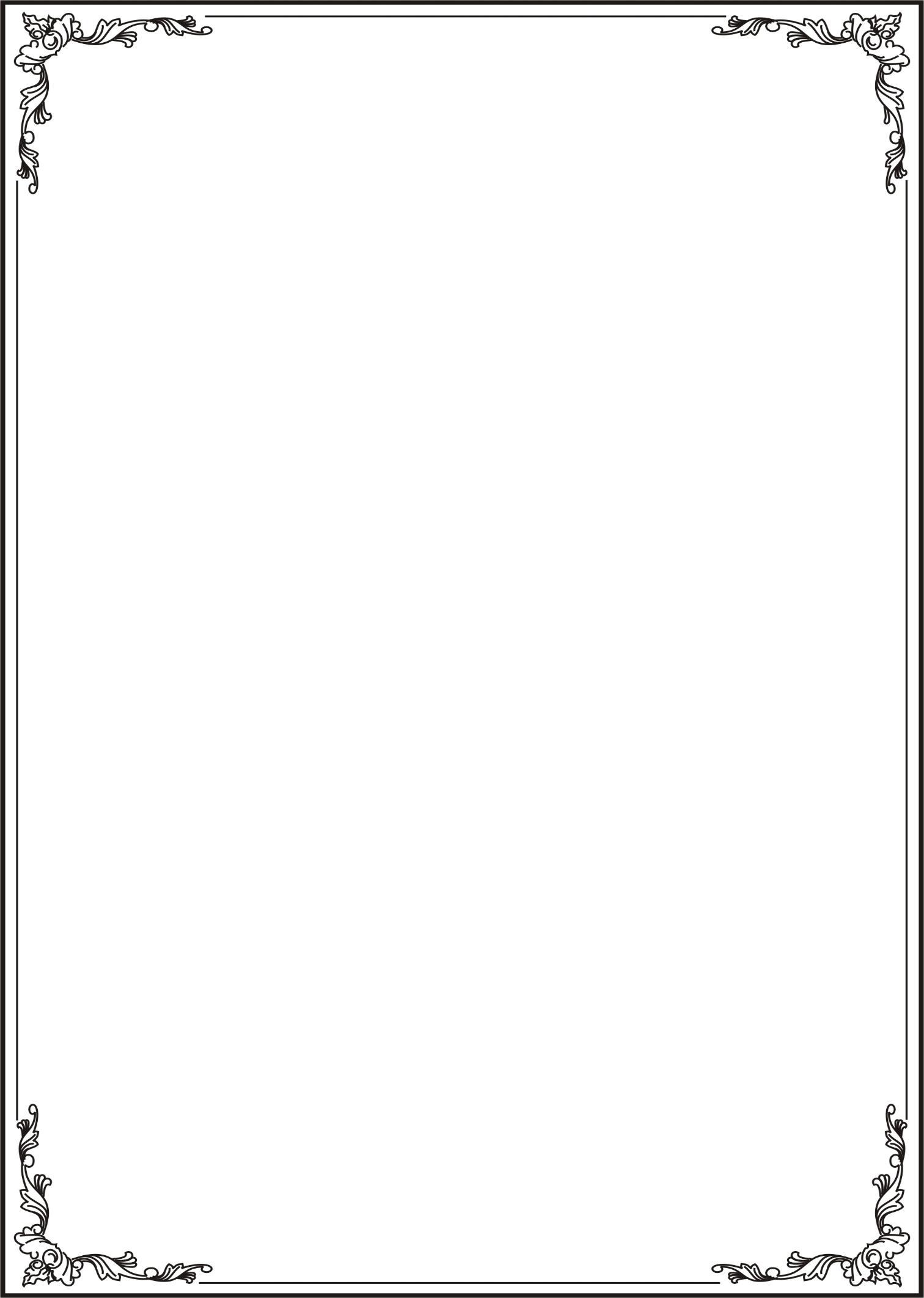
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

\_\_\_\_\_\_\_\_\_\_\_\_



**BÁO CÁO ĐỒ ÁN**

**CÔNG NGHỆ PHẦN MỀM**

**APP ĐẶT BÀN NHÀ HÀNG**

**GVHD: Thầy Huỳnh Xuân Phụng**

**SVTH : Nguyễn Quang Vũ 18110241**

**Bùi Hà Nhi 18110168**

**Huỳnh Trần Thảo Nhi 18110169**

**Hồ Chí Minh, ngày 15 tháng 06 năm 2021**

# **ĐIỂM SỐ**

|  |  |  |  |
| --- | --- | --- | --- |
| TIÊU CHÍ | NỘI DUNG | TRÌNH BÀY | TỔNG |
| ĐIỂM |  |  |  |

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

Giáo viên hướng dẫn

(***Ký và ghi họ tên***)

**Thầy Huỳnh Xuân Phụng**

# **LỜI CAM ĐOAN**

Chúng em xin cam đoan đề tài cuối kỳ “Dự án xây dựng ứng dụng App Đặt Bàn” là thành quả của quá trình học tập và nghiên cứu dưới sự hướng dẫn của giáo viên hướng dẫn là thầy Huỳnh Xuân Phụng. Mọi sự tham khảo sử dụng trong đồ án đều được trích dẫn từ các nguồn tài liệu trong báo cáo và tài liệu tham khảo.

TP.Hồ Chí Minh, ngày 15 tháng 06 năm 2021

Nhóm sinh viên thực hiện

**Bùi Hà Nhi - 18110168**

**Huỳnh Trần Thảo Nhi - 18110169**

**Nguyễn Quang Vũ - 18110241**

# **LỜI CẢM ƠN**

Để hoàn thành tốt đồ án và bài báo cáo này, chúng em xin gửi lời cảm ơn chân thành đến giảng viên, thầy Huỳnh Xuân Phụng, người đã hỗ trợ chúng em trong quá trình làm đề tài. Chúng em cảm ơn thầy đã giải đáp những thắc mắc và đưa ra những góp ý, chỉnh sửa kịp thời giúp chúng em khắc phục nhược điểm và hoàn thành tốt cũng như đúng thời hạn Khoa đã đề ra.

Chúng em cũng xin gửi lời cảm ơn chân thành các quý thầy cô trong khoa Đào tạo Chất Lượng Cao nói chung và ngành Công Nghệ Thông Tin nói riêng đã tận tình truyền đạt những kiến thức cần thiết giúp chúng em có nền tảng để làm nên đề tài này, đã tạo điều kiện để chúng em có thể tìm hiểu và thực hiện tốt đề tài. Cùng với đó, chúng em xin được gửi lời cảm ơn đến các bạn cùng khóa đã cung cấp nhiều thông tin và kiến thức hữu ích giúp chúng em có thể hoàn thiện hơn đề tài của mình.

Đề tài và bài báo cáo được chúng em thực hiện trong khoảng thời gian ngắn, với những hạn chế về kiến thức chuyên môn cũng như về mặt kỹ thuật và kinh nghiệm trong việc thực hiện một dự án. Do đó, trong quá trình làm nên đề tài có những thiếu sót là điều không thể tránh khỏi nên chúng em rất mong nhận được những ý kiến đóng góp quý báu của các quý thầy cô để kiến thức của chúng em được hoàn thiện hơn và chúng em có thể làm tốt hơn nữa trong những lần sau. Chúng em xin chân thành cảm ơn.

Cuối lời, chúng em kính chúc quý thầy cô luôn dồi dào sức khỏe và thành công hơn nữa trong sự nghiệp trồng người. Một lần nữa chúng em xin chân thành cảm ơn.

TP.Hồ Chí Minh, ngày 15 tháng 06 năm 2021

Nhóm sinh viên thực hiện

**Bùi Hà Nhi - 18110168**

**Huỳnh Trần Thảo Nhi - 18110169**

**Nguyễn Quang Vũ - 18110241**

# **PHÂN CÔNG**

## **1. Vai trò của các thành viên trong nhóm**

|  |  |  |
| --- | --- | --- |
|  | **Công việc** | **Phân công** |
| **Phân tích yêu cầu** | Xác định yêu cầu | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |
| Danh sách yêu cầu về user | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |
| **Phân tích chức năng** | Yêu cầu chức năng | Huỳnh Trần Thảo Nhi |
| Yêu cầu phi chức năng | Nguyễn Quang Vũ |
| User của phần mềm | Bùi Hà Nhi |
| **Đặc tả yêu cầu** | Sử dụng bảng | Bùi Hà Nhi |
| Sử dụng ngôn ngữ có cấu trúc | Huỳnh Trần Thảo Nhi |
| Sử dụng sơ đồ use case | Nguyễn Quang Vũ |
| Requirement Diagram | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |
| **Thiết kế lược đồ** | System Model Diagram | Bùi Hà Nhi |
| Use case Diagram tổng quan hệ thống | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |
| Use case Diagrams chi tiết cho từng chức năng | Huỳnh Trần Thảo Nhi |
| Class Diagram tổng quan hệ thống | Bùi Hà Nhi |
| Class Diagram chi tiết | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |
| MVC Architectures | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |
| Sequence Diagrams chi tiết cho từng chức năng | Huỳnh Trần Thảo Nhi |
| State Diagrams chi tiết cho từng chức năng | Bùi Hà Nhi |
| **Thiết kế kịch bản** | Use case Scenarios chi tiết cho từng chức năng | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |
| Test case Scenarios chi tiết cho từng chức năng | Bùi Hà Nhi |
| **Thiết kế giao diện** | Login Page | Bùi Hà Nhi |
| Signup Page | Huỳnh Trần Thảo Nhi |
| Menu Left Drawer | Nguyễn Quang Vũ |
| Home Page | Bùi Hà Nhi |
| Detail Restaurant Page | Huỳnh Trần Thảo Nhi |
| Booking Tables Page | Nguyễn Quang Vũ |
| Rating Layout | Huỳnh Trần Thảo Nhi |
| Add Restaurant Page | Nguyễn Quang Vũ |
| Và một số giao diện khác | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |
| **Code chức năng cho các phiên bản 1.0, 1.1, 2.0, 3.0** | 1. Chức năng Đăng nhập | Bùi Hà Nhi |
| 2. Chức năng Đăng ký | Huỳnh Trần Thảo Nhi |
| 3. Chức năng Đăng xuất | Nguyễn Quang Vũ |
| 4. Chức năng Tra cứu nhà hàng | Bùi Hà Nhi |
| 5. Chức năng Đặt bàn | Huỳnh Trần Thảo Nhi |
| 6. Chức năng Đánh giá nhà hàng | Nguyễn Quang Vũ |
| 7. Chức năng Theo dõi đơn hàng | Bùi Hà Nhi |
| 8. Chức năng Quản lý đơn hàng (Admin Role) | Huỳnh Trần Thảo Nhi |
| 9. Chức năng Quản lý nhà hàng (Admin Role) | Nguyễn Quang Vũ |
| 10. Chức năng Yêu thích nhà hàng | Nguyễn Quang Vũ |
| **Viết báo cáo và tổng hợp** | Báo cáo hoàn chỉnh đề tài cuối kỳ môn Công nghệ phần mềm, đề tài “Xây dựng ứng dụng App Đặt bàn” | - Nguyễn Quang Vũ  - Bùi Hà Nhi  - Huỳnh Trần Thảo Nhi |

## **2. Vai trò của Project Manager**

|  |  |  |  |
| --- | --- | --- | --- |
| **Họ tên** | **Giai đoạn** | **Nhiệm vụ** | **Đánh giá** |
| **Nguyễn Quyết Sinh** | Initiating phase | Integration management | Xây dựng và phát triển Project Charter. |
| Planning phase | Integration management | Xây dựng và phát triển Project Management Plan. |
| Scope management | Xác định và quản lý Scope (phạm vi), tạo Work Breakdown Structure và thu thập các yêu cầu. |
| Time management | Lập kế hoạch, xác định và phát triển lịch trình, hoạt động, ước tính nguồn lực và thời lượng hoạt động. |
| Costs management | Lập kế hoạch và ước tính chi phí cũng như xác định ngân sách. |
| Human Resource management | Lập kế hoạch và xác định nhu cầu nguồn nhân lực. |
| Risk management | Lập kế hoạch và xác định các rủi ro tiềm ẩn, thực hiện phân tích rủi ro định tính và định lượng cũng như hoạch định các chiến lược giảm thiểu rủi ro. |
| Executing | Integration management | Chỉ đạo và quản lý mọi công việc cho dự án |
| Human Resource management | Lựa chọn, phát triển và quản lý nhóm dự án |
| Risk management | Kiểm soát các rủi ro, cũng như giảm thiểu rủi ro. |
| Monitoring and controlling | Integration management | Giám sát và kiểm soát công việc của dự án và quản lý mọi thay đổi cần thiết |
| Scope management | Xác nhận và kiểm soát phạm vi của dự án |
| Time management | Kiểm soát phạm vi của dự án |
| Costs management | Kiểm soát chi phí dự án |
| Closing | Integration management | Kết thúc tất cả các giai đoạn của dự án |

# **MỤC LỤC**

**[ĐIỂM SỐ 3](#_Toc1762)**

**[LỜI CAM ĐOAN 4](#_Toc17933)**

**[LỜI CẢM ƠN 5](#_Toc28555)**

**[PHÂN CÔNG 6](#_Toc23878)**

[1. Vai trò của các thành viên trong nhóm 6](#_Toc14305)

[2. Vai trò của Project Manager 8](#_Toc21846)

**[1. PHẦN 1: GIỚI THIỆU 13](#_Toc15769)**

**[2. PHẦN 2: REQUIREMENT DOCUMENT 15](#_Toc12332)**

**[3. PHẦN 3: ĐẶC TẢ, PHÂN TÍCH VÀ THIẾT KẾ 20](#_Toc27122)**

**[3.1. Thiết kế use case tổng quan hệ thống 20](#_Toc22800)**

**[3.2. Class Diagrams 20](#_Toc913)**

**[3.3. Usecase 1: Đăng nhập 21](#_Toc29932)**

*[3.3.1. Mô hình MVC](#_Toc1265)* [21](#_Toc1265)

*[3.3.2. Usecase Diagram của usecase Đăng nhập](#_Toc17553)* [21](#_Toc17553)

*[3.3.3. Scenario của usecase Đăng nhập](#_Toc8922)* [21](#_Toc8922)

*[3.3.4. Sequence Diagram của usecase Đăng nhập](#_Toc11254)* [22](#_Toc11254)

*[3.3.5. State Diagram của usecase Đăng nhập](#_Toc29292)* [23](#_Toc29292)

*[3.3.6. Test case của usecase Đăng nhập](#_Toc20638)* [23](#_Toc20638)

**[3.4. Usecase 2: Đăng ký 24](#_Toc23673)**

*[3.4.1. Mô hình MVC](#_Toc21508)* [24](#_Toc21508)

*[3.4.2. Usecase Diagram của Đăng ký](#_Toc12475)* [24](#_Toc12475)

*[3.4.3. Scenario của usecase Đăng ký](#_Toc21120)* [25](#_Toc21120)

*[3.4.4. Sequence Diagram của usecase Đăng ký](#_Toc32677)* [26](#_Toc32677)

*[3.4.5. State Diagram của usecase Đăng ký](#_Toc21061)* [26](#_Toc21061)

*[3.4.6. Test case của usecase Đăng ký](#_Toc22503)* [26](#_Toc22503)

**[3.5. Usecase 3: Đăng xuất 28](#_Toc27498)**

*[3.5.1. Mô hình MVC](#_Toc11481)* [28](#_Toc11481)

*[3.5.2. Usecase Diagram của Đăng xuất](#_Toc31001)* [28](#_Toc31001)

*[3.5.3. Scenario của usecase Đăng xuất](#_Toc5474)* [28](#_Toc5474)

*[3.5.4. Sequence Diagram của usecase Đăng xuất](#_Toc25946)* [30](#_Toc25946)

*[3.5.5. State Diagram của usecase Đăng xuất](#_Toc20263)* [30](#_Toc20263)

*[3.5.6. Test case của usecase Đăng xuất](#_Toc27669)* [30](#_Toc27669)

**[3.6. Usecase 4: Tra cứu nhà hàng 31](#_Toc4086)**

*[3.6.1. Mô hình MVC](#_Toc22336)* [31](#_Toc22336)

*[3.6.2. Usecase Diagram của Tra cứu nhà hàng](#_Toc16335)* [31](#_Toc16335)

*[3.6.3. Scenario của usecase Tra cứu nhà hàng](#_Toc1358)* [32](#_Toc1358)

*[3.6.4. Sequence Diagram của usecase Tra cứu nhà hàng](#_Toc18428)* [33](#_Toc18428)

*[3.6.5. State Diagram của usecase Tra cứu nhà hàng](#_Toc21602)* [33](#_Toc21602)

*[3.6.6. Test case của usecase Tra cứu nhà hàng](#_Toc18448)* [34](#_Toc18448)

**[3.7. Usecase 5: Đặt bàn 34](#_Toc8148)**

*[3.7.1. Mô hình MVC](#_Toc18382)* [34](#_Toc18382)

*[3.7.2. Usecase Diagram của Đặt bàn nhà hàng](#_Toc11007)* [35](#_Toc11007)

*[3.7.3. Scenario của usecase Đặt bàn nhà hàng](#_Toc30014)* [35](#_Toc30014)

*[3.7.4. Sequence Diagram của usecase Đặt bàn nhà hàng](#_Toc27516)* [36](#_Toc27516)

*[3.7.5. State Diagram của usecase Đặt bàn nhà hàng](#_Toc14463)* [37](#_Toc14463)

*[3.7.6. Test case của usecase Đặt bàn nhà hàng](#_Toc30258)* [37](#_Toc30258)

**[3.8. Usecase 6: Đánh giá 38](#_Toc29461)**

*[3.8.1. Mô hình MVC](#_Toc11747)* [38](#_Toc11747)

*[3.8.2. Usecase Diagram của Đánh giá nhà hàng](#_Toc1818)* [38](#_Toc1818)

*[3.8.3. Scenario của usecase Đánh giá nhà hàng](#_Toc24653)* [38](#_Toc24653)

*[3.8.4. Sequence Diagram của usecase Đánh giá nhà hàng](#_Toc5295)* [39](#_Toc5295)

*[3.8.5. State Diagram của usecase Đánh giá nhà hàng](#_Toc17478)* [40](#_Toc17478)

*[3.8.6. Test case của usecase Đánh giá nhà hàng](#_Toc26936)* [40](#_Toc26936)

