**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA: CÔNG NGHỆ THÔNG TIN**

-----2022-----



**BÁO CÁO**

**Công nghệ phần mềm**

**LỚP : 19\_22**

**Giáo viên lý thuyết: Th.S. Bùi Tấn Lộc**

**SINH VIÊN THỰC HIỆN:**

**19120544 - Cao Thanh Khiết**

**19120533 - Ninh Duy Huy**

**19120557 - Trần Tuấn Kiệt**

**19120558 - Văn Quý Lâm**

**19120559 - Hà Duy Lãm**

# Chương I/ Requirement Gathering

## 1.1/ Client Problem Statement

### 1.1.1 Context

Nhu cầu nghỉ chân, tá túc của con người là cần thiết và nhà trọ đã có từ ra đời rất lâu rồi, để có chỗ nghỉ ngơi khi đi xa hay để ở tạm thời một nơi nào đó. Đến bây giờ nó càng ngày càng phát triển và nhu cầu đối với xã hội cũng càng tăng nhiều hơn với tên gọi là khách sạn.

Do nhu cầu đó nên những người chủ của khách sạn mong muốn có sự quản lý khách sạn thật tốt để phục vụ việc kinh doanh, nhưng con người thì rất khó để làm việc đó thật hiệu quả và chính xác, thậm chí còn phải tiêu một khoản tiền nhân sự khá lớn. Do vậy nên những ứng dụng quản lý khách sạn ra đời.

#### *1.1.1.1 Product Market*

Các phần mềm quản lý khách sạn trên thị trường:

- Sophia SMS: là sản phẩm thuộc công ty Sophia Software Solutions. Giao diện dễ học, dễ sử dụng; nghiệp vụ lễ tân chuẩn mực, triệt tiêu hoàn toàn các lỗi thông thường của người dùng; cách tính tiền linh hoạt cho nhiều mô hình kinh doanh; …

- Smile PMS: là sản phẩm thuộc công ty Minh Anh. Phần mềm đầy đủ tính năng, thể hiện đầy đủ về tình trạng phòng, hệ thống báo cáo chi tiết, …

- SkyHotel: thuộc công ty SkyViet. Thiết kế dễ nhìn, thuận tiện cho người sử dụng, dễ dàng thiết lập giá, phân quyền chi tiết, có chức năng giao ca.

#### *1.1.1.1 Comparisons of other Products*

| Bảng so sánh | Sophia SMS | Smile PMS | SkyHotel |
| --- | --- | --- | --- |
| Chức năng nổi trội | -Nghiệp vụ lễ tân    - Tính tiền linh hoạt | - Báo cáo chi tiết    - Thể hiện tình trạng phòng ở | - Có chức năng giao ca  - Phân quyền chi tiết |
| Hạn chế | - Chưa quá chi tiết | - Chưa có tính năng giao ca | - Chưa quá nghiệp vụ lễ tân |
| Thị phần | 12.5% | 15.6% | 13.4% |

### 1.1.1 Problem Statement (Business Vision)

Có thể thấy từ những nhu cầu và xu thế hay thậm chí là hạn chế còn tồn đọng tại các sản phẩm cùng loại trên chúng ta có thể định hình phần mềm cần có các chức năng như giao diện thân thiện với người dùng, có tính năng chia ca và quản lý lương, hệ thống báo cáo chi tiết, … Ngoài ra có thể thêm vào tính năng báo cáo lỗi dành cho người sử dụng để nhà phát triển có thể sửa lỗi nhanh chóng.

Một phần mềm vừa đầy đủ tính năng, dễ sử dụng và có khả năng sửa lỗi nếu có bất cứ lúc nào chắc chắn sẽ rất hút khách và có thể mở rộng ra trên phạm vi quốc tế.

### 1.1.2 User Statement of Requirements (Business Mission)

Một phần mềm muốn vừa hoàn thiện đầy đủ chức năng, vừa chi tiết mà an toàn dễ sử dụng thường sẽ có giá không hề rẻ, đó là rào cản lớn nhất để nhóm phát triển tiếp cận được các khách hàng có vốn đầu tư không cao. Vậy nên nhóm sẽ hướng tới một phần mềm có đầy đủ chức năng nhưng sẽ bị khóa vài chức năng cao cấp hơn và chỉ mở khi chi thêm tiền mua bản nâng cấp.

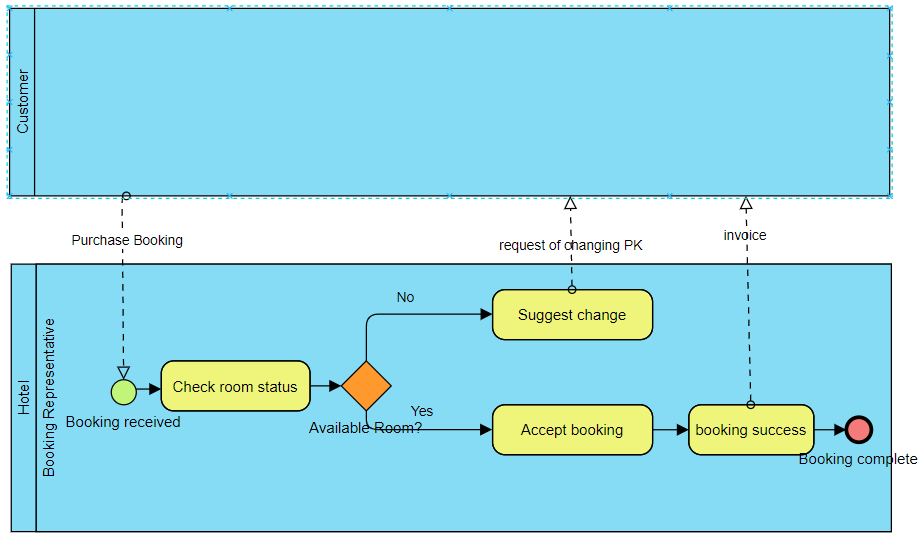
Phương án này vừa dành cho những khách hàng có kinh tế thấp nhưng muốn sử dụng sản phẩm tốt, vừa dành cho những doanh nghiệp có nguồn tài chính cao để mở rộng thêm những tính năng cao cấp hơn.

Hiện tại thì nhóm đang trong giai đoạn phát triển phần mềm và thêm các tính năng cũng như sửa chữa lỗi để sản phẩm thật hoàn thiện.

## 1.2 Business Process Modeling

### 1.2.1 Purchase Order Process

#### 1.2.1.1 As-is Process for Purchase Order Process



Hình 1.1: As-is Process for Purchase Order Process

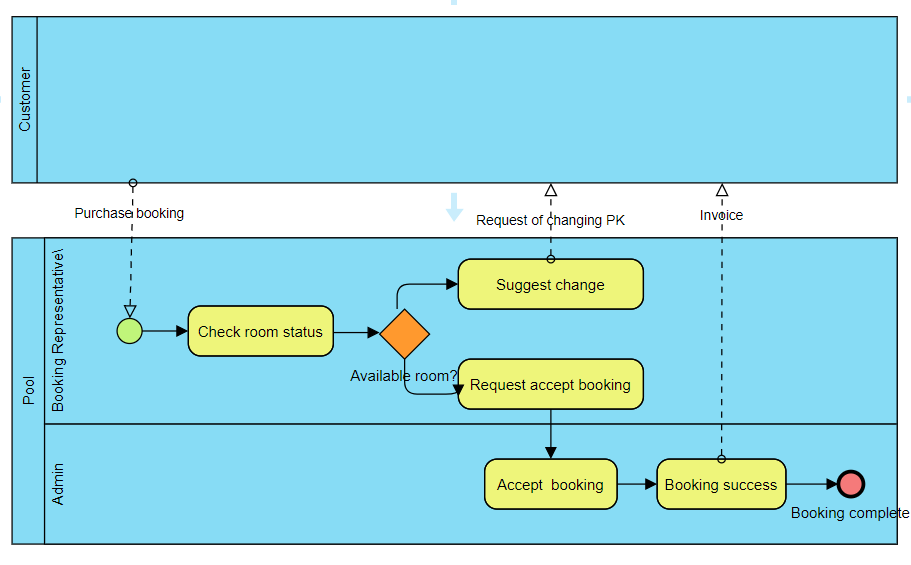
- **Check Room Status** : Khách sạn nhận được thông báo order phòng của khách hàng và đi kiểm tra status (phòng còn trống hay đã được đặt) của phòng.

- **Suggest change**: sau bước “**Check Room Status**” nếu kết quả kiểm tra status của phòng là đã được đặt thì sẽ gửi lại thông báo cho khách hàng là phòng đã có người đặt và để xuất phòng khách cho khách hàng.

- **Accept booking**: sau bước “**Check Room Status**” nếu kết quả kiểm tra status của phòng là trống thì chấp nhận order của khách hàng.

- **Booking success**: Xuất thông báo hóa đơn cho khách hàng, thay đổi status của phòng thành đã có người đặt và kết thúc order.

