**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**

**THẠCH THANH THIÊN – 14520870**

**TRẦN MINH THẮNG - 14521173**

**KHÓA LUẬN TỐT NGHIỆP**

**XÂY DỰNG ỨNG DỤNG THIẾT KẾ NỘI THẤT**

**SỬ DỤNG THỰC TẾ ẢO TĂNG CƯỜNG**

**Build an interior design application using augmented reality**

**KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM**

**TP. HỒ CHÍ MINH, 2019**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**

**THẠCH THANH THIÊN – 14520870**

**TRẦN MINH THẮNG - 14521173**

**KHÓA LUẬN TỐT NGHIỆP**

**XÂY DỰNG ỨNG DỤNG THIẾT KẾ NỘI THẤT**

**SỬ DỤNG THỰC TẾ ẢO TĂNG CƯỜNG**

**Build an interior design application using augmented reality**

**KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM**

**GIẢNG VIÊN HƯỚNG DẪN**

**THS. THÁI THỊ HÀN UYỂN**

**TP. HỒ CHÍ MINH, 2019**

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

ĐHQG TP. HỒ CHÍ MINH **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**TRƯỜNG ĐẠI HỌC Độc Lập – Tự Do – Hạnh Phúc**

**CÔNG NGHỆ THÔNG TIN** *TP. HCM, ngày tháng năm 2018*

**NHẬN XÉT KHOÁ LUẬN TỐT NGHIỆP**

**(CỦA CÁN BỘ HƯỚNG DẪN)**

**Tên khoá luận:**

**XÂY DỰNG ỨNG DỤNG THIẾT KẾ NỘI THẤT**

**SỬ DỤNG THỰC TẾ ẢO TĂNG CƯỜNG**

**Nhóm SV thực hiện: Cán bộ hướng dẫn:**

Thạch Thanh Thiên 14520870 Thạc sĩ Thái Thị Hàn Uyển

Trần Minh Thắng 14521173

**Đánh giá khoá luận:**

1. Về cuốn báo cáo:

Số trang: Số chương:

Số bảng số liệu: Số hình vẽ:

Số tài liệu tham khảo: Sản phẩm:

Một số nhận xét về hình thức cuốn báo cáo:

1. Về nội dung nghiên cứu:

1. Về chương trình ứng dụng:

1. Về thái độ làm việc của sinh viên:

**Đánh giá chung:**

**Điểm từng sinh viên:**

Thạch Thanh Thiên: ***/10***

Trần Minh Thắng: ***/10***

**Người nhận xét**

(Ký tên và ghi rõ họ tên)

**ThS. Thái Thị Hàn Uyển**

ĐHQG TP. HỒ CHÍ MINH **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**TRƯỜNG ĐẠI HỌC Độc Lập – Tự Do – Hạnh Phúc**

**CÔNG NGHỆ THÔNG TIN** *TP. HCM, ngày tháng năm 2018*

**NHẬN XÉT KHOÁ LUẬN TỐT NGHIỆP**

**(CỦA CÁN BỘ PHẢN BIỆN)**

**Tên khoá luận:**

**XÂY DỰNG ỨNG DỤNG THIẾT KẾ NỘI THẤT**

**SỬ DỤNG THỰC TẾ ẢO TĂNG CƯỜNG**

**Nhóm SV thực hiện: Cán bộ phản biện:**

Thạch Thanh Thiên 14520466

Trần Minh Thắng 14520466

**Đánh giá khoá luận:**

1. Về cuốn báo cáo:

Số trang: Số chương:

Số bảng số liệu: Số hình vẽ:

Số tài liệu tham khảo: Sản phẩm:

Một số nhận xét về hình thức cuốn báo cáo:

1. Về nội dung nghiên cứu:

1. Về chương trình ứng dụng:

1. Về thái độ làm việc của sinh viên:

**Đánh giá chung:**

**Điểm từng sinh viên:**

Thạch Thanh Thiên: ***/10***

Trần Minh Thắng: ***/10***

**Người nhận xét**

(Ký tên và ghi rõ họ tên)

**LỜI CẢM ƠN**

Sau 4 năm học tập và rèn luyện tại Trường Đại Học Công Nghệ Thông Tin, được sự chỉ bảo và giảng dạy nhiệt tình của quý thầy cô, đặc biệt quý thầy cô trong khoa Công Nghệ Phần Mềm, chúng em đã nhận được vô vàng kiến thức bổ ích về lý thuyết và thực hành trong thời gian qua. Trong khoảng thời gian thực hiện luận văn tốt nghiệp, chúng em càng học hỏi thêm được nhiều kiến thức, kinh nghiệm để áp dụng và đã hoàn thành được luận văn tốt nghiệp của chúng em.

Xin chân thành cảm ơn quý thầy cô Khoa Công Nghệ Phần mềm, đặc biệt là cô Thái Thị Hàn Uyển đã tận tình hướng dẫn chúng em hoàn thành luận văn tốt nghiệp này.

Bên cạnh đó, một lần nữa xin cảm ơn nhóm các bạn cùng lớp làm khóa luận đã động viên, thảo luận và góp ý cho nhóm đồng thời đã khơi thêm nguồn động lực cho nhóm để tiếp thêm động lực cho nhóm trong giai đoạn đầy khó khăn.

Ngoài ra, chúng em cũng cảm ơn các anh, chị, bạn bè và gia đình đã giúp đỡ và tạo điều kiện cho chúng em hoàn tất luận văn này.

Mặc dù đã rất cố gắng, tuy nhiên vì khả năng giới hạn nên vẫn không tránh khỏi những sai sót, chúng em rất mong nhận được sự thông cảm và góp ý chân tình từ quý thầy cô và các bạn. Một lần nữa xin được chân thành cảm ơn mọi người.

Thành phố Hồ Chí Minh, tháng 01 năm 2019

Thạch Thanh Thiên

Trần Minh Thắng

|  |  |
| --- | --- |
| ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |
|  | TP. HCM, ngày 16 tháng 09 năm 2018 |

**ĐỀ CƯƠNG CHI TIẾT**

|  |  |  |
| --- | --- | --- |
| **Tên đề tài: XÂY DỰNG ỨNG DỤNG THIẾT KẾ NỘI THẤT**  **SỬ DỤNG THỰC TẾ ẢO TĂNG CƯỜNG** | | |
| **Cán bộ hướng dẫn:** ThS. Thái Thị Hàn Uyển | | |
| **Thời gian thực hiện:** Từ ngày 03/09/2018 đến ngày 04/01/2019 | | |
| **Sinh viên thực hiện:**  Thạch Thanh Thiên - MSSV: 14520870  Trần Minh Thắng - MSSV: 14521173 | | |
| **Nội dung đề tài:**  ***Mục tiêu***:   * Tìm hiểu về công nghệ thực tế ảo tăng cường và xu hướng phát triển * Xây dựng phần mềm thiết kế nội thất bằng thực tế ảo tăng cường cung cấp cho người dùng trải nghiệm mua sắm mới mẻ, tiết kiệm thời gian và công sức. * Vận hành được phần mềm vào thực tế.   ***Đối tượng sử dụng:***   * Các công ty, showroom nội thất * Khách hàng muốn mua nội thất   ***Danh sách tính năng:***  **1. Cho xem trước model**   * Hiển thị model cùng các thông tin chi tiết liên quan (kích cỡ, màu sắc, xuất sứ, nhãn hiệu, giá thành…)   **2. Nhận dạng mặt phẳng để thiết kế**   * Quét và nhận diện mặt phẳng dùng để đặt model   **3. Thiết kế**   * Đặt model trên mặt phẳng đã được nhận diện * Cho phép xoay, di chuyển model. * Cho phép lựa chọn các thông số cho model (màu sắc, kích cỡ…) * Cho phép hiển thị các thông tin chi tiết của model   **4. Lưu lại thiết kế và chia sẻ thiết kế**   * Cho phép người dùng lưu lại hình ảnh và chia sẻ hình ảnh lên mạng xã hội   **5. Mua hàng và thanh toán**   * Cho phép người dùng chọn hàng, đặt mua và thanh toán hàng.   ***Phương pháp thực hiện:***   * Phân tích đề tài * Thiết kế yêu cầu * Nghiên cứu công nghệ áp dụng * Tiến hành phát triển phần mềm thông quá cái giai đoạn, thu thập ý kiển phản hồi và sửa đổi * Kiểm thử và hoàn thiện phần mềm   ***Kết quả mong đợi:***   * Xây dựng được ứng dụng với các chức năng mong muốn * Mang lại cho người dùng trải nghiệm mua sắm mới mẻ, trực quan, tiết kiệm thời gian và công sức, xóa đi nỗi lo món hàng đó có hợp hay không. * Thay đổi thói quen mua sắm của mọi người. | | |
| **Kế hoạch thực hiện** | | |
| **Thời gian** | **Công việc** | |
| Tuần 1 (3/9 – 9/9) | * Lên ý tưởng, thống nhất đề tài là **Xây dựng ứng dụng trang trí nội thất sử dụng thực tế ảo tăng cường** * Viết mô tả cho đề tài | |
| Tuần 2 – 3 (10/9 – 23/9) | * Tìm hiểu, nghiên cứu các công nghệ để áp dụng, so sánh đánh giá * Tìm hiểu các ứng dụng đã có trên thị trường, so sánh đánh giá * Thống nhất công nghệ sử dụng là  **+ React Native và Java Android dùng để build app**   **+ MaxST dùng để hiện thực AR**  **+ NodeJS với Express và PostgreSQL dùng để build server và database** | |
| Tuần 4 ( 24/9 – 30/9) | * Thiết kế chi tiết cho ứng dụng: usecase, database, layout | |
| Tuần 5 – 8 (1/10 – 28/10) | * Thiết kế database * Thiết kế server * Viết các API hỗ trợ cho ứng dụng | |
| Tuần 9 – 12 (29/10 – 25/11) | * Xây dựng ứng dụng dựa trên thiết kế chi tiết và thiết kế giao diện * Tiếp tục hoàn thiện database và server. * Tiếp tục viết các API cần thiết cho ứng dụng * Hiện thực tính năng AR * Dần hoàn thiện báo cáo cuối kỳ | |
| Tuần 13 – 16 (26/11 – 23/12) | * Khoảng thời gian dự trữ cho các công việc chưa hoàn thành * Test ứng dụng * Phát hành ứng dụng * Vận hành ứng dụng * Hoàn thành báo cáo | |
| **Xác nhận của CBHD**  (Ký tên và ghi rõ họ tên)  Thái Thị Hàn Uyển | | **TP. HCM, ngày 16 tháng 09 năm 2018**  **Sinh viên 1**  (Ký tên và ghi rõ họ tên)  Thạch Thanh Thiên  **Sinh viên 2**  (Ký tên và ghi rõ họ tên)  Trần Minh Thắng |

MỤC LỤC

[DANH MỤC HÌNH VẼ 14](#_Toc528969071)

[DANH MỤC BẢNG 15](#_Toc528969072)

[MỞ ĐẦU 16](#_Toc528969073)

[CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI 17](#_Toc528969074)

[**1.** **Đặt vấn đề** 17](#_Toc528969075)

[**2.** **Mô tả đề tài** 19](#_Toc528969076)

[**3.** **Mục tiêu và phạm vi đề tài** 19](#_Toc528969077)

[**4.** **Cấu trúc khóa luận** 20](#_Toc528969078)

[**CHƯƠNG 2: CÁC HỆ THỐNG LIÊN QUAN** 21](#_Toc528969079)

[**1.** **Housecraft** 21](#_Toc528969080)

[**2.** **IKEA Place** 22](#_Toc528969081)

[**3.** **DecorMatters** 24](#_Toc528969082)

[**CHƯƠNG 3: CÁC CÔNG NGHỆ SỬ DỤNG** 26](#_Toc528969083)

[**1.** **NodeJS** 26](#_Toc528969084)

[**1.1.** **Giới thiệu chung** 26](#_Toc528969085)

[**1.2.** **Đặc điểm nổi bật** 27](#_Toc528969086)

[**1.3.** **Lý do sử dụng** 27](#_Toc528969087)

[**2.** **Framework Express** 28](#_Toc528969088)

[**2.1.** **Giới thiệu chung** 28](#_Toc528969089)

[**2.2.** **Đặc điểm nổi bật** 28](#_Toc528969090)

[**2.3.** **Lý do sử dụng** 29](#_Toc528969091)

[**3.** **PostgreSQL** 30](#_Toc528969092)

[**3.1.** **Giới thiệu chung** 30](#_Toc528969093)

[**3.2.** **Đặc điểm nổi bật** 31](#_Toc528969094)

[**3.3.** **Lý do sử dụng:** 32](#_Toc528969095)

[**4.** **React Native** 32](#_Toc528969096)

[**4.1.** **Giới thiệu chung** 32](#_Toc528969097)

[**4.2.** **Đặc điểm nổi bật** 33](#_Toc528969098)

[**4.3.** **Lý do sử dụng** 34](#_Toc528969099)

[**CHƯƠNG 4: XÁC ĐỊNH YÊU CẦU** 35](#_Toc528969100)

[**1.** **Danh sách yêu cầu** 35](#_Toc528969101)

[**2.** **Mô tả yêu cầu** 35](#_Toc528969102)

[**CHƯƠNG 5: PHÂN TÍCH THIẾT KẾ** 36](#_Toc528969103)

[**1.** **Phân tích yêu cầu** 36](#_Toc528969104)

[**1.1.** **Sơ đồ usecase** 36](#_Toc528969105)

[**1.2.** **Danh sách người dùng** 38](#_Toc528969106)

[**1.3.** **Mô tả usecase** 38](#_Toc528969107)

[**2.** **Thiết kế cơ sở dữ liệu** 38](#_Toc528969108)

[**2.1.** **Lượt đồ dữ liệu khái niệm** 38](#_Toc528969109)

[**2.2.** **Mô tả các đối tượng dữ liệu** 39](#_Toc528969110)

[**3.** **Thiết kế kiến trúc phần mềm** 45](#_Toc528969111)

[**3.1. Kiến trúc tổng thể** 45](#_Toc528969112)

[**4.** **Thiết kế giao diện** 45](#_Toc528969113)

[**4.1.** **Sơ đồ liên kết giao diện** 45](#_Toc528969114)

[**4.2.** **Danh sách giao diện** 45](#_Toc528969115)

[**4.3.** **Mô tả giao diện** 45](#_Toc528969116)

# DANH MỤC HÌNH VẼ

Danh mục hình vẽ nằm đây

# DANH MỤC BẢNG

Danh mục bảng nằm đây

# MỞ ĐẦU

Mở đầu nằm đây

# CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

1. **Đặt vấn đề**

AR (augmented reality) là "công nghệ tích hợp đồ họa máy tính và mạng internet với thế giới thực để người dùng xem các đối tượng ảo trong môi trường thực". Các hình ảnh thực tế trước mắt người dùng được "tăng cường" hoặc bổ sung thêm các thông tin ảo. Nó giúp cho những hình ảnh thực tế trước mắt người dùng trở nên phong phú hơn với các hình ảnh ảo. Đến thời điểm hiện tại, người dùng có thể sử dụng ngay thiết bị smartphone của mình để trải nghiệm AR mà không cần phải sử dụng thêm công cụ khác.

