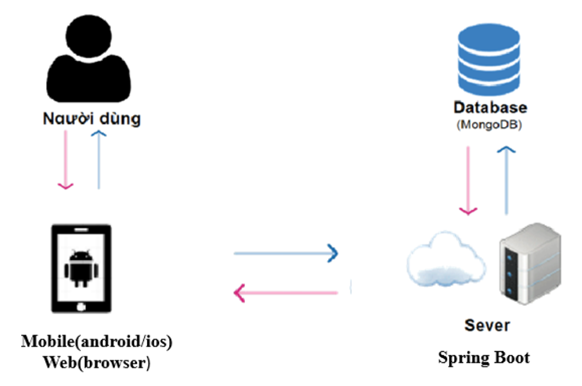
| **BR** | **Mô Tả** | **UC** |
| --- | --- | --- |
| **BR1: ……….** | | |
| BR 1.1 |  | UC#001 |
| BR 1.2 |  | UC#002 |
| BR 1.3 |  | UC#014 |
| **BR2: ………….** | | |
| BR 2.1 |  | UC#003 |
| BR 2.2 |  | UC#004 |
| BR 2.3 |  | UC#012 |

Bảng 1: Sơ đồ chức năng của hệ thống

# Giới thiệu công nghệ sử dụng

## Sơ đồ kiến trúc tổng thể

### Sơ đồ kiến trúc



* Kiến trúc tổng thể sẽ gồm có Người dùng, Ứng dụng, Server và Database.
* Người dùng sẽ thông qua các ứng dụng (ios, android, browser) để thao tác với frontend để gửi các resquest đến server, server sẽ trả lại các response cho ứng dụng và trả hiển thị lên cho người dùng.
* Khi người dùng có các request thao tác với datasebase, frontend sẽ gửi request đến server và server sẽ thao tác với database theo request của người dùng.

### Mô tả kiến trúc

Hệ thống được thiết kế với 2 phần chính, bao gồm: Client-side và Server-side. Cụ thể như sau:

* Client-side: Ứng dụng Android / IOS / Browser
* Người dùng tương tác với ứng dụng.
* Ứng dụng trao đổi dữ liệu với server thông qua phương thức HTTP.
* Server-side: Sử dụng SpringBoot kết nối CSDL MongoDB
* Server trao đổi dữ liệu với ứng dụng Android thông qua phương thức HTTP
* Server sử dụng SpringBoot kết nối tới cơ sở dữ liệu MongoDB.

## Công nghệ sử dụng

### JavaScript

* JavaScript là một ngôn ngữ lập trình đa nền tảng (cross-platform), ngôn ngữ lập trình kịch bản, hướng đối tượng. JavaScript là một ngôn ngữ nhỏ và nhẹ (small and lightweight). Khi nằm bên trong một môi trường (host environment), JavaScript có thể kết nối tới các object của môi trường đó và cung cấp các cách quản lý chúng (object).
* JavaScript chứa các thư viện tiêu chuẩn cho các object, ví dụ như: Array, Date, và Math, và các yếu tố cốt lõi của ngôn ngữ lập trình như: toán tử (operators), cấu trúc điều khiển (control structures), và câu lệnh. JavaScript có thể được mở rộng cho nhiều mục đích bằng việc bổ sung thêm các object; ví dụ:
* Client-side JavaScript - JavaScript phía máy khách, JavaScript được mở rộng bằng cách cung cấp các object để quản lý trình duyệt và Document Object Model (DOM) của nó. Ví dụ, phần mở rộng phía máy khách cho phép một ứng dụng tác động tới các yếu tố trên một trang HTML và phản hồi giống các tác động của người dùng như click chuột, nhập form, và chuyển trang.
* Server-side JavaScript - JavaScript phía Server, JavaScript được mở rộng bằng cách cung cấp thêm các đối tượng cần thiết để để chạy JavaScript trên máy chủ. Ví dụ, phần mở rộng phía server này cho phép ứng dụng kết nối với cơ sở dữ liệu (database), cung cấp thông tin một cách liên tục từ một yêu cầu tới phần khác của ứng dụng, hoặc thực hiện thao tác với các tập tin trên máy chủ.
* Trong ứng dụng, ta sẽ sử dụng Javascript để viết kịch bản ở phía frontend cho một trang web tĩnh. Javascript sẽ gửi các request đến của người sử dụng đến API và nhận lại các response từ phía server