#### 1.2.1.2 To-be Process for Purchase Order Process



Hình 1.2: To-be Process for Purchase Order Process

* **Request Pack Booking**: khác biệt so với “As-is Process for Purchase Order Process” chia online shop thành “Booking Representative” và “Admin” , Request Pack booking đóng vai trò truyền request (chấp nhận order phòng) từ “Booking Representative” tới “Admin”.

· Các Loại tác vụ:

o Manual task:

§ Order Received

o User task:

§ Purchase order

o Send task: Purchase

§ Purchase order

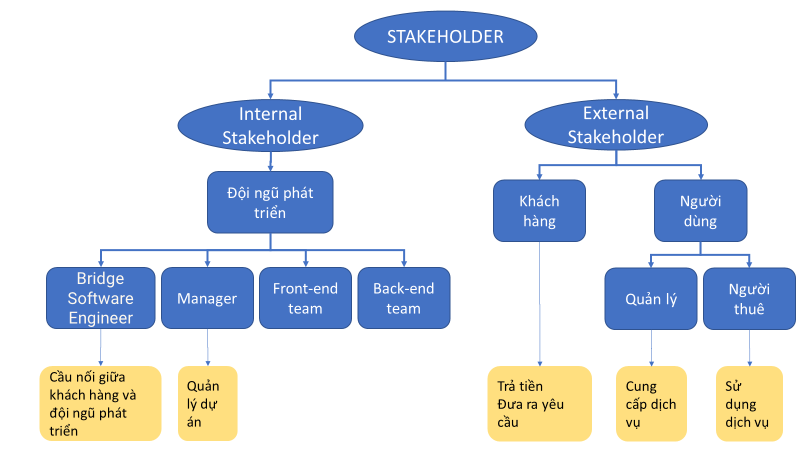
§ Request of Changing PO

o Service task:

§ Order Received

## 1.3/ System requirement

### 1.3.1/ Stakeholders



**Hình 1.3.1.1:** Mô hình cây phân cấp cho Stakeholder

- Manager Cao Thanh Khiết.

- Bridge Software Engineer Hà Duy Lãm.

### 1.3.2/ Initial User Roles

Dùng để tham khảo vẽ actor sơ đồ use case

| **Role** | **Definition** | **Goal** |
| --- | --- | --- |
| Guest | Là khách mà không đăng ký tài khoản, có thể tham khảo tổng quan web đặt phòng khách sạn, có thể tra cứu thông tin trước khi đặt phòng (loại phòng, giá phòng,..) và không được đăng kí phòng. | Có thể tra cứu được nhiều thông tin nhất (loại phòng, giá tiền, cơ sở vật chất,...), cảm thấy hài lòng nhất về dịch vụ đặt phòng khách sạn. |
| Customer | Là khách đã đăng ký tài khoản và có thể tiến hành đăng ký phòng, thông tin sẽ được lưu lại, có thể xem được các thông tin đặt phòng như Guest | Có thể tra cứu được nhiều thông tin về đặt phòng nhất, có thể xem thông tin những phòng đã đặt, có thể thêm nhận xét, đánh giá sau khi đã đặt phòng. |
| Admin | Là người chịu trách nhiệm quản lý khách sạn, tương tác với khách hàng, có thể thay đổi thông tin phòng, thông tin đặt phòng của khách, | Có thể thực hiện các tác vụ thay đổi dữ liệu về thông tin phòng và thông tin của khách hàng, lập các báo cáo thông qua thông tin khách hàng đã đặt, phục vụ tốt nhất cho khách hàng. |

### 1.3.3/ Functional requirements

#### 1.3.3.1/ Task Requirements

| **ID** | **Requirement** | **Role** | **Priority** | **Description** |
| --- | --- | --- | --- | --- |
|  | Login | Guest | Must | Đăng nhập vào tài khoản khách hàng đã đăng ký trước đó. |
|  | Registration | Guest, Customer | Must | Đăng ký thành viên của khách sạn để có thể đặt phòng, lưu thông tin của khách trong những lần đặt sau. |
|  | Reservation | Customer | Should | Cho phép khách hàng giữ chỗ đặt phòng online. |
|  | Booking | Customer | Must | Cho phép khách hàng đặt phòng và thanh toán online. |
|  | Change information | Customer, Admin | Must | Thay đổi thông tin đặt phòng của khách hàng (loại phòng, thông tin khách hàng,...) |
|  | Room filter | Guest, Customer | Must | Tìm kiếm phòng theo nhu cầu khách hàng. |
|  | Rating | Customer | Should | Đánh giá phòng, viết nhận xét sau khi khách đã đặt phòng. |
|  | Communication | Customer, Admin | Should | Giao tiếp giữa khách hàng và quản lý để giải quyết các vấn đề. |
|  | Change profile | Customer, Admin | Should | Thay đổi thông tin về tài khoản của khách hàng (tên, mật khẩu, ngày sinh,...) |
|  | Profile | Customer, Admin | Should | Xem thông tin về tài khoản người dùng đã đăng ký. |

#### 

#### 1.3.3.2/ User requirements in CRUD & Transform data

| **ID** | **Name** | **Priority** | **Role** | **Description** |
| --- | --- | --- | --- | --- |
|  | Login | Must | Guest | Query an item in database |
|  | Registration | Must | Guest, | Create a new item |
|  | Reservation | Must | Customer | Create a new item |
|  | Booking | Must | Customer | Create a new item |
|  | Change information | Must | Customer | Update an existing item |
|  | Room filter | Must | Guest, Customer | Search item, Query list of items in database |
|  | Rating | Should | Customer | Create a new item |
|  | Change profile | Should | Customer | Update an existing item |
|  | Profile | Should | Customer | View item details |
|  |  |  |  |  |

### 1.3.4/ Nonfunctional Requirements

#### 1.3.4.1/ Operational Nonfunctional Requirements

**- Budget:** Không có

**- Usability:** UI đơn giản, kèm theo hướng dẫn sử dụng ngay trên màn hình cho người thuê phòng. Đối với quản lý sẽ có hướng dẫn cụ thể bằng văn bản cách sử dụng từng chức năng.

**- Efficiency:** Đảm bảo ít màn hình nhất có thể, nhưng vẫn giữ đủ các chức năng. Từ đó người dùng tiết kiệm thời gian thao tác.

**- Correctness:** quá trình testing diễn ra song song với việc xây dựng phần mềm, đảm bảo từng chức năng hoạt động chính xác. Một thành viên sẽ có nhiệm vụ kiểm tra phần code của thành viên khác.

**- Dependability/Reliability:** kết nối thông qua Internet, phải đảm bảo kết nối mạng tối thiểu 6 Mbps. Không phụ thuộc nhiều vào cấu hình của thiết bị.

**- Security:**

+ Mật khẩu được đưa vào hàm hash một chiều, khi người dùng nhập mật khẩu, mật khẩu này sẽ được hash và so sánh trong database (ngay cả admin cũng không thể xem được mật khẩu chưa mã hóa trong database).

+ Phòng chống đoán mò mật khẩu: chỉ hiển thị “Đăng nhập không thành công” thay vì thông báo chi tiết username hoặc password bị sai. Nếu nhập sai mật khẩu quá 3 lần thì phải đợi 5 phút sau để tiếp tục.

+ Sử dụng các phiên bản framework mới nhất.

+ Phòng SQL Injection: sử dụng ORM (Object-Relational Mapping) framework. Hạn chế viết SQL thuần.

+ Giấu thông tin hệ thống: Không để thông tin về loại framework, phiên bản công khai.

+ Phòng tránh XSS (Cross Site Scripting): Sử dụng hàm encode có sẵn trong framework để chuyển các kí tự đặc biệt như “<”, “>” thành “&lt;”, “%gt;”.

#### 1.3.4.2/ Transitional Nonfunctional Requirements

**- Portability:** Tương thích tốt nhất qua Browser nhân Chronium, có thể sử dụng trên bất cứ thiết bị máy tính, điện thoại có browser (MacOS, Windows, iOS, Android, Linux). Nên update browser lên phiên bản mới nhất để có trải nghiệm tốt nhất.