Công nghệ AR đang từng bước thay đổi những trải nghiệm thưởng thức của mọi người và đang được phát triển để ứng dụng trong rất nhiều lĩnh vực như giải trí, giáo dục, kĩ thuật...

***Ứng dụng trong giải trí***

Các trò chơi chơi điện tử ứng dụng công nghệ AR thực tế đã có mặt trên thế giới từ vài năm sau khi smartphone bắt đầu thịnh hành. Trò chơi sẽ lấy bối cảnh của thế giới thực và bổ sung các thêm các thông tin vào để tạo thành các trò chơi hấp dẫn. Tuy nhiên, công nghệ AR chỉ mới thực sự trở nên phổ biến sau khi Nitendo cho ra mắt game Pokemon Go.

Với các thể loại game chiến đấu có tích hợp công nghệ tương tác ảo, ứng dụng game sẽ sử dụng cảnh thật như một phần để tạo nên chiến trường trong game. Điều này mang đến những trải nghiệm hết sức thú vị và sống động cho game thủ.

***Ứng dụng trong giáo dục***

Chỉ cần một chiếc điện thoại hoặc máy tính bảng, giáo viên có thể tạo ra một bài giảng rất sinh động bằng cách kết hợp giữa tài liệu trên giấy với các mô phỏng 3D mà học sinh có thể tương tác. Đây được dự báo sẽ là phương thức giảng dạy phổ biến trên các quốc gia tiên tiến nhằm tiến đến việc thay thế hoàn toàn sách vở truyền thống, giúp học sinh tiếp nhận các kiến thức về khoa học tự nhiên hiệu quả hơn.

***Ứng dụng trong thương mại***

AR đang được ứng dụng triệt để trong ngành quảng cáo hiện nay, tạo nên xu hướng mới – quảng cáo tương tác. Với các ứng dụng dẫn đường thế hệ mới sử dụng công nghệ AR, người dùng có thể tương tác trực tiếp với môi trường xung quanh trên điện thoại thông minh chạy iOS, Android. Từ hình ảnh trên print ads, billboard…, ngoài việc nhìn thấy hình ảnh thực được chụp hoặc scan qua camera, AR còn hiển thị trên màn hình điện tử với các thông tin cụ thể về sản phẩm hay nhãn hàng. Ngoài ra AR còn cung cấp các sản phẩm dùng thử trực quan. Ứng dụng AR đem đến một hướng thể hiện mới cho catalog các nhà bán lẻ trực tuyến, cho phép tương tác trên các tính năng trực tuyến. Sử dụng ứng dụng AR cho phép bạn xem sản phẩm theo mọi góc độ, và có thể trình chiếu sản phẩm trong không gian ba chiều mà khách hàng mong muốn. Hay nói cách khác, AR mang lại một loạt các lựa chọn mua sắm cho catalog và các nhà bán lẻ online, cung cấp một cái nhìn trực quan hiệu quả cho sản phẩm dẫn giúp cho việc mua sắm của khách hàng trở nên liền mạch.

Trước đây, khi muốn mua đồ nội thất để trang trí cho căn nhà của mình, vấn đề lớn nhất mà người mua trăn trở là liệu món đồ đó có thực sự phù hợp với không gian. Với ứng dụng AR, bạn có thể đặt thử đồ nội thất vào căn hộ một cách dễ dàng. Nó có thể mở rộng đồ nội thất, màu sắc, có thể chọn sofa hoàn hảo cho phòng khách của bạn mà không cần phải mất thời gian suy nghĩ và đắn đo. Việc thiết kế sẽ giảm bớt, và người dùng tự chọn mẫu và thử trước. Như vậy các chi phí sẽ giảm, thời gian lắp đặt nhanh hơn, vừa ý khách hàng hơn.

Việc áp dụng thực tế tăng cường AR là rất cần thiết với khách hàng. Người dùng có thể đi xung quanh và đặt đồ nội thất và đồ vật vào các không gian trong nhà của họ. Phương thức quảng cáo này là một mốc quan trọng trong việc thuyết phục khách hàng rằng đồ nội thất họ định mua có phù hợp trong căn nhà hay không? Khách hàng có thể xem xét và sắp đặt rất nhiều nội thất cho đến khi tìm thấy sản phẩm ưng ý.

1. **Mô tả đề tài**

Ứng dụng đồ nội thất hỗ trợ AR là một hệ thống cửa hàng bán đồ nội thất trực tuyến hỗ trợ người dùng có thể đặt hàng mua các sản phẩm đồ nội thất. Ngoài ra, hệ thống còn hỗ trợ công nghệ AR để người dùng hiển thị các sản phẩm đồ nội thất tương ứng với không gian thực tế, nhằm hỗ trợ người mua hàng kiểm tra sản phẩm nội thất có phù hợp với không gian nhà mình trước khi quyết định mua và tiếp nhận đồ nội thất được vận chuyển đến.

***Những chức năng chính của hệ thống:***

* Đối với khách hàng
* Đăng nhập, đăng ký tài khoản thành viên
* Chỉnh sửa thông tin cá nhân
* Đăng xuất khỏi hệ thống
* Xem, tìm kiếm, xem chi tiết sản phẩm
* Thêm sản phẩm, xem, cập nhật giỏ hàng
* Tạo, theo dõi, hủy đơn hàng
* Sử dụng các tính năng AR
  + Quét mặt phẳng
  + Đặt sản phẩm
  + Di chuyển, xoay, xóa sản phẩm
* Đối với quản trị viên
* Đăng nhập, đăng xuất hệ thống
* Thêm, xóa, cập nhật, xem chi tiết sản phẩm
* Thêm, xóa, cập nhật, xem chi tiết danh mục sản phẩm
* Tìm kiếm, tiếp nhận, xử lý, xem chi tiết đơn hàng
* Thống kê số lượng danh mục, mặt hàng, đơn hàng…

1. **Mục tiêu và phạm vi đề tài**
   1. ***Mục tiêu***

* Xây dựng ứng dụng hỗ trợ người dùng thực hiện các tính năng cơ bản của một ứng dụng đặt mua hàng trực tuyến
* Ứng dụng AR để hỗ trợ người dùng có thể đưa ra lựa chọn đúng đắn.
* Xây dựng ứng dụng có giao diện trực quan, dễ sử dụng
  1. ***Phạm vi đề tài***
* Phạm vi người dùng: người dùng có sử dụng điện thoại thông minh, có nhu cầu mua đồ nội thất trực tuyến

1. **Cấu trúc khóa luận**

**Chương 1: Giới thiệu đề tài:** Chương đầu tiên sẽ trình bày lý do chọn đề tài, mô tả chung về đề tài, giới hạn và phạm vi đề tài.

**Chương 2: Các hệ thống liên quan**: Chương này sẽ trình bày các hệ thông liên quan đến việc ứng dụng AR vào thiết kế nội thất

**Chương 3: Công nghệ sử dụng**: Trình bày một số công nghệ đã sử dụng trong luận văn.

**Chương 4: Xác định yêu cầu**: Xác định yêu cầu bài toán, đặc tả yêu cầu bài toán

**Chương 5: Phân tích thiết kế:** Trình bày phần phân tích thiết kế chức năng, cơ sở dữ liệu, giao diện hệ thống

**Chương 6: Kết luận và hướng phát triển**: Trình bày kết quả đạt được sau khi hoàn thành luận văn và hướng phát triển tiếp theo.

**CHƯƠNG 2: CÁC HỆ THỐNG LIÊN QUAN**

Phần khảo sát sẽ nêu lên một số hệ thống liên quan đến việc ứng dụng công nghệ thực tế ảo tăng cường vào việc thiết kế nội thất. Phần khảo sát mỗi hệ thống bao gồm mô tả chung và nhận xét hệ thống đó.

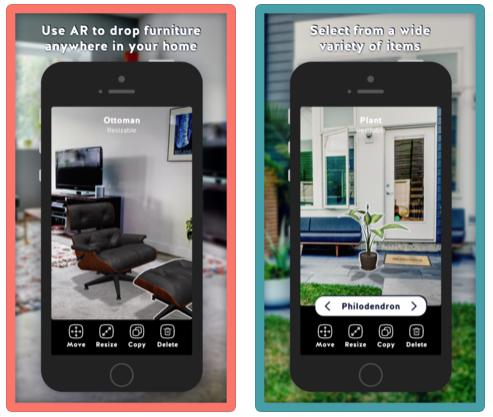
Thông qua việc khảo sát, có thể biết được ý tưởng đã tồn tại hay chưa, học hỏi những ưu điểm, chức năng hay của mỗi hệ thống, hiểu được nhu cầu của khách hàng, từ đó phát triển chức năng phù hợp cho hệ thống

1. **Housecraft**

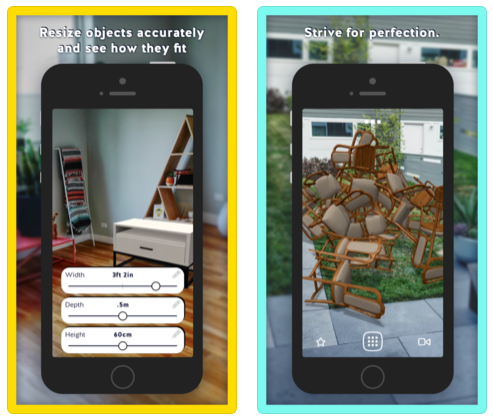
Là một ứng dụng kiểu game được phát hành bởi Sirvo LLC vào 2017, cho phép người chơi sử dụng camera của thiết bị và công nghệ thực tế ảo tăng cường để đặt và sắp xếp các mô hình 3D. Các mô hình 3D được hiển thị với tỉ lệ giống kích thước thật, và có thể thay đổi kích thước tùy ý theo nhu cầu sử dụng. Ứng dụng cho phép lưu lại thiết kế và cho phép áp dụng thiết kế đó vào một nơi khác. Ngoài ra, còn có chế độ **Disater** cho phép người dùng “phá”.

Các tính năng nổi trội:

* **Intuitive Interface** (nhận diện bề mặt): Được trang bị chức năng nhận diện mặt phẳng dùng để đặt vật thể lên
* **Precise Measurements** (Kích thước thật): Hiển thị các vật thể với kích thước thật, cho phép người dùng cảm nhận được độ lớn thật của vật thể và xác định vật có vừa với không gian hay không
* **Persistant Tracking** (Theo dõi liên tục): Cho phép vật thể luôn được đặt tại vị trí mà người dùng đã chỉ định.
* **Save Room Configureations** (Lưu thiết kế): Cho phép lưu các thiết kế mà người dùng đã thực hiện.
* **Disater** (Phá hoại): Cho phép người dùng phá hỏng các thiết kế nhằm mục đích giải trí.



*Hình 2.1: Layout ứng dụng Housecraft*



*Hình 2.2: Layout ứng dụng Housecraft*

**Ưu điểm:**

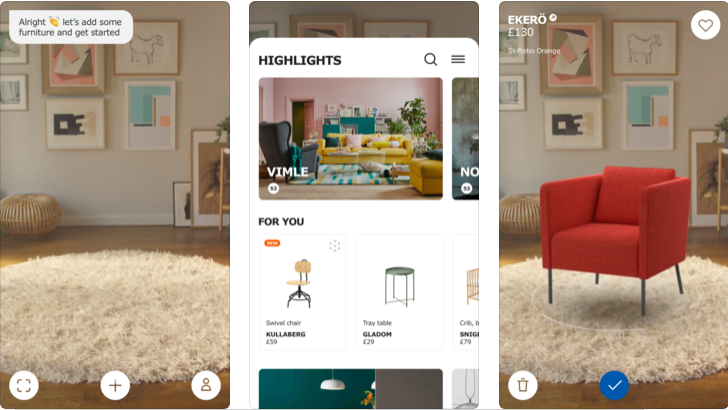
* Có chức năng nhận diện mặt sàn
* UI/UX tốt
* Cho phép hiển thị mô hình với kích thước thật.
* Cho phép lưu thiết kế

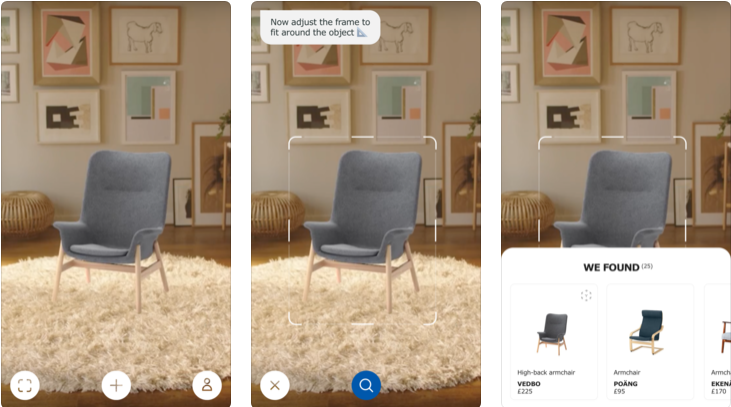
**Nhược điểm:**

* Giới hạn số mô hình 3D, chất lượng mô hình không cao
* Chỉ là game, không giúp người dùng mua bán sản phẩm

1. **IKEA Place**

Dành riêng cho iPhone, IKEA Place là ứng dụng được quảng cáo sẽ thay đổi hoàn toàn cách bạn mua sắm nội thất cho ngôi nhà. Sử dụng công nghệ thực tế tăng cường (Augmented Reality), người dùng có thể "xem trước" (preview) nội thất, sắp xếp chúng vào vị trí mong muốn. Sau đó, IKEA Place sẽ hiển thị đúng món đồ bạn lựa chọn dưới dạng 3D, cân bằng chính xác màu sắc và kích thước tùy theo người dùng. Như vậy, sẽ chẳng còn cảnh đi mua sắm bàn ghế, khệ nệ vác về nhà và ngậm ngùi mang đi thanh lý vì chúng không vừa vặn với căn phòng. Hiện có đến hơn 2000 mô hình cho người dùng thoải mái lựa chọn.





