

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



Đồ án tổng hợp - hướng Công nghệ phần mềm - CO3103

Báo cáo

PHÁT TRIỂN HỆ THỐNG IN 3D

Giảng viên hướng dẫn: ThS. Mai Đức Trung

Sinh viên thực hiện: 2110193 - Đỗ Nguyễn An Huy
2110342 - Nguyễn Minh Lộc
2114417 - Nguyễn Ngọc Phú
2112122 - Nguyễn Hồng Quân
2110527 - Nguyễn Hoàng Duy Tân
2112336 - Phạm Đức Thắng
2112594 - Trần Nguyễn Minh Tuệ

Mục lục

1. Tổng quan dự án	4
1.1. Bối cảnh dự án	4
1.2. Stakeholders của dự án	5
1.3. Phạm vi của dự án	5
1.3.1. Thủ công	5
1.3.2. Tự động	6
2. Phân tích yêu cầu	7
2.1. Yêu cầu chức năng	7
2.1.1. User Story	7
2.1.2. Chức năng hệ thống	8
2.2. Yêu cầu phi chức năng	11
3. Use-case diagram and use-case scenario	12
3.1. Use case lựa chọn mô hình 3D (Choosing 3D model)	12
3.2. Use case quản lý mô hình 3D (Managing 3D model)	14
3.3. Use case đặt hàng mô hình 3D (Making order 3D model)	16
3.4. Use case xác nhận và gửi đơn hàng (Confirm and send order)	18
3.5. Use case xử lý đơn hàng (Order processing)	19
3.6. Use case thanh toán đơn hàng (Payment order)	20
4. Framework & Technology	22
4.1. Git	22
4.2. GitHub	22
4.3. ReactJS	23
4.4. Fastify	23
4.5. Prisma	23
4.6. PostgreSQL	24
5. Framework & Technology	25
5.1. Git	25
5.2. GitHub	25
5.3. ReactJS	26
5.4. Fastify	26
5.5. Prisma	26
5.6. PostgreSQL	27

Danh mục hình vẽ

Hình 1: Use-case lựa chọn mô hình 3D	12
Hình 2: Use-case quản lý mô hình 3D	13
Hình 3: Use-case quản lý mô hình 3D	14
Hình 4: Use-case đặt hàng mô hình 3D	16
Hình 5: Use-case xác nhận và gửi đơn hàng	18
Hình 6: Use-case xử lý đơn hàng	20
Hình 7: Use-case thanh toán đơn hàng	21

Danh mục bảng biểu

Bảng 1: Use-case lựa chọn mô hình 3D	14
Bảng 2: Use-case quản lý mô hình 3D	15
Bảng 3: Use-case đặt hàng mô hình 3D	17
Bảng 4: Use-case xác nhận và gửi đơn hàng	19
Bảng 5: Use-case xử lý đơn hàng	20
Bảng 6: Use-case thanh toán đơn hàng	21

1. Tổng quan dự án

1.1. Bối cảnh dự án

Hệ thống in ấn 3D được sử dụng để đáp ứng nhu cầu cho những ai muốn sở hữu mô hình 3D tùy theo mục đích. Để làm được điều đó, chúng tôi sử dụng máy in ấn 3D có tên gọi là **FLSUN-V400**.

Vì giới hạn về tài nguyên phần cứng, hệ thống chỉ bao gồm một máy in 3D đặt ở vị trí cố định. Máy in có những thông số như sau: tên thương hiệu/nhà sản xuất, kiểu máy in, mô tả máy in, firmware, một địa chỉ IP cục bộ trong mạng LAN, dashboard web mặc định của máy in.

Để kích hoạt máy in hoạt động, bạn cần phải truy cập vào địa chỉ IP cục bộ của máy in và tải file định dạng `.gcode` tới dashboard web của máy in. Tuy nhiên, phương pháp này chỉ người quản lý máy in mới có thể làm được bởi vì những lý do sau:

- Máy in chỉ có thể được sử dụng trong mạng cục bộ (LAN network).
- Dashboard web của máy in tương đối phức tạp và không có sự phân quyền người dùng.
- Chưa có cơ chế ngăn chặn những hành động vô ý của người dùng ảnh hưởng đến máy in như: tải lên định dạng file không đúng, tải lên file `.gcode` không phù hợp với loại máy in,...

Vì thế, giải pháp phần mềm chúng tôi đưa ra là xây dựng một dịch vụ in 3D trực tuyến cho nhiều người sử dụng với các tính năng hỗ trợ như sau:

- **Nền tảng bán hàng trên web:** được truy cập thông qua Internet - nơi người dùng có thể đặt mua mô hình 3D theo một trong hai cách sau:
 - Tải lên file `.gcode` phù hợp với loại máy in (FLSUN-V400). Phần mềm không hỗ trợ việc tạo file `.gcode`, vì vậy người dùng phải chuẩn bị trước. Người dùng có thể cài đặt phần mềm [Ultimaker Cura](#) để cắt file `.stl` thành file `.gcode` và tải lên hệ thống.
 - Phần mềm sẽ đề xuất các mẫu mô hình 3D đã có sẵn các file `.gcode` cho người dùng lựa chọn.
- **Màn hình hiển thị sản phẩm phía người dùng:** bao gồm các thông tin sau: tiến độ in ấn, thời gian hoàn thành in ấn, giá cả,...
- **Màn hình hiển thị điều khiển phía người quản lý:** tương đối giống với dashboard web của máy in và có thêm một số tính năng như: xem được số lượng đơn đặt hàng và tình trạng mỗi đơn hàng, kích hoạt/bất hoạt máy in,...

Bởi vì hệ thống chỉ có một máy in, nếu có quá nhiều đơn đặt hàng cùng một thời điểm, chúng sẽ được xếp vào một hàng đợi. Người quản lý sẽ lấy sản phẩm ra khỏi máy in sau khi in xong, nhấn nút `Continue printing` trên hệ thống để đưa sản phẩm đầu tiên trong hàng đợi vào để tiếp tục việc in ấn.

Hệ thống chỉ chấp nhận file với định dạng `.gcode` và phù hợp với máy in **FLSUN-V400**, nếu không hệ thống sẽ tự động hủy quá trình tải file lên của khách hàng.

Hệ thống phải ghi lại lịch sử của mỗi đơn hàng, bao gồm: thời gian đặt hàng, thời gian vận chuyển, thông tin người dùng đặt hàng, chi phí,...

Hệ thống được cung cấp thông qua ứng dụng web-based app, **không hỗ trợ mobile app**.

1.2. Stakeholders của dự án

Dự án này bao gồm 2 stakeholders:

1. Người quản lý: Là người trực tiếp quản lý máy in, kiểm tra và cập nhật các đơn hàng sang các trạng thái.
2. Khách hàng: Là người đặt các mô hình 3D.

1.3. Phạm vi của dự án

Dự án được chia thành 2 mức thực hiện: **Thủ công** và **Tự động**. Nhóm sẽ ưu tiên hiện thực hệ thống ở mức thủ công.

1.3.1. Thủ công

Tất cả đơn hàng được gửi tới người quản lý qua hệ thống. Người quản lý sẽ tự mình upload các file `.gcode` trong các đơn đặt hàng lên dashboard web của máy in. Quy trình này yêu cầu hệ thống có các tính năng chính sau:

1. **Lựa chọn mô hình 3D**: cho phép khách hàng tải lên file `.gcode` phù hợp với loại máy in FLSUN-V400 hoặc bao gồm danh sách các mô hình 3D có sẵn đã tích hợp file `.gcode` (≥ 100 mô hình) cho khách hàng lựa chọn. Ngoài ra, hệ thống còn hỗ trợ phân loại mô hình theo danh mục, tìm kiếm theo tên và lọc theo giá tiền.
2. **Quản lý mô hình 3D**: Cho phép người quản lý thao tác với các mô hình 3D mà hệ thống đề xuất cho khách hàng, gồm thêm/xóa/sửa. Yêu cầu này đòi hỏi hệ thống phải phân quyền người dùng, bao gồm 2 vai trò: người quản lý và khách hàng.
3. **Đặt mô hình 3D**: Khách hàng cần phải đăng kí một tài khoản để có thể đặt được mô hình 3D sau khi upload file hoặc lựa chọn sản phẩm mẫu.
4. **Xác nhận và gửi đơn hàng**: Khách hàng sau khi đặt mô hình và xác nhận xong thì có thể gửi đơn hàng đi để được xử lý.
5. **Xử lý đơn hàng**: Người quản lý có thể xem thông tin đơn hàng và chuyển đơn hàng sang các trạng thái tiếp theo như: đã đặt, đang giao, đã giao, đã thanh toán,...
6. **Thanh toán online**: Thực hiện thanh toán qua Momo.