**- Reusability:** Có thể sử dụng module cho các mục đích khác. (Ví dụ màn hình đăng nhập có thể sử dụng cho nhiều phần mềm yêu cầu xác minh tài khoản khác)

#### 1.3.4.3/ Maintenance Nonfunctional Requirements

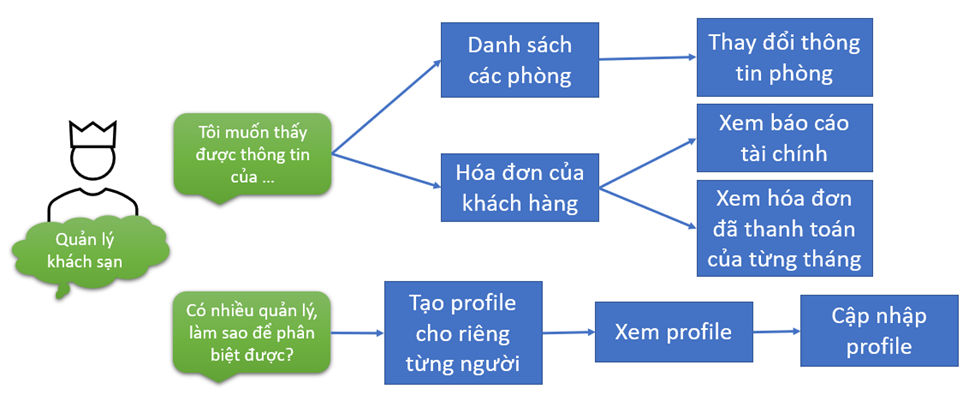
**- Modularity:** Đảm bảo chia ra thành các module.

**- Maintainability:** Trong quá trình xây dựng, ghi chú thích vào source code.

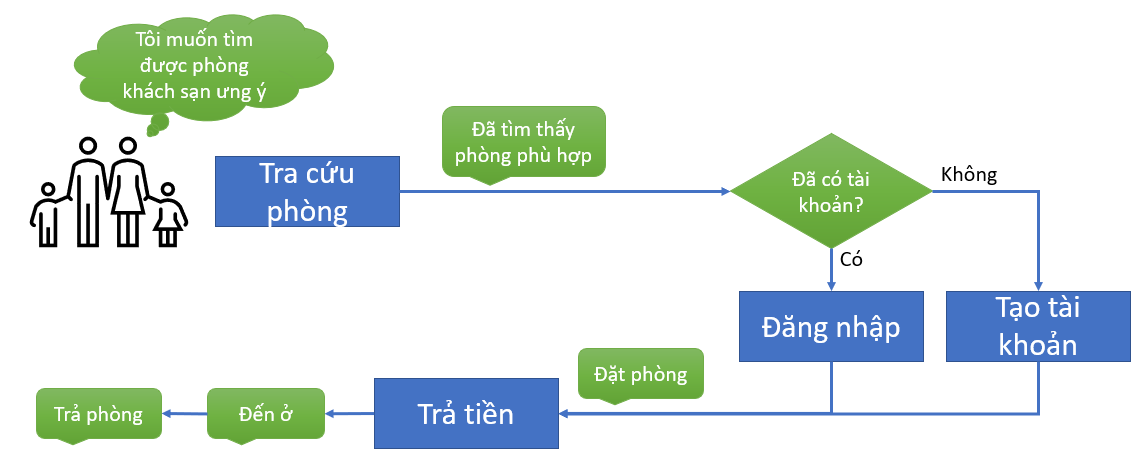
**- Scalability:** tối đa 1000 lượt truy cập qua website.

## 1.4/ MVP Planning

### 1.4.1 Tell a Story

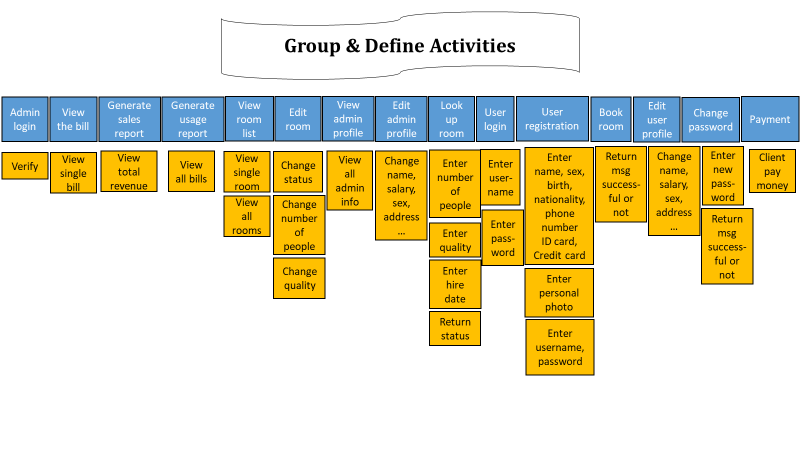


**Hình 1.4.1.1: Câu chuyện của quản lý khách sạn**



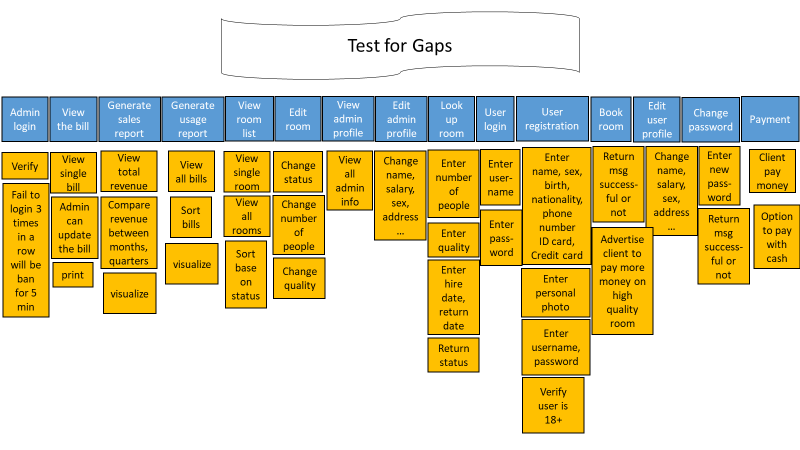
**Hình 1.4.1.2: Câu chuyện của khách muốn đặt phòng**

### 1.4.2 Group & Define Activities



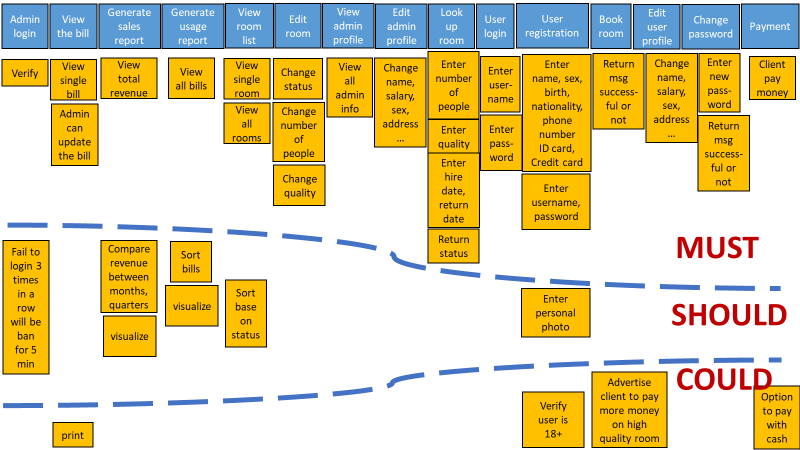
**Hình 1.4.2.1: Group & Define Activities**

### **1.4.3 Test for Gaps**

****

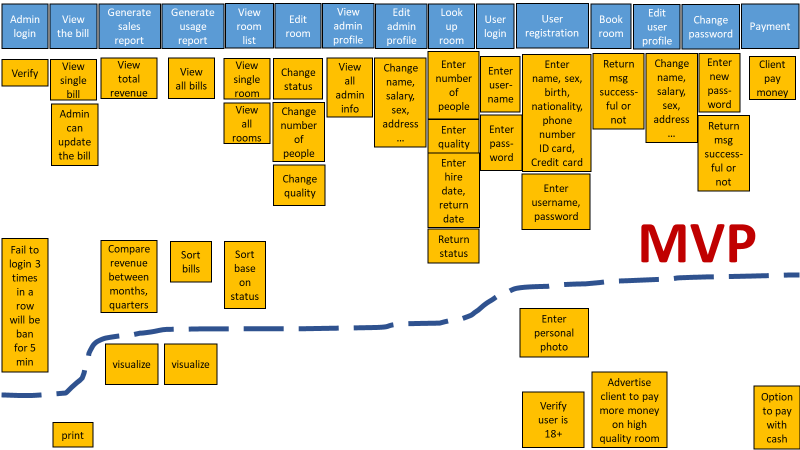
**Hình 1.4.3.1: Test for Gaps**

### 1.4.4 Prioritize



**Hình 1.4.4.1: Prioritize**

### 1.4.5 Define Iterations



**Hình 1.4.5.1: MVP - Define Iterations**

**- Release MVP:** Hoàn thành tối thiểu các chức năng cần thiết.

**- Bản alpha:** Các thành viên nội bộ (5 người) sẽ test các chức năng và report lại

**- Bản beta:** Sửa lại các lỗi trước đó, release với quy mô lớn hơn (15 người hoặc ít hơn). Không tiết lộ cho đối thủ cạnh tranh (các nhóm khác).

- **Bản 1.0.0:** Yêu cầu MVP phải hoàn chỉnh, sửa toàn bộ lỗi đã report.

