**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**ĐỒ ÁN 3 (PROJECT 3)**

**Xây dựng website tìm kiếm và cho thuê trọ**

**Đồng Xuân Phong**

phong.dx183966@sis.hust.edu.vn

**Ngành Công nghệ thông tin**

**Chuyên ngành Công nghệ phần mềm**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | ThS. Nguyễn Mạnh Tuấn |
| **Bộ môn:** | Công nghệ phần mềm |
| **Viện:** | Công nghệ thông tin – Truyền thông |
| **HÀ NỘI, 12/2021** | |

# Lời cam kết

Họ và tên sinh viên: Đồng Xuân Phong.. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Điện thoại liên lạc: 0346902568. . . . . . . . . . . Email: phong.dx183966@sis.hust.edu.vn. . . .

Lớp: CNTT 01. . . . . . . . . . . . . . . . . . . . . . . . . . . Hệ đào tạo: CNCN.. . . . . . . . . . . . . . . . . .

Tôi – *Đồng Xuân Phong* – cam kết Đồ án 3 (Project 3) (PROJECT 3) là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *ThS. Nguyễn Mạnh Tuấn*. Các kết quả nêu trong PROJECT 3 là trung thực, là thành quả của riêng tôi, không sao chép theo bất kỳ công trình nào khác. Tất cả những tham khảo trong PROJECT 3 – bao gồm hình ảnh, bảng biểu, số liệu, và các câu từ trích dẫn – đều được ghi rõ ràng và đầy đủ nguồn gốc trong danh mục tài liệu tham khảo. Tôi xin hoàn toàn chịu trách nhiệm với dù chỉ một sao chép vi phạm quy chế của nhà trường.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 27 tháng 12 năm 2021*  Tác giả PROJECT 3  Phong  *Đồng Xuân Phong* |

# Lời cảm ơn

Trước hết, em xin chân thành cảm ơn thầy giáo ThS. Nguyễn Mạnh Tuấn giảng viên Viện Công nghệ Thông tin và Truyền thông, Trường Đại học Bách Khoa Hà Nội đã tận tình giúp đỡ em trong quá trình thực hiện đồ án 3.

Em cũng xin chân thành cảm ơn các thầy cô giáo trong trường ĐHBKHN nói chung, các thầy cô trong Bộ môn CNTT nói riêng đã dạy cho em kiến thức về các môn đại cương cũng như các môn chuyên ngành, giúp em có được cơ sở lý thuyết vững vàng và tạo điều kiện giúp đỡ em trong suốt quá trình học tập.

Cuối cùng, em xin gửi lời cảm ơn tới trung tâm ITStudent đã giúp em trong việc tìm hiểu đề tài và cung cấp các tài liệu liên quan. Đồng thời em xin cảm ơn toàn thể bạn bè, người thân đã nhiệt tình ủng hộ, động viên và giúp đỡ để em có thể thực hiện tốt đề tài này cũng như đủ tự tin để thực hiện các đề tài tiếp theo hay làm việc sau này.

Mặc dù đã rất cố gắng, nhưng trong một khoảng thời gian có hạn và đây là lần đầu tiên thực hiện một đề tài, cũng như hạn chế về mặt kiến thức và kinh nghiệm của bản thân nên không thể tránh khỏi những thiếu sót.

Chính vì vậy, em rất mong nhận được sự góp ý của các thầy giáo, cô giáo cũng như của bạn bè và những người quan tâm đến đề tài được trình bày trong bài báo cáo này để giúp cho đề tài được hoàn thiện hơn. Đồng thời giúp em rút ra bài học và có nhiều kinh nghiệm hơn để thực hiện tốt hơn các đồ án sau cũng như làm việc sau này.

Em xin chân thành cảm ơn.

# Tóm tắt

Hàng năm, có rất nhiều sinh viên hay người đi làm từ mọi nơi đổ về thành phố để học tập và làm việc, đồng thời cũng có rất nhiều người có nhu cầu cho thuê phòng/nhà trọ mà hiện nay họ không sử dụng nhằm kiếm thêm thu nhập vì thế nên kéo theo nhu cầu sử dụng/cho thuê trọ càng tăng cao. Đặc biệt vào giai đoạn dịch Covid 19 hiện tại cũng có rất nhiểu người quyết định trở về quê thay vì tiếp tục bám trụ lại các thanh phố lớn làm cho càng có nhiều căn hộ, phòng trọ không được sử dụng. Mặc dù hiện nay cũng có không ít các trang, ứng dụng phục vụ vấn đề thuê trọ nhưng chủ yếu người chủ trọ/người đi thuê vẫn lựa chọn thông qua người quen hoặc các hội nhóm trên các diễn đàn như facebook,… để tìm kiếm thông tin. Các ứng dụng hay website không đảm bảo được tinh xác thực các thông tin, cũng như thông tin không được cập nhật một cách liên tục. Chính vì vậy, em chọn đề tài “Xây dựng website tìm kiếm và cho thuê trọ” nhằm cung cấp một công cụ giúp chủ trọ và người cần thuê trọ có thể kết nối với nhau một cách dễ dàng và thuận tiện hơn. Đối tượng người sử dụng chính của website sẽ là người cần thuê trọ và chủ các phòng trọ.

# Mục lục

[Lời cam kết ii](#_Toc92982650)

[Lời cảm ơn iii](#_Toc92982651)

[Tóm tắt iv](#_Toc92982652)

[Mục lục v](#_Toc92982653)

[Danh mục hình vẽ viii](#_Toc92982654)

[Danh mục bảng x](#_Toc92982655)

[Danh mục các từ viết tắt xi](#_Toc92982656)

[Danh mục thuật ngữ xii](#_Toc92982657)

[Chương 1 Giới thiệu đề tài 1](#_Toc92982658)

[1.1 Đặt vấn đề 1](#_Toc92982659)

[1.2 Mục tiêu và phạm vi đề tài 1](#_Toc92982660)

[1.3 Định hướng giải pháp 2](#_Toc92982661)

[1.4 Bố cục đồ án 2](#_Toc92982662)

[Chương 2 Khảo sát và phân tích yêu cầu 4](#_Toc92982663)

[2.1 Khảo sát hiện trạng 4](#_Toc92982664)

[2.2 Tổng quan chức năng 4](#_Toc92982665)

[2.2.1 Biểu đồ use case tổng quan 4](#_Toc92982666)

[2.2.2 Biểu đồ use case phân rã 5](#_Toc92982667)

[2.2.3 Quy trình nghiệp vụ 9](#_Toc92982668)

[2.3 Đặc tả chức năng 10](#_Toc92982669)

[2.3.1 Đặc tả use case “Tìm kiếm” 10](#_Toc92982670)

[2.3.2 Đặc tả use case “Yêu cầu thuê trọ” 11](#_Toc92982671)

[2.3.3 Đặc tả use case “Bình luận – đanh giá trọ” 12](#_Toc92982672)

[2.3.4 Đặc tả use case “Thêm mới trọ cho thuê” 13](#_Toc92982673)

[2.3.5 Đặc tả use case “Hủy thuê trọ” 14](#_Toc92982674)

[2.3.6 Đặc tả use case “Liên hệ quản trị viên” 15](#_Toc92982675)

[2.4 Yêu cầu phi chức năng 16](#_Toc92982676)

[Chương 3 Công nghệ sử dụng 17](#_Toc92982677)

[3.1 Spring Boot 17](#_Toc92982678)

[3.2 Mysql Databse 19](#_Toc92982679)

[Chương 4 Phát triển và triển khai ứng dụng 21](#_Toc92982680)

[4.1 Thiết kế kiến trúc 21](#_Toc92982681)

[4.1.1 Lựa chọn kiến trúc phần mềm 21](#_Toc92982682)

[4.1.2 Thiết kế tổng quan 22](#_Toc92982683)

[4.1.3 Thiết kế chi tiết gói 23](#_Toc92982684)

[4.2 Thiết kế chi tiết 27](#_Toc92982685)

[4.2.1 Thiết kế giao diện và minh họa 27](#_Toc92982686)

[4.2.2 Thiết kế lớp 37](#_Toc92982687)

[4.2.3 Thiết kế cơ sở dữ liệu 43](#_Toc92982688)

[4.3 Xây dựng ứng dụng 50](#_Toc92982689)

[4.3.1 Thư viện và công cụ sử dụng 50](#_Toc92982690)

[4.3.2 Kết quả đạt được 51](#_Toc92982691)

[4.3.3 Minh hoạ các chức năng chính 51](#_Toc92982692)

[4.4 Kiểm thử 51](#_Toc92982693)

[4.5 Triển khai 52](#_Toc92982694)

[Chương 5 Các giải pháp và đóng góp nổi bật 53](#_Toc92982695)

[Chương 6 Kết luận và hướng phát triển 55](#_Toc92982696)

[6.1 Kết luận 55](#_Toc92982697)

[6.2 Hướng phát triển 55](#_Toc92982698)

[Tài liệu tham khảo 56](#_Toc92982699)

# Danh mục hình vẽ

[Hình 1. Biểu đồ use case tổng quan 5](#_Toc92982600)

[Hình 2. Phân rã use case “Tìm kiếm” 5](#_Toc92982601)

[Hình 3. Phân rã use case “Quản lý thông tin tài khoản” 6](#_Toc92982602)

[Hình 4. Phân rã use case “Quản lý trọ đã thuê” 6](#_Toc92982603)

[Hình 5. Phân rã use case “Quản lý trọ đã cho thuê” 7](#_Toc92982604)

[Hình 6. Phân rã use case “Quản lý người thuê trọ” 7](#_Toc92982605)

[Hình 7. Phân rã use case “Quản lý danh mục và tiện ích” 8](#_Toc92982606)

[Hình 8. Phân rã use case “Quản lý người dùng” 8](#_Toc92982607)

[Hình 9. Phân rã use case “Quản lý trọ trong hệ thống” 9](#_Toc92982608)

[Hình 10. Quy trình nghiệp vụ chính của người thuê 9](#_Toc92982609)

[Hình 11. Quy trình nghiệp vụ chính của chủ trọ 10](#_Toc92982610)

[Hình 12. Sơ đồ kiến trúc tổng thể hệ thống 22](#_Toc92982611)

[Hình 13. Sơ đồ tổng quan tầng trình bày 23](#_Toc92982612)

[Hình 14. Sơ đồ tổng quan tầng ứng dụng 24](#_Toc92982613)

[Hình 15. Sơ đồ tổng quan tầng nghiệp vụ 25](#_Toc92982614)

[Hình 16. Sơ đồ tổng quan tầng truy xuất và lưu trữ dữ liệu 26](#_Toc92982615)

[Hình 17. Giao diện quản lý yêu cầu thuê của chủ trọ 28](#_Toc92982616)

[Hình 18. Giao diện quản lý trọ của chủ trọ 29](#_Toc92982617)

[Hình 19. Giao diện thêm mới/chỉnh sửa trọ 31](#_Toc92982618)

[Hình 20. Giao diện thông tin người dùng và quản lý trọ đã thuê/yêu cầu thuê 33](#_Toc92982619)

[Hình 21. Giao diện tìm kiếm 35](#_Toc92982620)

[Hình 22. Class MotelRoomController 37](#_Toc92982621)

[Hình 23. Class TenantController 38](#_Toc92982622)

[Hình 24. Class MotelRoom 39](#_Toc92982623)

[Hình 25. Class Tenant & TenantKey 40](#_Toc92982624)

[Hình 26. Biểu đồ trình tự ca sử dụng “Tìm kiếm” 41](#_Toc92982625)

[Hình 27. Biểu đồ trình tự ca sử dụng “Yêu cầu thuê trọ” 41](#_Toc92982626)

[Hình 28. Biểu đồ trình tự ca sử dụng “Thêm mới trọ cho thuê” 42](#_Toc92982627)

[Hình 29. Sơ đồ quan hệ các bảng dữ liệu 43](#_Toc92982628)

# Danh mục bảng

[Bảng 1. Đặc tả use case “Tìm kiếm” 11](#_Toc92982629)

[Bảng 2. Đặc tả use case “Yêu cầu thuê trọ” 12](#_Toc92982630)

[Bảng 3. Đặc tả use case “Bình luận - đánh giá trọ” 13](#_Toc92982631)

[Bảng 4. Đặc tả use case “Thêm mới trọ cho thuê” 14](#_Toc92982632)

[Bảng 5. Đặc tả use case “Hủy thuê trọ” 15](#_Toc92982633)

[Bảng 6. Đặc tả use case “Liên hệ với quản trị viên” 15](#_Toc92982634)

[Bảng 7. Đặc tả giao diện quản lý yêu cầu thuê của chủ trọ 29](#_Toc92982635)

[Bảng 8. Đặc tả giao diện quản lý trọ của chủ trọ 30](#_Toc92982636)

[Bảng 9. Đặc tả giao diện thêm mới/chỉnh sửa trọ 32](#_Toc92982637)

[Bảng 10. Đặc tả giao diện thông tin người dùng và quản lý trọ đã thuê/yêu cầu thuê 34](#_Toc92982638)

[Bảng 11. Đặc tả giao diện tìm kiếm 36](#_Toc92982639)

[Bảng 12. Bảng dữ liệu người dùng (user) 44](#_Toc92982640)

[Bảng 13. Bảng dữ liệu trọ cho thuê (motel\_room) 46](#_Toc92982641)

[Bảng 14. Bảng dữ liệu danh mục (category) 46](#_Toc92982642)

[Bảng 15. Bảng dữ liệu tiện ích (convenient) 47](#_Toc92982643)

[Bảng 16. Bảng dữ liệu phân quyền (role) 47](#_Toc92982644)

[Bảng 17. Bảng dữ liệu bình luận (report) 48](#_Toc92982645)

[Bảng 18. Bảng dữ liệu tiện ích của trọ (room\_has\_convenient) 49](#_Toc92982646)

[Bảng 19. Bảng dữ liệu thuê trọ (tenant) 50](#_Toc92982647)

[Bảng 20. Danh sách công cụ sử dụng 50](#_Toc92982648)

[Bảng 21. Thông tin ứng dụng 51](#_Toc92982649)

# Danh mục các từ viết tắt

|  |  |
| --- | --- |
| API | Application Programming Interface  Giao diện lập trình ứng dụng |
| **HTML** | HyperText Markup Language  Ngôn ngữ đánh dấu siêu văn bản |
| **CNTT** | Công nghệ thông tin |
| **PROJECT 3** | Đồ án 3 (Project 3) |
| **SV** | Sinh viên |
| **DTO** | Data transfer object  Đối tượng mang dữ liệu giữa các quy trình |
| **CSDL** | Cơ sở dữ liệu |
| **JPA** | Java Persitent API |

# Danh mục thuật ngữ

|  |  |
| --- | --- |
| Browser | Trình duyệt |
| **Cache memory** | Bộ nhớ đệm |
| **E-commerce** | Thương mại điện tử |
| **Bloatware** | Ứng dụng nhà sản xuất tích hợp vào thiết bị |
| **Interpreter** | Trình thông dịch |
| **Compiler** | Trình biên dịch |

# Giới thiệu đề tài

## Đặt vấn đề

Chúng ta đang sống và làm việc trong thời kì công nghiệp hoá, hiện đại hoá, thời kỳ công nghệ thông tin đang phát triển trên phạm vi toàn cầu. Các sản phẩm của ngành công nghệ thông tin là phần cứng, phần mềm và trí tuệ nhân tạo đang được ứng dụng vào hầu hết các lĩnh vực và các ngành nghề khác nhau của xã hội.

Hàng năm, có rất nhiều sinh viên hay người đi làm từ mọi nơi đổ về thành phố để học tập và làm việc, đồng thời cũng có rất nhiều người có nhu cầu cho thuê phòng/nhà trọ mà hiện nay họ không sử dụng nhằm kiếm thêm thu nhập vì thế nên kéo theo nhu cầu sử dụng/cho thuê trọ càng tăng cao. Tuy nhà trọ cho thuê rất nhiều, nhưng giữa một ma trận thông tin, mọi người khó có thể lựa chọn được cho mình một nhà trọ uy tín, cũng giống như vậy lượng người có nhu cầu thuê trọ cũng rất đông nhưng lại không dễ để có thể quảng bá về trọ thuê cho họ biết. Nếu là người mới chân ướt chân ráo đặt chân lên thành phố như những sinh viên thì quả thật còn khó khăn gấp ngàn lần. Vậy làm thế nào để có thể tìm được nhà trọ tốt, tìm phòng cho thuê đáp ứng được nhu cầu học tập, sinh hoạt và làm việc có giá phải chăng, phù hợp với nguồn tài chính của cá nhân, cũng như có thể quảng cáo phòng/nhà trọ hiện nay không sử dụng cần cho thuê?