**Ưu điểm:**

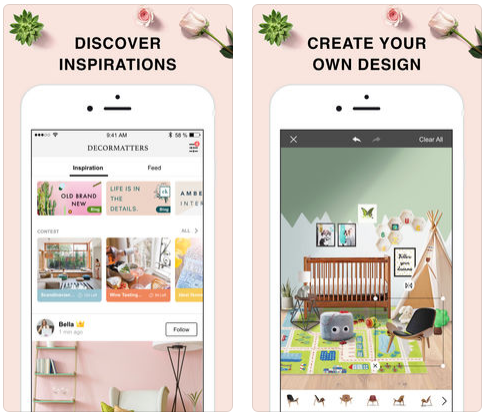
* Có chức năng nhận diện mặt sàn
* UI/UX tốt
* Hỗ trợ mua hàng
* Chất lượng model tốt
* Hỗ trợ share hình ảnh thiết kế
* Hỗ trợ hiển thị hình ảnh với kích thước thật
* Cho phép chọn màu sắc của model

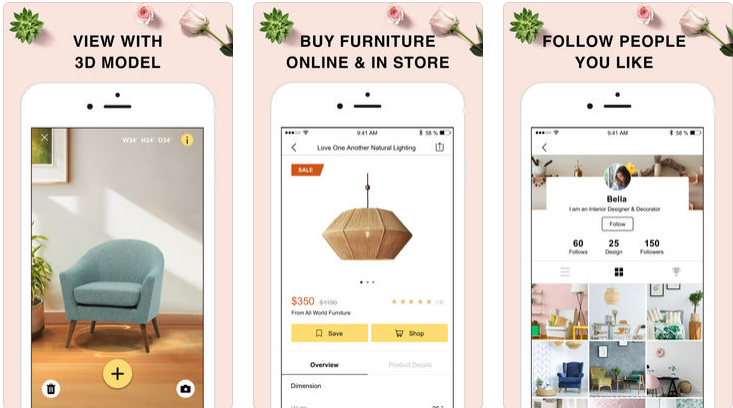
**Nhược điểm:**

* Chỉ hỗ trợ iOS (iPhone 6s trở lên)

1. **DecorMatters**

Là một ứng dụng game miễn phí cho người dùng thoải mái trang trí nội thất cho căn phòng thông qua các mẫu có sẵn (template) hoặc thông qua hình ảnh từ camera. Sau đó có thể chia sẻ thiết kế để nhận được các phản hồi từ cộng đồng





**Ưu điểm:**

* Có thể chia sẽ thiết kế

**Nhược điểm:**

* Chức năng AR giới hạn. Sử dụng model đặt trên một template có sẵn hoặc ảnh chụp phòng

**CHƯƠNG 3: CÁC CÔNG NGHỆ SỬ DỤNG**

Dựa trên nội dung chức năng, hệ thống sẽ chia thành 2 phần: web và di động. Phần di động có các tính năng của một ứng dụng mua sắm trực tuyến để hỗ trợ người mua có thể chọn, đặt hàng. Đồng thời hỗ trợ công nghệ AR giúp người dùng xem trước được sản phẩm. Phần web sẽ dùng cho người quản trị quản lý và điều hành các hoạt động của hệ thống

Như vậy, đối với server, nhóm cần một framework web có thể đảm nhận xử lý tất cả các request từ web và app, có thể xử lý thời gian thực. Về cơ sở dữ liệu, phải phù hợp với tính chất mua bán hàng và có thể lưu dữ liệu lớn. Ứng dụng có thể xây dựng native trên android. Về công nghệ thực tế ảo tăng cường, hiện có khá nhiều các lựa chọn trên thị trường, vì tính chất của ứng dụng, công nghệ đó phải hỗ trợ SLAM

Do đó, nhóm đã sử dụng những công nghệ sau:

* Framework web: Express trên nền NodeJS
* Hệ quản trị cơ sở dữ liệu: PostgreSQL
* Framework cho mobile: React-Native và Android Studio
* Công nghệ AR: Kudan AR SDK

1. **NodeJS**
   1. **Giới thiệu chung**

* Nodejs là một **nền tảng** (Platform) phát triển độc lập được xây dựng ở trên Javascript Runtime của Chrome mà chúng ta có thể xây dựng được các ứng dụng mạng một cách nhanh chóng và dễ dàng mở rộng.
* Nodejs được xây dựng và phát triển từ năm 2009, bảo trợ bởi công ty Joyent, trụ sở tại California, Hoa Kỳ.
* Phần Core bên dưới của Nodejs được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao.
* Nodejs tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực.
* Nodejs áp dụng cho các sản phẩm có lượng truy cập lớn, cần mở rộng nhanh, cần đổi mới công nghệ, hoặc tạo ra các dự án Startup nhanh nhất có thể.
  1. **Đặc điểm nổi bật**
* Đặc điểm nổi bật của Node.js là nó nhận và xử lý nhiều kết nối chỉ với một single-thread. Điều này giúp hệ thống tốn ít RAM nhất và chạy nhanh nhất khi không phải tạo thread mới cho mỗi truy vấn giống PHP. Ngoài ra, tận dụng ưu điểm non-blocking I/O của Javascript mà Node.js tận dụng tối đa tài nguyên của server mà không tạo ra độ trễ như PHP
* Shelling tools unix NodeJS sẽ tận dụng tối đa Unix để hoạt động. Tức là NodeJS có thể xử lý hàng nghìn Process và trả ra 1 luồng khiến cho hiệu xuất hoạt động đạt mức tối đa nhất và tuyệt vời nhất
* Streamming Data (Luồng dữ liệu) Các web thông thường gửi HTTP request và nhận phản hồi lại (Luồng dữ liệu). Giả xử sẽ cần xử lý 1 luồng giữ liệu cực lớn, NodeJS sẽ xây dựng các Proxy phân vùng các luồng dữ liệu để đảm bảo tối đa hoạt động cho các luồng dữ liệu khác.
* Ứng dụng Web thời gian thực Với sự ra đời của các ứng dụng di động & HTML 5 nên Node.js rất hiệu quả khi xây dựng những ứng dụng thời gian thực (real-time applications) như ứng dụng chat, các dịch vụ mạng xã hội
  1. **Lý do sử dụng**
* Các ứng dụng Nodejs được viết bằng javascript, ngôn ngữ này là một ngôn ngữ khá thông dụng. Theo tác giả của ngôn ngữ Javascript, Ryan Dahl: “Javascript có những đặc tính mà làm cho nó rất khác biệt so với các ngôn ngữ lập trình động còn lại, cụ thể là nó không có khái niệm về đa luồng, tất cả là đơn luồng và hướng sự kiện.”
* NodeJS được viết bằng ngôn ngữ Javascript nên việc xử lý JSON dễ dàng hơn.
* Nodejs chạy đa nền tảng phía Server, sử dụng kiến trúc hướng sự kiện Event-driven, cơ chế non-blocking I/O làm cho nó nhẹ và hiệu quả.
* Có thể chạy ứng dụng Nodejs ở bất kỳ đâu trên máy Mac – Window – Linux, hơn nữa cộng đồng Nodejs rất lớn và hoàn toàn miễn phí.
* NodeJS hỗ trợ giao thứ TCP vì vậy chúng ta dễ dàng xây dựng bất kỳ một phương thức custom nào.

1. **Framework Express**
   1. **Giới thiệu chung**

* Express js là một Framework nhỏ, nhưng linh hoạt được xây dựng trên nền tảng của Nodejs. Express hỗ trợ việc phát triển ứng dụng theo mô hình MVC, mô hình phổ biến cho việc lập trình web hiện nay. Về các package hỗ trợ: Expressjs có vô số các package hỗ trợ. Cho phép định nghĩa Middleware hỗ trợ cho việc tổ chức và tái sử dụng code. Định nghĩa routes và các request method đến server một cách dễ dàng. Hỗ trợ REST API.
  1. **Đặc điểm nổi bật**
* **Performance**
  + Express tính năng ứng dụng web như routing cơ bản, middleware, template engine và static files serving, từ đó hiệu suất I/O mạnh mẽ của Node.js không bị ảnh hưởng.
* **Generator** 
  + Express vẫn có generator tạo cấu trúc folder cụ thể. Sau khi cài đặt gói npm express-generator và tạo khung xương ứng dụng với lệnh generator, một folder ứng dụng với cấu trúc rõ ràng sẽ được tạo để giúp bạn quản lysimage, front-end static JavaScript, stylesheet file và HTML template files.
* **Middleware** 
  + Middleware về cơ bản chỉ là các hàm có truy cập toàn phần đến cả object request và response. Middleware sẽ áp đặt một số chỉ dẫn filtering trước khi giao quyền kiểm soát cho business l ogic thực sự hoặc cấp độ middleware tiếp theo.
* **Template Engine** 
  + Template engine cho phép lập trình viên nhúng biến backend vào file HTML, và khi được yêu cầu, template file sẽ được trả lại format HTML thông thường. Theo mặc định, express-generator sử dụng Pug (ban đầu được biết đến với tên Jade) template engine.
* **Database Integration** 
  + Express, là minimal framework, sẽ không bắt buộc database integaration. Khi sử dụng một một công nghệ lưu trữ dữ liệu cụ thể, dù là MySQL, MongoDB, PostgreSQL, Redis, ElasticSearch, ... Vấn đề chỉ nằm ở chỗ cài đặt gói npm làm database driver. Những database driver của bên thứ 3 này không phù hợp với cấu trúc thống nhất khi đi theo chỉ thị CRUB, dẫn đến việc thay đổi database phức tạp và tìm ẩn lỗi.
  1. **Lý do sử dụng**
* Server cần xử lý lượng request rất lớn từ cả web và ứng dụng Ngoài ra, phần ứng dụng cần đến tính cập nhật nhanh, liên tục trong chức năng cập nhật bản đồ, xem vị trí thành viên, gởi nhận dữ liệu ghi lại hành trình.

Mặc dù framework này khá mới, nhưng những tính năng đáp ứng được cho đề tài mà nó mang lại hết sức phù hợp cho vị trí trung tâm của hệ thống này. Express.js còn hỗ trợ cài thêm các gói khác như socket io, sequenlize, jwt để có thể hỗ trợ tối đa cho người lập tình.

* **Các gói package hỗ trợ:** 
  + **Sequelize** là một ORM dành cho Node.js và io.js. Nó hỗ trợ truy cập một cách dễ dàng đến PostgreSQL, MySQL, MariaDB, SQLite và MSSQL cùng với các tính năng như là relations, transaction, replication.
  + Để tạo 1 ánh xạ giữa 1 model và 1 table ta sử dụng phương thức define. Sequelize sẽ tự động tạo các attributes là createdAt và updatedAt.
  + Các loại relation gồm: One-To-One associations, One-To-Many associations, Belongs-To-Many associations giúp cho người lập trình tiết kiệm thời gian vì không phải viết lại các câu lệnh sql truyền thống.
  + Ngoài ra những tính năng trên còn có rất nhiều tính năng khác như: Transaction, Scope, Hooks.

* + **JSON Web Token (JWT)** là một chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và khép kín để truyền một cách an toàn thông tin giữa các bên dưới dạng đối tượng JSON. Thông tin này có thể được xác minh và đáng tin cậy vì nó có chứa chữ ký số. JWTs có thể được ký bằng một thuật toán bí mật (với thuật toán HMAC) hoặc một public / private key sử dụng mã hoá RSA.
  + JSON Web Token cũng là một cách hữu hiệu và bảo mật để trao đổi thông tin giữa nhiều ứng dụng, bởi vì JWT phải được ký (bằng cặp public / private key), không hoặc khó để mạo danh bằng JWT, ngoài ra, chữ ký cũng được tính toán dựa trên nội dung của header và nội dung payload, nhờ đó, bạn có thể xác thực được nội dung là nguyên bản, chưa được chỉnh sửa hoặc can thiệp. Tuy nhiên, một lưu ý hết sức quan trọng là do cấu trúc của JWT đơn giản nên JWT có thể dễ dàng bị decode, do vậy, không nên dùng JWT để transfer các thông tin nhạy cảm.