**- Bản 1.0.1:** Release tính năng tính toán, thống kê liên quan đến hóa đơn.

**- Bản 1.0.2:** Test lưu lượng truy cập ở mức tối đa là bao nhiêu.

**- Bản 1.0.3 cho đến 1.0.9:** Release các tính năng kèm theo đồng thời sửa các lỗi được report trước đó.

**- Bản 1.1.0:** Official release.

## 1.5/ Risk Management

### 1.5.1 Đánh giá sơ bộ

Nhìn qua sơ bộ về phần mềm quản lý khách sạn thì nó là loại dự án mà chúng ta chưa có kinh nghiệm về kinh phí và thời gian hoàn thành

Về phần công nghệ sử dụng thì trong dự án có sử dụng: JavaScript, Reactjs, Nodejs,…

Đánh giá về thành viên nhóm:

- Tất cả thành viên đều thành thạo ngôn ngữ Javascript và những nhánh của nó.

- Đảm bảo về mặt chất lượng.

- Chưa làm việc trơn tru với nhau.

- Chưa thống nhất về mặt thời gian hoàn thành.

### 1.5.2 Nhóm các rủi ro có thể giải quyết được

Vấn đề về kiến thức và công nghệ có thể giải quyết bằng cách học một khóa training cho người mới.

Do thiếu kiến thức thì rất khó để hoàn thành được phần mềm hoàn chỉnh vậy nên nhóm đã ra kế hoạch để bổ sung.

Hiện tại có rất nhiều trang web dạy JavaScript miễn phí nên nhóm đã lựa chọn trang web [https://fullstack.edu.vn](https://fullstack.edu.vn/) để train các thành viên trong nhóm. Mong muốn là sau khóa học này thì các thành viên trong nhóm sẽ trang bị đầy đủ kiến thức và có khả năng lập trình vững vàng.

### 1.5.3 Nhóm các rủi ro không giải quyết được

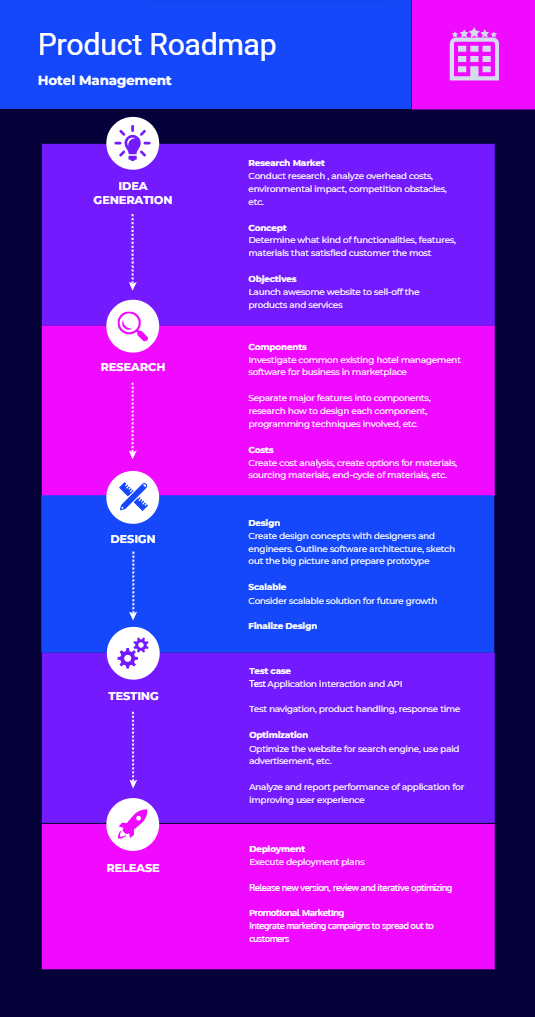
* Rủi ro lớn nhất của nhóm chính là kinh phí. Do các thành viên nhóm đều là sinh viên nên kinh phí đầu tư khá thấp dẫn đến việc thiếu hụt kinh phí khá nhiều.
* Về vấn đề này thì nhóm đã nhờ được người quen để có thể sử dụng một số dịch vụ giá rẻ

### 1.5.4 Nhóm các rủi ro chấp nhận được

Do một số thành viên nhóm có khả năng trễ deadline vì nhiều nguyên do khác nhau nên việc trễ deadline của nhóm dự định sẽ khiến việc chỉnh sửa lại sau cùng cũng bị trễ theo. Việc này nhóm cũng đã thống nhất là sẽ ra hẹn deadline sớm hơn và họp vào thời gian rảnh rỗi của mọi người để không bị trễ.

## 1.6/ High Level Abstract Product Planning

### 1.6.1/ High Level Abstract Product Roadmap



### 1.6.2/ High Level Abstract Development Planning

#### 1.6.2.1/ Development phases (RUP)

Date format: month/day/year

**Inception (4/3/2022 - 4/17/2022)**

Goals: this phase provides a general vision and the objective is to mitigate the risks by analysis. The scope of the project is understood exactly, and the stakeholders initializing the project should have a good understanding of the project’s ROI (return on investment).

* Project management
  + Plan the project and strategy
  + Evaluate project scope and risk
  + Managing/monitoring
  + Plan for next iteration
* Business modeling
  + Identify business processes
  + Refine business processes
  + Refine roles and responsibilities
* Requirements
  + Analyze the problem
  + Define the system
  + Manage the scope of system
  + Refine the system definition
  + Manage changing requirements
* Analysis and design
  + Perform architecture synthesis

**Elaboration (4/7/2022 - 4/28/2022)**

The goal is to baseline the architecture of the system to provide a stable basis for the bulk of the design and implementation in the construction phase. Addressing all the architectural risks of the project and producing prototypes of production-quality components that mitigate risks.

* Project management
  + Managing/monitoring
* Requirements
  + Software requirement specification
  + Manage changing requirements
* Architectural definition
  + Structure the implementation model
  + Refine the architecture
* Develop components/features

**Construction (4/16/2022 - 5/22/2022)**

Goals: This phase is actually building a project, the goal is clarifying the remaining requirements and completing the development. It manufacture the process, manage resources and control operations to optimize costs and quality, achieve useful version as rapidly as practical

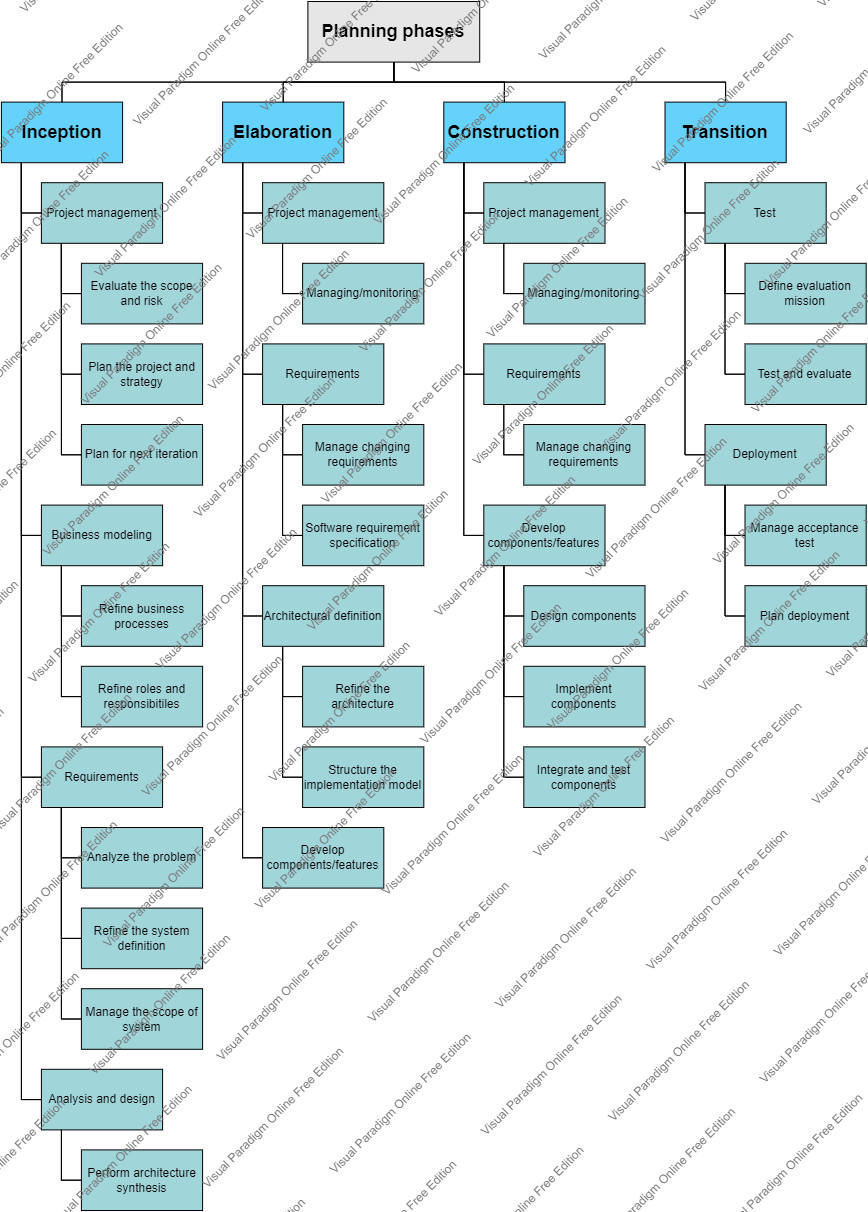
* Project management
  + Managing/monitoring
* Requirements
  + Manage changing requirements
* Develop components/features
  + Design components
  + Implement components
  + Integrate and test components

**Transition (5/13/2022 - 5/29/2022)**

The focus on the transition phase is that software is available and ready for delivery to end-users. Phase can span several iterations, testing products for release, making minor adjustments from user feedback, enhancing performance and usability.