**[3.9. Usecase 7: Theo dõi đơn hàng 41](#_Toc20846)**

*[3.9.1. Mô hình MVC](#_Toc32024)* [41](#_Toc32024)

*[3.9.2. Usecase Diagram của Theo dõi đơn hàng](#_Toc20707)* [41](#_Toc20707)

*[3.9.3. Scenario của usecase Theo dõi đơn hàng](#_Toc1560)* [41](#_Toc1560)

*[3.9.4. Sequence Diagram của usecase Theo dõi đơn hàng](#_Toc30182)* [42](#_Toc30182)

*[3.9.5. State Diagram của usecase Theo dõi đơn hàng](#_Toc21672)* [43](#_Toc21672)

*[3.9.6. Test case của usecase Theo dõi đơn hàng](#_Toc22675)* [43](#_Toc22675)

**[3.10. Usecase 8: Quản lý đơn hàng 44](#_Toc11316)**

*[3.10.1. Mô hình MVC](#_Toc4953)* [44](#_Toc4953)

*[3.10.2. Usecase Diagram của Quản lý đơn hàng](#_Toc30526)* [44](#_Toc30526)

*[3.10.3. Scenario của usecase Quản lý đơn hàng](#_Toc320)* [44](#_Toc320)

*[3.10.4. Sequence Diagram của usecase Quản lý đơn hàng](#_Toc32021)* [45](#_Toc32021)

*[3.10.5. State Diagram của usecase Quản lý đơn hàng](#_Toc19256)* [46](#_Toc19256)

*[3.10.6. Test case của usecase Quản lý đơn hàng](#_Toc11865)* [46](#_Toc11865)

**[3.11. Usecase 9: Quản lý nhà hàng 47](#_Toc9479)**

*[3.11.1. Mô hình MVC](#_Toc1723)* [47](#_Toc1723)

*[3.11.2. Usecase Diagram của Quản lý nhà hàng](#_Toc24213)* [47](#_Toc24213)

*[3.11.3. Scenario của usecase Quản lý nhà hàng](#_Toc13715)* [47](#_Toc13715)

*[3.11.4. Sequence Diagram của usecase Quản lý nhà hàng](#_Toc21668)* [49](#_Toc21668)

*[3.11.5. State Diagram của usecase Quản lý nhà hàng](#_Toc3135)* [49](#_Toc3135)

*[3.11.6. Test case của usecase Quản lý nhà hàng](#_Toc25654)* [50](#_Toc25654)

**[3.12. Usecase 10: Yêu thích nhà hàng 52](#_Toc26732)**

*[3.12.1. Mô hình MVC](#_Toc24107)* [52](#_Toc24107)

*[3.12.2. Usecase Diagram của Yêu thích nhà hàng](#_Toc16823)* [52](#_Toc16823)

*[3.12.3. Scenario của usecase Yêu thích nhà hàng](#_Toc9448)* [52](#_Toc9448)

*[3.12.4. Sequence Diagram của usecase Yêu thích nhà hàng](#_Toc32073)* [53](#_Toc32073)

*[3.12.5. State Diagram của usecase Yêu thích nhà hàng](#_Toc11258)* [54](#_Toc11258)

*[3.12.6. Test case của usecase Yêu thích nhà hàng](#_Toc32602)* [54](#_Toc32602)

**[4. PHẦN 4: HƯỚNG DẪN SỬ DỤNG 56](#_Toc29117)**

**[5. PHẦN 5: KẾT LUẬN 62](#_Toc4075)**

[5.1. Đóng góp của đề tài 62](#_Toc12261)

[5.2. Hạn chế của đề tài 62](#_Toc2713)

[5.1. Hướng phát triển của đề tài 62](#_Toc3078)

# **PHẦN 1: GIỚI THIỆU**

**Tên đề tài:** Xây dựng ứng dụng App Đặt bàn  
**Tên ứng dụng:** App Đặt bàn

**Mục đích chính:** Phát triển một khả năng mới có thể truy cập trên ứng dụng đặt bàn nhà hàng của Nhà hàng UTE giúp dễ dàngđặt lịch hẹn trước khi đến nhà hàng, quán ăn; thông qua việc sử dụng app, khách hàng có thể tiết kiệm thời gian hơn, nhà hàng cũng linh hoạt hơn trong khâu chuẩn bị, từ đó tối ưu trải nghiệm người dùng, thu hút thêm khách hàng.

**Người dùng hướng đến:** Ứng dụng Đặt bàn nhà hàng sẽ được giới thiệu để hỗ trợ nhà hàng trong việc đặt lịch hẹn, và cũng đảm bảo không có lịch hẹn nào bị sót.

**Chức năng chính của ứng dụng:**

* Xem các trang: Đăng nhập, đăng ký, trang chủ, tìm kiếm, giỏ hàng, thông tin người dùng, các nhà hàng, yêu thích.
* Xem các chuyên mục trong ứng dụng
* Đăng nhập
* Đăng ký
* Đăng xuất
* Tra cứu
* Đánh giá nhà hàng
* Đặt bàn nhà hàng
* Theo dõi đơn hàng
* Quản lý đơn hàng
* Quản lý nhà hàng
* Yêu thích nhà hàng

**Đặt bàn:** Chọn nhà hàng sau đó chọn <Book Your Table>

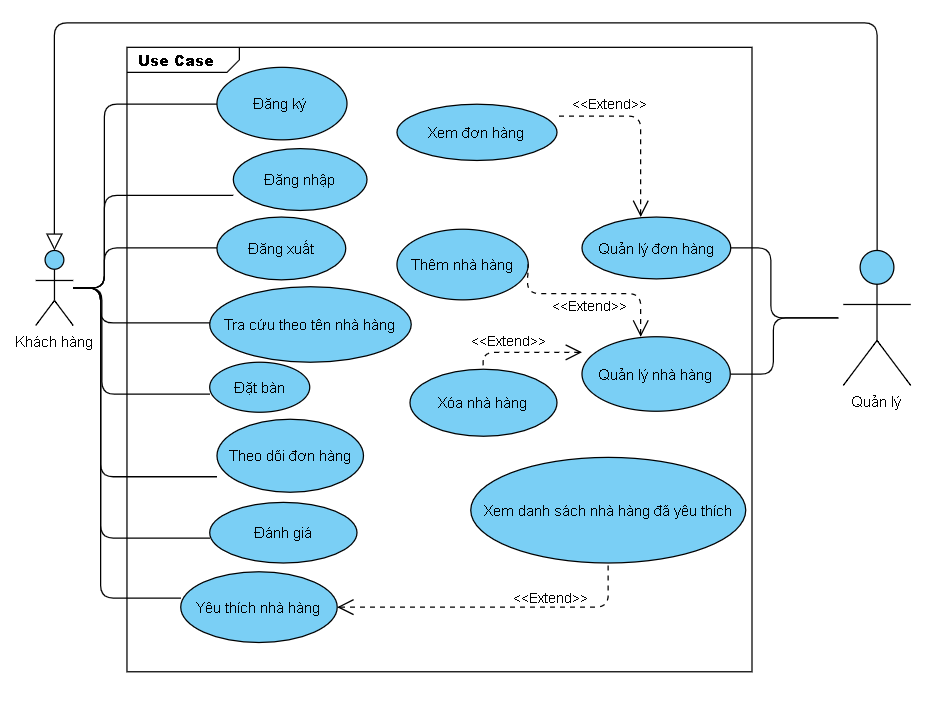
* Chọn nhà hàng muốn đến
* Xem giờ mở cửa của nhà hàng đó
* Xem danh sách món ăn
* Chọn vị trí dùng bữa
* Chọn số người
* Nhập số ghế
* Đánh giá nhà hàng đó
* Chọn đặt bàn
* Xem đơn đặt bàn trong giỏ hàng

# **PHẦN 2: REQUIREMENT DOCUMENT**

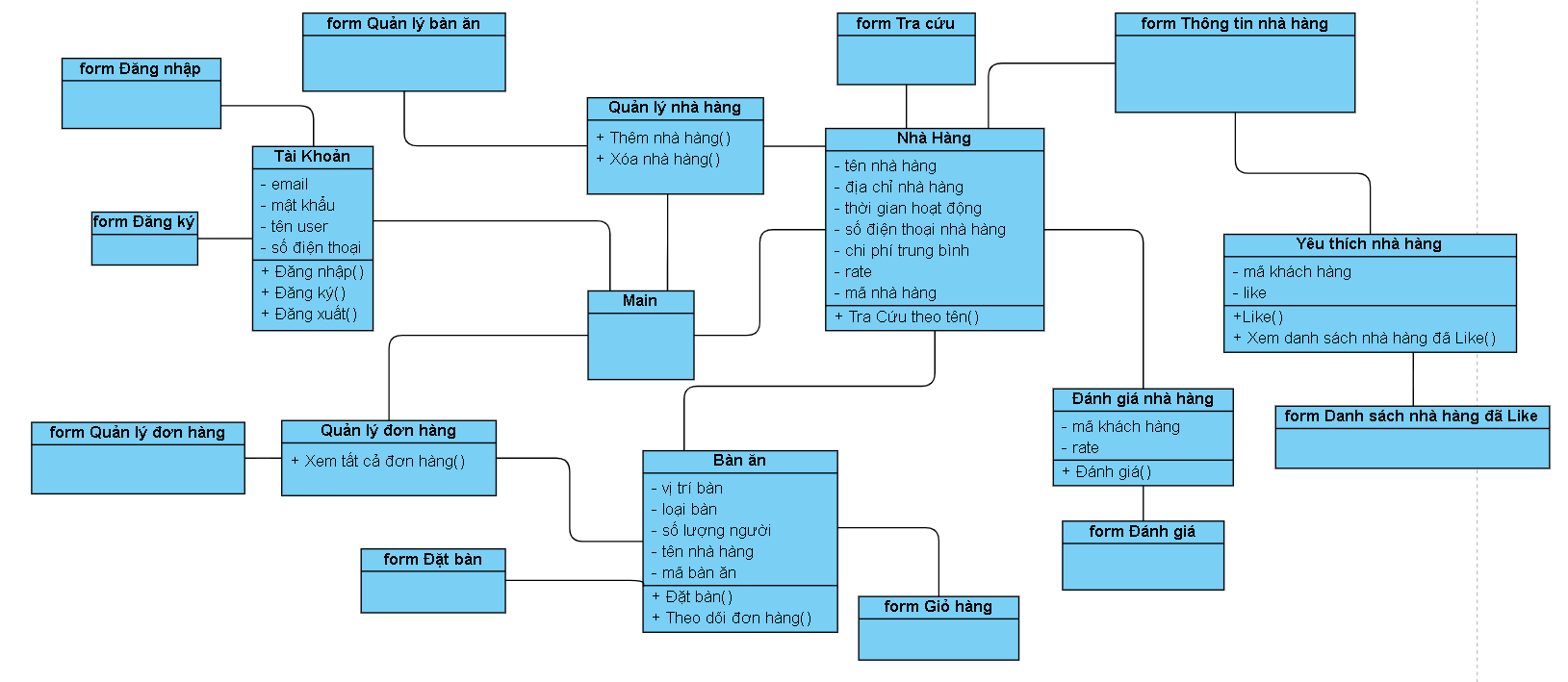
|  |  |
| --- | --- |
| **Chapter** | **Description** |
| **Preface** | \* Hướng dẫn đọc tài liệu   * Khách hàng và người dùng: * Hướng dẫn sử dụng hệ thống, các chức năng cơ bản của hệ thống, chức năng của người dùng, các quyền của người dùng. * Nhà phân tích: * Xem các đặc tả yêu cầu hệ thống. * Các mô hình Use case * Cấu trúc tổ chức và quản lý * Quy tắc nghiệp vụ * Nhà phát triển lập trình viên: * Dựa vào các mô hình xây dựng để tìm kiếm phần cần phát triển * Tối ưu các ràng buộc và đặc tả yêu cầu * Nhà kiểm thử * Kiểm tra các lỗi đã và đang gặp phải khi chưa có khả năng khắc phục dựa trên tài liệu   \* App đặt bàn nhà hàng trong tp.HCM   * Có các phiên bản được phát hành * Phiên bản thử nghiệm 1.0 * Phiên bản thử nghiệm lần hai 1.1 * Phiên bản thử nghiệm lần ba 2.0 * Phiên bản thử nghiệm lần bốn 3.0   \* Phiên bản thử nghiệm dùng tạm, mục đích để hiểu các yêu cầu của hệ thống. Phiên bản chính thực hiện tại 3.0, chưa có lịch sử thay đổi phiên bản tiếp theo. |
| **Introduction** | - Ứng dụng đặt bàn nhà hàng là một phần mềm trực tuyến có các chức năng phục vụ cho việc đặt bàn nhà hàng trên điện thoại di động, giải pháp hữu hiệu để khắc phục tình trạng lo ngại việc hết bàn trong ngày, giờ cao điểm.  - Phát triển một khả năng mới có thể truy cập trên ứng dụng đặt bàn nhà hàng của Nhà hàng giúp dễ dàng đặt lịch hẹn trước khi đến nhà hàng, quán ăn; thông qua việc sử dụng app, khách hàng có thể tiết kiệm thời gian hơn, nhà hàng cũng linh hoạt hơn trong khâu chuẩn bị, từ đó tối ưu trải nghiệm người dùng, thu hút thêm khách hàng.  - Ứng dụng Đặt bàn nhà hàng sẽ được giới thiệu để hỗ trợ nhà hàng trong việc đặt lịch hẹn, và cũng đảm bảo không có lịch hẹn nào bị sót. Ứng dụng chỉ hỗ trợ cho khách hàng và nhân viên của Nhà hàng; các đối tượng khác nằm ngoài phạm vi này không được hỗ trợ.  - Yêu cầu cần thiết người dùng phải đăng ký một tài khoản có username là gmail và password để đăng nhập vào hệ thống, khách hàng có thể tìm kiếm hoặc chọn nhà hàng để tìm những chỗ trống cần thiết, có thể review, đánh giá, quản lý có quyền thanh toán hoặc đổi món, cũng có thể thêm, sửa, xóa hay cập nhật thông tin nhà hàng và các chương trình khuyến mãi. |
| **Glossary** | Không có từ viết tắt |
| **User requirements definition** | - Dịch vụ được cung cấp cho khách hàng: khách hàng tìm kiếm được các nhà hàng, các loại bàn, danh sách món ăn, các chương trình khuyến mãi, được đánh giá sản phẩm, xem review sản phẩm để có thể so sánh được giá các sản phẩm ưu đãi nhất hay tích điểm nhận thưởng.  - Các yêu cầu chức năng của hệ thống:   * Đăng nhập: nếu đăng nhập thành công thì người dùng có thể truy cập vào app. Nếu đăng nhập sai thì sẽ được yêu cầu kiểm tra lại email và password. * Đăng ký: người dùng nhập thông tin vào form đăng ký, hệ thống sẽ kiểm tra xem tài khoản đã tồn tại hay chưa, nếu đã tồn tại thì yêu cầu người dùng nhập lại thông tin, nếu tài khoản chưa tồn tại thì hiện thông báo đăng ký thành công và yêu cầu đăng nhập lại để có thể truy cập app. * Đăng xuất: Ứng dụng sẽ hiện thông báo hỏi người dùng “Bạn có muốn thoát?” và nút “Yes” and“No”. * Đánh giá: Khách hàng có thể đánh giá 1 nhà hàng theo số sao. * Tra cứu món ăn, nhà hàng: người dùng nhập từ khóa liên quan tới món ăn, nhà hàng hoặc quận… * Đặt bàn: Khách hàng chọn nhà hàng, sau đó chọn loại bàn và số người đến nhà hàng. * Theo dõi đơn hàng: Khách hàng xem được các đơn hàng đã đặt bàn trước đó. * Quản lý đơn hàng: Admin có thể xem đơn hàng của khách hàng và hủy đơn đó nếu không hợp lệ. * Quản lý nhà hàng: Admin có thể thêm thông tin mới của nhà hàng vào trong chuỗi các nhà hàng của hệ thống.   - Các yêu cầu phi chức năng của hệ thống:   * Hệ thống phải liên tục cập nhật thông tin nếu chuỗi nhà hàng có những thay đổi. Ví dụ: mở thêm chi nhánh; nhà hàng thay đổi vị trí bàn, phòng ăn,...; nhà hàng thêm món ăn mới, hoặc xóa món ăn khỏi menu… * Băng thông mạng đủ (Sufficient network bandwidth) * Hiệu quả (tiêu thụ tài nguyên cho tải nhất định) * Yếu tố cảm xúc * Khả năng chịu lỗi (ví dụ: Giám sát, đo lường và quản lý hệ thống vận hành) * Tính linh hoạt (ví dụ: để đối phó với những thay đổi trong tương lai về yêu cầu) * Quốc tế hóa và bản địa hóa * Khả năng tương tác * Các vấn đề pháp lý và cấp phép hoặc khả năng tránh vi phạm bằng sáng chế * Khả năng bảo trì * Khả năng hoạt động * Hiệu suất / thời gian phản hồi (kỹ thuật hiệu suất) * Khả năng tương thích nền tảng * Quyền riêng tư (tuân thủ luật bảo mật) * Tính di động * Chất lượng (ví dụ: lỗi được phát hiện, lỗi đã phân phối, hiệu quả loại bỏ lỗi) * Độ tin cậy (ví dụ: Thời gian trung bình giữa / đến thất bại - MTBF / MTTF) * Thời gian đáp ứng * Khả năng mở rộng (ngang, dọc) * Khả năng tương thích phần mềm, công cụ, tiêu chuẩn, v.v. * Khả năng kiểm tra * Âm lượng * Người dùng ứng dụng: sử dụng username và password và phân quyền của mình để thực thi các thao tác mong muốn. * Password của người dùng được bảo mật bởi cơ sở dữ liệu Firebase Realtime Database và ngay cả người làm ra hệ thống cũng không thể biết được. |
| **System architecture** | Tính năng định vị nhà hàng gần đây nhằm gợi ý tới khách hàng vị trí nhà hàng gần nhất, có bản đồ chỉ dẫn đường đi dễ dàng.  Khách hàng truy cập vào app đặt bàn sau đó đăng nhập thông tin cá nhân của họ bằng email và password. Bằng tính năng đăng nhập thông tin, khách hàng có thể nhìn thấy thông tin đặt bàn, những ưu đãi họ được nhận và tham gia các chương trình khuyến mãi của nhà hàng.  Khách hàng sẽ được giữ chỗ miễn phí nhưng vẫn được tặng kèm ưu đãi, với các lợi ích tương tự như deal, voucher, mã giảm giá hấp dẫn… |
| **System requirement specification** | - Yêu cầu chức năng:   * Đăng nhập: đăng nhập vào hệ thống bằng email và password sau khi đăng ký tài khoản. * Đăng ký: người dùng đăng ký tài khoản để đăng nhập vào ứng dụng. * Đăng xuất: đăng xuất thoát khỏi hệ thống * Tra cứu nhà hàng: List thành phố, quận; hiển thị lịch, giờ, list vị trí bàn, vị trí phòng * Đánh giá: Hiện lên cửa sổ nhỏ để khách hàng chọn sao đánh giá * Đặt bàn: List các nhà hàng, list món ăn, list loại bàn, list số lượng người đến nhà hàng. * Theo dõi đơn hàng: List các đơn đặt bàn mà khách hàng đã đặt trước đó. * Quản lý đơn hàng: người dùng có thể xem được đơn đặt hàng, hủy đơn. * Quản lý nhà hàng: người dùng có thể thêm nhà hàng mới, hiện lên trang mới yêu cầu thêm các thông tin cần thiết của một nhà hàng.   - Yêu cầu phi chức năng: Hệ thống phải liên tục cập nhật thông tin nếu chuỗi nhà hàng có những thay đổi, băng thông mạng đủ, lưu trữ dữ liệu, yếu tố cảm xúc, khả năng mở rộng, tính linh hoạt, khả năng tương tác, khả năng bảo trì, mã nguồn mở,.... |
| **System models** | Mô hình thể hiện mối quan hệ giữa hệ thống và môi trường của nó. |
| **System evolution** | Nâng cấp và bảo trì hệ thống.  Nghiên cứu về xu hướng khách hàng trên thị trường để nắm bắt toàn bộ thông tin thị trường và tâm lý khách hàng để nâng cấp hệ thống phù hợp với thị trường.  Dự đoán khách hàng sẽ đưa ra những thay đổi như:   * Giao diện * Chi phí * Thay đổi dữ liệu * Cập nhật thêm tính năng * Phiên bản của ứng dụng |
| Appendices | Ngôn ngữ java. Phiên bản android studio 4.1.3 for Windows |
| Index |  |

