

**TRƯỜNG ĐẠI HỌC TIỀN GIANG
KHOA CÔNG NGHỆ THÔNG TIN**

LÊ LÂM KHÁNH DUY - 015101007

**NGHIÊN CỨU CÔNG NGHỆ MICROSERVICES
ỨNG DỤNG LẬP TRÌNH HỆ THỐNG CHỈ ĐẠO TUYẾN
VÀ NGHIÊN CỨU KHOA HỌC CHO NGÀNH Y TẾ**

**KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC
NGÀNH CÔNG NGHỆ THÔNG TIN**

NGƯỜI HƯỚNG DẪN

CH. LÊ ĐẶNG ĐĂNG KHOA

TS. DƯƠNG VĂN HIẾU

KHÓA 2015 – 2019

LỜI NÓI ĐẦU

Để hỗ trợ quy trình chuyển chỉ đạo tuyến từ các bệnh viện tuyến trên xuống các bệnh viện tuyến dưới và hỗ trợ quy trình quản lý NCKH cho ngành y tế, em chọn đề tài “Nghiên cứu công nghệ Microservices ứng dụng lập trình hệ thống chỉ đạo tuyến và nghiên cứu khoa học cho ngành y tế”.

LỜI CẢM ƠN



Đầu tiên em xin chân thành cảm ơn Ban Giám Hiệu cùng quý thầy cô trường Đại học Tiền Giang đặc biệt là toàn thể quý thầy cô trong khoa Công nghệ thông tin đã tận tình giảng dạy, trang bị cho em những kiến thức quý báu trong quá trình học tập tại trường để hoàn thành tốt khóa luận tốt nghiệp.

Tiếp theo, em xin chân thành cảm ơn sự giúp đỡ tận tình của các Anh/Chị trong công ty VNPT, hỗ trợ, giúp em tiếp cận với công nghệ Microservices, hướng dẫn em các quy trình của chỉ đạo tuyến và nghiên cứu khoa học cho ngành y tế.

Để hoàn thành tốt khóa luận tốt nghiệp này, em xin tỏ lòng biết ơn sâu sắc và lời cảm ơn chân thành đến thầy Dương Văn Hiếu – người đã tạo điều kiện để em có thể làm khóa luận theo đề tài của công ty VNPT. Bên cạnh đó là những ý kiến đóng góp của bạn bè đã cho em nguồn động viên lớn để hoàn thành nhiệm vụ của khóa luận. Qua đó, em đã đạt được nhiều tiến bộ về kiến thức cũng như những kỹ năng làm việc bổ ích.

Mặc dù đã rất cố gắng trong quá trình làm khóa luận tốt nghiệp, nhưng chắc chắn cũng không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự thông cảm và góp ý tận tình của quý thầy cô và các bạn. Em xin chân thành cảm ơn!

Cuối cùng em kính chúc quý Thầy/Cô và quý Anh/Chị trong công ty VNPT dồi dào sức khỏe và thành công trong công việc.

Em xin chân thành cảm ơn!

Tiền Giang, ngày 21 tháng 6 năm 2019

Sinh viên thực hiện

Lê Lâm Khánh Duy

ĐỀ CƯƠNG CHI TIẾT KHÓA LUẬN TỐT NGHIỆP

Tên đề tài: Nghiên cứu công nghệ Microservices ứng dụng lập trình hệ thống chỉ đạo tuyển và NCKH cho ngành y tế.

Người hướng dẫn: CH.Lê Đăng Đăng Khoa

TS. Dương Văn Hiếu

Thời gian thực hiện: 14/03/2019 – 21/06/2019.

Sinh viên thực hiện: Lê Lâm Khánh Duy

MSSV: 015101007

Loại đề tài: Xây dựng ứng dụng

Nội dung đề tài:

- Mục tiêu đề tài:

- + Tìm hiểu về Microservices.
- + Tìm hiểu về ReactJS, Typescript, Redux, JasperReports, Spring,... và các ngôn ngữ liên quan.
- + Tìm hiểu quy trình chỉ đạo tuyển, quy trình đăng ký, xét duyệt đề tài nghiên cứu khoa học cho ngành y tế.
- + Rèn luyện kỹ năng phân tích và giải quyết vấn đề.
- + Ứng dụng những kiến thức đã học và kiến thức đã tìm hiểu để xây dựng một hệ thống.

- Kết quả cần đạt được:

- + Công dụng: Hệ thống quản lý các thông tin cần lưu trữ của chỉ đạo tuyển và đề tài NCKH ngành y tế
- + Chức năng:
 - Đăng nhập dựa trên mô hình Microservices.
 - Quản lý thông tin của chỉ đạo tuyển, thống kê danh sách văn bản chỉ đạo tuyển, kế hoạch chỉ đạo tuyển, thông tin chỉ đạo tuyển hàng năm.
 - Quản lý thông tin của đề tài NCKH, thống kê danh sách đề tài đã đăng ký, đề tài được công nhận đề tài năm, đề tài được xét duyệt ứng dụng vào hoạt động của đơn vị hàng năm ở đơn vị.

- + Đặc điểm chính:
 - Hệ thống tích hợp hai phân hệ là chỉ đạo tuyến và NCKH cho ngành y tế.
 - Hệ thống hoạt động hiệu quả và chính xác.
- **Nhiệm vụ đề tài:**
 - + **Nghiên cứu thực tế:** Ứng dụng công nghệ Microservices để thiết kế hệ thống, kết hợp sử dụng ReactJS, Redux, Typescript, JasperReports giúp người lập trình có thêm kiến thức mới hỗ trợ lập trình web mạnh mẽ hơn.
 - + **Xác định phạm vi đề tài:** Đề tài để phục vụ các bệnh viện về quy trình chỉ đạo tuyến và NCKH ở bệnh viện.
 - + **Nghiên cứu tài liệu chuyên môn:** Trong quá trình thực hiện đề tài đã tham khảo một số tài liệu về lập trình Microservices, JavaScript, CSS, ReactJS, Redux, Typescript, JasperReports, Spring,...
- **Phân tích, thiết kế, cài đặt (hiện thực), thử nghiệm:**
 - + Phân tích quy trình chỉ đạo tuyến, quy trình đăng ký, xét duyệt đề tài NCKH.
 - + Thiết kế cơ sở dữ liệu.
 - + Tìm hiểu về ReactJS, Redux, Typescript, JasperReports....
 - + Lập trình Front-end.
 - + Lập trình Back-end.
 - + Thiết kế các báo cáo, thống kê.
 - + Chạy thử và sửa lỗi.
 - + Báo cáo và rút kinh nghiệm.
- **Yêu cầu khoa học công nghệ:**
 - + Hệ điều hành: Windows
 - + Trình soạn thảo mã nguồn: IntelliJ IDEA 2018
 - + Trình quản lý CSDL: MySQL Workbench 8.0 CE
 - + Trình duyệt web: Google Chrome, IE,....

Kế hoạch thực hiện

TT	Các nội dung,công việc chủ yếu cần thực hiện (các mốc đánh giá chủ yếu)	Từ ngày	Đến ngày
1	Phân tích đề tài	14/03/2019	21/03/2019
2	Thiết kế CSDL	22/03/2019	29/03/2019
3	Tìm hiểu về ReactJS, Redux, Typescript, JasperReports....	30/03/2019	13/04/2019
4	Lập trình Front-end	14/04/2019	05/05/2019
5	Lập trình Back-end	06/05/2019	06/06/2019
6	Thiết kế các báo cáo, thống kê	07/06/2019	10/06/2019
7	Chạy thử và sửa lỗi	11/06/2019	15/06/2019
8	Viết báo cáo và sửa lỗi đề tài	15/06/2019	20/06/2019

Xác nhận của người hướng dẫn

Ngày.....tháng.....năm 2019

Sinh viên thực hiện

Lê Lâm Khánh Duy

MỤC LỤC

Chương 1. TỔNG QUAN.....	1
1.1. Giới thiệu đề tài	1
1.1.1. Lý do chọn đề tài	1
1.1.2. Mục tiêu đề tài	1
1.1.3. Nhiệm vụ.....	1
1.1.4. Phạm vi nghiên cứu của đề tài	2
1.3. Các vấn đề tập trung giải quyết	2
Chương 2: CƠ SỞ LÝ THUYẾT VÀ THỰC TIỄN	4
2.1. Cơ sở lý thuyết.....	4
2.1.1. Giả thuyết nghiên cứu.....	4
2.1.2. Phương pháp nghiên cứu	4
2.1.3. Công nghệ sử dụng	5
2.1.4. Cơ sở lý thuyết.....	5
2.2. Cơ sở thực tiễn và ứng dụng của khóa luận.....	26
2.2.1. Bối cảnh thực tiễn	26
2.2.2. Ứng dụng	26
Chương 3: PHÂN TÍCH, THIẾT KẾ, CÀI ĐẶT GIẢI PHÁP	27
3.1. Phân tích yêu cầu	27
3.1.1. Phân tích yêu cầu chức năng.....	27
3.1.2. Phân tích yêu cầu phi chức năng.....	28
3.1.3. Phân tích quy trình.....	28
3.3. Phân tích thiết kế hệ thống.....	31
3.3.1. Hệ thống chỉ đạo tuyến	31
3.3.2. Hệ thống nghiên cứu khoa học	42
3.4. Cài đặt giải pháp	56
3.4.1. Xây dựng cấu trúc Microservice cho hệ thống chỉ đạo tuyến và hệ thống NCKH	56
3.4.2. Cài đặt hệ thống	63
3.4.3. Cách cài đặt hệ thống.....	70
3.4.4. Một số hình ảnh giao diện hệ thống.....	71
Chương 4: KẾT QUẢ VÀ HƯỚNG PHÁT TRIỂN	75
4.1. Kết quả đạt được	75
4.2. Hạn chế	75
4.3. Hướng phát triển	75
DANH MỤC TÀI LIỆU THAM KHẢO.....	76

DANH MỤC HÌNH

Hình 2.1. Minh họa mô hình kiến trúc nguyên khối [1]	6
Hình 2.2. Minh họa mô hình Microservices [1]	7
Hình 2.3. Mô hình Virtual DOM [2]	10
Hình 2.4. Giới thiệu về Props và State [3]	12
Hình 2.5. Mô phỏng Redux cho ứng dụng [3]	12
Hình 2.6. Mô hình hoạt động của Redux [3]	14
Hình 2.7. Minh họa TypeScript[4]	15
Hình 2.8. Các bước cài đặt JasperSoft Studio	16
Hình 2.9. Các bước cài đặt JasperSoft Studio	16
Hình 2.10. Các bước cài đặt JasperSoft Studio	17
Hình 2.11. Các bước tạo báo cáo bằng JasperSoft Studio	17
Hình 2.12. Các bước tạo báo cáo bằng JasperSoft Studio	18
Hình 2.13. Các bước tạo báo cáo bằng JasperSoft Studio	18
Hình 2.14. Các bước tạo báo cáo bằng JasperSoft Studio	19
Hình 2.15. Các bước tạo báo cáo bằng JasperSoft Studio	19
Hình 2.16. Các bước tạo báo cáo bằng JasperSoft Studio	20
Hình 2.17. Các bước tạo báo cáo bằng JasperSoft Studio	20
Hình 2.18. Các bước tạo báo cáo bằng JasperSoft Studio	21
Hình 2.19. Các bước tạo báo cáo bằng JasperSoft Studio	21
Hình 2.20. Các bước tạo báo cáo bằng JasperSoft Studio	22
Hình 2.21. Các bước tạo báo cáo bằng JasperSoft Studio	23
Hình 2.22. Các bước tạo báo cáo bằng JasperSoft Studio	23
Hình 2.23. Minh họa JHipster[6]	24
Hình 2.24. Minh họa tạo dự án JHipster	25
Hình 3.1. Sơ đồ Use case tổng quát (cán bộ phụ trách chỉ đạo tuyến)	31
Hình 3.2. Sơ đồ Use case tổng quát (nhân viên đơn vị tiếp nhận chỉ đạo tuyến)	31
Hình 3.3. Sơ đồ lớp	41
Hình 3.4. Mô hình cơ sở dữ liệu	42
Hình 3.5. Sơ đồ Use case tổng quát (Nhân viên quản lý NCKH)	42
Hình 3.6. Sơ đồ Use case tổng quát (Nhân viên đơn vị)	43
Hình 3.7. Sơ đồ Use case tổng quát (Cán bộ hội đồng NCKH)	44
Hình 3.8. Sơ đồ Use case tổng quát (Cán bộ đơn vị NCKH)	44
Hình 3.9. Sơ đồ lớp	55
Hình 3.10. Mô hình CSDL	56
Hình 3.11. Xây dựng cấu trúc cho hệ thống	57
Hình 3.12. Xây dựng cấu trúc cho hệ thống	58

Hình 3.13. Xây dựng cấu trúc cho hệ thống	58
Hình 3.14. Xây dựng cấu trúc cho hệ thống	58
Hình 3.14. Xây dựng dịch vụ gateway	59
Hình 3.15. Xây dựng dịch vụ gateway	59
Hình 3.16. Xây dựng dịch vụ gateway.....	60
Hình 3.17. Xây dựng dịch vụ gateway.....	60
Hình 3.18. Xây dựng dịch vụ gateway.....	61
Hình 3.19. Xây dựng dịch vụ gateway	61
Hình 3.20. Xây dựng dịch vụ chỉ đạo tuyến và NCKH	62
Hình 3.21. Xây dựng dịch vụ chỉ đạo tuyến và NCKH	62
Hình 3.23. Xây dựng dịch vụ chỉ đạo tuyến và NCKH	63
Hình 3.24. Xây dựng dịch vụ chỉ đạo tuyến và NCKH	63
Hình 3.25. Thêm dịch vụ vào IntteliJ IDEA 2018	64
Hình 3.26. Thêm dịch vụ vào IntteliJ IDEA 2018	65
Hình 3.28. Chạy dịch vụ trong IntteliJ IDEA 2018	65
Hình 3.29. Chạy dịch vụ trong IntteliJ IDEA 2018	66
Hình 3.30. Chạy dịch vụ trong IntteliJ IDEA 2018	66
Hình 3.31. Dịch vụ đã được chạy	66
Hình 3.32. Kết quả cài đặt hệ thống phân hệ chỉ đạo tuyến	67
Hình 3.33. Kết quả cài đặt hệ thống phân hệ chỉ đạo tuyến	67
Hình 3.35. Minh họa cài đặt lấy danh sách hình thức của dịch vụ chỉ đạo tuyến.....	68
Hình 3.36. Kết quả cài đặt hệ thống phân hệ NCKH.....	69
Hình 3.37. Kết quả cài đặt hệ thống phân hệ NCKH.....	69
Hình 3.38. Kết quả cài đặt hệ thống phân hệ NCKH.....	70
Hình 3.39. Minh họa cài đặt lấy danh sách loại đề tài của dịch vụ NCKH	70
Hình 3.40. Cách cài đặt hệ thống.....	71
Hình 3.41. Giao diện đăng nhập	71
Hình 3.42. Giao diện chính.....	72
Hình 3.43. Giao diện Quản lý nội dung chỉ đạo tuyến	72
Hình 3.44. Giao diện thêm mới nội dung chỉ đạo tuyến.....	72
Hình 3.45. Giao diện Xem thông tin nội dung chỉ đạo tuyến	73
Hình 3.46. Giao diện Cập nhật thông tin nội dung chỉ đạo tuyến	73
Hình 3.47. Giao diện Xóa nội dung chỉ đạo tuyến	73
Hình 3.48. Giao diện Thống kê chỉ đạo tuyến	74
Hình 3.49. Giao diện Xuất báo cáo chỉ đạo tuyến	74
Hình 3.50. Giao diện Dashboard	74

BẢNG CÁC TỪ VIẾT TẮT

Từ viết tắt	Mô tả
CSDL	Cơ sở dữ liệu
NCKH	Nghiên cứu khoa học
API	Application Programming Interface (giao diện chương trình ứng dụng)
Microservices	Mô hình chia nhỏ dịch vụ
JSX	JavaScript XML
JDBC	Java Database Connectivity (kết nối CSDL dùng Java)
JPA	Java Persistence API
VNPT	Vietnam Posts and Telecommunications Group (Tập đoàn Bưu chính Viễn thông Việt Nam)

TÓM TẮT KHÓA LUẬN

1. Mục tiêu

Tìm hiểu lý thuyết: Hiểu được cách hoạt động của Microservices, nắm được các kiến thức nền tảng của lập trình sử dụng ReactJS, Redux, TypeScript, Spring,...Nắm rõ quy trình chỉ đạo tuyển và quy trình đăng ký, xét duyệt đề tài NCKH cho ngành y tế.

Ứng dụng thực nghiệm: Xây dựng thành công hệ thống Microservices cho hệ thống chỉ đạo tuyển và nghiên cứu khoa học.

2. Nhiệm vụ

Tìm hiểu lý thuyết: Tìm hiểu cách hoạt động của Microservices, tìm hiểu lập trình sử dụng ReactJS, Redux, TypeScript, Spring...Tìm hiểu quy trình chỉ đạo tuyển và quy trình đăng ký, xét duyệt đề tài NCKH cho ngành y tế.

Ứng dụng thực nghiệm: Đặt ra và phân tích yêu cầu, liệt kê các chức năng cần thiết, thiết kế cơ sở dữ liệu, xây dựng cấu trúc ứng dụng, cấu hình ứng dụng, thiết kế giao diện, thiết kế các báo cáo, thống kê, cài đặt giải thuật và hiện thực các chức năng đã liệt kê.

3. Phương pháp giải quyết

Tìm hiểu lý thuyết: Phương pháp nghiên cứu lý thuyết, mô hình hóa, sơ đồ.

Ứng dụng thực nghiệm: Phương pháp thực nghiệm khoa học, phân tích và tổng kết kinh nghiệm.

4. Kết quả đạt được

Tìm hiểu lý thuyết: Hiểu được cách hoạt động của Microservices, nắm được các kiến thức nền tảng của lập trình sử dụng ReactJS, Redux, TypeScript, Spring, ...Nắm rõ quy trình chỉ đạo tuyển và quy trình đăng ký, xét duyệt đề tài NCKH cho ngành y tế.

Ứng dụng thực nghiệm: Xây dựng thành công hệ thống đáp ứng các chức năng đã đặt ra.

Chương 1. TỔNG QUAN

1.1. Giới thiệu đề tài

1.1.1. Lý do chọn đề tài

Hiện tại với tốc độ phát triển vượt trội của thông tin truyền thông và Internet, việc sử dụng công nghệ mới để thiết kế web cũng được các doanh nghiệp quan tâm. Trong số những công nghệ lập trình web hiện nay, Microservices được quan tâm đáng kể với nhiều ưu điểm về bảo mật và ổn định. Chính vì có ưu điểm về bảo mật cao nên được sử dụng nhiều trong y tế, chỉ đạo tuyến và NCKH là một bộ phận trong y tế, có vai trò quan trọng, cần phải được quản lý một cách bảo mật. Xuất phát từ đó, em chọn đề tài “Nghiên cứu công nghệ Microservices ứng dụng lập trình hệ thống chỉ đạo tuyến và NCKH cho ngành y tế”.

1.1.2. Mục tiêu đề tài

Tìm hiểu lý thuyết:

- Hiểu rõ mô hình Microservices, củng cố kiến thức và thành thạo kỹ năng lập trình ReactJS, Redux, Typescript, Spring, ...
- Học cách viết Microservices, cách cấu trúc chương trình một cách hiệu quả.
- Nắm được quy trình chỉ đạo tuyến và đăng ký, xét duyệt đề tài NCKH cho ngành y tế.

Ứng dụng thực nghiệm:

- Phát triển hệ thống cho phân hệ chỉ đạo tuyến và NCKH cho ngành y tế.
- Ứng dụng hoạt động hiệu quả, nhanh chóng.
- Giao diện đơn giản, dễ sử dụng.
- Xây dựng ứng dụng với các phân quyền cho từng nhóm người dùng.

1.1.3. Nhiệm vụ

Tìm hiểu lý thuyết:

- Tìm hiểu về mô hình Microservices, ôn tập lại các kiến thức và thực hành kỹ năng lập trình.
- Tìm hiểu cách cấu hình, các thư viện hỗ trợ Microservices.

- Tìm hiểu các quy trình chỉ đạo tuyển, quy trình đăng ký, xét duyệt đề tài NCKH.

Ứng dụng thực nghiệm:

- Thử nghiệm thành lập một mô hình Microservices nhỏ để hiểu cách thức hoạt động.
- Đặt ra và phân tích các yêu cầu dựa trên các quy trình.
- Liệt kê các chức năng cần thiết.
- Thiết kế cơ sở dữ liệu.
- Xây dựng cấu trúc ứng dụng.
- Cấu hình ứng dụng.
- Thiết kế giao diện.
- Cài đặt giải thuật và hiện thực các chức năng đã liệt kê.
- Sửa chữa, cải tiến, hoàn thiện và triển khai ứng dụng.

1.1.4. Phạm vi nghiên cứu của đề tài

Tìm hiểu lý thuyết: Để có thể sử dụng thành thạo Microservices thì cần phải tìm hiểu rất nhiều, nên trong phạm vi khóa luận này, em chỉ tập trung vào tìm hiểu những kỹ thuật cần thiết cho hệ thống ở mức độ biết cách sử dụng.

Ứng dụng thực nghiệm: Trong phạm vi khóa luận này, ứng dụng web của em chỉ sử dụng CSDL MySQL.

1.2. Các kết quả nghiên cứu liên quan tới đề tài

Hệ thống chỉ đạo tuyển cũ:

- Sử dụng công nghệ trước đây để thiết kế, chưa đảm bảo bảo mật
- Chưa đáp ứng nhu cầu hiện tại

Hệ thống quản lý NCKH cũ:

- Sử dụng công nghệ trước đây để thiết kế, chưa đảm bảo bảo mật
- Quy trình đăng ký, xét duyệt được cập nhật nên hệ thống chưa đáp ứng kịp

1.3. Các vấn đề tập trung giải quyết

Giải quyết các vấn đề sau:

- Thiết kế theo kiến trúc Microservice, đảm bảo tính bảo mật, hoạt động ổn định.
- Giao diện đơn giản, dễ sử dụng.
- Bố cục hợp lí, các menu rõ ràng.
- Có thể chạy trên các trình duyệt web trên máy tính.

Chương 2: CƠ SỞ LÝ THUYẾT VÀ THỰC TIỄN

2.1. Cơ sở lý thuyết

2.1.1. Giả thuyết nghiên cứu

Nếu xây dựng thành công hệ thống sẽ giúp việc chỉ đạo tuyển và quản lý đăng ký, xét duyệt đề tài NCKH nhanh chóng, dễ dàng hơn.

2.1.2. Phương pháp nghiên cứu

Phương pháp nghiên cứu lý thuyết: Nghiên cứu lý thuyết các vấn đề liên quan như cơ sở dữ liệu, ngôn ngữ lập trình, tài liệu. Nghiên cứu tài liệu:

- Tài liệu: “THÔNG TƯ QUY ĐỊNH VỀ CHỈ ĐẠO TUYỂN TRONG LĨNH VỰC KHÁM BỆNH, CHỮA BỆNH” của bộ Y tế năm 2016.
- Tài liệu: “QUY TRÌNH NGHIỆM THU ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CẤP CƠ SỞ QT.33.HT” của bệnh viện Bạch Mai.

Phương pháp quan sát khoa học: Quan sát khách quan các vấn đề liên quan đến quy trình chỉ đạo tuyển, quy trình đăng ký, xét duyệt đề tài NCKH của ngành y tế.

Phương pháp thực nghiệm: Chạy thử nghiệm phần mềm.

Phương pháp nghiên cứu dựa theo quy trình: Quy trình thiết kế gồm các bước sau đây:

- Xác định yêu cầu: Thu thập thông tin từ các văn bản pháp luật.
- Phân tích: Phân tích, nghiên cứu để giải quyết vấn đề từ các yêu cầu thu thập được từ bước xác định yêu cầu, vẽ các mô hình, lớp xử lý dữ liệu của bài toán.
- Thiết kế: Thiết kế giao diện và các giải thuật, các lớp xử lý từ việc phân tích yêu cầu để có thể hoàn thành hệ thống.
- Cài đặt: Sau khi hoàn thành cơ bản chức năng của hệ thống, cần cho chạy thử nghiệm để kiểm tra, tìm lỗi và sửa lỗi...
- Kiểm chứng: Thu thập, lấy ý kiến đánh giá của người sử dụng. Từ đó, tổng kết để thấy được những cái hoàn thiện, thiếu sót của hệ thống có thể chỉnh sửa kịp thời.

– Phát triển thực tế: Nếu đề tài được ứng dụng sẽ triển khai và cài đặt trên host để có thể chính thức hoạt động sẽ hỗ trợ cho quy trình chỉ đạo tuyển và đăng ký, xét duyệt đề tài NCKH.

2.1.3. Công nghệ sử dụng

Hệ điều hành: Windows 10

Ngôn ngữ lập trình chính: ReactJS

Hệ quản trị cơ sở dữ liệu: MySQL Workbench 8.0 CE

Môi trường lập trình: Trình soạn thảo mã nguồn: IntelliJ IDEA 2018

Trình duyệt web: Google Chrome, Microsoft Edge

2.1.4. Cơ sở lý thuyết

❖ Giới thiệu về Microservices

a. Tổng quan về Microservices

Microservices - còn được gọi là kiến trúc Microservices - là một kiểu kiến trúc cấu trúc một ứng dụng như một tập hợp các dịch vụ khác nhau.

Đặc điểm của Microservice: Có khả năng duy trì và kiểm tra cao, kết nối các thành phần riêng lẻ lại với nhau, có thể triển khai độc lập, tổ chức xung quanh khả năng kinh doanh.

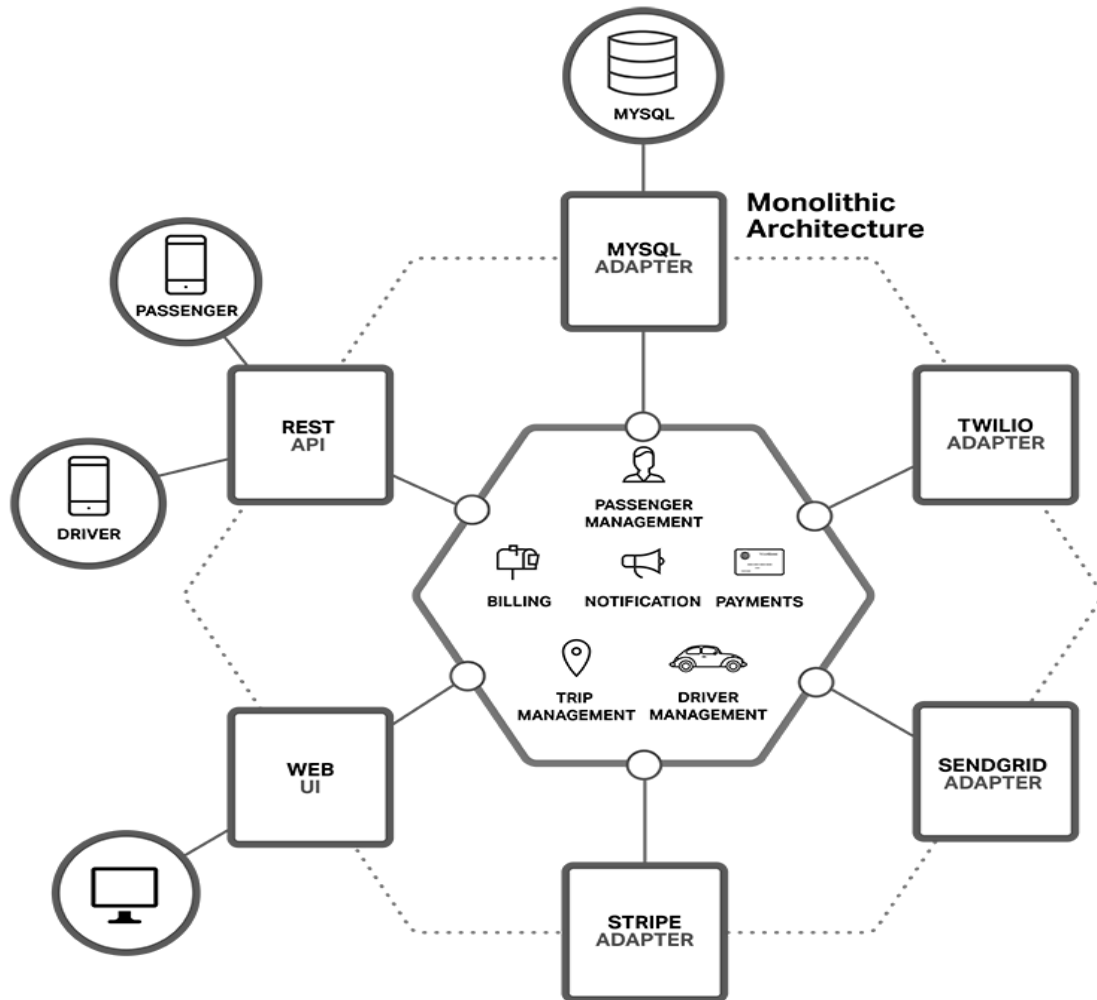
Kiến trúc microservice cho phép phân phối / triển khai liên tục các ứng dụng lớn, có độ phức tạp cao.

b. Phân biệt giữa kiến trúc nguyên khối và kiến trúc Microservices

Kiến trúc nguyên khối

Người lập trình xây dựng hệ thống chỉ trong một khối cung cấp các tính năng đáp ứng nhu cầu của người sử dụng. Khi hệ thống đưa vào sử dụng, và thành công của hệ thống sẽ thu hút nhiều người sử dụng, từ đó, nhu cầu về tính năng mới của hệ thống sẽ tăng cao. Khi phải thêm nhiều tính năng mới vào hệ thống, dữ liệu sẽ tăng lên, dẫn đến hệ thống sẽ càng ngày càng phức tạp, việc bảo trì, nâng cấp cho hệ thống cũng sẽ gặp rất nhiều khó khăn, thậm chí là không thể, hoặc chi phí bỏ ra để xây dựng hệ thống khác sẽ ít hơn chi phí sửa chữa cho hệ thống cũ. Khi hệ thống càng lớn, mỗi lần cho ra phiên bản mới cũng sẽ rất khó

khẩn, có thể làm trễ tiến độ của người sử dụng. Ngoài ra, hệ thống lớn sẽ ảnh hưởng đến khả năng làm việc của các lập trình viên tham gia phát triển hệ thống.