Xuất phát từ nhu cầu đó, em đã chọn đề tài “Xây dựng website tìm kiếm và cho thuê trọ” nhằm cung cấp một công cụ giúp chủ trọ và người cần thuê trọ có thể kết nối với nhau một cách dễ dàng và thuận tiện hơn. Đối tượng người sử dụng chính của website sẽ là người cần thuê trọ và chủ các phòng trọ.

## Mục tiêu và phạm vi đề tài

*Mục tiêu đề tài*

Mục tiêu của đề tài là tạo ra một trang web tìm kiếm và cho thuê trọ giúp cho người thuê trọ có thể dễ dàng tìm được trọ phù hợp và chủ trọ có thể đăng tải hay quảng cáo trọ mà họ cho thuê một cách rộng rãi để có thể tiếp cận đến người có nhu cầu thuê trọ.

Dự kiến website sẽ đạt được những mục tiêu chính sau:

* Đăng nhập, đăng ký, quản lý thông tin cá nhân.
* Tìm kiếm, đánh giá, bình luận, xem thông tin trọ và thuê trọ.
* Đăng bài, chỉnh sửa thông tin trọ muốn cho thuê.
* Thống kê đối với chủ trọ và quản lý.

Trang web tìm kiếm và cho thuê trọ giúp cho người thuê trọ và chủ trọ có thể kết nối với nhau một cách dễ dàng và thuận tiện hơn.

*Phạm vi đề tài*

* Đối với khách:
  + Khi chưa có tài khoản thì chỉ có thể xem thông tin và tìm kiếm trọ.
  + Khi đã có tài khoản thì có thể thuê trọ, hủy thuê/ yêu cầu thuê trọ hiện tại, cũng như bình luận và đánh giá trọ.
* Đối với chủ trọ: có thể đăng bài, chỉnh sửa thông tin trọ mình cho thuê, chấp nhận hoặc hủy yêu cầu thuê trọ, xem thông tin người thuê trọ của mình và xem thống kê trọ hiện tại mà mình quản lý.
* Đối với quản lý: có thể quản lý người dùng trong hệ thống (gồm cả chủ trọ và người thuê), quản lý trọ cho thuê trong hệ thống và thống kê toàn bộ trọ có trong hệ thống.

## Định hướng giải pháp

Từ việc xác định rõ nhiệm vụ cần giải quyết ở phần 1.2, em quyết định xây dựng đồ án dựa trên mô hình MVC (Model-View-Controller). Phân tích các usecase chính, từ đó thiết kế các quy trinh nghiệp vụ liên quan để giải quyết các bài toán mà đề tài đưa ra nhằm đáp ứng được các nhu cầu cần thiết cho người dùng khi sử dụng hệ thống.

## Bố cục đồ án

Phần còn lại của báo cáo Đồ án 3 (Project 3) này được tổ chức như sau.

Chương 2 trình bày về khảo sát đề tài và đưa ra tổng quan chức năng mà hệ thống cần có để tạo trải nghiệm tốt cho người dùng. Thông qua những hệ thống thực có trên thị trường, em sẽ có một cái nhìn tổng quan về hệ thống. Dựa trên đó em sẽ xác định và phân tích các chức năng cần thiết bằng các sơ đồ usecase, biểu đồ quy trình nghiệp vụ, đặc tả các ca sử dụng,… Từ những phân tích cơ bản đó em sẽ lựa chọn các công nghệ sử dụng trong Chương 3.

Chương 3, em giới thiệu về các công nghệ được sử dụng cũng như các nền tảng lý thuyết cần có để xây dựng hệ thống. Cụ thể trong đồ án này em sử dụng Spring Boot + Thymeleaf và MysqlDB. Từ những công nghệ này, em tiếp tục phân tích chi tiết hệ thống và triển khai ứng dụng trong chương tiếp theo - [Chương 4](#_Phát_triển_và)

[Chương 4](#_Phát_triển_và) sẽ trình bày về phát triển và triển khai ứng dụng, thiết kế kiến trúc, lớp, và đồng thời xây dựng mã nuồn của hệ thống. Cụ thể trong đồ án này sử dụng mô hình MVC (Model – View – Controller), phân tích, thiết kế chi tiết các gói thuộc các tầng, sự phụ thuộc của chúng và tương ứng với mỗi gói sẽ có các lớp tương ứng. Đồng thời, trong chương 4 này cũng trình bày về cụ thể, chi tiết về cơ sở dữ liệu, các bảng trong cơ sở dữ liệu và đặc tả các bảng đem lại một cái nhìn cụ thể nhất về cơ sở dữ liệu. Dựa trên những gì đã phân tích trên, em đã xây dựng hệ thống thực tế đầy đủ các chức năng cơ bản đã đề ra và trình bày kết quả kiểm thử, minh họa các chức năng chính trong hệ thống đã được xây dựng. Trong việc xây dựng ứng dụng, khó khăn là điều không thể tránh khỏi. Chương tiếp theo là [Chương 5](#_Các_giải_pháp) sẽ nêu lên những khó khăn trên và phương hướng giải quyết.

[Chương 5](#_Các_giải_pháp) nêu ra các giải pháp và đóng góp nổi bật khi xây dựng hệ thống, tạo nên sự đặc biệt cho ứng dụng. Tổng hợp các chương từ 1 đến 5 sẽ đưa ra các kết luận ở [Chương 6](#_Kết_luận_và) về toàn bộ quá trình phát triển hệ thống.

Cuối cùng là [Chương 6](#_Kết_luận_và), em đưa ra kết luận về ưu/nhược đã thực hiện qua đồ án, những gì em đã thực hiện được, chưa thực hiện được và tiến hành so sánh sản phẩm của mình với những sản phẩm thực tế từ đó đưa ra những điểm còn thiếu sót của sản phẩm và rút ra những kinh nghiệm cho bản thân khi xây dựng ứng dụng này. Thông qua quá trình đánh giá đó, em cũng rút ra được những định hướng công việc trong tương lai để có thể hoàn thiện, phát triển sản phẩm của mình.

# ­­Khảo sát và phân tích yêu cầu

## Khảo sát hiện trạng

Qua quá trình khảo sát hiện trạng, đa phần người đi thuê trọ sẽ đi tìm thông qua người quen hoặc qua các group/hội nhóm trên các mạng xã hội để tìm kiếm thông tin. Hay người cho thuê trọ cũng vậy đa phần quảng cáo thông qua người quen hoặc đăng lên các mạng xã hội. Hầu hết cả 2 đối tượng chính trên không thường xuyên sử dụng các ứng dụng chuyên biệt về thuê trọ để tìm kiếm hoặc đăng bài mà đều sử dụng các công cụ khác dẫn đến một vài nhược điểm như sau: khó kiểm soát hết các thông tin, so sánh giá cả hay vị trí của các địa điểm, đặc biệt khó xác định về tính chính xác của thông tin,…

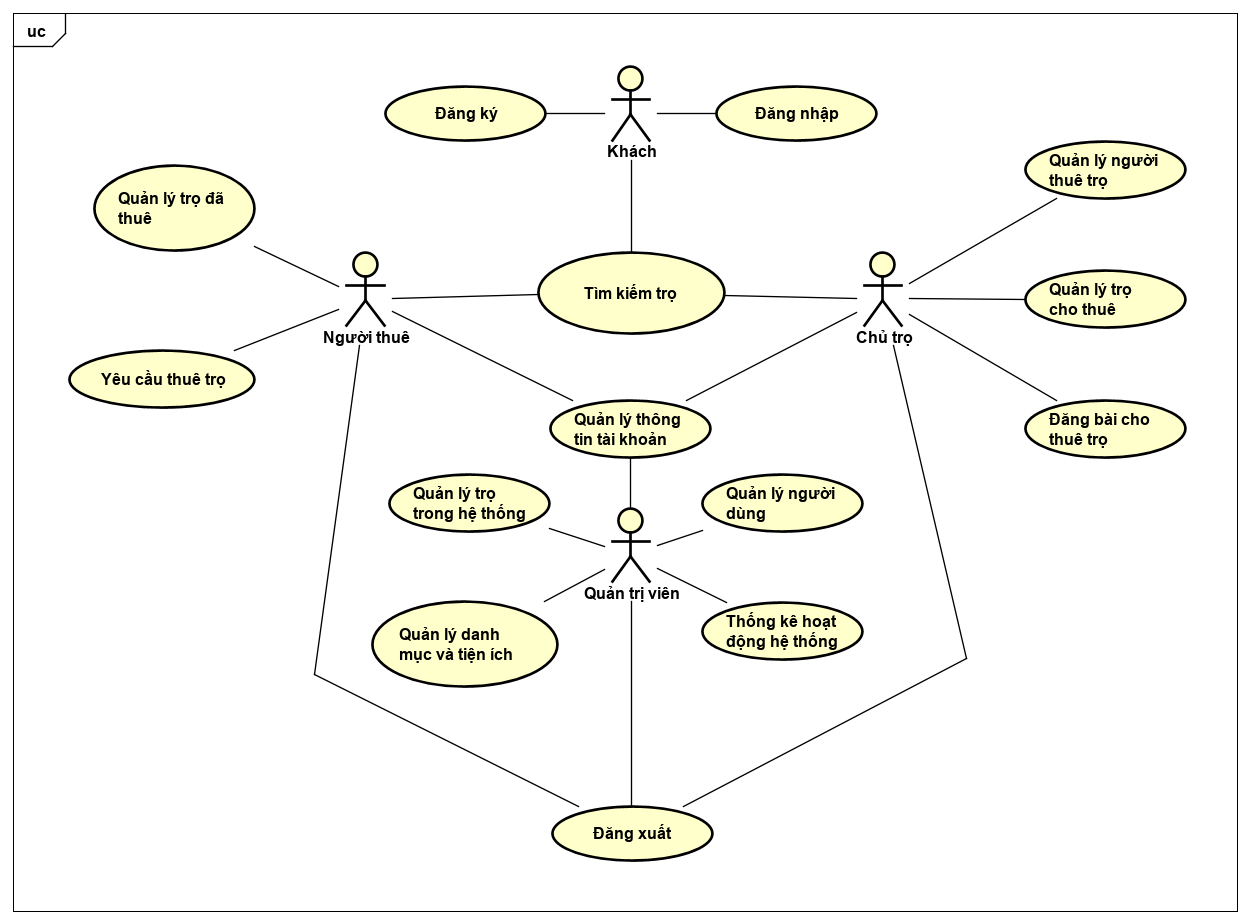
Qua đó, muốn thay đổi thói quen này của người dùng thì cần tạo sự tin tưởng về các thông tin có trên ứng dụng. Vì vậy em thực hiện đề trên nhằm đáp ứng nhu cầu của người dùng.

## Tổng quan chức năng

### Biểu đồ use case tổng quan

Hệ thống gồm 4 tác nhân: Khách, Người thuê trọ, Chủ trọ và Quản trị viên.

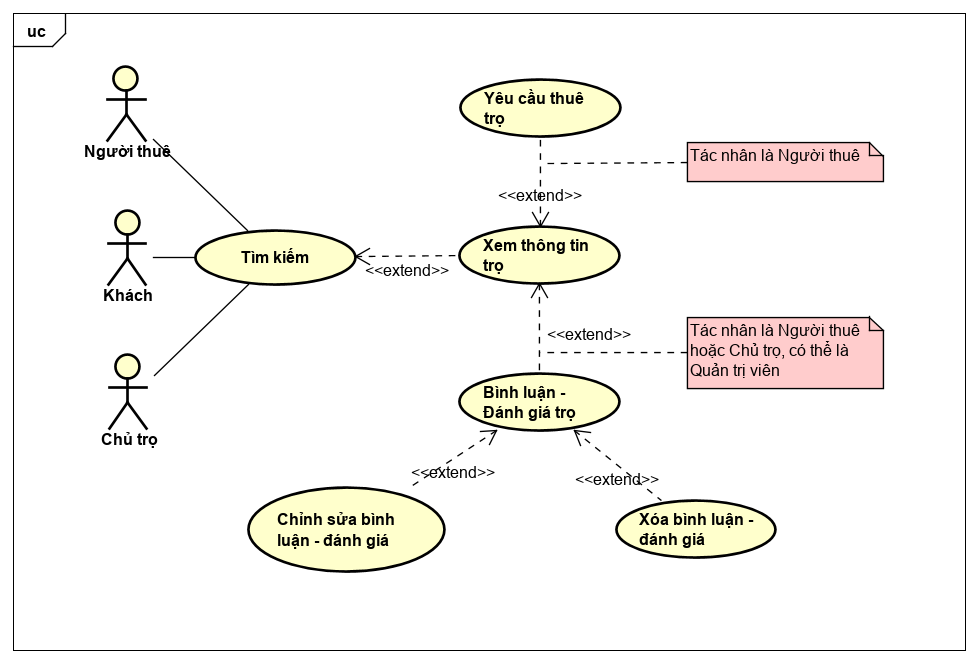
* Khách: Là tác nhân chưa đăng nhập vào hệ thống, có thể đăng nhập, đăng ký tài khoản, tìm kiếm và xem thông tin các phòng trọ hiện có trong hệ thống.
* Người thuê trọ: là tác nhân đăng nhập vào hệ thống với vai trò người thuê, có thể quản lý thông tin tài khoản, tìm kiếm, xem thông tin và yêu cầu thuê/hủy thuê phòng trọ.
* Chủ trọ: là tác nhân đăng nhập vào hệ thống với vai trò chủ trọ, có thể quản lý thông tin tài khoản, đăng bài cho thuê trọ và quản lý thông tin cũng như chấp nhận/hủy yêu cầu thuê trọ mà mình cho thuê.
* Quản trị viên: là tác nhân đăng nhập vào hệ thống với vai trò quản trị viên, quản trị viên có thể quản lý quản lý người dùng trong hệ thống (người thuê và người cho thuê) và phòng trọ hiện có trong hệ thống.

**

Hình 1. Biểu đồ use case tổng quan

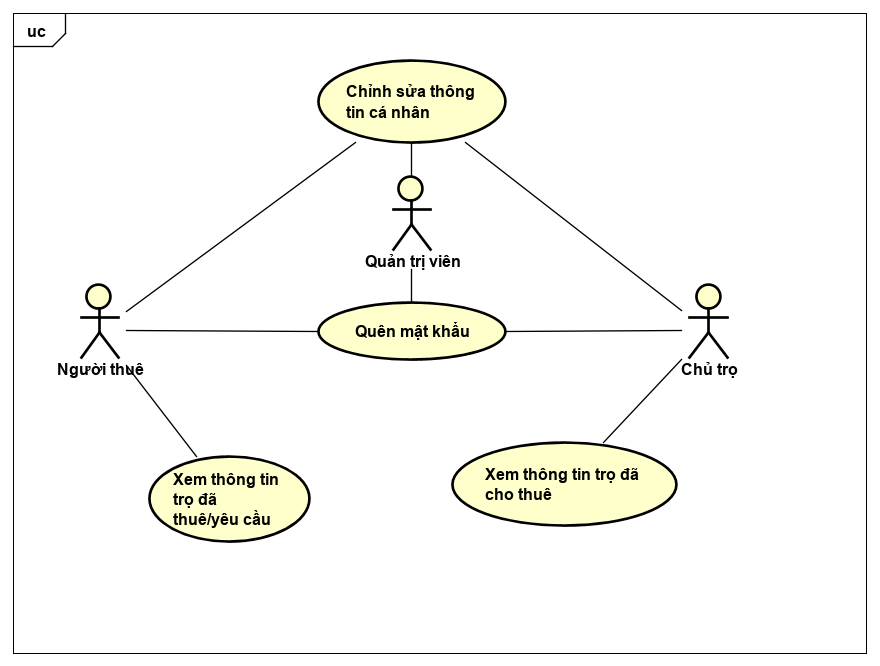
### Biểu đồ use case phân rã

Phân rã use case “Tìm kiếm”



