TRƯỜNG ĐẠI HỌC PHENIKAA

KHOA CÔNG NGHỆ THÔNG TIN  
   
   


Báo Cáo Dự Án Học Phần Phân Tích Và Thiết Kế Phần Mềm

Đề tài: Phát triển website quản lý phòng trọ cho thuê

Nhóm 12 – Lớp N02

|  |  |
| --- | --- |
| Thành viên: | * Ngô Thị Linh – 21011613 |
| Giảng viên hướng dẫn: | TS. Mai Thuý Nga |

Hà Nội, tháng 10 năm 2024

# BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |  |
| --- | --- | --- |
| **Vai trò** | | **Ngô Thị Linh** |
|  |
| **Phần** | **Nội dung công việc** | **Phần % công việc thực hiện** |  |
| **I** | Báo cáo BTL phần 1 | 100% |  |
| Báo cáo BTL phần 2 | 100% |  |
| Báo cáo BTL phần 3 | 100% |  |
| Báo cáo BTL phần 4 | 100% |  |
| Báo cáo BTL phần 5 | 100% |  |
| **II** | Fontend | 100% |  |
| Backend | 100% |  |
| Database | 100% |  |
| **III** | Slide | 100% |  |
| **IV** | Kiểm tra báo cáo | 100% |  |
| Đánh giá quá trình dự án | 100% |  |
| **V** | Toàn bộ dự án | 100% |  |

**Mục lục**

[1. Yêu cầu (Requirements) 3](#_Toc181475864)

[**1.1 Đặt vấn đề (Problem statement) 3**](#_Toc181475865)

[***1.1.1* Mô tả vấn đề 3**](#_Toc181475866)

[***1.1.2* Giải pháp 4**](#_Toc181475867)

[**Một số giải pháp cơ bản của một ứng dụng quản lý phòng trọ có thể phục vụ: 4**](#_Toc181475868)

[**1.2 Thuật ngữ (Glossary) 5**](#_Toc181475869)

[**1.3 Thông số kỹ thuật bổ sung 7**](#_Toc181475870)

[***1.3.1* Khả năng sử dụng: 7**](#_Toc181475871)

[***1.3.2* Hiệu suất: 7**](#_Toc181475872)

[***1.3.3* Bảo mật: 7**](#_Toc181475873)

[***1.3.4* Tương thích đa nền tảng: 8**](#_Toc181475874)

[***1.3.5* Đảm bảo và khả năng phục hồi: 8**](#_Toc181475875)

[**1.4 Mô hình hóa chức năng 8**](#_Toc181475876)

[***1.4.1* Các yêu cầu chức năng 8**](#_Toc181475877)

[***1.4.2* Sơ đồ Use-case 9**](#_Toc181475878)

[**1.5 Đặc tả các Use-case 10**](#_Toc181475879)

[***1.5.1* UC1. Use-case đăng nhập 10**](#_Toc181475880)

[***1.5.2* UC2: Use-case thống kê 12**](#_Toc181475881)

[***1.5.3* UC3: Use-case quản lý phòng trọ 16**](#_Toc181475882)

[***1.5.4* UC4: Use-case quản lý khách thuê 19**](#_Toc181475883)

[***1.5.5* UC5: Use-case quản lý chỉ số tiền điện, nước 21**](#_Toc181475884)

[***1.5.6* UC6: Use-case quản lý hóa đơn 23**](#_Toc181475885)

[2. Phân tích trường hợp sử dụng (Use-case analysis) 26](#_Toc181475886)

[**2.1 Phân tích kiến trúc hệ thống 27**](#_Toc181475887)

[***2.1.1* Kiến trúc mức cao của hệ thống 27**](#_Toc181475888)

[***2.1.2* Các đối tượng trừu tượng chính của hệ thống (Key abstractions) 30**](#_Toc181475889)

[**2.2 Thực thi trường hợp sử dụng (Use-case relizations) 30**](#_Toc181475890)

[***2.2.1* Các biểu đồ tuần tự (Sequence diagrams) 30**](#_Toc181475891)

[***2.2.2* Góc nhìn của các lớp trong hệ thống (Views of participating classes) 43**](#_Toc181475892)

[3. Thiết kế (Use-case design) 47](#_Toc181475893)

[**3.1 Xác định các thành phần thiết kế (Identify design elements) 47**](#_Toc181475894)

[***3.1.1* Xác định các lớp (Identify classes) 47**](#_Toc181475895)

[***3.1.2* Xác định các hệ thống con và giao diện (Identify subsystems and interfaces) 50**](#_Toc181475896)

[***3.1.3* Xác định các gói (Identify packages) 51**](#_Toc181475897)

[**3.2 Thiết kế trường hợp sử dụng (Use-case design) 51**](#_Toc181475898)

[***3.2.1* Thiết kế các biểu đồ tuần tự (Design sequence diagrams) 51**](#_Toc181475899)

[***3.2.2* Thiết kế biểu đồ lớp (Class diagrams) 58**](#_Toc181475900)

[**3.3 Thiết kế cơ sở dữ liệu (Database design) 60**](#_Toc181475901)

[***3.3.1* Lược đồ cơ sở dữ liệu 60**](#_Toc181475902)

[***3.3.2* Chi tiết các bảng 60**](#_Toc181475903)

[4. Cài đặt 62](#_Toc181475904)

[**4.1 Lựa chọn công nghệ 62**](#_Toc181475905)

[***4.1.1* Java 62**](#_Toc181475906)

[***4.1.2* Java Swing 62**](#_Toc181475907)

[***4.1.3* MySQL 62**](#_Toc181475908)

[***4.1.4* NetBeans 62**](#_Toc181475909)

[***4.1.5* XAMPP 62**](#_Toc181475910)

[***4.1.6* GitHub 63**](#_Toc181475911)

[**4.2 Cấu trúc mã nguồn 63**](#_Toc181475912)

[5. Kết luận 64](#_Toc181475913)

[Tài liệu tham khảo 65](#_Toc181475914)

# Yêu cầu (Requirements)

## Đặt vấn đề (Problem statement)

### Mô tả vấn đề

Hiện nay, sự phát triển vượt bậc của công nghệ thông tin (CNTT) đang từng bước làm thay đổi mạnh mẽ nền kinh tế và xã hội Việt Nam. Tin học không chỉ được ứng dụng rộng rãi trong các lĩnh vực quản lý, kinh doanh, và du lịch mà còn trở thành công cụ đắc lực giúp cải tiến quy trình làm việc trong nhiều ngành khác nhau. Ở Việt Nam, tuy ngành CNTT còn tương đối mới, nhưng nhờ vào tốc độ phát triển nhanh chóng, CNTT đã và đang ngày càng được ứng dụng sâu rộng, trở thành yếu tố quan trọng thúc đẩy sự tăng trưởng và tiến bộ xã hội.