1. **PostgreSQL**
   1. **Giới thiệu chung**

* PostgreSQL là hệ quản trị cơ sở dữ liệu được viết theo hướng mã nguồn mở và rất mạnh mẽ. Hệ quản trị cơ sở dữ liệu này đã có hơn 15 năm phát triển, đồng thời kiến ​​trúc đã được kiểm chứng và tạo được lòng tin với người sử dụng về độ tin cậy, tính toàn vẹn dữ liệu, và tính đúng đắn. PostgreSQL có thể chạy trên tất cả các hệ điều hành, bao gồm cả Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), và Windows. Do nó hoàn toàn tuân thủ ACID, có hỗ trợ đầy đủ các foreign keys, joins, views, triggers, và stored procedures (trên nhiều ngôn ngữ). Hệ quản trị này còn bao gồm các kiểu dữ liệu SQL: 2008 như INTEGER, NUMBER, BOOLEAN, CHAR, VARCHAR, DATE INTERVAL, và TIMESTAMPs. PostgreSQL cũng hỗ trợ lưu trữ các đối tượng có kiểu dữ liệu nhị phân lớn, bao gồm cả hình ảnh, âm thanh, hoặc video. Hệ quản trị cơ sở dữ liệu này được sử dụng thông qua giao diện của các ngôn ngữ C / C + +, Java, .Net, Perl, Python, Ruby, Tcl, ODBC…
* Là một hệ quản trị cơ sở dữ liệu mạnh, PostgreSQL tự hào có các tính năng phức tạp như kiểm soát truy cập đồng thời nhiều phiên bản (MVCC), khôi phục dữ liệu tại từng thời điểm (Recovery), quản lý dung lượng bảng (tablespaces), sao chép không đồng bộ, giao dịch lồng nhau (savepoints), sao lưu trực tuyến hoặc nội bộ, truy vấn phức tạp và tối ưu hóa, và viết trước các khai báo để quản lý và gỡ lỗi. PostgreSQL hỗ trợ bộ ký tự quốc tế, hỗ trợ bảng mã nhiều byte, Unicode, và cho phép định dạng, sắp xếp và phân loại ký tự văn bản (chữ hoa, thường). PostgreSQL còn được biết đến với khả năng mở rộng để nâng cao cả về số lượng dữ liệu quản lý và số lượng người dùng truy cập đồng thời. Đã từng có những hệ thống PostgreSQL hoạt động trong môi trường thực tế thực hiện quản lý vượt quá 4 terabyte dữ liệu. Sau đây là một số thong tin giới hạn khả năng của PostgreSQL.
  1. **Đặc điểm nổi bật**
* Là hệ quản trị CSDL mạnh, cung cấp các tính năng phức tạp nhưng cung cấp truy vấn đồng thời đa phiên bản (có trước Oracle); khôi phục dữ liệu tại từng thời điểm (recovery), quản lý dung lượng bảng (tablespaces), sao chép không đồng bộ, giao dịch lồng nhau (savepoints), sao lưu trực tuyến hoặc nội bộ, truy vấn phức tạp và tối ưu hóa.
* Có nhiều ngôn ngữ hộ trợ C++/C/Java/Perl, PHP, Ruby, ... nên có thể khai thác nhiều thư viện giao tiếp với hàng trăm chức năng từ chức năng cơ bản như chuỗi số đến thuật toán phức tạp như mã hóa, và đặc biệt tương thích với Oracle. Trigger và các thủ tục có thể được viết thêm bằng C thêm vào CSDL như là một thư viện, cho phép linh hoạt mở rộng khả năng của mình.
* Cho phép các nhà phát triển có thể sáng tạo các kiểu dữ liệu tiên tiến của riêng mình từ các dữ liệu gốc ban đầu như địa chỉ vật lý, địa chỉ mạng. Cũng có thể phát triển plugin riêng như thêm 1 hàm tối ưu hóa thay cho 1 phần của hệ thống.
* Hơn nữa, PostgreSQL có thể dùng trong nhiều trường hợp khác, chẳng hạn như tạo ra các khả năng mới như: Kiểu dữ liệu, Hàm, Toán tử, Hàm tập hợp, Phương pháp liệt kê.
  1. **Lý do sử dụng:**
* PostgreSQL sở hữu một hệ tính năng đa dạng giúp hỗ trợ các nhà phát triển xây dựng app, các nhà quản trị bảo vệ toàn vẹn dữ liệu, và tạo ra một môi trường chịu lỗi fault-tolerant giúp bạn quản lý dữ liệu bất kể tập dữ liệu lớn hay nhỏ. Bên cạnh hệ thống nguồn mở và miễn phí, PostgreSQL cũng có khả năng mở rộng tuyệt vời. Ví dụ, bạn có thể định nghĩa các kiểu dữ liệu riêng của bạn, xây dựng các hàm tùy chỉnh, hay viết mã từ các ngôn ngữ lập trình khác nhau mà không cần biên dịch lại cơ sở dữ liệu!
* PostgreSQL tuân theo tiêu chuẩn SQL nhưng không mâu thuẫn với các tính năng truyền thống hay có thể dẫn đến các quyết định kiến trúc gây hại. Nhiều tính năng theo tiêu chuẩn SQL được hỗ trợ.

1. **React Native**
   1. **Giới thiệu chung**

* React Native là một framework JavaScript dùng để viết các ứng dụng iOS và Android. Nó dựa trên React- thư viện JavaScript của Facebook dùng để xây dựng giao diện người dùng, nhưng thay vì hướng đến browser, nó hướng đến nền tảng di động. Nói cách khác, một lập trình viên web có thể viết ứng dụng di động dễ dàng, nhanh chóng trên một framework quen thuộc.
* Thư viện React Native được công bố bởi Facebook vào năm 2015. Đã có 20 phiên bản được công bố. Phiên bản hiện tại là v0.38.
* React Native cũng tương tự như React. React là một thư viện JavaScript xây dựng giao diện web, được phát triển và open source bởi Facebook vào năm 2013. Chỉ khác là dành cho mobile. Một vài sự khác biệt trong tên của tag sử dụng vẽ giao diện.
  1. **Đặc điểm nổi bật**
* Chỉ là React và Javascrip: React Native sử dụng hoàn toàn React và javascript để xây dựng ứng dụng. Ngoài ra, nền tảng này cũng cần thêm một số javascript packages, được lấy từ NPM.
* Các thành phần native mạnh mẽ: Xcode cung cấp lập trình viên Swift và Objective C rất nhiều thành phần để xây dựng ứng dụng. Những thành phần đó cũng được xây dựng trong React Native. Những thành phần cơ bản như Images, ListViews, ScrollViews, Slider, TabBars, v.v ... có thể thêm vào ứng dụng một cách dễ dàng. Nhiều iOS API cũng được liên kết vào React Native.
* Các thành phần bên thứ 3 mạnh mẽ: Bên cạnh lượng lớn thành phần native nhóm phát triển React Native cung cấp, cộng đồng phát triển React Native cũng đóng góp thêm nhiều thành phần khác. Các trang như React Components và React Parts chứa đựng hàng tá các thành phần thêm vào cho React, thực thi những mẫu thiết kế iOS riêng biệt.
* CSS-like Stylesheet: để cung cấp thẩm mỹ cho ứng dụng, React Native sử dụng một phiên bản rút gọn CSS hỗ trợ tạo kiểu cho thành phần. Hầu hết các màu sắc, loại, thuộc tính cơ bản có thể sử dụng trong React Native. Dòng xử lý cũng tương tự như HTML/CSS. Chỉ có khác là tất cả code style được đặc trong một file javascript. Để quy định vị trí của các thành phần trên màn hình, React Native sử dụng một phiên bản nhẹ được điều chỉnh của Flexbox, cho phép người dùng tạo ra một layout có thể mở rộng theo kích thước thiết bị.
* Kiểm thử: Hầu hết những ứng dụng native, việc triển khai và kiểm thử được thực hiện trên máy ảo. React Native sử dụng Google Chrome, cho phép lập trình viên debug ứng dụng bằng công cụ kiểm tra thành phần web trên Chrome. Lỗi, log, và cảnh báo đều hiện trên trên Chrome. React Native cũng có các phương thức để kiểm tra ứng dụng trên thiết bị thực. Đơn giản chỉ cần thay đổi cài đặt một chút trong file AppDelegate.m.

* 1. **Lý do sử dụng**
* Nhanh chóng: đối với lập trình viên web muốn tạo ứng dụng di động thì không gì tốt hơn là sử dụng một framework quen thuộc mà không cần phải đảm nhận thêm công việc của lập trình viên di động.
* Đa nền tảng: thật hiệu quả khi viết một lần có thể chạy trên nhiều hệ điều hành di động.
* So với các framework tạo ứng dụng di động đa nền tảng khác như Xamarin, PhoneGap, React Native là một công nghệ mới. Tuy nhiên, nó có rất nhiều ưu điểm vượt trội như là thời gian phát triển nhanh, trải nghiệm người dùng tốt, tải nhanh, đễ học và cộng đồng lớn. Nhiều bài viết khen ngợi framework này và khuyến khích sử dụng.

**CHƯƠNG 4: XÁC ĐỊNH YÊU CẦU**

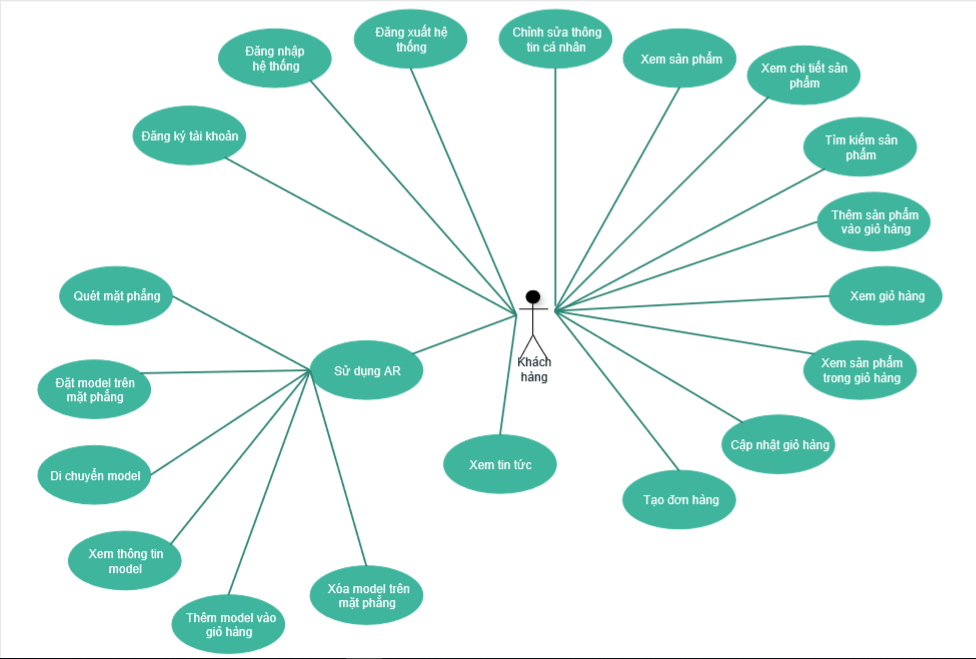
* 1. **Danh sách yêu cầu**

Danh sách các yêu cầu chức năng của hệ thống:

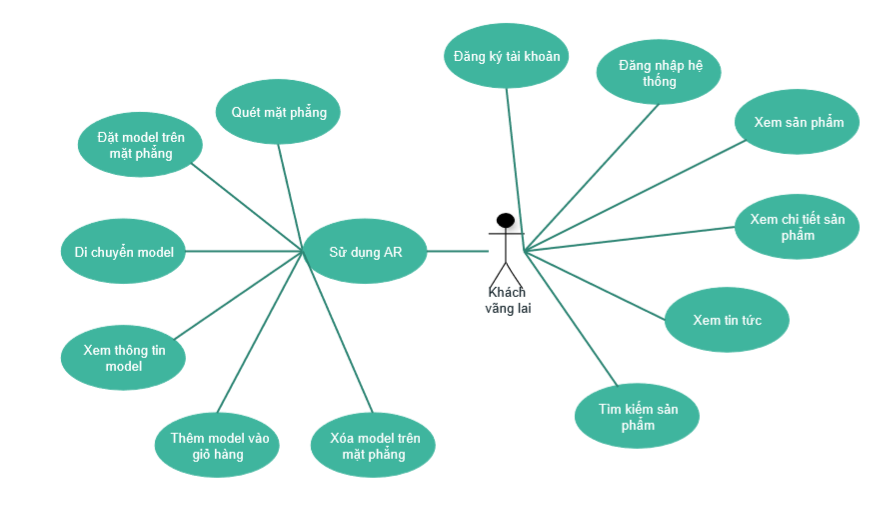
* Xem danh sách sản phẩm
* Tìm kiếm sản phẩm
* Xem chi tiết sản phẩm
* Nhận diện mặt phẳng
* Đặt mô hình trên mặt phẳng
* Di chuyển/Xoay/Xóa mô hình
* Lưu thiết kế
* Thêm sản phẩm vào giỏ hàng
* Xem chi tiết đơn hàng
* Đặt đơn hàng
  1. **Mô tả yêu cầu**
  2. **Xem danh sách sản phẩm**

**CHƯƠNG 5: PHÂN TÍCH THIẾT KẾ**

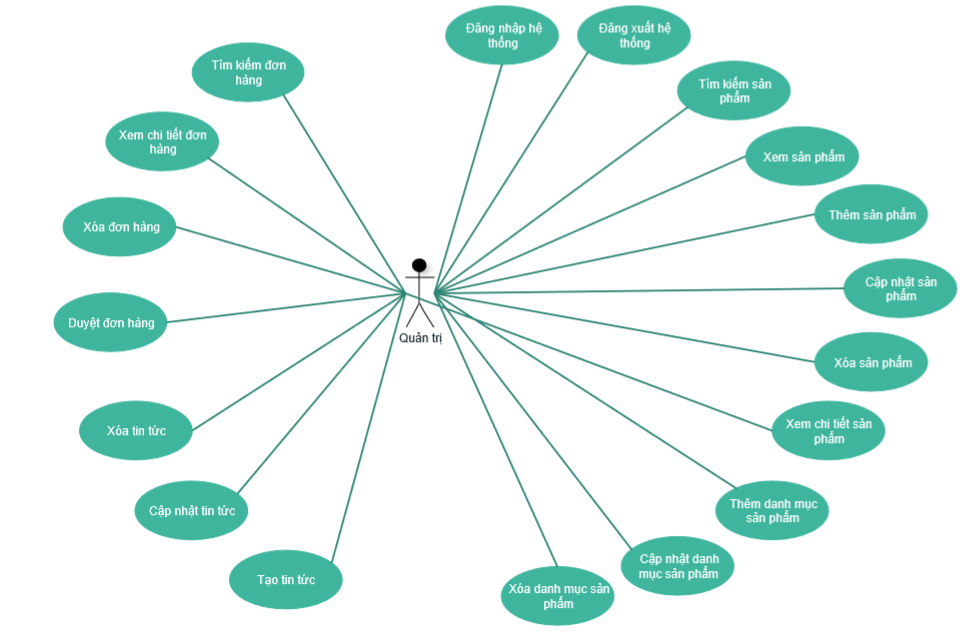
1. **Phân tích yêu cầu**
   1. **Sơ đồ usecase**