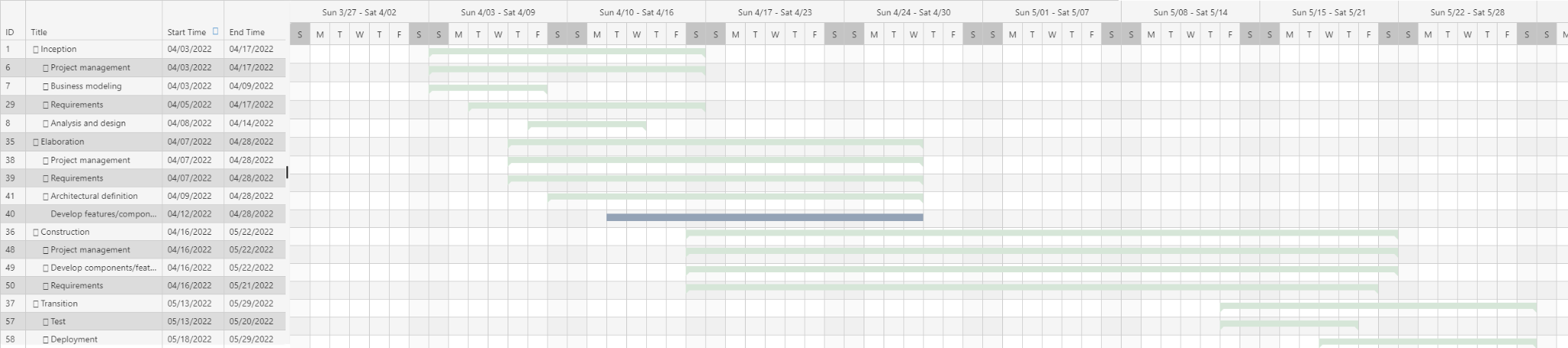
* Test
  + Define evaluation mission
  + Test and evaluate
* Deployment
  + Plan deployment
  + Manage Acceptance test

#### 1.6.2.2/ High Level Abstract Work Breakdown Structure

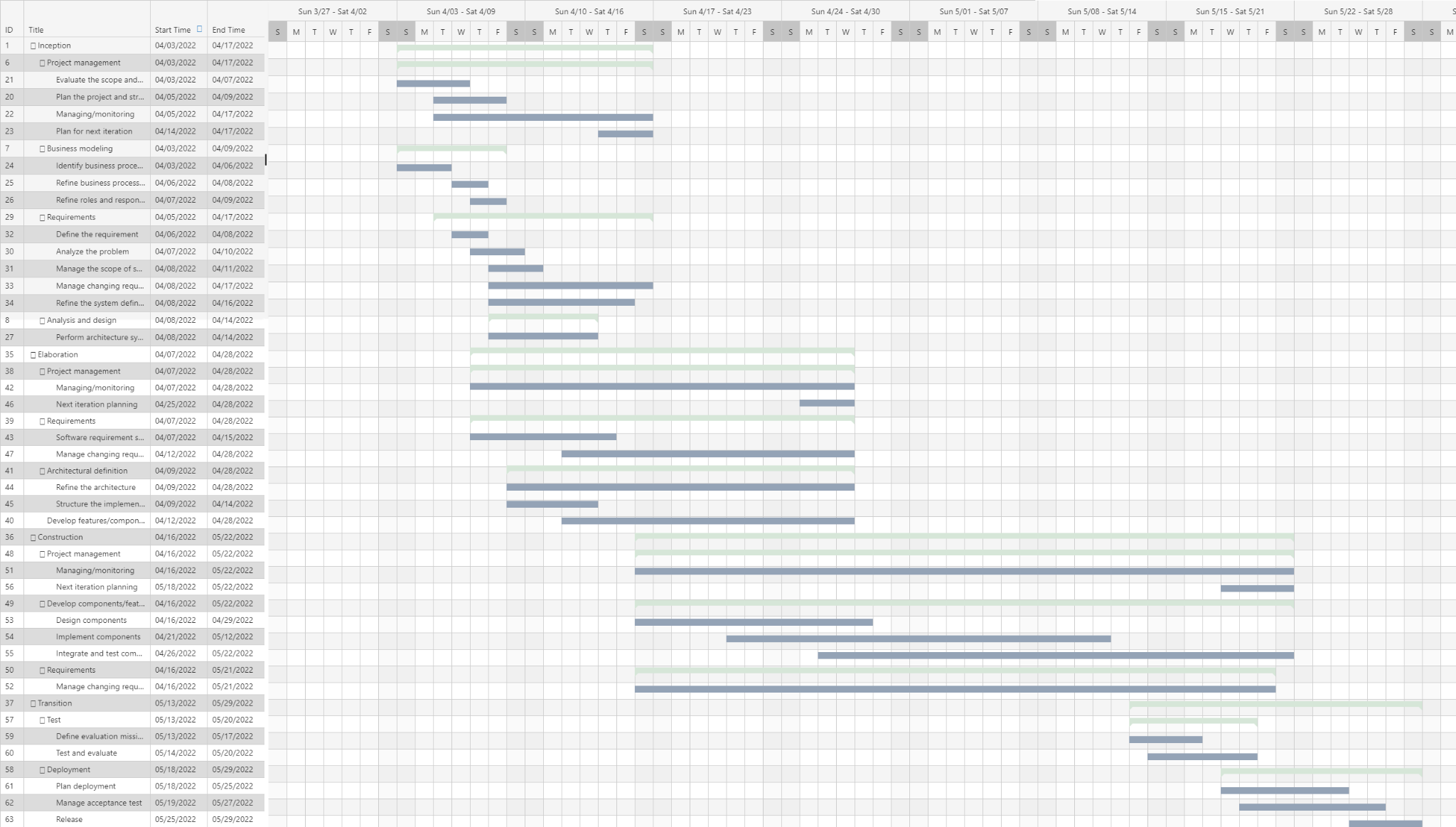


#### 1.6.2.3/ High Level Abstract Gantt Chart

* Overview Gantt Chart



* Detailed Gantt Chart (with detailed tasks)