Việc đưa CNTT vào công tác quản lý đã giúp giảm thiểu khối lượng công việc thủ công, tiết kiệm thời gian và nâng cao độ chính xác so với việc lưu trữ và xử lý trên giấy tờ trước đây. CNTT cho phép hệ thống hóa dữ liệu, tạo không gian lưu trữ thông tin gọn nhẹ hơn, hạn chế mất mát và thất lạc dữ liệu. Hệ thống công nghệ còn giúp tự động hóa quy trình, cung cấp các thông tin cụ thể hóa theo yêu cầu, từ đó nâng cao hiệu quả quản lý và dễ dàng phục vụ người dùng một cách thuận tiện.

Một trong những ứng dụng tiêu biểu của CNTT là hệ thống phần mềm quản lý nhà trọ. Mô hình quản lý này được xem như một bước tiến mới, hỗ trợ đắc lực cho chủ nhà trọ trong việc theo dõi và quản lý các thông tin liên quan đến việc cho thuê phòng. Hệ thống quản lý nhà trọ hoạt động độc lập, giúp các chủ nhà kiểm soát dễ dàng hơn, từ việc quản lý hợp đồng thuê, theo dõi tình trạng phòng, đến việc quản lý thu chi một cách chính xác và hiệu quả. Sự ra đời của phần mềm quản lý nhà trọ không chỉ giúp giảm thiểu khối lượng công việc mà còn mang lại sự tiện lợi và hiệu quả cao trong công tác quản lý, góp phần thúc đẩy sự phát triển chung của ngành CNTT trong đời sống xã hội.

### Giải pháp

Một số giải pháp cơ bản của một ứng dụng quản lý phòng trọ có thể phục vụ:

- *Quản lý phòng trọ*: Cho phép chủ nhà cập nhật thông tin chi tiết của các phòng cho thuê như diện tích, giá cả, tiện ích đi kèm, tình trạng phòng (đã cho thuê, sẵn sàng cho thuê).

- *Quản lý người thuê*: Cung cấp chức năng lưu trữ thông tin người thuê, bao gồm hồ sơ cá nhân, thông tin liên lạc và lịch sử thuê nhà, giúp chủ nhà dễ dàng theo dõi.

- *Quản lý thu chi*: Tích hợp hệ thống theo dõi và quản lý các khoản thu, chi liên quan đến việc cho thuê phòng, bao gồm tiền thuê hàng tháng, các khoản phí tiện ích, tiền cọc, và các chi phí phát sinh.

- *Thông báo và nhắc nhở*: Gửi thông báo tự động cho người thuê khi đến hạn thanh toán, hoặc cho chủ nhà khi hợp đồng sắp hết hạn, giúp đảm bảo tiến độ và hạn chế tình trạng nợ.

- *Quản lý bảo trì và sửa chữa*: Theo dõi yêu cầu bảo trì và sửa chữa của người thuê, đảm bảo phòng được bảo trì định kỳ, đáp ứng nhanh chóng các yêu cầu sửa chữa, nâng cao sự hài lòng của người thuê.

- *Bảo mật và quyền riêng tư*: Đảm bảo rằng thông tin cá nhân của người thuê và chủ nhà được bảo mật, tuân thủ các quy định về bảo mật và quyền riêng tư.

- *Báo cáo và thống kê*: Cung cấp báo cáo về tình hình cho thuê, tỷ lệ lấp đầy, thu nhập, và các thông tin phân tích khác, giúp chủ nhà đưa ra các quyết định tối ưu về kinh doanh.

## Thuật ngữ (Glossary)

|  |  |
| --- | --- |
| **Thuật ngữ** | **Giải thích** |
| Phòng trọ (Rental Room) | Loại phòng cho thuê dành cho người ở tạm thời hoặc dài hạn, thường có quy mô nhỏ và chi phí thấp hơn các loại hình nhà ở khác. |
| Chủ nhà (Landlord) | Người sở hữu hoặc quản lý phòng trọ, chịu trách nhiệm cho thuê và duy trì phòng trọ. |
| Người thuê (Tenant) | Cá nhân hoặc nhóm người thuê phòng trọ để ở, thường trả phí thuê định kỳ cho chủ nhà. |
| Hợp đồng thuê (Rental Agreement) | Thỏa thuận pháp lý giữa chủ nhà và người thuê, xác định các điều khoản về giá thuê, thời hạn và trách nhiệm của mỗi bên. |
| Tiền thuê (Rent) | Khoản tiền mà người thuê phải trả cho chủ nhà hàng tháng hoặc theo kỳ hạn đã thỏa thuận trong hợp đồng thuê. |
| Quản lý khách thuê (Tenant Management) | Theo dõi, lưu trữ thông tin cá nhân, tình trạng thuê, và lịch sử thanh toán của khách thuê. |
| Quản lý phòng trọ (Room Management) | Tổ chức, duy trì, và cập nhật thông tin phòng trọ, bao gồm trạng thái phòng và các điều kiện liên quan đến việc cho thuê. |
| Quản lý hóa đơn (Invoice Management) | Tạo, lưu trữ và theo dõi các hóa đơn tiền thuê và tiện ích liên quan đến khách thuê. |
| Quản lý chỉ số tiền điện, nước (Utility Meter Management) | Ghi nhận và cập nhật chỉ số tiêu thụ điện, nước của từng phòng để tính toán phí tiện ích hàng tháng. |
| Quản lý thống kê (Statistics Management) | Tổng hợp, báo cáo các số liệu về tỉ lệ phòng trống, tình hình thu chi, và hiệu suất kinh doanh để hỗ trợ ra quyết định quản lý. |
| Thuật ngữ | Giải thích |
| Phòng trọ (Rental Room) | Loại phòng cho thuê dành cho người ở tạm thời hoặc dài hạn, thường có quy mô nhỏ và chi phí thấp hơn các loại hình nhà ở khác. |
| Chủ nhà (Landlord) | Người sở hữu hoặc quản lý phòng trọ, chịu trách nhiệm cho thuê và duy trì phòng trọ. |
| Người thuê (Tenant) | Cá nhân hoặc nhóm người thuê phòng trọ để ở, thường trả phí thuê định kỳ cho chủ nhà. |

*Bảng 2. Bảng thuật ngữ*

## Thông số kỹ thuật bổ sung

Nếu kết hợp được các yêu cầu phi chức năng này thì hệ thống quản lý trọ hoạt động an toàn, hiệu quả và mang lại trải nghiệm tích cực cho người sử dụng trang web.

### Khả năng sử dụng:

* *Giao diện dễ hiểu:* Giao diện thiết kế đơn giản, trực quan giúp người dùng nhanh chóng nắm bắt và sử dụng các tính năng.
* *Ngôn ngữ bổ sung*: Bổ sung thêm ngôn ngữ khi lượng người dùng tăng, thể hiện tính linh hoạt và đáp ứng nhu cầu đa dạng của hệ thống.