1.3.2. Tự động

Tất cả đơn hàng được lưu trên hệ thống. Người quản lý chỉ cần nhấn nút `Start printing` thì file `.gcode` tự động được gửi tới máy in mà không cần thông qua dashboard web của máy in. Tuy nhiên, cần có một người trực tại máy in để gỡ sản phẩm ra khỏi bàn in sau khi in xong và nhấn nút `Continue printing` để tiếp tục in sản phẩm tiếp theo trong hàng đợi của hệ thống.

Mức hiện thực này bao gồm tất cả các tính năng của mức hiện thực thủ công, giảm thiểu tác vụ cho người quản lý và hỗ trợ thêm một số tính năng cho cả hai đối tượng, bao gồm:

- **Màn hình điều khiển máy in phía người quản lý:** Người quản lý có thể xem được tiến độ sản phẩm đang được in, nhiệt độ sản phẩm, khối lượng nhựa in cần sử dụng, thời gian in xong. Ngoài ra, có thể thay đổi tốc độ in, tạm dừng in, kích hoạt/bất hoạt máy in.
- **Màn hình hiển thị sản phẩm phía khách hàng:** Khách hàng cũng có thể xem được tiến độ in và thời gian in xong của sản phẩm mà mình đặt mua.

2. Phân tích yêu cầu

Như đã đề cập ở phần Phạm vi dự án, nhóm sẽ ưu tiên mức hiện thực thủ công bao gồm sáu tính năng chính trên. Phần này sẽ mô tả chi tiết về toàn bộ yêu cầu chức năng và yêu cầu phi chức năng của hệ thống ở mức thủ công:

2.1. Yêu cầu chức năng

2.1.1. User Story

2.1.1.1. Đối với khách hàng

- Khách hàng có thể lựa chọn các mô hình 3D có sẵn trong hệ thống.
- Khách hàng có thể xem tất cả các mô hình 3D ở mục **All Things**.
- Khách hàng có thể tìm kiếm các mô hình 3D theo tên.
- Khách hàng có thể tìm kiếm các mô hình bằng cách lọc theo danh mục, giá tiền, mốc thời gian đăng.
- Khách hàng có thể xem thông tin chi tiết của mô hình 3D.
- Khách hàng có thể Upload file .gcode của riêng họ để đặt in.
- Khách hàng phải đăng nhập vào tài khoản user để thực hiện chức năng đặt hàng.
- Khách hàng có thể thêm mô hình 3D có sẵn vào giỏ hàng để lưu lại thông tin mô hình tiến hành đặt hàng. Khi thêm vào giỏ, khách hàng có thể chọn số lượng.
- Khách hàng có thể xem và chỉnh sửa thông tin giỏ hàng.
- Khách hàng có thể xóa một hoặc nhiều mô hình ra khỏi giỏ hàng cùng lúc.
- Đối với trường hợp đặt in theo file .gcode của riêng họ, khách hàng có thể tương tác hệ thống như đối với các mô hình 3D có sẵn.
- Khách hàng có thể xem đề xuất giá tiền để in mô hình 3D dựa vào các file .gcode.
- Khách hàng có thể kết hợp upload file và lựa chọn các mô hình 3D có sẵn.
- Khách hàng có thể xác nhận và gửi đơn hàng.
- Khách hàng điền thông tin đặt hàng bao gồm tên khách hàng, số điện thoại, địa chỉ nhận hàng (phường, quận, địa chỉ thêm do khách hàng cung cấp) và ghi chú (nếu có).
- Khách hàng có thể xem được phí ship dựa trên khoảng cách giao hàng và thời gian giao hàng dự kiến.
- Sau bước xác nhận đơn hàng và trước bước thanh toán, khách hàng có thể quay lại giỏ hàng để mua thêm/xóa các mô hình 3D, chỉnh sửa thông tin đặt hàng.
- Khách hàng có thể chọn phương thức thanh toán là tiền mặt hoặc thanh toán online qua

Momo.

- Khách hàng có thể hủy đơn hàng nếu vẫn chưa được in (trạng thái **Đang chờ xử lý**).
- Khách hàng có thể theo dõi tình trạng đơn hàng: Đang chờ xử lý; Đang in; Đang giao; Đã thanh toán.
- Khách hàng có thể lựa chọn 2 phương thức thanh toán: Thanh toán bằng tiền mặt hoặc Thanh toán qua Momo.
- Trường hợp khách hàng lựa chọn thanh toán qua Momo, màn hình sẽ xuất hiện một mã vạch

(đã kèm số tiền) để người dùng quét mã. Mã sẽ có hiệu lực trong vòng 10 phút.

- Hệ thống thông báo thanh toán khách hàng thành công/thất bại.

2.1.1.2. Đối với người quản lý

- Người quản lý có thể thêm mô hình 3D vào hệ thống: giá tiền, hình ảnh minh họa cũng phải được thêm vào.
- Người quản lý có thể xóa mô hình 3D, việc xóa mô hình 3D không ảnh hưởng tới các đơn hàng đã tiếp nhận trước đó.
- Người quản lý có thể sửa mô hình 3D, cập nhật không ảnh hưởng đến các đơn hàng đã thanh toán.
- Người quản lý có thể xem danh sách các đơn đặt hàng và cập nhật trạng thái của chúng.
- Người quản lý có thể nhấn chọn từng đơn hàng để xem thông tin chi tiết.
- Người quản lý có thể nhấn chọn xử lý để từ chối hoặc chuyển đơn hàng sang trạng thái tiếp theo.
- Người quản lý có thể lựa chọn từ chối hoặc chấp nhận nhiều đơn hàng cùng một lúc.

2.1.2. Chức năng hệ thống

2.1.2.1. Lựa chọn mô hình 3D

Trường hợp khách hàng lựa chọn các mô hình 3D có sẵn, hệ thống phải cung cấp các chức năng:

- Phân chia các mô hình 3D theo các danh mục: Fashion, Hobby, Learning, Tools, Toys & Games, Art, Household.
- Hệ thống có mục **All Things** bao gồm tất cả các loại mô hình 3D.
- Các nhóm mô hình 3D được phân thành nhiều trang, mỗi trang chứa tối đa 10 mô hình.
- Mỗi mô hình 3D có một nút Like, hệ thống sẽ mặc định sắp xếp các mô hình 3D theo tiêu chí số lượt Like từ cao đến thấp.
- Sắp xếp các mô hình 3D theo giá tiền, mốc thời gian đăng. Lọc các mô hình 3D theo danh mục, khoảng thời gian. Giữ nguyên trạng thái lọc và tiêu chí sắp xếp khi chuyển trang. Trạng thái ban đầu của bộ lọc là **No filter**.

- Tìm kiếm các mô hình 3D theo tên. Hệ thống sẽ cố gắng tìm những mô hình 3D có tên giống như từ khóa đã nhập hoặc có tên gần giống. Nếu không tìm thấy, hiển thị danh sách rỗng kèm thông báo **Không tìm thấy**. Được phép áp dụng bộ lọc và tiêu chí sắp xếp khi màn hình xuất ra danh sách kết quả. Để quay về trạng thái trước khi tìm kiếm, nhấn nút **X** trên thanh tìm kiếm.

Trường hợp khách hàng upload file .gcode của riêng họ, hệ thống phải cung cấp các chức năng:

- Chỉ cho phép các file định dạng .gcode được upload lên hệ thống.
- Các file .gcode phải được generate từ chính loại máy in FLSUN-V400.

2.1.2.2. Quản lý mô hình 3D

Người quản lý phải đăng nhập vào tài khoản admin để thực hiện chức năng này, bao gồm các thao tác:

- Thêm/xóa/sửa mô hình 3D.
- Khi thêm mô hình 3D: giá tiền, hình ảnh minh họa cũng phải được thêm vào.
- Việc xóa mô hình 3D không ảnh hưởng tới các đơn hàng đã tiếp nhận trước đó.
- Các mô hình 3D được chỉnh sửa, cập nhật không ảnh hưởng đến các đơn hàng đã thanh toán.

2.1.2.3. Đặt mô hình 3D

Khách hàng phải đăng nhập vào tài khoản user để thực hiện chức năng này.

Trường hợp khách hàng chọn các mẫu mô hình 3D có sẵn, hệ thống phải cung cấp các chức năng:

- Thêm mô hình 3D vào giỏ hàng để lưu lại thông tin mô hình hoặc tiến hành đặt hàng. Khi thêm vào giỏ, khách hàng có thể chọn số lượng.
- Thông tin mô hình 3D trong giỏ hàng bao gồm: tên mô hình, đơn giá, số lượng.
- Khi người dùng thêm mô hình 3D nhiều lần, những mô hình trùng tên sẽ được cộng dồn với số lượng và giá tương ứng.
- Giỏ hàng phải thống kê được các mô hình 3D, số lượng, đơn giá của từng mô hình và tổng tiền của giỏ hàng.
- Xem và chỉnh sửa thông tin giỏ hàng.
- Khách hàng có thể xóa một hoặc nhiều mô hình ra khỏi giỏ hàng cùng lúc.