# **PHẦN 3: ĐẶC TẢ, PHÂN TÍCH VÀ THIẾT KẾ**

## **Thiết kế use case tổng quan hệ thống**

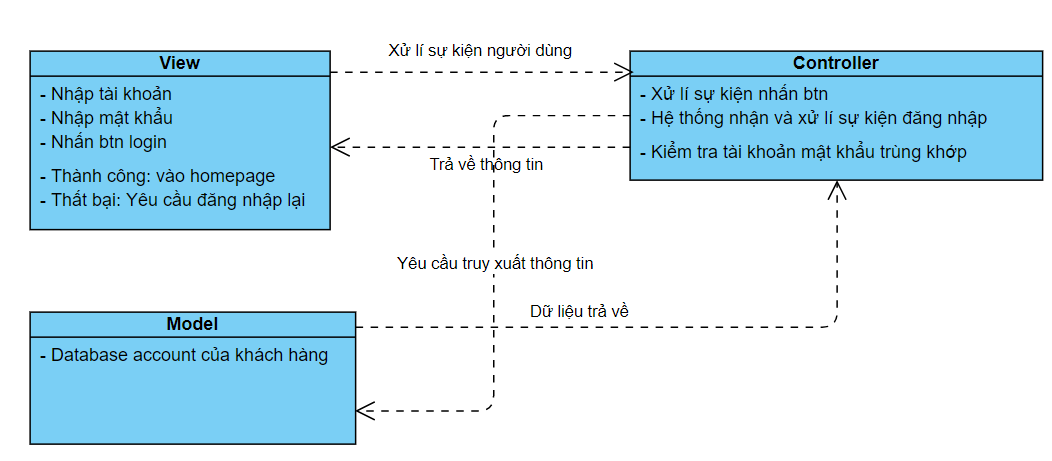


## **Class Diagrams**

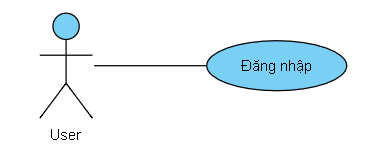


## **Usecase 1: Đăng nhập**

### ***Mô hình MVC***



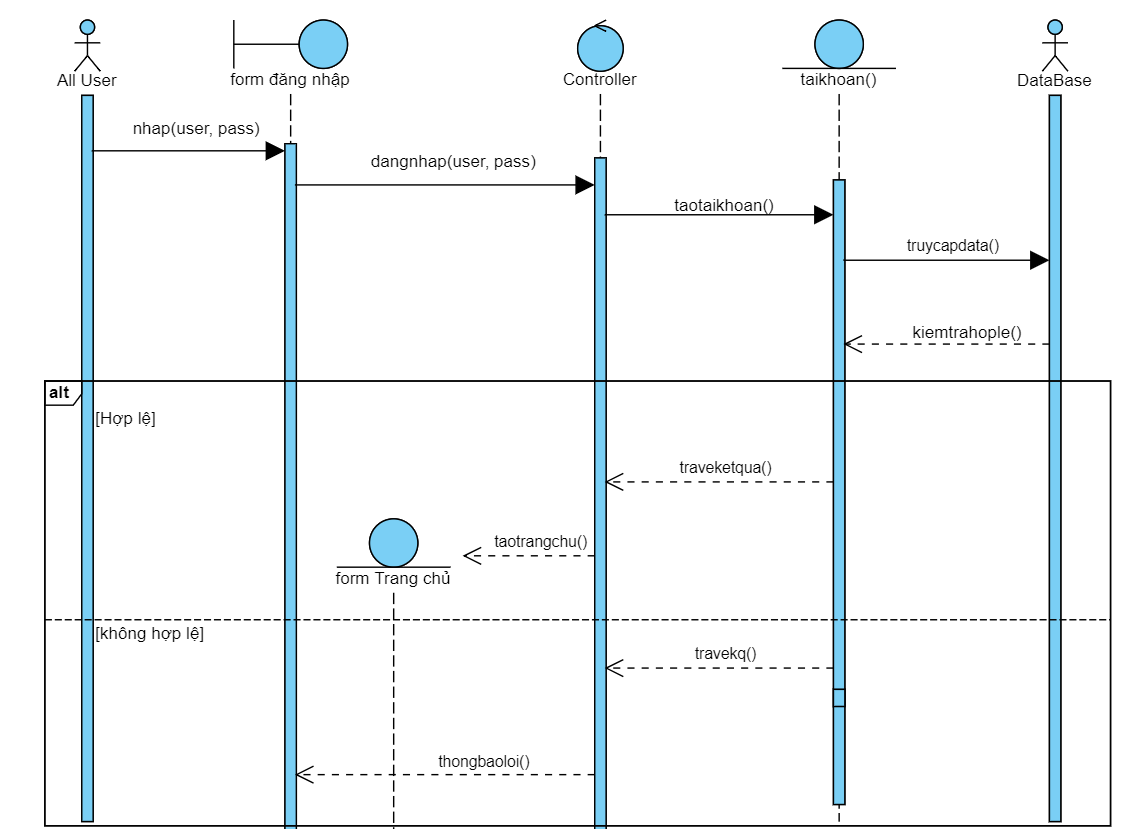
### ***Usecase Diagram của usecase Đăng nhập***



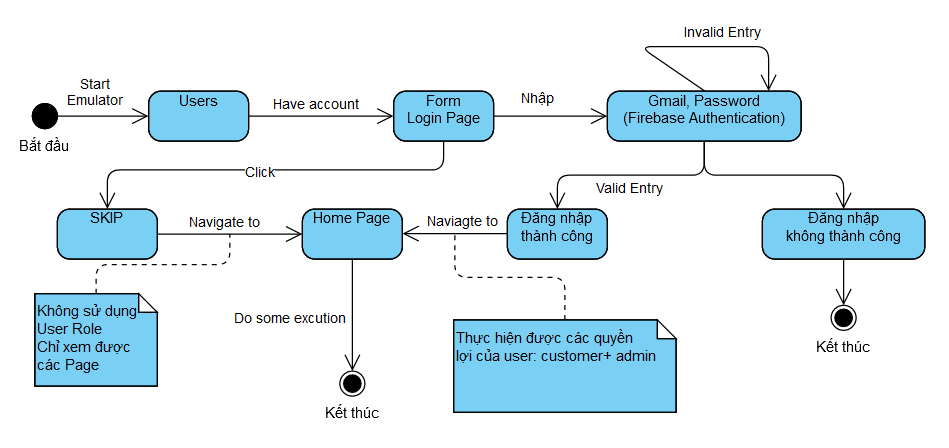
### ***Scenario của usecase Đăng nhập***

|  |  |
| --- | --- |
| **Name** | **Đăng nhập hệ thống** |
| **Brief description** | Người dùng đăng nhập vào hệ thống |
| **Actor(s)** | Khách hàng, Admin |
| **Pre-conditions** | Người dùng đã được tạo tài khoản trong hệ thống |
| **Post-conditions** | Nếu đăng nhập thành công: Người dùng được xác thực và hệ thống hiển thị nội dung trang chủ.  Nếu đăng nhập không thành công: Người dùng không truy cập được và hệ thống |
| **Flow of events** |  |
| Basic flow  (Thành công) | Use case bắt đầu khi người dùng cần thực hiện chức năng nào đó của hệ thống cần xác thực quyền truy cập  1. Hệ thống hiển thị cửa sổ cho người dùng nhập email và password  2. Người dùng nhập email và password  3. Hệ thống xác thực email và password  4. Hệ thống xác định vai trò của người dùng  5. Hệ thống mở màn hình trang chủ và mở các chức năng được phép truy cập của người dùng. |
| Alternative flow  (Thất bại) | Nếu người dùng nhập sai email hay/và password. Những công việc sau được thực hiện:  1. Hệ thống mô tả lý do xác thực bị sai  2. Hệ thống lưu ý người dùng đăng nhập lại  3. Khi người dùng nhập lại email và password, bước 3 ở Basic flow được thực hiện. |
| **Extension point** | Không có |

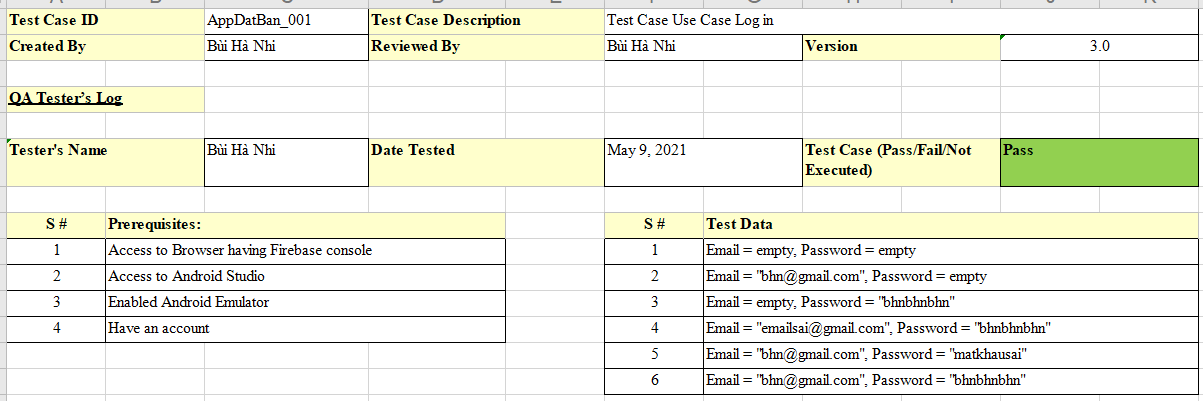
### ***Sequence Diagram của usecase Đăng nhập***

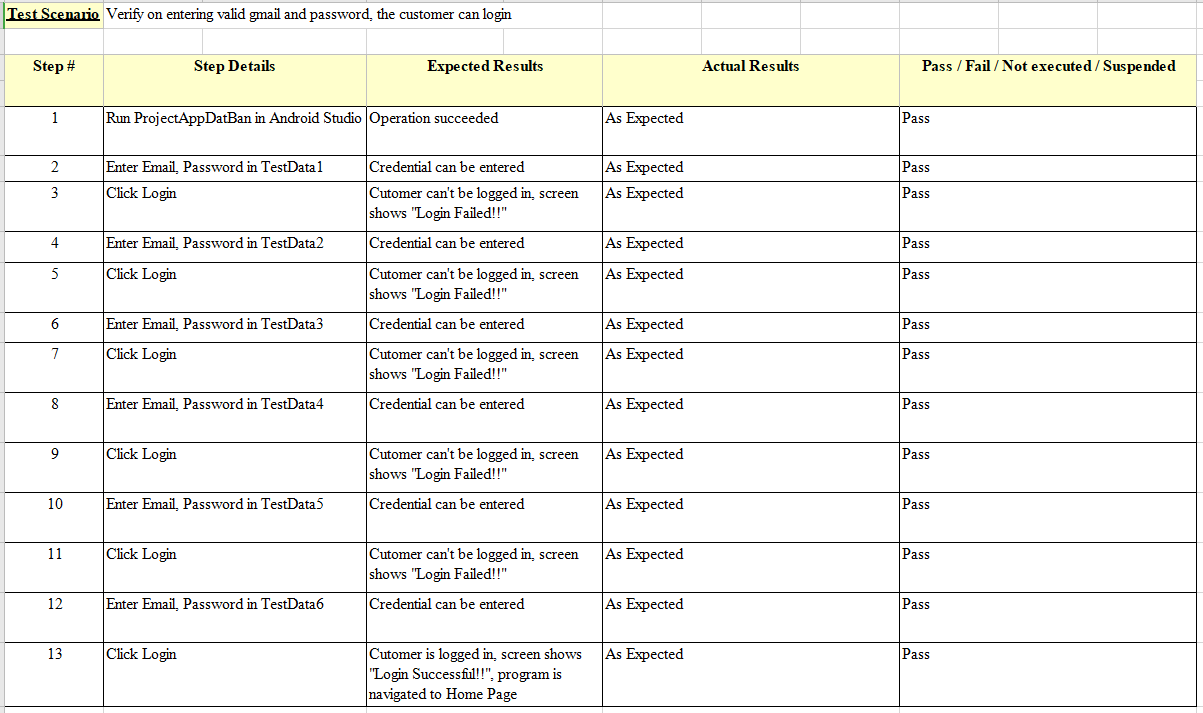


### ***State Diagram của usecase Đăng nhập***



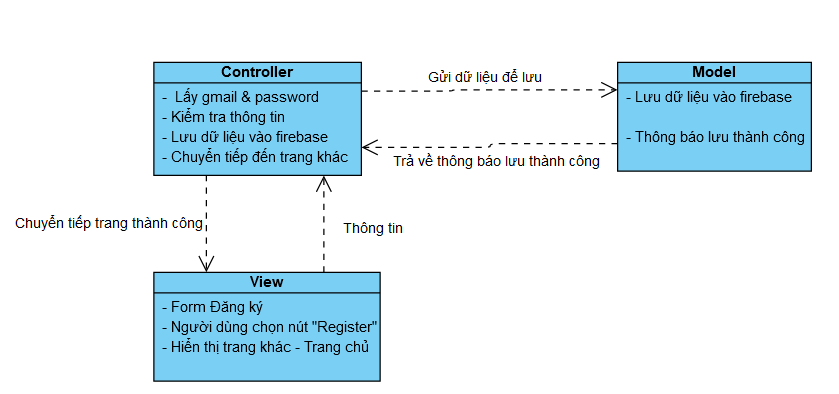
### ***Test case của usecase Đăng nhập***