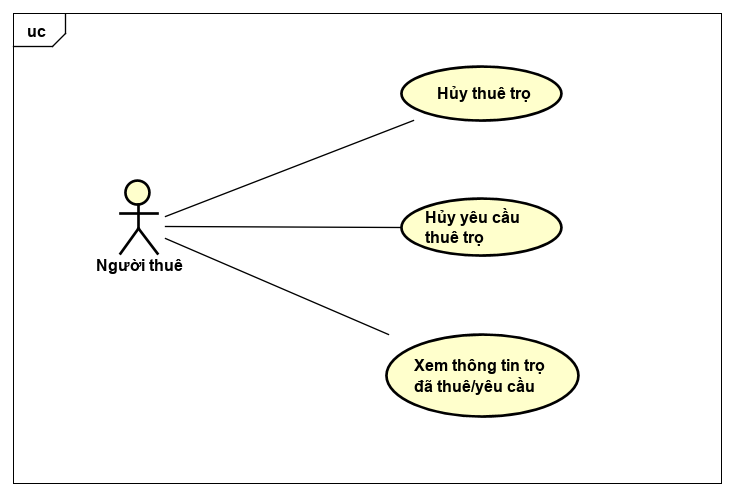
Hình 2. Phân rã use case “Tìm kiếm”

Phân rã use case “Quản lý thông tin tài khoản”



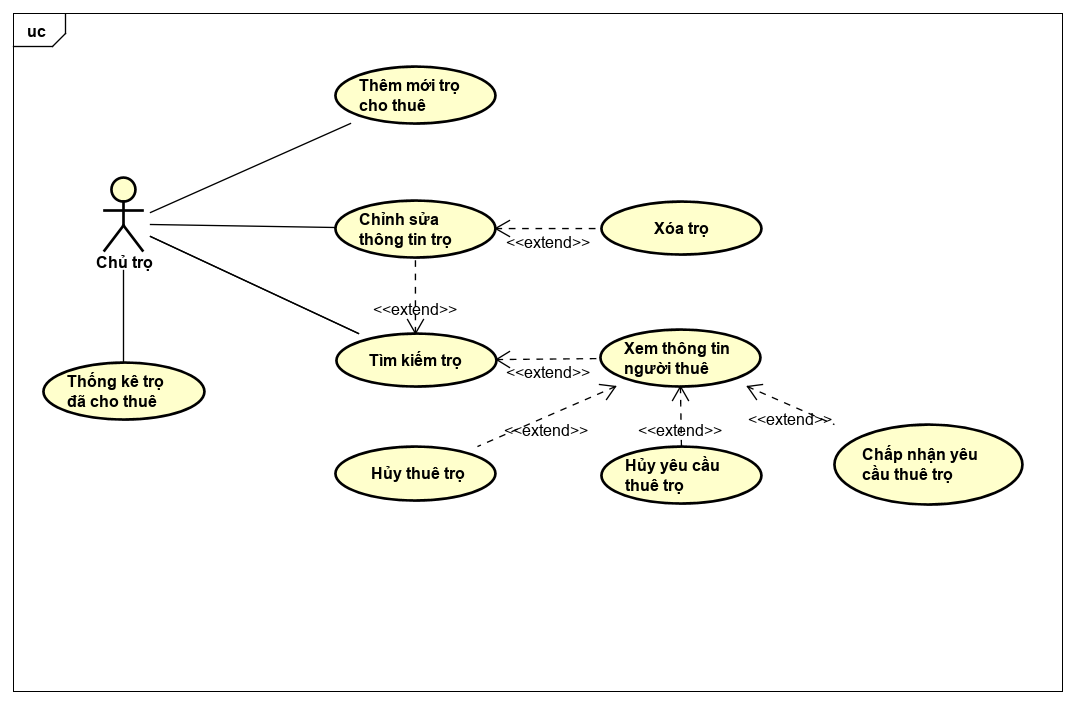
Hình 3. Phân rã use case “Quản lý thông tin tài khoản”

Phân rã use case “Quản lý trọ đã thuê”



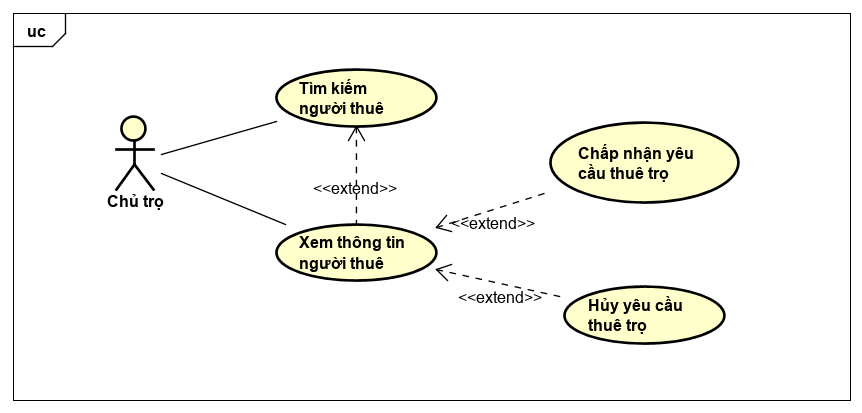
Hình 4. Phân rã use case “Quản lý trọ đã thuê”

Phân rã use case “Quản lý trọ đã cho thuê”



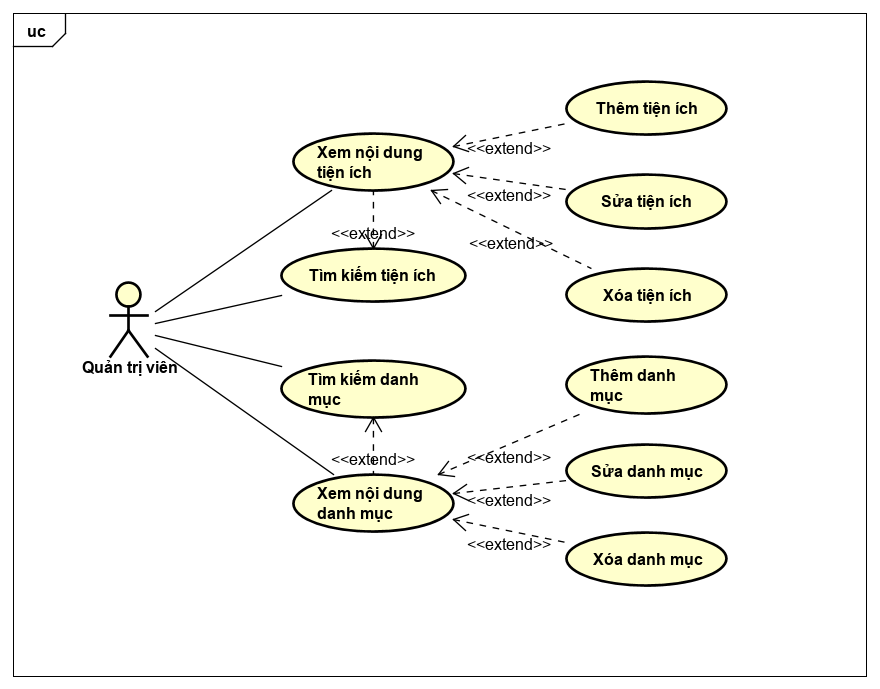
Hình 5. Phân rã use case “Quản lý trọ đã cho thuê”

Phân rã use case “Quản lý người thuê trọ”



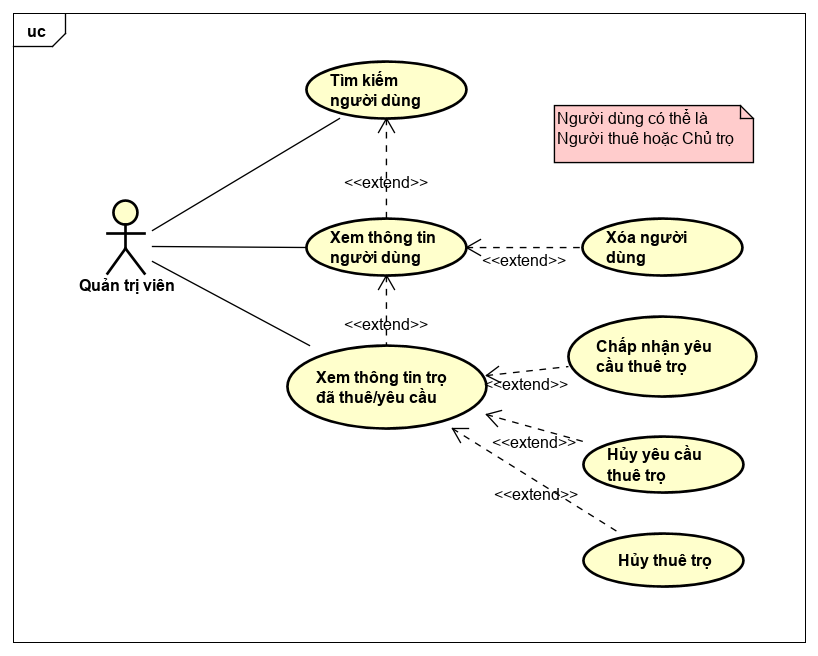
Hình 6. Phân rã use case “Quản lý người thuê trọ”

Phân rã use case “Quản lý danh mục và tiện ích”



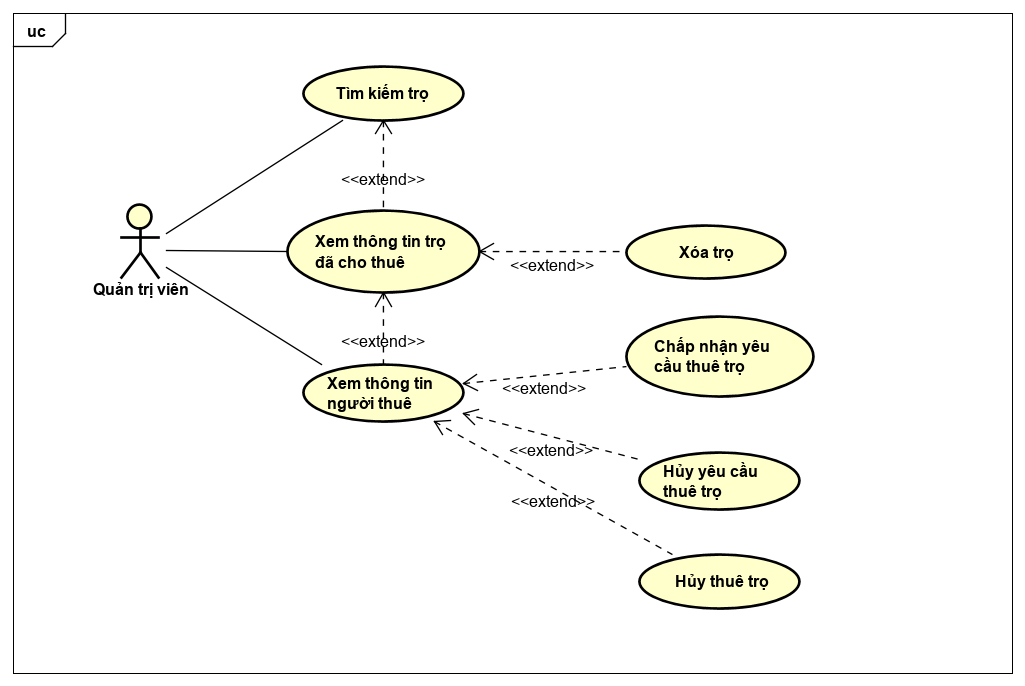
Hình 7. Phân rã use case “Quản lý danh mục và tiện ích”

Phân rã use case “Quản lý người dùng”



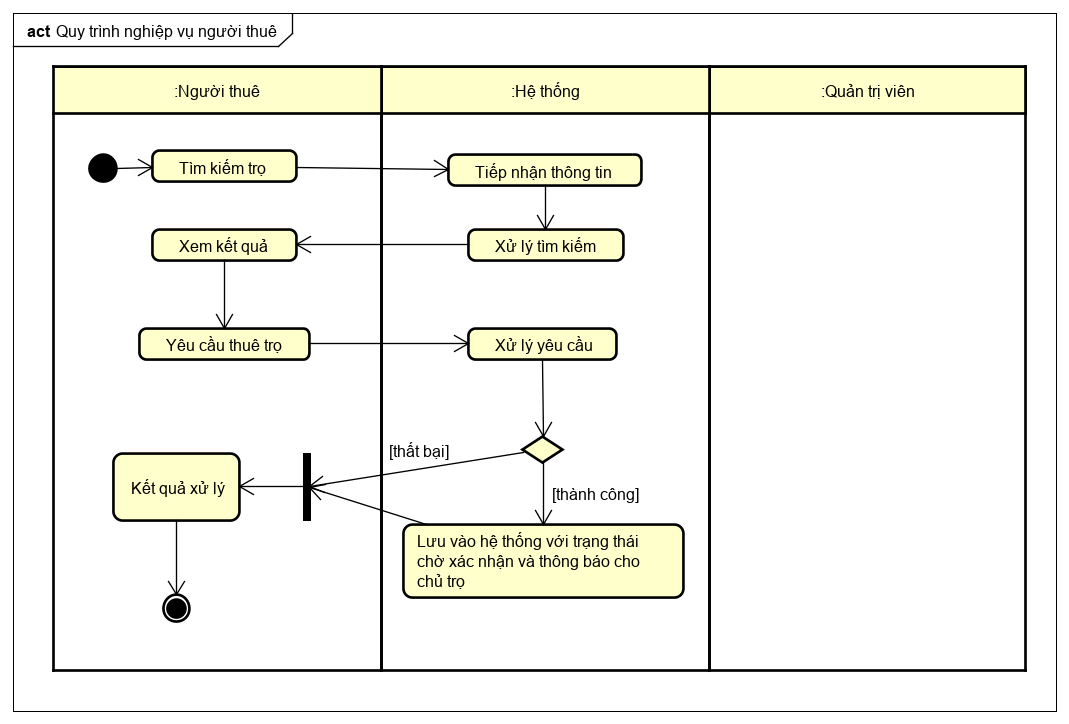
Hình 8. Phân rã use case “Quản lý người dùng”

Phân rã use case “Quản lý trọ trong hệ thống”

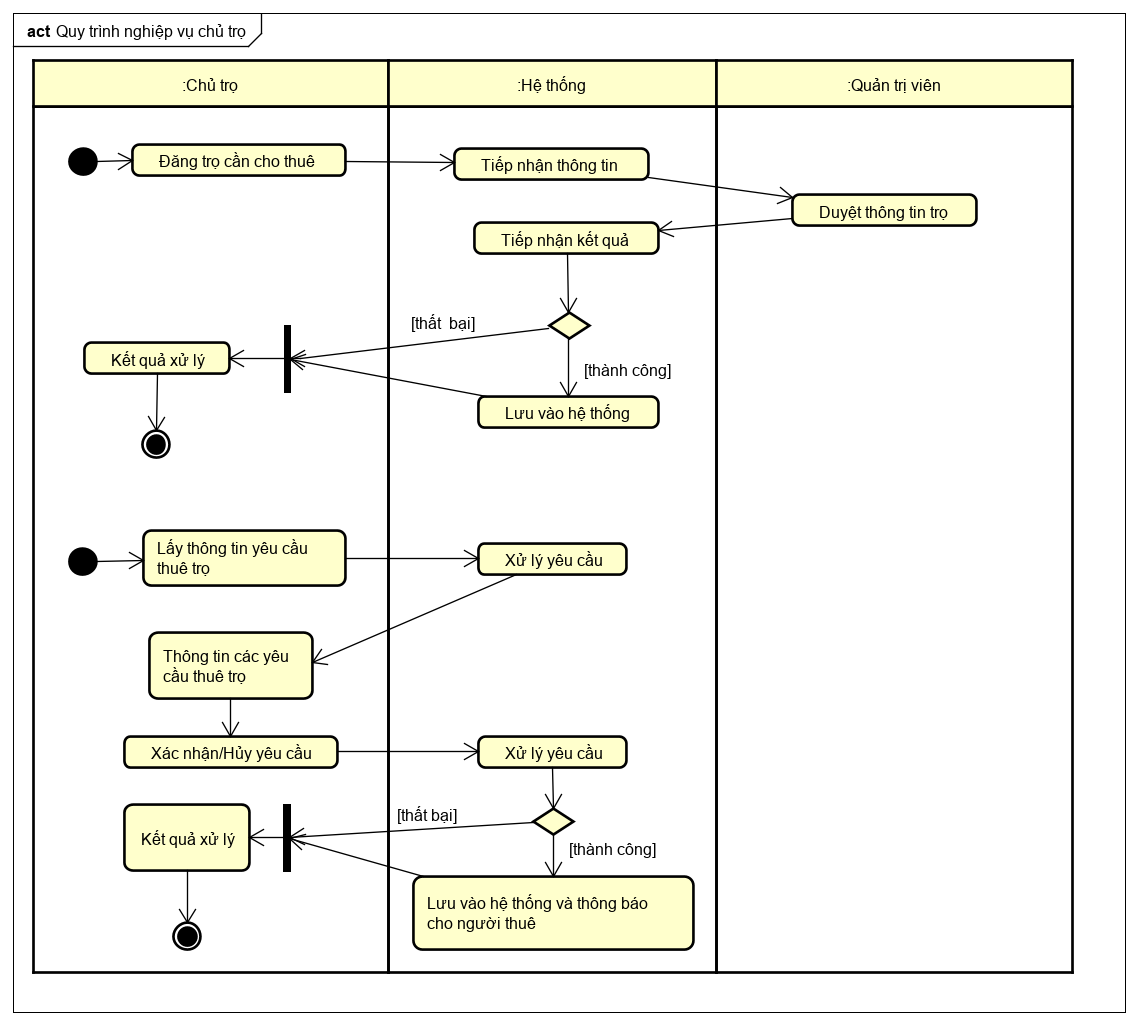


Hình 9. Phân rã use case “Quản lý trọ trong hệ thống”