### Hiệu suất:

* *Thời gian tải nhanh*: Đảm bảo trang web tải nhanh ngay cả khi có nhiều người dùng, nâng cao trải nghiệm.
* *Xử lý dữ liệu lớn*: Tối ưu hóa hệ thống để xử lý dữ liệu về phòng trọ và người thuê một cách hiệu quả.

### Bảo mật:

* *Bảo vệ dữ liệu cá nhân*: Đảm bảo an toàn cho dữ liệu người dùng, xây dựng lòng tin.
* *Công nghệ bảo mật tiêu chuẩn*: Sử dụng mã hóa SSL, xác thực hai yếu tố, và bảo vệ khỏi các rủi ro an ninh mạng.

### Tương thích đa nền tảng:

* *Truy cập từ nhiều thiết bị*: Hệ thống có thể truy cập từ máy tính, điện thoại, và máy tính bảng, giúp người dùng dễ dàng truy cập từ các nền tảng khác nhau.

### Đảm bảo và khả năng phục hồi:

* *Khả năng sẵn sàng cao*: Hệ thống hoạt động liên tục, hạn chế thời gian ngừng hoạt động.
* *Phục hồi nhanh chóng*: Hệ thống cần khả năng phục hồi nhanh để tránh gián đoạn dịch vụ.

## Mô hình hóa chức năng

### Các yêu cầu chức năng

Dưới đây là các yêu cầu chức năng được phân cấp theo nhóm chức năng của sơ đồ use case mà bạn đã cung cấp:

*UC1. Nhóm chức năng đăng nhập cho người quản trị*

* Hệ thống cho phép người quản trị đăng nhập vào chương trình bằng tài khoản.

*UC2. Nhóm chức năng thống kê phòng trọ*

* Hệ thống hiển thị tổng doanh thu mỗi tháng.
* Hệ thống hiển thị số lượng phòng trống.
* Hệ thống hiển thị danh sách khách còn nợ tiền phòng.
* Hệ thống hiển thị tổng số khách hàng.

*UC3. Nhóm chức năng quản lý phòng trọ*

* Hệ thống cho phép thêm, sửa, xóa thông tin phòng trọ.
* Hệ thống hiển thị danh sách các phòng trọ.

*UC4. Nhóm chức năng quản lý thuê phòng*

* Hệ thống cho phép thêm, sửa, xóa thông tin khách hàng thuê phòng.
* Hệ thống hiển thị danh sách thông tin khách hàng đã thuê phòng.

*UC5. Nhóm chức năng quản lý chỉ số điện nước*

* Hệ thống hiển thị chỉ số điện nước của từng phòng theo tháng.
* Hệ thống tính tiền điện nước hàng tháng cho từng phòng.

*UC6. Nhóm chức năng hóa đơn*

* Hệ thống cho phép tìm kiếm, thêm, xóa và xuất hóa đơn.
* Hệ thống hiển thị thông tin các hóa đơn trong cơ sở dữ liệu.

Các yêu cầu trên được phân cấp chi tiết theo từng nhóm chức năng cụ thể, giúp việc quản lý và phát triển từng phần trong hệ thống dễ dàng và hiệu quả hơn.

### Sơ đồ Use-case

**Sơ đồ Use-case tổng quát**

A diagram of a diagram

Description automatically generated with medium confidence

[*Hình 1.1. Sơ đồ use-case tổng quát*](#_heading=h.ghcmwqkqvshf)

Quản trị viên: Là người quản trị hệ thống, có các quyền và chức năng cao nhất với hệ thống: đăng nhập, quản lý phòng trọ, quản lý người thuê trọ, quản lý chỉ số điện nước, hóa đơn và thống kê với các chức năng như thêm, cập nhật, xóa, xem chi tiết.

## Đặc tả các Use-case

### UC1. Use-case đăng nhập

A red arrow pointing to a white oval

Description automatically generated

*Hình 1.2. Use case đăng nhập*

|  |  |
| --- | --- |
| Use case | Đăng nhập |
| Tác nhân | Admin |
| Mô tả ngắn gọn | Use case này cho phép Admin đăng nhập vào hệ thống. |
| Tiền điều kiện | Admin có tài khoản trong hệ thống |
| Luồng điều kiện | 1. Người dùng chạy chương trình. 2. Hệ thống hiển thị màn hình đăng nhập. 3. Người dùng nhập tên đăng nhập, mật khẩu và nhấn nút “Đăng nhập”. 4. Hệ thống kiểm tra thông tin đăng nhập.  - Nếu hợp lệ: người dùng được truy cập theo quyền hạn. - Nếu không hợp lệ: hệ thống hiển thị thông báo lỗi và yêu cầu nhập lại. |
| Ngoại lệ | - Nếu không kết nối được với cơ sở dữ liệu: Hệ thống hiển thị thông báo lỗi và kết thúc use case. - Nếu người dùng quên mật khẩu: Cung cấp tùy chọn lấy lại mật khẩu. |
| Hậu điều kiện | Người dùng đã đăng nhập thành công và có thể truy cập vào hệ thống theo quyền hạn tương ứng. Nếu không thành công, hệ thống vẫn ở trạng thái chưa đăng nhập. |

*Bảng 3. Use case đăng nhập*

A screenshot of a screenshot of a website

Description automatically generated

*Hình 1.3. Giao diện đăng nhập*

### UC2: Use-case thống kê

A group of ovals with text

Description automatically generated

*Hình 1.4. Use case thống kê*

|  |  |
| --- | --- |
| Use case | Thống kê |
| Tác nhân | Admin |
| Mô tả ngắn gọn | Use case này cho phép Admin xem thông tin về tổng doanh thu, số phòng còn trống, khách nợ tiền phòng, tổng khách hàng được thống kê theo các bảng chi tiết. |
| Tiền điều kiện | Admin đăng nhập thành công vào hệ thống |
| Luồng điều kiện | 1. Hệ thống hiển thị màn hình thống kê.  2. Người dùng chọn 1 trong 4 chức năng: - Thống kê tổng doanh thu - Thống kê số phòng còn trống - Thống kê khách nợ tiền phòng - Thống kê tổng khách hàng 3. Hệ thống hiển thị cửa sổ với bảng thống kê chi tiết tương ứng với lựa chọn của người dùng. 4. Use case kết thúc. |
| Ngoại lệ | Không có ngoại lệ đặc biệt |
| Hậu điều kiện | Hệ thống hiển thị bảng thống kê chi tiết cho chức năng đã chọn của Admin. |