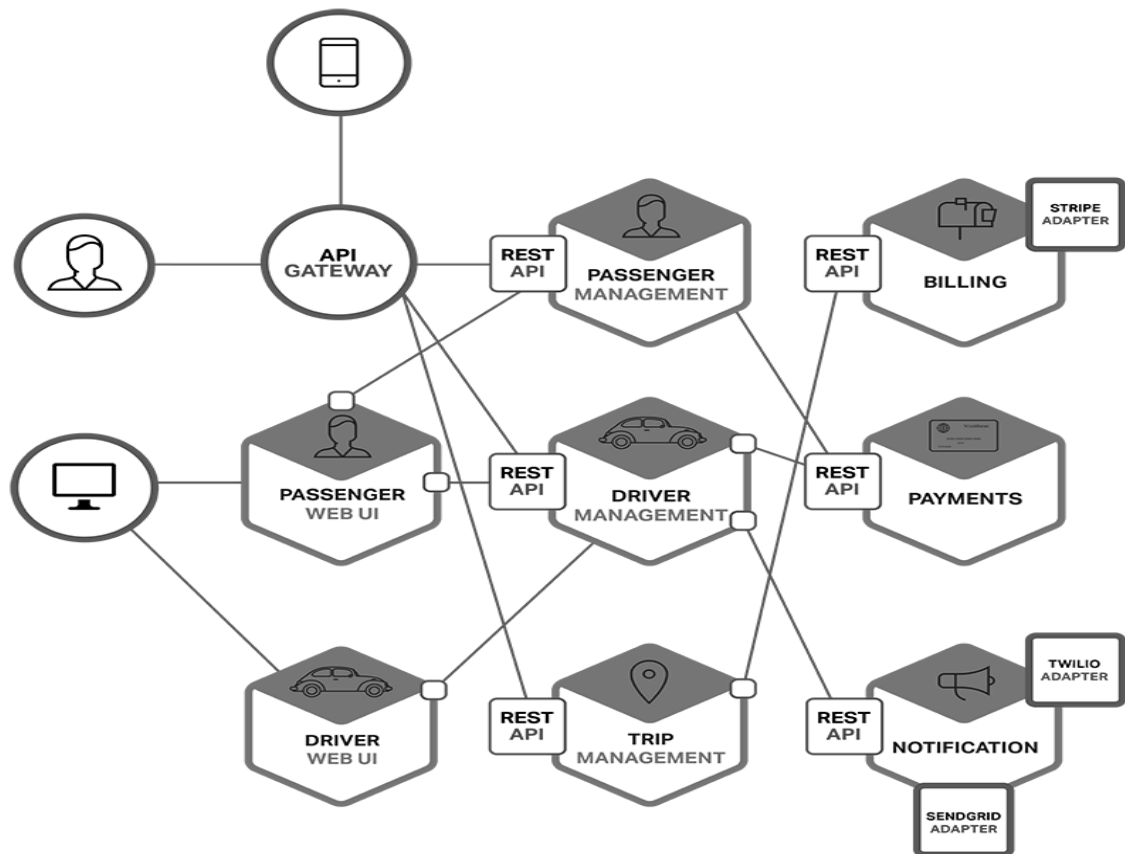
Hình 2.1. Minh họa mô hình kiến trúc nguyên khối [1]

Kiến trúc Microservices

Để giải quyết những vấn đề của kiến trúc nguyên khối, ý tưởng được đề xuất là chia nhỏ hệ thống lớn ra thành những dịch vụ nhỏ kết nối với nhau. Mỗi dịch vụ nhỏ thực hiện một tập các chức năng chuyên biệt như quản lý đơn hàng, quản lý khách hàng,... Mỗi dịch vụ là một ứng dụng nhỏ kết nối thông qua bộ chuyển đổi (Adapter) khác nhau. Một số dịch vụ có nhiệm vụ làm API cho phép giao tiếp với các dịch vụ nhỏ khác hay các ứng dụng khác gọi tới. Khi vận hành, mỗi dịch vụ nhỏ được chạy trong một máy ảo (Virtual Machine) hoặc Docker Container (ảo hóa tầng ứng dụng).

Như vậy, mỗi vùng chức năng giờ đã được thực thi bởi một dịch vụ nhỏ, ứng dụng web cũng có thể chia nhỏ hơn cho từng nhóm đối tượng người dùng (quản lý, khách hàng,..). Thiết kế giao diện cho từng nhóm đối tượng người dùng giúp tối ưu hóa trải nghiệm, tốc độ nhanh hơn, dễ tương thích hơn trong khi cấu trúc chức năng đơn giản hơn.

Kiến trúc microservices chú trọng nhiều đến quan hệ giữa ứng dụng và CSDL. Thay vì dùng chung một cơ sở dữ liệu giữa các dịch vụ, mỗi dịch vụ sẽ có CSDL riêng. Cách này khác so với cơ sở dữ liệu tập trung (Toàn bộ ứng dụng sử dụng chung một CSDL thống nhất). Hệ quả là sẽ có sự dư thừa dữ liệu, cơ chế ràng buộc khóa ngoại cho quan hệ dữ liệu không thể áp dụng với bảng ở 2 cơ sở dữ liệu tách biệt.



Hình 2.2. Minh họa mô hình Microservices [1]

Kết luận phân biệt giữa hai kiến trúc

Kiến trúc một khối sẽ hữu hiệu đối với ứng dụng đơn giản, ít chức năng. Nó bộc lộ nhiều nhược điểm khi ứng dụng phát triển lớn nhiều chức năng.

Kiến trúc Microservices chia nhỏ kiến trúc một khối ra các dịch vụ nhỏ. Microservices sẽ hiệu quả, phù hợp cho những ứng dụng phức tạp, liên tục phát triển nếu được thiết kế đúng và tận dụng các công nghệ quản lý, vận hành tự động.

c. Ưu điểm của kiến trúc Microservices

Đơn giản hóa hệ thống lớn: Vì hệ thống được phân chia ra thành nhiều dịch vụ nhỏ hơn giao tiếp với nhau nên hệ thống lớn sẽ giảm sự phức tạp của hệ thống đi rất nhiều.

Chia nhỏ ứng dụng một khối công kênh thành các dịch vụ nhỏ dễ quản lý, bảo trì nâng cấp, tự do chọn, nâng cấp công nghệ mới: Hệ thống được chia ra thành nhiều dịch vụ nhỏ hơn nên khi xảy ra lỗi ở một chức năng nào đó trong hệ thống, chỉ cần tìm đến dịch vụ thực hiện chức năng đó để bảo trì sửa chữa, tương tự quá trình nâng cấp, phát triển hệ thống cũng sẽ đơn giản hơn rất nhiều.

Microservices thúc đẩy tách rạch rời các khối chức năng : Điều rất khó thực hiện với ứng dụng một khối. Nếu muốn tạo sự rạch rời trong ứng dụng một khối, sẽ phải thiết kế theo mô hình kiến trúc đặc biệt và phải liên tục tái cấu trúc hệ thống.

Mỗi dịch vụ nhỏ sẽ phát triển dễ hơn, nhanh hơn, dễ viết mã kiểm thử tự động cho từng dịch vụ nhỏ, điều mà kiến trúc nguyên khối có thể gặp khó khăn.

Một số dịch vụ nhỏ có thuê ngoài phát triển mà vẫn bảo mật hệ thống và mã nguồn phần dịch vụ còn lại. Đội phát triển có nhiều lựa chọn công nghệ mới, khung phần mềm (framework), CSDL mới, đa dạng để nâng cấp từng dịch vụ nhỏ, chọn môi trường tối ưu nhất để chạy. Các dịch vụ có thể bật tắt để kiểm nghiệm so sánh, tăng tốc quá trình cải tiến giao diện. Việc triển khai phần mềm đều đặn khả thi với Microservices. Dịch vụ nhỏ đóng gói có thể chuyển từ môi trường phát triển sang môi trường chạy thật không phải cấu hình thủ công lại, không phải copy tập tin quá lớn.

d. Nhược điểm của Microservices

Nhược điểm đầu tiên của Microservices là Microservices nhấn mạnh kích thước nhỏ gọn của dịch vụ. Chia quá nhiều sẽ dẫn đến rời rạc, vụn vặt, khó kiểm soát. Việc lưu dữ liệu cục bộ bên trong những dịch vụ quá nhỏ sẽ khiến dữ liệu phân tán quá mức cần thiết.

Do hệ thống được chia thành những dịch vụ nhỏ nên phải xử lý sự cố khi kết nối chậm, lỗi khi thông điệp không gửi được hoặc thông điệp gửi đến nhiều đích đến vào các thời điểm khác nhau.

Đảm bảo giao dịch phân tán cập nhật dữ liệu đúng đắn vào nhiều dịch vụ nhỏ khác nhau rất khó, đôi khi là không thể so với đảm bảo giao dịch cập nhật vào nhiều bảng trong một cơ sở dữ liệu trung tâm.

Kiểm thử tự động một dịch vụ trong kiến trúc Microservices đôi khi yêu cầu phải chạy cả các dịch vụ nhỏ khác mà nó phụ thuộc. Do đó khi phân rã ứng dụng một khối thành Microservices cần luôn kiểm tra mức độ ràng buộc giữa các dịch vụ một cách cẩn thận.

Nếu các dịch vụ nhỏ thiết kế phụ thuộc vào nhau theo chuỗi công việc A gọi B, B gọi C, C gọi D. Nếu một mắt xích có giao tiếp API thay đổi, các mắt xích khác có thể sẽ bị ảnh hưởng. Vì vậy, việc thiết kế dịch vụ tốt sẽ giảm tối đa ảnh hưởng lan truyền đến các dịch vụ khác, và cần phải có quy tắc phân luồng để trách tình trạng tắc nghẽn ở một dịch vụ nào đó.

❖ Giới thiệu về ReactJS, Redux, Typescript, JasperReports

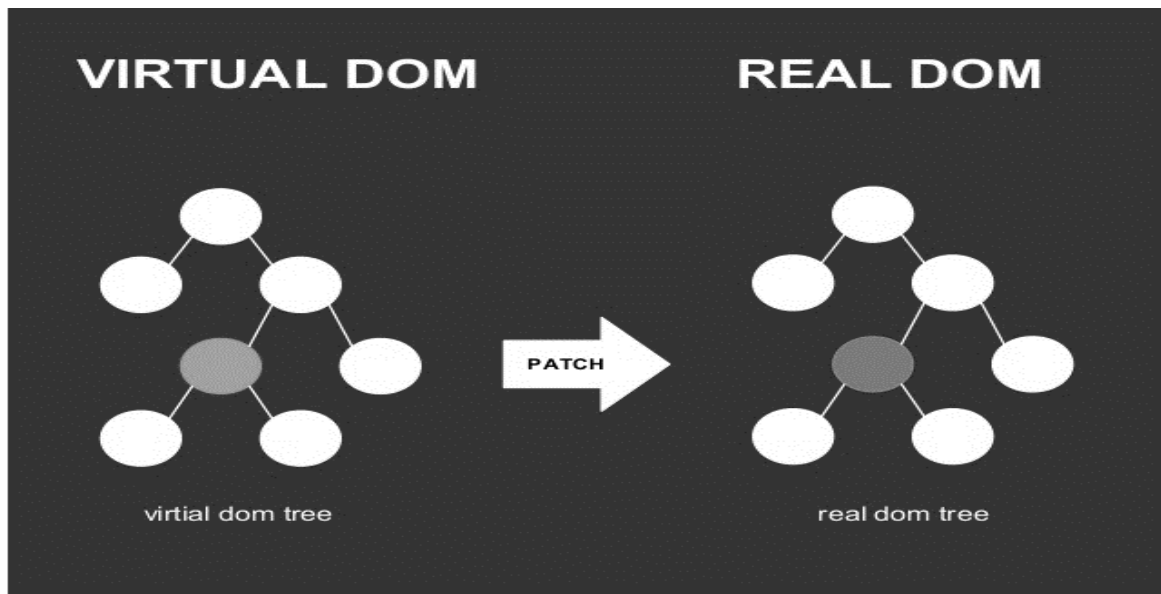
a. Giới thiệu về ReactJS

ReactJS là một thư viện giao diện người dùng, được Facebook phát triển để hỗ trợ việc xây dựng những thành phần giao diện người dùng có tính tương tác cao, có trạng thái và có thể sử dụng lại được. ReactJS được sử dụng tại Facebook trong sản xuất, và cả Instagram cũng được viết hoàn toàn trên ReactJS.

Một trong những điểm hấp dẫn của ReactJS là thư viện này không chỉ hoạt động trên phía người dùng, mà còn được kết xuất (render) trên máy chủ và có thể kết nối với nhau. ReactJS so sánh sự thay đổi giữa các giá trị của lần kết xuất này với lần kết xuất trước và cập nhật ít thay đổi nhất trên DOM (Document Object Model – Mô hình đối tượng tài liệu).

Mô hình đối tượng tài liệu ảo (Virtual DOM – DOM ảo):

Công nghệ DOM ảo giúp tăng hiệu năng cho ứng dụng. Việc chỉ nút gốc mới có trạng thái và khi nó thay đổi sẽ tái cấu trúc lại toàn bộ, đồng nghĩa với việc cây DOM cũng sẽ phải thay đổi một phần, điều này sẽ ảnh hưởng đến tốc độ xử lý. ReactJS sử dụng Virtual DOM để cải thiện vấn đề này. Virtual DOM là một đối tượng trong Javascript, mỗi đối tượng chứa đầy đủ thông tin cần thiết để tạo ra một DOM, khi dữ liệu thay đổi nó sẽ tính toán sự thay đổi giữa đối tượng và cấu trúc cây thật, điều này sẽ giúp tối ưu hoá việc kết xuất lại cây DOM thật.



Hình 2.3. Mô hình Virtual DOM [2]

ReactJS sử dụng cơ chế luồng dữ liệu một chiều. Dữ liệu được truyền từ thành phần cha đến thành phần con thông qua ‘props’. Luồng dữ liệu đơn giản giúp chúng ta dễ dàng kiểm soát cũng như sửa lỗi. Với các đặc điểm ở trên, ReactJS dùng để xây dựng các ứng dụng lớn mà dữ liệu của chúng thay đổi liên tục theo thời gian. Dữ liệu thay đổi thì hầu hết kèm theo sự thay đổi về thông tin được hiển thị trên giao diện. Ví dụ như Facebook: trên mục bảng tin của người dùng cùng lúc sẽ có các bài đăng khác nhau và mỗi bài đăng lại có số lượt thích, chia sẻ, bình luận liên tục thay đổi. Khi đó ReactJS sẽ rất hữu ích để sử dụng.

Giới thiệu về JSX

JSX là một dạng ngôn ngữ cho phép viết các mã HTML trong Javascript. JSX thực hiện tối ưu hóa trong khi biên dịch sang mã Javascript. Các mã này cho

thời gian thực hiện nhanh hơn nhiều so với một mã tương đương viết trực tiếp bằng Javascript.

Ngược với Javascript, JSX là kiểu được biên dịch trước khi chạy, giống như Java, C++. Vì thế các lỗi sẽ được phát hiện ngay trong quá trình biên dịch. Ngoài ra, nó cũng cung cấp tính năng gỡ lỗi khi biên dịch rất tốt.

JSX được thiết kế dựa trên Javascript, vì vậy rất dễ dàng để cho các lập trình viên Javascripts có thể sử dụng.

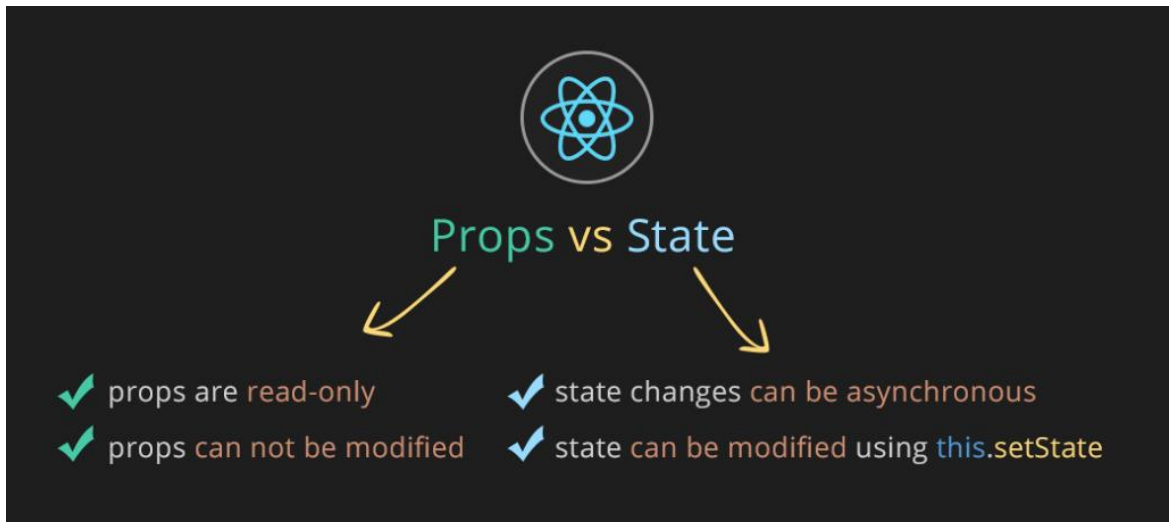
Giới thiệu về Components (thành phần)

ReactJS được xây dựng xung quanh các thành phần, gọi là Component, chứ không dùng khung mẫu như các framework khác. Trong ReactJS, chúng ta xây dựng trang web sử dụng những Component nhỏ. Chúng ta có thể tái sử dụng một Component ở nhiều nơi, với các trạng thái hoặc các thuộc tính khác nhau, trong một Component lại có thể chứa Component khác. Mỗi Component trong React có một trạng thái riêng, có thể thay đổi, và ReactJS sẽ thực hiện cập nhật Component dựa trên những thay đổi của trạng thái. Mọi thứ trong React đều là Component. Chúng giúp bảo trì mã lệnh khi làm việc với các dự án lớn. Một React Component đơn giản chỉ cần một phương thức kết xuất, gọi là Render Method. Có rất nhiều phương thức khả dụng khác, nhưng Render Method là phương thức chủ yếu.

Giới thiệu về Props và States

Props: Giúp các Component tương tác với nhau, Component nhận giá trị trả về gọi là props, và trả thuộc tính mô tả những gì Component con sẽ kết xuất. Props là phân tử không thay đổi.

States: Thể hiện trạng thái của ứng dụng, khi state thay đổi thì Component đồng thời kết xuất lại để cập nhật giao diện người dùng. States có thể thay đổi bằng cách sử dụng phương thức `this.setState()`;



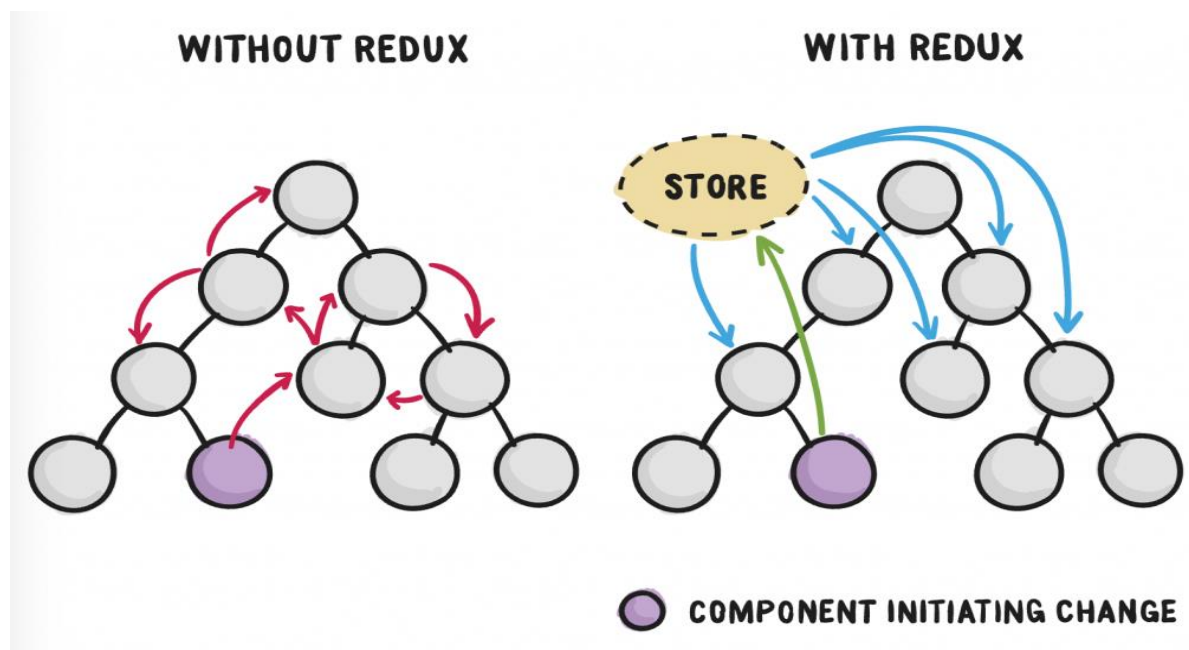
Hình 2.4. Giới thiệu về Props và State [3]

b. Giới thiệu về Redux

Redux là một công cụ quản lý cho các ứng dụng Javascript, có thể gọi Redux là một kho chứa. Nó giúp bạn viết các ứng dụng hoạt động một cách nhất quán, chạy trong các môi trường khác nhau (client, server, and native).

Mặc dù chủ yếu được sử dụng với React, nhưng nó có thể được sử dụng với bất kỳ thư viện hoặc Framework JavaScript nào khác.

Với Redux, States của ứng dụng được giữ trong một nơi gọi là Store và mỗi Component đều có thể truy cập bất kỳ States nào mà chúng muốn từ Store này.



Hình 2.5. Mô phỏng Redux cho ứng dụng [3]

Cách hoạt động của Redux:

Redux có một Store lưu trữ toàn bộ State của nó. Mỗi Component có thể truy cập trực tiếp đến State được lưu trữ thay vì phải gửi Props từ Component này đến Component khác. Có 3 thành phần của Redux: Actions, Store, Reducers.

Action:

Actions đơn giản là các sự kiện. Chúng là cách mà chúng ta gửi dữ liệu từ ứng dụng đến Store. Những dữ liệu này có thể là từ sự tương tác của người dùng với ứng dụng, gọi từ API hoặc cũng có thể là từ gửi dữ liệu của bảng.

Actions được gửi bằng cách sử dụng phương thức `Store.dispatch()`, chúng phải có một kiểu để biểu lộ loại sự kiện để thực hiện. Chúng cũng phải có một khối dữ liệu, gọi là payload chứa thông tin.

Reducers:

Reducers là các phương thức nguyên thủy (Pure Functions) chúng lấy State hiện tại của ứng dụng, thực hiện một Action và trả về một state mới. Những State này được lưu như những đối tượng và chúng định rõ cách State của một ứng dụng thay đổi trong việc phản hồi một Action được gửi đến Store.

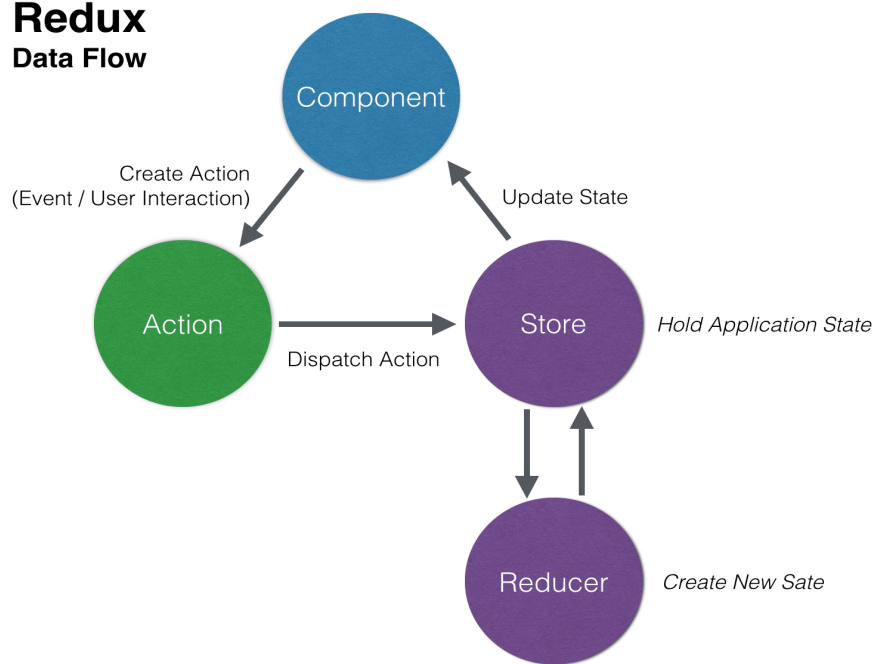
Phương thức nguyên thủy: Một hàm thuần túy (pure) chỉ là một hàm bình thường với hai ràng buộc bổ sung mà nó buộc thỏa mãn:

- Với một tập hợp các dữ liệu đầu vào, hàm luôn luôn trả về cùng một kết quả.
- Hàm này không sinh ra sự kiện phụ.

Store:

Store lưu trạng thái ứng dụng và nó là duy nhất trong bất kỳ một ứng dụng Redux nào. Người lập trình có thể truy cập các State được lưu, cập nhật State.

Redux Data Flow



Hình 2.6. Mô hình hoạt động của Redux [3]

c. Giới thiệu về TypeScript

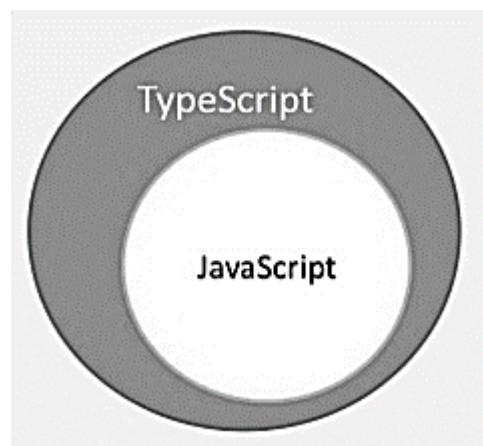
TypeScript là một dự án mã nguồn mở được phát triển bởi Microsoft, nó có thể được coi là một phiên bản nâng cao của Javascript bởi việc bổ sung tùy chọn kiểu tĩnh và lớp hướng đối tượng mà điều này không có ở Javascript. TypeScript có thể sử dụng để phát triển các ứng dụng chạy ở Client-side (Angular2) và Server-side (NodeJS). TypeScript sử dụng tất cả các tính năng của của ECMAScript 2015 (viết tắt là ES6) như classes, modules.

Lí do nên sử dụng Typescript:

- Dễ phát triển dự án lớn: Với việc sử dụng các kỹ thuật mới nhất và lập trình hướng đối tượng nên TypeScript giúp chúng ta phát triển các dự án lớn một cách dễ dàng.
- Nhiều Framework lựa chọn: Hiện nay các Javascript Framework đã dần khuyến khích nên sử dụng TypeScript để phát triển.
- Hỗ trợ các tính năng của Javascript phiên bản mới nhất: TypeScript luôn đảm bảo việc sử dụng đầy đủ các kỹ thuật mới nhất của Javascript, ví dụ như version ECMAScript 2015 (ES6).

– Là mã nguồn mở: TypeScript là một mã nguồn mở nên người dùng hoàn toàn có thể sử dụng mà không mất phí, bên cạnh đó còn được cộng đồng hỗ trợ.