Trường hợp khách hàng đặt in các mô hình 3D dựa trên các file .gcode của họ, ngoài các thao tác trên giỏ hàng tương tự như trên, hệ thống còn phải cung cấp các chức năng:

- Khi nhấn nút Upload file, hệ thống sẽ hiện ra một modal window yêu cầu lựa chọn đơn hàng hoặc tạo một đơn hàng mới để chứa file đó.
- Cho phép upload nhiều file cho một đơn hàng.

- Đề xuất giá tiền để in mô hình 3D dựa vào các file .gcode.
- Cho phép kết hợp upload file và lựa chọn các mô hình 3D có sẵn.

2.1.2.4. Xác nhận và gửi đơn hàng

Hệ thống phải cung cấp các chức năng:

- Cho phép chọn một hoặc nhiều mô hình 3D từ giỏ hàng để tiến hành đặt hàng.
- Có form cung cấp thông tin đặt hàng bao gồm tên khách hàng, số điện thoại, địa chỉ nhận hàng (phường, quận, địa chỉ thêm do khách hàng cung cấp) và ghi chú (nếu có).
- Hệ thống cung cấp tính năng tính phí ship dựa trên khoảng cách giao hàng và hiển thị thời gian giao hàng dự kiến.
- Phí ship được tính bằng 5000 VND cho 3 kilomet đầu tiên; 3000 VND cho mỗi kilomet tiếp theo và không vượt quá 30000 VND. Khu vực giao hàng được giới hạn trong phạm vi TP HCM.
- Thời gian giao dự kiến được tính dựa trên quãng đường và lưu lượng giao thông tại thời điểm đặt hàng.
- Sau bước xác nhận đơn hàng và trước bước thanh toán, khách hàng có thể quay lại giỏ hàng để mua thêm/xóa các mô hình 3D, chỉnh sửa thông tin đặt hàng.
- Khách hàng có thể chọn phương thức thanh toán là tiền mặt hoặc thanh toán online qua Momo.
- Sau khi tiến hành đặt hàng thành công, các mô hình 3D đã được đặt sẽ bị xóa khỏi giỏ hàng và lịch sử đặt hàng sẽ được ghi lại vào hệ thống.
- Cho phép hủy đơn hàng nếu vẫn chưa được in (trạng thái **Đang chờ xử lý**).
- Khách hàng có thể theo dõi tình trạng đơn hàng: Đang chờ xử lý; Đang in; Đang giao; Đã thanh toán.

2.1.2.5. Xử lý đơn hàng

Người quản lý sẽ xem danh sách các đơn đặt hàng và cập nhật trạng thái của chúng. Việc này sẽ yêu cầu hệ thống cung cấp các tính năng sau:

- Hiển thị danh sách đơn hàng thành các mục tương ứng với trạng thái của chúng. Trạng thái đơn hàng bao gồm: Đang chờ xử lý -> Đang in -> Đang giao -> Đã thanh toán.
- Ở mỗi mục, đơn hàng được sắp xếp mặc định dựa trên thời gian đơn hàng đó được ghi nhận.
- Người quản lý có thể nhấn chọn từng đơn hàng để xem thông tin chi tiết.
- Người quản lý có thể nhấn chọn xử lý để từ chối hoặc chuyển đơn hàng sang trạng thái tiếp theo.
- Người quản lý có thể lựa chọn từ chối hoặc chấp nhận nhiều đơn hàng cùng một lúc.

2.1.2.6. Thanh toán đơn hàng

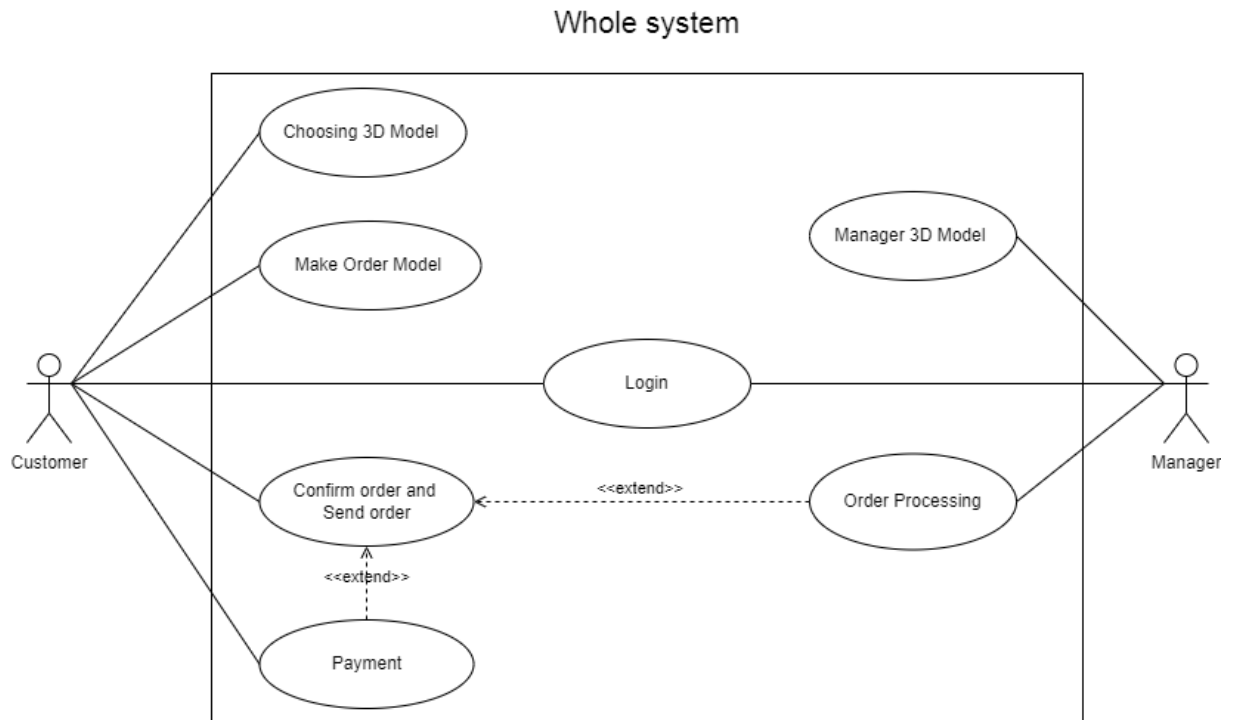
Hệ thống hỗ trợ khách hàng thanh toán đơn hàng bằng tiền mặt và qua ví điện tử Momo:

- Khách hàng có thể lựa chọn 2 phương thức thanh toán: **Thanh toán bằng tiền mặt** hoặc **Thanh toán qua Momo**.
- Trường hợp khách hàng lựa chọn thanh toán qua Momo, màn hình sẽ xuất hiện một mã vạch (đã kèm số tiền) để người dùng quét mã. Mã sẽ có hiệu lực trong vòng 10 phút.
- Hệ thống thông báo thanh toán thành công/thất bại.