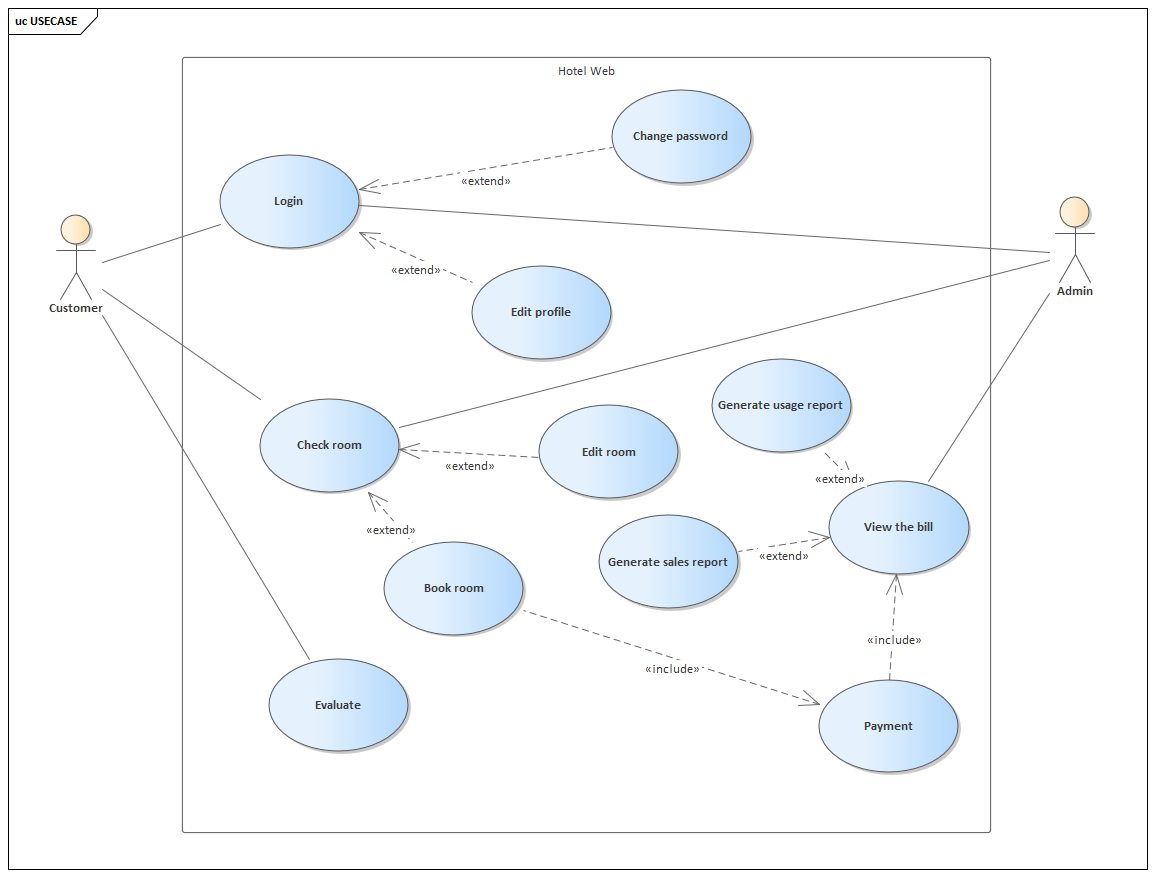


## 1.7/ User Requirements Modeling

### 1.7.1/ Use Case

#### 1.7.1.1 / UC Diagram for web:

##### Use case:

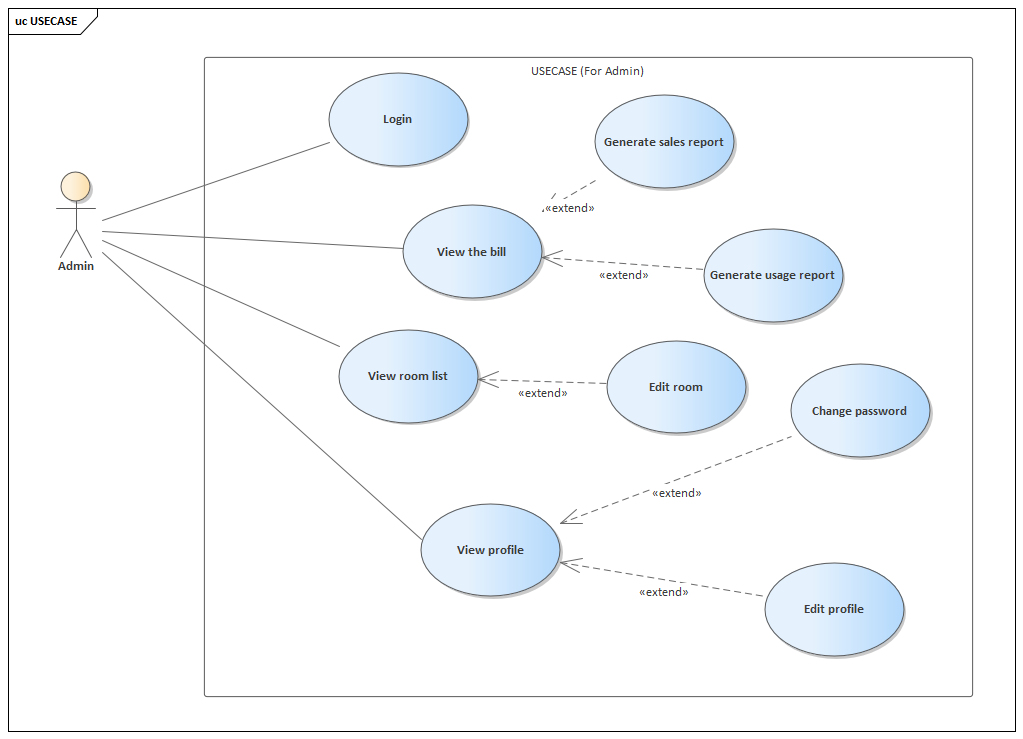


#### 

#### 

#### 1.7.1.2/ UC Diagram for admin:

##### Use case:



##### Đặc tả:

| *Use Case ID* | UC -100 |
| --- | --- |
| *Tên UseCase* | Login |
| *Tóm tắt* | Use này mô tả cách một người dùng đăng nhập vào hệ thống |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập vào hệ thống |
| *Kết quả* | Người dùng đăng nhập thành công |
| *Kịch bản chính(flow, scenario)* | 1. Hệ thống yêu cầu actor nhập tài khoản và mật khẩu  2. Actor nhập tên và mật khẩu  3. Hệ thống kiểm tra tên và mật khẩu được actor nhập và cho phép actor đăng nhập vào hệ thống  4. Kết thúc use case |
| *Kịch bản phụ* | A1. Tại bước 1 nếu actor chưa có tài khoản và mật khẩu thì phải xin cấp từ DBA.  A3. Tại bước 3, nếu actor nhập sai tên và mật khẩu. Hệ thống sẽ hiển thị thông báo, có thể thông báo với DBA để cấp lại hoặc hủy đăng nhập, lúc này use case kết thúc |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-101 |
| --- | --- |
| *Tên UseCase* | Generate sales report |
| *Tóm tắt* | Use case này mô tả việc actor tạo báo cáo theo nhu cầu của Admin |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống  Use case View the bill phải được thực hiện |
| *Kết quả* | Admin tạo được báo cáo cần thiết |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện USE CASE View the bill  2. Admin CHỌN tạo báo cáo doanh thu theo loại phòng  3. Hệ thống xử lý tính toán và tạo form báo cáo  4. Hệ thống xuất báo cáo trên giao diện web  5. Kết thúc use case |
| *Kịch bản phụ* | A3. Nếu quá trình xử lý gặp vấn đề, hệ thống sẽ thông báo và kết thúc use case |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-102 |
| --- | --- |
| *Tên UseCase* | View the bill |
| *Tóm tắt* | Use case này mô tả việc actor xem các hóa đơn đã thanh toán theo từng tháng |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống |
| *Kết quả* | Admin xem được danh sách các hóa đơn đã thanh toán theo từng tháng |
| *Kịch bản chính(flow, scenario)* | 1. Admin bấm CHỌN tháng để xem danh sách hóa đơn của tháng đó  2. Hệ thống tiếp nhận yêu cầu  3. Hệ thống đưa ra danh sách theo yêu cầu  4. Admin bấm CHỌN (TẠO BÁO CÁO) để tạo báo cáo  5. Thực hiện USE CASE generate report  6. Kết thúc use case |
| *Kịch bản phụ* | A2. Nếu yêu cầu không phù hợp hệ thống sẽ thông báo kết thúc use case.  A4. Nếu admin không bấm CHỌN thì kết thúc use case. |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-103 |
| --- | --- |
| *Tên UseCase* | Generate usage report |
| *Tóm tắt* | Use case này mô tả việc actor xem các hóa đơn đã thanh toán theo từng tháng |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống |
| *Kết quả* | Admin xem được danh sách các hóa đơn đã thanh toán theo từng tháng |
| *Kịch bản chính(flow, scenario)* | 1. Admin bấm CHỌN tháng để xem danh sách hóa đơn của tháng đó  2. Hệ thống tiếp nhận yêu cầu  3. Hệ thống đưa ra danh sách theo yêu cầu  4. Admin bấm CHỌN (TẠO BÁO CÁO) để tạo báo cáo  5. Thực hiện USE CASE generate report  6. Kết thúc use case |
| *Kịch bản phụ* | A2. Nếu yêu cầu không phù hợp hệ thống sẽ thông báo kết thúc use case.  A4. Nếu admin không bấm CHỌN thì kết thúc use case. |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-104 |
| --- | --- |
| *Tên UseCase* | View room list |
| *Tóm tắt* | Use case này mô tả việc actor xem các tính chất của các phòng có trong khách sạn |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống |
| *Kết quả* | Admin xem được các tính chất của các phòng có trong khách sạn |
| *Kịch bản chính(flow, scenario)* | 1. Admin bấm CHỌN loại phòng  2. Hệ thống tiếp nhận yêu cầu  3. Hệ thống đưa ra danh sách phòng theo yêu cầu  4. Admin bấm CHỌN tên phòng, mã phòng  5. Hệ thống tiếp nhận yêu cầu  6. Hệ thống đưa ra thông tin, tình trạng phòng theo yêu cầu  7. Thực hiện use case Edit room  8. Kết thúc use case |
| *Kịch bản phụ* | A7. Nếu admin không thực hiện use case Edit room thì kết thúc use case |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-105 |
| --- | --- |
| *Tên UseCase* | Edit room |
| *Tóm tắt* | Use case này mô tả việc actor chỉnh sửa các thông tin của một phòng có trong khách sạn |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống |
| *Kết quả* | Admin chỉnh sửa được những thông tin của phòng |
| *Kịch bản chính(flow, scenario)* | 1. Use case View room list được thực hiện  2. Admin CHỌN nút chỉnh sửa  3. Hệ thống tiếp nhận yêu cầu  4. Hệ thống hiển thị form cho admin thông tin để chỉnh sửa  5. Admin chỉnh sửa thuộc tính của phòng  6. Admin xác nhận việc chỉnh sửa  7. Hệ thống xác nhận việc chỉnh sửa  8. Kết thúc use case |
| *Kịch bản phụ* | A7. Nếu hệ thống không thể xác nhận việc chỉnh sửa, hệ thống sẽ thông báo và kết thúc use case. |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-106 |
| --- | --- |
| *Tên UseCase* | Manage profile |
| *Tóm tắt* | Use case này mô tả việc hệ thống cho phép admin xem thông tin profile của mình |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống |
| *Kết quả* | Hệ thống cho phép admin xem thông tin profile |
| *Kịch bản chính(flow, scenario)* | 1. Admin CHỌN xem profile  2. Hệ thống ghi nhận yêu cầu  3. Hệ thống xác minh người truy xuất  4. Hệ thống truy xuất trên cơ sở dữ liệu  5. Hệ thống xác nhận yêu cầu  6. Thực hiện use case View profile  7. Kết thúc use case |
| *Kịch bản phụ* | A3. Nếu hệ thống không thể xác minh admin, sẽ hiển thị thông báo và kết thúc use case |
| *Ràng buộc phi chức năng* | Mã profile của admin phải có trên cơ sở dữ liệu |