– TypeScript là Javascript: Bản chất của TypeScript là biên dịch tạo ra các đoạn mã Javascript nên người dùng có thể chạy bất kỳ môi trường nào miễn ở đó có hỗ trợ biên dịch Javascript. Ngoài ra bạn có thể sử dụng trộn lẫn cú pháp của Javascript vào bên trong TypeScript, điều này giúp các lập trình viên tiếp cận TypeScript dễ dàng hơn.



Hình 2.7. Minh họa TypeScript[4]

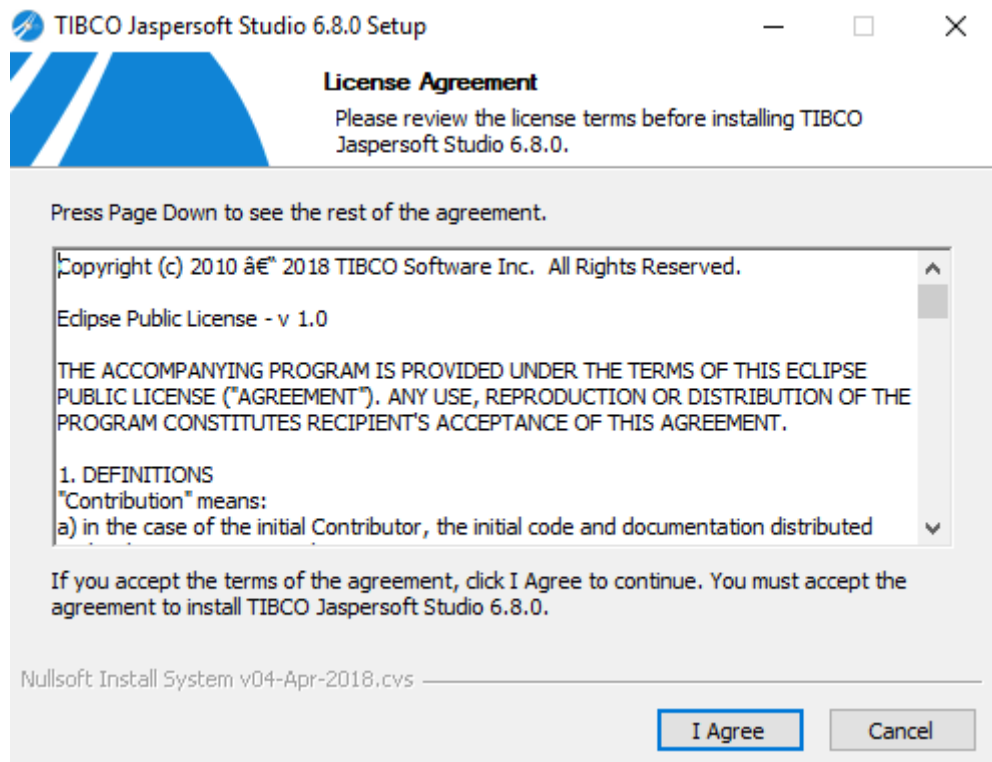
❖ Giới thiệu về JasperReports

JasperReports là một công cụ tạo báo cáo mã nguồn mở phổ biến nhất hiện tại. Nó được viết hoàn toàn bằng Java và cho phép sử dụng với nhiều loại nguồn dữ liệu và tạo ra những báo cáo cho phép hiển thị, in ấn hoặc xuất ra những định dạng như HTML, PDF, Excel, Word,...

Hầu hết các ứng dụng trong thực tế cần phải xuất các báo cáo theo nghiệp vụ cũng như theo yêu cầu của khách hàng. Vì vậy, JasperReports sẽ là một công cụ tạo báo cáo đơn giản, mạnh mẽ sẽ giúp tiết kiệm được nhiều công sức, thời gian. Các báo cáo được thiết kế bằng một phần mềm cung cấp riêng để tạo báo cáo có tên JasperSoft Studio.

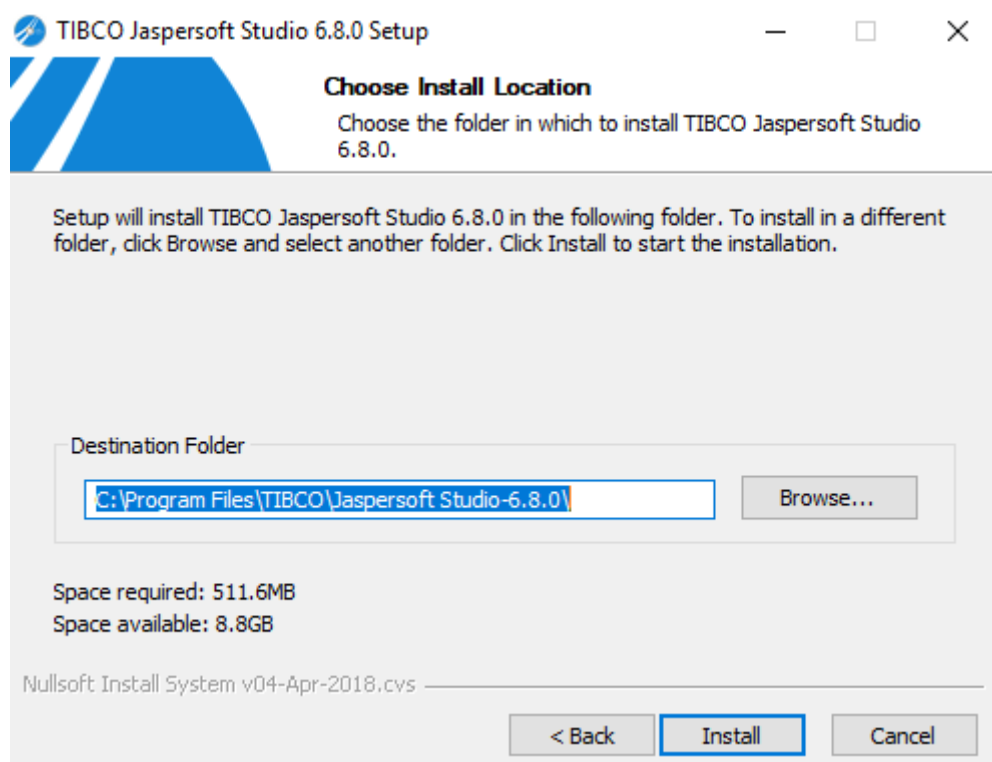
a. Cài đặt JasperSoft Studio

Bước 1: Chọn I Agree.



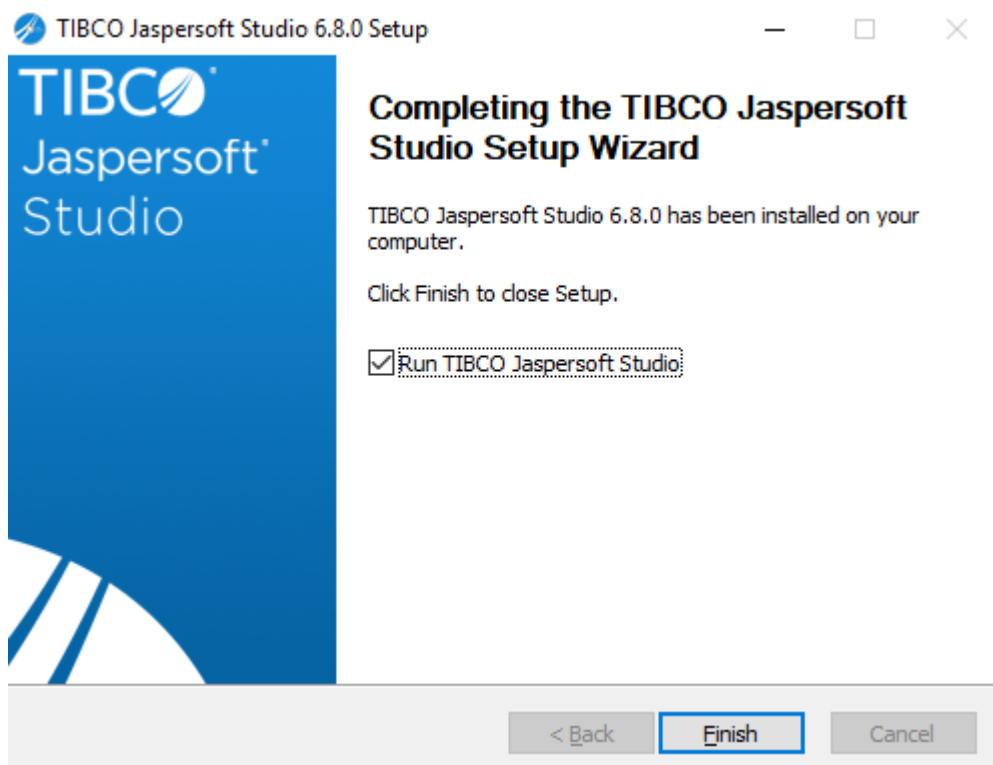
Hình 2.8. Các bước cài đặt JasperSoft Studio

Bước 2: Chọn thư mục để cài phần mềm → Nhấn Install.



Hình 2.9. Các bước cài đặt JasperSoft Studio

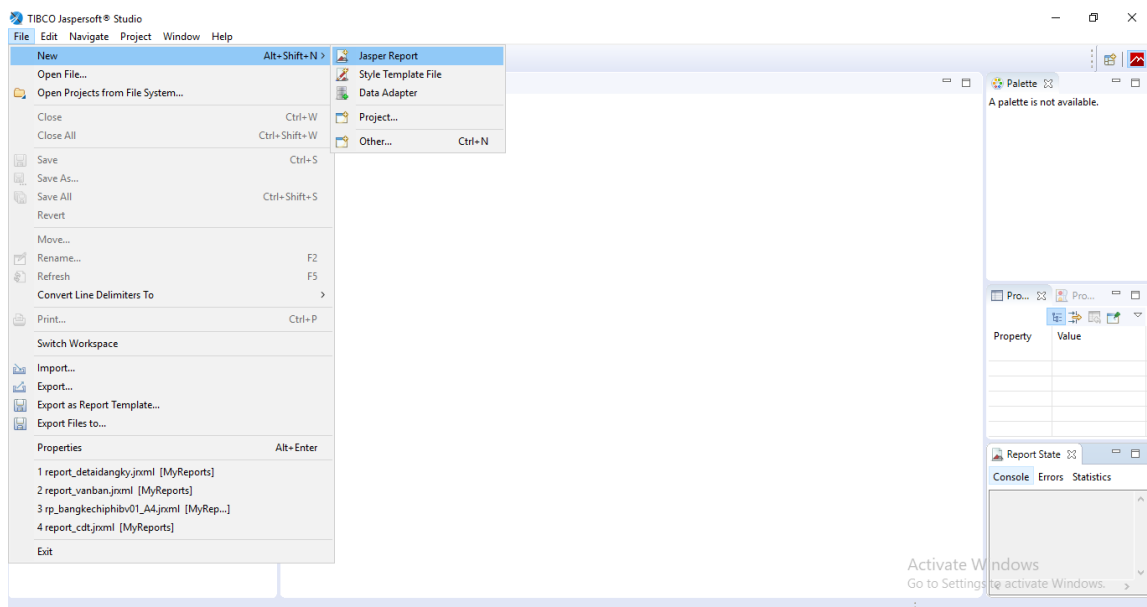
Bước 3: Sau khi phần mềm cài đặt xong → Chọn Finish.



Hình 2.10. Các bước cài đặt JasperSoft Studio

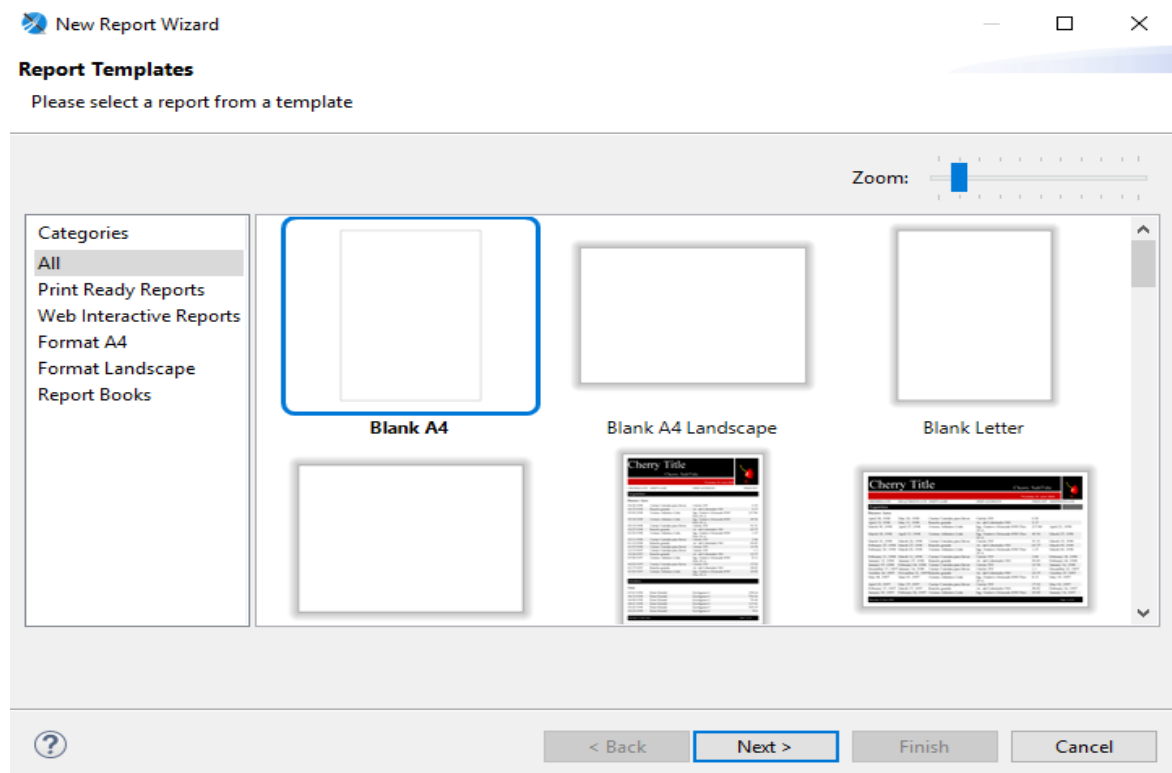
b. Tạo báo cáo bằng JasperSoft Studio

Bước 1: Chọn File → New → Jasper Report.



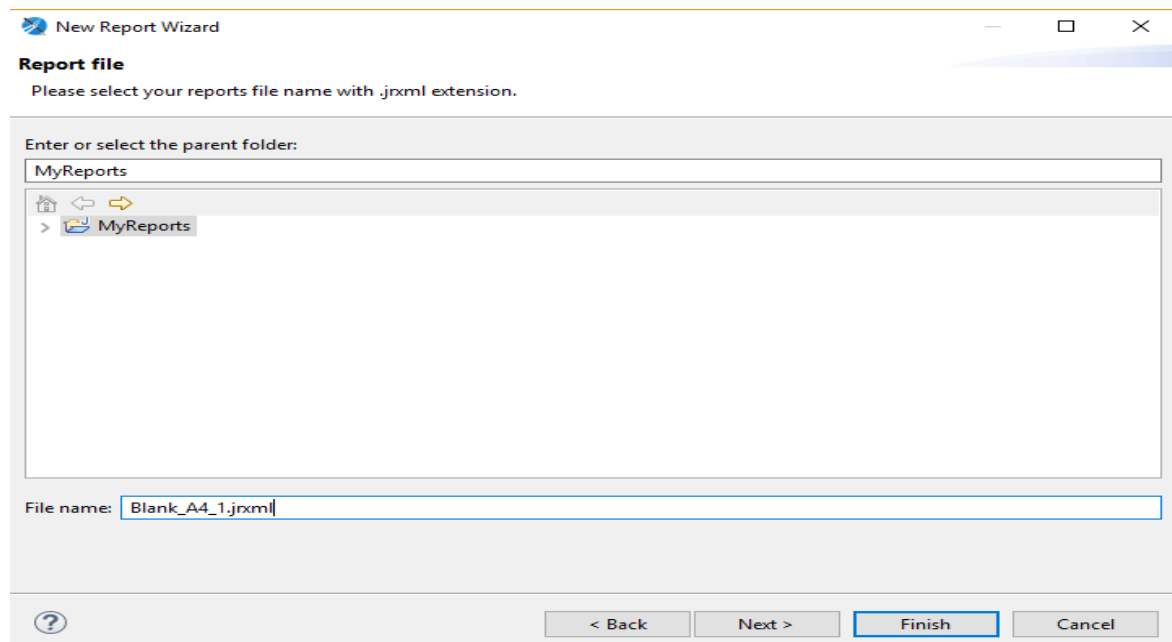
Hình 2.11. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 2: Chọn vào loại khung cho Report. Jasper hỗ trợ rất nhiều loại, ở đây chọn loại khổ A4 giấy đứng → Nhấn Next.



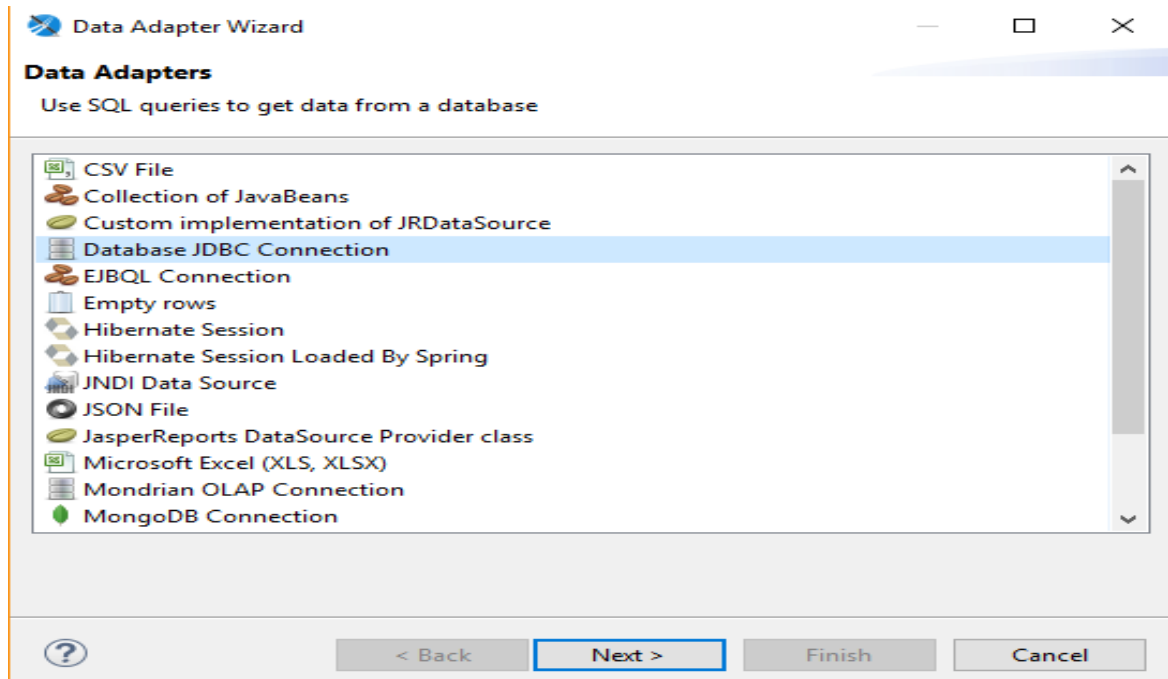
Hình 2.12. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 3: Chọn thư mục lưu trữ và đặt tên cho báo cáo → Nhấn Next.



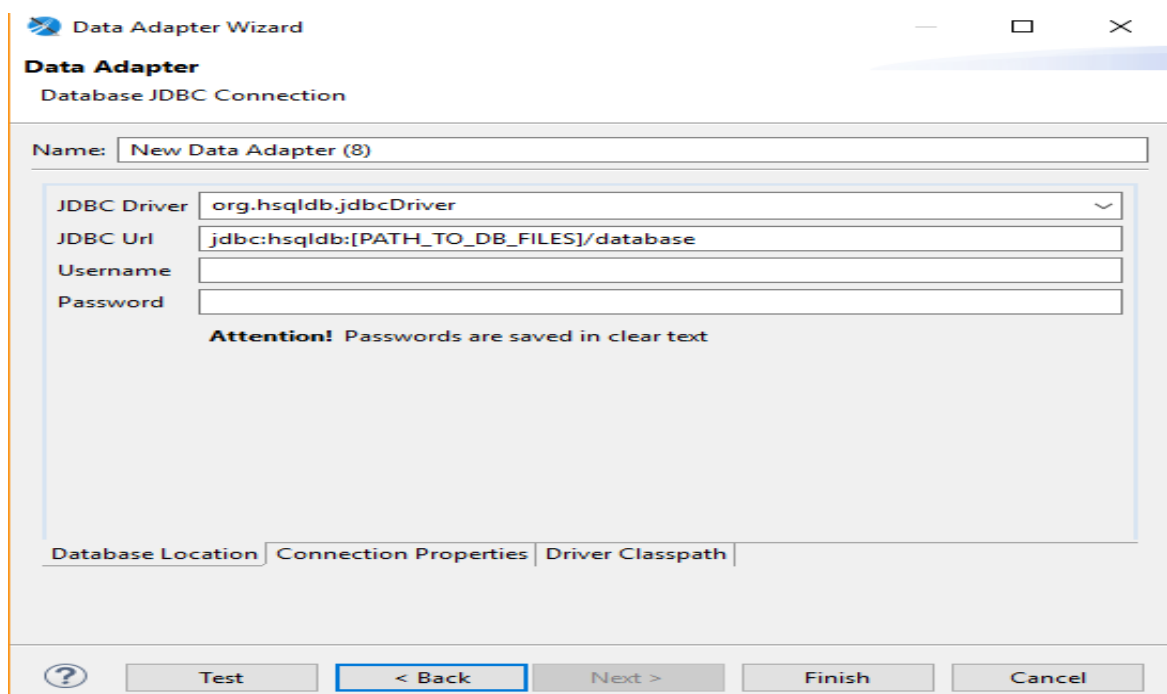
Hình 2.13. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 4: Chọn kết nối tới CSDL bằng cách: Chọn New → Chọn loại kết nối CSDL. Jasper hỗ trợ rất nhiều kết nối đến CSDL, ở đây chọn Database JDBC Connection → Nhấn Next.



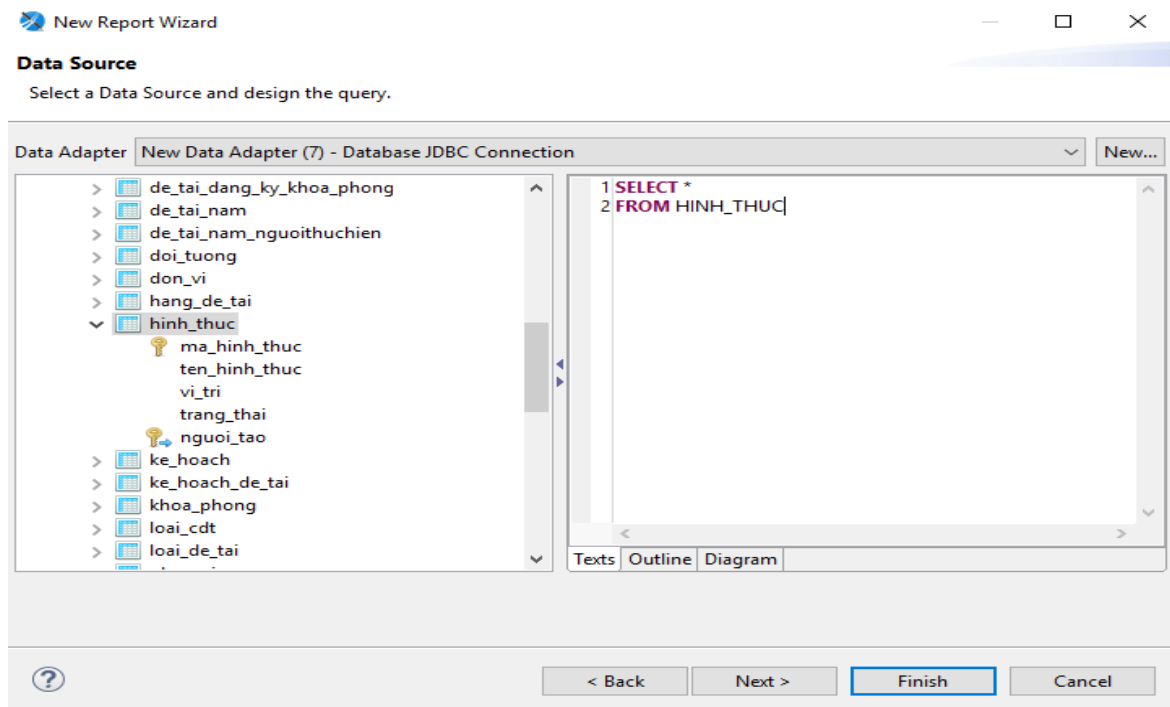
Hình 2.14. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 5: Nhập các thông tin kết nối đến CSDL, nhấn Test để kiểm tra kết nối CSDL → Nhấn Finish.



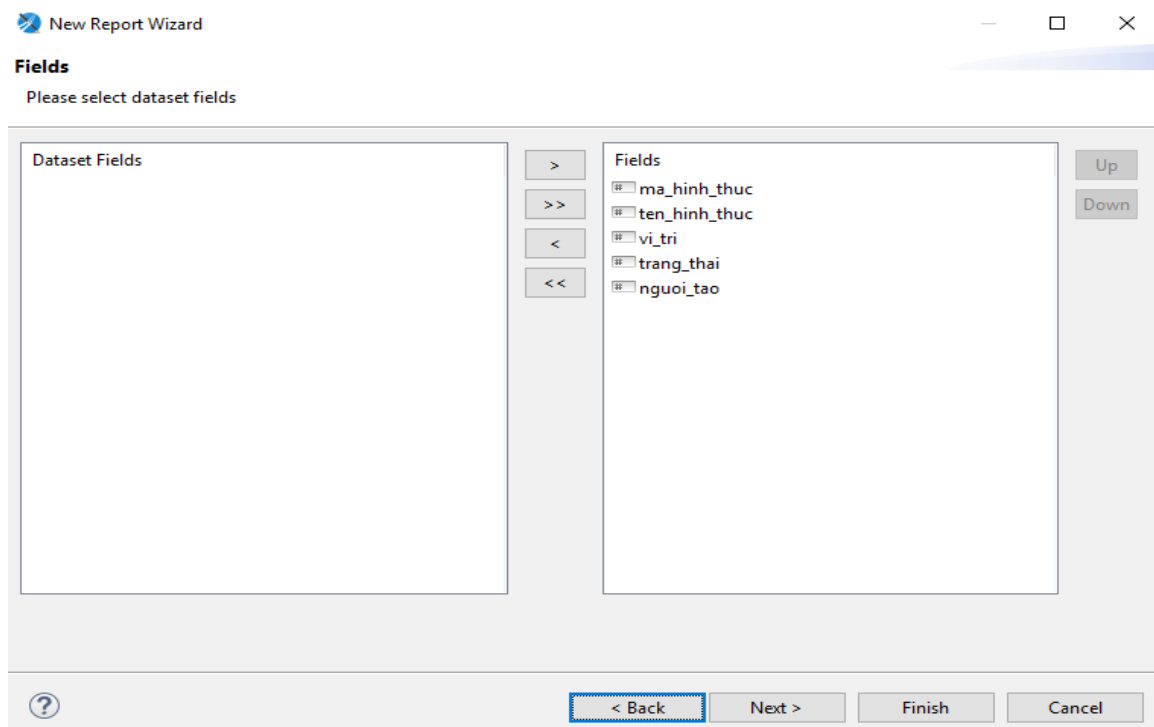
Hình 2.15. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 6: Nhập câu lệnh truy vấn → Nhấn Next.



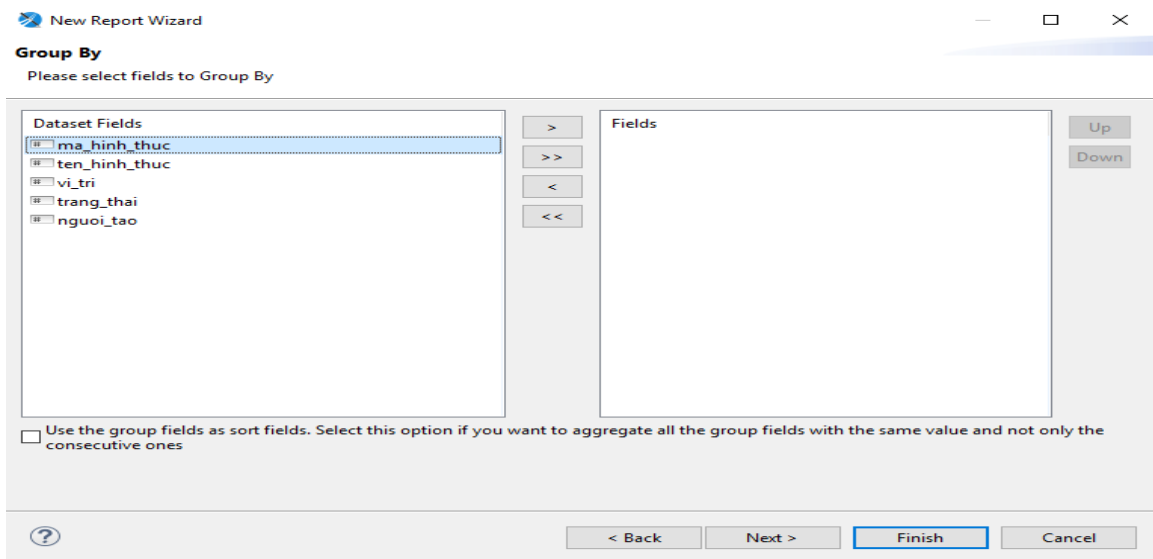
Hình 2.16. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 7: Chọn các trường dữ liệu sẽ tạo báo cáo → Nhấn > → Nhấn Next.