2.2. Yêu cầu phi chức năng

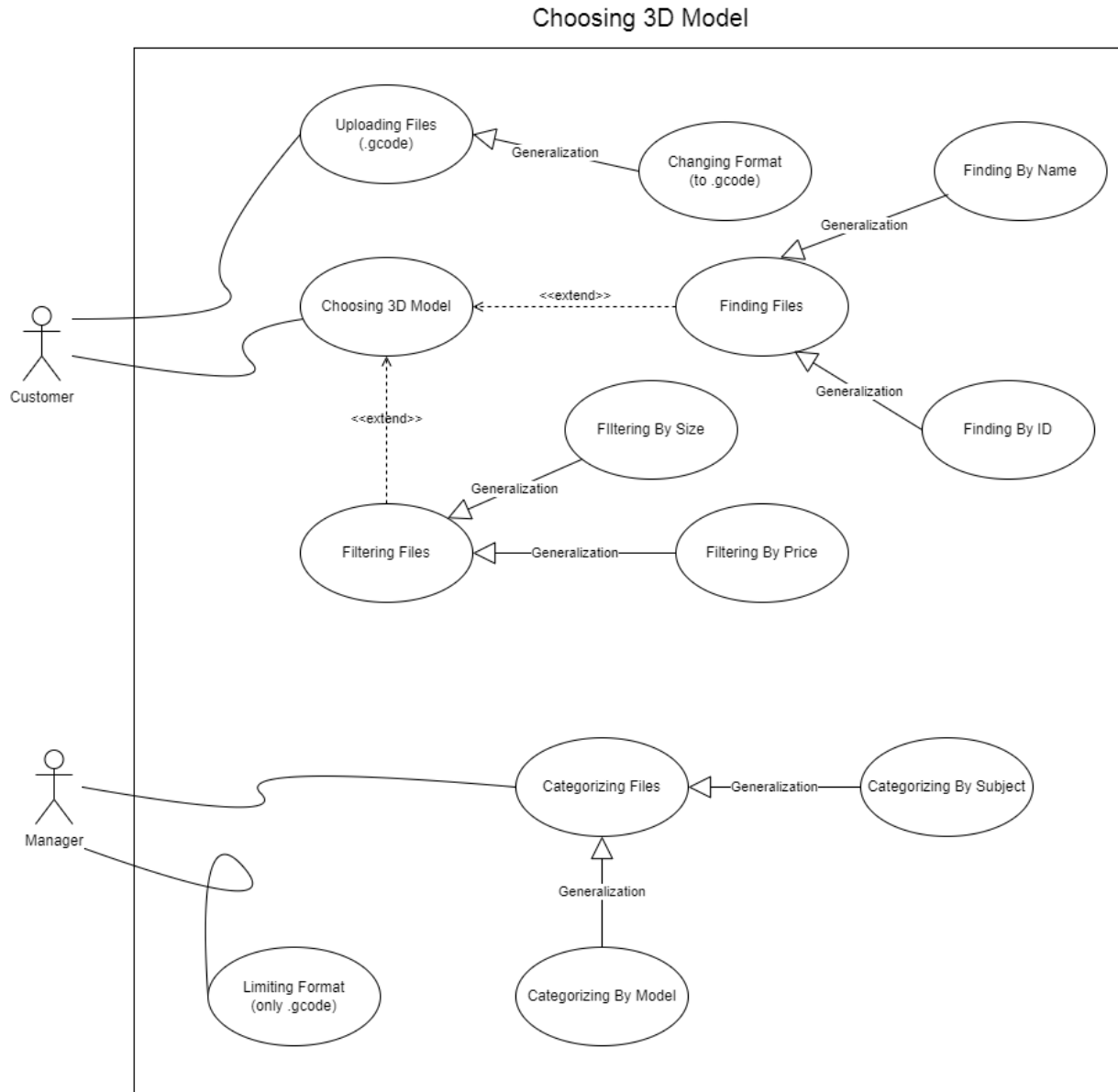
- Hệ thống được truy cập thông qua web-based.
- hệ thống xử lý nhiều đơn đặt hàng in và xác lập độ ưu tiên trong cơ chế FCFS.
- Độ tin cậy (Reliability):
 - Hệ thống duy trì dữ liệu/phục hồi về trạng thái trước khi có lỗi.
- Tính sẵn sàng (Availability):
 - Hệ thống phải hoạt động 24/7.
- Khả năng tiếp cận (Accessibility):
 - UI phải được hiển thị chính xác trên nhiều kích cỡ màn hình khác nhau:
 - Màn hình desktop: 1280x720 - 1920x1080.
 - Màn hình tablet: 601x962 - 1280x800.
 - Màn hình mobile: 360x640 - 414x896.
 - Hỗ trợ trên các trình duyệt khác nhau: Chrome, Edge, Firefox, Safari.
- Độ bảo mật (Security):
 - Tuân thủ theo tiêu chuẩn OWASP

3. Use-case diagram and use-case scenario



Hình 1: Use-case lựa chọn mô hình 3D

3.1. Use case lựa chọn mô hình 3D (Choosing 3D model)



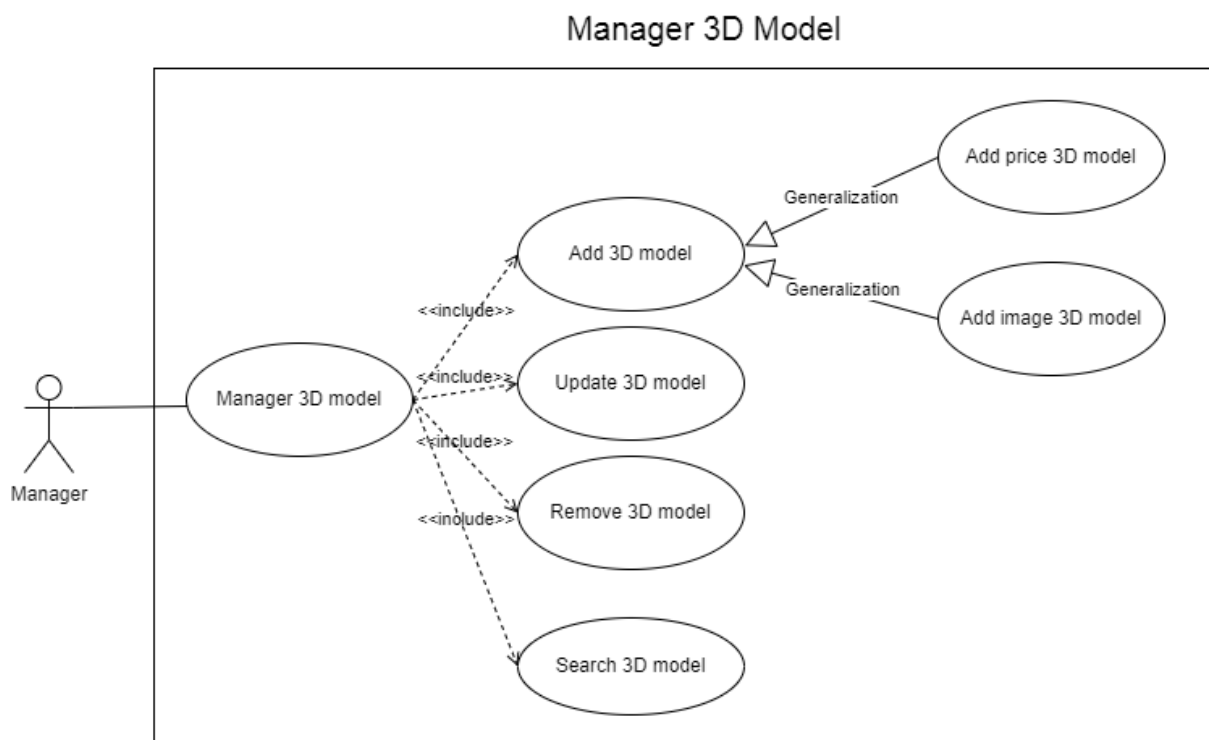
Hình 2: Use-case quản lý mô hình 3D

Use Case Name	Lựa chọn mô hình 3D
Actors	Khách hàng.
Brief Description	Khách hàng sử dụng chức năng này để lựa chọn các mô hình 3D có sẵn trong hệ thống hoặc có thể lựa chọn file .gcode để in 3D.
Preconditions	Khách hàng đã truy cập được trang web của hệ thống.
Postconditions	Khách hàng đã lựa chọn được mô hình 3D hoặc Upload được file .gcode lên hệ thống.
Normal Flow	1. Khách hàng có thể xem tất cả mô hình 3D ở mục All things .

	<p>2. Khách hàng có thể xem các mô hình 3D theo các mục: Fashion, Hobby, Learning, Tools, Toys & Games, Art, Household.</p> <p>3. Khách hàng có thể tìm kiếm mô hình 3D theo tên mô hình.</p> <p>4. Khách hàng có thể tìm kiếm mô hình 3D theo các bộ lọc: theo danh mục, theo giá, theo mốc thời gian, theo khoảng thời gian.</p> <p>5. Khách hàng có thể xem chi tiết mô hình 3D.</p> <p>6. Khách hàng có thể Upload file .gcode của mô hình 3D lên hệ thống.</p>
Exceptional flow	<p>Ở bước 3, Nếu hệ thống tìm kiếm không có kết quả trả về, hệ thống sẽ hiển thị danh sách rỗng và thông báo Không tìm thấy, Khách hàng có thể tìm kiếm theo bước 4, Để quay về trạng thái trước tìm kiếm, khách hàng có thể nhấn nút X trên thanh tìm kiếm.</p>