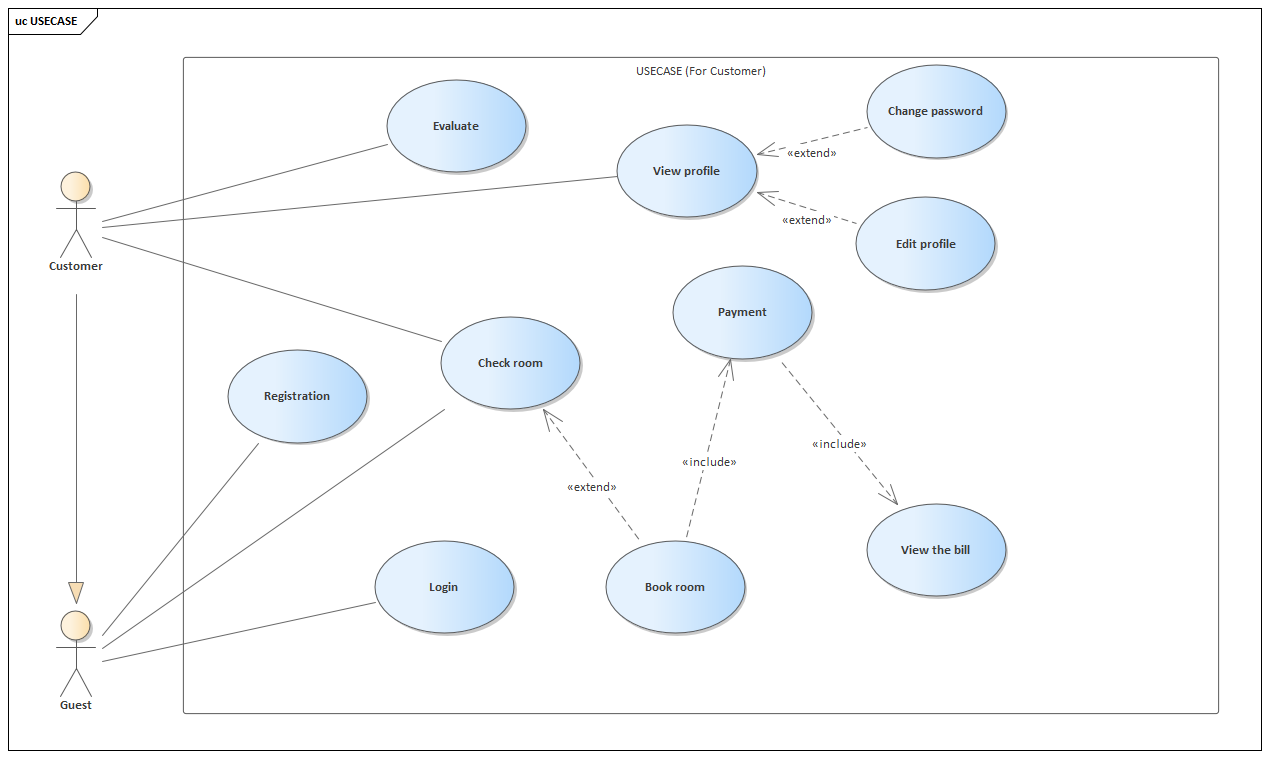
| *Use Case ID* | UC-107 |
| --- | --- |
| *Tên UseCase* | View profile |
| *Tóm tắt* | Use case này mô tả việc admin xem thông tin profile của mình |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống |
| *Kết quả* | Hệ thống hiển thị thông tin profile |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện USE CASE Manage profile  2. Hệ thống hiển thị view thông tin của admin  3. Thực hiện use case Change password  4. Thực hiện use case Edit profile  5. Kết thúc use case |
| *Kịch bản phụ* | A3. Nếu admin không muốn thực hiện use case Change password thì bỏ qua bước 3  A4. Nếu admin không muốn thực hiện use case Edit profile, thì kết thúc use case |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-108 |
| --- | --- |
| *Tên UseCase* | Change password |
| *Tóm tắt* | Use case này mô tả việc admin thay đổi password cho tài khoản của mình |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống |
| *Kết quả* | Hệ thống xác nhận việc thay đổi password |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện use case View profile  2. Admin bấm CHỌN thay đổi mật khẩu  3. Admin nhập mật khẩu cũ  4. Admin nhập mật khẩu mới và CHỌN xác nhận  5. Hệ thống ghi nhận yêu cầu  6. Hệ thống xác nhận yêu cầu  7. Kết thúc use case |
| *Kịch bản phụ* | A6. Nếu hệ thống không xác nhận yêu cầu, sẽ thông báo và quay lại bước 3 hoặc kết thúc use case |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-109 |
| --- | --- |
| *Tên UseCase* | Edit profile |
| *Tóm tắt* | Use case này mô tả việc admin thay đổi thông tin profile của mình |
| *Tác nhân* | Admin |
| *Điều kiện tiên quyết* | Admin phải đăng nhập được vào hệ thống |
| *Kết quả* | Hệ thống xác nhận việc thông tin |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện use case View profile  2. Admin bấm CHỌN edit profile  3. Hệ thống ghi nhận yêu cầu  4. Hệ thống xác nhận yêu cầu  5. Hệ thống hiển thị form để admin chỉnh sửa  6. Admin chỉnh sửa profile  7. Admin xác nhận việc chỉnh sửa  8. Hệ thống ghi nhận yêu cầu  9. Hệ thống xác nhận yêu cầu  10. Kết thúc use case |
| *Kịch bản phụ* | A9. Nếu hệ thống không xác nhận yêu cầu, sẽ thông báo và kết thúc use case |
| *Ràng buộc phi chức năng* | Các thông tin được nhập sẽ phải đảm bảo các ràng buộc về ngữ nghĩa. |

#### 1.7.1.3 / UC Diagram for customer:

##### Use case:



##### Đặc tả:

| *Use Case ID* | UC-200 |
| --- | --- |
| *Tên UseCase* | Evaluate |
| *Tóm tắt* | Use case này mô tả việc customer đánh giá dịch vụ của khách sạn |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* | Customer phải đăng nhập được vào hệ thống |
| *Kết quả* | Customer đánh giá được dịch vụ của hệ thống |
| *Kịch bản chính(flow, scenario)* | 1. Customer vấn CHỌN đánh giá  2. Hệ thống ghi nhận yêu cầu  3. Hệ thống xác nhận yêu cầu  4. Hệ thống hiển thị form cho customer đánh giá  5. Customer đánh giá dịch vụ  6. Kết thúc use case |
| *Kịch bản phụ* | A3. Nếu hệ thống không xác nhận yêu cầu, sẽ thông báo và kết thúc use case |
| *Ràng buộc phi chức năng* | Các thông tin được nhập sẽ phải đảm bảo các ràng buộc về ngữ nghĩa. |