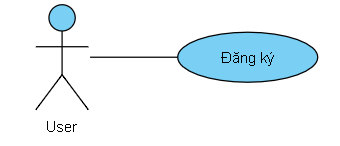


## **Usecase 2: Đăng ký**

### ***Mô hình MVC***



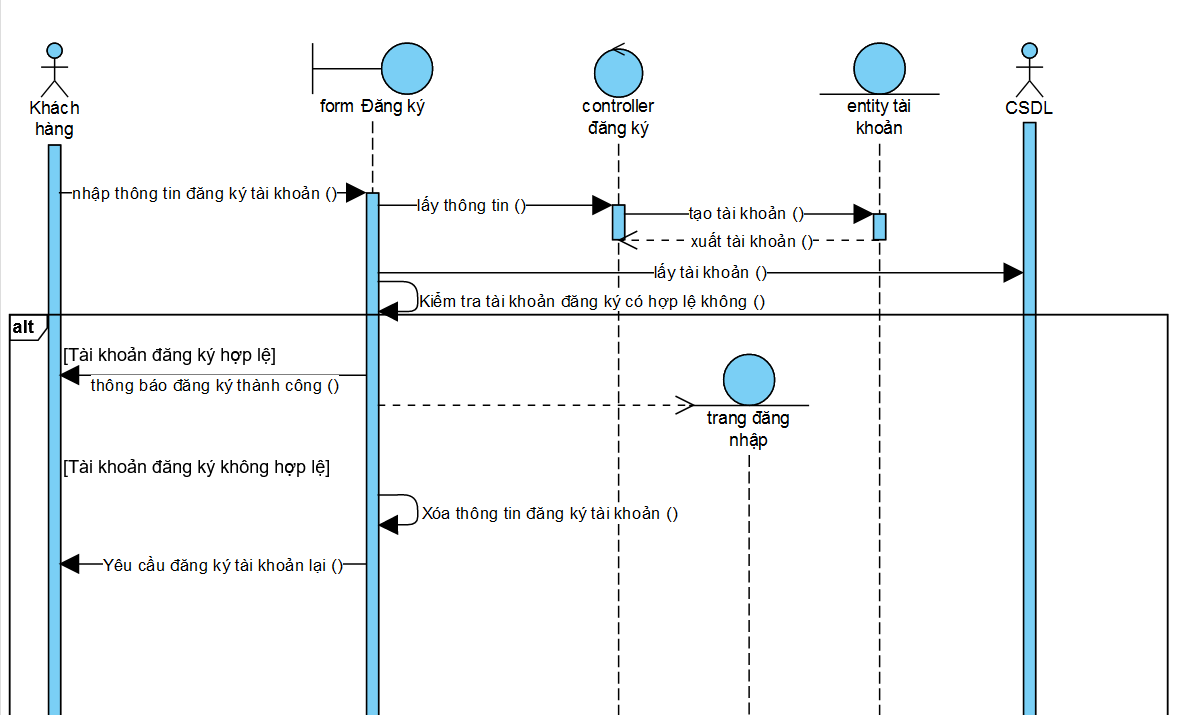
### ***Usecase Diagram của Đăng ký***



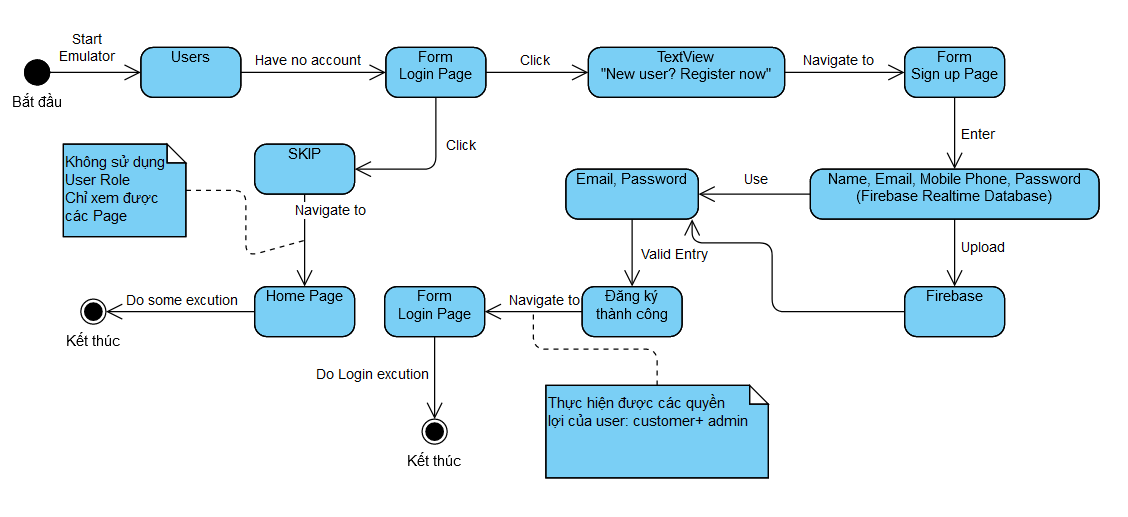
### ***Scenario của usecase Đăng ký***

|  |  |
| --- | --- |
| **Name** | **Đăng ký hệ thống** |
| **Brief description** | Người dùng đăng ký vào hệ thống |
| **Actor(s)** | Khách hàng, Admin |
| **Pre-conditions** | Không có |
| **Post-conditions** | Nếu đăng ký thành công: Hiển thị thông báo thành công.  Nếu đăng ký không thành công: Hiển thị thông báo lỗi. |
| **Flow of events** |  |
| Basic flow  (Thành công) | Use case bắt đầu khi người dùng cần thực hiện chức năng nào đó của hệ thống cần xác thực quyền truy cập  1. Hệ thống hiển thị cửa sổ cho người dùng đăng nhập  2. Người dùng chọn mục Đăng ký tài khoản  3. Hệ thống hiển thị form đăng ký tài khoản  4. Người dùng điền các thông tin cần thiết để đăng ký tài khoản  5. Người dùng chọn nút Đăng ký tài khoản  6. Hệ thống hiển thị thông báo đăng ký tài khoản thành công và chuyển về trang đăng nhập |
| Alternative flow  (Thất bại) | Nếu người dùng nhập username trùng với username của người dùng khác. Những công việc sau được thực hiện:  1. Hệ thống hiển thị “Registration failed! Please try again later”  2. Khi người dùng nhập username khác và các thông tin cần thiết, bước 5 ở Basic flow được thực hiện. |
| **Extension point** | Không có |

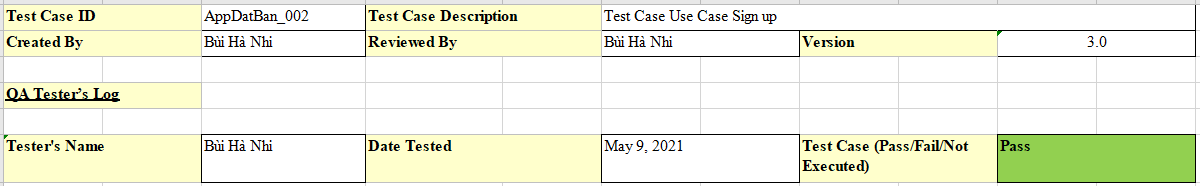
### ***Sequence Diagram của usecase Đăng ký***

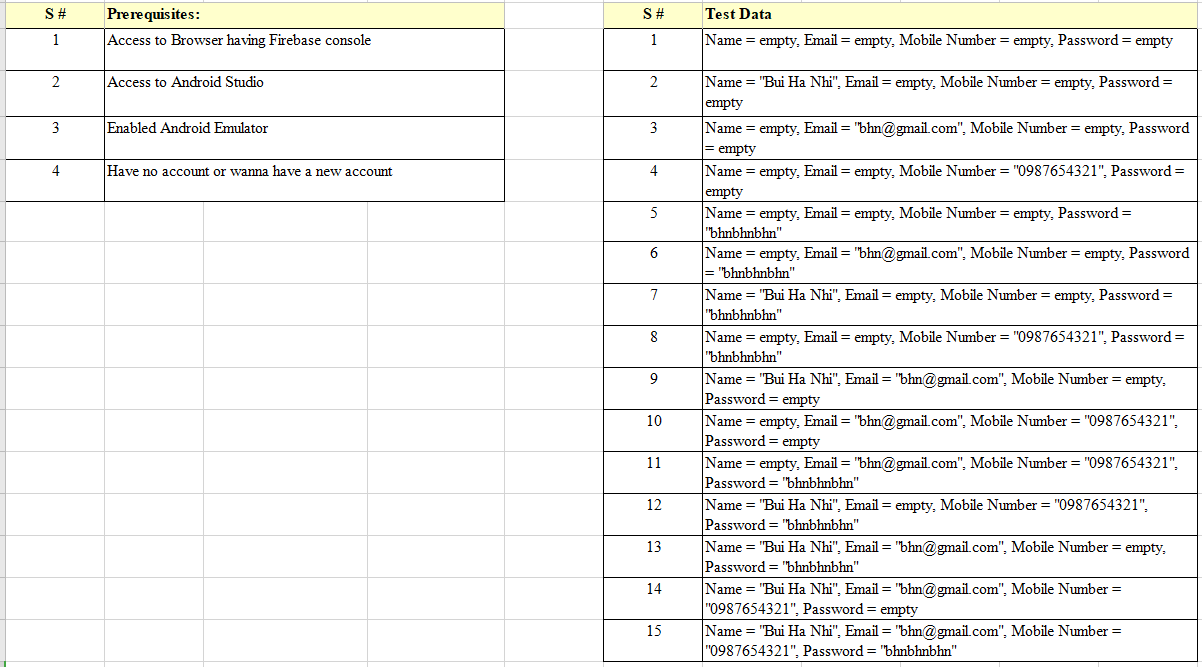


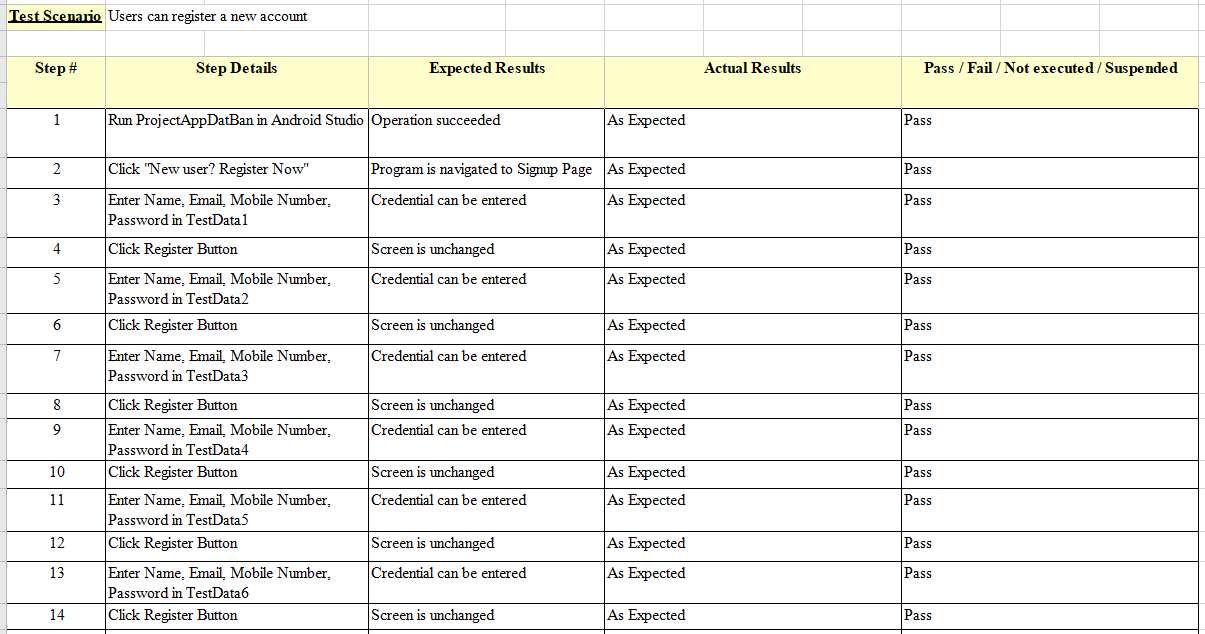
### ***State Diagram của usecase Đăng ký***

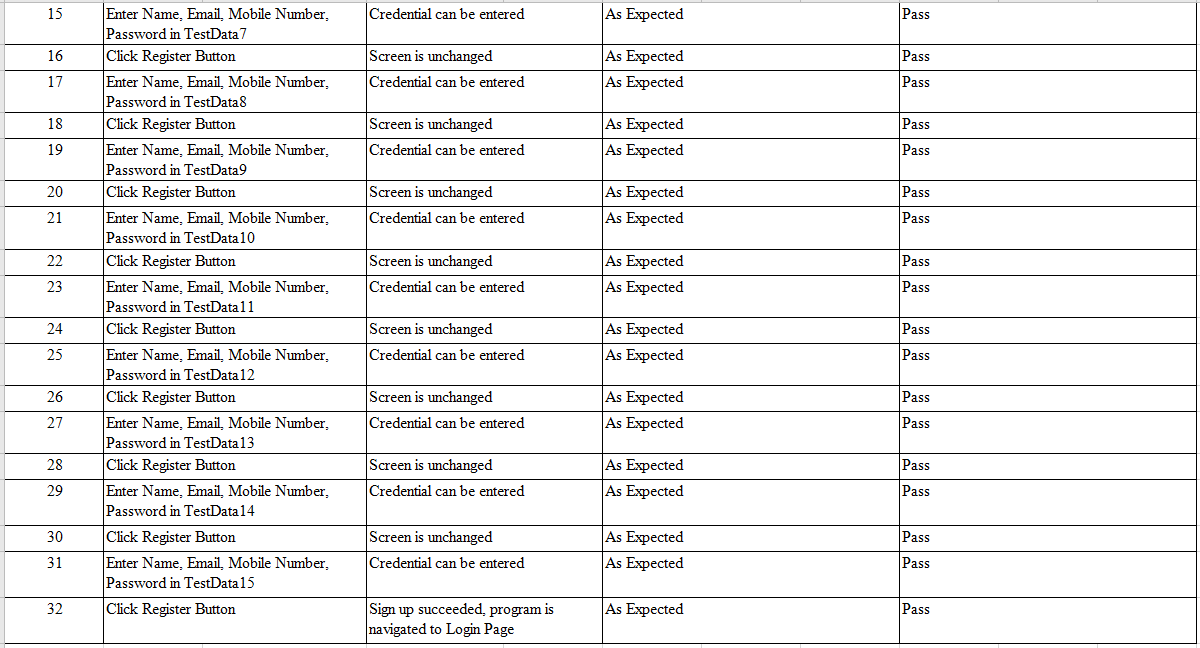


### ***Test case của usecase Đăng ký***



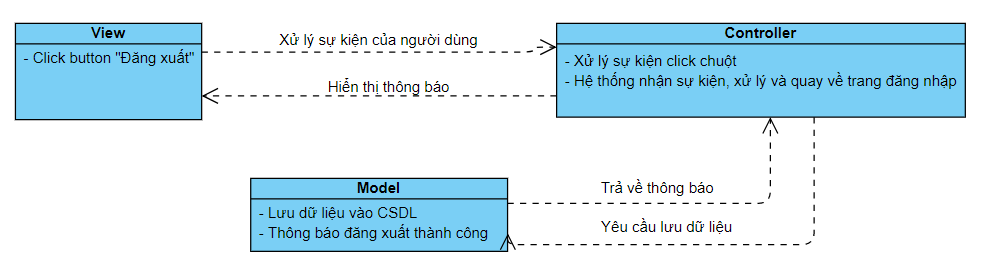




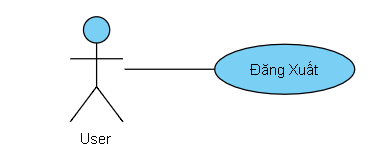


## **Usecase 3: Đăng xuất**

### ***Mô hình MVC***



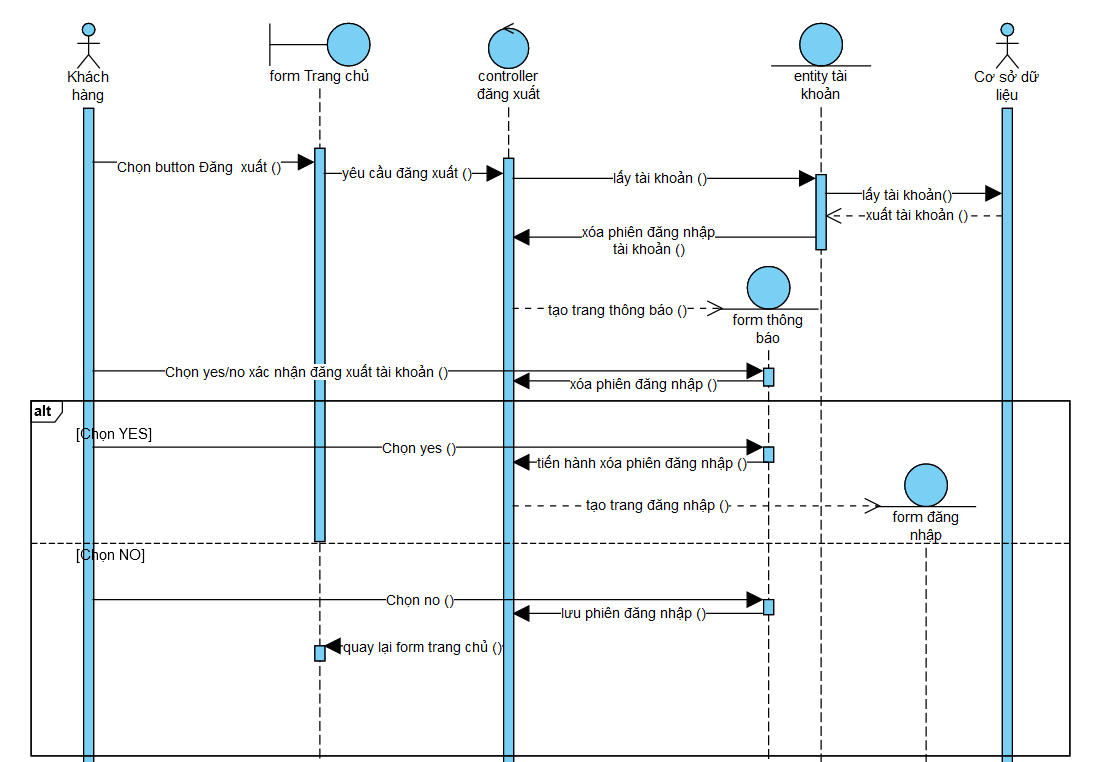
### ***Usecase Diagram của Đăng xuất***



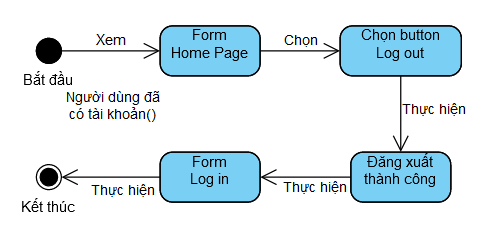
### ***Scenario của usecase Đăng xuất***

|  |  |
| --- | --- |
| **Name** | **Đăng xuất hệ thống** |
| **Brief description** | Người dùng đăng xuất khỏi hệ thống |
| **Actor(s)** | Khách hàng, Người quản lý |
| **Pre-conditions** | Người dùng đang trong phiên đăng nhập |
| **Post-conditions** | Nếu đăng xuất thành công: Hệ thống hiển thị form Đăng nhập.  Nếu đăng xuất không thành công: Hệ thống hiển thị thông báo lỗi. |
| **Flow of events** |  |
| Basic flow  (Thành công) | Use case bắt đầu khi người dùng cần thực hiện chức năng đăng xuất  1. Hệ thống hiển thị cửa sổ cho người dùng đăng xuất khỏi hệ thống  2. Người dùng nhấn nút đăng xuất  3. Hệ thống hiện thông báo xác nhận muốn thoát chương trình hiện ra  4. Người dùng nhấn nút xác định  5. Form chức năng chính đóng lại đồng thời form Đăng nhập hiện ra |
| Alternative flow  (Thất bại) | Không có |
| **Extension point** | Không có |

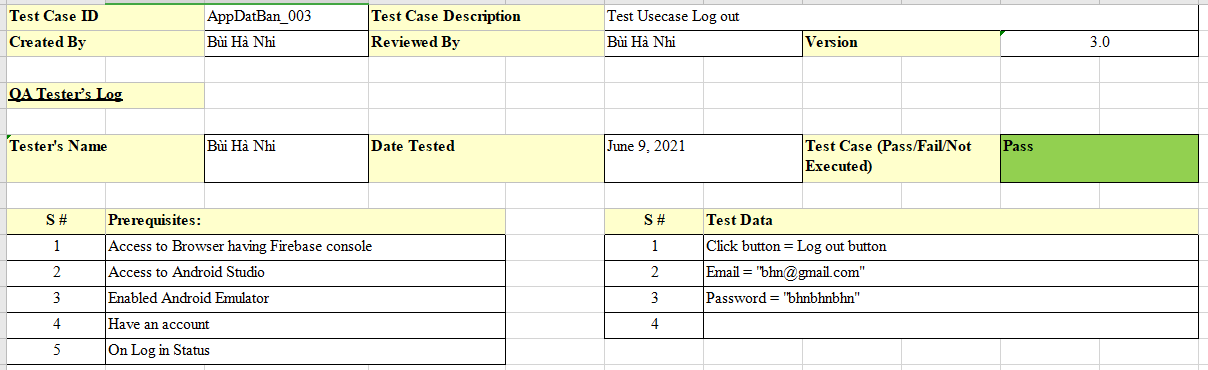
### ***Sequence Diagram của usecase Đăng xuất***