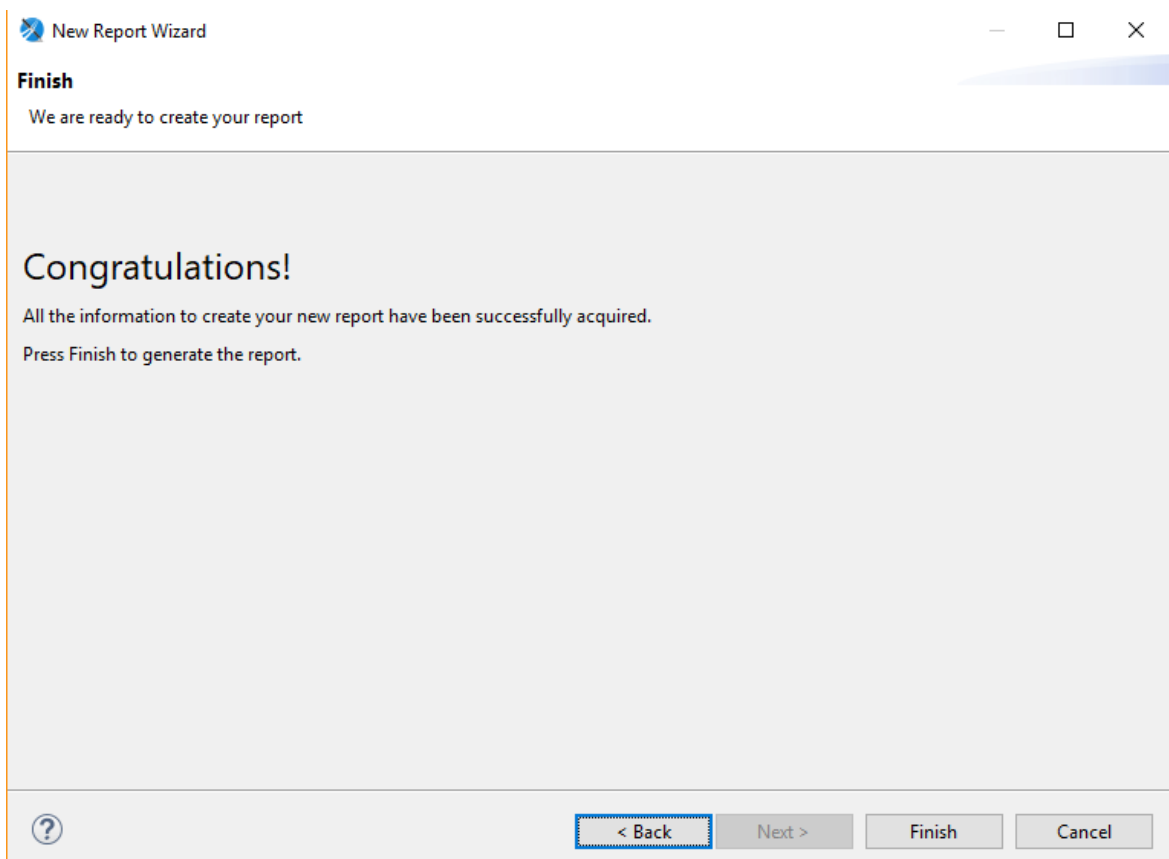
Hình 2.17. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 8: Chọn phân nhóm cho báo cáo, nếu không có phân nhóm thì bỏ qua → Nhấn Next.



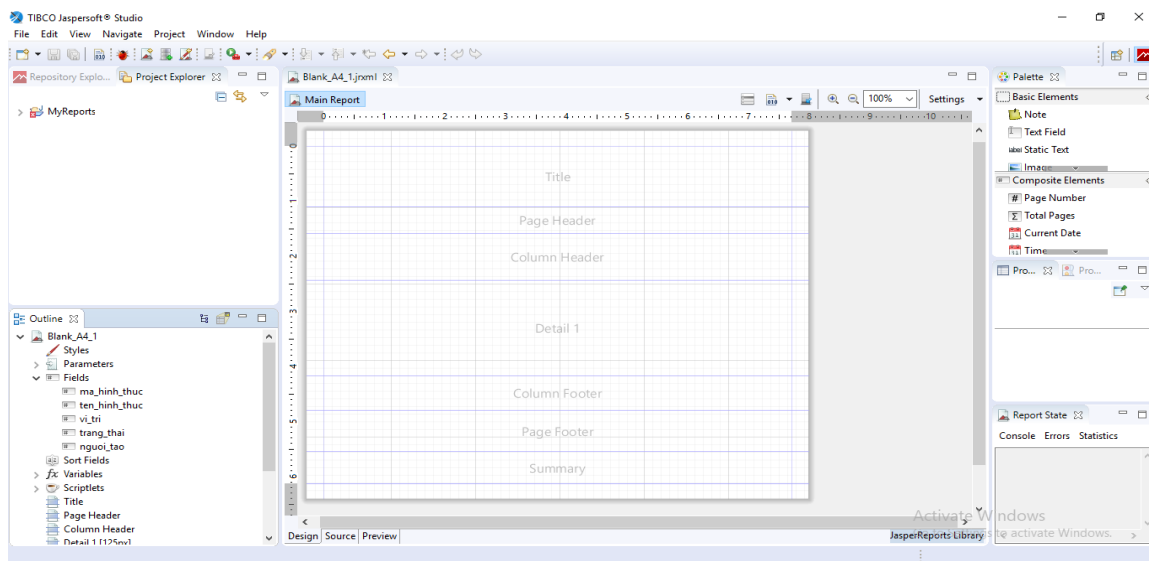
Hình 2.18. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 9: Nhấn Finish để hoàn tất tạo báo cáo



Hình 2.19. Các bước tạo báo cáo bằng JasperSoft Studio

Bước 10: Sau khi xong các bước trên, giao diện sẽ như bên dưới, bắt đầu tiến hành thiết kế Report bằng cách chọn các điều khiển ở khung Palette



Hình 2.20. Các bước tạo báo cáo bằng JasperSoft Studio

Các điều khiển cơ bản:

Static Text: Văn bản tĩnh

Text Field: Trường văn bản, chứa văn bản có thể sử dụng mã lệnh đối với văn bản.

Image: Hình ảnh, dùng để hiển thị hình ảnh trong báo cáo.

Note: Ghi chú thích của người thiết kế.

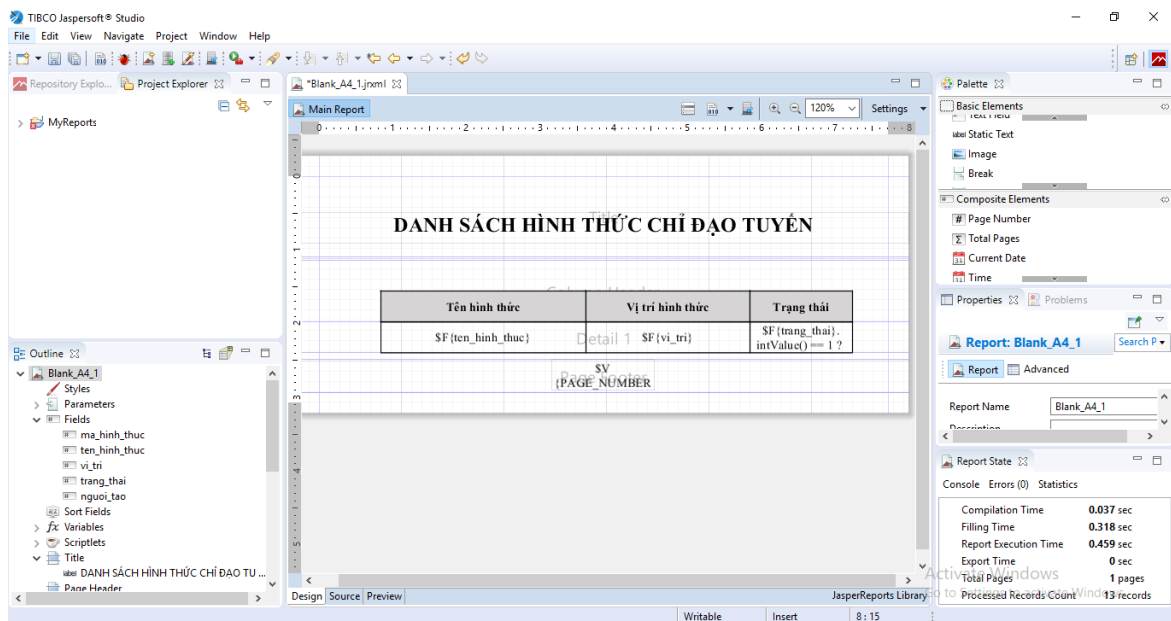
Page Number: Là một Text Field có sẵn mã lệnh hiển thị số trang.

Total Page: Là một Text Field có sẵn mã lệnh hiển thị tổng số trang của báo cáo.

Current Date: Là một Text Field có sẵn mã lệnh hiển thị thời gian hiện tại.

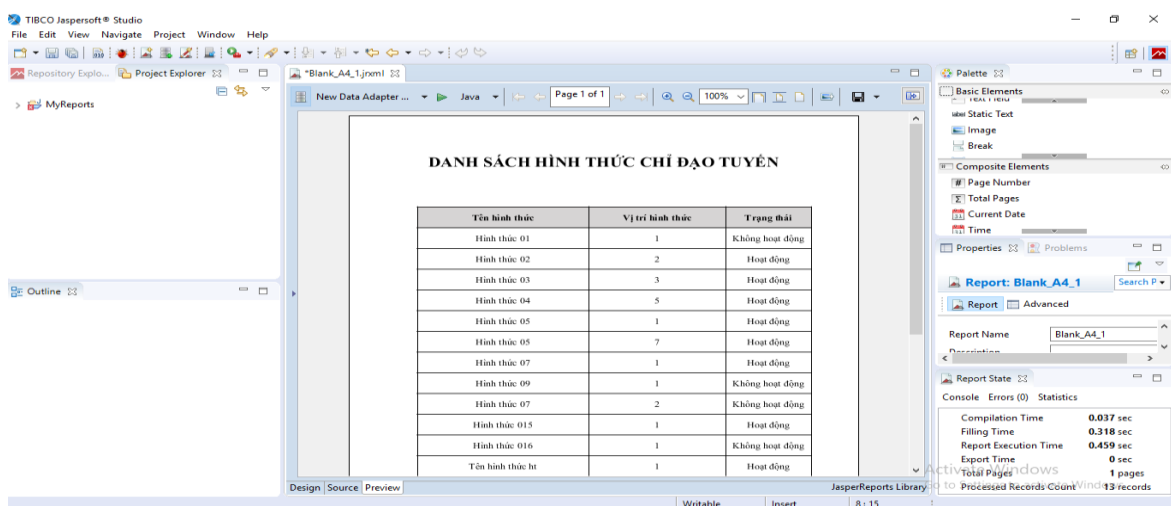
Ngoài những điều khiển cơ bản để thiết kế báo cáo, Jasper Reports cũng cho phép người dùng tạo ra các biến giá trị(Variables), các tham số truyền vào(Params), Fields(Các trường dữ liệu, mặc định sẽ là những trường dữ liệu mà người dùng đã lấy từ câu truy vấn ở bước 6).

Bước 11: Sau khi thiết kế xong, giao diện sẽ như bên dưới → Nhấn Preview để xem kết quả.



Hình 2.21. Các bước tạo báo cáo bằng JasperSoft Studio

Kết quả:



Hình 2.22. Các bước tạo báo cáo bằng JasperSoft Studio

❖ Giới thiệu về Spring

a. Giới thiệu về Spring trong Java

Spring là một Framework phát triển các ứng dụng. Nó giúp tạo các ứng dụng có hiệu năng cao, dễ kiểm thử, sử dụng lại code...

Spring là một mã nguồn mở, được phát triển, chia sẻ và có cộng đồng người dùng rất lớn.

Spring được chia làm nhiều module khác nhau, tùy theo mục đích phát triển ứng dụng mà ta dùng 1 trong các module đó.

Trong số các module của Spring, em sẽ sử dụng module Spring JDBC để xây dựng kiến trúc ứng dụng.[5]

b. Giới thiệu về Spring JDBC

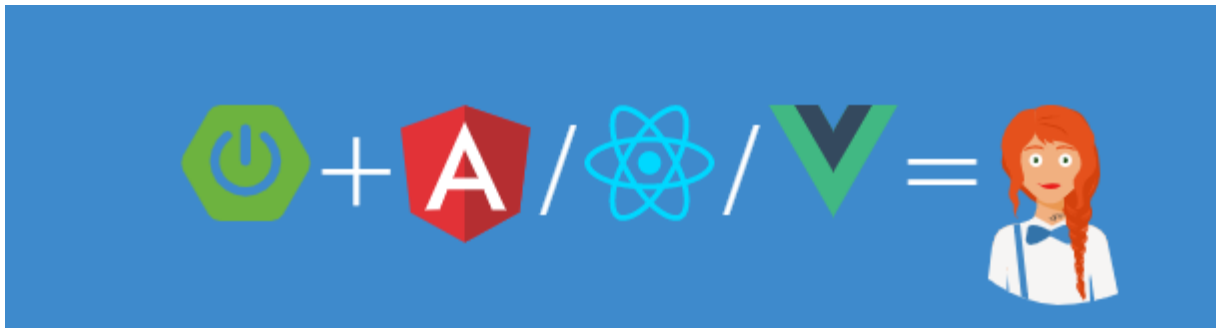
JDBC là công cụ mạnh mẽ để truy xuất dữ liệu quan hệ. Nếu không cần phải dùng đến một công cụ ánh xạ dữ liệu, cần truy xuất nhanh chóng đến CSDL, nếu ưu tiên tốc độ hơn là sự tiện dụng, JDBC sẽ là sự lựa chọn thay cho JPA/Hibernate.

JDBCTemplate: Là một lớp trung tâm trong Spring JDBC, nó bao gồm các luận lý phổ biến nhất trong việc sử dụng JDBC API để truy cập vào cơ sở dữ liệu, chẳng hạn như tạo kết nối, tạo các lệnh truy vấn, xóa, sửa, cập nhập dữ liệu... Lớp này có thể tìm thấy trong gói `org.springframework.jdbc.core`.

❖ Sử dụng JHipster để tạo nhanh mô hình Microservice

a. Giới thiệu JHipster

JHipster, viết tắt của Java Hipster là một framework được phát triển bởi Yeoman và được sử dụng để phát triển các dự án Spring Boot + ReactJS hoặc AngularJS.



Hình 2.23. Minh họa JHipster[6]

b. Yêu cầu hệ thống

Cài đặt đầy đủ thư viện cho JHipster, bao gồm:

Cài đặt và cấu hình Java 8 trên máy tính

Cài đặt NodeJS từ trang chủ

Cài Yeoman bằng cách vào command prompt và nhập `npm install -g yo`

c. Khởi tạo dự án

Tất cả câu lệnh đều nhập thủ công trên command prompt của máy tính. Để tạo dự án JHipster, thực hiện :

– Tạo một thư mục trống là nơi sẽ chứa dự án bằng lệnh: `mkdir myapplication`

– Chuyển đến thư mục vừa tạo bằng lệnh: `cd myapplication`

– Tạo tự động ứng dụng JHipster bằng lệnh: `jhipster`

– Sau đó, lựa chọn những thứ phù hợp với dự án của người dùng.

Bây giờ người dùng đã có một dự án với:

– Backend: Spring Boot + Spring Security

– Database:

+ Mysql (production)

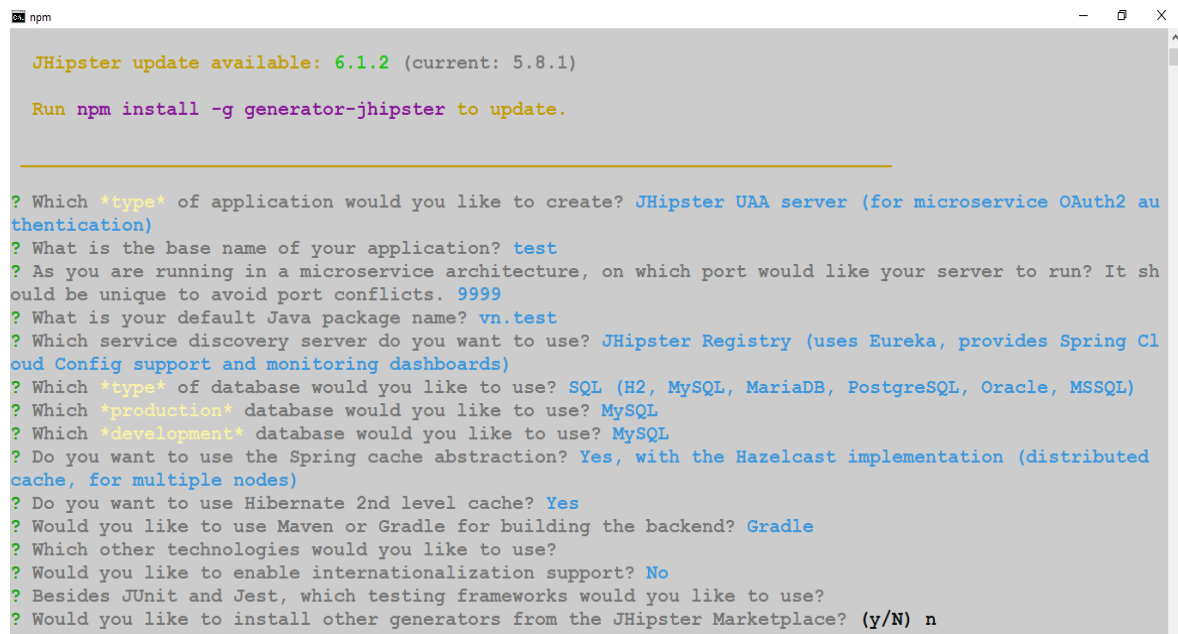
+ H2 with disk-based (development)

– Frontend: Bootstrap + SASS + React (hoặc thấp hơn tùy vào phiên bản JHipster người dùng cài đặt ở trên).

Sau khi chạy xong người dùng sẽ nhận được kết quả như bên dưới. Đó cũng là hướng dẫn để người dùng xây dựng dự án của mình trên local.

Chạy ứng dụng Spring Boot bằng cách nhập: `./mvnw`

Sau đó dùng lệnh để chạy: `npmstart[6]`



```
JHipster update available: 6.1.2 (current: 5.8.1)

Run npm install -g generator-jhipster to update.

? Which *type* of application would you like to create? JHipster UAA server (for microservice OAuth2 authentication)
? What is the base name of your application? test
? As you are running in a microservice architecture, on which port would like your server to run? It should be unique to avoid port conflicts. 9999
? What is your default Java package name? vn.test
? Which service discovery server do you want to use? JHipster Registry (uses Eureka, provides Spring Cloud Config support and monitoring dashboards)
? Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? Which *production* database would you like to use? MySQL
? Which *development* database would you like to use? MySQL
? Do you want to use the Spring cache abstraction? Yes, with the Hazelcast implementation (distributed cache, for multiple nodes)
? Do you want to use Hibernate 2nd level cache? Yes
? Would you like to use Maven or Gradle for building the backend? Gradle
? Which other technologies would you like to use?
? Would you like to enable internationalization support? No
? Besides JUnit and Jest, which testing frameworks would you like to use?
? Would you like to install other generators from the JHipster Marketplace? (y/N) n
```

Hình 2.24. Minh họa tạo dự án JHipster

2.2. Cơ sở thực tiễn và ứng dụng của khóa luận

2.2.1. Bối cảnh thực tiễn

❖ Bối cảnh thực tiễn

Internet được phát triển rộng khắp trên thế giới, và ngày 19 tháng 11 năm 1997 là ngày đầu Việt Nam được hòa vào mạng Internet toàn cầu. Internet xuất hiện cũng đã làm xuất hiện nhu cầu về lập trình web, các nền tảng lập trình web ban đầu cũng được hình thành, như HTML, CSS,... Qua nhiều năm phát triển không ngừng, lập trình web đã có nhiều bước tiến đáng kể và được sử dụng ngày càng nhiều và yêu cầu đặt ra cũng tăng cao, các công nghệ lập trình web cũ dần trở nên lỗi thời hoặc không còn đảm bảo nữa. Việc sử dụng công nghệ cũ trong lập trình web tiềm ẩn nhiều rủi ro, tính bảo mật chưa cao, nên việc sử dụng công nghệ mới trong lập trình web là rất cần thiết. Chính vì vậy, việc sử dụng công nghệ mới, trong đó có Microservice sẽ mang lại rất nhiều lợi ích.

❖ Nhu cầu thực tế

Cần phải có một hệ chỉ đạo tuyến cho ngành để kịp thời chuyển tuyến các chỉ đạo tuyến từ các bệnh viện tuyến trên xuống tuyến dưới một cách an toàn, bảo mật.

Cần phải có một hệ thống quản lý việc đăng ký, xét duyệt đề tài NCKH trong các đơn vị ngành y tế.

2.2.2. Ứng dụng

Khóa luận này là một tài liệu tham khảo có ích cho các sinh viên khi tìm hiểu về mô hình Microservices.

Sản phẩm phần mềm của khóa luận này có thể ứng dụng vào sử dụng ở một đơn vị ngành y tế.

Chương 3: PHÂN TÍCH, THIẾT KẾ, CÀI ĐẶT GIẢI PHÁP

3.1. Phân tích yêu cầu

3.1.1. Phân tích yêu cầu chức năng

Đầy đủ các thuộc tính của một chỉ đạo tuyến và một đề tài NCKH

Có phân quyền cho nhiều nhóm người sử dụng:

Phân hệ chỉ đạo tuyến:

- Cán bộ phụ trách chỉ đạo tuyến:

- + Quản lý thông tin về hình thức chỉ đạo tuyến, loại chỉ đạo tuyến, nội dung chỉ đạo tuyến, đối tượng tiếp nhận chỉ đạo tuyến, đánh giá chỉ đạo tuyến, văn bản chỉ đạo tuyến, kế hoạch chỉ đạo tuyến và chỉ đạo tuyến.

- + Thống kê các văn bản chỉ đạo tuyến, kế hoạch chỉ đạo tuyến, và chỉ đạo tuyến hàng năm.

- Nhân viên của đơn vị tuyến dưới:

- Xem thông tin chỉ đạo tuyến tiếp nhận của đơn vị.

- Phân hệ NCKH:

- + Nhân viên quản lý NCKH

- Quản lý loại đề tài, hạng đề tài, chức vụ hội đồng, chủ thể đề tài, nhóm đề tài, kế hoạch đề tài.

- + Nhân viên phòng NCKH của đơn vị

- Quản lý đề tài đăng ký tại đơn vị.

- Xét duyệt tên đề tài đã đăng ký để trình lên hội đồng.

- Thống kê đề tài đăng ký hàng năm.

- + Cán bộ hội đồng NCKH

- Xét duyệt đề cương, xét duyệt báo cáo tổng kết, xét duyệt công nhận đề tài năm, xét duyệt đề tài ứng dụng tại đơn vị.

- Thống kê đề tài đăng ký, thống kê đề tài công nhận đề tài năm, đề tài ứng dụng tại đơn vị hàng năm.

- + Cán bộ đơn vị NCKH

- Cập nhật biên bản ứng dụng, cập nhật báo cáo quý năm, cập nhật văn bản áp dụng đề tài tại đơn vị.

- . Thống kê đề tài đăng ký, thống kê đề tài công nhận đề tài năm, đề tài ứng dụng tại đơn vị hàng năm.

3.1.2. Phân tích yêu cầu phi chức năng

Giao diện: trực quan, đơn giản, dễ sử dụng, phù hợp với người không rành về Công nghệ thông tin.

Bảo mật: Không bị khai thác lỗi.

Có thể hoạt động trên các trình duyệt cơ bản trên máy tính.

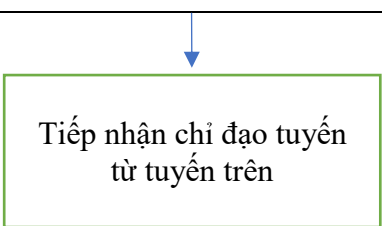
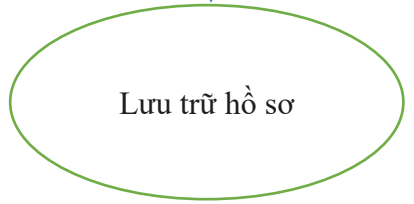
Chạy nhanh, ổn định.

3.1.3. Phân tích quy trình

❖ Phân tích quy trình chỉ đạo tuyển

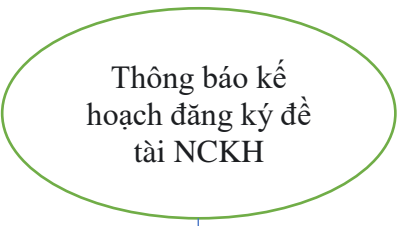
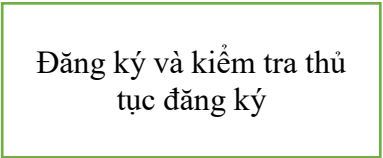
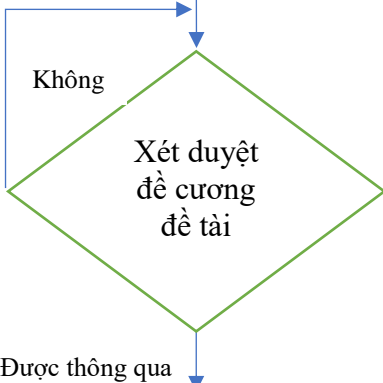
Quy trình chỉ đạo tuyển được mô tả thông qua bảng bên dưới.

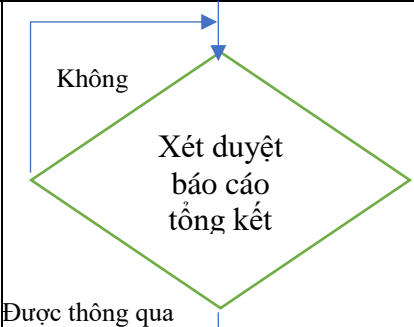
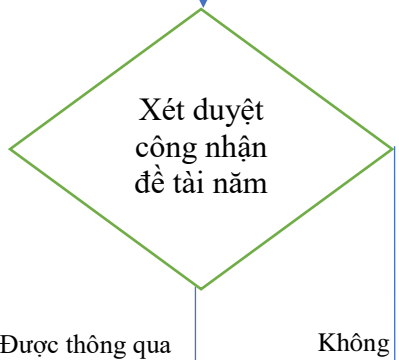
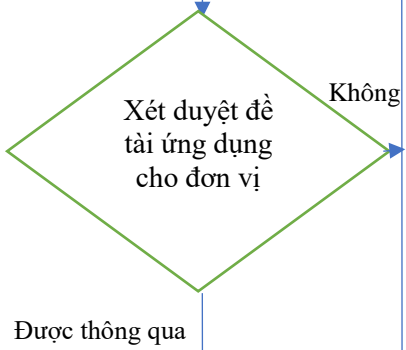
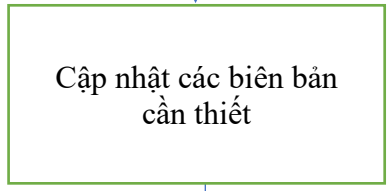
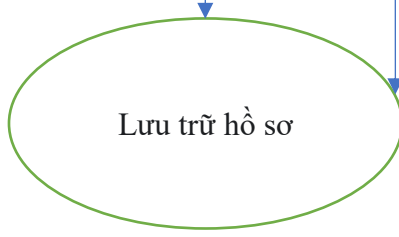
Trách nhiệm	Nội dung quy trình	Mô tả, biểu mẫu
Trưởng bộ phận thực hiện chỉ đạo tuyển	<pre> graph TD A([Xây dựng kế hoạch chỉ đạo tuyển]) --> B[Xây dựng chỉ đạo tuyển] B --> C[Chuyển tuyển chỉ đạo tuyển xuống tuyến dưới] </pre>	Theo chỉ đạo của giám đốc bệnh viện, các trưởng khoa, trưởng phòng và trưởng phòng kế hoạch tổng hợp xây dựng kế hoạch chỉ đạo tuyển.
Bộ phận thực hiện chỉ đạo tuyển	<pre> graph TD A([Xây dựng kế hoạch chỉ đạo tuyển]) --> B[Xây dựng chỉ đạo tuyển] B --> C[Chuyển tuyển chỉ đạo tuyển xuống tuyến dưới] </pre>	Các cán bộ được phân công xây dựng chỉ đạo tuyển, bao gồm: hình thức, loại, nội dung, đối tượng tiếp nhận, đánh giá, văn bản chỉ đạo tuyển.
Bộ phận thực hiện chỉ đạo tuyển	<pre> graph TD A([Xây dựng kế hoạch chỉ đạo tuyển]) --> B[Xây dựng chỉ đạo tuyển] B --> C[Chuyển tuyển chỉ đạo tuyển xuống tuyến dưới] </pre>	Các cán bộ được phân công thực hiện chuyển tuyển chỉ đạo tuyển xuống các tuyến dưới.