| *Use Case ID* | UC-201 |
| --- | --- |
| *Tên UseCase* | View profile |
| *Tóm tắt* | Use case này mô tả việc Customer xem thông tin profile của mình |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* | Customer phải đăng nhập được vào hệ thống |
| *Kết quả* | Hệ thống hiển thị thông tin profile |
| *Kịch bản chính(flow, scenario)* | 1. Customer CHỌN xem profile  2. Hệ thống ghi nhận yêu cầu  3. Hệ thống xác nhận yêu cầu  4. Hệ thống hiển thị view thông tin của Customer  5. Thực hiện use case Change password  6. Thực hiện use case Edit profile  7. Kết thúc use case |
| *Kịch bản phụ* | A3. Nếu Customer không muốn thực hiện use case Change password thì bỏ qua bước 3  A4. Nếu Customer không muốn thực hiện use case Edit profile, thì kết thúc use case |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC-202 |
| --- | --- |
| *Tên UseCase* | Change password |
| *Tóm tắt* | Use case này mô tả việc Customer thay đổi password cho tài khoản của mình |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* | Customer phải đăng nhập được vào hệ thống |
| *Kết quả* | Hệ thống xác nhận việc thay đổi password |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện use case View profile  2. Customer bấm CHỌN thay đổi mật khẩu  3. Hệ thống tiếp nhận yêu cầu  4. Hệ thống xác nhận yêu cầu  5. Hệ thống hiển thị form để Customer nhập  6. Customer nhập mật khẩu cũ  7. Customer nhập mật khẩu mới và CHỌN xác nhận  8. Hệ thống ghi nhận yêu cầu  9. Hệ thống xác nhận yêu cầu  10. Kết thúc use case |
| *Kịch bản phụ* | A4. Nếu hệ thống không xác nhận yêu cầu, sẽ thông báo và kết thúc use case  A9. Nếu hệ thống không xác nhận yêu cầu, sẽ thông báo và quay lại bước 6 hoặc kết thúc use case |
| *Ràng buộc phi chức năng* |  |

s

| *Use Case ID* | UC-203 |
| --- | --- |
| *Tên UseCase* | Edit profile |
| *Tóm tắt* | Use case này mô tả việc Customer thay đổi thông tin profile của mình |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* | Customer phải đăng nhập được vào hệ thống |
| *Kết quả* | Hệ thống xác nhận việc thông tin |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện use case View profile  2. Customer bấm CHỌN edit profile  3. Hệ thống ghi nhận yêu cầu  4. Hệ thống xác nhận yêu cầu  5. Hệ thống hiển thị form để Customer chỉnh sửa  6. Customer chỉnh sửa profile  7. Customer xác nhận việc chỉnh sửa  8. Hệ thống ghi nhận yêu cầu  9. Hệ thống xác nhận yêu cầu  10. Kết thúc use case |
| *Kịch bản phụ* | A9. Nếu hệ thống không xác nhận yêu cầu, sẽ thông báo và kết thúc use case |
| *Ràng buộc phi chức năng* | Các thông tin được nhập sẽ phải đảm bảo các ràng buộc về ngữ nghĩa. |

| *Use Case ID* | UC -204 |
| --- | --- |
| *Tên UseCase* | Login |
| *Tóm tắt* | Use này mô tả cách Customer đăng nhập vào hệ thống |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* | Customer phải có tài khoản và mật khẩu tồn tại trong cơ sở dữ liệu |
| *Kết quả* | Customer đăng nhập thành công |
| *Kịch bản chính(flow, scenario)* | 1. Hệ thống yêu cầu actor nhập tài khoản và mật khẩu  2. Actor nhập tên và mật khẩu  3. Hệ thống kiểm tra tên và mật khẩu được actor nhập và cho phép actor đăng nhập vào hệ thống  4. Kết thúc use case |
| *Kịch bản phụ* | A4. Nếu hệ thống không xác nhận, sẽ hiển thị thông báo và kết thúc usecase |
| *Ràng buộc phi chức năng* | Tài khoản và mật khẩu của customer phải tồn tại trên cơ sở dữ liệu |

| *Use Case ID* | UC -205 |
| --- | --- |
| *Tên UseCase* | Registration |
| *Tóm tắt* | Use này mô tả cách Customer đăng ký tài khoản để đăng nhập vào hệ thống |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* |  |
| *Kết quả* | Customer đăng nhập thành công |
| *Kịch bản chính(flow, scenario)* | 1. Customer bấm CHỌN đăng ký để đăng ký tài khoản  2. Hệ thống tiếp nhận yêu cầu  3. Hệ thống chấp nhận yêu cầu  4. Hệ thống hiển thị form đăng ký  5. Customer nhập vào form đăng ký  6. Customer chọn xác nhận để xác nhận đăng ký tài khoản  7. Kết thúc use case. |
| *Kịch bản phụ* | A3. Nếu hệ thống không chấp nhận yêu cầu, sẽ thông báo và kết thúc use case |
| *Ràng buộc phi chức năng* | Form đăng ký phải bị ràng buộc chức năng và dữ liệu trong cơ sở dữ liệu  Tài khoản tạo mới không được trùng với tài khoản trong hệ thống  Phải điền đầy đủ thông tin vào form |

| *Use Case ID* | UC -206 |
| --- | --- |
| *Tên UseCase* | Check room |
| *Tóm tắt* | Use case này mô tả Customer có thể xem được các dịch vụ và tình trạng các loại phòng hiện có ở khách sạn |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* |  |
| *Kết quả* | Customer xem được danh sách các dịch vụ và tình trạng các phòng hiện có ở khách sạn |
| *Kịch bản chính(flow, scenario)* | 1. Customer truy cập trang web  2. Customer lựa chọn loại phòng  3. Customer chọn tình trạng phòng  4. Hệ thống xử lý yêu cầu  5. Hệ thống xác nhận yêu cầu  6. Hệ thống đưa ra list danh sách các phòng phù hợp với yêu cầu  7. Customer lựa chọn phòng mong muốn  8. Thực hiện USE CASE Book room  9. Kết thúc use case |
| *Kịch bản phụ* | A3. Nếu hệ thống không chấp nhận yêu cầu, sẽ thông báo và kết thúc use case  A8. Nếu Customer không thực hiện use case Book room thì kết thúc use case |
| *Ràng buộc phi chức năng* | Form đăng ký phải bị ràng buộc chức năng và dữ liệu trong cơ sở dữ liệu  Tài khoản tạo mới không được trùng với tài khoản trong hệ thống  Phải điền đầy đủ thông tin vào form |

| *Use Case ID* | UC -207 |
| --- | --- |
| *Tên UseCase* | Book room |
| *Tóm tắt* | Use case này mô tả việc customer đặt phòng của khách sạn |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* | Customer phải đăng nhập vào hệ thống |
| *Kết quả* | Customer đặt được phòng của khách sạn |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện USECASE Check room  2. Customer bấm CHỌN phòng muốn đặt  3. Hệ thống tiếp nhận yêu cầu  4. Hệ thống xác nhận yêu cầu  5. Hệ thống hiển thị view để Customer xác nhận  6. Customer chọn số người  7. Customer chọn loại khách  8. Customer nhập cmnd, địa chỉ  9. Customer nhập email, số điện thoại  10. Customer CHỌN thanh toán  11. Thực hiện USE CASE Payment  12. Kết thúc use case |
| *Kịch bản phụ* | A4. Nếu hệ thống không tiếp nhận yêu cầu, sẽ thông báo và kết thúc use case |
| *Ràng buộc phi chức năng* | Không thể đặt phòng khi phòng trong tình trạng ‘Đang hoạt động’  Giới hạn số người tối đa là 3  Chỉ có 2 loại khách  Tất cả thông tin đều là bắt buộc |

| *Use Case ID* | UC -208 |
| --- | --- |
| *Tên UseCase* | Payment |
| *Tóm tắt* | Use case này mô tả việc customer thanh toán phòng đã đặt |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* | Customer phải đăng nhập vào hệ thống  Use case Book room phải được thực hiện |
| *Kết quả* | Customer đặt được phòng của khách sạn |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện USE CASE Book room  2. Customer CHỌN thanh toán  3. Hệ thống ghi nhận yêu cầu  4. Hệ thống xác nhận yêu cầu  5. Hệ thống đưa ra form để người dùng điền thông tin  6. Customer thanh toán  7. Thực hiện use case View the bill  8. Kết thúc use case |
| *Kịch bản phụ* | A3. Nếu hệ thống không xác nhận yêu cầu, sẽ thông báo và kết thúc use case.  A7. Nếu hệ thống không xác nhận việc thanh toán thì quay lại bước 5 hoặc kết thúc use case. |
| *Ràng buộc phi chức năng* |  |

| *Use Case ID* | UC -209 |
| --- | --- |
| *Tên UseCase* | View the bill |
| *Tóm tắt* | Use case này mô tả việc hệ thống xuất hóa đơn cho khách hàng |
| *Tác nhân* | Customer |
| *Điều kiện tiên quyết* | Use case Payment phải được thực hiện  Customer phải đăng nhập vào hệ thống |
| *Kết quả* | Customer đặt được phòng của khách sạn |
| *Kịch bản chính(flow, scenario)* | 1. Thực hiện Use case Payment  2. Hệ thống xác nhận và ghi nhận việc thanh toán  3. Hệ thống xử lý  4. Hệ thống xuất hóa đơn xác nhận cho Customer  5. Kết thúc USECASE |
| *Kịch bản phụ* | A2. Nếu hệ thống không xác nhận việc thanh toán, sẽ quay lại bước 1 hoặc thông báo và kết thúc USECASE |
| *Ràng buộc phi chức năng* |  |