*Bảng 4. Use case thống kê*

A screenshot of a computer

Description automatically generated

*Hình 1.5. Giao diện thống kê phòng trọ*

A screenshot of a computer

Description automatically generated

*Hình 1.6. Giao diện danh sách phòng*

A screenshot of a computer

Description automatically generated

*Hình 1.7. Giao diện danh sách khách hàng*

A screenshot of a computer

Description automatically generated

*Hình 1.8. Giao diện thông tin khách nợ phòng*

A screenshot of a computer

Description automatically generated

*Hình 1.9. Giao diện Thống kê*

### UC3: Use-case quản lý phòng trọ

A yellow and black diagram

Description automatically generated with medium confidence

*Hình 1.10. Use case quản lý phòng trọ*

|  |  |
| --- | --- |
| Use case | Quản lý phòng trọ |
| Tác nhân | Admin |
| Mô tả ngắn gọn | Use case cho phép Admin xem, thêm, sửa, xóa thông tin phòng trọ từ bảng tblQlyPhongTro. |
| Tiền điều kiện | Admin đăng nhập thành công vào hệ thống. |
| Luồng điều kiện | 1. Admin chọn “Quản lý phòng trọ” trên menu.  2. Hệ thống lấy thông tin từ tblQlyPhongTro và hiển thị danh sách phòng trọ.  3. Admin thực hiện chức năng thêm, xóa, sửa phòng trọ như mô tả.  4. Hệ thống cập nhật và hiển thị thông báo kết quả.  5. Use case kết thúc. |
| Ngoại lệ | - Nếu nhập thông tin không hợp lệ, hệ thống báo lỗi.  - Nếu mất kết nối CSDL, hệ thống báo lỗi và Use case kết thúc. |
| Hậu điều kiện | Dữ liệu phòng trọ được cập nhật và hiển thị trên bảng thống kê. |

*Bảng 5. Use case quản lý phòng trọ*

A screenshot of a computer

Description automatically generated

*Hình 1.11. Giao diện Quản lý phòng trọ*

A screenshot of a computer

Description automatically generated

*Hình 1.12. Giao diện Thêm phòng trọ*

### UC4: Use-case quản lý khách thuê

A diagram of a network

Description automatically generated

*Hình 1.13. Use case quản lý khách thuê*

|  |  |
| --- | --- |
| Use case | Quản lý khách thuê |
| Tác nhân | Admin |
| Mô tả ngắn gọn | Use case cho phép Admin xem, thêm, sửa, xóa thông tin về khách thuê từ bảng tblKhachThue. |
| Tiền điều kiện | Admin đăng nhập thành công vào hệ thống. |
| Luồng điều kiện | 1. Admin chọn “Quản lý người phòng trọ” trên menu.  2. Hệ thống lấy thông tin khách thuê từ tblKhachThue và hiển thị danh sách.  3. Admin thực hiện chức năng thêm, xóa, sửa khách thuê theo yêu cầu.  4. Hệ thống cập nhật và hiển thị thông báo kết quả.  5. Use case kết thúc. |
| Ngoại lệ | - Nếu thông tin khách thuê không hợp lệ, hệ thống báo lỗi.  - Nếu mất kết nối CSDL, hệ thống báo lỗi và Use case kết thúc. |
| Hậu điều kiện | Dữ liệu khách thuê được cập nhật và hiển thị trên bảng thống kê. |

*Bảng 6. Use case quản lý khách thuê*

A screenshot of a computer

Description automatically generated

*Hình 1.14. Giao diện Quản lý khách thuê*

A screenshot of a computer

Description automatically generated

*Hình 1.15. Giao diện Thêm khách thuê*

### UC5: Use-case quản lý chỉ số tiền điện, nước

A diagram of a person

Description automatically generated with medium confidence

*Hình 1.16. Use case quản lý chỉ số tiền điện, nước*

|  |  |
| --- | --- |
| Use case | Quản lý chỉ số điện nước |
| Tác nhân | Admin |
| Mô tả ngắn gọn | Use case cho phép Admin tính tiền điện và nước cho các phòng trọ theo tháng. |
| Tiền điều kiện | Admin đăng nhập thành công vào hệ thống. |
| Luồng điều kiện | 1. Admin chọn “Chỉ số điện-nước” trên menu.  2. Hệ thống lấy thông tin điện-nước từ tblQlyPhongTro và hiển thị trên màn hình.  3. Admin tính tiền điện và nước cho các phòng trọ và thực hiện chức năng Reset nếu cần.  4. Hệ thống hiển thị kết quả tính toán và cập nhật dữ liệu.  5. Use case kết thúc. |
| Ngoại lệ | - Nếu mất kết nối CSDL, hệ thống hiển thị thông báo lỗi và Use case kết thúc. |
| Hậu điều kiện | Thông tin điện và nước của các phòng trọ được cập nhật và hiển thị trên màn hình. |

*Bảng 7. Use case quản lý chỉ số tiền điện, nước*

A screenshot of a computer

Description automatically generated

*Hình 1.17. Giao diện Quản lý chỉ số điện, nước*

### UC6: Use-case quản lý hóa đơn

A diagram of a diagram

Description automatically generated

*Hình 1.18. Use case quản lý hóa đơn*

|  |  |
| --- | --- |
| Use case | Hoá đơn |
| Tác nhân | Admin |
| Mô tả ngắn gọn | Use case cho phép Admin xem, thêm, sửa, xoá thông tin hoá đơn của khách thuê từ bảng tblHoaDon. |
| Tiền điều kiện | Admin đăng nhập thành công vào hệ thống. |
| Luồng điều kiện | 1. Admin chọn “Hoá đơn” trên menu.  2. Hệ thống lấy thông tin hoá đơn từ tblHoaDon và hiển thị trên màn hình.  3. Admin thực hiện các chức năng như tìm kiếm, tải lại, thêm mới, sửa hoặc xoá hoá đơn.  4. Use case kết thúc. |
| Ngoại lệ | - Nếu nhập thông tin hoá đơn không hợp lệ, hệ thống hiển thị thông báo lỗi.  - Nếu mất kết nối CSDL, hệ thống hiển thị thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Thông tin hoá đơn của khách thuê được cập nhật và hiển thị lên màn hình. |

*Bảng 8. Use case quản lý hóa đơn*

A screenshot of a computer

Description automatically generated

*Hình 1.19. Giao diện Quản ý hóa đơnA screenshot of a computer

Description automatically generated*

*Hình 1.20. Giao diện Thêm hóa đơn*

# Phân tích trường hợp sử dụng (Use-case analysis)

## Phân tích kiến trúc hệ thống

### Kiến trúc mức cao của hệ thống

Hệ thống quản lý nhà trọ cho thuê là một ứng dụng đa lớp, bao gồm các thành phần phục vụ cho mục tiêu quản lý phòng trọ. Hệ thống cho phép người quản trị theo dõi thông tin phòng trọ, quản lý khách hàng, quản lý thu chi...