Trách nhiệm	Nội dung quy trình	Mô tả, biểu mẫu
Cán bộ bệnh viện tiếp nhận được phân công	 <p>Tiếp nhận chỉ đạo tuyến từ tuyến trên</p>	Các cán bộ được phân công thực hiện nhận chỉ đạo tuyến từ tuyến trên và thực hiện theo nội dung chỉ đạo tuyến.
Cán bộ bệnh viện tiếp nhận được phân công	 <p>Lưu trữ hồ sơ</p>	Các cán bộ được phân công lưu trữ hồ sơ theo từng năm

❖ **Phân tích quy trình đăng ký, xét duyệt đề tài NCKH**

Quy trình đăng ký, xét duyệt NCKH được mô tả thông qua bảng bên dưới.

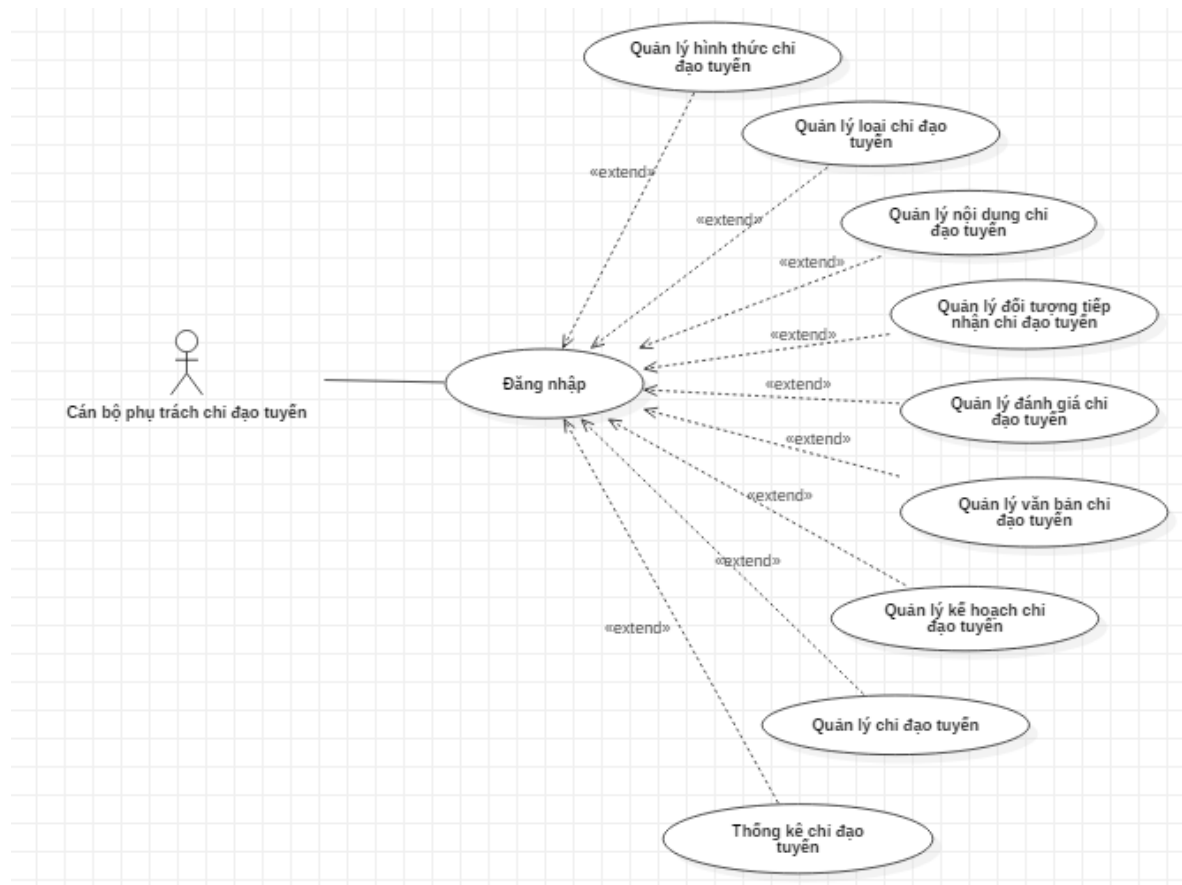
Trách nhiệm	Nội dung quy trình	Mô tả, biểu mẫu
Hội đồng NCKH	 <p>Thông báo kế hoạch đăng ký đề tài NCKH</p>	<ul style="list-style-type: none"> - Gửi công văn cho các đơn vị về nội dung kế hoạch đăng ký NCKH, thông báo thời hạn đăng ký; - Cung cấp các biểu mẫu trên trang web của bệnh viện.
Thư ký phòng NCKH các đơn vị	 <p>Đăng ký và kiểm tra thủ tục đăng ký</p>	<ul style="list-style-type: none"> - Các đơn vị đăng ký đề tài qua phần mềm quản lý NCKH tại đơn vị; - Cán bộ kiểm tra thông tin đăng ký đề tài của các đơn vị.
Hội đồng NCKH	 <p>Xét duyệt đề cương đề tài</p> <p>Không</p> <p>Được thông qua</p>	<ul style="list-style-type: none"> - Cán bộ xét duyệt đề cương của đề tài đã đăng ký của các đơn vị. - Nếu xét duyệt đề cương là không duyệt thì sẽ phải làm lại đề cương và xét duyệt lại.

Trách nhiệm	Nội dung quy trình	Mô tả, biểu mẫu
Hội đồng NCKH		<ul style="list-style-type: none"> - Cán bộ xét duyệt báo cáo tổng kết của đề tài đã đăng ký của các đơn vị. - Nếu xét duyệt báo cáo tổng kết là không duyệt thì sẽ phải làm lại báo cáo tổng kết và xét duyệt lại.
Hội đồng NCKH		<ul style="list-style-type: none"> - Cán bộ xét duyệt đề tài để công nhận đề tài năm; - Nếu xét duyệt công nhận là không duyệt thì đề tài sẽ được lưu trữ hồ sơ.
Chủ tịch hội đồng NCKH đề xuất theo yêu cầu của đơn vị		<ul style="list-style-type: none"> - Cán bộ xét duyệt đề tài để ứng dụng vào hoạt động của các đơn vị; - Nếu xét duyệt của đề tài là không cho phép thì đề tài sẽ được lưu trữ hồ sơ.
Hội đồng NCKH; Bộ phận thụ hưởng đề tài		<ul style="list-style-type: none"> - Cán bộ cập nhật các biên bản cần thiết, bao gồm: Biên bản kế hoạch ứng dụng đề tài, báo cáo quý năm, văn bản áp dụng đề tài lâu dài.
Thư ký hội đồng NCKH		Các thư ký hội đồng NCKH lưu trữ hồ sơ theo từng năm.

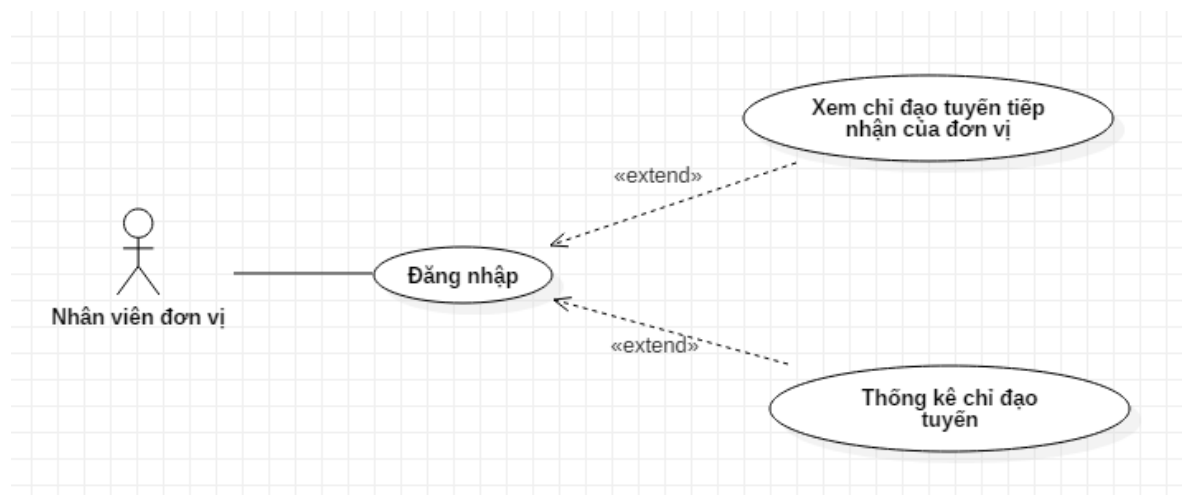
3.3. Phân tích thiết kế hệ thống

3.3.1. Hệ thống chỉ đạo tuyến

❖ Sơ đồ Use case tổng quát



Hình 3.1. Sơ đồ Use case tổng quát (cán bộ phụ trách chỉ đạo tuyến)



Hình 3.2. Sơ đồ Use case tổng quát (nhân viên đơn vị tiếp nhận chỉ đạo tuyến)

❖ **Đặc tả Use case**

Use case Đăng nhập

Mã use case	UC01
Tên use case	Đăng nhập
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyển, nhân viên đơn vị
Tiền điều kiện	Không có
Hậu điều kiện	Đăng nhập thành công vào hệ thống
Điều kiện tối thiểu	Yêu cầu nhân viên quản lý đăng nhập lại
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình đăng nhập 2. Nhân viên quản lý nhập thông tin tài khoản và mật khẩu 3. Nhân viên quản lý nhấn “Đăng nhập” 4. Hệ thống chứng thực tài khoản 5. Hệ thống thông báo đăng nhập thành công
Ngoại lệ	4.1. Hệ thống chứng thực thất bại, yêu cầu nhân viên nhập lại thông tin tài khoản, mật khẩu.

Use case Quản lý hình thức chỉ đạo tuyển

Mã use case	UC02
Tên use case	Quản lý hình thức chỉ đạo tuyển.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyển.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Cập nhật thành công hình thức chỉ đạo tuyển.
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước.
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý hình thức chỉ đạo tuyển. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Cán bộ phụ trách nhập thông tin cần thiết của một hình thức chỉ đạo tuyển. 4. Cán bộ phụ trách xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin hình thức chỉ đạo tuyển. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý hình thức chỉ đạo tuyển.

Use case Quản lý loại chỉ đạo tuyến

Mã use case	UC03
Tên use case	Quản lý loại chỉ đạo tuyến.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyến.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Cập nhật thành công loại chỉ đạo tuyến.
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước.
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý loại chỉ đạo tuyến. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Cán bộ phụ trách nhập thông tin cần thiết của một loại chỉ đạo tuyến. 4. Cán bộ phụ trách xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin loại chỉ đạo tuyến. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	4.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý loại chỉ đạo tuyến.

Use case Quản lý nội dung chỉ đạo tuyến

Mã use case	UC04
Tên use case	Quản lý nội dung chỉ đạo tuyến.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyến.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Cập nhật thành công nội dung chỉ đạo tuyến.
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước.
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý nội dung chỉ đạo tuyến. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Cán bộ phụ trách nhập thông tin cần thiết của một nội dung chỉ đạo tuyến. 4. Cán bộ phụ trách xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin nội dung chỉ đạo tuyến. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý nội dung chỉ đạo tuyến.

Use case Quản lý đối tượng tiếp nhận chỉ đạo tuyến

Mã use case	UC05
Tên use case	Quản lý đối tượng tiếp nhận chỉ đạo tuyến.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyến.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Cập nhật thành công đối tượng tiếp nhận chỉ đạo tuyến.
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước.
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý đối tượng tiếp nhận chỉ đạo tuyến. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Cán bộ phụ trách nhập thông tin cần thiết của một đối tượng tiếp nhận chỉ đạo tuyến. 4. Cán bộ phụ trách xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin đối tượng tiếp nhận. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	4.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý đối tượng tiếp nhận chỉ đạo tuyến.

Use case Quản lý đánh giá chỉ đạo tuyến

Mã use case	UC06
Tên use case	Quản lý đánh giá chỉ đạo tuyến.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyến.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Cập nhật thành công đánh giá chỉ đạo tuyến.
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước.
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý đánh giá chỉ đạo tuyến. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Cán bộ phụ trách nhập thông tin cần thiết của một đánh giá chỉ đạo tuyến. 4. Cán bộ phụ trách xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin đánh giá chỉ đạo tuyến. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	4.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý đánh giá chỉ đạo tuyến.

Use case Quản lý văn bản chỉ đạo tuyến

Mã use case	UC07
Tên use case	Quản lý văn bản chỉ đạo tuyến.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyến.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Cập nhật thành công văn bản chỉ đạo tuyến.
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước.
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý văn bản chỉ đạo tuyến. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Cán bộ phụ trách nhập thông tin cần thiết của một văn bản chỉ đạo tuyến. 4. Cán bộ phụ trách xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin văn bản chỉ đạo tuyến. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	4.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý văn bản chỉ đạo tuyến.

Use case Quản lý kế hoạch chỉ đạo tuyến

Mã use case	UC08
Tên use case	Quản lý kế hoạch chỉ đạo tuyến.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyến.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Cập nhật thành công kế hoạch chỉ đạo tuyến.
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước.
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý kế hoạch chỉ đạo tuyến. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Cán bộ phụ trách nhập thông tin cần thiết của một kế hoạch chỉ đạo tuyến. 4. Cán bộ phụ trách xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin kế hoạch chỉ đạo tuyến. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	4.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý kế hoạch chỉ đạo tuyến.

Use case Quản lý chỉ đạo tuyển

Mã use case	UC09
Tên use case	Quản lý chỉ đạo tuyển.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyển.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Cập nhật thành công chỉ đạo tuyển.
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước.
Sự kiện chính	1. Hệ thống hiển thị màn hình quản lý chỉ đạo tuyển. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Cán bộ phụ trách nhập thông tin cần thiết của một chỉ đạo tuyển. 4. Cán bộ phụ trách xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin chỉ đạo tuyển. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	4.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý chỉ đạo tuyển.

Use case Thống kê chỉ đạo tuyển

Mã use case	UC10
Tên use case	Thống kê chỉ đạo tuyển.
Tác nhân chính	Cán bộ phụ trách chỉ đạo tuyển.
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Xuất thống kê chỉ đạo tuyển
Điều kiện tối thiểu	Không có dữ liệu để xuất
Sự kiện chính	1. Hệ thống hiển thị màn hình thống kê chỉ đạo tuyển. 2. Cán bộ phụ trách nhấn vào nút chức năng muốn thống kê 3. Hệ thống hiển thị kết quả thống kê
Ngoại lệ	Không có

❖ Các lớp

Hình thức

Lớp hình thức dùng để lưu thông tin về các hình thức chỉ đạo tuyển

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maHinhThuc	Kiểu số nguyên dương	Mỗi hình thức chỉ đạo tuyến có một mã riêng để phân biệt
2	tenHinhThuc	Kiểu chuỗi kí tự	Tên của hình thức chỉ đạo tuyến
3	viTri	Kiểu số nguyên dương	Vị trí ưu tiên của hình thức chỉ đạo tuyến
4	trangThai	Kiểu số nguyên	Trạng thái của hình thức chỉ đạo tuyến

Loại

Lớp loại dùng để lưu thông tin về các loại chỉ đạo tuyến

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maLoai	Kiểu số nguyên dương	Mỗi loại chỉ đạo tuyến có một mã riêng để phân biệt
2	tenLoai	Kiểu chuỗi kí tự	Tên của loại chỉ đạo tuyến

Nội dung

Lớp nội dung dùng để lưu thông tin về nội dung của chỉ đạo tuyến

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maNoiDung	Kiểu số nguyên dương	Mỗi nội dung chỉ đạo tuyến có một mã riêng để phân biệt
2	tenNoiDung	Kiểu chuỗi kí tự	Tên của nội dung chỉ đạo tuyến
3	viTri	Kiểu số nguyên dương	Vị trí ưu tiên của nội dung chỉ đạo tuyến
4	trangThai	Kiểu số nguyên	Trạng thái của nội dung chỉ đạo tuyến

Đối tượng

Lớp đối tượng dùng để lưu thông tin các đối tượng tiếp nhận của chỉ đạo tuyến

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maDoiTuong	Kiểu số nguyên dương	Mỗi đối tượng tiếp nhận chỉ đạo tuyến có một mã riêng để phân biệt
2	tenDoiTuong	Kiểu chuỗi kí tự	Tên của đối tượng tiếp nhận chỉ đạo tuyến
3	viTri	Kiểu số nguyên dương	Vị trí ưu tiên của đối tượng tiếp nhận chỉ đạo tuyến
4	trangThai	Kiểu số nguyên	Trạng thái của đối tượng tiếp nhận chỉ đạo tuyến

Đánh giá

Lớp đánh giá dùng để lưu thông tin về các đánh giá của chỉ đạo tuyến

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maDanhGia	Kiểu số nguyên dương	Mỗi đánh giá chỉ đạo tuyến có một mã riêng để phân biệt
2	tenDanhGia	Kiểu chuỗi kí tự	Tên của đánh giá chỉ đạo tuyến
3	viTri	Kiểu số nguyên dương	Vị trí ưu tiên của đánh giá chỉ đạo tuyến
4	trangThai	Kiểu số nguyên	Trạng thái của đánh giá chỉ đạo tuyến

Văn bản

Lớp văn bản dùng để lưu thông tin của các văn bản của chỉ đạo tuyến

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maVanBan	Kiểu số nguyên dương	Mỗi văn bản chỉ đạo tuyến có một mã riêng để phân biệt
2	noiDungTrichYeu	Kiểu chuỗi kí tự	Nội dung trích yếu của văn bản chỉ đạo tuyến

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
3	ngayBanHanh	Kiểu ngày tháng	Ngày ban hành văn bản chỉ đạo tuyển
4	soCongVan	Kiểu chuỗi kí tự	Số công văn của văn bản chỉ đạo tuyển
5	tapTin	Kiểu chuỗi kí tự	Tập tin văn bản chỉ đạo tuyển
6	trangThai	Kiểu số nguyên	Trạng thái của văn bản chỉ đạo tuyển

Kế hoạch

Lớp kế hoạch dùng để lưu thông tin về các kế hoạch của chỉ đạo tuyển

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maKeHoach	Kiểu số nguyên dương	Mỗi kế hoạch chỉ đạo tuyển có một mã riêng để phân biệt
2	noiDungTrichYeu	Kiểu chuỗi kí tự	Nội dung trích yếu của kế hoạch chỉ đạo tuyển
3	ngayBanHanh	Kiểu ngày tháng	Ngày ban hành kế hoạch chỉ đạo tuyển
4	tapTin	Kiểu chuỗi kí tự	Tập tin kế hoạch chỉ đạo tuyển

Chỉ đạo tuyển

Lớp chỉ đạo tuyển dùng để lưu thông tin của chỉ đạo tuyển

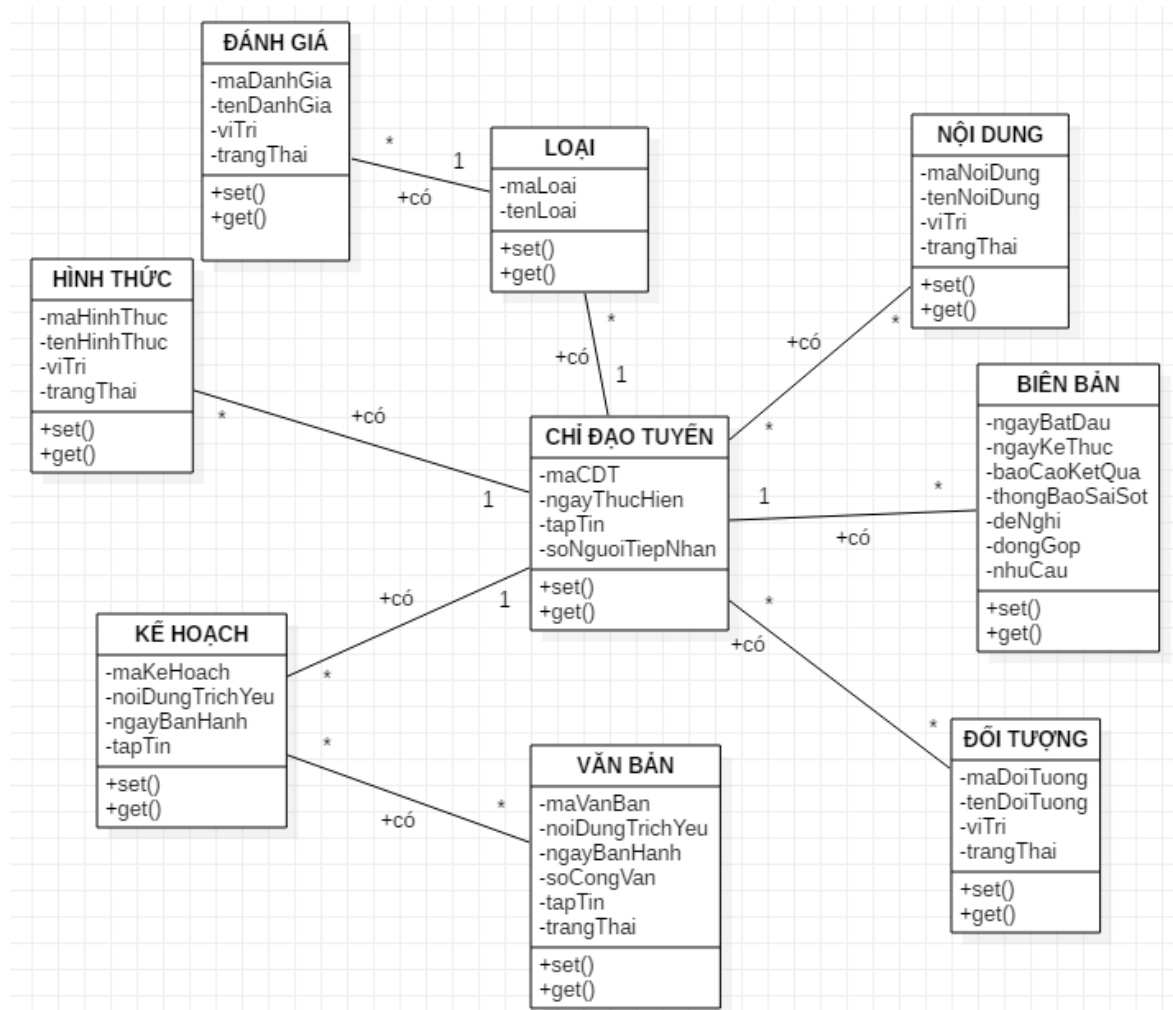
STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maCDT	Kiểu số nguyên dương	Mỗi chỉ đạo tuyển có một mã riêng để phân biệt
3	ngayThucHien	Kiểu ngày tháng	Ngày thực hiện chỉ đạo tuyển
4	tapTin	Kiểu chuỗi kí tự	Tập tin chỉ đạo tuyển
5	soNguoiTiepNhan	Kiểu số nguyên dương	Số người tiếp nhận chỉ đạo tuyển ở đơn vị

Biên bản

Lớp biên bản dùng để lưu thông tin của các biên bản của chỉ đạo tuyến

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	ngayBatDau	Kiểu số nguyên dương	Mỗi biên bản có một ngày bắt đầu
2	ngayKetThuc	Kiểu chuỗi kí tự	Mỗi biên bản có một ngày kết thúc
3	baoCaoKetQua	Kiểu số nguyên dương	Mỗi biên bản có một báo cáo kết quả
4	thongBaoSaiSot	Kiểu số nguyên	Mỗi biên bản có một thông báo sai sót
5	deNghị	Kiểu chuỗi kí tự	Mỗi biên bản có một đề nghị của đoàn chỉ đạo tuyến
6	dongGop	Kiểu chuỗi kí tự	Mỗi biên bản có một đóng góp của đoàn chỉ đạo tuyến
7	nhuCau	Kiểu chuỗi kí tự	Mỗi biên bản có một nhu cầu của đoàn chỉ đạo tuyến

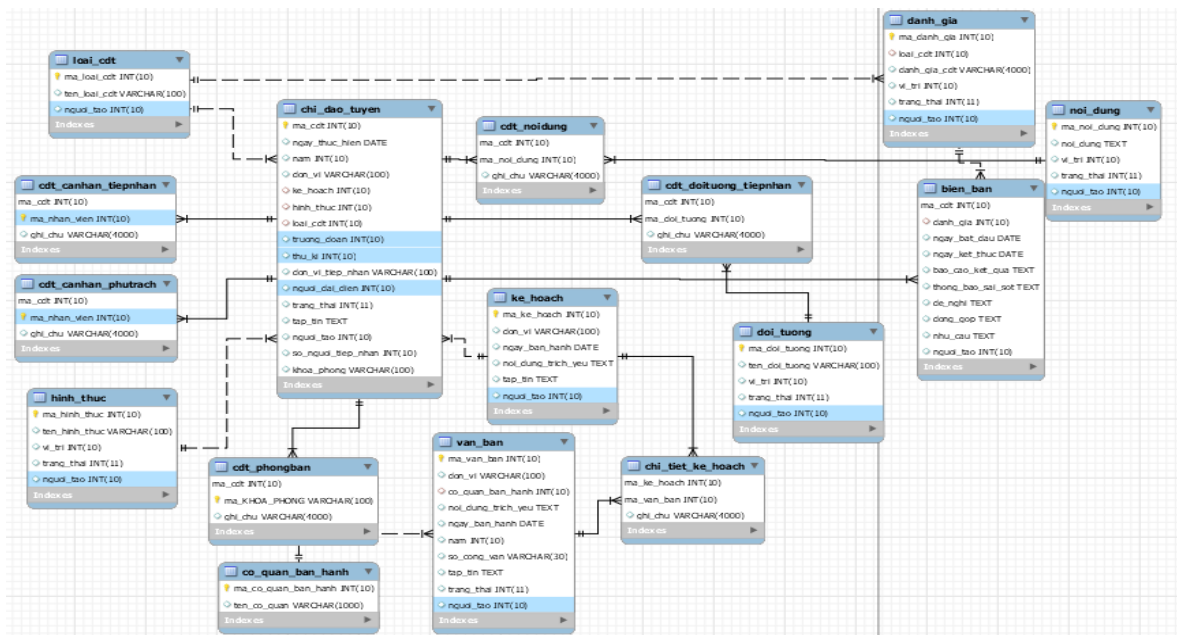
❖ Sơ đồ lớp



Hình 3.3. Sơ đồ lớp

❖ Mô hình cơ sở dữ liệu

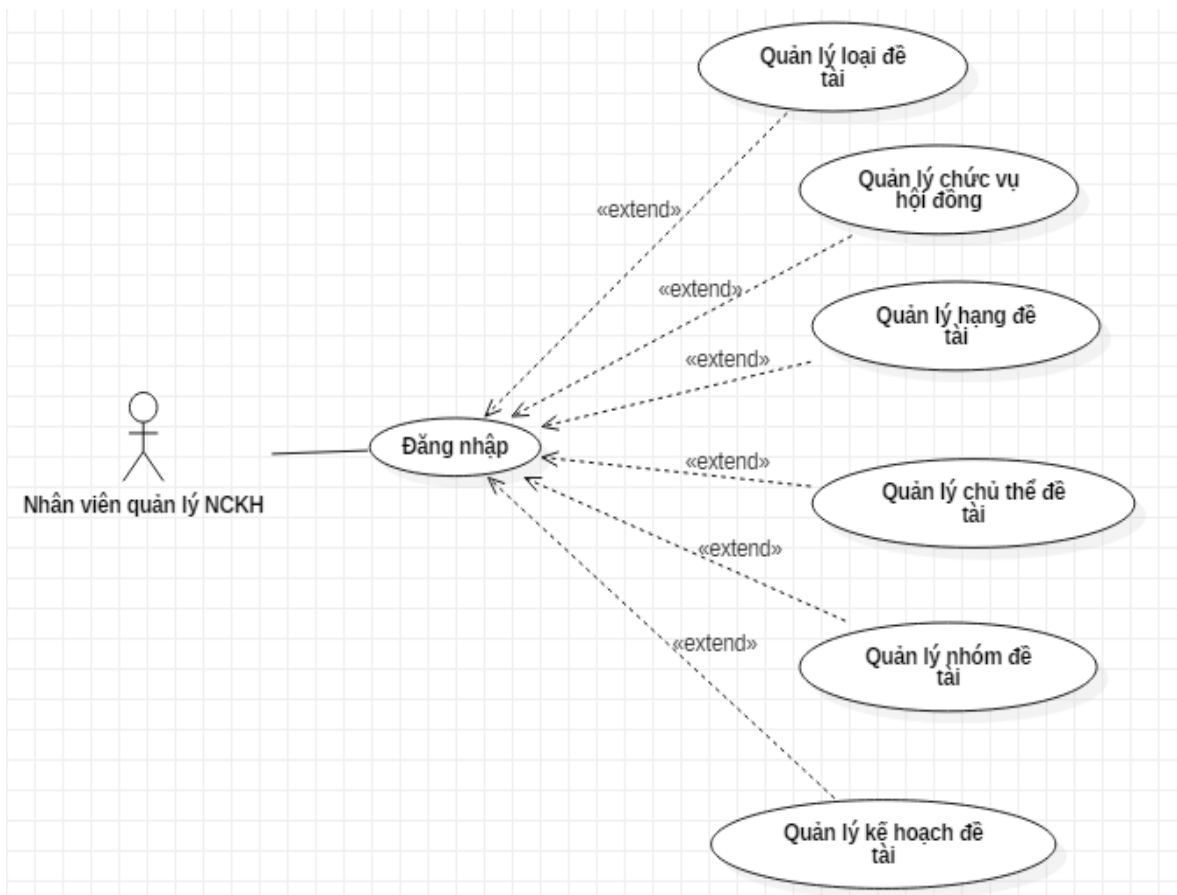
Từ sơ đồ lớp ở trên, em xây dựng mô hình cơ sở dữ liệu, ngoài các lớp chính đã trình bày ở trên, em có sử dụng thêm một số lớp khác để lưu trữ thông tin.