Bảng 1: Use-case lựa chọn mô hình 3D

3.2. Use case quản lý mô hình 3D (Managing 3D model)



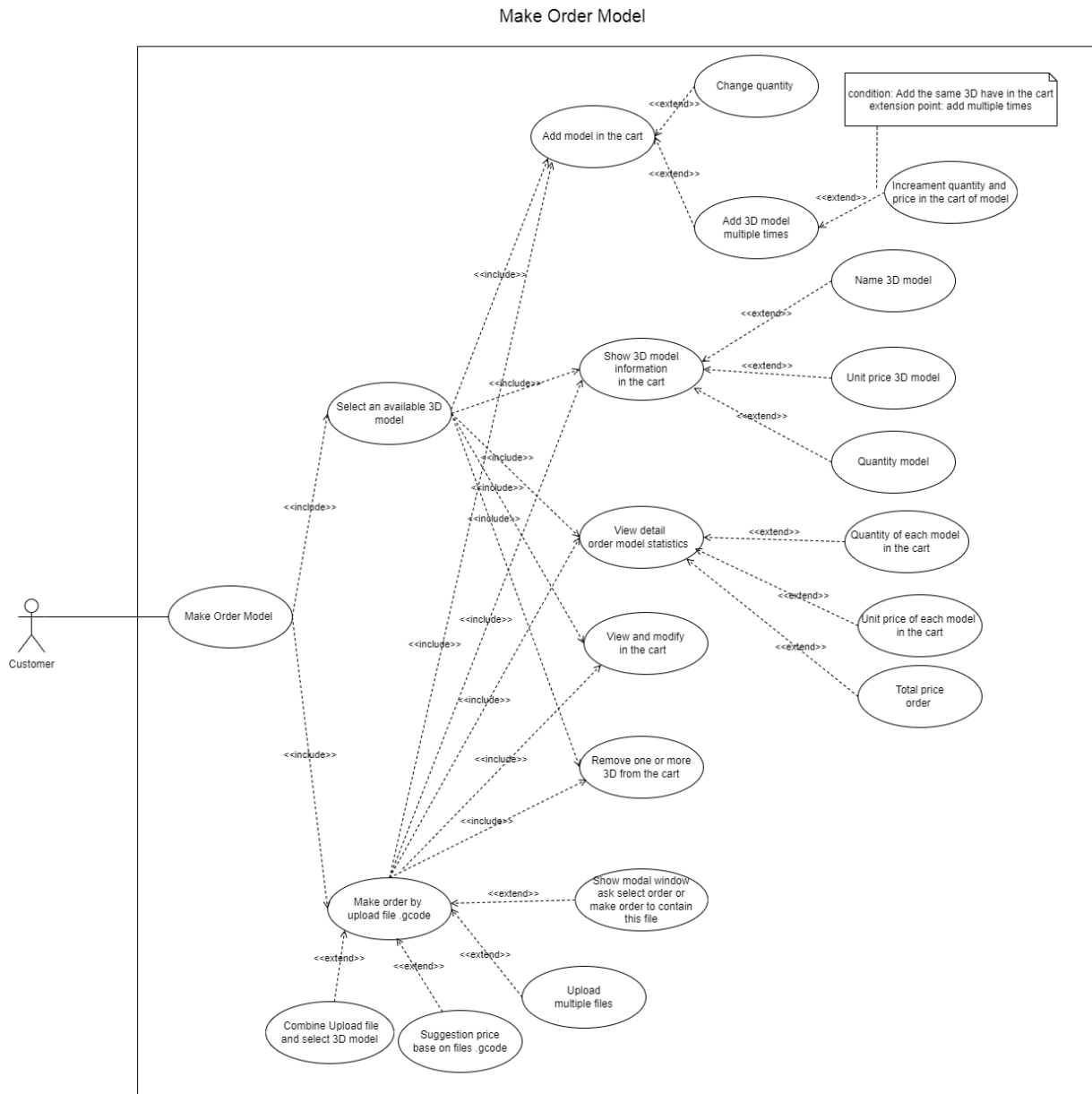
Hình 3: Use-case quản lý mô hình 3D

Use Case Name	Quản lý mô hình 3D
Actors	Người quản lý.

Brief Description	Người quản lý sử dụng chức năng này để quản lý các mô hình 3D trong hệ thống.
Preconditions	Người quản lý đã truy cập được trang web và đăng nhập vào tài khoản admin hệ thống.
Postconditions	Người quản lý đã thêm, sửa, xóa được các mô hình 3D trong hệ thống.
Normal Flow	<ol style="list-style-type: none">1. Người quản lý thêm mô hình 3D vào hệ thống, bao gồm cả hình ảnh minh họa và giá tiền.2. Người quản lý sửa mô hình 3D trong hệ thống.3. Người quản lý xóa mô hình 3D trong hệ thống.
Exceptional flow	<p>2.1. Ở bước 2, Nếu người quản lý chỉnh sửa, cập nhật thông tin mô hình không ảnh hưởng đến các đơn hàng đã được khách hàng đã thanh toán.</p> <p>3.1. Ở bước 3, Nếu người quản lý xóa thông tin mô hình không ảnh hưởng đến các đơn hàng đã được tiếp nhận trước đó.</p>

Bảng 2: Use-case quản lý mô hình 3D

3.3. Use case đặt hàng mô hình 3D (Making order 3D model)



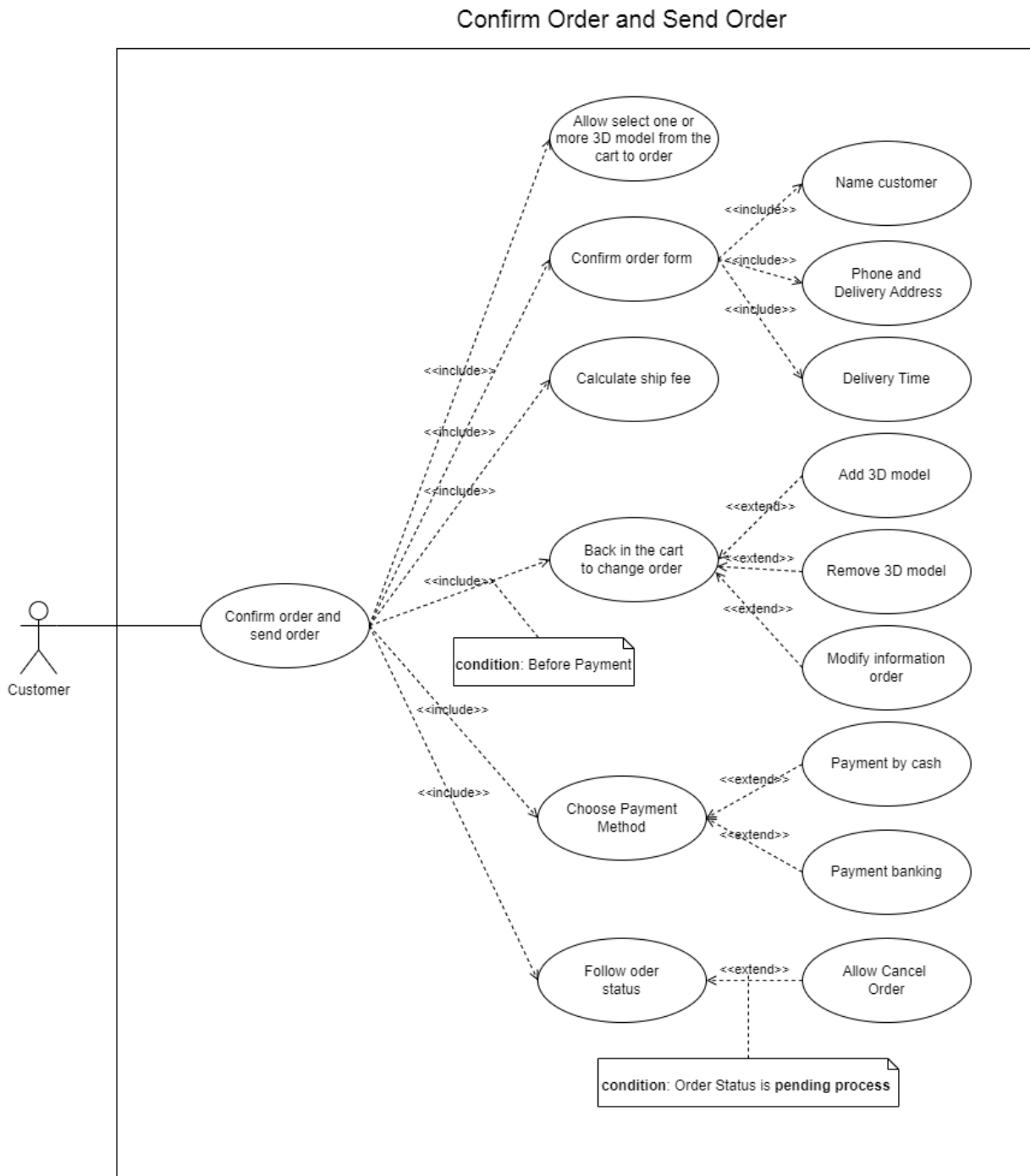
Hình 4: Use-case đặt hàng mô hình 3D

Use Case Name	Đặt hàng mô hình 3D
Actors	Khách hàng.
Brief Description	Khách hàng sử dụng chức năng này để đặt hàng mô hình 3D.
Preconditions	Khách hàng đã truy cập được trang web của hệ thống và lựa chọn được mô hình 3D hoặc upload được file .gcode lên hệ thống.
Postconditions	Khách hàng đã đặt hàng mô hình 3D thành công.