### SpringBoot

* Spring là framework phát triển ứng dụng phổ biến nhất dành cho Java Enterprise. Ban đầu nó được viết bởi Rod Johnson và lần đầu tiên được phát hành theo giấy phép Apache 2.0 vào tháng 6 năm 2003. Spring có kích thướng nhẹ, phiên bản cơ bản của Spring framework có kích thước khoảng 2MB.
* Spring framework là một Java Platform mã nguồn mở, một giải pháp gọn nhẹ dành cho Java Enterprise. Với Spring Framework các nhà phát triển có thể tạo ra các mã có hiệu suất cao, dễ kiểm thử và có thể sử dụng lại được.
* Các tính năng core của Spring Framework có thể được sử dụng trong việc phát triển bất kỳ ứng dụng Java nào. Bên cạnh đó, phần mở rộng được sử dụng để xây dựng các ứng dụng web trên nền tảng Java EE. Mục tiêu của Spring Framework là làm cho việc phát triển ứng dụng J2EE dễ dàng hơn và thúc đẩy việc lập trình tốt hơn bằng mô hình POJO-based.
* Dưới đây là các ưu điểm của Spring Boot:
* Dễ dàng để phát triển các ứng dụng dựa trên Spring với Java hoặc Groovy.
* Giảm thiểu thời gian phát triển và tăng năng xuất
* Tránh việc phải viết nhiều mã nguyên mẫu (boilerplate Code), Annotations và các cấu hình XML.
* Dễ dàng để bạn tương tác các ứng dụng Spring Boot với các hệ sinh thái của Spring như Spring JDBC, Spring ORM, Spring Data, Spring Security etc.
* Đi theo cách tiếp cận “Nguyên tắc cấu hình mặc định” để giảm thiểu thời gian và nỗ lực cần thiết để phát triển ứng dụng.
* Cung cấp các Server nhúng (Embedded HTTP servers) như là Tomcat, Jetty.... để phát triển và test các ứng dụng web nhanh chóng và dễ dàng.
* Cung cấp công cụ CLI (Command Line Interface) dể phát triển và test các ứng dụng Spring Boot (Java hoặc Groovy) từ các dòng lệnh (command prompt) rất dễ dàng và nhanh chóng.
* Cung cấp rất nhiều các plugin để phát triển và test các ứng dụng Spring Boot nhanh chóng sử dụng các công cụ Build như Maven và Gradle
* Cung cấp nhiều plugin để làm việc với các cơ sở dữ liệu nhúng (embedded database) và các cơ sở dữ liệu lưu trữ trên bộ nhớ (in-memory Databases) một cách dễ dàng.
* Trong ứng dụng, Spring Boot sẽ được sử dụng để tạo ra một server gồm có các model, các interface để thao tác với cơ sở dữ liệu và cung cấp các API để các ứng dụng thao tác với server.

### MongoDB

* MongoDB là một chương trình cơ sở dữ liệu 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. Chính do cấu trúc linh hoạt này nên MongoDB có thể được dùng để lưu trữ các dữ liệu có cấu trúc phức tạp và đa dạng và không cố định (hay còn gọi là Big Data).
* Ưu điểm của MongoDB:
* Document oriented
* Hiệu suất cao
* Tính sẵn sàng cao – Nhân rộng
* Khả năng mở rộng cao – Sharding
* Năng động – Không có lược đồ cứng nhắc.
* Linh hoạt – thêm / xóa trường có ít hoặc không ảnh hưởng đến ứng dụng
* Dữ liệu không đồng nhất
* Không joins
* Phân phối được
* Biểu diễn dữ liệu trong JSON hoặc BSON
* Tích hợp dễ dàng với BigData Hadoop
* Ngôn ngữ truy vấn dựa trên tài liệu mạnh mẽ như SQL
* Trong ứng dụng, MongoDB sẽ được sử dụng để làm database lưu trữ dữ liệu của các Model trong server.
* Database là nơi lưu trữ dữ liệu của toàn bộ hệ thống. Hệ thống sử dụng MongoDB. Database sẽ được cài trên cùng một máy với máy chủ chứa web service và sử dụng các kết nối nội bộ để thao tác.