**Hình 5.1 Sơ đồ usecase của Khách hàng**



*Hình 5.2 Sơ đồ usecase của Khách vãng lai*



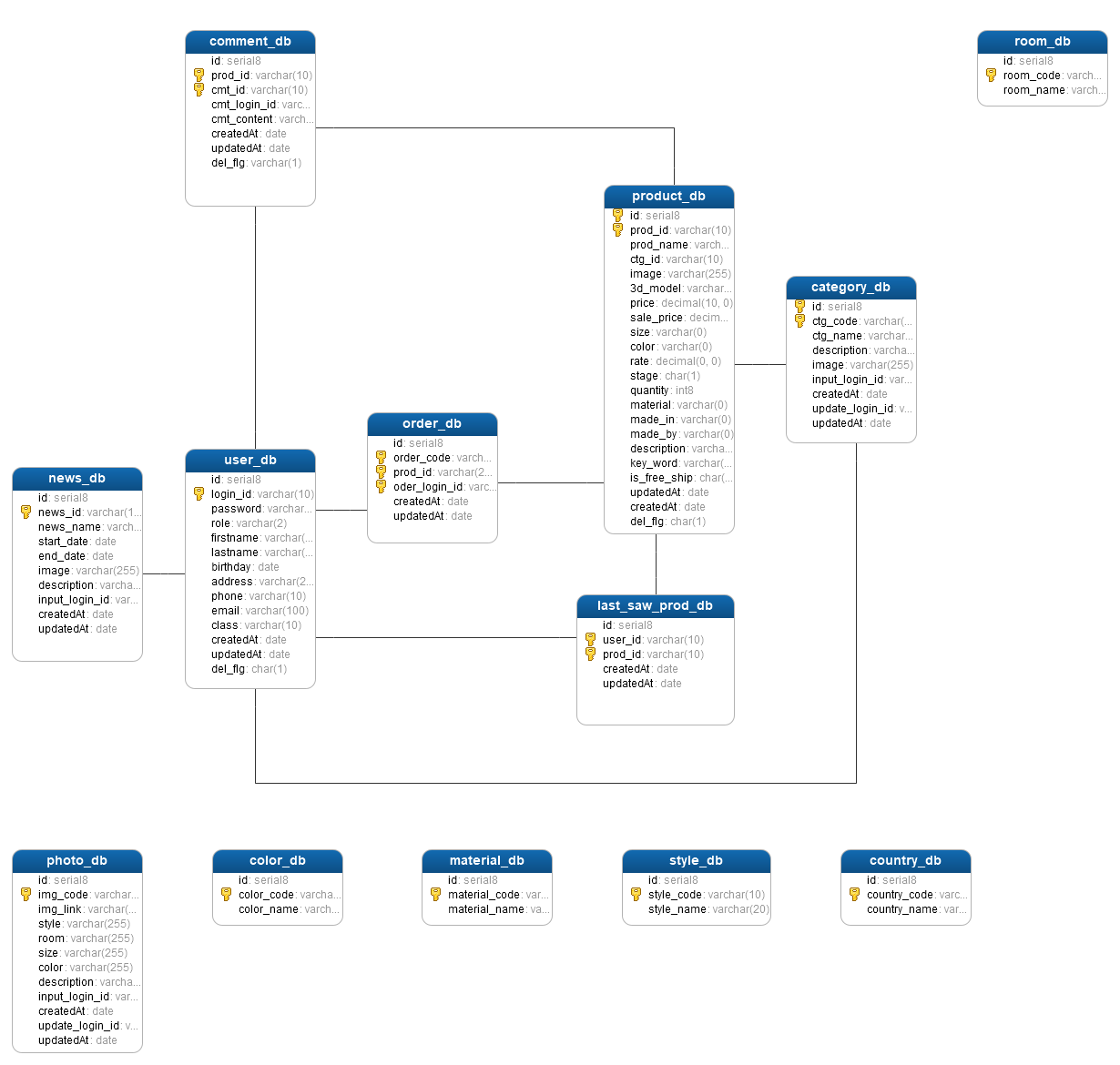
*Hình 5.3 Sơ đồ usecase của Quản trị viên*

* 1. **Danh sách người dùng**

|  |  |
| --- | --- |
| **Loại người dùng** | **Mô tả** |
| Admin (Người quản trị) | Là người quản trị hệ thống, admin đăng nhập tài khoản có toàn quyền sử dụng, cấu hình, thêm xóa sửa cơ sở dữ liệu, xem chi tiết các đơn hàng, quản lý thông tin danh mục, sản phẩm, tin tức, đơn hàng… |
| Customer (Khách hàng) | Là khách hàng cần mua sản phẩm. Họ có thể chọn các mẫu ưa thích. Họ dạo quanh cửa hàng, mua từng sản phẩm, tìm kiếm, đặt sản phẩm vào giỏ hàng, tạo đơn hàng, đánh giá, phản hồi về sản phẩm, sử dụng những mô hình 3d của sản phẩm để thiết kế cho căn phòng, chụp ảnh và chia sẻ thiết kế đó. Customer là khách hàng thành viên đã đăng ký tài khoản và mật khẩu để tiện cho giao dịch |
| Free User (Khách vãng lai) | Là những người chỉ xem sản phẩm trong danh mục nhưng không mua, có thể xem quảng cáo, giá cả, chất lượng sản phẩm. Khách vãng lai muốn mua hàng phải đăng ký thành viên. |

* 1. **Mô tả usecase**
     1. **Đăng ký thành viên**
        + - **Tóm tắt**
* Usecase cho phép actor đăng ký để trở thành thành viên.
* Khách vãng lai là actor cho usecase này
  + - * + **Dòng sự kiện**
* Usecase bắt đầu khi actor chọn chức năng “Đăng ký”
  + - * + **Dòng sự kiện chính**
* Actor chọn chức năng đăng ký
* Hệ thống hiển thị mục đăng ký thành viên
* Actor điền vào các thông tin: Tên đăng nhập, mật khẩu, xác nhận mật khẩu, địa chỉ email, điện thoại.
* Actor chọn nút đăng ký
* Hệ thống lưu toàn bộ thông tin mà actor vừa xác nhận xuống database
* Hệ thống hiển thị màn hình của thành viên
  + - * + **Các dòng sự kiện khác**
* Nếu actor không điền đầy đủ các thông tin bắt buộc thì hệ thống không cho đăng ký và thông báo bắt buộc phải điền đầy đủ thông tin này
  + - * + **Các yêu cầu đặt biệt**
* Không có
  + - * + **Điều kiện tiên quyết**
* Không có
  + - * + **Điều kiện hoàn tất**
* Có thông báo cho actor biết tình trạng đăng ký có được hoàn thành hay không
* Phải chuyển đến giao diện đăng nhập thành công
  + - * + **Điểm mở rộng**
* Sửa thông tin thành viên
  + 1. **Đăng nhập thành viên**
       - * **Tóm tắt**
* Usecase này cho phép actor đăng nhập vào hệ thống với vai trò là khách hàng
* Khách hàng và Khách vãng lai là actor cho usecase này.
  + - * + **Dòng sự kiện**
* Usecase bắt đầu khi actor chọn chức năng đăng nhập vào hệ thống với vai trò khách hàng.
  + - * + **Dòng sự kiện chính**
* Actor chọn chức năng đăng nhập
* Hệ thống hiển thị form gồm tên đăng nhập và mật khẩu
* Actor nhập các thông tin trên form và chọn đăng nhập
* Hệ thống kiểm tra tên truy cập và mật khẩu, xác nhận là đúng và cho phép actor đăng nhập vào hệ thống với vai trò khách hàng.
  + - * + **Các dòng sự kiện khác**
* Nếu actor nhập tên truy cập hoặc mật khẩu sai thì hệ thống sẽ báo lỗi.
  + - * + **Các yêu cầu đặt biệt**
* Không có
  + - * + **Điều kiện tiên quyết**
* Không có
  + - * + **Điều kiện hoàn tất**
* Hệ thống thông báo việc đăng nhập có thành công hay không.
  + - * + **Điểm mở rộng**
* Không có
  + 1. **Chỉnh sửa thông tin thành viên**
       - * **Tóm tắt**
* Usecase này cho phép actor chỉnh sửa những thông tin của mình trong hệ thống.
* Khách hàng là actor cho usecase này.
  + - * + **Dòng sự kiện**
* Usecase bắt đầu khi actor chọn chức năng chỉnh sửa thông tin thành viên.
  + - * + **Dòng sự kiện chính**
* Actor chọn chức năng chỉnh sửa thông tin thành viên
* Hệ thống xác định tên tài khoản và mật khẩu của actor đang truy cập.
* Hệ thống tìm trong database những thông tin của actor tương ứng với tên đăng nhập và hiển thị các thông tin đó trên một form.
* Actor chỉnh sửa lại thông tin
* Actor chọn cập nhật hoặc thoát
  + - * + **Các dòng sự kiện khác**
* Nếu actor chọn cập nhật thì hệ thống sẽ cập nhật lại những thông tin mà actor vừa chỉnh sửa vào database
* Nếu actor chọn thoát thì usecase kết thúc
  + - * + **Các yêu cầu đặt biệt**
* Không có
  + - * + **Điều kiện tiên quyết**
* Actor đã đăng nhập vào hệ thống
  + - * + **Điều kiện hoàn tất**
* Hệ thống phải hiển thị thông báo cho actor biết về quá trình chỉnh sửa thông tin có thành công hay không.
  + - * + **Điểm mở rộng**
* Không có.
  + 1. **Bỏ sản phẩm vào giỏ hàng**
       - * **Tóm tắt**
* Usecase này cho phép tạo ra giỏ hàng ảo cho khách hàng.
* Giỏ hàng ảo sẽ chứa các sản phẩm mà khách hàng chọn mua
* Khách hàng là actor cho usecase này.
  + - * + **Dòng sự kiện**
* Usecase bắt đầu khi actor chọn mua một sản phẩm bằng cách click button thêm vào.
  + - * + **Dòng sự kiện chính**
* Actor chọn nút thêm vào dưới từng sản phẩm
* Hệ thống lấy mã sản phẩm vừa được chọn
* Hệ thống lấy thông tin về khách hàng (mã khách hàng)
* Hệ thống tìm mã giỏ hàng đã tồn tại tỏng hệ thống hay chưa
* Nếu giỏ hàng chưa được tạo thì tạo ra một mã giỏ hàng, lưu các thông tin vào database: mã giỏ hàng, mã khách hàng, mã sản phẩm, số lượng sản phẩm, ngày tạo.
* Nếu giỏ hàng đã tồn tại, hệ thống lưu những thông tin vào database: mã giỏ hàng, mã khách hàng, mã sản phẩm, số lượng sản phẩm, ngày tạo.
  + - * + **Các dòng sự kiện khác**
* Actor chưa đăng nhập thành viên, khi actor nhấn nút thêm vào thì hệ thống yêu cầu actor đăng nhập thành viên. Khách actor hoàn tất các thủ tục đó thì usecase kết thúc
  + - * + **Các yêu cầu đặt biệt**
* Không có
  + - * + **Điều kiện tiên quyết**
* Actor phải đăng nhập vào hệ thống với vai trò là khách hàng
  + - * + **Điều kiện hoàn tất**
* Không có
  + - * + **Điểm mở rộng**
* Không có
  + 1. **Tìm sản phẩm**
       - * **Tóm tắt**
* Usecase này cho phép actor tìm một sản phẩm trong cửa hàng
* Khách hàng và Khách vãng lai là actor cho usecase này
  + - * + **Dòng sự kiện**
* Usecase bắt đầu khi actor chọn chức năng “Tìm kiếm sản phẩm”
  + - * + **Dòng sự kiện chính**
* Actor nhập vào từ khóa cần tìm (tên sản phẩm hoặc thông tin liên quan)
* Actor chọn chức năng tìm kiếm
* Hệ thống tìm trong database những sản phẩm có thông tin tương ứng với từ khóa actor nhập vào.
* Hệ thống hiển thị các sản phẩm tìm được
  + - * + **Các dòng sự kiện khác**
* Nếu không có sản phẩm nào được tìm thấy thì hệ thống sẽ thông báo cho actor biết là không tìm thấy sản phẩm
* Hệ thống quay lại đầu usecase
  + - * + **Các yêu cầu đặt biệt**
* Không có
  + - * + **Điều kiện tiên quyết**
* Không có
  + - * + **Điều kiện hoàn tất**
* Tất cả các sản phẩm tìm được được hiển thị
  + - * + **Điểm mở rộng**
* Không có
  + 1. **Xem sản phẩm**
       - * **Tóm tắt**
* Usecase này cho phép actor xem tất cả các sản phẩm có trong danh mục
* Khách hàng và Khách vãng lai là actor cho usecase này.
  + - * + **Dòng sự kiện**
* Usecase bắt đầu khi actor chọn một danh mục sản phẩm.
  + - * + **Dòng sự kiện chính**
* Actor chọn một danh mục sản phẩm
* Hệ thống lấy mã danh mục mà actor chọn
* Hệ thống hiển thị các sản phẩm thuộc danh mục đó.
  + - * + **Các dòng sự kiện khác**
* Actor có thể xem chi tiết sản phẩm hoặc bỏ sản phẩm được chọn vào giỏ hàng
  + - * + **Các yêu cầu đặt biệt**
* Không có
  + - * + **Điều kiện tiên quyết**
* Không có
  + - * + **Điều kiện hoàn tất**
* Tất cả các sản phẩm tìm được được hiển thị
  + - * + **Điểm mở rộng**
* Xem chi tiết sản phẩm
* Bỏ sản phẩm vào giỏ hàng
  + 1. **Xem chi tiết sản phẩm**
       - * **Tóm tắt**
* Usecase này cho phép actor xem chi tiết sản phẩm mà actor quan tâm
* Khách hàng và Khách vãng lai là actor cho usecase này.
  + - * + **Dòng sự kiện**
* Usecase bắt đầu khi actor chọn một sản phẩm trong danh sách sản phẩm.
  + - * + **Dòng sự kiện chính**
* Actor chọn một sản phẩm có trong danh mục
* Hệ thống lưu lại mã sản phẩm mà khách hàng vừa chọn
* Hệ thống tìm trong database những thông tin ứng với mã sản phẩm đó: Tên sản phẩm, hình ảnh minh họa, giá sản phẩm, loại sản phẩm, thông tin chi tiết.
  + - * + **Các dòng sự kiện khác**
* Không có
  + - * + **Các yêu cầu đặt biệt**
* Không có
  + - * + **Điều kiện tiên quyết**
* Không có
  + - * + **Điều kiện hoàn tất**
* Không có
  + - * + **Điểm mở rộng**
* Bỏ sản phẩm vào giỏ hàng
  + 1. **Tạo đơn đặt hàng**
       - * **Tóm tắt**
* Usecase này cho phép actor tạo đơn đặt hàng.
* Khách hàng là actor cho usecase này.
  + - * + **Dòng sự kiện**
* Usecase bắt đầu khi actor chức năng tạo đơn đặt hàng.
  + - * + **Dòng sự kiện chính**
* Actor chọn chức năng tạo đơn đặt hàng
* Hệ thống hiển thị một form yêu cầu actor nhập các thông tin của actor, sau đó hệ thống load thông tin của khách hàng lên form
* Actor điền các thông tin được yêu cầu: họ và tên, địa chỉ email, điện thoại, địa chỉ.
* Actor chọn đặt hàng hoặc thoát
  + - * + **Các dòng sự kiện khác**
* Nếu actor chọn đặt hàng, hệ thống xác nhận thông tin từ session về giỏ hàng và khách hàng, các thông tin từ đơn đặt hàng về người mua và lưu xuống database. Sau đó hệ thống tạo các đơn đặt hàng tương ứng với các sản phẩm có trong giỏ hàng và lưu thông tin đơn đặt hàng vào database
* Nếu actor chọn thoát thì usecase kết thúc
* Actor có thể xem giỏ hàng ngay khi lập đơn đặt hàng để xác nhận lại các thông tin về những sản phẩm mà actor muốn đặt hàng
* Actor có thể chỉnh sửa số lượng mua của sản phẩm hay bỏ bất kỳ sản phẩm nào trong giỏ hàng
* Nếu actor nhập không đầy đủ các thông tin được yêu cầu, hệ thống sẽ thông báo lỗi buộc actor nhập đầy đủ các thông tin trước khi ấn nút đặt hàng.
  + - * + **Các yêu cầu đặt biệt**
* Không có
  + - * + **Điều kiện tiên quyết**
* Actor đã đăng nhập vào hệ thống với vai trò khách hàng
* Giỏ hàng phải tồn tại ít nhất 1 sản phẩm
  + - * + **Điều kiện hoàn tất**
* Hệ thống phải hiển thị một thông báo cho actor biết về những thông tin cần làm tiếp theo sau khi đặt hàng
* Nếu việc đặt hàng thất bại thì hệ thống cũng phải thông báo cho actor
  + - * + **Điểm mở rộng**
* Xem giỏ hàng
* Giảm sản phẩm trong giỏ hàng
* Sửa số lượng sản phẩm có trong giỏ hàng