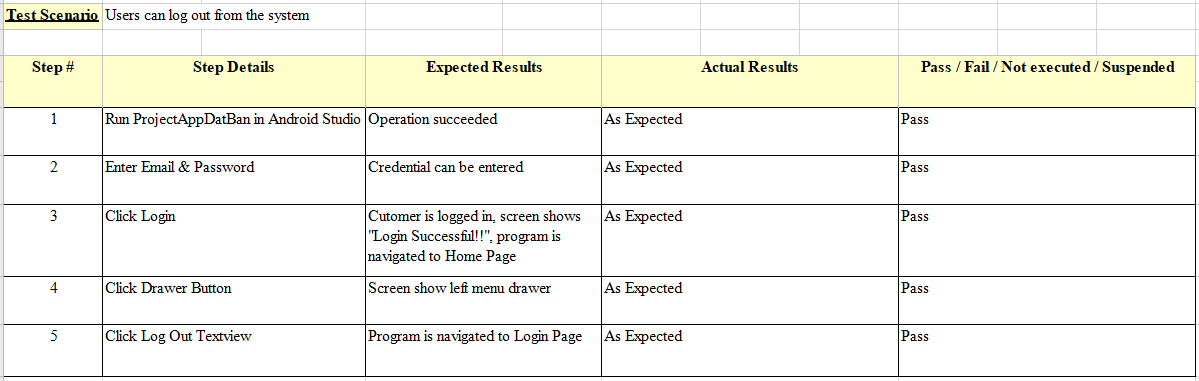


### ***State Diagram của usecase Đăng xuất***



### ***Test case của usecase Đăng xuất***

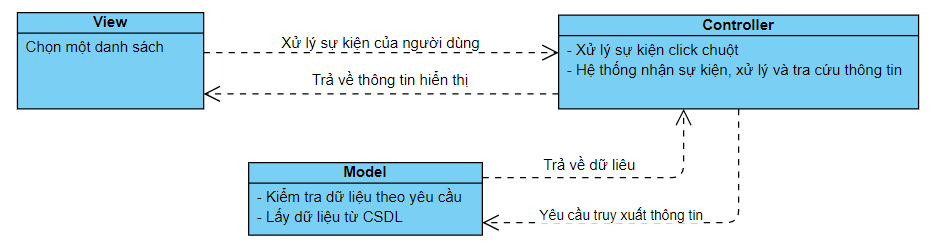




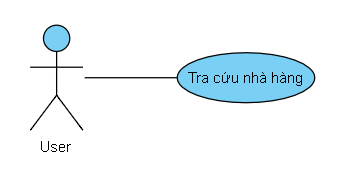
## **Usecase 4: Tra cứu nhà hàng**

### ***Mô hình MVC***

#### 



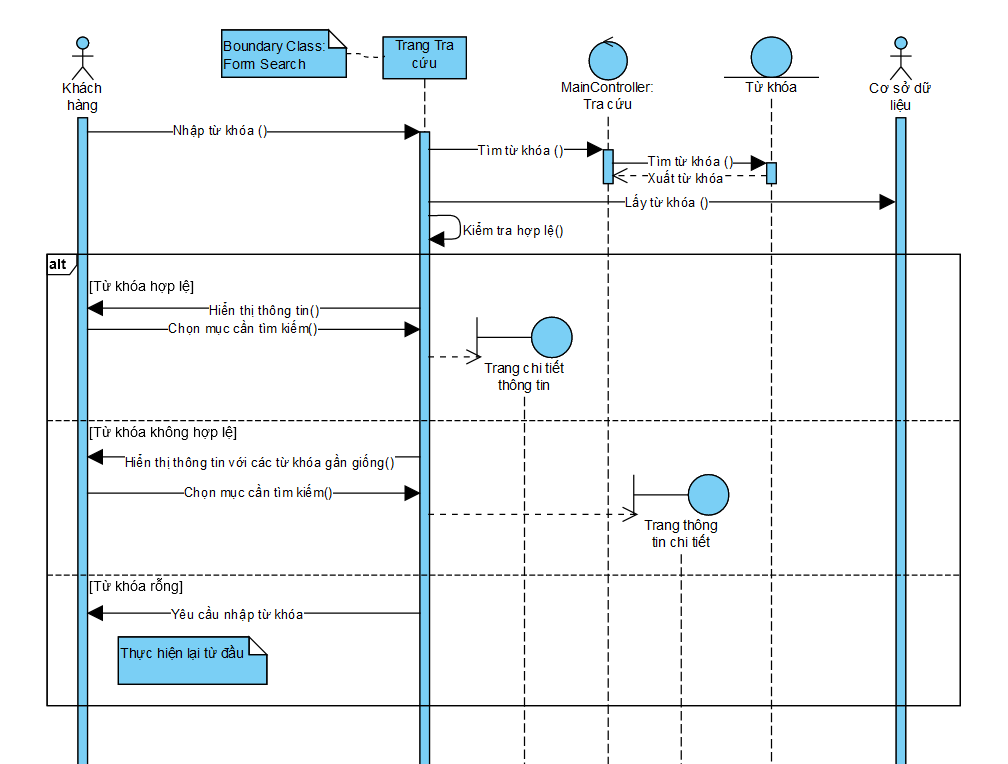
### ***Usecase Diagram của Tra cứu nhà hàng***



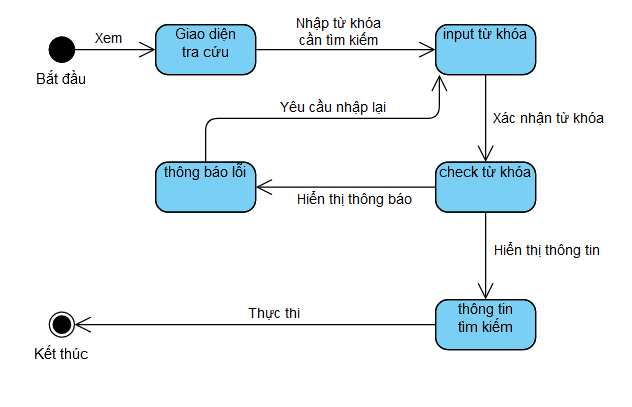
### ***Scenario của usecase Tra cứu nhà hàng***

|  |  |
| --- | --- |
| **Name** | **Tra cứu theo từ khóa** |
| **Brief description** | Người dùng tra cứu theo tên nhà hàng |
| **Actor(s)** | Khách hàng |
| **Pre-conditions** | Tên nhà hàng đã có trong hệ thống |
| **Post-conditions** | Nếu tra cứu thành công: Hệ thống trả về cho người dùng cardview chứa thông tin về nhà hàng theo từ khóa tra cứu  Nếu tra cứu không thành công: Giao diện trống và người dùng nhập lại |
| **Flow of events** |  |
| Basic flow  (Thành công) | Use case bắt đầu khi người dùng chọn chức năng tra cứu.  1. Hệ thống hiển thị cửa sổ cho người dùng nhập từ khóa về  2. Người dùng nhập từ khóa theo tên nhà hàng  3. Hệ thống hiển thị card nhà hàng chứa thông tin chi nhánh nhà hàng như: địa chỉ, đáng giá, menu, book a table... |
| Alternative flow  (Thất bại) | Nếu người dùng nhập sai từ khóa. Những công việc sau được thực hiện:  1. Hệ thống lưu ý người dùng nhập lại.  2. Khi người dùng nhập lại từ khóa tên nhà hàng bước 3 ở Basic flow được thực hiện. |
| **Extension point** | Không có |

### ***Sequence Diagram của usecase Tra cứu nhà hàng***

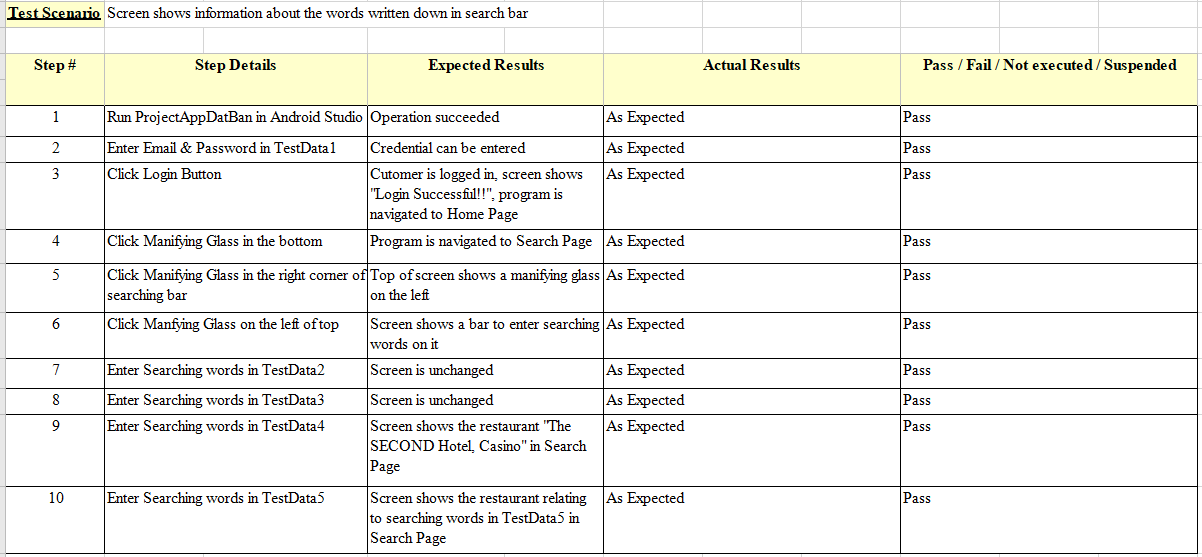


### ***State Diagram của usecase Tra cứu nhà hàng***



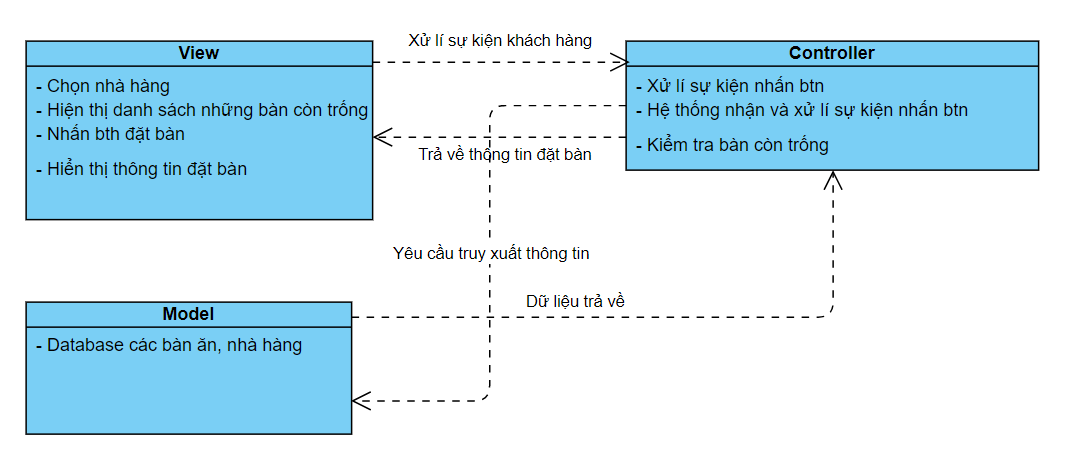
### ***Test case của usecase Tra cứu nhà hàng***

#### 

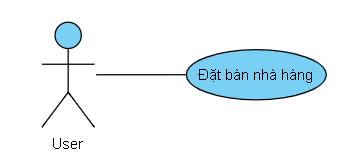


## **Usecase 5: Đặt bàn**

### ***Mô hình MVC***



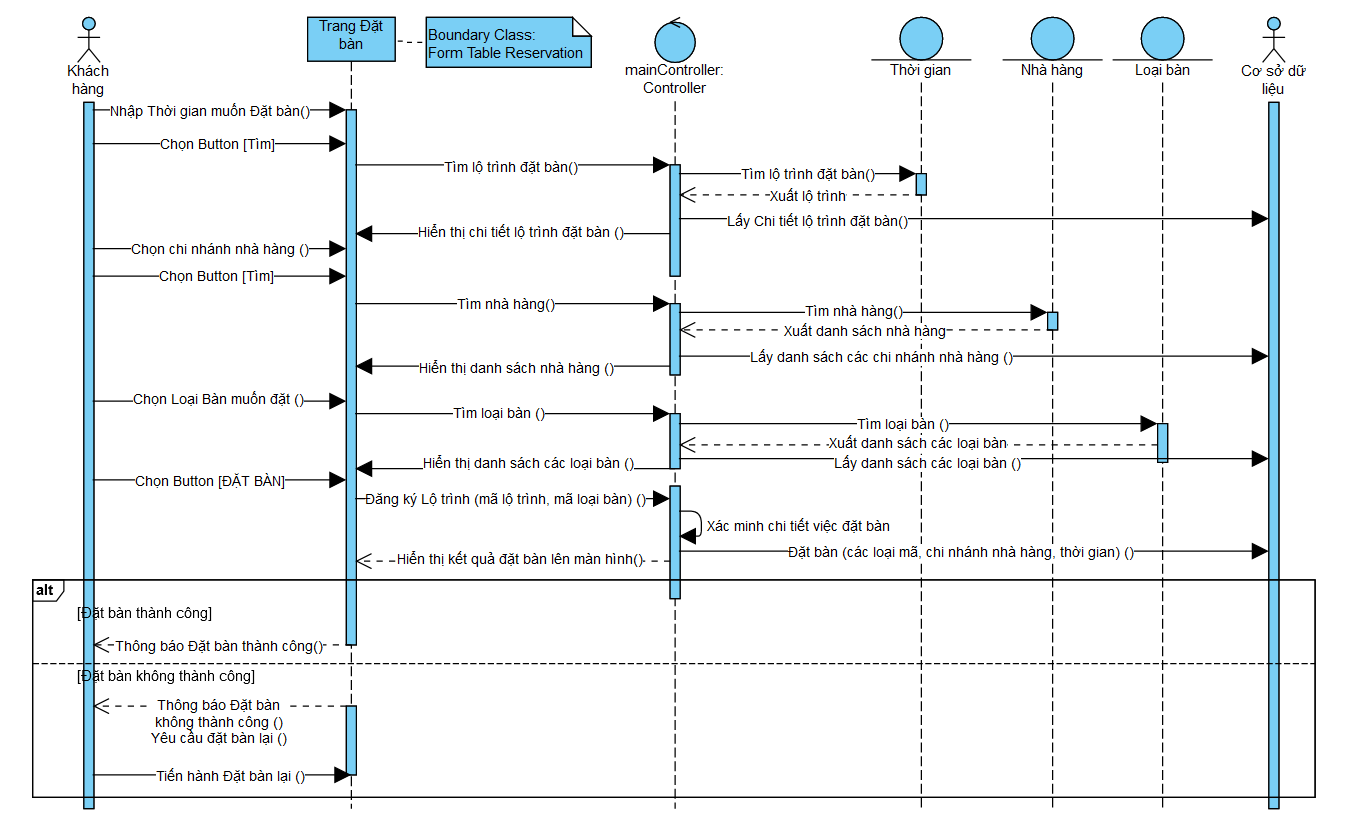
### ***Usecase Diagram của Đặt bàn nhà hàng***



### ***Scenario của usecase Đặt bàn nhà hàng***

|  |  |
| --- | --- |
| **Name** | **Đặt bàn** |
| **Brief description** | Người dùng đặt bàn trong hệ thống |
| **Actor(s)** | Khách hàng |
| **Pre-conditions** | Địa chỉ chi nhánh nhà hàng, tên nhà hàng đã có trong hệ thống |
| **Post-conditions** | Nếu đặt bàn thành công: Hệ thống trả về cho người dùng các thông tin về nhà hàng đã đặt |
| **Flow of events** |  |
| Basic flow  (Thành công) | Use case bắt đầu khi người dùng chọn chức năng đặt bàn.  1. Hệ thống hiển thị chi tiết nhà hàng cho người dùng đặt bàn  2. Người dùng chọn nhà hàng  3. Hệ thống hiển thị tên, chi nhánh nhà hàng, đánh giá, giờ mở cửa, menu,… để người dùng đặt bàn  4. Người dùng tiến hành đặt bàn  5. Người dùng nhấn nút “Book Your Table” để đặt bàn  6. Người dùng chọn nơi để dùng bữa, chọn số người đến nhà hàng, nhập số ghế.  7. Người dùng nhấn nút “Submit”  8. Hệ thống xác thực tên nhà hàng  9. Hệ thống hiển thị thông tin về tên nhà hàng, nơi ngồi, số người |
| Alternative flow  (Thất bại) | Nếu người dùng chọn sai hoặc chọn những chức năng chưa hoàn thiện. Những công việc sau được thực hiện:  1. Hệ thống mô tả chi nhánh nhà hàng, ưu đãi, bàn ăn... không tồn tại  2. Hệ thống lưu ý người dùng tra cứu lại  3. Khi người dùng nhập lại mã hoặc tên chi nhánh nhà hàng, ưu đãi, bàn ăn..., bước 3 ở Basic flow được thực hiện. |
| **Extension point** | Không có |