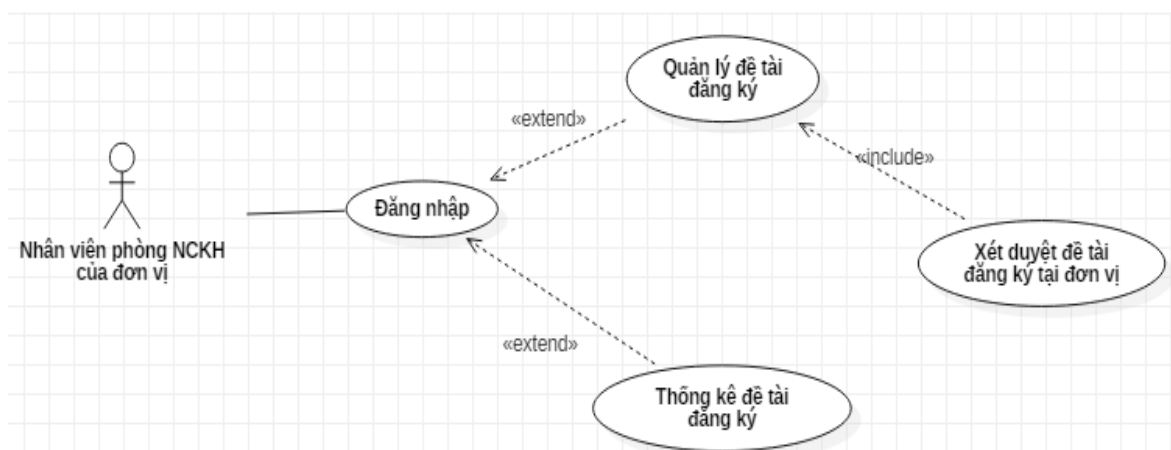
Hình 3.4. Mô hình cơ sở dữ liệu

3.3.2. Hệ thống nghiên cứu khoa học

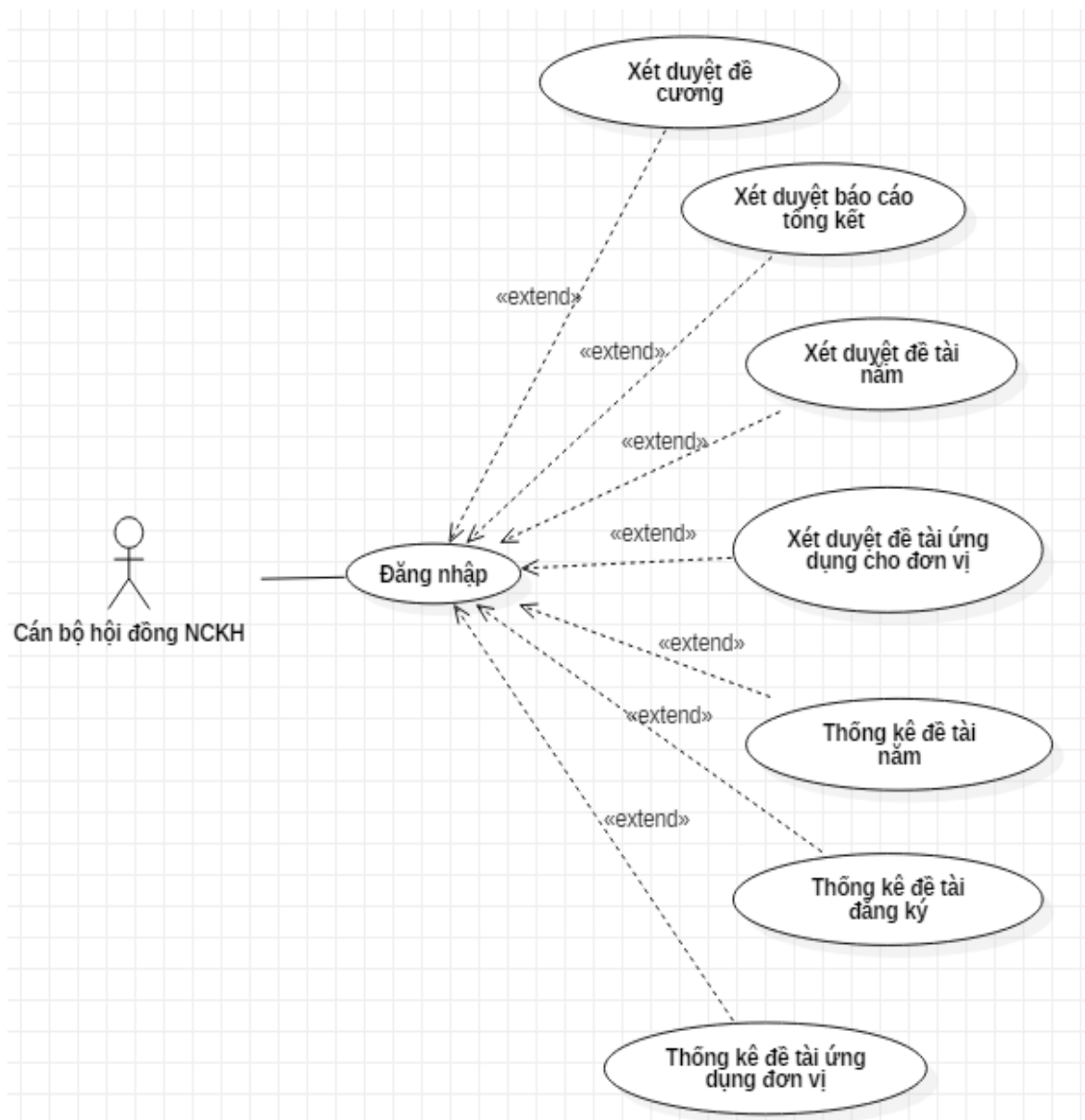
❖ Sơ đồ Use case tổng quát



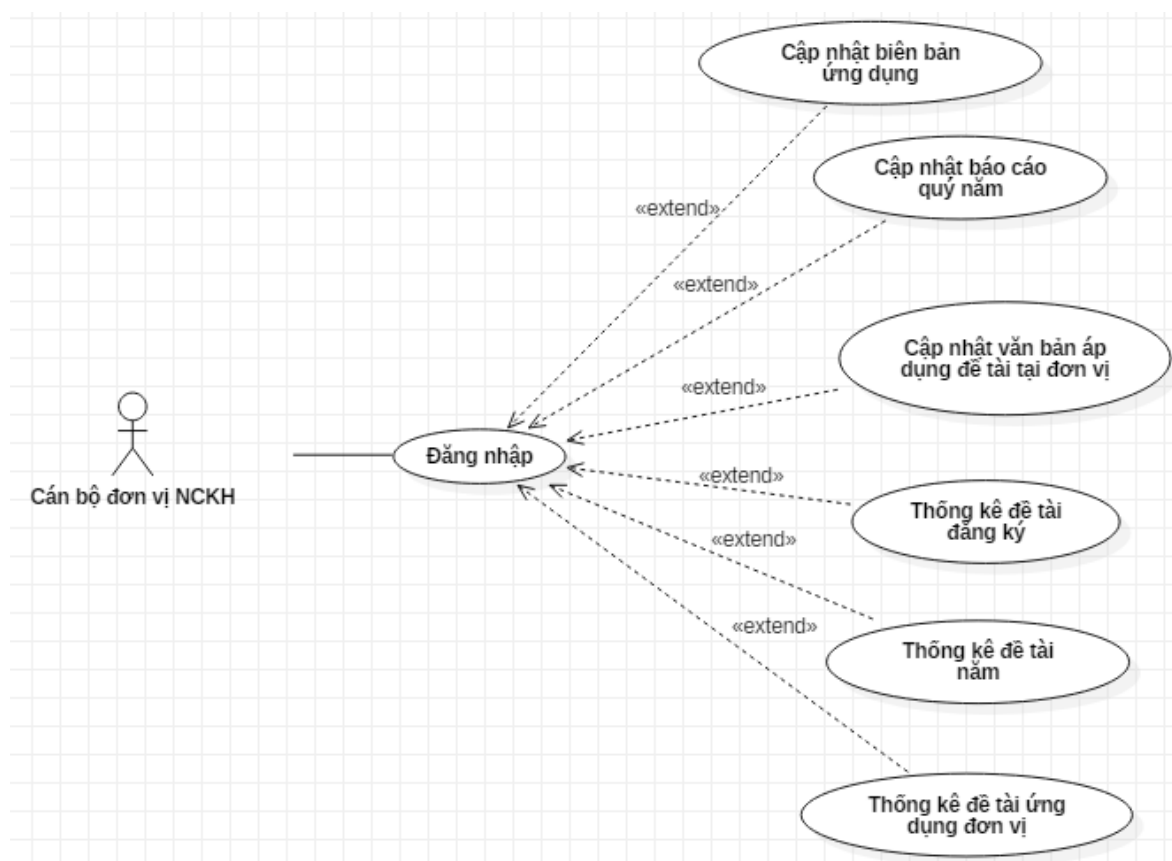
Hình 3.5. Sơ đồ Use case tổng quát (Nhân viên quản lý NCKH)



Hình 3.6. Sơ đồ Use case tổng quát (Nhân viên đơn vị)



Hình 3.7. Sơ đồ Use case tổng quát (Cán bộ hội đồng NCKH)



Hình 3.8. Sơ đồ Use case tổng quát (Cán bộ đơn vị NCKH)

❖ Đặc tả Use case

Use case Đăng nhập

Mã use case	UC11
Tên use case	Đăng nhập
Tác nhân chính	Nhân viên đơn vị, cán bộ hội đồng, cán bộ hội đồng NCKH, nhân viên quản lý NCKH
Tiền điều kiện	Không có
Hậu điều kiện	Đăng nhập thành công vào hệ thống
Điều kiện tối thiểu	Yêu cầu nhân viên quản lý đăng nhập lại
Sự kiện chính	1. Hệ thống hiển thị màn hình đăng nhập. 2. Nhân viên quản lý nhập thông tin tài khoản và mật khẩu. 3. Nhân viên quản lý nhấn “Đăng nhập”. 4. Hệ thống chứng thực tài khoản. 5. Hệ thống thông báo đăng nhập thành công
Ngoại lệ	4.1. Hệ thống chứng thực thất bại, yêu cầu nhân viên nhập lại thông tin tài khoản, mật khẩu.

Use case Quản lý loại đề tài

Mã use case	UC12
Tên use case	Quản lý loại đề tài
Tác nhân chính	Nhân viên quản lý NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Cập nhật thành công loại đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý loại đề tài. 2. Nhân viên nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Nhân viên nhập thông tin cần thiết của loại đề tài. 4. Nhân viên xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin loại đề tài. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý loại đề tài.

Use case Quản lý chức vụ hội đồng

Mã use case	UC13
Tên use case	Quản lý chức vụ hội đồng
Tác nhân chính	Nhân viên quản lý NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Cập nhật thành công chức vụ hội đồng
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý chức vụ hội đồng. 2. Nhân viên nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Nhân viên nhập thông tin cần thiết của chức vụ hội đồng. 4. Nhân viên xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin chức vụ hội đồng. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý chức vụ hội đồng.

Use case Quản lý hạng đề tài

Mã use case	UC14
Tên use case	Quản lý hạng đề tài
Tác nhân chính	Nhân viên quản lý NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Cập nhật thành công hạng đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý hạng đề tài. 2. Nhân viên nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Nhân viên nhập thông tin cần thiết của hạng đề tài. 4. Nhân viên xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin hạng đề tài. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý hạng đề tài.

Use case Quản lý chủ thể đề tài

Mã use case	UC15
Tên use case	Quản lý chủ thể đề tài
Tác nhân chính	Nhân viên quản lý NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Cập nhật thành công hạng đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý hạng đề tài. 2. Nhân viên nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Nhân viên nhập thông tin cần thiết của hạng đề tài. 4. Nhân viên xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin hạng đề tài. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý hạng đề tài.

Use case Quản lý nhóm đề tài

Mã use case	UC16
Tên use case	Quản lý nhóm đề tài
Tác nhân chính	Nhân viên quản lý NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Cập nhật thành công nhóm đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý nhóm đề tài. 2. Nhân viên nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Nhân viên nhập thông tin cần thiết của nhóm đề tài. 4. Nhân viên xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin nhóm đề tài. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý nhóm đề tài.

Use case Quản lý kế hoạch đề tài

Mã use case	UC17
Tên use case	Quản lý kế hoạch đề tài
Tác nhân chính	Nhân viên quản lý NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Cập nhật thành công kế hoạch đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý kế hoạch đề tài. 2. Nhân viên nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Nhân viên nhập thông tin cần thiết của kế hoạch đề tài. 4. Nhân viên xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin kế hoạch đề tài. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý kế hoạch đề tài.

Use case Quản lý đề tài đăng ký

Mã use case	UC18
Tên use case	Quản lý đề tài đăng ký
Tác nhân chính	Nhân viên phòng NCKH của đơn vị
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Cập nhật thành công đề tài đăng ký
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình quản lý đề tài đăng ký. 2. Nhân viên nhấn vào nút chức năng muốn thực hiện (Thêm, sửa, xóa). 3. Nhân viên nhập thông tin cần thiết của đề tài đăng ký. 4. Nhân viên xác nhận lại thông tin và cập nhật dữ liệu. 5. Hệ thống chứng thực thông tin đề tài đăng ký. 6. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình quản lý đề tài đăng ký.

Use case Xét duyệt đề tài đăng ký tại đơn vị

Mã use case	UC19
Tên use case	Xét duyệt đề tài đăng ký tại đơn vị
Tác nhân chính	Nhân viên phòng NCKH của đơn vị
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Xét duyệt thành công đề tài đăng ký tại đơn vị
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình xét duyệt đề tài đăng ký tại đơn vị. 2. Nhân viên nhấn chọn đề tài cần xét duyệt và nhấn nút “Xét duyệt”. 3. Nhân viên nhập thông tin xét duyệt cần thiết của đề tài. 4. Nhân viên nhấn nút “Lưu” để lưu thông tin xét duyệt. 5. Hệ thống chứng thực thông tin xét duyệt. 6. Hệ thống thông báo xét duyệt thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình xét duyệt đề tài.

Use case Thống kê đề tài đăng ký

Mã use case	UC20
Tên use case	Thống kê đề tài đăng ký
Tác nhân chính	Nhân viên phòng NCKH của đơn vị, cán bộ Hội đồng NCKH, cán bộ đơn vị NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Xuất thống kê đề tài đăng ký
Điều kiện tối thiểu	Không có dữ liệu để xuất
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình thống kê đề tài đăng ký. 2. Nhân viên chọn các thông tin đề tài cần thống kê và nhấn nút In. 3. Hệ thống hiển thị kết quả thống kê
Ngoại lệ	Không có

Use case Xét duyệt đề cương

Mã use case	UC21
Tên use case	Xét duyệt đề cương
Tác nhân chính	Cán bộ Hội đồng NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Xét duyệt thành công đề tài cương của đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình xét duyệt đề cương của đề tài. 2. Nhân viên nhấn chọn đề tài cần xét duyệt và nhấn nút “Xét duyệt”. 3. Nhân viên nhập thông tin xét duyệt cần thiết của đề tài. 4. Nhân viên nhấn nút “Lưu” để lưu thông tin xét duyệt. 5. Hệ thống chứng thực thông tin xét duyệt. 6. Hệ thống thông báo xét duyệt thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình xét duyệt đề cương.

Use case Xét duyệt đề cương

Mã use case	UC22
Tên use case	Xét duyệt báo cáo tổng kết
Tác nhân chính	Cán bộ Hội đồng NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Xét duyệt thành công đề tài cương của đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình xét duyệt báo cáo tổng kết của đề tài. 2. Nhân viên nhấn chọn đề tài cần xét duyệt và nhấn nút “Xét duyệt”. 3. Nhân viên nhập thông tin xét duyệt cần thiết của đề tài. 4. Nhân viên nhấn nút “Lưu” để lưu thông tin xét duyệt. 5. Hệ thống chứng thực thông tin xét duyệt. 6. Hệ thống thông báo xét duyệt thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình xét duyệt báo cáo tổng kết.

Use case Xét duyệt đề tài năm

Mã use case	UC23
Tên use case	Xét duyệt đề tài năm
Tác nhân chính	Cán bộ Hội đồng NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Xét duyệt thành công đề tài năm
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình xét duyệt đề tài năm. 2. Nhân viên nhấn chọn đề tài cần xét duyệt và nhấn nút “Xét duyệt”. 3. Nhân viên nhập thông tin xét duyệt cần thiết của đề tài. 4. Nhân viên nhấn nút “Lưu” để lưu thông tin xét duyệt. 5. Hệ thống chứng thực thông tin xét duyệt. 6. Hệ thống thông báo xét duyệt thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình xét duyệt đề tài năm.

Use case Xét duyệt đề tài ứng dụng cho đơn vị

Mã use case	UC24
Tên use case	Xét duyệt đề tài ứng dụng cho đơn vị
Tác nhân chính	Cán bộ Hội đồng NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Xét duyệt thành công đề tài ứng dụng cho đơn vị
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình xét duyệt đề tài ứng dụng cho đơn vị. 2. Nhân viên nhấn chọn đề tài cần xét duyệt và nhấn nút “Xét duyệt”. 3. Nhân viên nhập thông tin xét duyệt cần thiết của đề tài. 4. Nhân viên nhấn nút “Lưu” để lưu thông tin xét duyệt. 5. Hệ thống chứng thực thông tin xét duyệt. 6. Hệ thống thông báo xét duyệt thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình xét duyệt đề tài ứng dụng cho đơn vị.

Use case Thống kê đề tài năm

Mã use case	UC25
Tên use case	Thống kê đề tài năm
Tác nhân chính	Nhân viên phòng NCKH của đơn vị, cán bộ đơn vị NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Xuất thống kê đề tài năm
Điều kiện tối thiểu	Không có dữ liệu để xuất
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình thống kê đề tài năm. 2. Nhân viên chọn các thông tin đề tài cần thống kê và nhấn nút In. 3. Hệ thống hiển thị kết quả thống kê
Ngoại lệ	Không có

Use case Thống kê đề tài ứng dụng tại đơn vị

Mã use case	UC26
Tên use case	Thống kê đề tài ứng dụng tại đơn vị

Mã use case	UC26
Tác nhân chính	Nhân viên phòng NCKH của đơn vị, cán bộ đơn vị NCKH
Tiền điều kiện	Đăng nhập thành công vào hệ thống.
Hậu điều kiện	Xuất thống kê đề tài ứng dụng tại đơn vị
Điều kiện tối thiểu	Không có dữ liệu để xuất
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình thống kê đề tài ứng dụng tại đơn vị. 2. Nhân viên chọn các thông tin đề tài cần thống kê và nhấn nút In. 3. Hệ thống hiển thị kết quả thống kê
Ngoại lệ	Không có

Use case Cập nhật biên bản ứng dụng

Mã use case	UC27
Tên use case	Cập nhật biên bản ứng dụng
Tác nhân chính	Cán bộ đơn vị NCKH (đơn vị đã áp dụng đề tài)
Tiền điều kiện	Đăng nhập thành công vào hệ thống
Hậu điều kiện	Cập nhật thành công biên bản ứng dụng cho đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	<ol style="list-style-type: none"> 1. Hệ thống hiển thị màn hình cập nhật biên bản ứng dụng. 2. Nhân viên nhấn chọn đề tài cần cập nhật biên bản và nhấn nút “Cập nhật biên bản”. 3. Nhân viên nhập thông tin cập nhật biên bản cần thiết của đề tài. 4. Nhân viên nhấn nút “Lưu” để lưu thông tin cập nhật. 5. Hệ thống chứng thực thông tin cập nhật. 6. Hệ thống thông báo cập nhật biên bản thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình cập nhật biên bản ứng dụng.

Use case Cập nhật báo cáo quý năm

Mã use case	UC28
Tên use case	Cập nhật báo cáo quý năm
Tác nhân chính	Cán bộ đơn vị NCKH (đơn vị đã áp dụng đề tài)
Tiền điều kiện	Đăng nhập thành công vào hệ thống

Mã use case	UC28
Hậu điều kiện	Cập nhật thành công báo cáo quý năm cho đề tài
Điều kiện tối thiểu	Loại bỏ thông tin đã nhập và quay lại bước trước
Sự kiện chính	1. Hệ thống hiển thị màn hình cập nhật báo cáo quý năm. 2. Nhân viên nhấn chọn đề tài cần cập nhật và nhấn nút “Cập nhật”. 3. Nhân viên nhập thông tin cập nhật biên bản cần thiết của đề tài. 4. Nhân viên nhấn nút “Lưu” để lưu thông tin cập nhật. 5. Hệ thống chứng thực thông tin cập nhật. 6. Hệ thống thông báo cập nhật biên bản thành công.
Ngoại lệ	5.1. Hệ thống chứng thực thất bại, quay lại màn hình cập nhật báo cáo quý năm.

❖ Các lớp

Loại đề tài

Lớp loại đề tài dùng để lưu thông tin loại đề tài của đề tài NCKH

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maLoai	Kiểu số nguyên dương	Mỗi loại đề tài có một mã riêng để phân biệt
2	tenLoai	Kiểu chuỗi kí tự	Tên của loại đề tài
3	viTri	Kiểu số nguyên dương	Vị trí ưu tiên của loại đề tài
4	trangThai	Kiểu số nguyên	Trạng thái loại đề tài

Hạng đề tài

Lớp hạng đề tài dùng để lưu thông tin hạng đề tài của đề tài NCKH

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maHang	Kiểu số nguyên dương	Mỗi hạng đề tài có một mã riêng để phân biệt
2	tenHang	Kiểu chuỗi kí tự	Tên của hạng đề tài
3	viTri	Kiểu số nguyên dương	Vị trí ưu tiên của hạng đề tài
4	trangThai	Kiểu số nguyên	Trạng thái hạng đề tài

Chủ thể đề tài

Lớp hạng đề tài dùng để lưu thông tin hạng đề tài của đề tài NCKH

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maChuThe	Kiểu số nguyên dương	Mỗi chủ thể đề tài có một mã riêng để phân biệt
2	tenChuThe	Kiểu chuỗi kí tự	Tên của chủ thể đề tài
3	viTri	Kiểu số nguyên dương	Vị trí ưu tiên của chủ thể đề tài
4	trangThai	Kiểu số nguyên	Trạng thái chủ thể đề tài

Nhóm đề tài

Lớp nhóm đề tài dùng để lưu thông tin nhóm đề tài của đề tài NCKH

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maNhom	Kiểu số nguyên dương	Mỗi nhóm đề tài có một mã riêng để phân biệt
2	tenNhom	Kiểu chuỗi kí tự	Tên của nhóm đề tài
3	viTri	Kiểu số nguyên dương	Vị trí ưu tiên của nhóm đề tài
4	trangThai	Kiểu số nguyên	Trạng thái nhóm đề tài

Văn bản áp dụng

Lớp văn bản áp dụng dùng để lưu thông tin văn bản áp dụng đề tài ở đơn vị

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	trangThai	Kiểu số nguyên	Trạng thái của đề tài đã cập nhật văn bản hay chưa
2	tapTin	Kiểu chuỗi kí tự	Tập tin văn bản áp dụng

Văn bản chấp thuận

Lớp văn bản chấp thuận dùng để lưu thông tin văn bản chấp thuận đề tài ở đơn vị

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	trangThai	Kiểu số nguyên	Trạng thái của đề tài đã cập nhật văn bản chấp thuận hay chưa
2	tapTin	Kiểu chuỗi kí tự	Tập tin văn bản chấp thuận

Biên bản kế hoạch

Lớp biên bản kế hoạch dùng để lưu thông tin biên bản kế hoạch của đề tài

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	trangThai	Kiểu số nguyên	Trạng thái của đề tài đã cập nhật biên bản kế hoạch hay chưa
2	tapTin	Kiểu chuỗi kí tự	Tập tin biên bản kế hoạch

Đề tài đăng ký

Lớp đề tài đăng ký dùng để lưu thông tin đề tài đăng ký

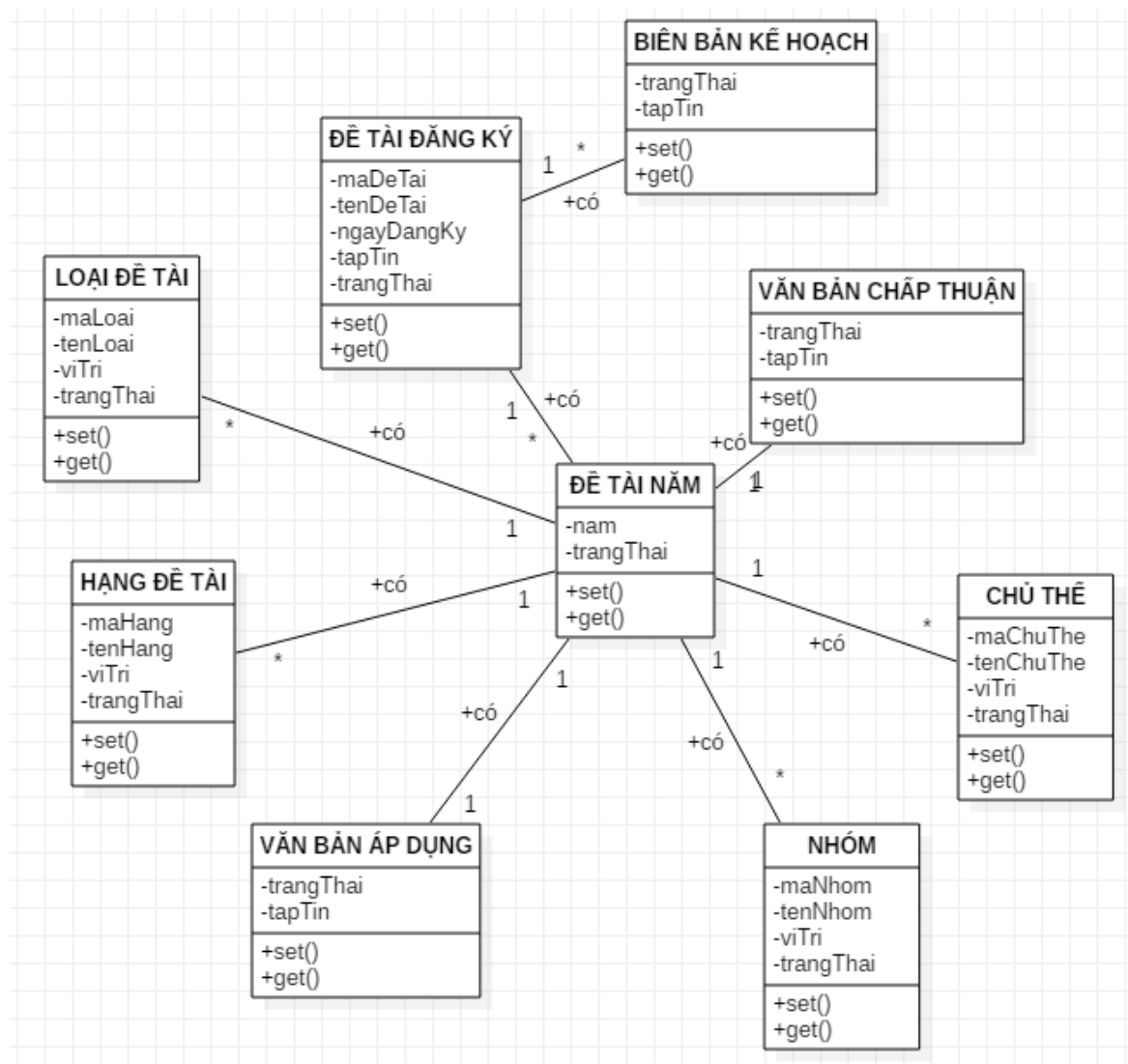
STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	maDeTai	Kiểu số nguyên dương	Mỗi đề tài có một mã riêng để phân biệt
2	tenDeTai	Kiểu chuỗi kí tự	Tên đề tài đăng ký
3	ngayDangKy	Kiểu ngày tháng	Ngày đăng ký đề tài
4	tapTin	Kiểu chuỗi kí tự	Tập tin của đề tài
5	trangThai	Kiểu số nguyên	Trạng thái của đề tài

Đề tài năm

Lớp đề tài năm dùng để lưu thông tin của các đề tài công nhận hằng năm

STT	Thuộc tính	Kiểu dữ liệu	Ý nghĩa
1	nam	Kiểu số nguyên dương	Năm mà đề tài được công nhận đề tài năm
2	trangThai	Kiểu số nguyên	Trạng thái của đề tài

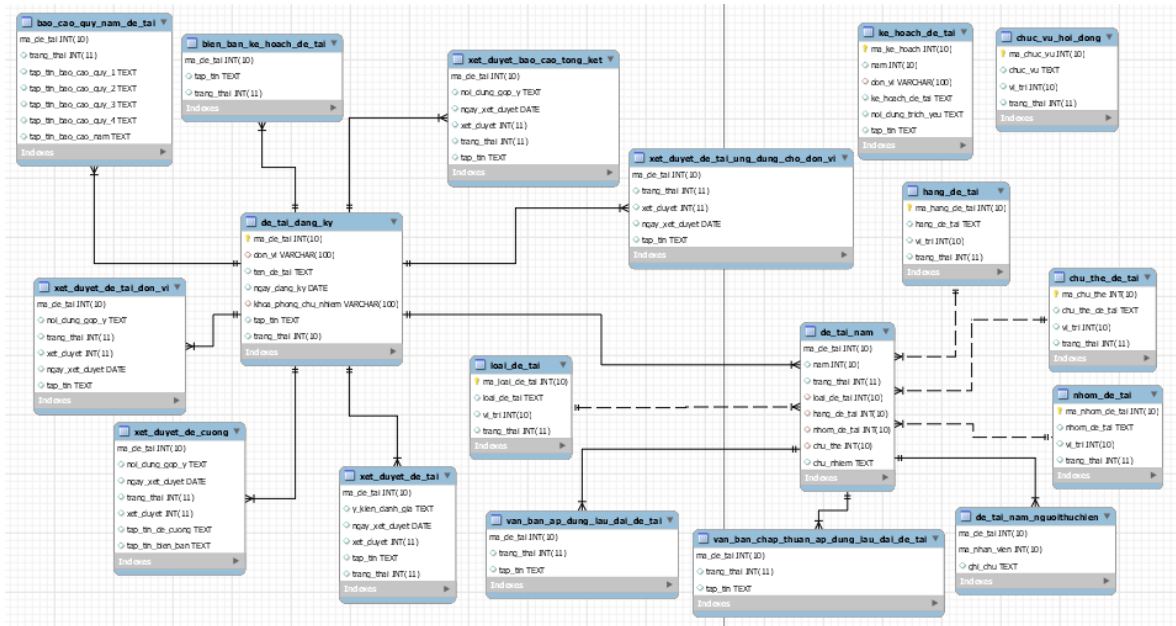
❖ Sơ đồ lớp



Hình 3.9. Sơ đồ lớp

❖ Mô hình cơ sở dữ liệu

Từ sơ đồ lớp ở trên, em xây dựng mô hình cơ sở dữ liệu, ngoài các lớp chính đã trình bày ở trên, em có sử dụng thêm một số lớp khác để lưu trữ thông tin.