1. **Thiết kế cơ sở dữ liệu**
   1. **Lượt đồ dữ liệu khái niệm**

****

* 1. **Mô tả các đối tượng dữ liệu**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| category\_db | | | | |
| Mô tả: Lưu thông tin danh mục sản phẩm | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | ctg\_code | character varying (10) | Khác null Khóa chính | Mã danh mục |
| 3 | ctg\_name | character varying(255) | Khác null | Tên danh mục |
| 4 | description | character varying (1000) | Khác null | Mô tả cho danh mục |
| 5 | image | character varying (255) | Khác null | Hình ảnh đại diện cho danh mục |
| 6 | input\_login\_id | character varying (10) | Khác null Khóa ngoại | Tên tài khoản nhập danh mục |
| 7 | createdAt | date | Khác null | Ngày tạo (được tạo bởi Sequelize) |
| 8 | update\_login\_id | character varying (10) | Khác null  Khóa ngoại | Tên tài khoản cập nhập danh mục |
| 9 | updatedAt | date | Khác null | Ngày cập nhật (được tạo bởi Sequelize) |
| 10 | del\_flg | character (1) | Khác null | Cờ xóa |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| color\_db | | | | |
| Mô tả: Lưu thông tin màu sắc | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | color\_code | character varying (10) | Khác null Khóa chính | Mã màu sắc |
| 3 | color\_name | character varying(10) | Khác null | Tên màu sắc |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| comment\_db | | | | |
| Mô tả: Lưu thông tin đánh giá sản phẩm | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | prod\_id | character varying (10) | Khác null Khóa chính  Khóa ngoại | Mã sản phẩm |
| 3 | cmt\_id | character varying(255) | Khác null  Khóa chính | Mã bình luận |
| 4 | cmt\_login\_id | character varying (1000) | Khác null  Khóa ngoại | Tên tài khoản bình luận |
| 5 | cmt\_content | character varying (255) | Khác null | Nội dung bình luận |
| 6 | createdAt | character varying (10) | Khác null | Ngày tạo (được tạo bởi Sequelize) |
| 7 | updatedAt | date | Khác null | Ngày cập nhật (được tạo bởi Sequelize) |
| 8 | del\_flg | character (1) | Khác null | Cờ xóa |
| country\_db | | | | |
| Mô tả: Lưu thông tin quốc gia sản xuất | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | country\_code | character varying (10) | Khác null Khóa chính | Mã quốc gia |
| 3 | county\_name | character varying(255) | Khác null | Tên quốc gia |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| last\_saw\_prod\_db | | | | |
| Mô tả: Lưu thông tin sản phẩm xem gần nhất | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | user\_id | character varying (10) | Khác null Khóa chính  Khóa ngoại | Tên tài khoản |
| 3 | prod\_id | character varying(255) | Khác null  Khóa chính  Khóa ngoại | Mã sản phẩm |
| 4 | createdAt | date | Khác null | Ngày tạo (được tạo bởi Sequelize) |
| 5 | updatedAt | date | Khác null | Ngày cập nhật (được tạo bởi Sequelize) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| material\_db | | | | | |
| Mô tả: Lưu thông tin chất liệu sản phẩm | | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | material\_code | character varying (10) | Khác null Khóa chính | Mã chất liệu |
| 3 | material\_name | character varying(255) | Khác null | Tên chất liệu |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| news\_db | | | | |
| Mô tả: Lưu thông tin tin tức | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | news\_id | character varying (10) | Khác null Khóa chính | Mã số tin tức |
| 3 | news\_name | character varying(255) | Khác null | Tiêu đề của tin tức |
| 4 | start\_date | date | Khác null | Ngày bắt đầu |
| 5 | end\_date | date | Khác null | Ngày kết thúc |
| 6 | image | character varying (255) | Khác null | Hình ảnh đại diện cho tin tức |
| 7 | description | character varying (1000) | Khác null | Mô tả |
| 8 | input\_login\_id | character varying (10) | Khác null  Khóa ngoại | Tên tài khoản tạo tin tức |
| 9 | createdAt | date | Khác null | Ngày tạo (được tạo bởi Sequelize) |
| 10 | update\_login\_id | character (10) | Khác null  Khóa ngoại | Tên tài khoản cập nhật tin tức |
| 11 | updatedAt | date | Khác null | Ngày cập nhật (được tạo bởi Sequelize) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| order\_db | | | | |
| Mô tả: Lưu thông tin đơn hàng | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | order\_code | character varying (10) | Khác null Khóa chính | Mã đơn hàng |
| 3 | prod\_id | character varying(255) | Khác null  Khóa chính  Khóa ngoại | Mã sản phẩm |
| 4 | order\_login\_id | character varying (1000) | Khác null  Khóa chính  Khóa ngoại | Tài khoản đặt hàng |
| 5 | createdAt | date | Khác null | Ngày tạo (được tạo bởi Sequelize) |
| 6 | updatedAt | date | Khác null | Ngày cập nhật (được tạo bởi Sequelize) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| photo\_db | | | | |
| Mô tả: Lưu thông tin hình ảnh | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | img\_code | character varying (10) | Khác null Khóa chính | Mã hình ảnh |
| 3 | img\_link | character varying(255) | Khác null | Đường dẫn đến hình ảnh |
| 4 | style | character varying (255) | Khác null | Phong cách của nội dung ảnh |
| 5 | room | character varying (255) | Khác null | Loại phòng của nội dung ảnh |
| 6 | size | character varying (255) | Khác null | Kích cỡ của nội dung ảnh |
| 7 | color | character varying (255) | Khác null | Màu sắc của nội dung ảnh |
| 8 | description | character varying (1000) | Khác null | Mô tả cho nội dung ảnh |
| 9 | createdAt | date | Khác null | Ngày tạo (được tạo bởi Sequelize) |
| 10 | updatedAt | date | Khác null | Ngày cập nhật (được tạo bởi Sequelize) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| product\_db | | | | |
| Mô tả: Lưu thông tin sản phẩm | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | prod\_id | character varying (10) | Khác null Khóa chính | Mã sản phẩm |
| 3 | prod\_name | character varying(255) | Khác null | Tên sản phẩm |
| 4 | ctg\_id | character varying (10) | Khác null  Khóa ngoại | Mã danh mục sản phẩm |
| 5 | image | character varying (255) | Khác null | Hình ảnh đại diện cho sản phẩm |
| 6 | 3d\_model | character varying (255) | Khác null | Đường dẫn đến mô hình 3D của sản phẩm |
| 7 | price | decimal | Khác null | Giá sản phẩm |
| 8 | sale\_price | decimal | Khác null | Giá sau khi giảm giá |
| 9 | size | character varying (255) | Khác null | Kích cỡ của sản phẩm |
| 10 | color | character varying (255) | Khác null | Màu sắc của sản phẩm |
| 11 | rate | decimal | Khác null | Điểm đánh giá của sản phẩm |
| 12 | stage | character (1) | Khác null | Trạng thái của sản phẩm |
| 13 | quantity | int | Khác null | Số lượng sản phẩm trong kho |
| 14 | material | character varying (255) | Khác null | Chất liệu của sản phẩm |
| 15 | made\_in | character varying (255) | Khác null | Xuất xứ của sản phẩm |
| 16 | made\_by | character varying (255) | Khác null | Công ty làm ra sản phẩm |
| 17 | desciption | character varying (1000) | Khác null | Mô tả sản phẩm |
| 18 | key\_word | character varying (255) | Khác null | Từ khóa dùng để tìm kiếm |
| 19 | is\_free\_ship | character (1) | Khác null | Sản phẩm có được miễn phí vận chuyển không |
| 20 | createdAt | date | Khác null | Ngày tạo (được tạo bởi Sequelize) |
| 21 | updatedAt | date | Khác null | Ngày cập nhật (được tạo bởi Sequelize) |
| 22 | del\_flg | character (1) | Khác null | Cờ xóa |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| room\_db | | | | |
| Mô tả: Lưu thông tin loại phòng | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | room\_code | character varying (10) | Khác null Khóa chính | Mã phòng |
| 3 | room\_name | character varying(255) | Khác null | Tên phòng |

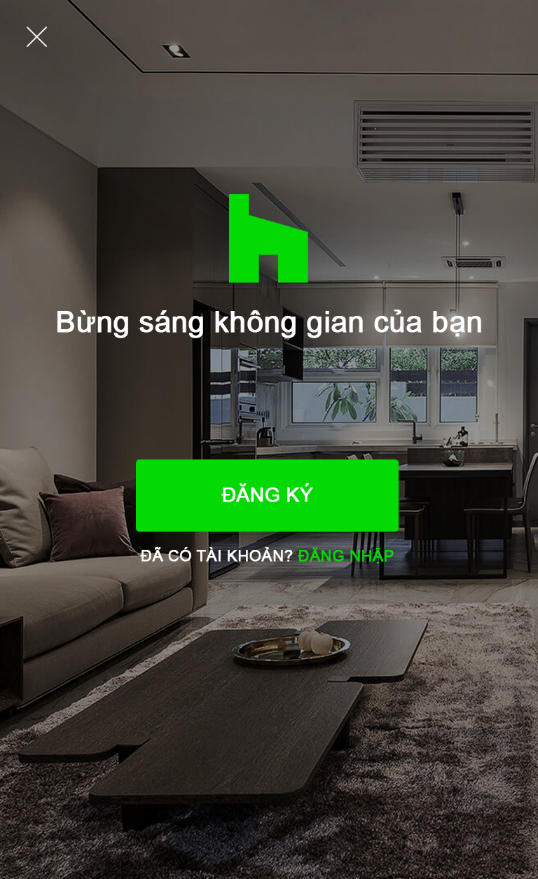
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| style\_db | | | | |
| Mô tả: Lưu thông tin phong cách | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | style\_code | character varying (10) | Khác null Khóa chính | Mã phong cách |
| 3 | style\_name | character varying(255) | Khác null | Tên phong cách |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| user\_db | | | | |
| Mô tả: Lưu thông tin người dùng | | | | |
| STT | **Tên** | **Kiểu** | **Ràng buộc** | **Ý nghĩa** |
| 1 | id | bigint | Khác null | Mã số (được tạo bởi Sequelize) |
| 2 | login\_id | character varying (10) | Khác null Khóa chính | Tên tài khoản |
| 3 | password | character varying (20) | Khác null | Mật khẩu |
| 4 | role | character varying (2) | Khác null | Vai trò của tài khoản |
| 5 | firstname | character varying (255) | Khác null | Tên |
| 6 | lastname | character varying (255) | Khác null | Họ |
| 7 | birthday | date |  | Ngày sinh |
| 8 | address | character varying (255) |  | Địa chỉ |
| 9 | phone | character varying (10) |  | Số điện thoại |
| 10 | email | character varying (100) |  | Địa chỉ email |
| 11 | class | character varying (10) | Khác null | Lớp người dùng |
| 12 | createdAt | date | Khác null | Ngày tạo (được tạo bởi Sequelize) |
| 13 | updatedAt | date | Khác null | Ngày cập nhật (được tạo bởi Sequelize) |
| 14 | del\_flg | character (1) | Khác null | Cờ xóa |