### Quy trình nghiệp vụ

**

Hình 10. Quy trình nghiệp vụ chính của người thuê

**

Hình 11. Quy trình nghiệp vụ chính của chủ trọ

## Đặc tả chức năng

### Đặc tả use case “Tìm kiếm”

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã Use case** | UC01 | **Tên Use case** | Tìm kiếm |
| **Tác nhân** | Khách/Người thuê/Chủ trọ | | |
| **Sự kiện kích hoạt** | Yêu cầu tìm kiếm | | |
| **Mô tả** | Tìm kiếm trọ theo yêu cầu | | |
| **Tiền điều kiện** | Không | | |
| **Luồng sự kiện chính** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 1 | Hệ thống | Hiển thị trang chủ | | 2 | Tác nhân | Yêu cầu tìm kiếm | | 3 | Hệ thống | Hiển thị giao diện tìm kiếm | | 4 | Tác nhân | Nhập các thông tin cần thiết. Tìm kiếm | | 5 | Hệ thống | Hiển thị kết quả cho người dùng | | | |
| **Luồng sự kiện thay thế** | Không | | |
| **Hậu điều kiện** | Hiển thị kết quả tìm kiếm trọ | | |

Bảng 1. Đặc tả use case “Tìm kiếm”

### Đặc tả use case “Yêu cầu thuê trọ”

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã Use case** | UC02 | **Tên Use case** | Yêu cầu thuê trọ |
| **Tác nhân** | Người thuê | | |
| **Sự kiện kích hoạt** | Yêu cầu thuê | | |
| **Mô tả** | Yêu cầu thuê trọ mà người dùng muốn thuê | | |
| **Tiền điều kiện** | Tìm kiếm trọ | | |
| **Luồng sự kiện chính** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 1 | Hệ thống | Hiển thị giao diện kết quả tìm kiếm | | 2 | Người thuê | Chọn xem thông tin trọ cần biết thêm chi tiết | | 3 | Hệ thống | Hiển thị thông tin trọ mà khách chọn | | 4 | Người thuê | Yêu cầu thuê | | 5 | Hệ thống | Gửi thông báo xác nhận yêu cầu trên | | 6 | Người thuê | Xác nhận yêu cầu | | 7 | Hệ thống | Hiển thị kết quả cho người dùng | | | |
| **Luồng sự kiện thay thế** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 6a | Người thuê | Hủy yêu cầu trên | | 7a | Hệ thống | Trả về giao diện trước khi yêu cầu | | 7b | Hệ thống | Thông báo lỗi nếu có lỗi hệ thống | | | |
| **Hậu điều kiện** | Hiển thị trọ mới gửi yêu cầu | | |

Bảng 2. Đặc tả use case “Yêu cầu thuê trọ”

### Đặc tả use case “Bình luận – đanh giá trọ”

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã Use case** | UC03 | **Tên Use case** | Bình luận - đánh giá trọ |
| **Tác nhân** | Người thuê/Chủ trọ/Quản trị viên | | |
| **Sự kiện kích hoạt** | Yêu cầu bình luận - đánh giá trọ | | |
| **Mô tả** | Bình luận - đánh giá trọ mà bạn muốn | | |
| **Tiền điều kiện** | Xem thông tin trọ (đã đăng nhập) | | |
| **Luồng sự kiện chính** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 1 | Hệ thống | Hiển thị giao diện thông tin chi tiết của trọ | | 2 | Tác nhân | Nhập thông tin bình luận và đánh giá. Gửi yêu cầu | | 3 | Hệ thống | Kiểm tra định dạng các trường thông tin | | 4 | Hệ thống | Hiển thị kết quả cho người dùng và cập nhật thông tin chỉnh sửa | | | |
| **Luồng sự kiện thay thế** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 3a | Tác nhân | Thông báo lỗi: Cần nhập các trường đúng định dạng | | 4a | Hệ thống | Thông báo lỗi nếu có lỗi hệ thống | | | |
| **Hậu điều kiện** | Bình luận – đánh giá trọ | | |

Bảng 3. Đặc tả use case “Bình luận - đánh giá trọ”

### Đặc tả use case “Thêm mới trọ cho thuê”

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã Use case** | UC04 | **Tên Use case** | Thêm mới trọ |
| **Tác nhân** | Chủ trọ | | |
| **Sự kiện kích hoạt** | Yêu cầu thêm mới trọ | | |
| **Mô tả** | Giúp chủ trọ cho thuê trọ hiện tại không dùng | | |
| **Tiền điều kiện** | Đã đăng ký với tư cách chủ trọ. Đã ở trong giao diện quản lý trọ cho thuê | | |
| **Luồng sự kiện chính** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 1 | Hệ thống | Hiển thị giao diện quản lý trọ cho thuê | | 2 | Chủ trọ | Yêu cầu thêm mới trọ | | 3 | Hệ thống | Hiển thị giao diện thêm mới trọ | | 4 | Chủ trọ | Nhập các thông tin cần thiết của trọ. Gửi yêu cầu | | 5 | Hệ thống | Kiểm tra định dạng tài khoản | | 6 | Hệ thống | Hiển thị kết quả cho chủ trọ và lưu vào cơ sở dữ liệu | | | |
| **Luồng sự kiện thay thế** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 4a | Chủ trọ | Hủy thêm mới trọ | | 5a | Hệ thống | Trả về giao diện quản lý trọ | | 5b | Hệ thống | Thông báo lỗi: Cần nhập các trường đúng định dạng | | 6a | Hệ thống | Thông báo lỗi nếu hệ thống gặp sự cố | | | |
| **Hậu điều kiện** | Hệ thống hiển thị trọ mới thêm | | |

Bảng 4. Đặc tả use case “Thêm mới trọ cho thuê”

### Đặc tả use case “Hủy thuê trọ”

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã Use case** | UC05 | **Tên Use case** | Hủy thuê trọ |
| **Tác nhân** | Người thuê/Chủ trọ/Quản trị viên | | |
| **Sự kiện kích hoạt** | Hủy thuê trọ | | |
| **Mô tả** | Hủy thuê trọ hiện tại của người dùng | | |
| **Tiền điều kiện** | Đã thuê trọ | | |
| **Luồng sự kiện chính** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 1 | Hệ thống | Hiển thị giao diện thuê trọ | | 2 | Tác nhân | Chọn hủy thuê trọ hiện tại | | 3 | Hệ thống | Gửi thông báo xác nhận yêu cầu trên | | 4 | Tác nhân | Xác nhận yêu cầu | | 5 | Hệ thống | Hiển thị kết quả cho người dùng | | | |
| **Luồng sự kiện thay thế** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 4a | Tác nhân | Hủy yêu cầu trên | | 5a | Hệ thống | Trả về giao diện trước khi yêu cầu | | 5b | Hệ thống | Thông báo lỗi nếu có lỗi hệ thống | | | |
| **Hậu điều kiện** | Hủy thuê trọ | | |

Bảng 5. Đặc tả use case “Hủy thuê trọ”

### Đặc tả use case “Liên hệ quản trị viên”

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã Use case** | UC06 | **Tên Use case** | Liên hệ với quản trị viên |
| **Tác nhân** | Khách/Người thuê/Chủ trọ | | |
| **Sự kiện kích hoạt** | Yêu cầu liên hệ với quản trị viên | | |
| **Mô tả** | Phản hồi cho quản trị viên khi cần thiết | | |
| **Tiền điều kiện** | Không | | |
| **Luồng sự kiện chính** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 1 | Hệ thống | Hiển thị trang chủ | | 2 | Tác nhân | Yêu cầu liên hệ | | 3 | Hệ thống | Hiển thị giao diện liên hệ | | 4 | Tác nhân | Nhập các thông tin cần thiết. Gửi yêu cầu | | 5 | Hệ thống | Kiểm tra định dạng các trường thông tin | | 6 | Hệ thống | Hiển thị kết quả cho người dùng và gửi mail cho quản trị viên | | | |
| **Luồng sự kiện thay thế** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 5a | Hệ thống | Thông báo lỗi: Nhập các trường đúng định dạng | | 6a | Hệ thống | Thông báo lỗi nếu có lỗi hệ thống | | | |
| **Hậu điều kiện** | Phản hồi cho quản trị viên khi cần thiết | | |

Bảng 6. Đặc tả use case “Liên hệ với quản trị viên”

## Yêu cầu phi chức năng

* Chức năng (Functionality): Không
* Tính dễ dùng (Usability):

Giao diện(web) người dùng tương thích với các loại màn hình, thân thiện với người sử dụng

* Tính ổn định (Reliability):

Hệ thống phải hoạt động liên tục 24giờ/ngày, 7 ngày/tuần, với thời gian dừng hoạt động không quá 10 giờ/tuần

* Hiệu suất (Performance):

Hệ thống phải hỗ trợ từ 30 -50 người dùng truy xuất CSDL trung tâm bất kỳ lúc nào.

Hệ thống phải hoàn thành yêu cầu được nhận với thời gian dưới 2 phút

* Sự hỗ trợ (Supportability): Thông qua liên lạc với quản trị viên
* Các rằng buộc thiết kết (Design Constraint):

Hệ thống tích hợp với Cơ sở dữ liệu MySQL

# Công nghệ sử dụng

## Spring Boot

3.1.1. Tổng quan về Spring Boot

[Spring Boot](https://topdev.vn/viec-lam-it/spring-boot-kt4317) là một dự án phát triển bởi JAVA(ngôn ngữ java) trong hệ sinh thái Spring framework. Nó giúp cho các lập trình viên chúng ta đơn giản hóa quá trình lập trình một ứng dụng với Spring, chỉ tập trung vào việc phát triển business cho ứng dụng.

Để phát triển một ứng dụng web cơ bản HelloWorld sử dụng Spring framework bạn sẽ cần ít nhất 5 công đoạn sau:

* Tạo một project sử dụng Maven với các dependency cần thiết của Spring MVC và Servlet API.
* Một tập tin web.xml để khai báo DispatcherServlet của Spring MVC.
* Một tập tin cấu hình của Spring MVC.
* Một class Controller trả về một trang “Hello World” khi có request đến.
* Cuối cùng là phải có một web server dùng để triển khai ứng dụng lên chạy.

Trong các công đoạn này, chỉ có công đoạn tạo một class Controller thì có thể khác cho các ứng dụng khác nhau vì mỗi ứng dụng có một yêu cầu khác nhau. Còn các công đoạn khác thì như nhau.

Giờ đây với Spring Boot, chúng ta có thể tạo dự án Spring một cách nhanh chóng và cấu hình cũng đơn giản dùng Visual Studio Code hay các ứng dụng tích hợp như Intellij Ultimate, Eclipse, NetBeans,…  
Dưới đây là một số tính năng nổi bật cũng như ưu điểm của của Spring Boot:

* Tạo các ứng dụng Spring độc lập
* Nhúng trực tiếp Tomcat, Jetty hoặc Undertow (không cần phải deploy ra file WAR)
* Các starter dependency giúp việc cấu hình Maven đơn giản hơn
* Tự động cấu hình Spring khi cần thiết
* Không sinh code cấu hình và không yêu cầu phải cấu hình bằng XML …

3.1.2. Một số nội dung cơ bản dùng trong đồ án này

1. *Spring Boot JPA*

Spring Boot JPA là một bản ghi chi tiết của Java để quản lý dữ liệu quan hệ trong các ứng dụng Java. Nó cho phép chúng ta truy cập và lưu trữ dữ liệu giữa các object/class Java và database quan hệ. JPA tuân theo Object-Relation Mapping (ORM). Nó là một tập hợp các interface. Nó cũng cung cấp một API EntityManager runtime để xử lý các câu query và giao dịch trên các object dựa trên database. Nó sử dụng ngôn ngữ truy vấn hướng đối tượng độc lập nền tảng JPQL (Java Persistent Query Language).

JPA bao gồm ba lĩnh vực:

* The Java Persistence API
* Object-Relational metadata
* Bản thân API, được định nghĩa trong persistence package.

JPA không phải là một framework. Nó định nghĩa một khái niệm có thể được thực thi bởi bất kỳ framework nào.

1. *Template engine*

Thymeleaf là một Java template engine dùng để xử lý và tạo HTML, XML, Javascript, CSS và text.

Mục tiêu chính của thymeleaf là mang lại các template tự nhiên, đồng nhất, đơn giản (nature templates) cho công việc phát triển.

Các lợi ích của Thymeleaf Với thymeleaf, ta chỉ cần sử dụng file HTML là có thể hiển thị tất cả mọi thứ (không cần jsp ...). Thymealeaf sẽ tham gia vào renderd các file HTML dưới dạng các thuộc tính trong các thẻ HTML. Do đó ta không cần phải thêm bất kỳ thẻ non-HTML nào. Vì là HTML nên ta có thể xem các file view mà không cần khởi chạy server. Thymeleaf hỗ trợ cơ chế cache, do đó ta có thể cache dữ liệu hoặc custom để hiển thị view khi có thay đổi mà không cần restart server.

1. *Loombok*

**Lombok**là một thư viện Java giúp tự sinh ra các hàm setter/getter, hàm khởi tạo, toString… và tinh gọn chúng.

Thực tế thì hầu hết các IDE hiện này đều hỗ trợ sinh code tự động ví dụ như trên IntelliJ IDE, ta click chuột phải vào file cần sinh code và chọn Source và chọn các method cần sinh ra.

1. *Spring security*

**Spring Security** là một trong những core feature quan trọng của Spring Framework, nó giúp chúng ta phân quyền và xác thực người dùng trước khi cho phép họ truy cập vào các tài nguyên của chúng ta.

## Mysql Databse

3.2.1. Giới thiệu chung

SQL (Structed Query Language) Ngôn ngữ truy vấn có cấu trúc: là ngôn ngữ theo chuẩn ANSI để định nghĩa và xử lý dữ liệu trong một cơ sở dữ liệu quan hệ. Các phát biểu SQL dùng để truy tìm và cập nhật dữ liệu trong một cơ sở dữ liệu.

MySQL là một trong những hệ thống quản lý cơ sở dữ liệu (CSDL) mã nguồn mở quan hệ SQL sử dụng trên web phổ biến nhất hiện nay.

*Cách thức hoạt động của MySQL*

MySQL dựa trên mô hình client-server. Cốt lõi của MySQL là máy chủ MySQL, xử lý tất cả các hướng dẫn cơ sở dữ liệu (hoặc các lệnh). Máy chủ MySQL có sẵn như là một chương trình riêng biệt để sử dụng trong môi trường mạng client-server và như một thư viện có thể được nhúng (hoặc liên kết) vào các ứng dụng riêng biệt.

MySQL hoạt động cùng với một số chương trình tiện ích hỗ trợ quản trị cơ sở dữ liệu MySQL. Các lệnh được gửi đến MySQLServer thông qua máy khách MySQL, được cài đặt trên máy tính.

MySQL ban đầu được phát triển để xử lý cơ sở dữ liệu lớn một cách nhanh chóng. Mặc dù MySQL thường chỉ được cài đặt trên một máy, nhưng nó có thể gửi cơ sở dữ liệu đến nhiều vị trí, vì người dùng có thể truy cập thông qua các giao diện máy khách MySQL khác nhau. Các giao diện này gửi các câu lệnh SQL đến máy chủ và sau đó hiển thị kết quả.

*Các tính năng cốt lõi của MySQL*

MySQL cho phép dữ liệu được lưu trữ và truy cập trên nhiều công cụ lưu trữ, bao gồm InnoDB, CSV và NDB. MySQL cũng có khả năng sao chép dữ liệu và phân vùng bảng để có hiệu suất và độ bền tốt hơn. Người dùng MySQL không bắt buộc phải học các lệnh mới; họ có thể truy cập dữ liệu của mình bằng các lệnh [SQL](https://viettuts.vn/sql) tiêu chuẩn.