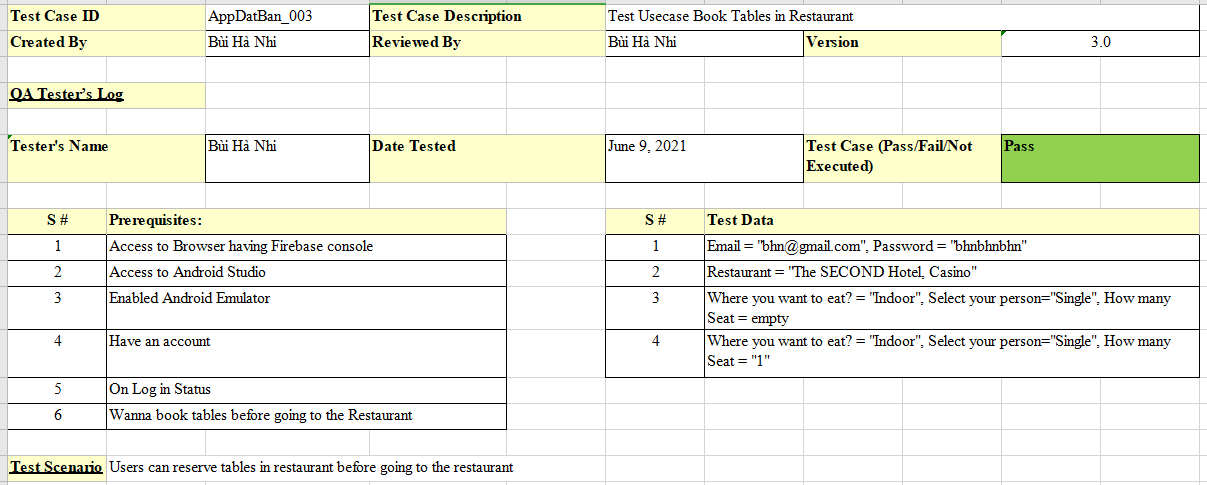
### ***Sequence Diagram của usecase Đặt bàn nhà hàng***

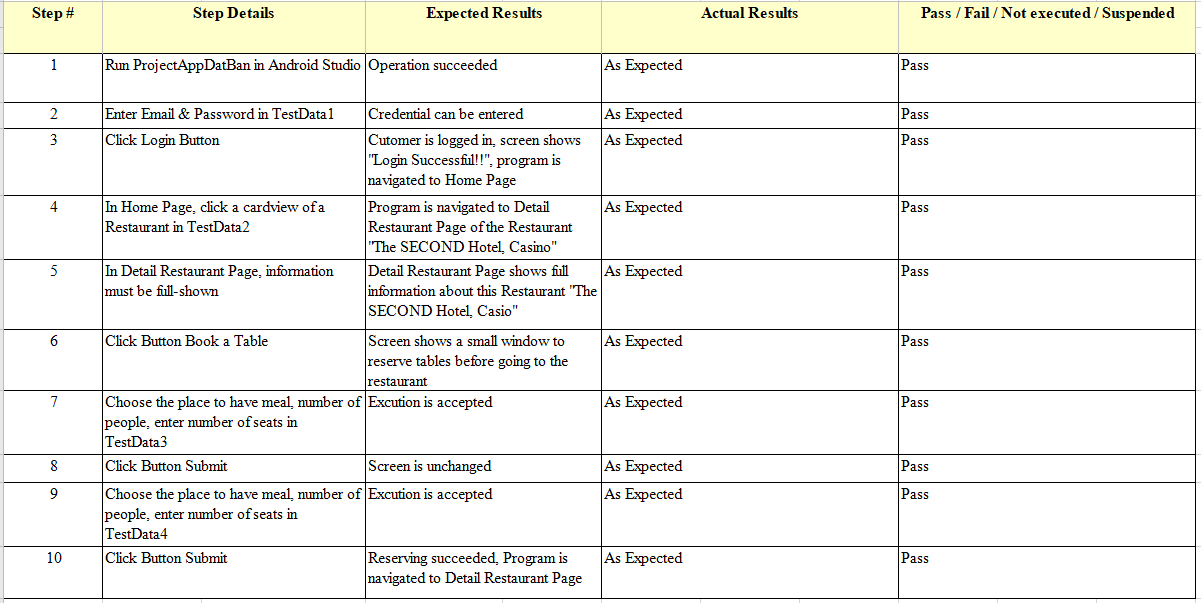


### ***State Diagram của usecase Đặt bàn nhà hàng***

#### 

### ***Test case của usecase Đặt bàn nhà hàng***

****

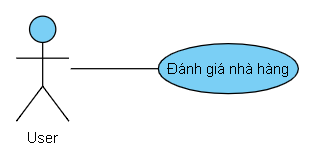
****

## **Usecase 6: Đánh giá**

### ***Mô hình MVC***

## 

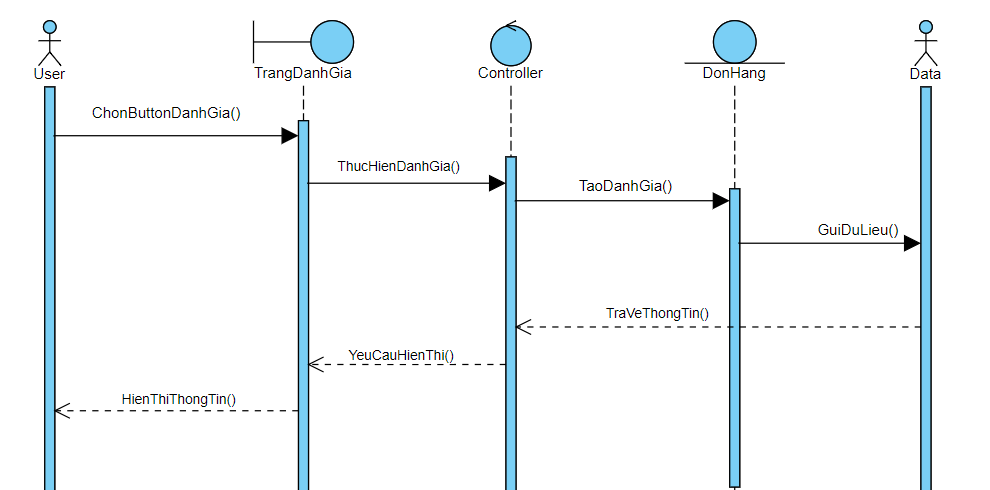
### ***Usecase Diagram của Đánh giá nhà hàng***



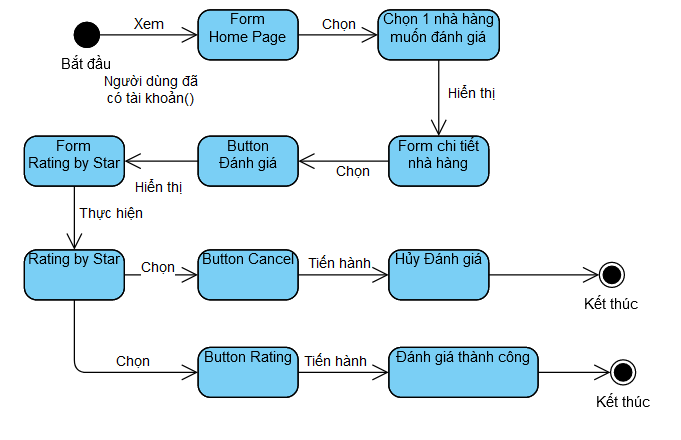
### ***Scenario của usecase Đánh giá nhà hàng***

|  |  |
| --- | --- |
| **Name** | **Đánh giá** |
| **Brief description** | Người dùng để lại đánh giá cho nhà hàng trong hệ thống |
| **Actor(s)** | Khách hàng |
| **Pre-conditions** | Người dùng đang đăng nhập trên phiên bản hiện tại |
| **Post-conditions** | Nếu đánh giá thành công: Hệ thống hiển thị đánh giá của người dùng |
| **Flow of events**  Basic flow  (Thành công) | Use case bắt đầu khi người dùng chọn chức năng đánh giá.  1. Hệ thống hiển thị cửa sổ cho người dùng đánh giá.  2. Người dùng tiến hành đánh giá cho nhà hàng.  3. Người dùng chọn Rate now để đăng tải đánh giá của mình  4. Hệ thống hiển thị đánh giá nhà hàng của khách hàng và các khách hàng khác |
| Alternative flow  (Thất bại) | Không có |
| **Extension point** | Không có |

### ***Sequence Diagram của usecase Đánh giá nhà hàng***

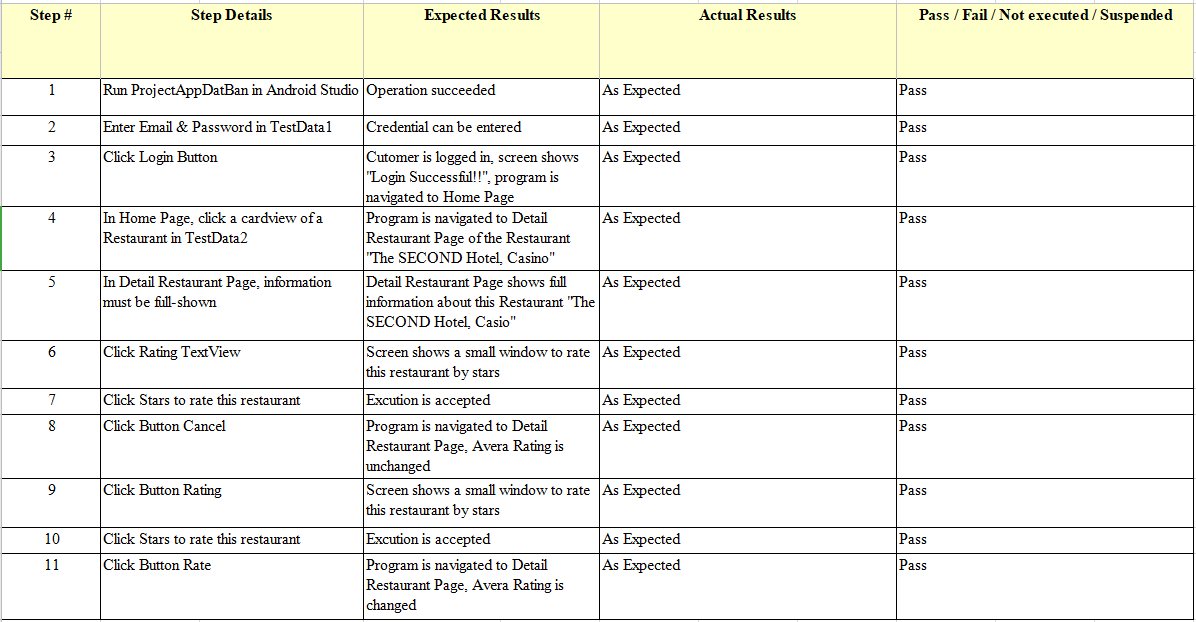


### ***State Diagram của usecase Đánh giá nhà hàng***



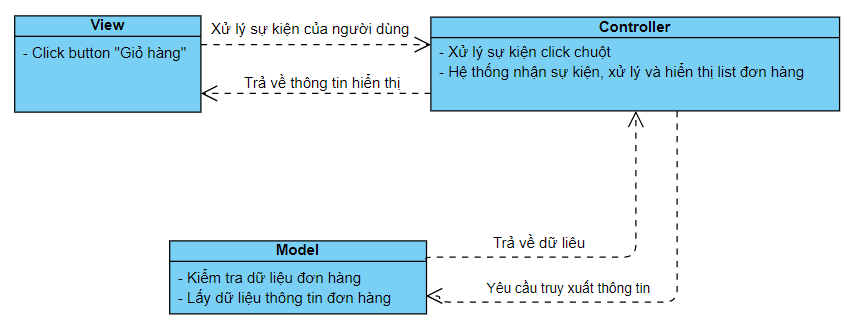
### ***Test case của usecase Đánh giá nhà hàng***

#### 

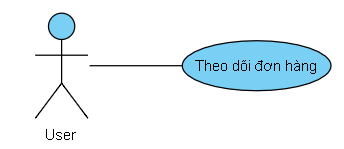


## **Usecase 7: Theo dõi đơn hàng**

### ***Mô hình MVC***



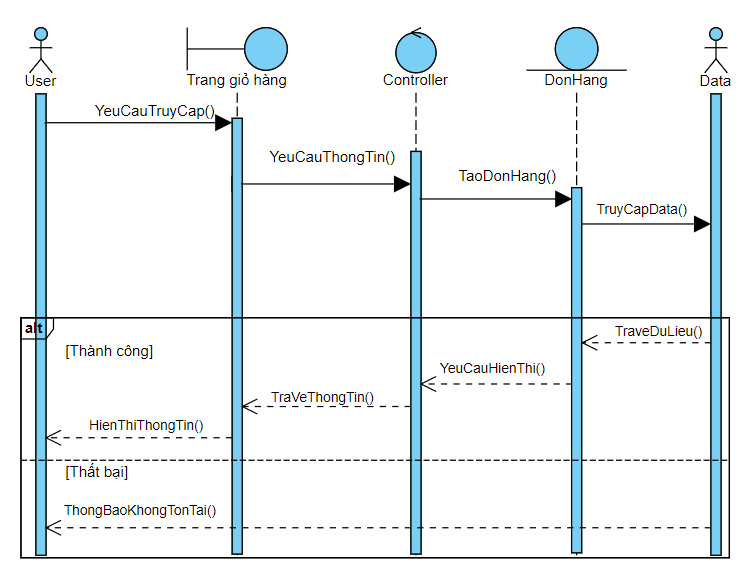
### ***Usecase Diagram của Theo dõi đơn hàng***



### ***Scenario của usecase Theo dõi đơn hàng***

|  |  |
| --- | --- |
| **Actor(s)** | Người dùng |
| **Pre-conditions** | Người dùng phải có hóa đơn |
| **Post-conditions** | Nếu xem hoặc hủy đơn hàng thành công: Hệ thống trả về cho người dùng các thông tin của đơn hàng.  Nếu xem hoặc hủy đơn hàng thất bại: Hệ thống thông báo cho người dùng việc xem hoặc hủy đơn hàng không hợp lệ. |
| **Flow of events** |  |
| Basic flow  (Thành công) | Use case bắt đầu khi người dùng chọn chức năng theo dõi đơn hàng:  1. Hệ thống hiển thị cửa sổ chứa các đơn hàng  2. Người dùng chọn đơn hàng muốn thao tác  3. Hệ thống hiển thị thông tin đơn hàng. |
| Alternative flow  (Thất bại) | Không có |
| **Extension point** | Không có |

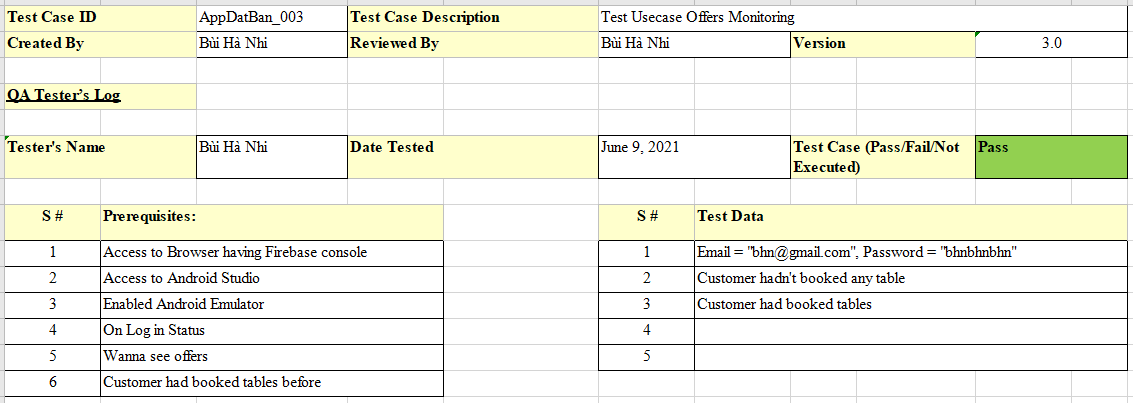
### ***Sequence Diagram của usecase Theo dõi đơn hàng***

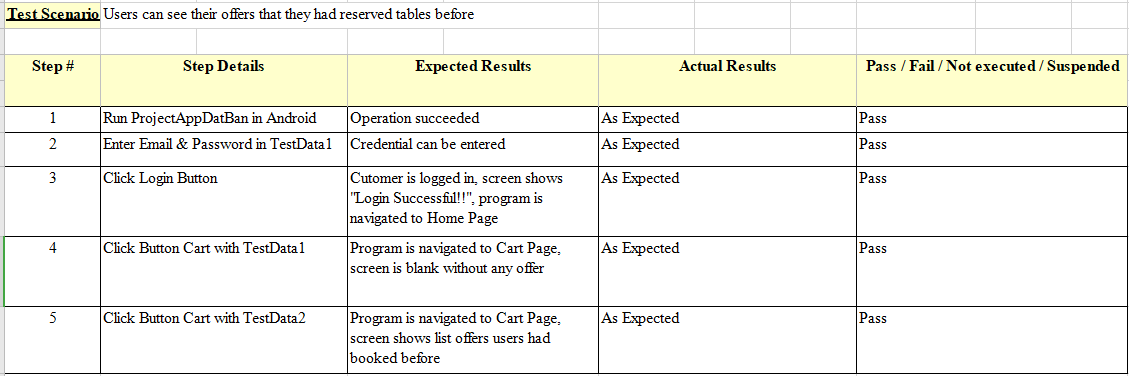


### ***State Diagram của usecase Theo dõi đơn hàng***

#### 

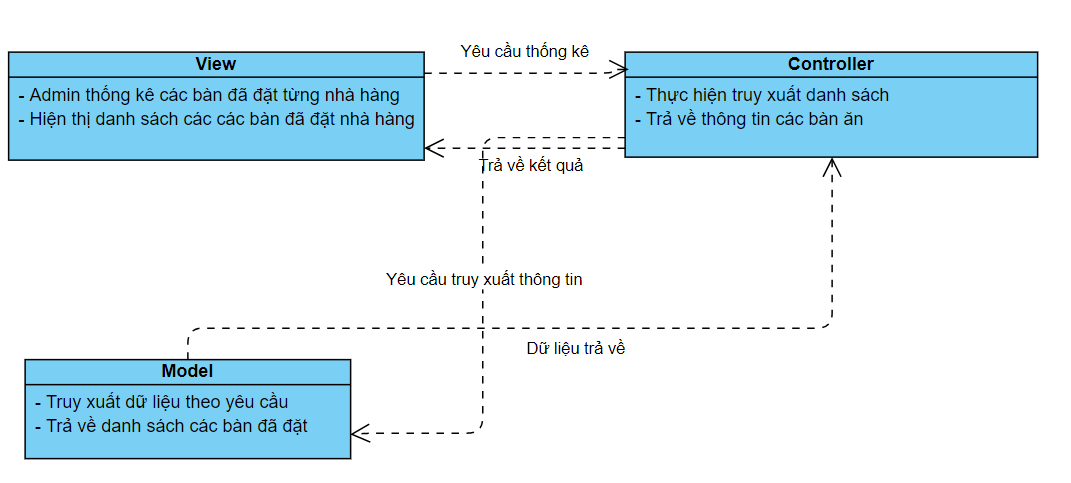
### ***Test case của usecase Theo dõi đơn hàng***