**Actor (Tác nhân):**

* Quản trị viên (Admin): Người quản lý hệ thống, có quyền điều hành các hoạt động như thêm, sửa, xóa thông tin phòng trọ và quản lý khách hàng.

**Form (Giao diện người dùng):**

* Giao diện Người Dùng (UI): Đây là lớp giao diện giữa người dùng và hệ thống. Người dùng có thể tương tác với hệ thống thông qua các form và màn hình trong ứng dụng, được xây dựng bằng Java Swing.

**Control (Lớp điều khiển):**

* Lớp Điều Khiển (Controller): Thực hiện điều khiển chính các chức năng trong từng Use Case. Mỗi Use Case sẽ có một lớp điều khiển tương ứng để xử lý các yêu cầu từ người dùng, gửi yêu cầu tới API hoặc tương tác với các lớp Entity để lấy thông tin.

**API (Cổng API):**

* API Gateway: Đây là lớp trung gian để nhận và trả các yêu cầu từ giao diện người dùng (UI) đến các dịch vụ khác trong hệ thống. API Gateway xử lý các yêu cầu từ UI, chuyển đổi và giao tiếp với các dịch vụ khác.

**Entity (Thực thể):**

* Lớp Thực Thể (Entity): Lớp này chứa các thông tin chính của hệ thống và đại diện cho dữ liệu sẽ được lưu trữ lâu dài. Các thực thể chính bao gồm:
  + Thực thể Phòng Trọ (Room): Thông tin chi tiết về các phòng trọ, bao gồm tình trạng, giá thuê và các tiện ích.
  + Thực thể Khách Hàng (Customer): Thông tin cá nhân của khách hàng thuê phòng.
  + Thực thể Hóa Đơn (Invoice): Chi tiết về các hóa đơn thanh toán của khách hàng.
  + Thực thể Chỉ Số Điện Nước (UtilityMeter): Thông tin về chỉ số điện nước từng phòng.

**Database (Cơ sở dữ liệu):**

* Cơ sở dữ liệu (Database): Đây là nơi lưu trữ dữ liệu lâu dài cho hệ thống, bao gồm các bảng thông tin về phòng trọ, khách hàng, hóa đơn, và chỉ số điện nước.

**Sơ đồ kiến trúc mức cao của hệ thống:**

**A screenshot of a computer

Description automatically generated**

### Các đối tượng trừu tượng chính của hệ thống (Key abstractions)

- *Người quản trị (Admin):* Đối tượng chịu trách nhiệm quản lý toàn bộ hệ thống, bao gồm các chức năng đăng nhập, quản lý phòng trọ, khách hàng và hóa đơn.

*- Phòng trọ (Room):* Đối tượng đại diện cho mỗi phòng trọ được quản lý, chứa thông tin về loại phòng, tình trạng, giá thuê, và chỉ số điện nước.

- *Khách hàng (Customer):* Đối tượng lưu trữ thông tin người thuê phòng, bao gồm thông tin cá nhân, lịch sử thuê phòng và tình trạng thanh toán.

- *Hóa đơn (Invoice):* Đối tượng đại diện cho các hóa đơn thanh toán điện, nước và tiền thuê phòng, bao gồm các thông tin về số tiền, ngày thanh toán và trạng thái.

- *Chỉ số điện nước (UtilityReading):* Đối tượng lưu trữ chỉ số điện nước hàng tháng của từng phòng, hỗ trợ tính toán chi phí cho mỗi khách thuê.

## Thực thi trường hợp sử dụng (Use-case relizations)

### Các biểu đồ tuần tự (Sequence diagrams)

2.1.1.1. Đăng nhập

- Sơ đồ lớp phân tích:

A group of white dots on a black background

Description automatically generated

*Hình 2.1. Sơ đồ lớp phân tích đăng nhập*

- Sơ đồ trình tự:

A black and white screen with white dots

Description automatically generated

*Hình 2.2. Sơ đồ trình tự lớp đăng nhập*

2.1.1.2. Thống kê

- Sơ đồ lớp phân tích:

A group of white dots on a black background

Description automatically generated

*Hình 2.3. Sơ đồ lớp phân tích thống kê*

- Sơ đồ trình tự:

A black and white screen with white lines

Description automatically generated

*Hình 2.4. Sơ đồ trình tự lớp thống kê*

2.1.1.3. Quản lý phòng trọ

- Sơ đồ lớp phân tích:

A group of white dots on a black background

Description automatically generated

*Hình 2.5. Sơ đồ lớp phân tích quản llys phòng trọ*

- Sơ đồ trình tự:

A black background with white lines

Description automatically generated

*Hình 2.6. Sơ đồ trình tự lớp quản lý phòng trọ*

2.1.1.4. Quản lý khách thuê

- Sơ đồ lớp phân tích:

A group of white dots on a black background

Description automatically generated

*Hình 2.7. Sơ đồ lớp phân tích quản lý khách thuê*

- Sơ đồ trình tự:

A black background with white lines

Description automatically generated

*Hình 2.8. Sơ đồ trình tự lớp quản lý khách thuê*

2.1.1.5. Quản lý chỉ số điện, nước

- Sơ đồ lớp phân tích:

A white dots on a black background

Description automatically generated

*Hình 2.9. Sơ đồ lớp phân tích quản lý chỉ số điện, nước*

- Sơ đồ trình tự:

A black and white screen with white lines

Description automatically generated

*Hình 2.10. Sơ đồ trình tự lớp quản lý chỉ số điện, nước*

2.1.1.6. Quản lý hóa đơn

- Sơ đồ lớp phân tích:

A white dots on a black background

Description automatically generated

*Hình 2.11. Sơ đồ lớp phân tích quản lý hóa đơn*

- Sơ đồ trình tự:

A black background with white lines

Description automatically generated

*Hình 2.12. Sơ đồ trình tự lớp quản lý hóa đơn*

### Góc nhìn của các lớp trong hệ thống (Views of participating classes)

*Lớp Tài Khoản Quản Trị (AdminAccount):*

* Thuộc tính:
  + userName: Tên đăng nhập.
  + pass: Mật khẩu.
* Phương thức:
  + kiemTraTaiKhoan(): Kiểm tra thông tin đăng nhập (tên đăng nhập và mật khẩu). Phương thức này sẽ so sánh thông tin nhập vào với thông tin được lưu trữ trong cơ sở dữ liệu.