MySQL được viết bằng C và C++ và có thể truy cập và có sẵn trên hơn 20 nền tảng, bao gồm Mac, Windows, Linux và Unix. RDBMS hỗ trợ cơ sở dữ liệu lớn với hàng triệu bản ghi và hỗ trợ nhiều loại dữ liệu.

Để bảo mật, MySQL sử dụng một đặc quyền truy cập và hệ thống mật khẩu được mã hóa cho phép xác minh dựa trên máy chủ. Các máy khách MySQL có thể kết nối với Máy chủ MySQL bằng một số giao thức, bao gồm cả giao thức TCP/IP trên bất kỳ nền tảng nào. MySQL cũng hỗ trợ một số chương trình máy khách và tiện ích, chương trình dòng lệnh và công cụ quản trị như MySQL Workbench.

*Các nhánh của MySQL bao gồm:*

* Drijection: một hệ thống quản lý cơ sở dữ liệu nguồn mở nhẹ được phát triển dựa trên MySQL 6.0.
* MariaDB: một sự thay thế phổ biến do cộng đồng phát triển cho MySQL sử dụng các lệnh và API của MySQL.
* Percona Server với XtraDB: một phiên bản nâng cao của MySQL được biết đến với khả năng mở rộng theo chiều ngang.

3.2.2. Một vài ưu điểm của MysqlDB

*Linh hoạt và dễ dùng*

Bạn có thể sửa source code để đáp ứng nhu cầu của bạn mà không phải thanh toán têm bất kỳ chi phí nào. Quá trình cài đặt cũng rất đơn giản và thường không quá 30 phút.

*Hiệu năng cao*

Nhiều server clusters sử dụng MySQL. Bất kể bạn lưu trữ dữ liệu lớn của các trang thương mại điện tử hoặc những hoạt động kinh doanh nặng nề liên quan đến công nghệ thông tin, MySQL cũng có thể đáp ứng được với tốc độ cao, mượt mà.

*Tiêu chuẩn trong ngành*

Ngành công nghệ và dữ liệu đã sử dụng MySQL nhiều năm, vì vậy nó là một kỹ năng căn bản một chuyên gia lập trình. Người dùng MySQL cũng có thể triển khai dự án nhanh và thuê các chuyên gia dữ liệu với mức phí nếu họ cần.

*An toàn*

An toàn dữ liệu luôn là vấn đề quan trọng nhất khi chọn phần mềm RDBMS. Với hệ thống phân quyền truy cập và quản lý tài khoản, MySQL đặt tiêu chuẩn bảo mật rất cao. Mã hóa thông tin đăng nhập và chứng thực từ host đều khả dụng.

Tổng kết lại khi sử dụng Spring Boot chúng ta không mất qua nhiều công đoạn và thời gian để tạo một dự án. Mọi thứ đã được Spring Boot tự động kết hợp với hệ CSDL Mysql, làm đơn giản trong việc thao tác với CSDL và khởi tạo các model cần thiết.

# Phát triển và triển khai ứng dụng

## Thiết kế kiến trúc

### Lựa chọn kiến trúc phần mềm

Thông qua việc phân tích, xây dựng các use case ở trên, em quyết định thiết kế kiến trúc của ứng dụng dựa trên mô hình MVC (Model-View-Controller)

Giới thiệu cơ bản mô hình MVC:

* Model chứa tất cả các nội dung cụ thể của ứng dụng và xử lý logic, bao gồm cả:
* Mọi đối tượng nội dung
* Truy cập dữ liệu bên ngoài
* Mọi chức năng xử lý trong ứng dụng
* View chứa tất cả các giao diện chức năng cụ thể và cho phép:
* Trình bày mọi nội dung và chức năng logic
* Truy cập dữ liệu bên ngoai
* Các chức năng được người dùng yêu cầu
* Controller cung cấp truy cập vào Model và View, điều khiển dòng dữ liệu

Đồng thời em xây dựng các controller tập trung vào các thao tác với mỗi model. Mỗi model sẽ có 1 bộ gồm (controller – service – repository)

Repository do Spring Boot cung cấp nhằm tạo sự thuận lợi khi thao tác với cơ sở dữ liệu, giảm các đoạn mã tương tự nhau nhưng vẫn phải viết khi triển khai như thêm, sửa, xóa,… giảm được sự ảnh hưởng từ việc phải thực thi quá nhiều các lớp trong tầng truy cập dữ liệu.

Ngoài sử dụng mô hình MVC trong thiết kế kiến trúc của ứng dụng, em có tạo thêm 1 package service nhằm bổ trợ phần controller, có tác dụng triển khai các nghiệp vụ cụ thể đối với mỗi thao tác để controller tập trung vào việc điều khiển luồng hoạt động của ứng dụng.

### Thiết kế tổng quan



Hình 12. Sơ đồ kiến trúc tổng thể hệ thống

Điển hình là kiến trúc 5 tầng (từ trên xuống):

* Tầng trình bày: Chuyển các dữ liệu cho người dùng và biến đổi các hành động của người dùng thành các sự kiện vào của hệ thống
* Tầng ứng dụng: Bao gồm các đối tượng điều khiển và dẫn dắt các quy luật của ứng dụng
* Tầng nghiệp vụ: Bao gồm các đối tượng nghiệp vụ (hay lĩnh vực), cùng sự cài đặt các quy tắc quản lý chúng
* Tầng truy cập dữ liệu: Quản lý việc truy cập (đọc/viết) các đối tượng nghiệp vụ từ các phương tiện lưu trữ dữ liệu
* Tầng lưu trữ dữ liệu: Bảo đảm sự lưu giữ lâu dài các dữ liệu

Chi tiết các tầng, các package có trong hệ thống sẽ được đề cập ở phần tiếp theo

### Thiết kế chi tiết gói

1. *Sơ đồ lớp tổng quan tầng trình bày*

*Ảnh có chứa văn bản

Mô tả được tạo tự động*

Hình 13. Sơ đồ tổng quan tầng trình bày

Tầng trình bày bao gồm các giao diện dành cho khách thuê, chủ trọ và quản lý

Giao diễn dành cho mỗi loại tác nhân đều có thể chuyển đổi qua lại một cách dễ dàng, không bị hạn chế do xây dựng ứng dụng web. Các thao tác trên giao diện đều được xử lý thông qua javascript, gọi các API tương ứng để xử lý nghiệp vụ, kiểm tra các thông tin nhập từ bàn phím.

1. *Sơ đồ lớp tổng quan tầng ứng dụng*

*Ảnh có chứa văn bản

Mô tả được tạo tự động*

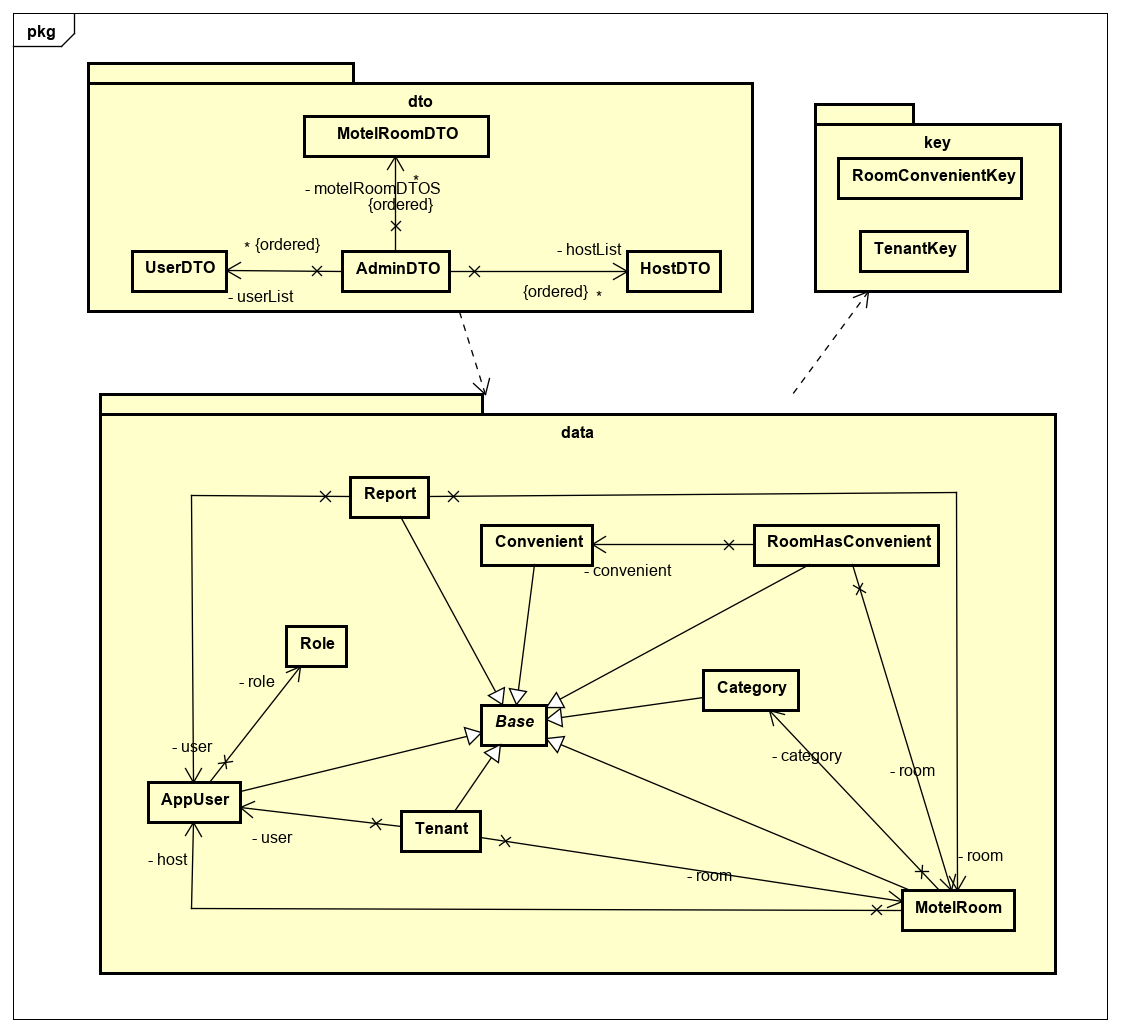
Hình 14. Sơ đồ tổng quan tầng ứng dụng

Tầng ứng dụng bao gồm các config cơ bản của ứng dụng và các controller điều khiển các luồng hoạt động mà người dùng thao tác.

Trong package config chứa những cài đặt cơ bản của hệ thống về nguồn của những file tĩnh trong ứng dụng, cài đặt về bảo mật, kiểm soát quyền truy cập các API của người dùng nhằm tránh việc truy cập, thay đổi dữ liệu bừa bãi.

Mỗi controller đều xây dựng từ các model, các tác nhân trong hệ thống và độc lập trong việc xử lý các nghiệp vụ tương ứng, tránh việc chồng chéo các nghiệp vụ và hướng đến thiết kế kiến trúc theo mô hình Microservices

1. *Sơ đồ lớp tổng quan tầng nghiệp vụ*

**

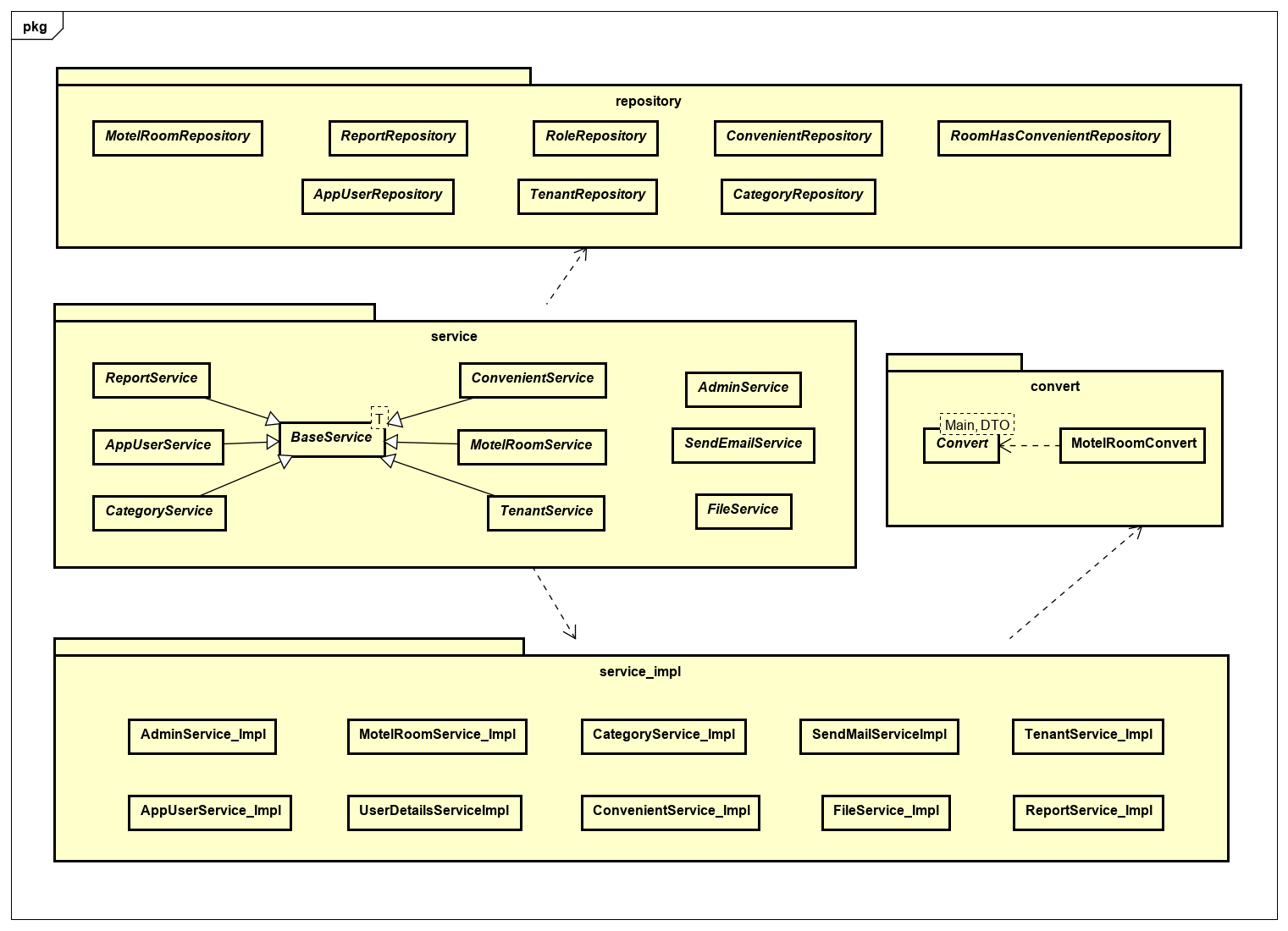
Hình 15. Sơ đồ tổng quan tầng nghiệp vụ

Trong tầng nghiệp vụ bao gồm các đối tượng nghiệp vụ cần quản lý

Mỗi đối tượng đều được ánh xạ từ các bảng trong cơ sở dữ liệu nhằm tạo thuận lợi cho việc quản lý thao tác trên các đối tượng.

Package DTO được tạo ra nhằm mục đíchgiảm số lần gọi các method giữa các tiến trình xử lý, tạo ra một đối tượng hoàn chỉnh có đầy đủ thông tin cần thiết khi sử dụng trong ứng dụng. Còn một vài hạn chế do cách sử dụng với đồ án hiện tại chưa thực sử được tối ưu nên mỗi dto sẽ hơi nặng về các thuộc tính dữ liệu trong đó.

1. *Sơ đồ lớp tổng quan tầng truy xuất và tầng lưu trữ dữ liệu*

**

Hình 16. Sơ đồ tổng quan tầng truy xuất và lưu trữ dữ liệu

Tầng truy xuất và lưu trữ diệu có tác dụng chính là đọc và ghi các dữ liệu cần thiết sau khi hoàn thành các thao tác nghiệp vụ do controller kiểm soát

Package repository là do Spring Boot hỗ trợ nhằm tạo sự thuận tiện khi thao tác với dữ liệu

Package service có tác dụng bổ trợ cho controller để thao tác các nghiệp vụ chính, thông qua việc sử dụng các repository.

## Thiết kế chi tiết

### Thiết kế giao diện và minh họa

Thiết kế với giao diện, bố cục website thể hiện đẹp, mang tính mỹ thuật với độ hiển thị nội dung co giãn phù hợp trên tất cả các màn hình thiết bị như desktop, laptop, tablet, smartphone, với mọi độ phân giải màn hình. Màu sắc, kiến trúc phù hợp không rối mắt.