Hình 3.10. Mô hình CSDL

3.4. Cài đặt giải pháp

Hiện trạng hệ thống: Hệ thống phải được tích hợp vào hệ thống có các dịch vụ khác, sử dụng một số thông tin của các dịch vụ đã có. Hệ thống sẽ gồm một số dịch vụ như sau:

Dịch vụ 1: Dịch vụ quản lý về nhân viên, phân quyền người dùng

Dịch vụ 2: Dịch vụ về chứng thực đăng nhập

Dịch vụ 3: Dịch vụ Registry của JHipster (có thể tải dịch vụ này tại địa chỉ sau:

<https://www.jhipster.tech/jhipster-registry/>).

Dịch vụ 4: Dịch vụ chỉ đạo tuyến

Dịch vụ 5: Dịch vụ NCKH

Dịch vụ 6: Dịch vụ Gateway

3.4.1. Xây dựng cấu trúc Microservice cho hệ thống chỉ đạo tuyến và hệ thống NCKH

Căn cứ vào hiện trạng ở trên, cùng với điều kiện về thời gian, và kiến thức, em xin đưa ra đề xuất cấu trúc cho hệ thống chỉ đạo tuyến và hệ thống NCKH như sau:

Theo cấu trúc của Microservice, hệ thống chỉ đạo tuyến và hệ thống NCKH sẽ có các dịch vụ sau:

Dịch vụ 1: Dịch vụ chứng thực đăng nhập (JHipster UAA server (for microservice OAuth2 authentication)).

Dịch vụ 2: Dịch vụ để hiển thị giao diện cho hệ thống (Microservice gateway).

Dịch vụ 3: Dịch vụ Registry của JHipster.

Dịch vụ 4: Dịch vụ của phân hệ chỉ đạo tuyến (Microservice application).

Dịch vụ 5: Dịch vụ của phân hệ NCKH (Microservice application).

Do hạn chế về thời gian thực hiện đề tài và giúp cho việc lập trình được nhanh hơn, em quyết định gộp lại thành 3 dịch vụ, cụ thể, em sẽ cài đặt dịch vụ của phân hệ NCKH bên trong dịch vụ của phân hệ chỉ đạo tuyến.

Như vậy, các dịch vụ cần cài đặt sẽ là:

Dịch vụ 1: Dịch vụ để hiển thị giao diện cho hệ thống (Microservice gateway).

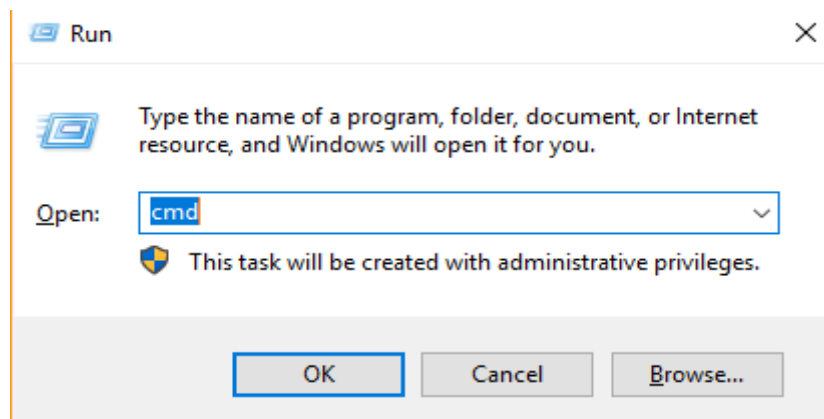
Dịch vụ 2: Dịch vụ Registry của JHipster.

Dịch vụ 3: Dịch vụ của phân hệ chỉ đạo tuyến (bao gồm phân hệ NCKH).

Xây dựng kiến trúc Microservice cho ứng dụng:

Từ cấu trúc bên trên, em có thể xây dựng được hệ thống chỉ đạo tuyến và NCKH theo một phân kiến trúc của Microservice, và đảm bảo được về chức năng của hệ thống.

Bước 1: Mở Command Prompt trên máy tính bằng phím tắt Windows + R, nhập cmd và nhấn Enter.



Hình 3.11. Xây dựng cấu trúc cho hệ thống

Bước 2: Tạo thư mục mới và di chuyển vào thư mục đó bằng cách gõ:

```
mkdir ten_thu_muc
```

```
cd ten_thu_muc
```

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17134.829]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Admin>mkdir test

C:\Users\Admin>cd test

C:\Users\Admin\test>
```

Hình 3.12. Xây dựng cấu trúc cho hệ thống
Bước 3: Để tạo một ứng dụng: Gõ jhipster và nhấn Enter.

```
Select npm
https://www.jhipster.tech

v5.8.1
C:\Users\Admin\test

https://www.jhipster.tech/creating-an-app/
https://opencollective.com/generator-jhipster

JHipster
JHipster update available: 6.1.2 (current: 5.8.1)
Run npm install -g generator-jhipster to update.

? Which *type* of application would you like to create? (Use arrow keys)
> Monolithic application (recommended for simple projects)
  Microservice application
  Microservice gateway
  JHipster UAA server (for microservice OAuth2 authentication)
```

Hình 3.13. Xây dựng cấu trúc cho hệ thống

Để có thể cài đặt cấu trúc hệ thống, em sẽ cài đặt lần lượt các dịch vụ nhỏ của hệ thống.

Sử dụng khởi tạo ứng dụng tự động với JHipster, đối với từng dịch vụ, em sẽ tạo ra một thư mục riêng để lưu trữ:

gateway	4/11/2019 3:31 PM	File folder
jhipster-registry	3/23/2019 1:46 PM	File folder
service-chi-dao-tuyen	6/15/2019 4:19 PM	File folder

Hình 3.14. Xây dựng cấu trúc cho hệ thống

Sau đó, em sẽ tiến hành cài đặt từng dịch vụ nhỏ vào từng thư mục đã tạo ở trên

❖ Tạo dịch vụ Gateway:

Bước 1: Chọn Microservice gateway và nhấn Enter

```
? Which *type* of application would you like to create?
  Monolithic application (recommended for simple projects)
  Microservice application
> Microservice gateway
  JHipster UAA server (for microservice OAuth2 authentication)
```

Hình 3.14. Xây dựng dịch vụ gateway

Bước 2: Nhập như hình bên dưới, mỗi lần nhập nhấn Enter một lần. Sau khi nhập xong nhấn Enter và JHipster sẽ tiến hành cài đặt cấu trúc cần thiết cho dịch vụ gateway. Ở đây, em sẽ thiết kế giao diện sử dụng React.

```
? Which *type* of application would you like to create? Microservice gateway
? What is the base name of your application? gateway
? As you are running in a microservice architecture, on which port would like your server to run? It should be unique to avoid port conflicts. 8080
? What is your default Java package name? vn.test
? Which service discovery server do you want to use? JHipster Registry (uses Eureka, provides Spring Cloud Config support and monitoring dashboards)
? Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? Which *production* database would you like to use? MySQL
? Which *development* database would you like to use? MySQL
? Do you want to use Hibernate 2nd level cache? Yes
? Would you like to use Maven or Gradle for building the backend? Gradle
? Which other technologies would you like to use?
? Which *Framework* would you like to use for the client? React
? Would you like to enable *SASS* stylesheet preprocessor? No
? Would you like to enable internationalization support? No
? Besides JUnit and Jest, which testing frameworks would you like to use?
? Would you like to install other generators from the JHipster Marketplace? (y/N) N
```

Hình 3.15. Xây dựng dịch vụ gateway

Sau khi cài đặt xong:

Cấu trúc thư mục sẽ có dạng như sau:

Trong File Explorer:

gradle	6/28/2019 6:50 PM	File folder	
node_modules	6/28/2019 6:52 PM	File folder	
src	6/28/2019 6:50 PM	File folder	
webpack	6/28/2019 6:50 PM	File folder	
.editorconfig	6/28/2019 6:50 PM	EDITORCONFIG File	1 KB
.gitattributes	6/28/2019 6:50 PM	Text Document	4 KB
.gitignore	6/28/2019 6:50 PM	Text Document	2 KB
.huskyrc	6/28/2019 6:50 PM	HUSKYRC File	1 KB
.prettierignore	6/28/2019 6:50 PM	PRETTIERIGNORE ...	1 KB
.prettierrc	6/28/2019 6:50 PM	PRETTIERRC File	1 KB
.yo-rc.json	6/28/2019 6:53 PM	JSON File	2 KB
build.gradle	6/28/2019 6:50 PM	GRADLE File	11 KB
gradle.properties	6/28/2019 6:50 PM	PROPERTIES File	3 KB
gradlew	6/28/2019 6:50 PM	File	6 KB
gradlew.bat	6/28/2019 6:50 PM	Windows Batch File	3 KB
package.json	6/28/2019 6:50 PM	JSON File	6 KB
package-lock.json	6/28/2019 6:52 PM	JSON File	633 KB
README.md	6/28/2019 6:50 PM	MD File	7 KB
settings.gradle	6/28/2019 6:50 PM	GRADLE File	1 KB
tsconfig.json	6/28/2019 6:50 PM	JSON File	1 KB
tsconfig.test.json	6/28/2019 6:50 PM	JSON File	1 KB
tslint.json	6/28/2019 6:50 PM	JSON File	6 KB

Hình 3.16. Xây dựng dịch vụ gateway

Trong màn hình Command Prompt:

```
added 1875 packages from 1504 contributors and audited 38384 packages in 153.241s
found 295 vulnerabilities (63 low, 14 moderate, 218 high)
  run `npm audit fix` to fix them, or `npm audit` for details
Application successfully committed to Git.

If you find JHipster useful consider sponsoring the project https://www.jhipster.tech/sponsors/
Server application generated successfully.

Run your Spring Boot application:
./gradlew (gradlew if using Windows Command Prompt)

Client application generated successfully.

Start your Webpack development server with:
npm start

> gateway@0.0.0 cleanup C:\Users\Admin\test\gateway
> rimraf build/www

INFO! Congratulations, JHipster execution is complete!
```

Hình 3.17. Xây dựng dịch vụ gateway

Bước 3: Nhập `npm install` để cài đặt các gói cần thiết. Sau đó nhập `npm start` để chạy dịch vụ

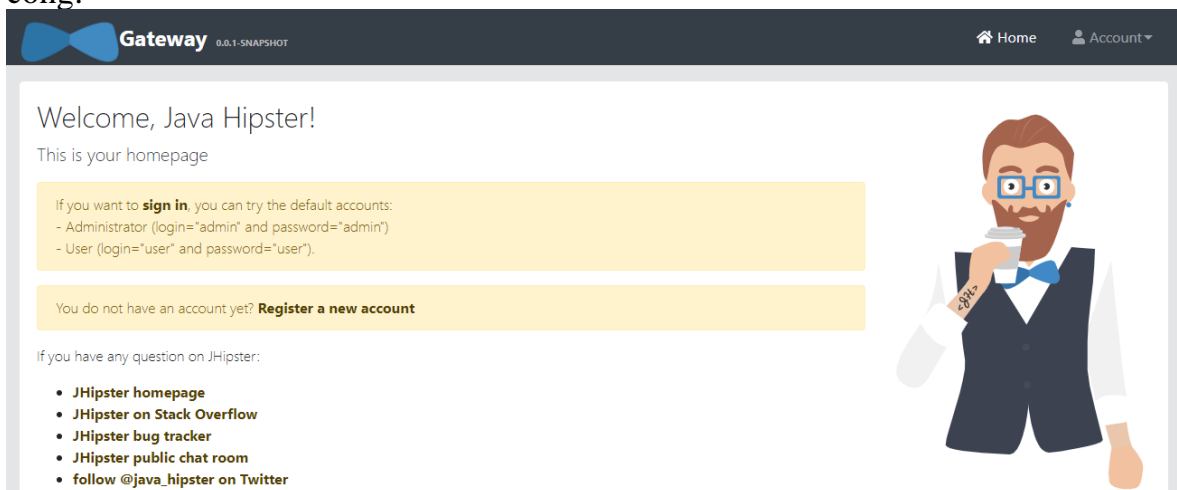
```
✓ Compile modules
✓ Build modules
✓ Optimize modules
✓ Emit files

DONE Compiled successfully in 22958ms

i wdm: 969 modules
i wdm: Compiled successfully.
[Browsersync] Proxying: http://localhost:9060
[Browsersync] Access URLs:
-----
    Local: http://localhost:9000
    External: http://192.168.1.5:9000
-----
    UI: http://localhost:3001
    UI External: http://localhost:3001
-----
```

Hình 3.18. Xây dựng dịch vụ gateway

Lúc này, trên trình duyệt mặc định của hệ điều hành Windows sẽ mở ra một trang web có đường dẫn là: <http://localhost:9000/#/> . Trang web như sau là đã thành công:



Hình 3.19. Xây dựng dịch vụ gateway

❖ Tạo dịch vụ cho phân hệ chỉ đạo tuyến và NCKH:

Bước 1: Chọn Microservice application và nhấn Enter


```
? Which *type* of application would you like to create?
  Monolithic application (recommended for simple projects)
> Microservice application
  Microservice gateway
  JHipster UAA server (for microservice OAuth2 authentication)
```

Hình 3.20. Xây dựng dịch vụ chỉ đạo tuyến và NCKH

Bước 2: Nhập như hình bên dưới, mỗi lần nhập nhấn Enter một lần. Sau khi nhập xong nhấn Enter và JHipster sẽ tiến hành cài đặt cấu trúc cần thiết cho dịch vụ chỉ đạo tuyến và NCKH.


















```
? Which *type* of application would you like to create? Microservice application
? What is the base name of your application? chidaotuyen
? As you are running in a microservice architecture, on which port would like your server to run? I
could be unique to avoid port conflicts. 8081
? What is your default Java package name? vn.vnpt.chidaotuyen
? Which service discovery server do you want to use? JHipster Registry (uses Eureka, provides Sprin
oud Config support and monitoring dashboards)
? Which *type* of authentication would you like to use? JWT authentication (stateless, with a token
? Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQ
? Which *production* database would you like to use? MySQL
? Which *development* database would you like to use? MySQL
? Do you want to use the Spring cache abstraction? Yes, with the Hazelcast implementation (distribu
cache, for multiple nodes)
? Do you want to use Hibernate 2nd level cache? Yes
? Would you like to use Maven or Gradle for building the backend? Gradle
? Which other technologies would you like to use?
? Would you like to enable internationalization support? No
? Besides JUnit and Jest, which testing frameworks would you like to use?
? Would you like to install other generators from the JHipster Marketplace? (y/N) N
```

Hình 3.21. Xây dựng dịch vụ chỉ đạo tuyến và NCKH

Sau khi cài đặt xong:

Cấu trúc thư mục sẽ có dạng như sau:

Trong File Explorer:

	gradle	6/28/2019 7:09 PM	File folder	
	node_modules	6/28/2019 7:10 PM	File folder	
	src	6/28/2019 7:09 PM	File folder	
	.editorconfig	6/28/2019 7:09 PM	EDITORCONFIG File	1 KB
	.gitattributes	6/28/2019 7:09 PM	Text Document	4 KB
	.gitignore	6/28/2019 7:09 PM	Text Document	2 KB
	.prettierrc	6/28/2019 7:09 PM	PRETTIERIGNORE ...	1 KB
	.prettierrc	6/28/2019 7:09 PM	PRETTIERRC File	1 KB
	.yo-rc.json	6/28/2019 7:09 PM	JSON File	2 KB
	build.gradle	6/28/2019 7:09 PM	GRADLE File	10 KB
	gradle.properties	6/28/2019 7:09 PM	PROPERTIES File	3 KB
	gradlew	6/28/2019 7:09 PM	File	6 KB
	gradlew.bat	6/28/2019 7:09 PM	Windows Batch File	3 KB
	package.json	6/28/2019 7:09 PM	JSON File	1 KB
	package-lock.json	6/28/2019 7:10 PM	JSON File	184 KB
	README.md	6/28/2019 7:09 PM	MD File	4 KB
	settings.gradle	6/28/2019 7:09 PM	GRADLE File	1 KB

Hình 3.22. Xây dựng dịch vụ chỉ đạo tuyến và NCKH

Trong màn hình Command Prompt:

```
create src\test\resources\config\bootstrap.yml
create src\test\java\vn\vnpt\chidaotuyen\security\jwt\TokenProviderTest
create src\test\java\vn\vnpt\chidaotuyen\security\jwt\JWTFilterTest.java
create .gitignore
create .gitattributes
create .editorconfig
create package.json
create src\main\docker\central-server-config\README.md
create README.md

Installing generator-jhipster@5.8.1 locally using npm

> spawn-sync@1.0.15 postinstall C:\Users\Admin\test\service-chi-dao-tuyen\
> node postinstall

npm notice created a lockfile as package-lock.json. You should commit this
added 512 packages from 330 contributors and audited 3747 packages in 32.9
found 13 vulnerabilities (2 moderate, 11 high)
  run `npm audit fix` to fix them, or `npm audit` for details
Application successfully committed to Git.

If you find JHipster useful consider sponsoring the project https://www.jh
Server application generated successfully.

Run your Spring Boot application:
./gradlew (gradlew if using Windows Command Prompt)
INFO! Congratulations, JHipster execution is complete!
```

Hình 3.23. Xây dựng dịch vụ chỉ đạo tuyến và NCKH

Bước 3: Nhập npm install để cài đặt các gói cần thiết. Sau đó nhập gradlew để chạy dịch vụ

```
> Task :bootRun
19:15:28.034 [main] DEBUG org.springframework.boot.devtools.settings.DevToolsSettings - Included pattern
ns for restart : []
19:15:28.044 [main] DEBUG org.springframework.boot.devtools.settings.DevToolsSettings - Excluded pattern
ns for restart : [/spring-boot-actuator/target/classes/, /spring-boot-devtools/target/classes/, /spring
-boot/target/classes/, /spring-boot-starter-[w-]+/, /spring-boot-autoconfigure/target/classes/, /spring
-boot-starter/target/classes/]
19:15:28.046 [main] DEBUG org.springframework.boot.devtools.restart.ChangeableUrls - Matching URLs for
reloading : [file:/C:/Users/Admin/test/service-chi-dao-tuyen/build/classes/java/main/, file:/C:/Users/A
min/test/service-chi-dao-tuyen/build/resources/main/]

??? ???    ??? ?????????? ?????????? ?????????? ?????????? ?????????? ??????????
??? ???    ??? ?????????? ?????????? ?????????? ?????????? ?????????? ??????????
??? ?????????? ???    ?????????? ?????????? ???    ?????????? ??????????
??? ???    ?????????? ???    ?????????? ?????????? ???    ?????????? ??????????
????????? ???    ??? ?????????? ???    ?????????? ???    ?????????? ???    ???
????????? ???    ??? ?????????? ???    ?????????? ???    ?????????? ???    ???
```

Hình 3.24. Xây dựng dịch vụ chỉ đạo tuyến và NCKH

3.4.2. Cài đặt hệ thống

Cài đặt tất cả các chức năng cần thiết của chỉ đạo tuyến và quản lý NCKH trong Microservice application(Ở đây em sử dụng Spring JDBC để kết nối với CSDL). Vào thư mục: C:\Users\Admin\test\service-chi-dao-tuyen\src\main\java\vn\vnpt\chidaotuyen tạo các thư mục con cần thiết như sau:

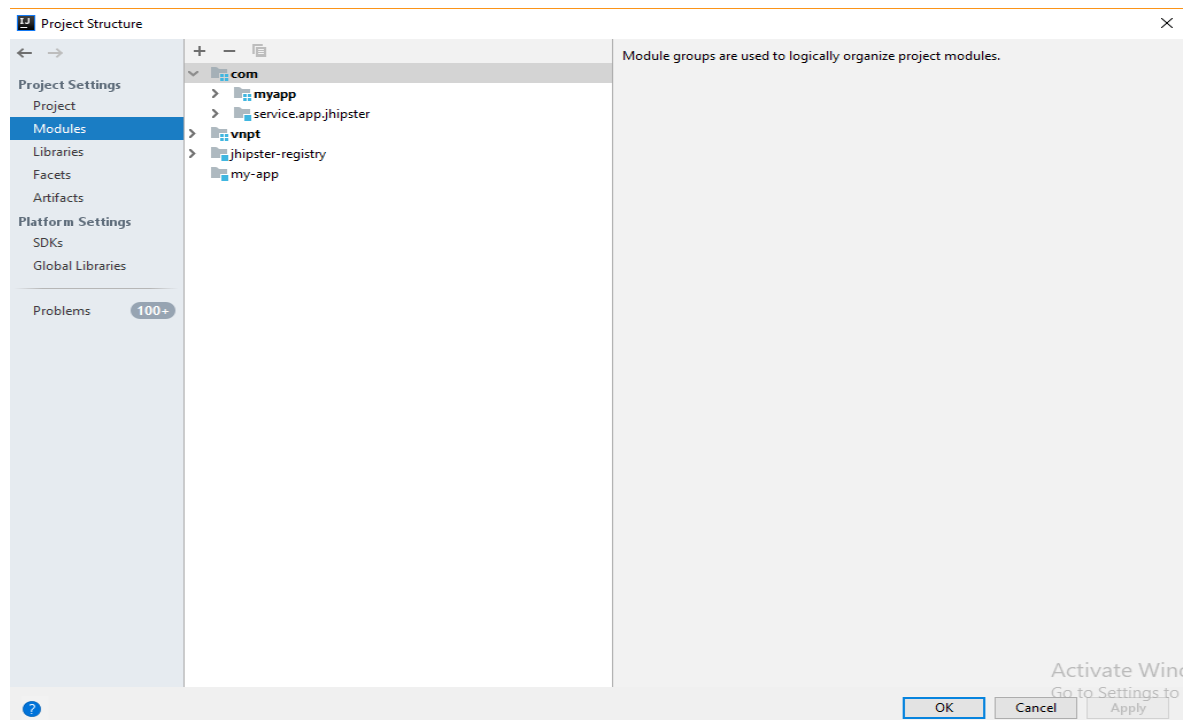
1. Obj: Dùng để chứa các đối tượng cần thiết để ánh xạ dữ liệu từ CSDL.
2. dao: Dùng để chứa các lớp đại diện để thực hiện các chức năng như lấy dữ liệu, cập nhật dữ liệu,...
3. controller: Dùng để tạo các đường dẫn và tiếp nhận các yêu cầu từ gateway, tạo các báo cáo,...

Để chạy hệ thống, cần phải chạy tất cả dịch vụ đã nói ở mục 3.4.1, theo thứ tự sau: JHipster Registry → UAA → Application → Gateway.

Để có thể chạy hệ thống nhanh hơn, em sử dụng trình soạn thảo mã là IntelliJ IDEA 2018.

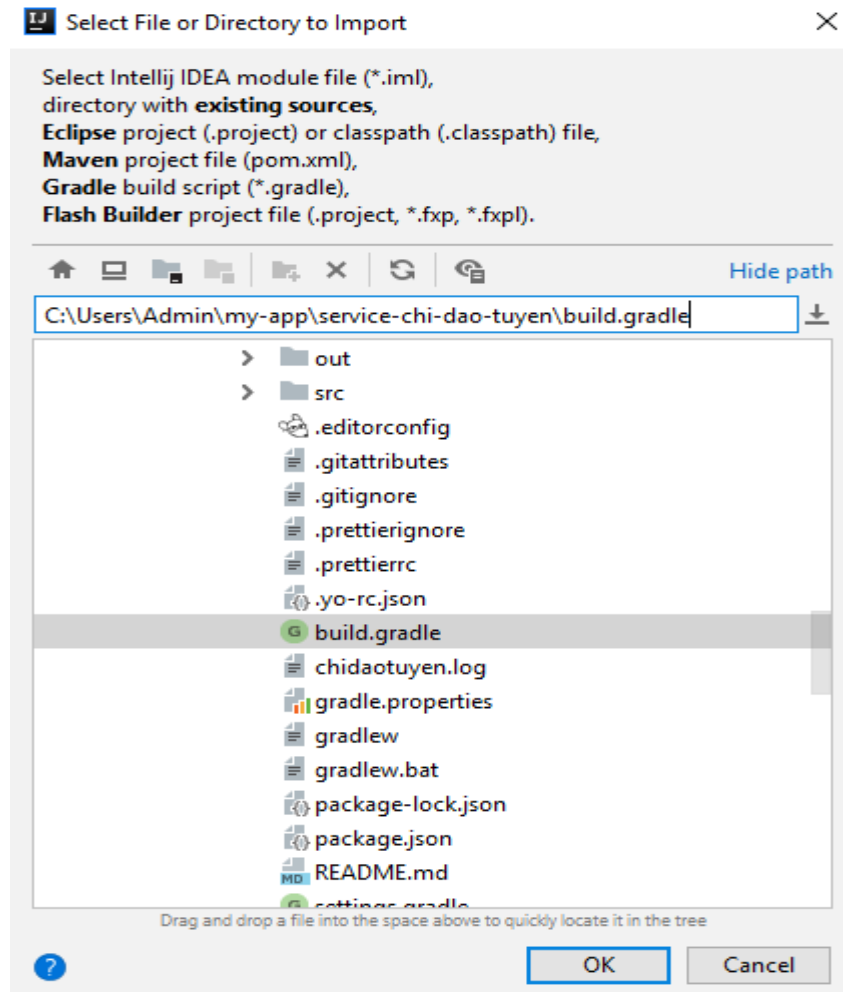
Các bước thêm dịch vụ vào IntelliJ IDEAD 2018:

Bước 1: Chọn File → Project Structure → Modules.



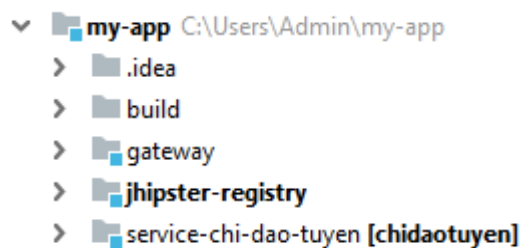
Hình 3.25. Thêm dịch vụ vào IntelliJ IDEA 2018

Bước 2: Nhấn vào dấu + → Chọn Import Modules → Chọn đường dẫn đến các dịch vụ đã tạo ở mục 3.4.1, chọn tập tin build.gradle → Nhấn OK.



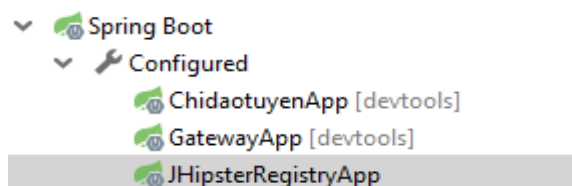
Hình 3.26. Thêm dịch vụ vào IntelliJ IDEA 2018

Cấu trúc thư mục sẽ như hình sau:



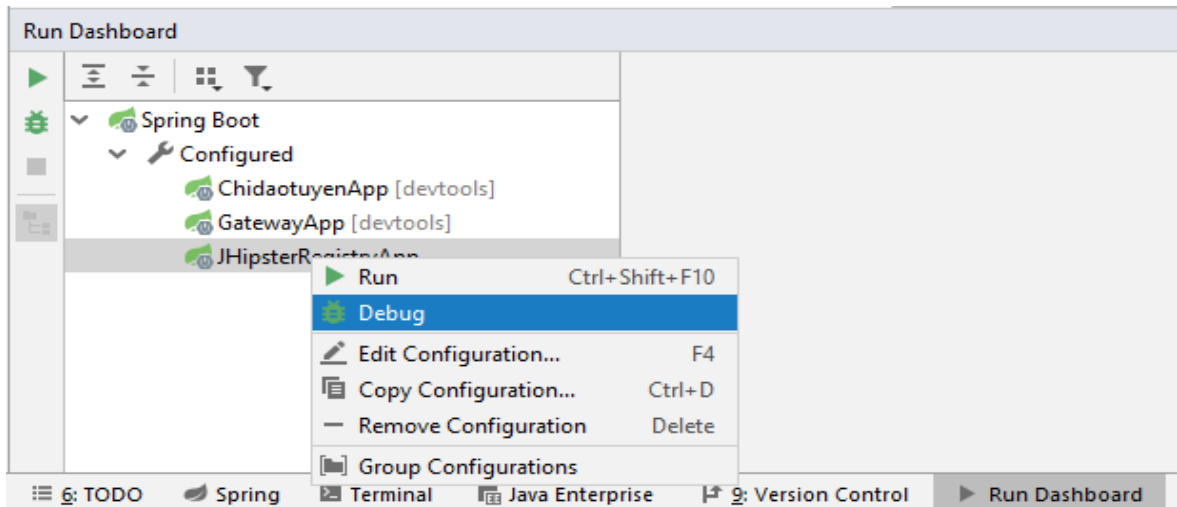
Hình 3.27. Thêm dịch vụ vào IntelliJ IDEA 2018

Bước 4: Chọn mục Run Dashboard



Hình 3.28. Chạy dịch vụ trong IntelliJ IDEA 2018

Bước 5: Nhấn chuột phải vào dịch vụ → Chọn Debug (Gỡ lỗi) hoặc Run (Chạy), ở đây sẽ chọn Debug. Làm tương tự với tất cả dịch vụ còn lại.



Hình 3.29. Chạy dịch vụ trong IntelliJ IDEA 2018

Bước 6: Tại trang Terminal → Chọn đường dẫn tới dịch vụ gateway → nhập `npm start` sau đó đợi dịch vụ chạy. Sau khi chạy xong, sẽ hiển thị như hình 3.30.