*Lớp Hóa Đơn (HoaDon):*

* Thuộc tính:
  + maHD: Mã hóa đơn (mã định danh duy nhất).
  + maKT: Mã khách thuê (tham chiếu đến bảng tblKhachThue).
  + maPhong: Mã phòng (tham chiếu đến bảng tblQuyPhongTro).
  + tgian: Thời gian lập hóa đơn.
* Phương thức:
  + getmaHD(): Lấy mã hóa đơn.
  + setmaHD(): Thiết lập mã hóa đơn.
  + getmaKT(): Lấy mã khách thuê.
  + setmaKT(): Thiết lập mã khách thuê.
  + getmaPhong(): Lấy mã phòng.
  + setmaPhong(): Thiết lập mã phòng.
  + gettgian(): Lấy thời gian lập hóa đơn.
  + settgian(): Thiết lập thời gian lập hóa đơn.
  + createTblHoaDon(): Tạo một bản ghi hóa đơn mới trong cơ sở dữ liệu.
  + deleteTblHoaDon\_byid(): Xóa một bản ghi hóa đơn khỏi cơ sở dữ liệu dựa trên ID.

*Lớp Khách Thuê (KhachThue):*

* Thuộc tính:
  + maKT: Mã khách thuê (mã định danh duy nhất).
  + hoten: Họ tên khách thuê.
  + ngaySinh: Ngày sinh.
  + ngheNghiep: Nghề nghiệp.
  + gioiTinh: Giới tính.
  + queQuan: Quê quán.
  + maPhong: Mã phòng (tham chiếu đến bảng tblQuyPhongTro).
* Phương thức:
  + getmaKT(): Lấy mã khách thuê.
  + setmaKT(): Thiết lập mã khách thuê.
  + gehoten(): Lấy họ tên khách thuê.
  + sethoten(): Thiết lập họ tên khách thuê.
  + getngaySinh(): Lấy ngày sinh khách thuê.
  + setngaySinh(): Thiết lập ngày sinh khách thuê.
  + getngheNghiep(): Lấy nghề nghiệp khách thuê.
  + setngheNghiep(): Thiết lập nghề nghiệp khách thuê.
  + getgioiTinh(): Lấy giới tính khách thuê.
  + setgioiTinh(): Thiết lập giới tính khách thuê.
  + getqueQuan(): Lấy quê quán khách thuê.
  + setqueQuan(): Thiết lập quê quán khách thuê.
  + getmaPhong(): Lấy mã phòng.
  + setmaPhong(): Thiết lập mã phòng.
  + createTblKhachThue(): Tạo một bản ghi khách thuê mới trong cơ sở dữ liệu.
  + deleteTblKhachThue\_byid(): Xóa một bản ghi khách thuê khỏi cơ sở dữ liệu dựa trên ID.
  + updateTblKhachThue(): Cập nhật thông tin khách thuê.

*Lớp Quản Lý Phòng Trọ (QuanLyPhongTro):*

* Thuộc tính:
  + maPhong: Mã phòng (mã định danh duy nhất).
  + soNguoi: Số lượng người tối đa trong phòng.
  + giaThue: Giá thuê phòng.
  + dienTich: Diện tích phòng.
  + doTuongThue: Đối tượng thuê (ví dụ: cá nhân, gia đình).
  + soDien: Chỉ số điện hiện tại.
  + soNuoc: Chỉ số nước hiện tại.
  + soDienMoi: Chỉ số điện mới.
  + soNuocMoi: Chỉ số nước mới.
* Phương thức:
  + getmaPhong(): Lấy mã phòng.
  + setmaPhong(): Thiết lập mã phòng.
  + getsoNguoi(): Lấy số lượng người tối đa.
  + setsoNguoi(): Thiết lập số lượng người tối đa.
  + getgiaThue(): Lấy giá thuê phòng.
  + setgiaThue(): Thiết lập giá thuê phòng.
  + getdienTich(): Lấy diện tích phòng.
  + setdienTich(): Thiết lập diện tích phòng.
  + getdoTuongThue(): Lấy đối tượng thuê.
  + setdoTuongThue(): Thiết lập đối tượng thuê.
  + getsoDien(): Lấy chỉ số điện hiện tại.
  + setsoDien(): Thiết lập chỉ số điện hiện tại.
  + getsoNuoc(): Lấy chỉ số nước hiện tại.
  + setsoNuoc(): Thiết lập chỉ số nước hiện tại.
  + getsoDienMoi(): Lấy chỉ số điện mới.
  + setsoDienMoi(): Thiết lập chỉ số điện mới.
  + getsoNuocMoi(): Lấy chỉ số nước mới.
  + setsoNuocMoi(): Thiết lập chỉ số nước mới.
  + createTblQuyPhongTro(): Tạo một bản ghi phòng trọ mới trong cơ sở dữ liệu.
  + deleteTblQuyPhongTro\_byid(): Xóa một bản ghi phòng trọ khỏi cơ sở dữ liệu dựa trên ID.
  + updateTblQuyPhongTro(): Cập nhật thông tin phòng trọ.

*Mối quan hệ giữa các lớp:*

* Khách Thuê ↔ Hóa Đơn: Một khách thuê có thể có nhiều hóa đơn, nhưng mỗi hóa đơn chỉ thuộc về một khách thuê.
* Phòng Trọ ↔ Khách Thuê: Một phòng trọ chỉ có thể được thuê bởi một khách thuê tại một thời điểm (giả sử không cho phép chia sẻ phòng). Tuy nhiên, một khách thuê có thể thuê nhiều phòng trọ khác nhau trong các thời điểm khác nhau.
* Phòng Trọ ↔ Hóa Đơn: Một phòng trọ có thể có nhiều hóa đơn liên quan đến các lần thuê khác nhau, và mỗi hóa đơn chỉ liên quan đến một phòng trọ.
* Admin ↔ Khách Thuê/Phòng Trọ/Hóa Đơn: Admin quản lý tất cả thông tin về khách thuê, phòng trọ và hóa đơn. (Admin có quyền truy cập và quản lý tất cả các đối tượng khác)
* ThongKe ↔ Khách Thuê/Phòng Trọ/Hóa Đơn: Lớp Thống kê truy xuất dữ liệu từ Khách Thuê, Phòng Trọ và Hóa Đơn để tạo ra các báo cáo thống kê.