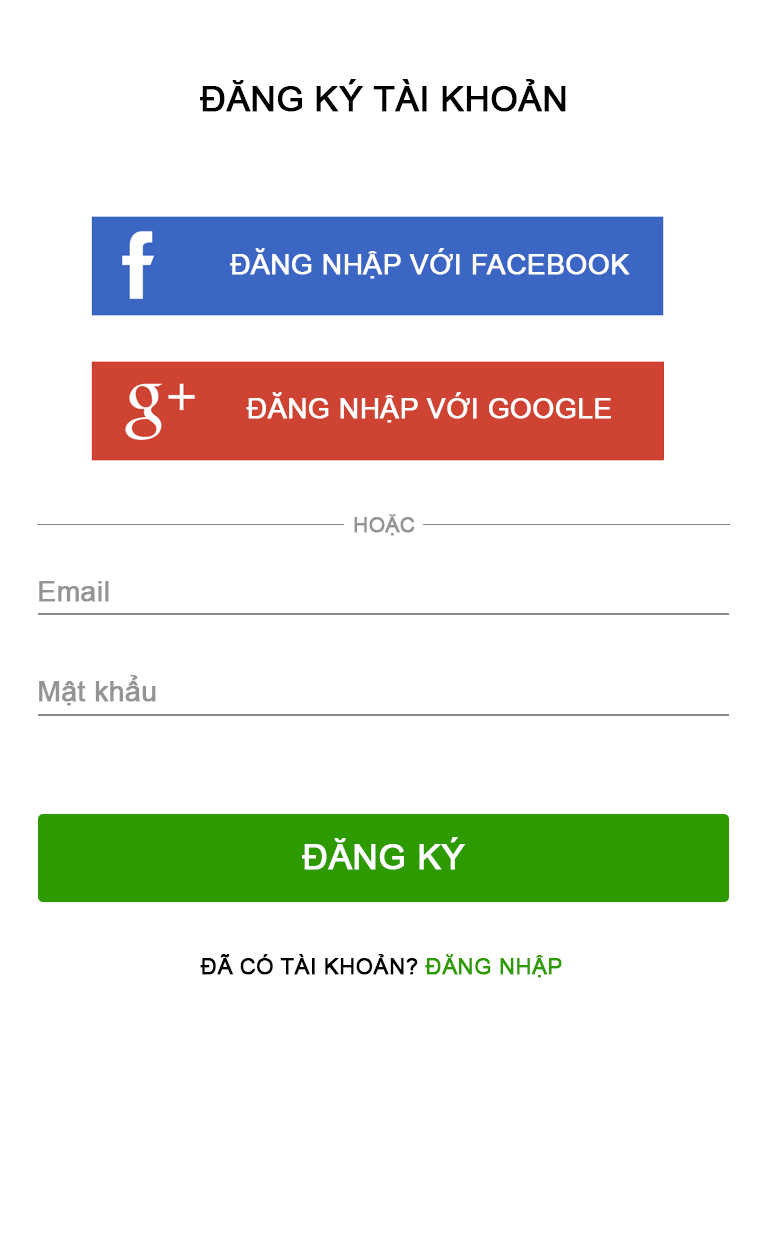
1. **Thiết kế kiến trúc phần mềm**

**3.1. Kiến trúc tổng thể**

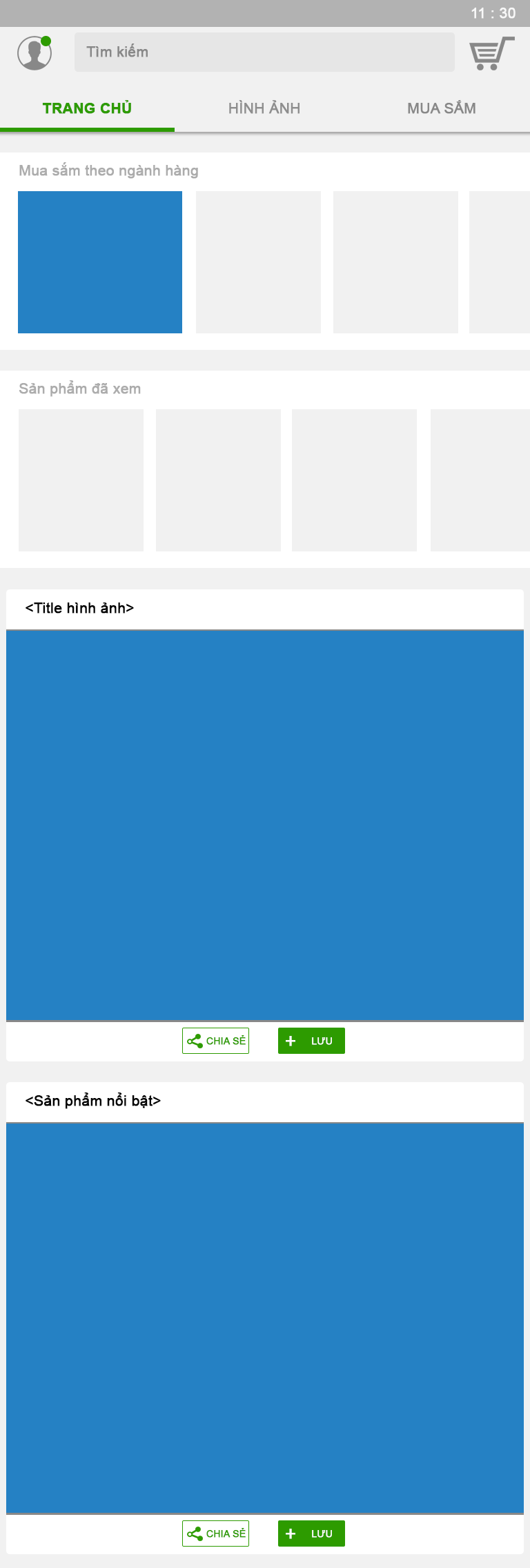
1. **Thiết kế giao diện**
   1. **Sơ đồ liên kết giao diện**
   2. **Danh sách giao diện**
   3. **Mô tả giao diện**