Normal Flow	<ol style="list-style-type: none">Khách hàng đặt hàng in bằng chọn mô hình 3D có sẵn trong hệ thống.<ol style="list-style-type: none">Thêm mô hình 3D vào giỏ hàng và chọn số lượng.Xem lại thông tin mô hình 3D bao gồm: tên mô hình 3D, số lượng, đơn giá.Giỏ hàng hiển thị thống kê được các mô hình 3D, số lượng, đơn giá từng mô hình và tổng tiền của đơn hàng.Khách hàng có thể chỉnh sửa đơn hàng, có thể tùy chỉnh số lượng và xóa một hoặc nhiều mô hình 3D trong giỏ hàng.Khách hàng đặt hàng in các mô hình 3D bằng file .gcode.<ol style="list-style-type: none">Khách hàng thêm file .gcode vào giỏ hàng và chọn số lượng, cho phép Upload nhiều file.Xem lại thông tin file .gcode bao gồm: tên file .gcode, số lượng, đơn giá.Giỏ hàng hiển thị thống kê được các file .gcode, số lượng, đơn giá và tổng tiền của đơn hàng.Khách hàng có thể chỉnh sửa đơn hàng, có thể tùy chỉnh số lượng và xóa một hoặc nhiều file .gcode trong giỏ hàng.
Exceptional flow	<p>Ở bước 1.2 và 2.2, Nếu khách hàng không chọn số lượng mô hình 3D, hệ thống sẽ mặc định số lượng đơn hàng là 1.</p> <p>Ở bước 1.4 và 2.4, Nếu khách hàng xóa hết các mô hình 3D hoặc file .gcode trong giỏ hàng, hệ thống sẽ hiển thị thông báo Giỏ hàng trống.</p>

Bảng 3: Use-case đặt hàng mô hình 3D

3.4. Use case xác nhận và gửi đơn hàng (Confirm and send order)



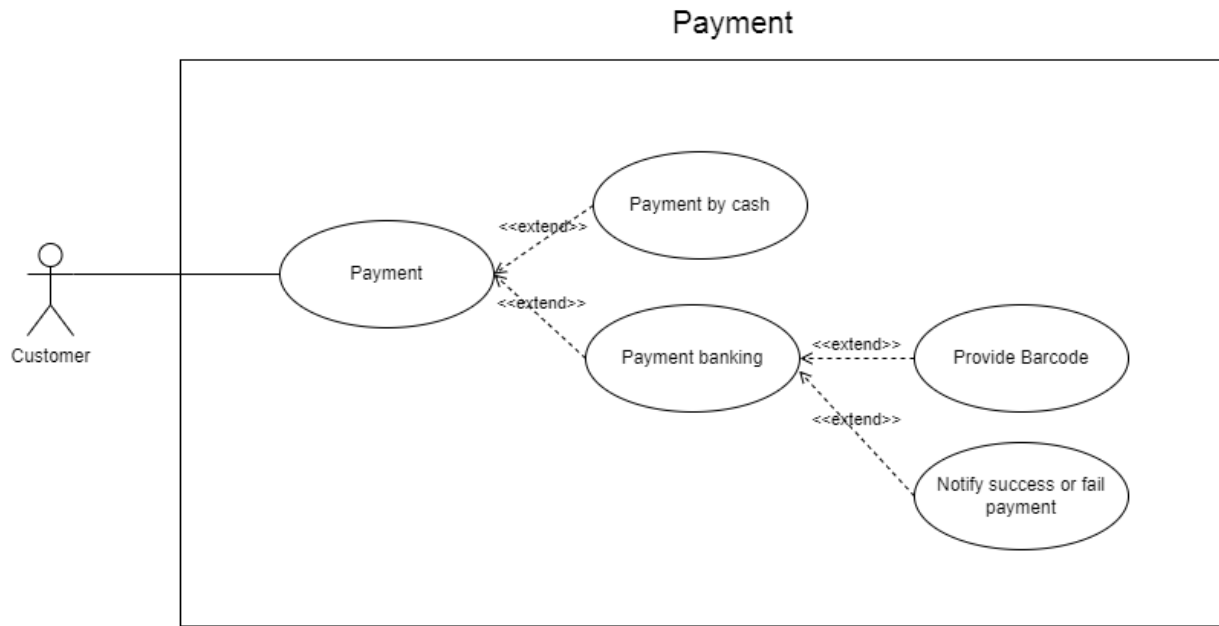
Hình 5: Use-case xác nhận và gửi đơn hàng

Use Case Name	Xác nhận và gửi đơn hàng
Actors	Khách hàng.
Brief Description	Khách hàng sử dụng chức năng này để xác nhận đơn hàng và gửi đơn hàng đi để được xử lý.

Preconditions	Khách hàng đã lựa chọn được mô hình 3D hoặc upload được file .gcode lên hệ thống.
Postconditions	Đơn hàng đã được xác nhận và gửi đi để được xử lý.
Trigger	Khách hàng chọn các mô hình cần đặt và nhấn nút Đặt hàng trong giỏ hàng.
Normal Flow	<ol style="list-style-type: none">1. Hệ thống hiển thị lại thông tin người đặt bao gồm: tên khách hàng, số điện thoại và địa chỉ giao hàng, thời gian giao hàng dự kiến.2. Hệ thống tính toán phí giao hàng và hiển thị lên màn hình.3. Khách hàng lựa chọn phương thức thanh toán.4. Khách hàng nhấn nút Xác nhận đơn hàng.5. Hệ thống gửi thông báo đến khách hàng và người quản lý về việc đơn hàng đã được xác nhận và gửi đi để được xử lý.6. Khách hàng theo dõi trạng thái đơn hàng trên hệ thống.
Exceptional flow	Ở bước 3 , Nếu khách hàng chưa liên kết tài khoản ngân hàng với tài khoản của mình, hệ thống sẽ yêu cầu khách hàng liên kết tài khoản ngân hàng với tài khoản của mình.
Alternative flow	Ở bước 3 , Trước khi chọn phương thức thanh toán, khách hàng được phép thoát khỏi bảng xác nhận đơn hàng và quay về giỏ hàng để chỉnh sửa đơn hàng.

Bảng 4: Use-case xác nhận và gửi đơn hàng

3.5. Use case xử lý đơn hàng (Order processing)

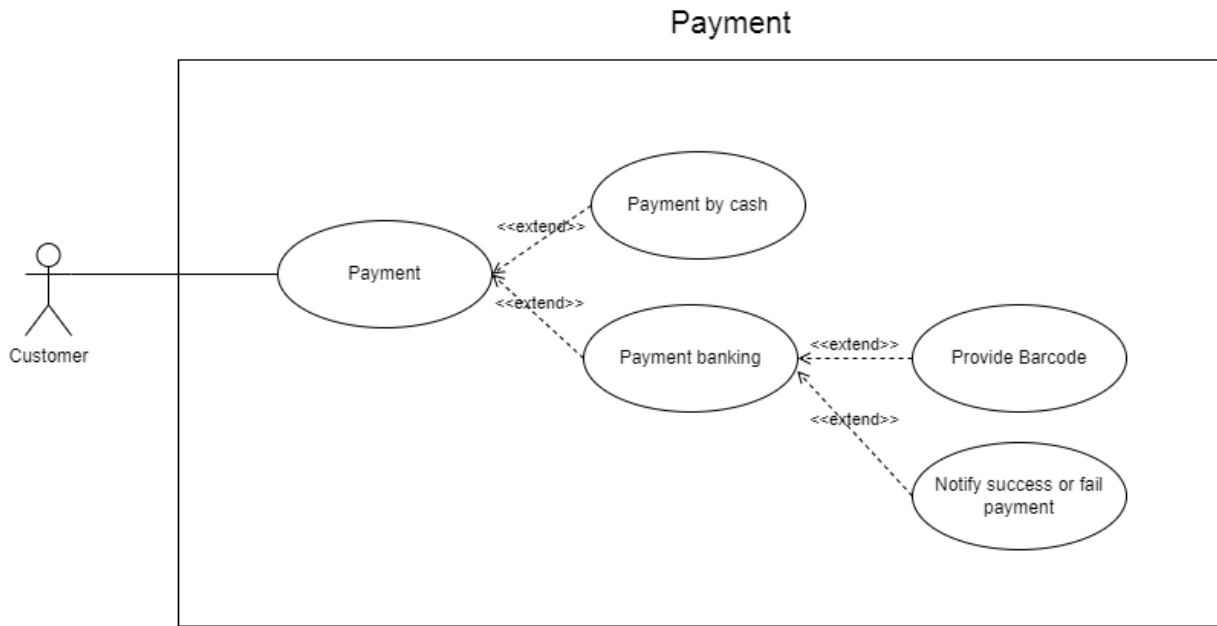


Hình 6: Use-case xử lý đơn hàng

Use Case Name	Xử lý đơn hàng
Actors	Người quản lý
Brief Description	Người quản lý sẽ xem danh sách đơn hàng thành các mục tương ứng với trạng thái của đơn hàng.
Preconditions	Người quản lý đăng nhập quyền admin hệ thống và truy cập vào trang quản lý đơn hàng.
Postconditions	Người quản lý cập nhật được trạng thái của đơn hàng.
Normal Flow	<ol style="list-style-type: none">Người quản lý xem danh sách đơn hàng theo các mục và trạng thái xử lý của chúng. Trạng thái đơn hàng bao gồm: Đang chờ xử lý, Đang in, Đang giao, Đã Thanh toán.Người quản lý có thể nhấn chọn vào một đơn hàng để xem chi tiết.Người quản lý có thể chọn xử lý để từ chối hoặc chuyển đơn hàng sang trạng thái tiếp theo.Người quản lý có thể từ chối một hay nhiều đơn hàng cùng lúc và gửi thông báo từ chối đến khách hàng.