### React Native

React Native là framework được tạo bởi Facebook, cho phép các dev sử dụng JavaScript để làm mobile apps trên cả Android và iOS với cảm nhận và giao diện native. React Native giải quyết được vấn đề về hiệu suất trên Hybrid và tối ưu được chi phí khi mà phải viết nhiều loại ngôn ngữ native cho từng nền tảng di động.

Lý do sử dụng React Native:

* Hiệu quả về mặt thời gian khi mà bạn muốn phát triển một ứng dụng trên cả 2 nền tảng IOS và ANDROID
* Hiệu năng tương đối ổn định.
* Cộng đồng đang phát triển mạnh.
* Tiết kiệm chi phí thuê nhân lực
* Ứng dụng tin cậy và ổn định.
* Xây dựng cho nhiều hệ điều hành khác nhau với ít native code nhất.

## Server

# Đặc tả hệ thống

## Phân hệ người sử dụng

|  |  |  |  |
| --- | --- | --- | --- |
| UC #0001 | | **ĐĂNG NHẬP** | **Độ phức tạp: Low/ High** |
| **Mô tả** | | … | |
| **Tác nhân** | **Chính** | Người dùng | |
| **Phụ** | Không | |
| **Tiền điều kiện** | | Người dùng truy cập vào hệ thống. | |
| **Hậu điều kiện** | **Thành công** |  | |
| **Lỗi** |  | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính** | | | |
| Luồng chính:   * . | | | |
| **Luồng sự kiện phát sinh** | | | |
| **….** | | | |
|  | | | |
| **Các yêu cầu đặc biệt khác** | | | |
| Không | | | |
| **Giao diện minh họa** | | | |
| Hình …: Lời giải thích về hình | | | |
| Hình …: Lời giải thích về hình | | | |
| **THIẾT KẾ UML** | | | |
| **Sơ đồ Lớp phân tích (Analysis class diagram)** | | | |
| Hình …: Lời giải thích về hình | | | |
| Hình …: Lời giải thích về hình | | | |
| **Sơ đồ Trình tự (Sequence diagram)** | | | |
| Hình …: Lời giải thích về hình | | | |

# Thiết kế hệ thống

## Kiến trúc tổng thể

Hình …: Kiến trúc tổng thể hệ thống

## Thiết kế cơ sở dữ liệu

Hình …: Cơ sở dữ liệu của hệ thống

### Chi tết bảng….

|  |  |  |  |
| --- | --- | --- | --- |
| Số TT | Tên trường | Ý nghĩa | Kiểu dữ liệu |
| 1. |  |  |  |
| 2. |  |  |  |

Bảng …: Lời giải thích về bảng

# Cài đặt hệ thống

## Môi trường cài đặt

|  |  |
| --- | --- |
| Yêu cầu | Công nghệ |
| Hệ điều hành |  |
| Ngôn ngữ thiết kế server |  |
| Cơ sở dữ liệu |  |
| … |  |

Bảng …: Lời giải thích về bảng

## Cấu trúc thư mục

### Server

Hình …: Lời giải thích về hình

* Giải thích thành phần

### Client

Hình …: Lời giải thích về hình

* Giải thích thành phần

**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

Đã làm được gì trong khóa luận

Hướng phát triển của khóa luận

**TÀI LIỆU THAM KHẢO**

1. Nội dung tham khảo: [link web/](https://nodejs.org/en/docs/)
2. Tên tác giả, Tên nhà xuất bản (Năm xuất bản). Tên sách.