Danh sách các màn hình trong hệ thống:

*Với người thuê:*

1. Trang chủ
2. Đăng nhập / Đăng ký tài khoản
3. Tìm kiếm trọ
4. Thông tin chi tiết của trọ
5. Trang thông tin cá nhân và trọ thuê
6. Thông tin mới và Liên hệ

*Với chủ trọ:*

1. Trang chủ
2. Quản lý trọ cho thuê
3. Quản lý người thuê trọ
4. Yêu cầu thuê trọ mới
5. Trang thông tin cá nhân

*Với quản lý:*

1. Trang chủ
2. Quản lý trọ cho thuê
3. Quản lý người thuê trọ
4. Trang thông tin cá nhân

**Hình ảnh minh họa:**

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, đen

Mô tả được tạo tự động

Hình 17. Giao diện quản lý yêu cầu thuê của chủ trọ

|  |  |  |  |
| --- | --- | --- | --- |
| Đặc tả màn hình | Control | Hoạt động | Chức năng |
| Giao diện quản lý yêu cầu thuê của chủ trọ | Thanh menu các màn hình của quản lý | Khởi tạo | Chuyển giữa các màn hình |
| Bảng thông tin các yêu cầu | Khởi tạo | Hiển thị những yêu cầu thuê trọ của người dùng |
| Lựa chọn “Sắp xếp” | Khởi tạo | Hiển thị các cách sắp xếp dữ liệu hiển thị |
| Nút tại cột “Người thuê” hoặc “Phòng thuê” | Click | Xem thông tin người yêu cầu hoặc phòng thuê |
| Nút tại cột “Hành động” | Click | Xác nhận yêu cầu thuê hoặc hủy yêu cầu |
| Thanh thông tin về ứng dụng | Khởi tạo | Các thông tin về đội ngũ phát triển, thông tin liên hệ,… |

Bảng 7. Đặc tả giao diện quản lý yêu cầu thuê của chủ trọ

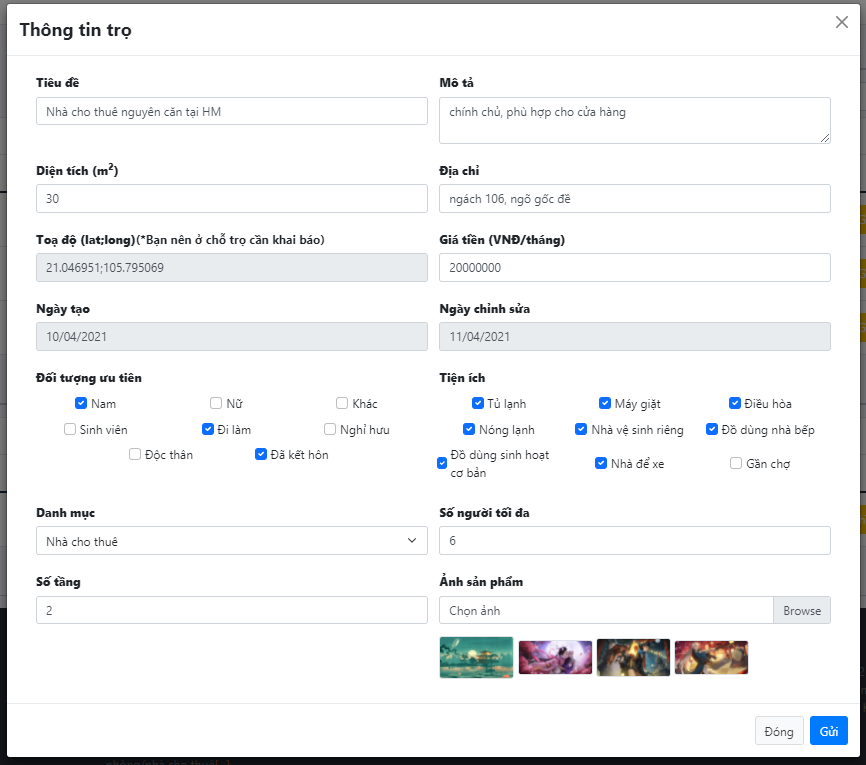
Ảnh có chứa văn bản

Mô tả được tạo tự động

Hình 18. Giao diện quản lý trọ của chủ trọ

|  |  |  |  |
| --- | --- | --- | --- |
| Đặc tả màn hình | Control | Hoạt động | Chức năng |
| Giao diện quản lý trọ của chủ trọ | Thanh menu các màn hình của quản lý | Khởi tạo | Chuyển giữa các màn hình |
| Bảng thông tin các trọ mà chủ trọ đang sở hữu | Khởi tạo | Hiển thị những các trọ mà chủ trọ quản lý (trọ còn trống, đã đầy,…) |
| Lựa chọn “Sắp xếp” | Khởi tạo | Hiển thị các cách sắp xếp dữ liệu hiển thị |
| Nút tại cột “Thông tin” hoặc “Số người thuê” | Click | Xem thông tin người đang thuê (người đang yêu cầu thuê,…) hoặc phòng thuê |
| Nút tại cột “Hành động” | Click | Xóa trọ đang quản lý hoặc xem thông tin trọ dưới góc nhìn người thuê |
| Thanh thông tin về ứng dụng | Khởi tạo | Các thông tin về đội ngũ phát triển, thông tin liên hệ,… |

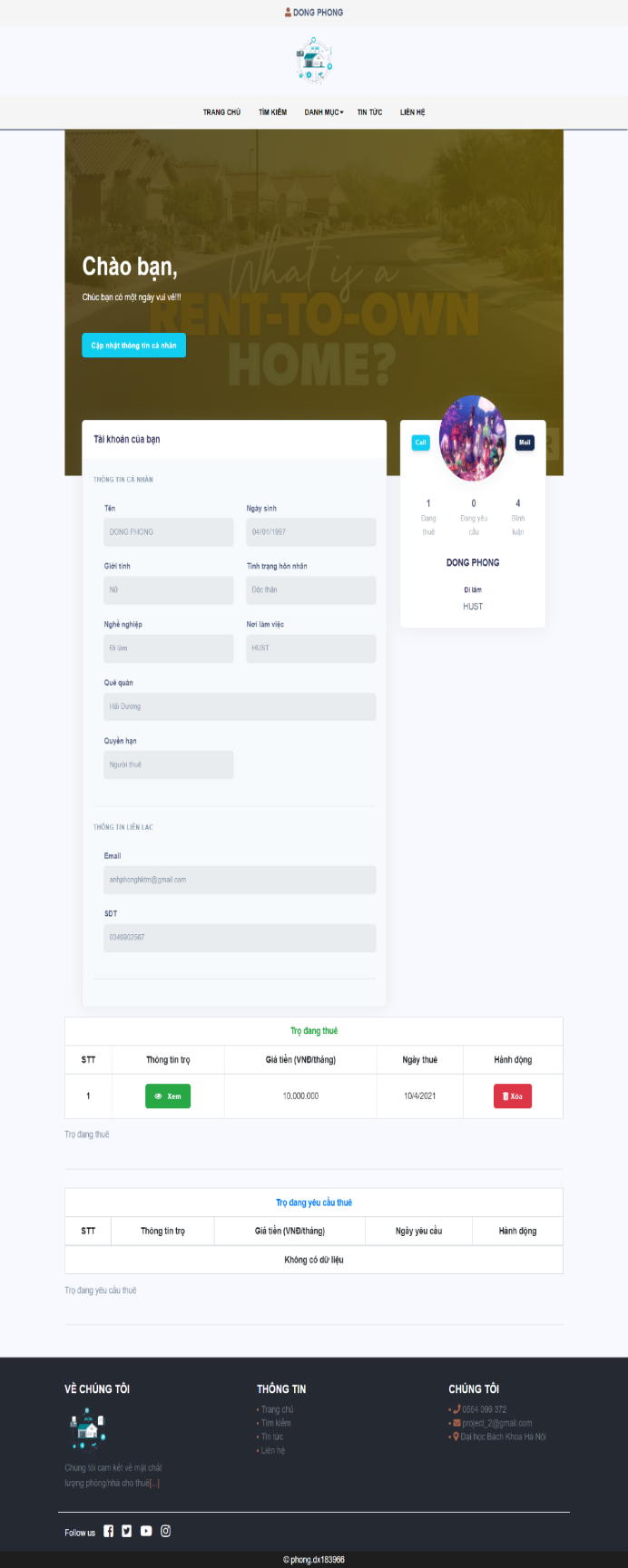
Bảng 8. Đặc tả giao diện quản lý trọ của chủ trọ



Hình 19. Giao diện thêm mới/chỉnh sửa trọ

|  |  |  |  |
| --- | --- | --- | --- |
| Đặc tả màn hình | Control | Hoạt động | Chức năng |
| Giao diện thêm mới/chỉnh sửa trọ | Thông tin trọ cần thiết | Khởi tạo | Hiển thị các thông tin của trọ cần tạo mới hoặc trọ muốn sửa |
| Nút “Upload File” | Khởi tạo | Upload các file ảnh của trọ |
| Nút “Đóng” | Click | Hủy thêm/sửa trọ |
| Nút “Gửi” | Click | Gửi thông tin trọ mới để tạo hoặc sửa thông tin trọ đã có |

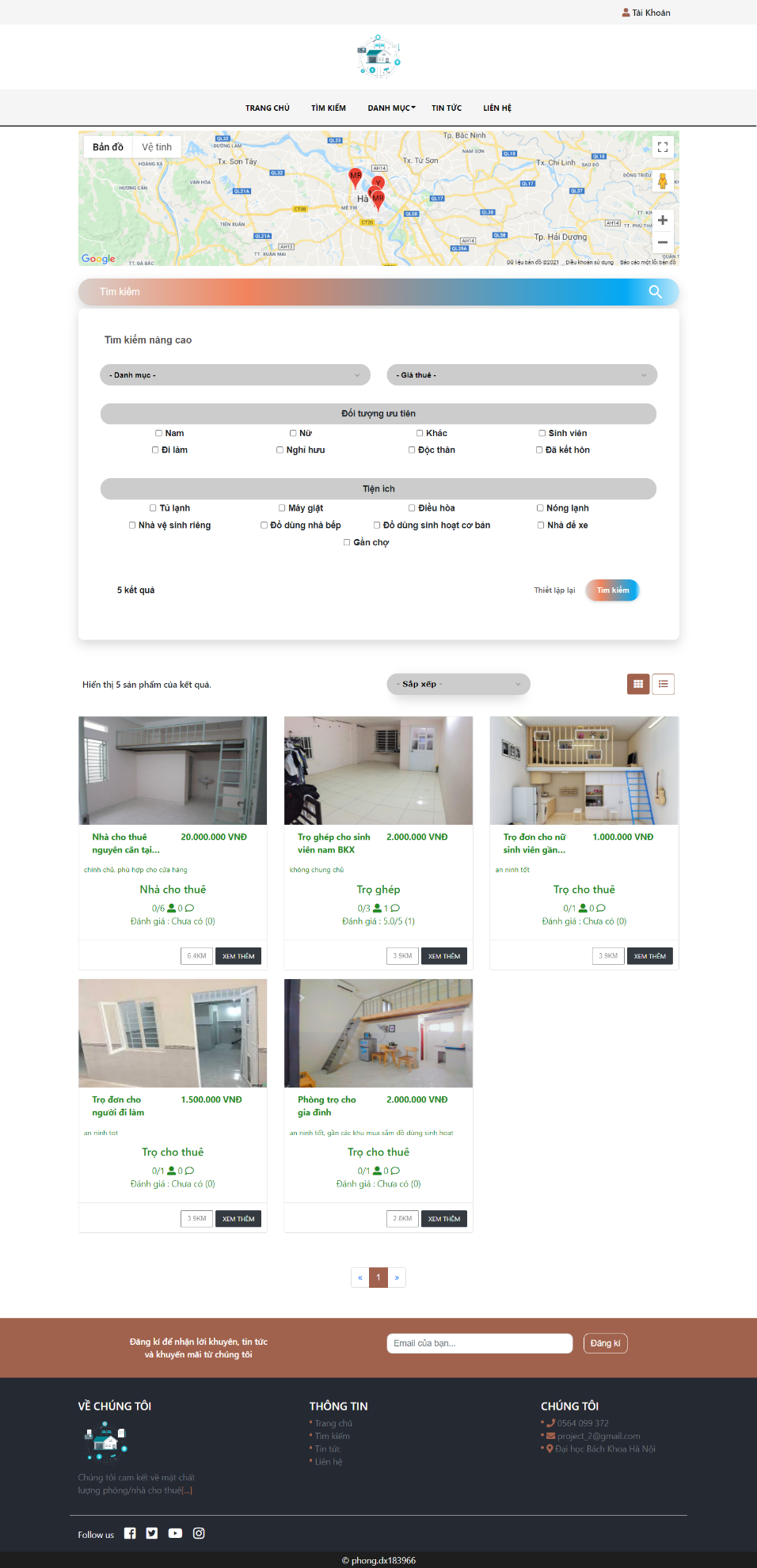
Bảng 9. Đặc tả giao diện thêm mới/chỉnh sửa trọ



Hình 20. Giao diện thông tin người dùng và quản lý trọ đã thuê/yêu cầu thuê

|  |  |  |  |
| --- | --- | --- | --- |
| Đặc tả màn hình | Control | Hoạt động | Chức năng |
| Giao diện thông tin người dùng và quản lý trọ đã thuê/yêu cầu thuê | Thanh menu các màn hình của người dùng | Khởi tạo | Chuyển giữa các màn hình |
| Thông tin cá nhân của người dùng | Khởi tạo | Hiển thị những các thông tin cá nhân |
| Thông tin các trọ mà người dùng thuê hoặc | Khởi tạo | Hiển thị các trọ mà người dùng đã thuê hoặc yêu cầu |
| Nút “Xem” | Click | Xem thông tin của trọ |
| Nút tại cột “Xóa” | Click | Xóa trọ đã thuê hoặc yêu cầu thuê |
| Nút “Cập nhật thông tin cá nhấn” | Click | Cập nhật thông tin cá nhân |
| Thanh thông tin về ứng dụng | Khởi tạo | Các thông tin về đội ngũ phát triển, thông tin liên hệ,… |

Bảng 10. Đặc tả giao diện thông tin người dùng và quản lý trọ đã thuê/yêu cầu thuê



Hình 21. Giao diện tìm kiếm

|  |  |  |  |
| --- | --- | --- | --- |
| Đặc tả màn hình | Control | Hoạt động | Chức năng |
| Giao diện tìm kiếm | Thanh menu các màn hình của người dùng | Khởi tạo | Chuyển giữa các màn hình |
| Bản đồ các trọ quanh bạn | Khởi tạo | Dựa vào vị trí của người thuê để hiển thị các trọ gần đó |
| Thông tin các trường có thể tìm kiếm | Khởi tạo | Hiển thị những các thông tin cần tìm |
| Kết quả tìm kiếm trọ | Sau khi ấn nút “Tìm kiếm” | Hiển thị các trọ mà người dùng tìm được |
| Nút “Tìm kiếm” | Click | Tìm kiếm các trọ theo thông tin yêu cầu |
| Nút xem theo chế độ | Click | Xem kết quả tìm kiếm theo cột hoặc hàng |
| Lựa chọn “Sắp xếp” | Click | Sắp xếp các trọ theo mong muốn |
| Thanh phân trang | Click | Chuyển giữa các trang khi có quá nhiều dữ liệu |
| Thanh thông tin về ứng dụng | Khởi tạo | Các thông tin về đội ngũ phát triển, thông tin liên hệ,… |

Bảng 11. Đặc tả giao diện tìm kiếm

### Thiết kế lớp

Một vài model và controller chính.