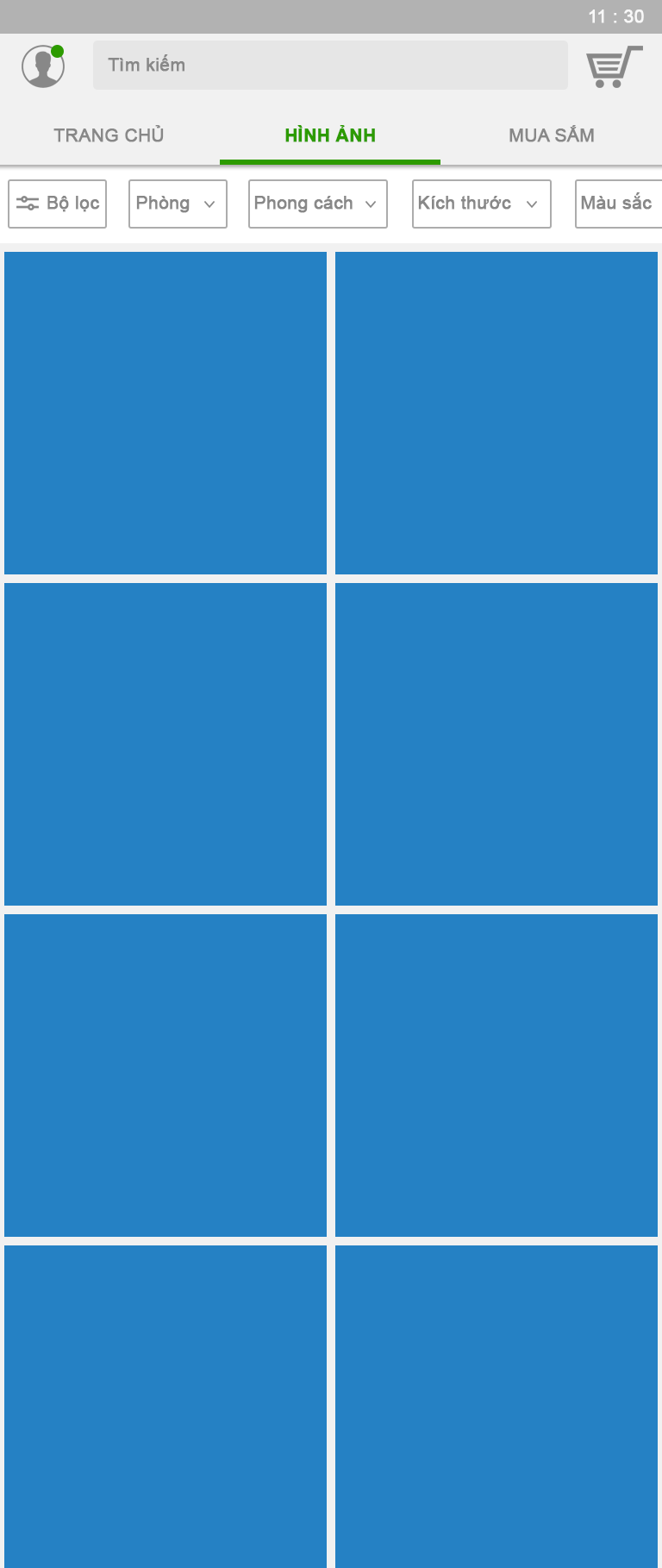
****

*Màn hình khởi tạo ứng dụng*

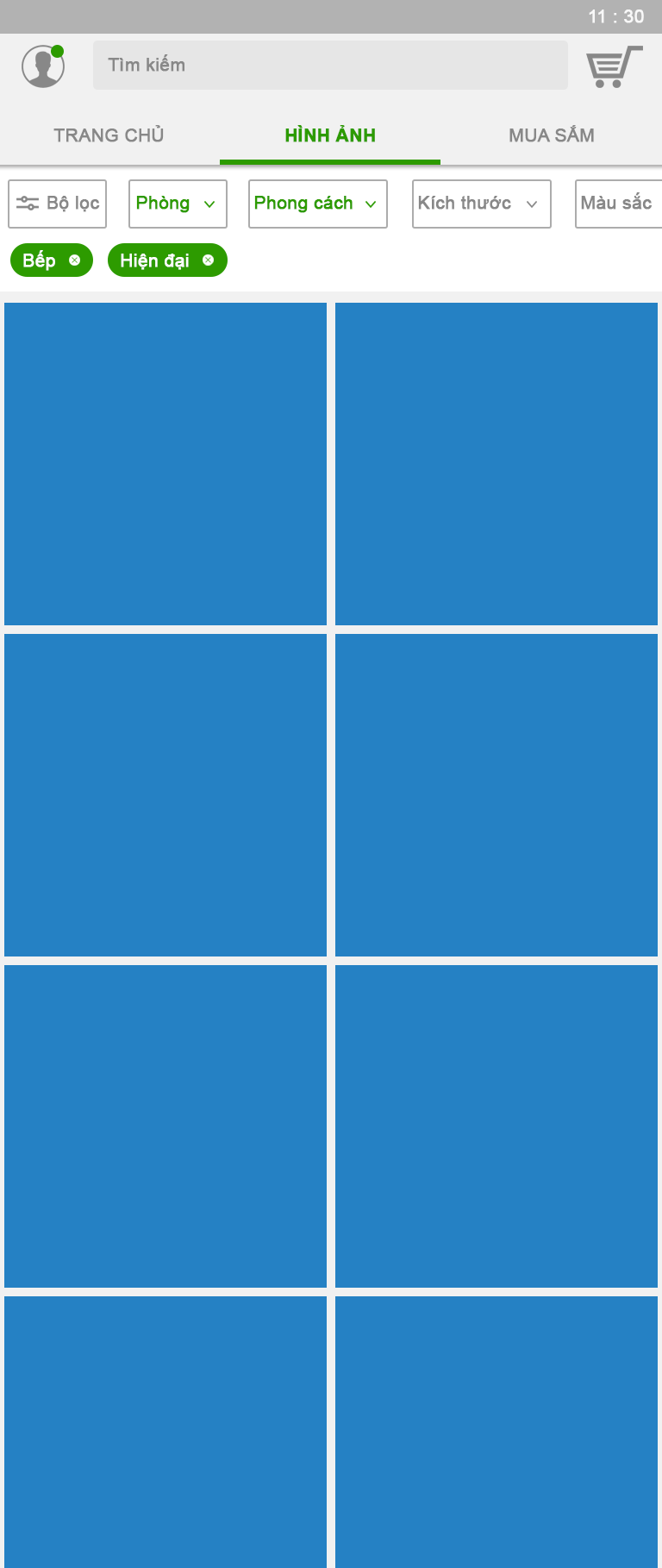
****

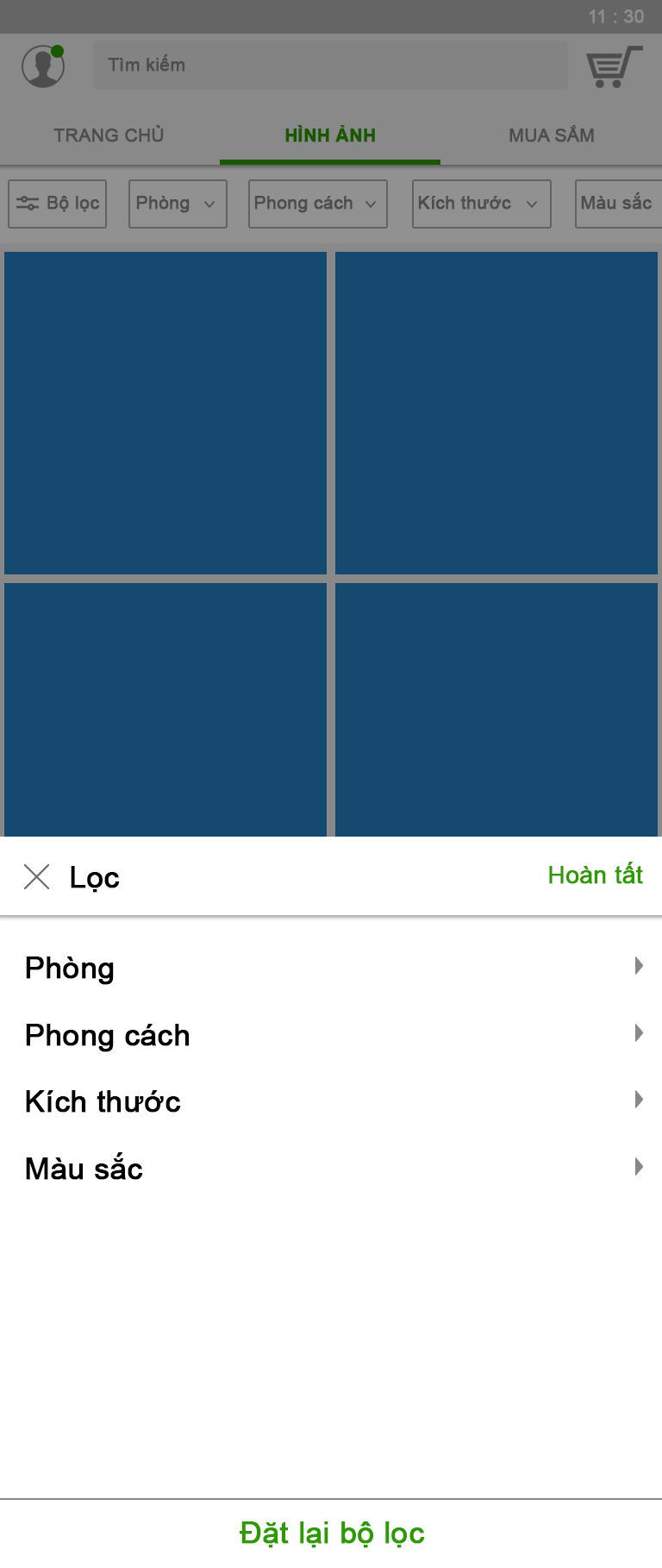
*Màn hình đăng ký tài khoản*

****

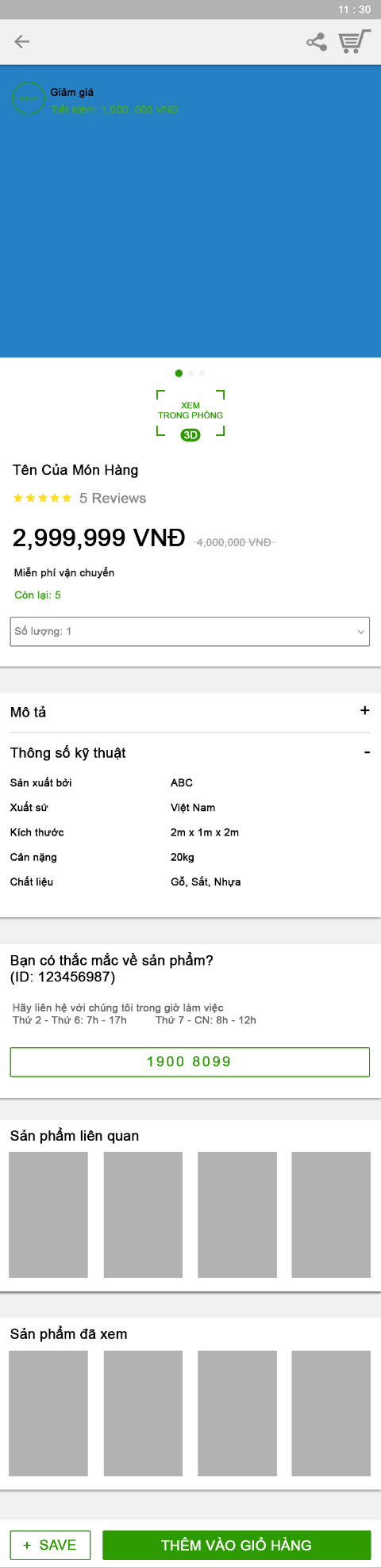
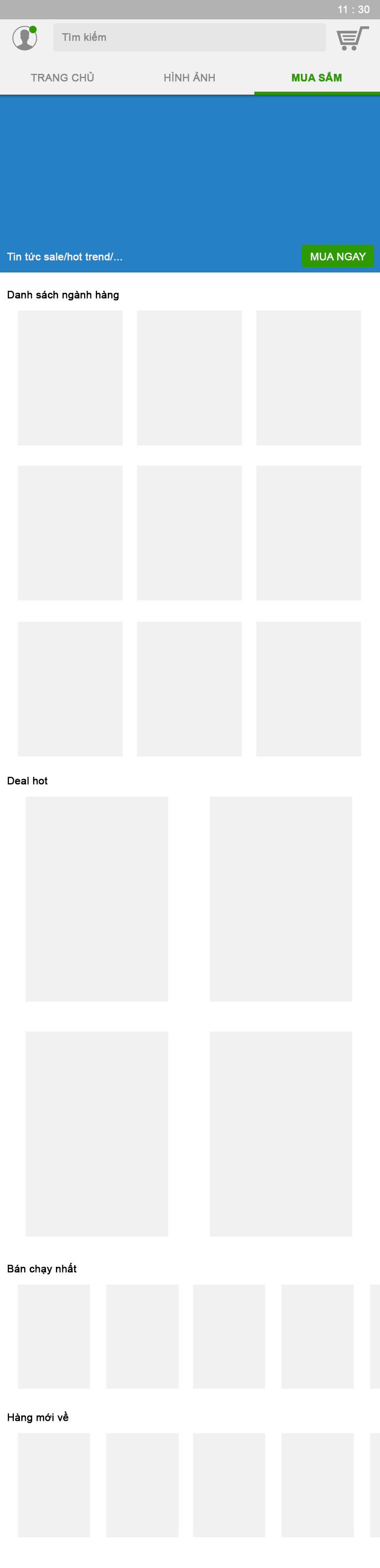
*Trang chủ*****

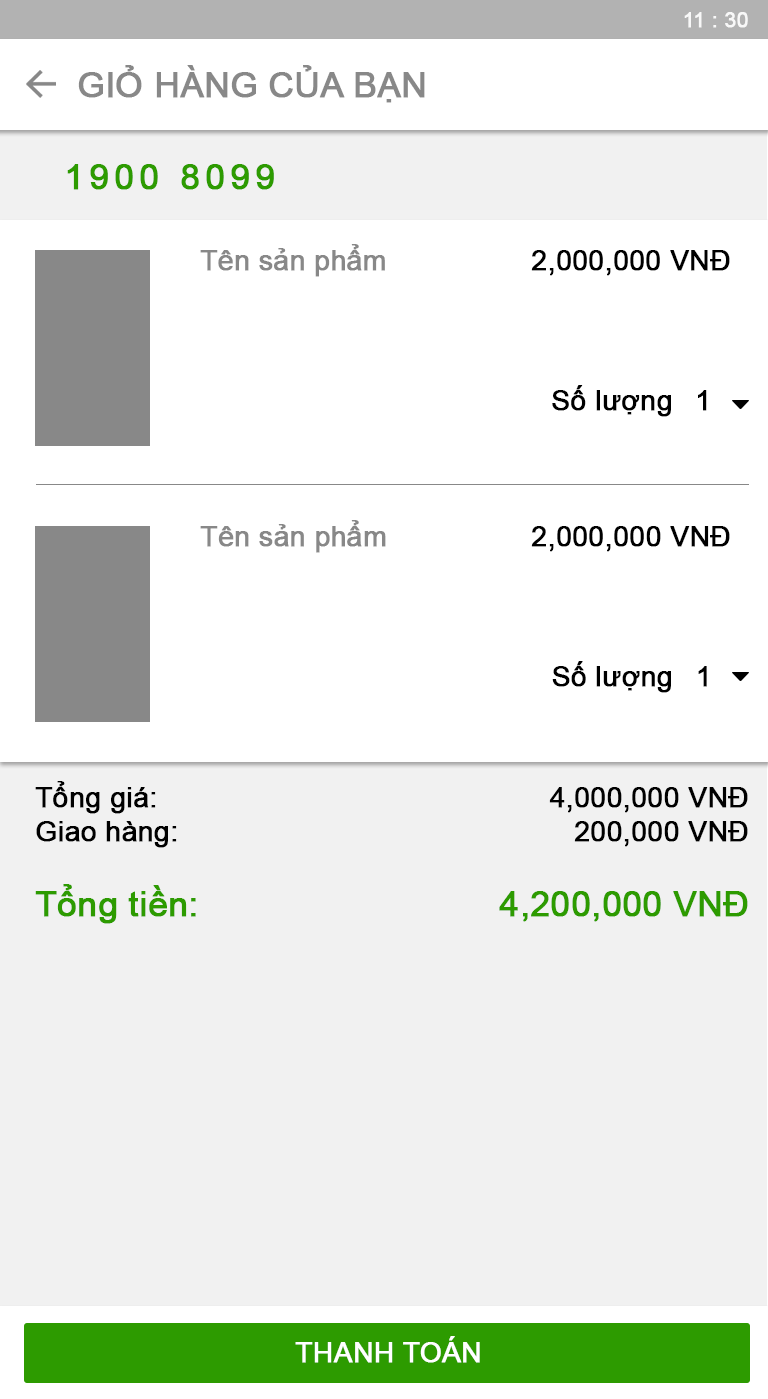
*Hình ảnh*

****

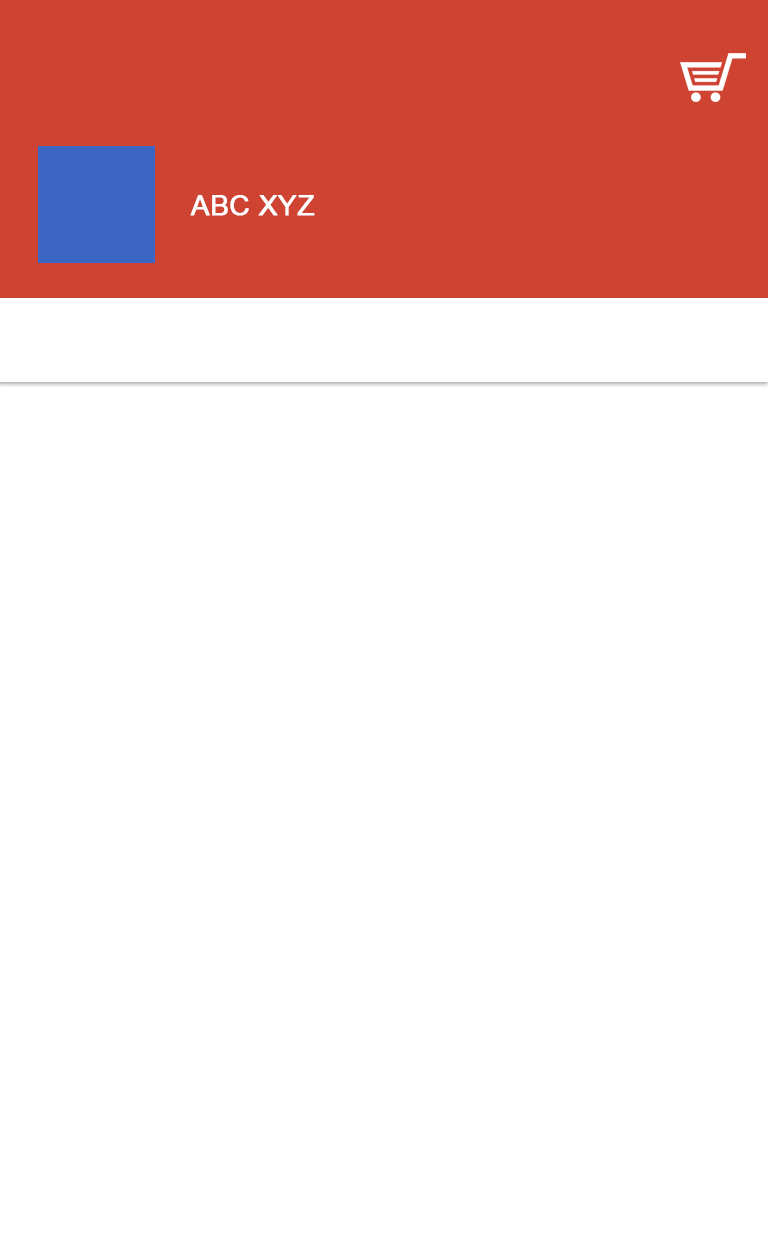
*Lọc hình ảnh*

*Bộ lọc hình ảnh*

**

****

*Giỏ hàng*

****

*Trang cá nhân*