```
dmin\my-app\gateway\node_modules\thread-loader\dist\cjs.js??ref--4-2!C:\Users\Admin\my-app\gateway\node_modules\ts-loader\index.js??ref--4-3!
gateway\node_modules\tslint-loader\index.js!C:\Users\Admin\my-app\gateway\src\main\webapp\app\shared\reducers\index.ts
i [wdm]: Compiled with warnings.
[Browsersync] Proxying: http://localhost:9060
[Browsersync] Access URLs:
-----
    Local: http://localhost:9000
    External: http://192.168.1.6:9000
-----
    UI: http://localhost:3001
    UI External: http://localhost:3001
-----
```

Hình 3.30. Chạy dịch vụ trong IntelliJ IDEA 2018

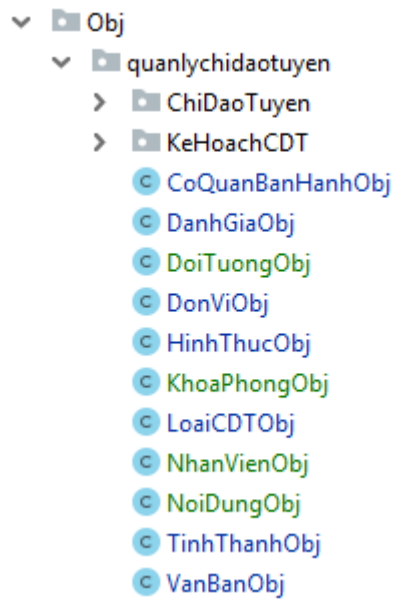
Bước 7: Mở một trình duyệt web và nhập địa chỉ: <http://localhost:9000/#/>



Hình 3.31. Dịch vụ đã được chạy

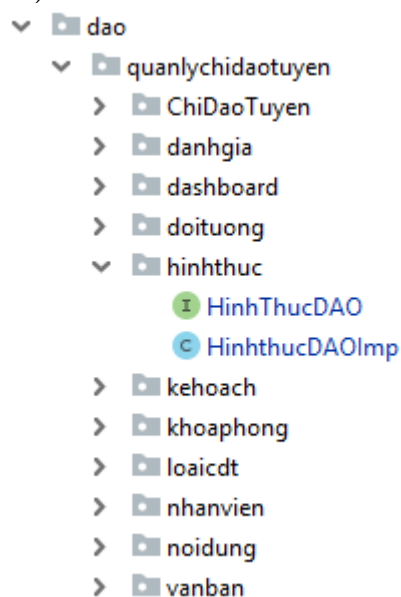
❖ Kết quả cài đặt hệ thống phân hệ chỉ đạo tuyến:

Trong gói Obj, tạo một gói có tên `quanlychidaotuyen`, sau đó cài đặt các đối tượng ánh xạ các bảng trong CSDL (hình 3.37).



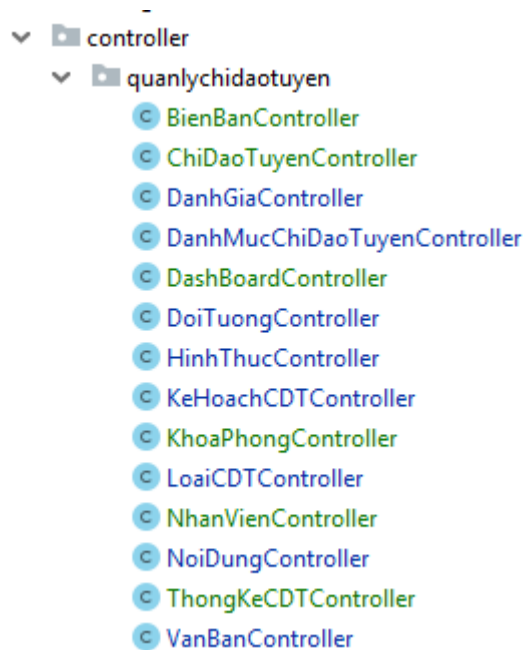
Hình 3.32. Kết quả cài đặt hệ thống phân hệ chỉ đạo tuyến

Trong gói dao, tạo một gói có tên quanlychidaotuyen, sau đó tạo thêm các gói sẽ chứa các interface, các class cài đặt để thực hiện các phương thức thao tác lên các bảng trong CSDL(hình 3.38).



Hình 3.33. Kết quả cài đặt hệ thống phân hệ chỉ đạo tuyến

Trong gói controller, tạo một gói có tên quanlychidaotuyen, sau đó cài đặt các class để tạo các đường dẫn và tiếp nhận các yêu cầu từ gateway, tạo các báo cáo,... (hình 3.39).



Hình 3.34. Kết quả cài đặt hệ thống phân hệ chỉ đạo tuyển

```

@RestController
@RequestMapping("/api")
@Api(value = "HinhThucController", description = "Danh mục hình thức Controller")
public class HinhThucController {
    @Autowired
    private HinhThucDAO hinhThucDAO;

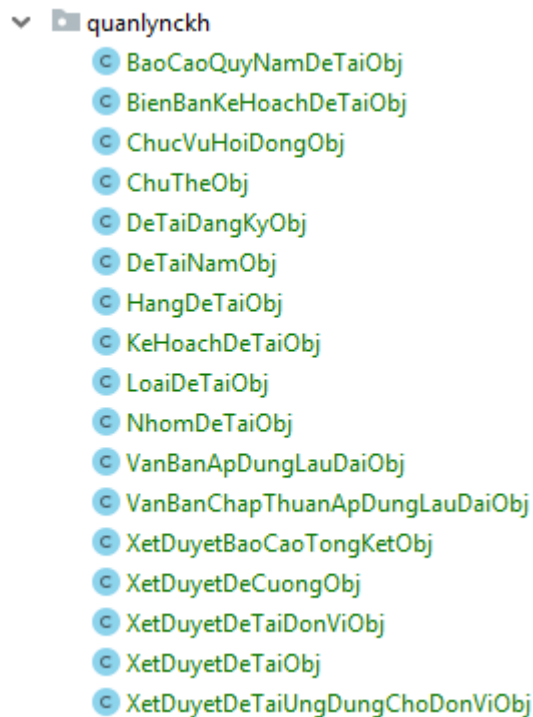
    @ApiOperation("Lấy danh sách hình thức")
    @RequestMapping(value = "hinh-thuc/lay-danh-sach-hinh-thuc",
        produces = "application/json; charset=utf-8",
        method = RequestMethod.GET)
    public @ResponseBody
    List danhSachHinhThuc() { return hinhThucDAO.layDanhSachHinhThuc(); }
}

```

Hình 3.35. Minh họa cài đặt lấy danh sách hình thức của dịch vụ chỉ đạo tuyển

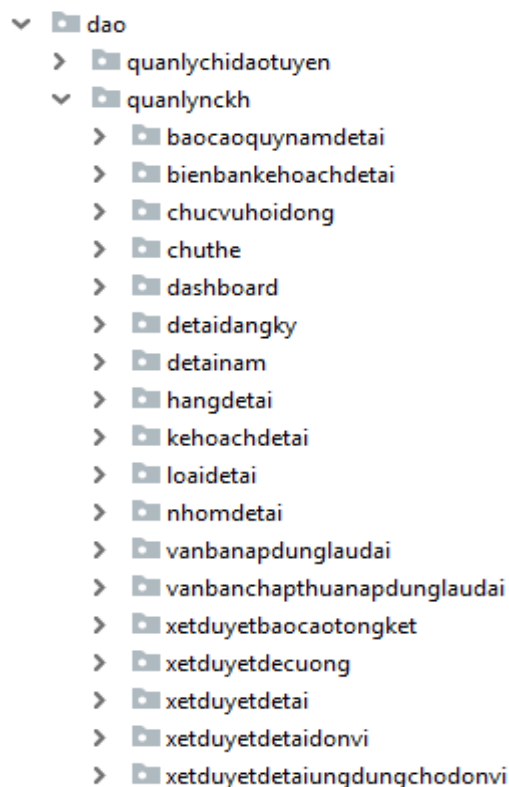
❖ Kết quả cài đặt hệ thống phân hệ chỉ đạo tuyển:

Trong gói Obj, tạo một gói có tên quanlynckh, sau đó cài đặt các đối tượng ánh xạ các bảng trong CSDL(hình 3.41).



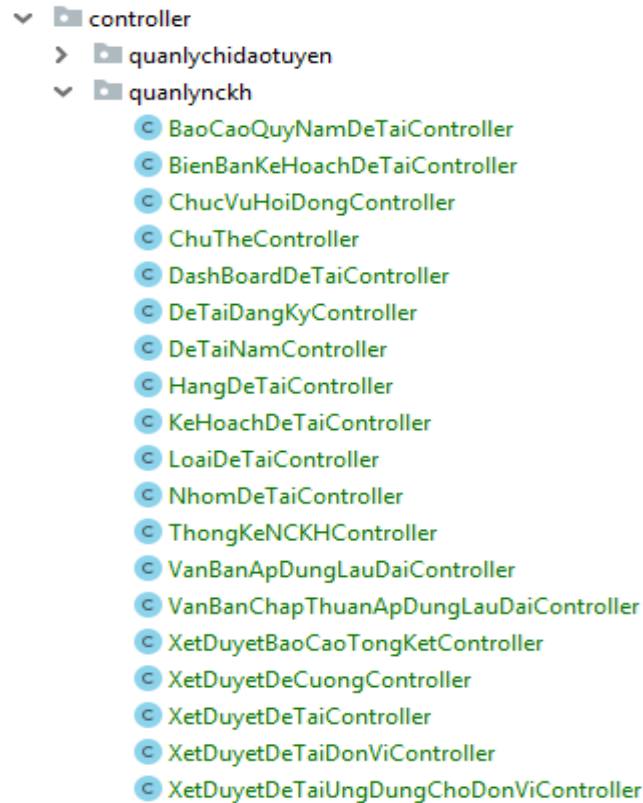
Hình 3.36. Kết quả cài đặt hệ thống phân hệ NCKH

Trong gói dao, tạo một gói có tên quanlynckh, sau đó tạo thêm các gói sẽ chứa các interface, các class cài đặt để thực hiện các phương thức thao tác lên các bảng trong CSDL(hình 3.42).



Hình 3.37. Kết quả cài đặt hệ thống phân hệ NCKH

Trong gói controller, tạo một gói có tên quanlynckh, sau đó cài đặt các class để tạo các đường dẫn và tiếp nhận các yêu cầu từ gateway, tạo các báo cáo,... (hình 3.43).



Hình 3.38. Kết quả cài đặt hệ thống phân hệ NCKH

```
@RestController
@RequestMapping("/api")
@Api(value = "LoaiDeTaiController", description = "Danh mục loại đề tài Controller")
public class LoaiDeTaiController {
    @Autowired
    private LoaiDeTaiDAO loaiDeTaiDAO;

    @ApiOperation("Lấy danh sách loại đề tài")
    @RequestMapping(value = "loai-de-tai/lay-danh-sach-loai-de-tai",
        produces = "application/json; charset=utf-8",
        method = RequestMethod.GET)
    public @ResponseBody
    List getDanhSachLoaiDeTai() { return loaiDeTaiDAO.getDanhSachLoaiDeTai(); }
```

Hình 3.39. Minh họa cài đặt lấy danh sách loại đề tài của dịch vụ NCKH

3.4.3. Cách cài đặt hệ thống

- Mở MySQL Workbench 8.0 CE, chọn File → Open SQL Script...Chọn đường dẫn đến tập tin db_chi_dao_tuyen.db trong đĩa CD, sau đó chọn Query → Execute (All or Selection).

- Thêm tất cả các modules vào IntelliJ IDEA theo hướng dẫn ở mục 3.4.2.

- Tại thanh Project, chọn đường dẫn tới tập tin application-dev.yml như sau:
service-chi-dao-tuyen/src/main/resources/config/ application-dev.yml
- Sửa đường dẫn của CSDL theo cấu hình của MySQL đã cài tại dòng url:

```
datasource:
  type: com.zaxxer.hikari.HikariDataSource
  url: jdbc:mysql://localhost:3306/db_chi_dao_tuyen?useUnicode=true&characterEncoding=utf8&useSSL=false
  username: root
  password: 1234
  hikari:
    poolName: Hikari
    auto-commit: false
```

Hình 3.40. Cách cài đặt hệ thống

- Tại Run Dashboard, nhấn chuột phải vào JHipsterRegistryApp, chọn Debug, tắt cả dịch vụ còn lại làm tương tự.
- Tại Terminal, di chuyển đến thư mục chứa dịch vụ gateway bằng lệnh cd duong_dan_thu_muc. Sau đó nhập lệnh: npm start và nhấn Enter, sau đó chờ cho dịch vụ chạy.
- Mở một trình duyệt, nhập đường dẫn: <http://localhost:9000/#/>, thực hiện đăng nhập và thực hiện các chức năng.

3.4.4. Một số hình ảnh giao diện hệ thống

❖ Giao diện đăng nhập

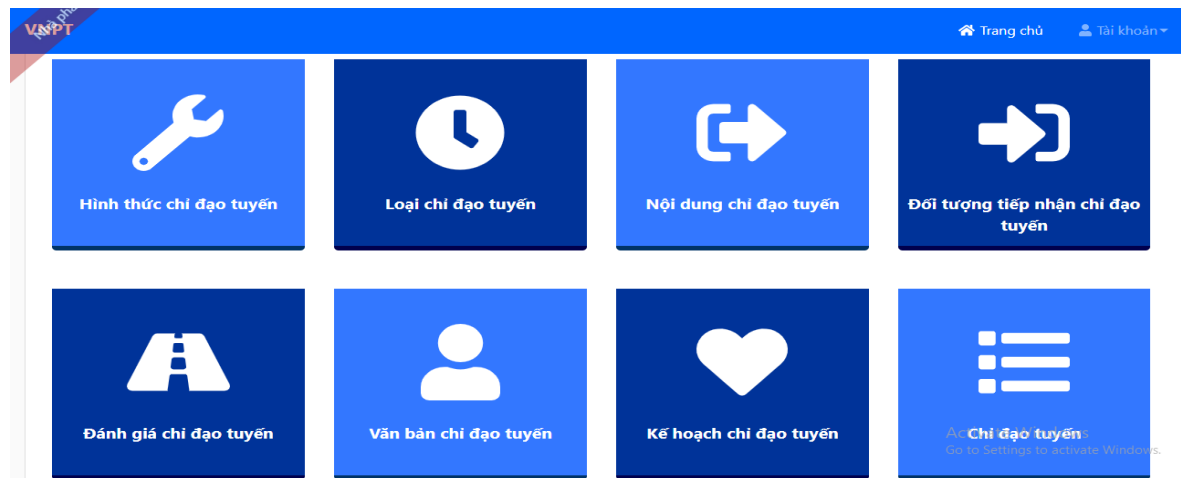
Đăng nhập với tài khoản, mật khẩu đã được cấp trước đó, tùy vào phân quyền mà hệ thống sẽ hiển thị chức năng.

Tài khoản: admin1

Mật khẩu: admin1

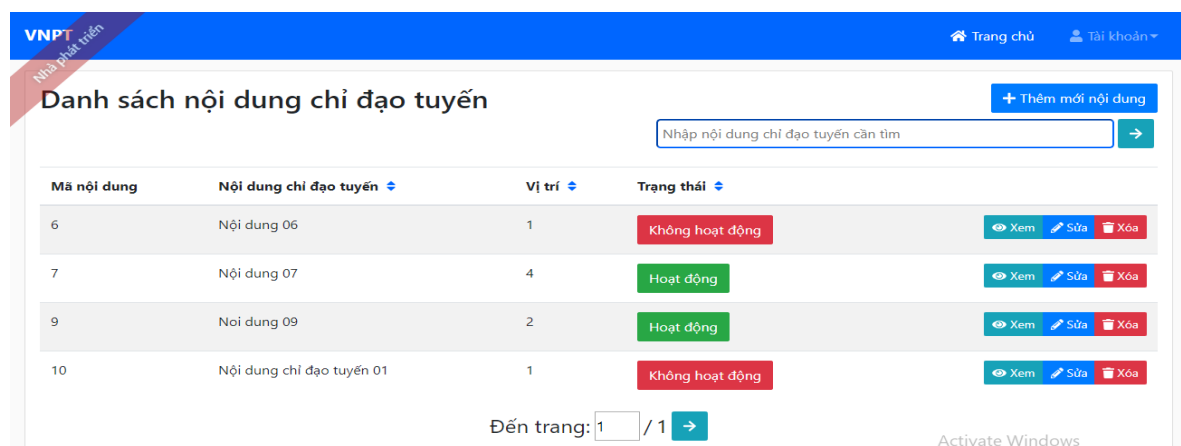
Hình 3.41. Giao diện đăng nhập

❖ **Giao diện màn hình chính với phân quyền Cán bộ quản lý chỉ đạo tuyển**



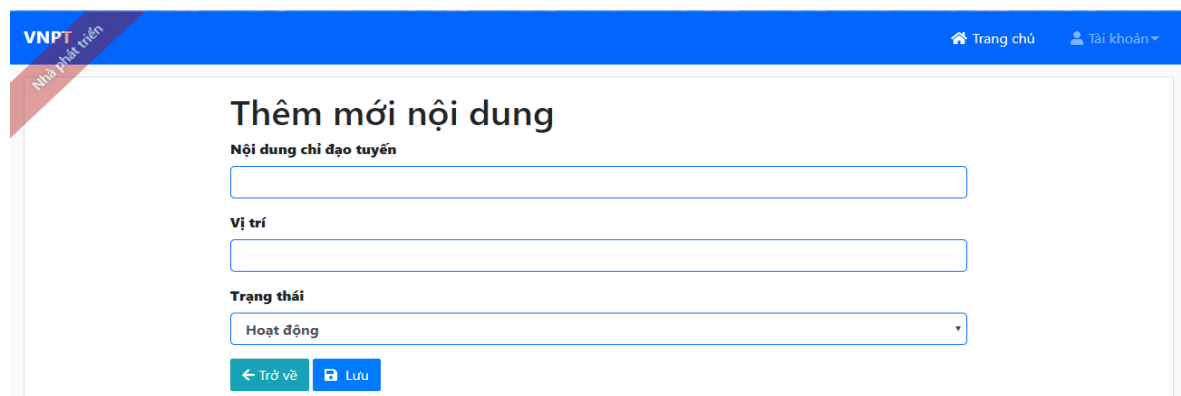
Hình 3.42. Giao diện chính

❖ **Giao diện Quản lý nội dung chỉ đạo tuyển**



Hình 3.43. Giao diện Quản lý nội dung chỉ đạo tuyển

❖ **Giao diện Thêm mới nội dung chỉ đạo tuyển**



Hình 3.44. Giao diện thêm mới nội dung chỉ đạo tuyển

❖ Giao diện Xem thông tin nội dung chỉ đạo tuyến

Thông tin nội dung

Mã nội dung
6

Tên nội dung
Nội dung 06

Vị trí
1

Trạng thái
Không hoạt động

← Trở về

Hình 3.45. Giao diện Xem thông tin nội dung chỉ đạo tuyến

❖ Giao diện Cập nhật nội dung chỉ đạo tuyến

Cập nhật nội dung

Mã nội dung
10

Nội dung chỉ đạo tuyến
Nội dung chỉ đạo tuyến 01

Vị trí
1

Trạng thái
Hoạt động

← Trở về Lưu

Hình 3.46. Giao diện Cập nhật thông tin nội dung chỉ đạo tuyến

❖ Giao diện Xóa nội dung chỉ đạo tuyến

Xác nhận xóa nội dung chỉ đạo tuyến

Xóa nội dung chỉ đạo tuyến này?

Xóa Không

Hình 3.47. Giao diện Xóa nội dung chỉ đạo tuyến

❖ Giao diện Thống kê chỉ đạo tuyến

Chọn thông tin chỉ đạo tuyến cần thống kê ×

Tỉnh thành Tiền Giang ▼

Đơn vị TRUNG TÂM Y TẾ TỈNH TIỀN GIANG ▼

Năm 2019 ▼

← Đóng
👁 In bảng

Hình 3.48. Giao diện Thống kê chỉ đạo tuyến

❖ Giao diện Xuất báo cáo chỉ đạo tuyến

In

Tổng số: 10 trang giấy

In
Hủy

Máy in đích Microsoft Print to P ▼

Trang Tất cả ▼

Màu Màu ▼

Cài đặt khác ▼

In bằng hộp thoại hệ thống...
(Ctrl+Shift+P) ☒

Ngày thực hiện: 09/06/2019

Nội dung trích yếu	Hình thức	Loại chỉ đạo tuyến	Tiếp nhận	Đơn vị tiếp nhận	Trưởng đoàn	Thư ký	Khoa phòng thực hiện	Đại diện
Nội dung trích yếu của kế hoạch 01	Hình thức 09	Loại chỉ đạo tuyến 10	3	BỆNH VIỆN ĐA KHOA TỈNH TIỀN GIANG	Nguyễn Văn A	Lê Thị B	Khoa Nhi	Nguyễn Thị D
NDTY test 002	Tên hình thức ht	Loại chỉ đạo tuyến 11	5	BỆNH VIỆN ĐA KHOA TỈNH TIỀN GIANG	Nguyễn Văn A	Lê Thị B	Khoa Nhi	Trần Văn C

Tổng số chỉ đạo tuyến trong ngày 09/06/2019 là: 2

Tổng số chỉ đạo tuyến trong danh sách này là: 20

Ngày.....tháng.....năm 20

Người in
(Ký, ghi rõ họ tên)

Hình 3.49. Giao diện Xuất báo cáo chỉ đạo tuyến

❖ Giao diện Dashboard



Hình 3.50. Giao diện Dashboard

Chương 4: KẾT QUẢ VÀ HƯỚNG PHÁT TRIỂN

4.1. Kết quả đạt được

Kết quả Tìm hiểu lý thuyết:

- Đã hiểu rõ mô hình Microservice, củng cố được kiến thức và nâng cao khả năng lập trình.
- Có thêm nhiều kiến thức về ReactJS, Redux, TypeScript, JasperReports, Spring.... Biết cách sử dụng nhiều thư viện.
- Tích lũy thêm kinh nghiệm phân tích, thiết kế hệ thống theo yêu cầu.
- Biết được quy trình chỉ đạo tuyển, quy trình đăng ký, xét duyệt đề tài trong ngành y tế.

Kết quả Ứng dụng thực nghiệm:

- Xây dựng được phần mềm thông qua quá trình nghiên cứu, khảo sát, phân tích thiết kế hệ thống, phân tích thiết kế cơ sở dữ liệu, đặc tả và thiết kế giao diện.
- Thực hiện thành công chức năng cơ bản đã đặt ra.
- Được chạy thử nghiệm, kiểm tra.

4.2. Hạn chế

Cấu hình hệ thống khá phức tạp.

Giao diện chưa được bắt mắt.

Chưa tận dụng được hết tối ưu của kiến trúc Microservice (chưa tách hai phân hệ ra thành hai service).

4.3. Hướng phát triển

Xây dựng hệ thống tương thích với nhiều thiết bị sử dụng trình duyệt khác nhau.

Cần tách hai phân hệ chỉ đạo tuyển và phân hệ NCKH ra thành hai dịch vụ với hai CSDL độc lập.

Sử dụng công nghệ đồng bộ dữ liệu cho hai CSDL.

DANH MỤC TÀI LIỆU THAM KHẢO

- [1]. Trịnh Minh Cường(11/10/2015), Giới thiệu về Microservices (kiến trúc nhiều dịch vụ nhỏ), <https://techmaster.vn/posts/33594/gioi-thieu-ve-microservices>, truy cập cuối cùng vào ngày 19/06/2019.
- [2]. Do KhanhToan(30/05/2017), Giới thiệu về ReactJS - Phần I (Các khái niệm cơ bản), <https://viblo.asia/p/gioi-thieu-ve-reactjs-phan-i-cac-khai-niem-co-ban-V3m5WzjblO7>, truy cập cuối cùng vào ngày 19/06/2019.
- [3]. Chris Ho(09/09/2017), Redux thật là đơn giản! (Phần 1), <https://insights.innovatube.com/redux-redux-thật-là-đơn-giản-phần-1-76a3fa2c31ab>, truy cập cuối cùng vào ngày 19/06/2019
- [4]. TypeScript – Overview, https://www.tutorialspoint.com/typescript/typescript_overview.htm, truy cập cuối cùng vào ngày 19/ 06/ 2019.
- [5]. Stackjava(19/12/2017), Tổng quan về Spring Framework, <https://viblo.asia/p/tong-quan-ve-spring-framework-YWOZryEyKQ0>, truy cập cuối cùng vào ngày 28/06/2019.
- [6]. HoangManh(20/03/2019), Giới thiệu về JHipster Stack, <https://viblo.asia/p/gioi-thieu-ve-jhipster-stack-6J3ZgmdL5mB>, truy cập cuối cùng vào ngày 19/06/2019.
- [1], Các tài liệu của JasperReports, <https://community.jaspersoft.com/documentation?version=54936>, truy cập cuối cùng vào 19/06/2019.