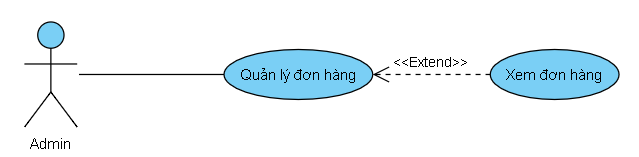


## **Usecase 8: Quản lý đơn hàng**

### ***Mô hình MVC***



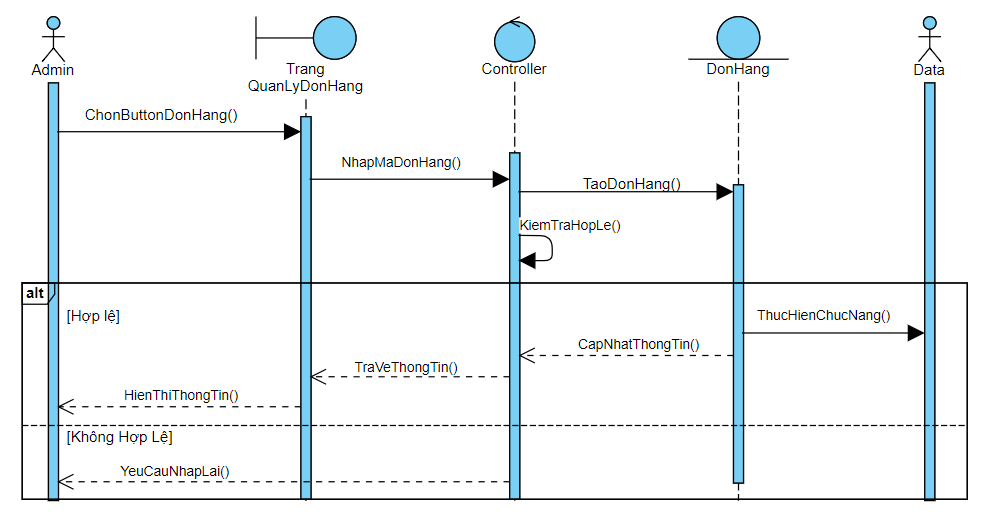
### ***Usecase Diagram của Quản lý đơn hàng***



### ***Scenario của usecase Quản lý đơn hàng***

|  |  |
| --- | --- |
| **Name** | **Quản lý đơn hàng** |
| **Brief description** | Người dùng quản lý đơn hàng trong chi nhánh nhà hàng |
| **Actor(s)** | Admin |
| **Pre-conditions** | Thông tin đơn hàng trong chi nhánh nhà hàng |
| **Post-conditions** | Nếu xem, xóa đơn hàng của nhà hàng thành công: Hệ thống trả về cho người dùng các thông tin của đơn hàng trong nhà hàng đó |
| **Flow of events** |  |
| Basic flow  (Thành công) | Use case bắt đầu khi người dùng cần chọn chức năng quản lý đơn hàng trong chi nhánh nhà hàng  1. Hệ thống hiển thị cửa sổ chứa các đơn hàng, thông tin chi tiết đơn hàng.  2. Hệ thống hiển thị các chức năng quản lý đơn hàng trong chi nhánh nhà hàng: xóa đơn hàng của nhà hàng đó.  5. Người dùng chọn chức năng xóa đơn hàng trong nhà hàng.  6. Hệ thống hiển thị thông tin đơn hàng của nhà hàng đó vừa xóa thành công |
| Alternative flow  (Thất bại) | Không có |
| **Extension point** | Không có |

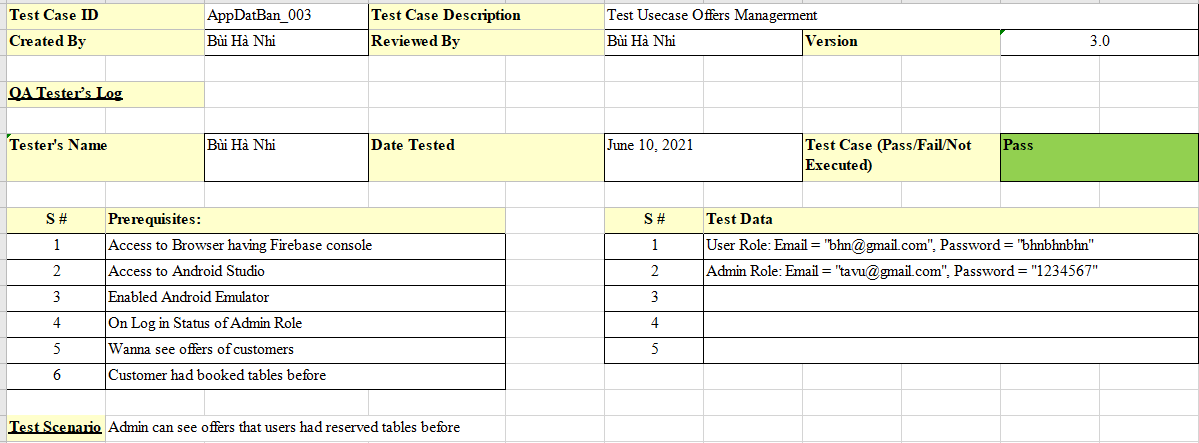
### ***Sequence Diagram của usecase Quản lý đơn hàng***

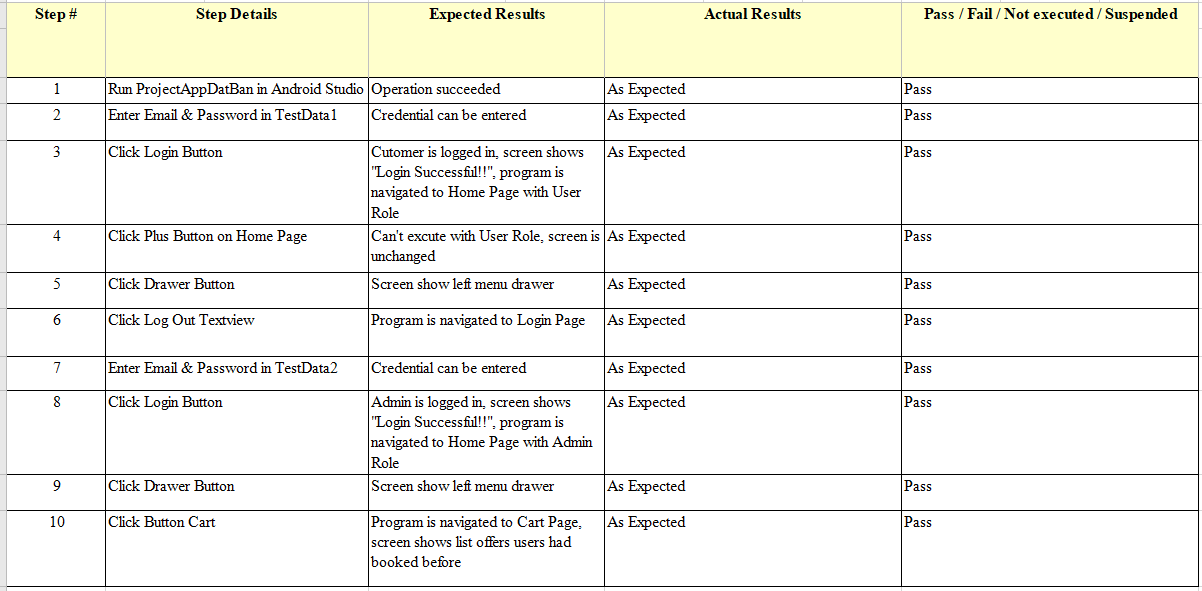


### ***State Diagram của usecase Quản lý đơn hàng***

#### 

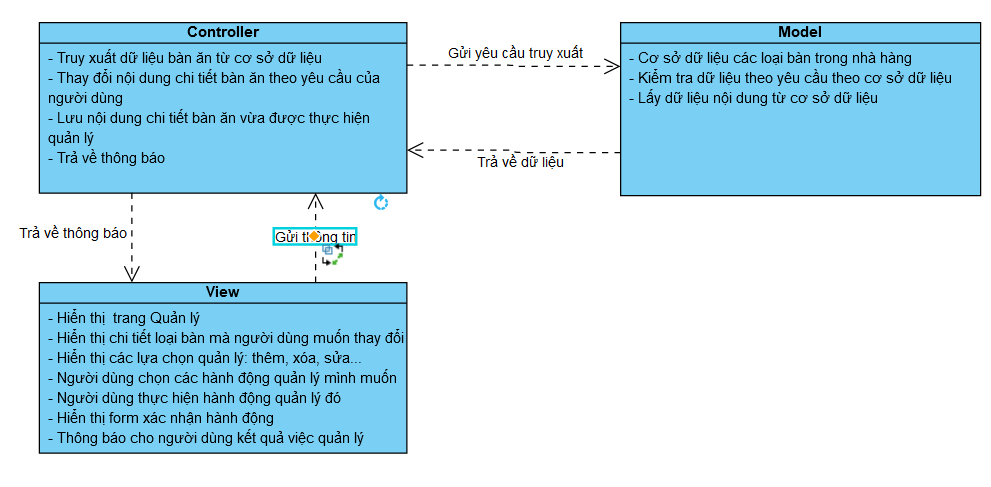
### ***Test case của usecase Quản lý đơn hàng***



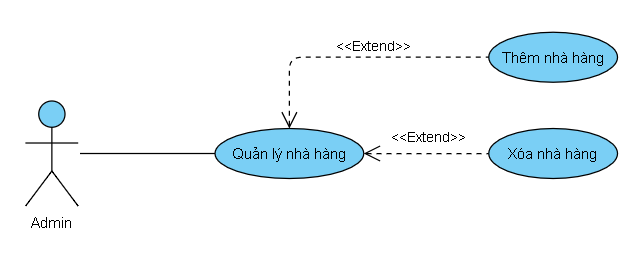


## **Usecase 9: Quản lý nhà hàng**

### ***Mô hình MVC***



### ***Usecase Diagram của Quản lý nhà hàng***

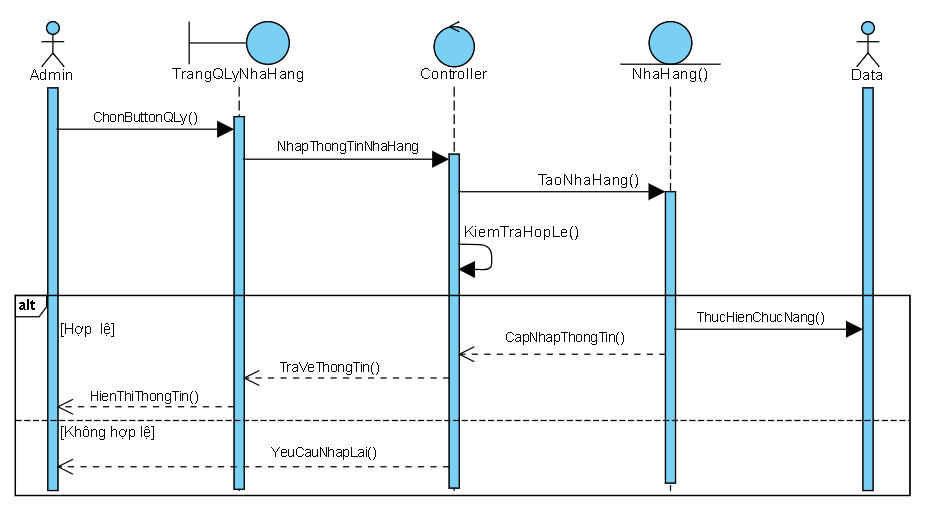


### ***Scenario của usecase Quản lý nhà hàng***

|  |  |
| --- | --- |
| **Name** | **Quản lý nhà hàng** |
| **Brief description** | Người dùng quản lý nhà hàng trong chuỗi các nhà hàng của ứng dụng |
| **Actor(s)** | Người quản lý |
| **Pre-conditions** | Thông tin nhà hàng trong hệ thống app đặt bàn |
| **Post-conditions** | Nếu thêm, xóa thông tin nhà hàng trong chuỗi các nhà hàng của ứng dụng thành công: Hệ thống trả về cho người dùng các thông tin mới vừa được cập nhật của nhà hàng trong chuỗi các nhà hàng của ứng dụng.  Nếu thêm, xóa thông tin nhà hàng không thành công: Hệ thống thông báo cho người dùng việc thêm, xóa thông tin nhà hàng không hợp lệ |
| **Flow of events**  Basic flow  (Thành công) | Use case bắt đầu khi người dùng cần chọn chức năng quản lý nhà hàng  1. Hệ thống hiển thị cửa sổ cho người dùng chọn nhập các thông tin nhà hàng được thêm  2. Hệ thống hiển thị các chức năng thêm thông tin nhà hàng trong chuỗi các nhà hàng của ứng dụng  5. Người dùng thực hiện thêm thông tin nhà hàng trong chuỗi các nhà hàng của ứng dụng  6. Hệ thống hiển thị thông tin nhà hàng vừa được thêm thành công trên trang chủ |
| Alternative flow  (Thất bại) | Nếu người dùng nhập thông tin nhà hàng không hợp lệ. Những công việc sau được thực hiện:  1. Hệ thống mô tả thêm nhà hàng không thành công  2. Hệ thống lưu ý người dùng thực hiện thêm thông tin nhà hàng lại  3. Khi người dùng nhập lại mô tả thông tin mới nhà hàng, bước 3 ở Basic flow được thực hiện. |
| **Extension point** | Không có |

### ***Sequence Diagram của usecase Quản lý nhà hàng***

#### 

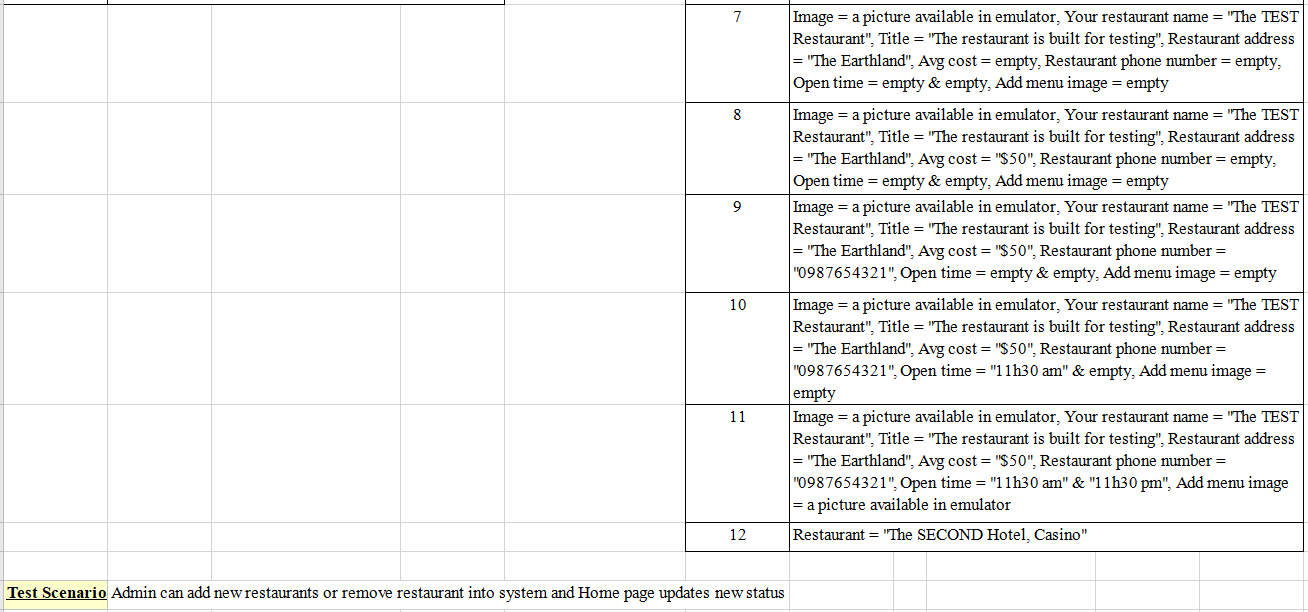


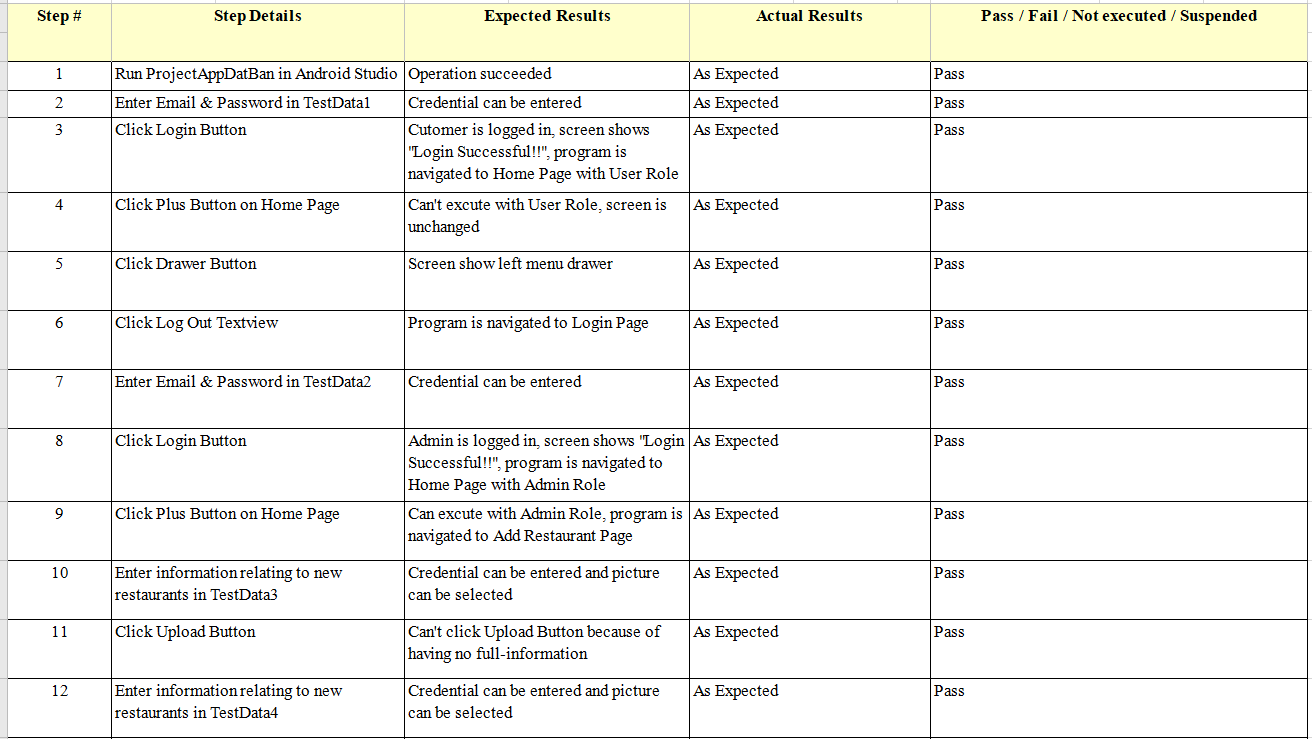
### ***State Diagram của usecase Quản lý nhà hàng***