1. Class MotelRoomController

Ảnh có chứa văn bản

Mô tả được tạo tự động

Hình 22. Class MotelRoomController

***Attribute*** Không

***Operation/Method***

|  |  |  |  |
| --- | --- | --- | --- |
| *#* | *Tên* | *Kiểu dữ liệu trả về* | *Mô tả (mục đích)* |
| 1 | findById | MotelRoom | Lấy ra thông tin trọ từ cơ sở dữ liệu bằng id |
| 2 | Insert | MotelRoom | Thêm mới trọ |
| 3 | update | MotelRoom | Cập nhật thông tin trọ trong cơ sở dữ liệu |
| 4 | Delete | Boolean | Xóa trọ |
| 5 | findAll | List< MotelRoom > | trả về danh sách tất cả trọ trong cơ sở dữ liệu |
| 6 | Search\_sort | List< MotelRoom > | trả về tất cả các xe thỏa mã tìm kiếm |
| 7 | findAllByUser | List< MotelRoom > | trả về danh sách tất cả trọ được thuê hoặc yêu cầu bởi người dùng |
| 8 | findAllByHost | List< MotelRoom > | trả về danh sách tất cả trọ của chủ sở hữu |
| 9 | findAllByAdmin | List< MotelRoom > | trả về danh sách tất cả trọ có trong hệ thống cho quản lý |

1. Class TenantController

Ảnh có chứa văn bản

Mô tả được tạo tự động

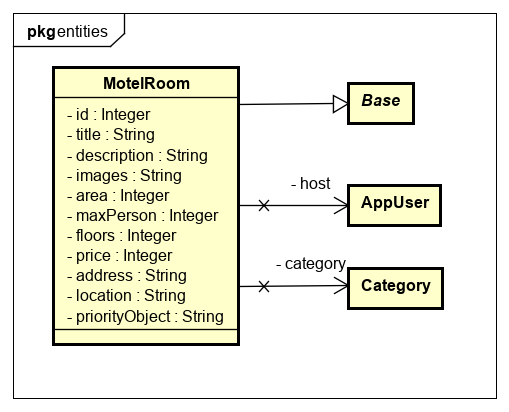
Hình 23. Class TenantController

***Attribute*** Không

***Operation/Method***

|  |  |  |  |
| --- | --- | --- | --- |
| *#* | *Tên* | *Kiểu dữ liệu trả về* | *Mô tả (mục đích)* |
| 1 | Insert | Tenant | Khởi tạo yêu cầu thuê trọ |
| 2 | update | Tenant | Xác nhận yêu cầu hoặc hủy |
| 3 | Delete | Boolean | Hủy thuê trọ |
| 4 | findAll | List< Tenant > | trả về danh sách các trọ có người thuê hoặc yêu cầu |
| 5 | findAllByUser | List< Tenant > | trả về danh sách các trọ của người thuê hoặc yêu cầu |
| 6 | findAllByHost | List< Tenant > | trả về danh sách tất cả trọ của chủ sở hữu có người thuê hoặc yêu cầu |
| 7 | findReqByHost | List< Tenant > | trả về danh sách tất cả trọ của chủ sở hữu có yêu cầu |

1. Class MotelRoom



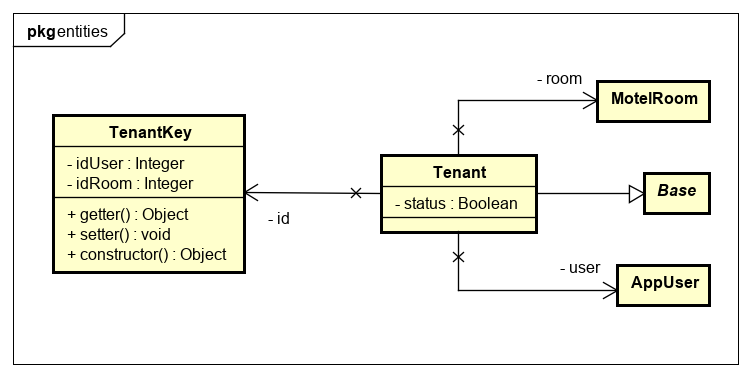
Hình 24. Class MotelRoom

***Attribute***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *#* | *Tên* | *Kiểu dữ liệu trả về* | *Giá trị mặc định* | *Mô tả* |
| 1 | id | Integer | NULL | ID của trọ |
| 2 | Title | String | NULL | Tiêu đề trọ |
| 3 | Description | String | NULL | Mô tả sơ bộ |
| 4 | Images | String | NULL | Đường dẫn ảnh |
| 5 | Area | Integer | 0 | Diện tích |
| 6 | maxPerson | Integer | 1 | Số người tối đa |
| 7 | Floors | Integer | 1 | Số tầng |
| 8 | Price | Integer | NULL | Giá tiền |
| 9 | Address | String | NULL | Địa chỉ |
| 10 | Location | String | NULL | Vị trí tọa độ |
| 11 | priorityObject | String | NULL | Đối tượng ưu tiên |
| 12 | Id\_Host | Integer | NULL | Id chủ trọ |
| 13 | Id\_category | Integer | NULL | Id loại trọ |
| 14 | HistoryPrice | List | NULL | Lịch sửa thay đổi giá |

***Operation/Method***  Các hàm getter/setter cơ bản

1. Class Tenant & TenantKey



Hình 25. Class Tenant & TenantKey

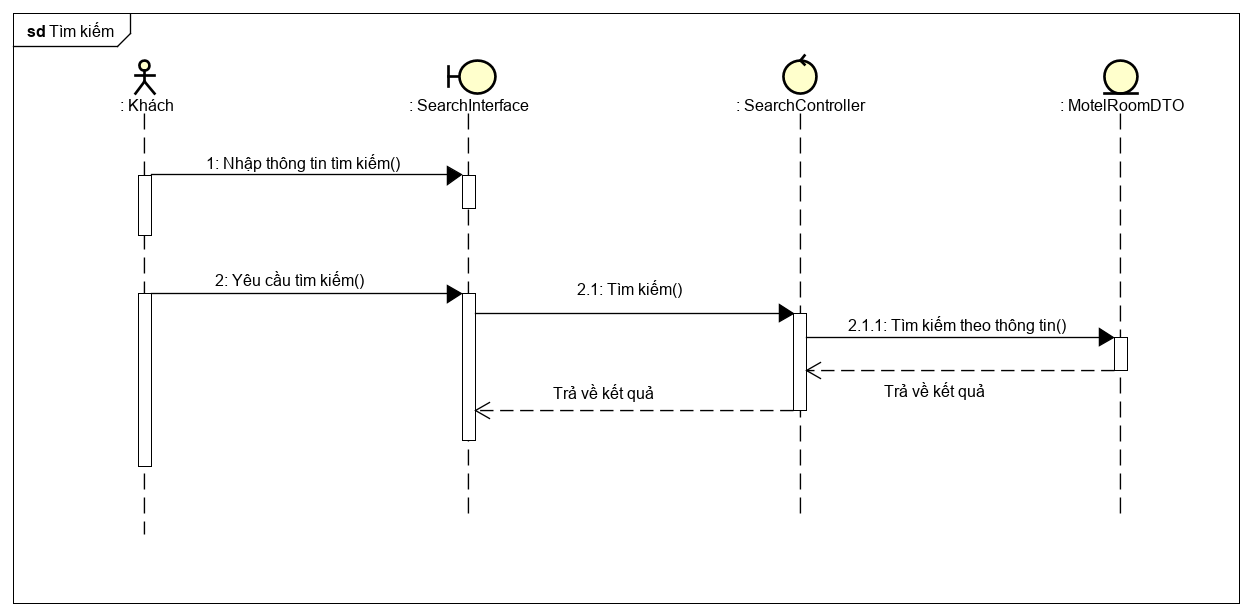
***Attribute***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *#* | *Tên* | *Kiểu dữ liệu trả về* | *Giá trị mặc định* | *Mô tả* |
| 1 | idUser | Integer | NULL | ID của người thuê |
| 2 | idRoom | Integer | NULL | ID của trọ |
| 3 | Status | Boolean | NULL | Trạng thái thuê |

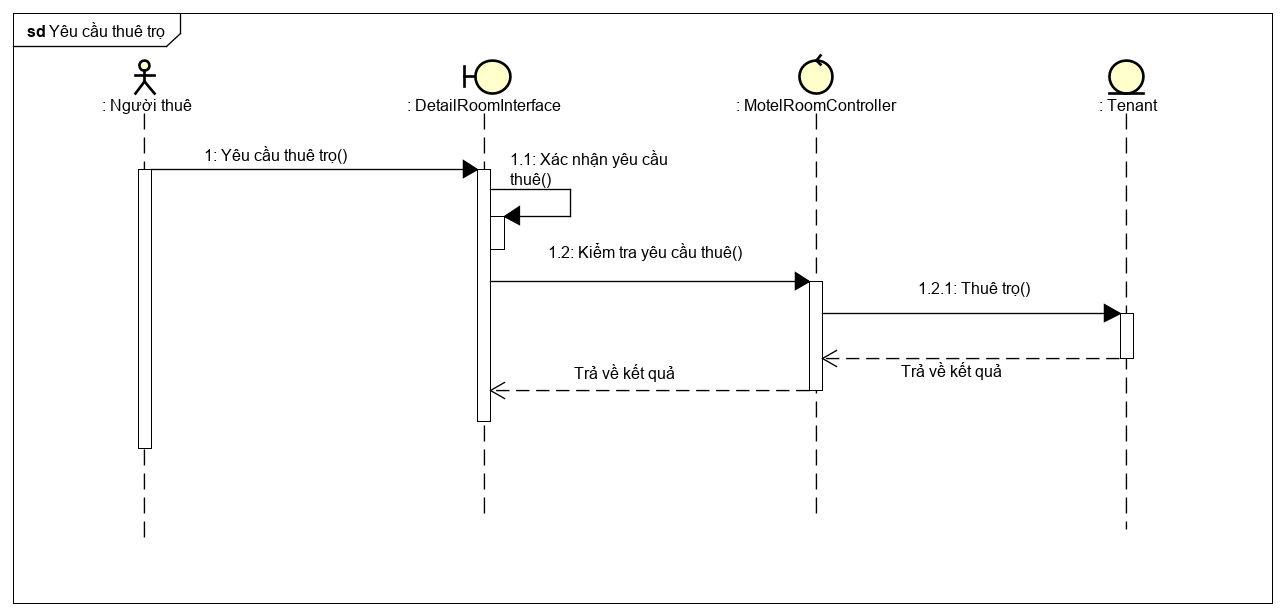
***Operation/Method***  Các hàm getter/setter cơ bản

**Lưu ý: Class Base** chứa các thuộc tinh cơ bản như modifyDate(Ngày chỉnh sửa), createDate(Ngày khởi tạo), modifyBy(Chỉnh sửa bởi ai), createBy(Tạo bởi ai), delete(Tình trạng bản ghi đã bị xóa hay chưa).

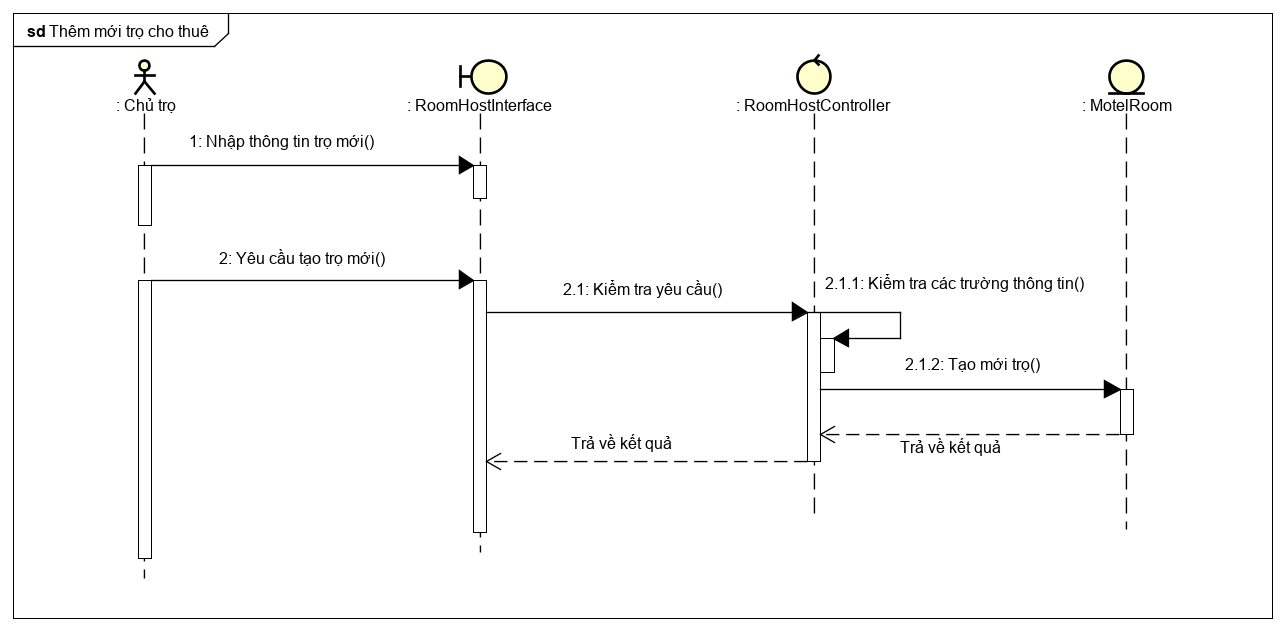
Một số biểu đồ trình tự minh họa cho thiết kế lớp:

**

Hình 26. Biểu đồ trình tự ca sử dụng “Tìm kiếm”

**

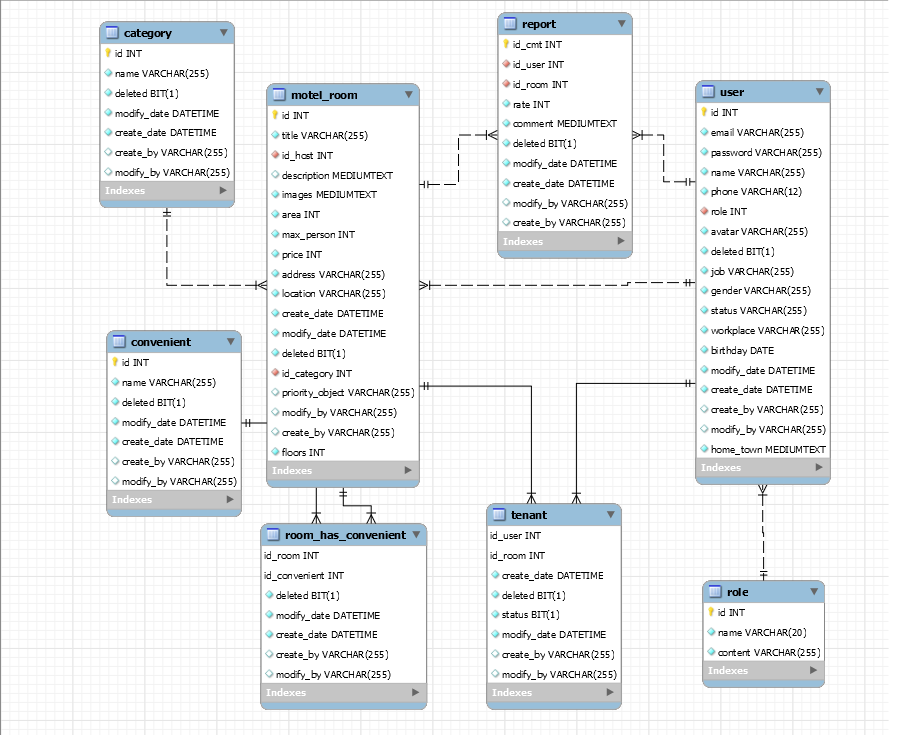
Hình 27. Biểu đồ trình tự ca sử dụng “Yêu cầu thuê trọ”

**

Hình 28. Biểu đồ trình tự ca sử dụng “Thêm mới trọ cho thuê”

### Thiết kế cơ sở dữ liệu

***Sơ đồ quan hệ giữa các bảng***



Hình 29. Sơ đồ quan hệ các bảng dữ liệu

Bảng người dùng (user)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ràng buộc | NULL |
| 1 | id | Mã người dùng | INT | Khóa chính | Không |
| 2 | email | Mail cá nhân | VARCHAR |  | Không |
| 3 | password | Mật khẩu tài khoản | VARCHAR |  | Không |
| 4 | name | Họ và tên | VARCHAR |  | Không |
| 5 | phone | Số điện thoại | DATETIME |  | Không |
| 6 | role | Vai trò người dùng | INT | Khóa ngoài, tham chiếu đến bảng phân quyền | Không |
| 7 | avatar | Ảnh đại diện (đường dẫn ảnh) | VARCHAR |  | Không |
| 8 | deleted | Bị xóa hay chưa | BIT |  | Không |
| 9 | job | Nghề nghiệp | VARCHAR |  | Không |
| 10 | gender | Giới tính | VARCHAR |  | Không |
| 11 | status | Tình trạng hôn nhân | VARCHAR |  | Không |
| 12 | workplace | Nơi làm việc | VARCHAR |  | Không |
| 13 | birthday | Ngày sinh | DATE |  | Không |
| 14 | modify\_date | Ngày chỉnh sửa | DATETIME |  | Không |
| 15 | create\_date | Ngày khởi tạo | DATETIME |  | Không |
| 16 | create\_by | Người khởi tạo | VARCHAR |  | Có |
| 17 | modify\_by | Người chỉnh sửa | VARCHAR |  | Có |
| 18 | home\_town | Quê quán | MEDIUMTEXT |  | Không |