# Thiết kế (Use-case design)

## Xác định các thành phần thiết kế (Identify design elements)

### Xác định các lớp (Identify classes)

A black and white image of a black background

Description automatically generated

*Hình 3.1. Biểu đồ lớp đăng nhập*

A screenshot of a computer screen

Description automatically generated

*Hình 3.2. Biểu đồ lớp thống kê*

A screenshot of a computer

Description automatically generated

*Hình 3.3. Biểu đồ lớp quản lý nhà trọ*

A screenshot of a computer

Description automatically generated

*Hình 3.4. Biểu đồ lớp quản lý khách thuê*

A screenshot of a computer

Description automatically generated

*Hình 3.5. Biểu đồ lớp quản lý chỉ số điện, nước*

A black and white text

Description automatically generated

*Hình 3.6. Biểu đồ lớp quản lý hóa đơn*

### Xác định các hệ thống con và giao diện (Identify subsystems and interfaces)

* Giao diện người dùng (UI):
  + Trang chủ
  + Chi tiết phòng trọ
  + Quản lý người thuê
  + Quản lý hóa đơn
  + Theo dõi thông tin thuê phòng
* Máy chủ xử lý (Backend):
  + Xử lý yêu cầu
  + Cơ sở dữ liệu (quản lý dữ liệu phòng trọ, người thuê)
  + Xử lý thanh toán (quản lý các giao dịch thanh toán)

A screenshot of a computer screen

Description automatically generated

*Hình 3.7. Hệ thống con*

### Xác định các gói (Identify packages)

A screenshot of a computer

Description automatically generated

*Hình 3.8. Gói quản lý thông tin Đăng nhập*

A black and white image of a diagram

Description automatically generated

*Hình 3.9. Gói quản lý thông tin Thống kê*

A black and white image of a diagram

Description automatically generated

*Hình 3.10. Gói quản lý thông tin Quản lý Phòng trọ*

A black and white screen shot

Description automatically generated*Hình 3.11. Gói quản lý thông tin Quản lý khách thuê*

A black and white rectangular object with black text

Description automatically generated

*Hình 3.12. Gói quản lý thông tin Quản lý chỉ số điện, nước*

A white rectangular object with black text

Description automatically generated

*Hình 3.13. Gói quản lý thông tin Quản lý hóa đơn*

## Thiết kế trường hợp sử dụng (Use-case design)

### Thiết kế các biểu đồ tuần tự (Design sequence diagrams)

A black screen with white dots and lines

Description automatically generated

*Tuần tự đăng nhập*

A black and white screen with white lines

Description automatically generated

*Tuần tự thống kê*

A black background with white lines

Description automatically generated

*Tuần tự Quản lý phòng trọ*

A black and white screen with white lines

Description automatically generated

*Tuần tự Quản lý khách thuê*

A black and white screen with white lines

Description automatically generated

*Tuần tự Quản lý chỉ số điện, nước*

A black background with white lines

Description automatically generated

*Tuần tự Quản lý hóa đơn*

### Thiết kế biểu đồ lớp (Class diagrams)

A group of white dots on a black background

Description automatically generated

*Biểu đồ lớp đăng nhập*

A group of white dots on a black background

Description automatically generated

*Biểu đồ lớp Thống kê*

A group of white dots on a black background

Description automatically generated

*Biểu đồ lớp Quản lý phòng trọ*

A group of white dots on a black background

Description automatically generated

*Biểu đồ lớp Quản lý khách thuê*

A group of white dots on a black background

Description automatically generated

*Biểu đồ lớp Quản lý chỉ số điện, nước*

A group of white dots on a black background

Description automatically generated

*Biểu đồ lớp Quản lý hóa đơn*

## Thiết kế cơ sở dữ liệu (Database design)

### Lược đồ cơ sở dữ liệu

A screenshot of a computer

Description automatically generated

### Chi tiết các bảng

Bảng *acc\_admin*: Bảng này lưu trữ thông tin tài khoản quản trị viên.

* Bao gồm hai cột:
  + UserName: Tên tài khoản của quản trị viên, kiểu dữ liệu là VARCHAR.
  + Pass: Mật khẩu của tài khoản, kiểu dữ liệu là VARCHAR.

Bảng *tblKhachThue*: Bảng này lưu trữ thông tin về khách thuê phòng.

* Các cột trong bảng bao gồm:
  + maKT: Mã định danh của khách thuê, là khóa chính, kiểu dữ liệu INT.
  + HoTen: Họ và tên của khách thuê, kiểu dữ liệu VARCHAR.
  + NgaySinh: Ngày sinh của khách thuê, kiểu dữ liệu DATE.
  + NgheNghiep: Nghề nghiệp của khách thuê, kiểu dữ liệu VARCHAR.
  + GioiTinh: Giới tính của khách thuê, kiểu dữ liệu VARCHAR.
  + Sdt: Số điện thoại liên hệ của khách thuê, kiểu dữ liệu VARCHAR.
  + QueQuan: Quê quán của khách thuê, kiểu dữ liệu VARCHAR.
  + maPhong: Mã phòng mà khách thuê đang thuê, là khóa ngoại liên kết với bảng tblQlPhongTro, kiểu dữ liệu INT.

Bảng *tblQlPhongTro*: Bảng này lưu trữ thông tin về các phòng trọ.

* Các cột trong bảng bao gồm:
  + maPhong: Mã phòng trọ, là khóa chính, kiểu dữ liệu INT.
  + dienTich: Diện tích của phòng trọ, kiểu dữ liệu FLOAT.
  + soNguoi: Số người tối đa có thể ở trong phòng, kiểu dữ liệu INT.
  + giaThue: Giá thuê phòng, kiểu dữ liệu DECIMAL.
  + DoiTuongThue: Đối tượng cho thuê (ví dụ: sinh viên, gia đình), kiểu dữ liệu VARCHAR.
  + TinhTrang: Tình trạng của phòng (trống, đã thuê, bảo trì, ...), kiểu dữ liệu VARCHAR.
  + chiSoDienMoi: Chỉ số điện mới, kiểu dữ liệu INT.
  + chiSoDienCu: Chỉ số điện cũ, kiểu dữ liệu INT.
  + chiSoNuocMoi: Chỉ số nước mới, kiểu dữ liệu INT.
  + chiSoNuocCu: Chỉ số nước cũ, kiểu dữ liệu INT.

Bảng *tblHoaDon*: Bảng này lưu trữ thông tin về hóa đơn thanh toán.

* Các cột trong bảng bao gồm:
  + maHD: Mã hóa đơn, là khóa chính, kiểu dữ liệu INT.
  + maKT: Mã khách thuê, là khóa ngoại liên kết với bảng tblKhachThue, kiểu dữ liệu INT.
  + maPhong: Mã phòng, là khóa ngoại liên kết với bảng tblQlPhongTro, kiểu dữ liệu INT.
  + tgian: Thời gian lập hóa đơn, kiểu dữ liệu DATE.

Mối quan hệ giữa các bảng:

* Bảng tblKhachThue có mối quan hệ với tblQlPhongTro qua cột maPhong, giúp xác định phòng mà khách thuê đang ở.
* Bảng tblHoaDon có mối quan hệ với tblKhachThue qua cột maKT và với tblQlPhongTro qua cột maPhong, giúp quản lý hóa đơn cho từng khách thuê và phòng tương ứng.

# Cài đặt

## Lựa chọn công nghệ

### Java

Java là ngôn ngữ lập trình chính của hệ thống, được sử dụng do khả năng đa nền tảng, bảo mật cao, và sự hỗ trợ mạnh mẽ cho lập trình hướng đối tượng. Java đảm bảo tính ổn định của hệ thống và phù hợp với yêu cầu phát triển ứng dụng quản lý nhà trọ.