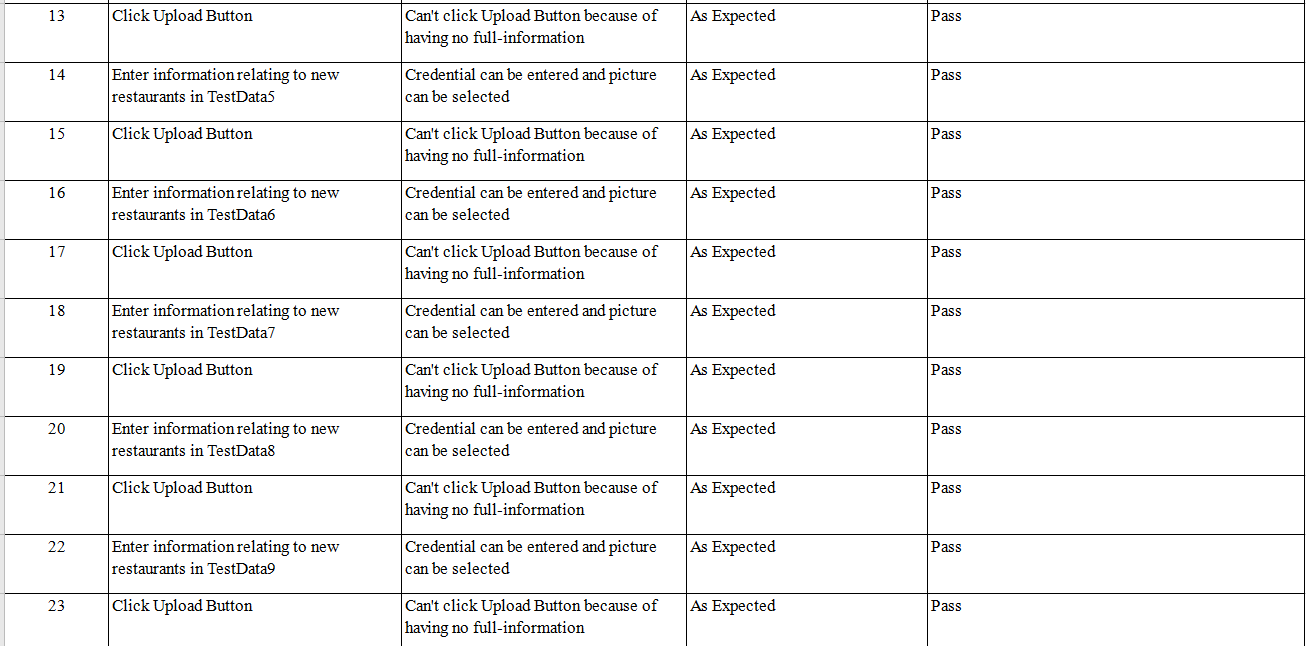
#### 

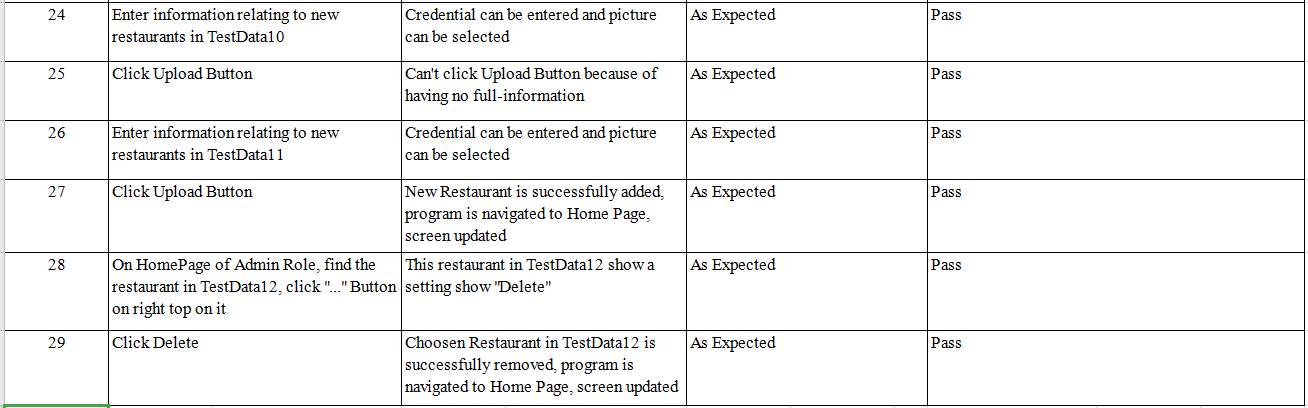
### ***Test case của usecase Quản lý nhà hàng***

#### 



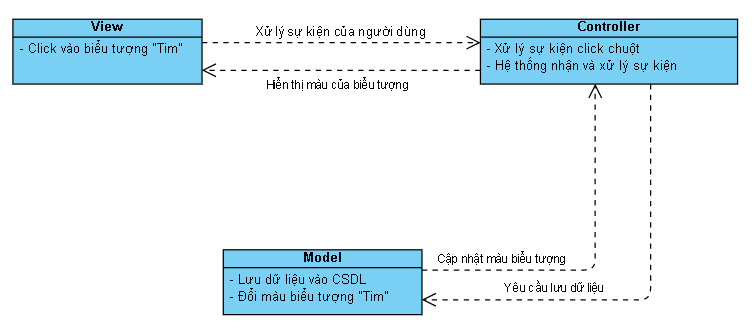




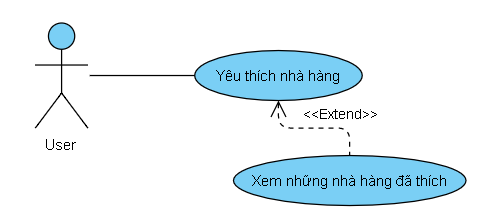


## **Usecase 10: Yêu thích nhà hàng**

### ***Mô hình MVC***



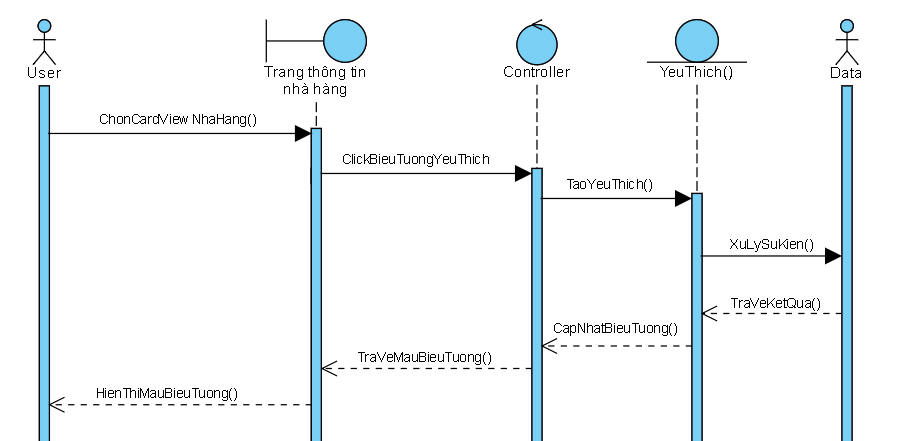
### ***Usecase Diagram của Yêu thích nhà hàng***



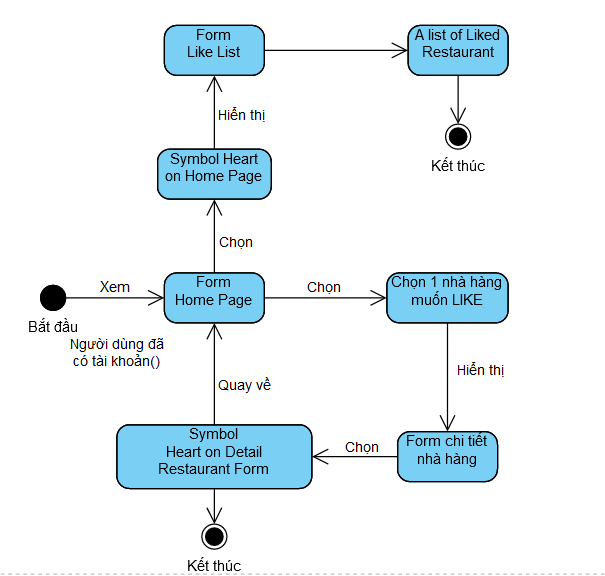
### ***Scenario của usecase Yêu thích nhà hàng***

|  |  |
| --- | --- |
| **Name** | **Yêu thích** |
| **Brief description** | Người dùng like cho nhà hàng trong hệ thống |
| **Actor(s)** | Khách hàng |
| **Pre-conditions** | Người dùng đang đăng nhập trên phiên bản hiện tại |
| **Post-conditions** | Nếu like thành công, những nhà hàng đã like sẽ vào cart like. Hệ thống hiển thị nhà hàng đã like của người dùng |
| **Flow of events**  Basic flow  (Thành công) | Use case bắt đầu khi người dùng chọn chức năng yêu thích.  1. Người dùng vào nhà hàng sẽ hiện thông tin chi tiết nhà hàng.  2. Người dùng tiến hành like cho nhà hàng, icon like nằm trên ảnh đại diện của nhà hàng  3 Những nhà hàng khi like sẽ vào cart like, hệ thống hiển thị những nhà hàng mà khách hàng đã like và khách hàng có thể xem. |
| Alternative flow  (Thất bại) | Không có |
| **Extension point** | Không có |

### ***Sequence Diagram của usecase Yêu thích nhà hàng***

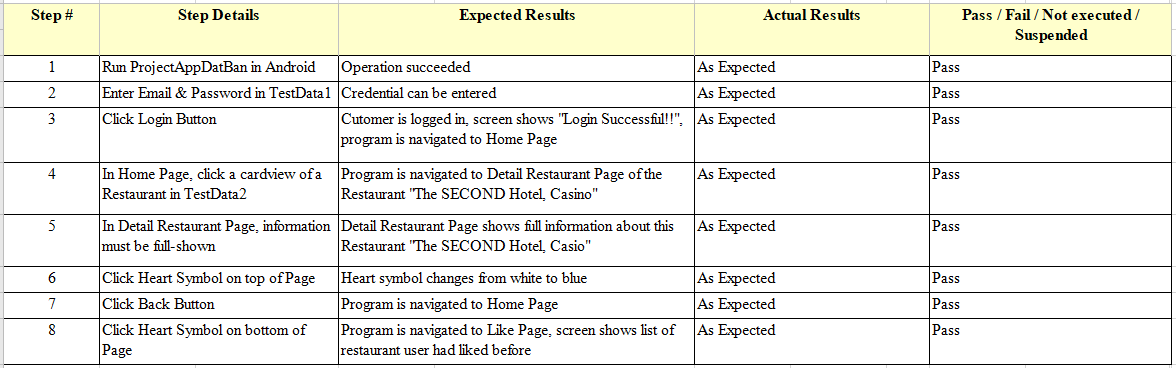


### ***State Diagram của usecase Yêu thích nhà hàng***



### ***Test case của usecase Yêu thích nhà hàng***

### 



# **PHẦN 4: HƯỚNG DẪN SỬ DỤNG**

|  |  |  |
| --- | --- | --- |
| Chức năng |  | Giao diện |
| 1. Đăng nhập | * Người dùng nhập Email và password đã đăng ký để đăng nhập vào hệ thống với tư cách là người dùng. |  |
| * Trường hợp nhập sai Email hoặc Password sẽ hiển thị dòng thông báo “Login failed!” |  |
| * Trường hợp vô tình bỏ trống Email sẽ hiển thị thông báo “Please enter email”. * Trường hợp bỏ trống Password sẽ hiện thông báo “Please enter password” |  |
| 1. Đăng ký | Nếu người dùng chưa có tài khoản có thể nhấn vào “” để tiến hành đăng ký tài khoản.  Người dùng phải điền đầy đủ thông tin:   * Name: chỉ có thể nhập ký tự * Email: phải chứa ký tự ‘@’ * Mobile number: chỉ chứa số * Password yêu cầu phải có ít nhất 6 ký tự. |  |
| 1. Đăng xuất | Người dùng chọn vào dấu 3 gạch bên gốc phải của màn hình home, sẽ hiện ra slide menu chọn Log Out để có thể rời khỏi app |  |
| 1. Đặt bàn | Click chọn vào nhà hàng mà người dùng muốn đến, ở trang này sẽ có đầy đủ thông tin của nhà hàng đó   * Tên, địa chỉ nhà hàng * Rate * Thời gian mở cửa * Phone * Menu * Ước tính giá cho 2 người * Chức năng đặt bàn nếu người dùng có nhu cầu |  |
| Khi chọn vào chức năng “Book a Table”, sẽ hiển thị các câu hỏi yêu cầu người dùng phải chọn   * “Where do you want to eat?”: hỏi người dùng muốn chọn vị trí nào, bên trong “Indoor” hay bên ngoài “Outdoor” nhà hàng. * “Select your person”: Chọn hình thức phục vụ, 1 người “Single”, 2 người “Couple”, gia đình “Family”. * “How many Seats?”: số ghế mà người dùng muốn đặt, có thể nhập số hoặc chữ. * “Submit”: để hoàn tất việc đặt bàn. |  |
| 1. Đánh giá | * Click vào biểu tượng “” sẽ hiện ra form đánh giá, người dùng chỉ cần chạm vào số sao tương ứng mà mình muốn đánh giá cho nhà hàng và click vào “” để kết thúc việc đánh giá |  |
| 1. Quản lý nhà hàng | Khi đăng nhập với quyền Admin, ở trang Home, Admin có thể thêm mới một nhà hàng khi click vào biểu tượng “”, điền đầy đủ thông tin được yêu cầu của nhà hàng mới   * “Đăng ảnh đại diện cho nhà hàng”: ở đây bắt buộc phải có hình ảnh của nhà hàng. * “Your restaurant name”: yêu cầu phải nhập tên của nhà hàng. * “title”: cần một tiêu đề cho cho nhà hàng mới. * “Restaurant address”: nhập địa chỉ của nhà hàng. * “Avg cost”: ước tính chi phí trung bình cho một bữa ăn. * “Restaurant Phone number”: nhập số điện thoại của nhà hàng. * “Open time”: chọn thời gian hoạt động cho nhà hàng.   rồi chọn “upload”, vậy là một nhà hàng mới đã được thêm. |  |
| 1. Quản lý đơn hàng | Admin có thể xóa một nhà hàng khi click vào biểu tượng “” popup trên card view rồi chọn “Delete restaurant” |  |
| 1. Theo dõi đơn hàng | Admin và user có thể xem và theo dõi những nhà hàng mà user đã book, hiển thị trong cart |  |
| 1. Tra cứu | * Tra cứu nhà hàng bằng tên nhà hàng |  |
| 1. Yêu thích | * Có thể đánh giá bằng việc chọn vào biểu tượng “” để thể hiện sự yêu thích nhà hàng đó. * Khi like 1 nhà hàng bất kỳ những nhà hàng đã like sẽ hiển thị trong cart like trên thanh bottom nav |  |

# **PHẦN 5: KẾT LUẬN**

## **5.1. Đóng góp của đề tài**

Qua việc thực hiện đề tài trên, nhóm đã tạo ra một ứng dụng đơn giản, các tính năng dễ sử dụng và thân thiện với người sử dụng. Trong quá trình thực hiện và hoàn thành đề tài, nhóm đã đạt được những mục tiêu sau:

* Biết cách sử dụng và hiểu được cơ chế làm việc của Firebase và lập trình Java Mobile trên Android Studio.
* Xây dựng ứng dụng có thể áp dụng được thực tế và phù hợp với mọi đối tượng.
* Áp dụng các kiến thức đã học vào ứng dụng.
* Tiếp cận đến các công nghệ mới của Mobile để thực hiện đề tài.

## **5.2. Hạn chế của đề tài**

Trong quá trình thực hiện và hoàn thành đề tài trên, nhóm nhận thấy được các hạn chế sau:

* Chưa tìm ra và thực hiện được nhiều tính năng mới so với các ứng dụng đặt bàn trong thực tế.
* Xử lý chắp vá còn nhiều hạn chế xử lý lỗi trong giao diện.
* Đặt tên biến còn đơn giản, chỉ dễ hiểu đối với người thực hiện.
* Một số tính năng chưa phù hợp với thực tế.
* Thiết kế cơ sở dữ liệu chưa hợp lý, còn dư thừa.
* Code giao diện vẫn còn rườm rà.
* Chưa kiểm thử hết tất cả các trường hợp có thể xảy ra.

## **5.1. Hướng phát triển của đề tài**

Với nỗ lực của bản thân, nhóm đã cố gắng hoàn thành yêu cầu đề tài. Do thời gian và kiến thức có hạn nên ứng dụng của nhóm mới chỉ đi sâu vào các chức năng đơn giản. Nhóm cần có hướng phát triển thêm ứng dụng trở thành một ứng dụng đặt bàn hoàn chỉnh hơn. Cung cấp các tính năng liên quan nhiều vào thực tế, giao diện đẹp mắt, đơn giản và dễ sử dụng, trên app có chức năng thêm nhà hàng sau này có thể cho các chủ nhà hàng thêm nhà hàng.

-- HẾT --