Bảng 5: Use-case xử lý đơn hàng

3.6. Use case thanh toán đơn hàng (Payment order)



Hình 7: Use-case thanh toán đơn hàng

Use Case Name	Thanh toán đơn hàng
Actors	Khách hàng
Brief Description	Khách hàng sử dụng chức năng này để thanh toán đơn hàng.
Preconditions	Khách hàng đã đăng nhập vào tài khoản của mình và đặt hàng thành công.
Postconditions	Khách hàng đã thanh toán đơn hàng thành công.
Normal Flow	<ol style="list-style-type: none"> Khách hàng chọn phương thức thanh toán. Nếu khách hàng chọn thanh toán bằng MoMo, hệ thống sẽ gửi một mã vạch (đã kèm số tiền thanh toán) đến khách hàng. Khách hàng quét mã vạch và thanh toán. Hệ thống sẽ gửi thông báo đến khách hàng và người quản lý về việc khách hàng đã thanh toán thành công.
Exceptional flow	<ol style="list-style-type: none"> 1.1. Ở bước 2, Mã vạch có thời hạn trong 10 phút thanh toán, nếu quá thời hạn, hệ thống sẽ gửi lại mã vạch mới. 1.2. Ở bước 2, Nếu khách hàng chọn thanh toán bằng tiền mặt, hệ thống sẽ gửi thông báo đến khách hàng và người quản lý về việc khách hàng đã chọn thanh toán bằng tiền mặt.
Alternative flow	<ol style="list-style-type: none"> 1.1. Ở bước 1, khách hàng có thể chọn thanh toán bằng tiền mặt.

Bảng 6: Use-case thanh toán đơn hàng

<<<<<<< HEAD

4. Framework & Technology

Phần này sẽ giới thiệu sơ lược về các công nghệ được sử dụng trong dự án. Nhóm quản lý phiên bản mã nguồn với Git và GitHub, sử dụng framework ReactJs cho front-end, Fastify cho back-end, ánh xạ tới cơ sở dữ liệu với Prisma và sử dụng PostgreSQL để quản lý cơ sở dữ liệu. Dưới đây nhóm sẽ giới thiệu tóm lược về mỗi công nghệ đã nêu:

4.1. Git

Git là một hệ thống quản lý phiên bản phân tán nguồn mở, được Linus Torvalds tạo ra vào năm 2005 để giải quyết nhu cầu quản lý mã nguồn hiệu quả trong quá trình phát triển nhân Linux. Hệ thống này đã trở thành một trong những hệ thống quản lý phiên bản phổ biến nhất được sử dụng hiện nay. Git được sử dụng để theo dõi những thay đổi trong mã nguồn trong quá trình phát triển phần mềm và cho phép nhiều lập trình viên làm việc đồng thời trên cùng một cơ sở mã nguồn.

Một số ưu điểm của Git bao gồm tốc độ, sự đơn giản, tính chất phân tán hoàn toàn, hỗ trợ phát triển song song và tính toàn vẹn. Git được thiết kế để hoạt động nhanh và hiệu quả, ngay cả khi làm việc với các cơ sở mã nguồn lớn. Nó có giao diện dòng lệnh đơn giản và trực quan giúp lập trình viên dễ dàng tìm hiểu và sử dụng. Git là một hệ thống kiểm soát phiên bản phân phối hoàn toàn, có nghĩa là mọi lập trình viên đều có bản sao hoàn chỉnh của cơ sở mã trên máy cục bộ của họ. Điều này giúp dễ dàng làm việc ngoại tuyến và cộng tác với các lập trình viên khác. Git cũng hỗ trợ việc phát triển song song, cho phép nhiều lập trình viên làm việc đồng thời trên cùng một cơ sở mã mà không có xung đột. Git sử dụng các hàm băm mật mã để đảm bảo tính toàn vẹn của cơ sở mã, khiến cho việc sửa đổi mã một cách vô tình hoặc cố ý mà không bị phát hiện là gần như không thể.

4.2. GitHub

GitHub là một nền tảng web bổ sung cho các chức năng của Git. GitHub cung cấp dịch vụ lưu trữ cho các kho Git (Git repositories) cùng với nhiều tính năng quản lý dự án và cộng tác. Với GitHub, người dùng có thể lưu trữ kho Git của họ trên Internet, giúp chúng có thể truy cập được từ mọi nơi có kết nối Internet. Cách tiếp cận này giúp đơn giản hóa việc cộng tác vì các thành viên trong nhóm có thể dễ dàng sao chép các kho Git, tạo nhánh và gửi “yêu cầu kéo” (Pull Requests) để xem xét và tích hợp mã.

GitHub cũng hỗ trợ việc quản lý dự án bằng cách cung cấp các công cụ để theo dõi sự cố, báo cáo lỗi và quản lý các yêu cầu tính năng. Ngoài ra, nó tích hợp tốt với các dịch vụ tích hợp liên tục (Continuous Integration) như GitHub Actions, cho phép các quy trình được kiểm thử và triển khai tự động.

4.3. ReactJS

ReactJS là một thư viện JavaScript được sử dụng để xây dựng giao diện người dùng. Nó được tạo ra bởi Jordan Walke, một kỹ sư phần mềm tại Facebook và được Facebook phát triển và duy trì. ReactJS là một thư viện khai báo, hiệu quả và linh hoạt, chịu trách nhiệm về lớp View của ứng dụng. Nó cho phép các nhà phát triển tạo các thành phần UI có thể tái sử dụng, có thể được lồng ghép với các thành phần khác để xây dựng các ứng dụng phức tạp từ các khối xây dựng đơn giản. ReactJS sử dụng cơ chế dựa trên DOM ảo để điền dữ liệu vào HTML DOM, cơ chế này hoạt động nhanh vì nó chỉ thay đổi các thành phần DOM riêng lẻ thay vì tải lại toàn bộ DOM mỗi lần.

ReactJS có một số lợi thế so với các JavaScript framework khác như dễ sử dụng, tốc độ nhanh và có khả năng mở rộng. Nó giúp các lập trình viên xây dựng các ứng dụng web có kích thước lớn và sửa đổi dữ liệu mà tránh việc tải lại trang không cần thiết. ReactJS chỉ xử lý lớp View trong mẫu MVC của giao diện người dùng ứng dụng. ReactJS cập nhật và hiển thị các thành phần một cách hiệu quả, đồng thời xử lý khéo léo các bản cập nhật DOM. Nó có thể được tích hợp vào bất kỳ ứng dụng nào, cho dù đó là một phần của trang hay một trang hoàn chỉnh hay thậm chí là toàn bộ ứng dụng. ReactJS cũng có khả năng tương thích ngược, nghĩa là nó cũng có thể được sử dụng với các phiên bản trình duyệt cũ hơn.

4.4. Fastify

Fastify là một web framework dành cho Node.js tập trung chủ yếu vào việc cung cấp trải nghiệm tốt lập trình viên với chi phí thấp và một kiến trúc plugin mạnh mẽ. Fastify lấy cảm hứng từ Hapi và Express và là một trong những web framework nhanh nhất hiện tại. Fastify có khả năng mở rộng thông qua các hook, plugin và decorators. Fastify có một trình logging được tích hợp sẵn, từ đó gần như loại bỏ chi phí cho tác vụ logging. Fastify thân thiện với người dùng và được xây dựng để mang tính diễn đạt cao mà không làm giảm hiệu suất và tính bảo mật.

Một trong những ưu điểm chính của Fastify nằm ở tốc độ, đạt được nhờ tối thiểu hóa chi phí và tập trung vào hiệu suất. Nó cung cấp tính năng tự động serialize và deserialize dữ liệu đến và đi, hỗ trợ việc xác thực các request, tham số định tuyến và chuẩn hóa đầu vào, khiến nó trở thành một lựa chọn an toàn để phát triển web.