### Java Swing

Java Swing được lựa chọn để xây dựng giao diện người dùng (UI). Đây là một thư viện GUI mạnh mẽ cho Java, hỗ trợ phát triển các form và màn hình tương tác, giúp người dùng dễ dàng quản lý thông tin về phòng trọ, khách hàng, hóa đơn và chỉ số điện nước. Java Swing cung cấp các thành phần giao diện như button, textbox, bảng dữ liệu và các hộp thoại.

### MySQL - Xampp

Hệ quản trị cơ sở dữ liệu MySQL được chọn để lưu trữ và quản lý dữ liệu cho hệ thống. MySQL cung cấp khả năng truy vấn SQL linh hoạt và hiệu quả, hỗ trợ quản lý thông tin về khách thuê, hóa đơn, phòng trọ và các chỉ số điện nước một cách an toàn. Hệ thống sử dụng MySQL giúp dễ dàng truy xuất và cập nhật dữ liệu, đảm bảo tính toàn vẹn và bảo mật thông tin trong quá trình vận hành.

### NetBeans

NetBeans là môi trường phát triển tích hợp (IDE) được sử dụng trong dự án. Nó hỗ trợ đầy đủ các công cụ như biên dịch, gỡ lỗi và quản lý thư viện, giúp cho quá trình phát triển và kiểm thử ứng dụng Java trở nên nhanh chóng và hiệu quả.

### XAMPP

XAMPP là một phần mềm cung cấp môi trường máy chủ cục bộ bao gồm Apache và MySQL. Nó cho phép thử nghiệm và kiểm tra ứng dụng trên máy tính cục bộ trước khi triển khai thực tế, đảm bảo tính ổn định và khả năng hoạt động của hệ thống.

### GitHub

GitHub là nền tảng quản lý mã nguồn chính của dự án. Nó cung cấp các công cụ quản lý phiên bản, cộng tác nhóm, và theo dõi thay đổi mã nguồn, giúp nhóm phát triển duy trì sự đồng bộ, dễ dàng cập nhật và giải quyết các vấn đề phát sinh trong quá trình phát triển hệ thống.

## Cấu trúc mã nguồn

*Thư mục mã nguồn:*

|  |  |
| --- | --- |
| A screenshot of a computer  Description automatically generatedA screenshot of a computer program  Description automatically generatedA screenshot of a computer  Description automatically generated | **- build/classes**: Chứa các tệp đã được biên dịch.   * Các thư mục Controller, Model, và View cho thấy dự án này tuân theo mô hình MVC, trong đó:   + Controller chứa logic để quản lý các tương tác của người dùng.   + Model bao gồm các mô hình dữ liệu (đại diện cho các thực thể như HoaDon, KhachThue, v.v.).   + View chứa các biểu mẫu và thành phần giao diện người dùng.   **- database/qll.sql**: Tệp này có thể chứa các tập lệnh SQL để thiết lập hoặc tương tác với cơ sở dữ liệu, như tạo bảng hoặc chèn dữ liệu ban đầu.  **- nbproject**: Chứa các tệp cấu hình của dự án NetBeans.   * private/config.properties: Lưu trữ các cấu hình riêng của dự án. * Các tệp .properties và .xml khác quản lý các thiết lập và cấu hình biên dịch của dự án.   **- src**:   * **Controller**: Chứa các lớp DAO như Connections.java, HoaDonDao.java, giúp quản lý tương tác giữa ứng dụng và cơ sở dữ liệu. * **lib**: Lưu trữ các thư viện bên ngoài (.jar) cần thiết cho dự án, bao gồm thư viện SQL và lịch. * **Model**: Định nghĩa các lớp dữ liệu (HoaDon.java, KhachThue.java, v.v.) đại diện cho các thực thể khác nhau. * **View**: Chứa các tệp .form và .java cho giao diện đồ họa người dùng (ví dụ: Login.form, MainForm.java), hỗ trợ các tương tác như đăng nhập và quản lý phòng.   **- Các tệp khác**:   * build.xml: Tệp Ant để tự động hóa việc biên dịch và triển khai. * README.md: Có thể cung cấp tổng quan về dự án, hướng dẫn thiết lập hoặc hướng dẫn sử dụng. |

Link github: <https://github.com/Ngolinh141/Phantich.git>

# Kết luận

Dự án quản lý nhà trọ đã được xây dựng với mục tiêu cung cấp một hệ thống quản lý trực quan, hiệu quả, đáp ứng nhu cầu của người cho thuê trọ trong bối cảnh đô thị hiện đại. Qua quá trình phân tích, thiết kế, và lựa chọn công nghệ phù hợp như Java, Java Swing, MySQL, NetBeans, XAMPP, và GitHub, hệ thống không chỉ đảm bảo được tính ổn định và bảo mật, mà còn mang lại trải nghiệm người dùng tốt với giao diện thân thiện và dễ sử dụng.

Việc sử dụng MySQL làm cơ sở dữ liệu đã giúp quản lý dữ liệu của hệ thống một cách khoa học và bảo mật, đảm bảo tính nhất quán và dễ dàng truy xuất thông tin. Java Swing hỗ trợ tạo ra các form giao diện phong phú, trực quan, giúp người dùng tương tác dễ dàng với hệ thống. GitHub đóng vai trò quan trọng trong việc quản lý phiên bản mã nguồn, tăng cường khả năng cộng tác và đảm bảo tiến độ dự án.

Kết luận, hệ thống quản lý nhà trọ đã đạt được mục tiêu đặt ra và là nền tảng để mở rộng tính năng trong tương lai. Dự án này có tiềm năng phát triển thành một giải pháp quản lý nhà trọ hoàn thiện hơn, đáp ứng tốt hơn nhu cầu của người dùng trong một thị trường phát triển không ngừng.

**Định hướng tương lai:**

* Thêm chức năng: Quản lý thiết bị trong phòng trọ, Quản lý Thu chi các loại phụ phí khác như: tiền gửi xe, …

# Tài liệu tham khảo

1. K. CNTT, Slide bài giảng môn học Phân tích thiết kế phần mềm, 2024.
2. Trên JavatPoint: <https://www.javatpoint.com/>
3. Website W3School: <https://www.w3schools.com/java/default.asp>
4. StackOverflow: <https://stackoverflow.com/>
5. Roger S PRESSMAN, Bruce R. MAXIM, Software engineering a practitioner’s approach, New York, 2024.
6. Roger S. FRESSMAN, Bruce R. MAXIM, Software engineering a practitioner’s approach, New York: McGraw-Hill, 2015.
7. Kathy Sierra & Bert Bates, Headfirst Design Patterns -OReilly, New York: Sumary, 2008.
8. Alan Dennis, Systems Analysis and Design\_ An Object-Oriented Approach with UML, Hoboken: ShutterStock, 2015.