Bảng 12. Bảng dữ liệu người dùng (user)

Bảng trọ thuê (motel\_room)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ràng buộc | NULL |
| 1 | id | Mã trọ | INT | Khóa chính | Không |
| 2 | title | Tiêu đề | VARCHAR |  | Không |
| 3 | id\_host | Mã chủ trọ | VARCHAR | Khóa ngoài, tham chiếu đến bảng người dùng | Không |
| 4 | description | Mô tả | MEDIUMTEXT |  | Không |
| 5 | images | Ảnh minh họa (đường dẫn ảnh) | MEDIUMTEXT |  | Không |
| 6 | area | Diện tích | INT |  | Không |
| 7 | max\_person | Số người tối đa | INT |  | Không |
| 8 | price | Giá tiền | INT |  | Không |
| 9 | address | Địa chỉ | VARCHAR |  | Không |
| 10 | location | Tọa độ  (lat, lng) | VARCHAR |  | Không |
| 11 | create\_date | Ngày khởi tạo | DATETIME |  | Không |
| 12 | modify\_date | Ngày chỉnh sửa | DATETIME |  | Không |
| 13 | deleted | Bị xóa hay chưa | BIT |  | Không |
| 14 | id\_category | Mã danh mục | INT | Khóa ngoài, tham chiếu đến bảng danh mục | Không |
| 15 | priority\_object | Đối tượng ưu tiên | VARCHAR |  | Có |
| 16 | modify\_by | Người chỉnh sửa | VARCHAR |  | Không |
| 17 | create\_by | Người khởi tạo | VARCHAR |  | Không |
| 18 | floors | Số tầng | INT |  | Không |
| 19 | History\_price | Lịch sử giá | TEXT |  | Không |

Bảng 13. Bảng dữ liệu trọ cho thuê (motel\_room)

Bảng danh mục (category)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ràng buộc | NULL |
| 1 | id | Mã danh mục | INT | Khóa chính | Không |
| 2 | name | Tên danh mục | VARCHAR |  | Không |
| 3 | deleted | Bị xóa hay chưa | BIT |  | Không |
| 4 | modify\_date | Ngày chỉnh sửa | DATETIME |  | Không |
| 5 | create\_date | Ngày khởi tạo | DATETIME |  | Không |
| 6 | create\_by | Người khởi tạo | VARCHAR |  | Không |
| 7 | modify\_by | Người chỉnh sửa | VARCHAR |  | Không |

Bảng 14. Bảng dữ liệu danh mục (category)

Bảng tiện ích (convenient)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ràng buộc | NULL |
| 1 | id | Mã tiện ích | INT | Khóa chính | Không |
| 2 | name | Tên tiện ích | VARCHAR |  | Không |
| 3 | deleted | Bị xóa hay chưa | BIT |  | Không |
| 4 | modify\_date | Ngày chỉnh sửa | DATETIME |  | Không |
| 5 | create\_date | Ngày khởi tạo | DATETIME |  | Không |
| 6 | create\_by | Người khởi tạo | VARCHAR |  | Không |
| 7 | modify\_by | Người chỉnh sửa | VARCHAR |  | Không |

Bảng 15. Bảng dữ liệu tiện ích (convenient)

Bảng phân quyền (role)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ràng buộc | NULL |
| 1 | id | Mã quyền | INT | Khóa chính | Không |
| 2 | name | Tên quyền | VARCHAR |  | Không |
| 3 | content | Nội dung quyền hạn | VARCHAR |  | Không |

Bảng 16. Bảng dữ liệu phân quyền (role)

Bảng bình luận (report)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ràng buộc | NULL |
| 1 | id\_cmt | Mã bình luận | INT | Khóa chính | Không |
| 2 | id\_user | Mã người dùng | INT | Khóa ngoài, tham chiếu đến bảng người dùng | Không |
| 3 | id\_room | Mã trọ | INT | Khóa ngoài, tham chiếu đến bảng trọ thuê | Không |
| 4 | rate | Đánh giá  (Từ 1 => 5\*) | INT |  | Không |
| 5 | comment | Nội dung bình luận | MEDIUMTEXT |  | Không |
| 6 | deleted | Bị xóa hay chưa | BIT |  | Không |
| 7 | modify\_date | Ngày chỉnh sửa | DATETIME |  | Không |
| 8 | create\_date | Ngày khởi tạo | DATETIME |  | Không |
| 9 | modify\_by | Người chỉnh sửa | VARCHAR |  | Không |
| 10 | create\_by | Người khởi tạo | VARCHAR |  | Không |

Bảng 17. Bảng dữ liệu bình luận (report)

Bảng tiện ích của trọ (room\_has\_convenient)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ràng buộc | NULL |
| 1 | id\_room | Mã trọ | INT | Khóa chính,  Khóa ngoài, tham chiếu đến bảng trọ thuê | Không |
| 2 | id\_convenient | Mã tiện ích | INT | Khóa chính,  Khóa ngoài, tham chiếu đến bảng người dùng | Không |
| 3 | deleted | Bị xóa hay chưa | BIT |  | Không |
| 4 | modify\_date | Ngày chỉnh sửa | DATETIME |  | Không |
| 5 | create\_date | Ngày khởi tạo | DATETIME |  | Không |
| 6 | create\_by | Người khởi tạo | VARCHAR |  | Không |
| 7 | modify\_by | Người chỉnh sửa | VARCHAR |  | Không |

Bảng 18. Bảng dữ liệu tiện ích của trọ (room\_has\_convenient)

Bảng thuê trọ (tenant)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ràng buộc | NULL |
| 1 | id\_user | Mã người thuê | INT | Khóa chính,  Khóa ngoài, tham chiếu đến bảng người dùng | Không |
| 2 | id\_room | Mã trọ | INT | Khóa chính,  Khóa ngoài, tham chiếu đến bảng trọ thuê | Không |
| 3 | create\_date | Ngày khởi tạo | DATETIME |  | Không |
| 4 | deleted | Bị xóa hay chưa | BIT |  | Không |
| 5 | status | Đã được chấp nhận hay chưa | BIT |  | Không |
| 6 | modify\_date | Ngày chỉnh sửa | DATETIME |  | Không |
| 7 | create\_by | Người khởi tạo | VARCHAR |  | Không |
| 8 | modify\_by | Người chỉnh sửa | VARCHAR |  | Không |

Bảng 19. Bảng dữ liệu thuê trọ (tenant)

## Xây dựng ứng dụng

### Thư viện và công cụ sử dụng

**Danh sách công cụ sử dụng**

|  |  |  |
| --- | --- | --- |
| **Mục đích** | **Công cụ** | **Địa chỉ URL** |
| IDE lập trình | Intellij IDE  Visual Studio Code | <https://www.jetbrains.com/idea/>  <https://code.visualstudio.com> |
| Ngôn ngữ lập trình | Java | <https://www.java.com> |
| Framework | Spring | <https://spring.io> |
| Cơ sở dữ liệu | MySQL Workrech 8.0 CE | <https://www.mysql.com> |
| Thiết kế giao diện | HTML  CSS  JS | <https://www.w3schools.com> |

Bảng 20. Danh sách công cụ sử dụng

Lý do sử dụng:

* IDE lập trình:
* Intellij là IDE lập trình hỗ trợ việc xây dựng website, kết nối cơ sở dữ liệu và đặc biệt là các framework của Spring.
* Visual Studio Code là IDE lập trình hỗ trợ cho việc thiết kế giao diện tiện lợi với HTML/CSS/JS
* Ngôn ngữ lập trình: Java là ngôn ngữ lập trình hướng đối tượng đồng thời hỗ trợ việc lập trình web phần backend
* Framework: Spring hỗ trợ rất nhiều cho xây dựng và phát triển web, quản lý cơ sở dữ liệu, xác thực và bảo mật
* Cơ sở dữ liệu: MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở phổ biến trong quá trình xây dựng, phát triển ứng dụng. Đây là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có khả năng thay đổi mô hình sử dụng phù hợp với điều kiện công việc khả chuyển. MySQL hoạt động trên nhiều hệ điều hành, cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.

### Kết quả đạt được

Hoàn thiện các yêu cầu cơ bản của đề tài, chạy được đúng theo các thiết kế từ trước, mới triển khai trên local.

**Thông tin ứng dụng**

|  |  |
| --- | --- |
| **Thông số** | **Công cụ** |
| Số dòng code | ~3000 |
| Số lớp | ~60 |
| Dung lượng  mã nguồn | ~10MB |

Bảng 21. Thông tin ứng dụng

### Minh hoạ các chức năng chính

Tương tự phần Thiết kế giao diện [4.2.1](#_Thiết_kế_giao)

## Kiểm thử

Sử dụng kiểm thử hộp xám một cách cơ bản với một vài chứ năng chinh

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Chức năng** | **Dữ liệu test** | **Kết quả mong muốn** | **Kết quả thực tế** | **Kết quả** | **Nhận xét** |
| Tìm kiếm trọ | Null | Lấy tất cả các trọ trong hệ thống | Lấy tất cả các trọ trong hệ thống | Thành công |  |
| {  Category:1,  Price: 1  } | Hiển thị các trọ tương ứng | Hiển thị 3 trọ | Thành công | Price: Loại giá tiền (dưới 1 triệu,…)  Category: Loại trọ |
| Yêu cầu thuê trọ | Null | Chưa yêu cầu | Thất bại do chưa yêu cầu | Thành công | Do chưa tạo tài khoản |
| {  idUser: 1,  idRoom: 2  } | Yêu cầu thành công | Yêu cầu thành công | Thành công |  |
| {  idUser: 2,  idRoom: 2  } | Yêu cầu thất bại | Thất bại | Thành công | Do trọ đã đầy |
| Hủy yêu cầu | Null | Chưa yêu cầu | Thất bại do chưa yêu cầu | Thành công | Do chưa đăng nhập |
| {  idUser: 1,  idRoom: 2  } | Yêu cầu thành công | Yêu cầu thành công | Thành công |  |
| {  idUser: 1,  idRoom: 2  } | Yêu cầu thất bại | Thất bại | Thành công | Do bạn chưa tạo yêu cầu |
| Xác nhận thuê trọ | {  idUser: 1,  idRoom: 2  } | Yêu cầu thành công | Yêu cầu thành công | Thành công |  |
| {  idUser: 2,  idRoom: 2  } | Yêu cầu thất bại | Thất bại | Thành công | Do trọ bị đầy |

## Triển khai

Khi truy cập vào website, người dùng sẽ được sử dụng các chức năng tương ứng với quyền của mình (như người thuê, chủ trọ hay quản lý) thông qua việc gửi các yêu cầu đến Server và sẽ nhận được các phản hồi phù hợp.

Hiện tại chương trình mới triển khai trên local (máy tính cá nhân) chưa được triển khai trong thực tế.

# Các giải pháp và đóng góp nổi bật

**5.1. Đặt vấn đề**

Vấn đề kết nối với CSDL và ánh xạ các bảng trong CSDL với các đối tượng model trong xây dựng phần mềm là một vấn đề giải quyết chưa triệt để nếu sử dụng các cách cũ như dùng JDBC để kết nối. Như khi tạo đối tượng để kết nối CSDL hay việc kết nối các bảng dữ liệu với model cũng cần xử lý qua nhiều bước, cần nhiều phương thức để xử lý.

**5.2. Giải pháp**

Trong phạm vi của Project 3, em sử dụng Spring Data JPA được hỗ trợ với Spring Boot để giải quyết vấn đề này.

Spring Boot JPA là một bản ghi chi tiết của Java để quản lý dữ liệu quan hệ trong các ứng dụng Java. Nó cho phép chúng ta truy cập và lưu trữ dữ liệu giữa các object/class Java và database quan hệ. JPA tuân theo Object-Relation Mapping (ORM). Nó là một tập hợp các interface. Nó cũng cung cấp một API EntityManager runtime để xử lý các câu query và giao dịch trên các object dựa trên database. Nó sử dụng ngôn ngữ truy vấn hướng đối tượng độc lập nền tảng JPQL (Java Persistent Query Language).

JPA bao gồm ba lĩnh vực:

* The Java Persistence API
* Object-Relational metadata
* Bản thân API, được định nghĩa trong persistence package.

**5.3. Kết quả đạt được**

JPA đơn giản hơn, gọn gàng hơn và ít tốn công sức hơn là JDBC, SQL hay ánh xạ viết tay. JPA thích hợp cho các ứng dụng phức tạp không tuân theo hiệu suất. Ưu điểm chính của JPA so với JDBC là: trong JPA, dữ liệu được biểu thị bằng các object và class trong khi trong JDBC dữ liệu được biểu diễn bằng các bảng và bản ghi. Nó sử dụng POJO để biểu diễn dữ liệu liên tục giúp đơn giản hóa việc lập trình database. Có một số ưu điểm khác của JPA:

* JPA tránh viết Ngôn Ngữ Định Nghĩa Dữ Liệu (DDL) bằng phương ngữ SQL dành riêng cho database. Thay vào đó nó cho phép ánh xạ trong XML hoặc sử dụng các annotation Java.
* JPA cho phép chúng ta tránh viết Ngôn Ngữ Thao Tác Dữ Liệu (DML) bằng phương ngữ SQL dành riêng cho database.
* JPA cho phép chúng ta save và load các object và đồ thị Java mà không cần bất kỳ ngôn ngữ DML nào.
* Khi chúng ta cần thực hiện các truy vấn JPQL, nó cho phép chúng ta viết các truy vấn dưới dạng các entity Java mà không phải là bảng và cột SQL.

Ví dụ minh họa:

Ảnh có chứa văn bản

Mô tả được tạo tự động

Ngoài ra còn sử dụng Loombok để tự động tạo các hàm getter/setter hay constructor cần thiết.

Hầu hết các chức năng đều thực thi các thiết kế ban đầu, không có quá nhiều khó khăn trong việc triển khai.

# Kết luận và hướng phát triển

## Kết luận

*Ưu điểm*

* + Thực hiện được các đặc tả yêu cầu đã đề ra từ bước phân tích và thiết kế ban đầu
  + Tạo ra sản phẩm hoàn chỉnh là 1 website có đầy đủ các chức năng
  + Sản phẩm giúp nhằm kết nối người thuê và chủ trọ một cách dễ dàng

*Hạn chế*

* + Việc tối ưu hiệu quả của sản phẩm chưa được tốt
  + Giao diện của sản phẩm vẫn cần chỉnh sửa
  + Việc phân quyền người dùng còn chưa rõ ràng, cần phân tích kĩ hơn
  + Cơ sở dữ liệu vẫn còn thiếu sót
  + Một số công việc liên quan đến thông báo còn cần phải người tự gửi
  + Quy trình kiểm thử còn sơ sài, không được thực hiện một cách đầy đủ

## Hướng phát triển

* + Tối ưu hóa sản phẩm và chỉnh sửa giao diện đẹp mắt hơn. Đồng thời liên kết đăng nhập qua google, facebook
  + Sử dụng các công nghệ như React, Vue,.. thay thế cho template engine
  + Phân tích thiết kế hệ thống về đề tài một cách đầy đủ, chi tiết và hợp lý
  + Xây dựng cơ sở dữ liệu đầy đủ và chi tiết hơn, bao quát toàn cảnh được đề tài
  + Phát triển sản phẩm trên ứng dụng máy tính cũng như trên điện thoại thông minh
  + Về mặt thị trường, website có thể mở rộng địa bàn hoạt động ra cả nước, cũng như phát triển thêm các sản phẩm mới như tìm kiếm bất động sản,…
  + Thực hiện tốt quy trình kiểm thử và khảo sát thị trường
  + Tạo hệ thống tự cập nhật các thông tin mới liên quan đến nhà trọ và hệ thống trả lời email feeback cho người dùng

# Tài liệu tham khảo

|  |  |  |
| --- | --- | --- |
| [1] |  | W3Schools, “W3Schools Online Web Tutorials,” [Trực tuyến]. Available: https://www.w3schools.com. |
| [2] |  | o7planning, “O7 Planning,” [Trực tuyến]. Available: https://o7planning.org/vi/11669/spring-boot. |
| [3] |  | springboot, “Spring Boot,” [Trực tuyến]. Available: https://spring.io/projects/spring-boot. |
| [4] |  | SpringJPA, “Giới thiệu về Spring JPA,” [Trực tuyến]. Available: https://techmaster.vn/posts/36255/gioi-thieu-ve-spring-jpa. |