4.5. Prisma

Prisma là một ORM thế hệ mới giúp đơn giản hóa quy trình làm việc với cơ sở dữ liệu và thay thế các ORM truyền thống. Nó cung cấp khả năng truy cập cơ sở dữ liệu an toàn kiểu (type-safe) với Prisma Client, di chuyển (migrate) dữ liệu với Prisma Migrate cũng như quản lý dữ liệu trực quan với Prisma

Studio. Prisma Client có thể được sử dụng để xây dựng các API GraphQL, REST, gRPC và hơn thế nữa. Prisma hiện hỗ trợ nhiều hệ thống quản lý cơ sở dữ liệu như MySQL, PostgreSQL và MongoDB.

Một số lợi ích của Prisma có thể kể đến như an toàn kiểu, API hiện đại và đọc/ghi dữ liệu quan hệ một cách linh hoạt. Prisma thống nhất quyền truy cập vào nhiều cơ sở dữ liệu cùng một lúc và từ đó giảm đáng kể độ phức tạp trong các quy trình làm việc trên nhiều cơ sở dữ liệu. Nó cũng có hệ thống sự kiện và streaming theo thời gian thực cho cơ sở dữ liệu, đảm bảo người dùng nhận được cập nhật cho tất cả các sự kiện quan trọng xảy ra trong cơ sở dữ liệu của mình.

4.6. PostgreSQL

PostgreSQL, được phát triển và duy trì bởi một nhóm các nhà phát triển phần mềm PostgreSQL Global Development Group, là một hệ thống cơ sở dữ liệu quan hệ đối tượng nguồn mở sử dụng và mở rộng ngôn ngữ SQL kết hợp với nhiều tính năng lưu trữ và mở rộng quy mô một cách an toàn cho các khối lượng dữ liệu phức tạp nhất. PostgreSQL nổi tiếng bởi kiến trúc đã được chứng minh, độ tin cậy, tính toàn vẹn dữ liệu, bộ tính năng mạnh mẽ, khả năng mở rộng và cộng đồng nguồn mở đẳng cấp sau cung cấp nhất quán các giải pháp sáng tạo và hiệu quả.

PostgreSQL đi kèm với nhiều tính năng nhằm giúp các lập trình viên xây dựng ứng dụng, các quản trị viên bảo vệ tính toàn vẹn của dữ liệu và xây dựng môi trường có khả năng đối phó với lỗi, đồng thời giúp người dùng quản lý dữ liệu của mình dù lớn hay nhỏ. Ngoài việc là nguồn mở và miễn phí, PostgreSQL còn có khả năng mở rộng cao. Chẳng hạn như người dùng có thể tạo loại dữ liệu của riêng mình, xây dựng các hàm tùy chỉnh, thậm chí viết mã từ các ngôn ngữ lập trình khác nhau mà không cần biên dịch lại cơ sở dữ liệu.

4.6.1.1.1.1.

5. Framework & Technology

Phần này sẽ giới thiệu sơ lược về các công nghệ được sử dụng trong dự án. Nhóm quản lý phiên bản mã nguồn với Git và GitHub, sử dụng framework ReactJs cho front-end, Fastify cho back-end, ánh xạ tới cơ sở dữ liệu với Prisma và sử dụng PostgreSQL để quản lý cơ sở dữ liệu. Dưới đây nhóm sẽ giới thiệu tóm lược về mỗi công nghệ đã nêu:

5.1. Git

Git là một hệ thống quản lý phiên bản phân tán nguồn mở, được Linus Torvalds tạo ra vào năm 2005 để giải quyết nhu cầu quản lý mã nguồn hiệu quả trong quá trình phát triển nhân Linux. Hệ thống này đã trở thành một trong những hệ thống quản lý phiên bản phổ biến nhất được sử dụng hiện nay. Git được sử dụng để theo dõi những thay đổi trong mã nguồn trong quá trình phát triển phần mềm và cho phép nhiều lập trình viên làm việc đồng thời trên cùng một cơ sở mã nguồn.

Một số ưu điểm của Git bao gồm tốc độ, sự đơn giản, tính chất phân tán hoàn toàn, hỗ trợ phát triển song song và tính toàn vẹn. Git được thiết kế để hoạt động nhanh và hiệu quả, ngay cả khi làm việc với các cơ sở mã nguồn lớn. Nó có giao diện dòng lệnh đơn giản và trực quan giúp lập trình viên dễ dàng tìm hiểu và sử dụng. Git là một hệ thống kiểm soát phiên bản phân phối hoàn toàn, có nghĩa là mọi lập trình viên đều có bản sao hoàn chỉnh của cơ sở mã trên máy cục bộ của họ. Điều này giúp dễ dàng làm việc ngoại tuyến và cộng tác với các lập trình viên khác. Git cũng hỗ trợ việc phát triển song song, cho phép nhiều lập trình viên làm việc đồng thời trên cùng một cơ sở mã mà không có xung đột. Git sử dụng các hàm băm mật mã để đảm bảo tính toàn vẹn của cơ sở mã, khiến cho việc sửa đổi mã một cách vô tình hoặc cố ý mà không bị phát hiện là gần như không thể.

5.2. GitHub

GitHub là một nền tảng web bổ sung cho các chức năng của Git. GitHub cung cấp dịch vụ lưu trữ cho các kho Git (Git repositories) cùng với nhiều tính năng quản lý dự án và cộng tác. Với GitHub, người dùng có thể lưu trữ kho Git của họ trên Internet, giúp chúng có thể truy cập được từ mọi nơi có kết nối Internet. Cách tiếp cận này giúp đơn giản hóa việc cộng tác vì các thành viên trong nhóm có thể dễ dàng sao chép các kho Git, tạo nhánh và gửi “yêu cầu kéo” (Pull Requests) để xem xét và tích hợp mã.

GitHub cũng hỗ trợ việc quản lý dự án bằng cách cung cấp các công cụ để theo dõi sự cố, báo cáo lỗi và quản lý các yêu cầu tính năng. Ngoài ra, nó tích hợp tốt với các dịch vụ tích hợp liên tục (Continuous Integration) như GitHub Actions, cho phép các quy trình được kiểm thử và triển khai tự động.

5.3. ReactJS

ReactJS là một thư viện JavaScript được sử dụng để xây dựng giao diện người dùng. Nó được tạo ra bởi Jordan Walke, một kỹ sư phần mềm tại Facebook và được Facebook phát triển và duy trì. ReactJS là một thư viện khai báo, hiệu quả và linh hoạt, chịu trách nhiệm về lớp View của ứng dụng. Nó cho phép các nhà phát triển tạo các thành phần UI có thể tái sử dụng, có thể được lồng ghép với các thành phần khác để xây dựng các ứng dụng phức tạp từ các khối xây dựng đơn giản. ReactJS sử dụng cơ chế dựa trên DOM ảo để điền dữ liệu vào HTML DOM, cơ chế này hoạt động nhanh vì nó chỉ thay đổi các thành phần DOM riêng lẻ thay vì tải lại toàn bộ DOM mỗi lần.

ReactJS có một số lợi thế so với các JavaScript framework khác như dễ sử dụng, tốc độ nhanh và có khả năng mở rộng. Nó giúp các lập trình viên xây dựng các ứng dụng web có kích thước lớn và sửa đổi dữ liệu mà tránh việc tải lại trang không cần thiết. ReactJS chỉ xử lý lớp View trong mẫu MVC của giao diện người dùng ứng dụng. ReactJS cập nhật và hiển thị các thành phần một cách hiệu quả, đồng thời xử lý khéo léo các bản cập nhật DOM. Nó có thể được tích hợp vào bất kỳ ứng dụng nào, cho dù đó là một phần của trang hay một trang hoàn chỉnh hay thậm chí là toàn bộ ứng dụng. ReactJS cũng có khả năng tương thích ngược, nghĩa là nó cũng có thể được sử dụng với các phiên bản trình duyệt cũ hơn.

5.4. Fastify

Fastify là một web framework dành cho Node.js tập trung chủ yếu vào việc cung cấp trải nghiệm tốt lập trình viên với chi phí thấp và một kiến trúc plugin mạnh mẽ. Fastify lấy cảm hứng từ Hapi và Express và là một trong những web framework nhanh nhất hiện tại. Fastify có khả năng mở rộng thông qua các hook, plugin và decorators. Fastify có một trình logging được tích hợp sẵn, từ đó gần như loại bỏ chi phí cho tác vụ logging. Fastify thân thiện với người dùng và được xây dựng để mang tính diễn đạt cao mà không làm giảm hiệu suất và tính bảo mật.

Một trong những ưu điểm chính của Fastify nằm ở tốc độ, đạt được nhờ tối thiểu hóa chi phí và tập trung vào hiệu suất. Nó cung cấp tính năng tự động serialize và deserialize dữ liệu đến và đi, hỗ trợ việc xác thực các request, tham số định tuyến và chuẩn hóa đầu vào, khiến nó trở thành một lựa chọn an toàn để phát triển web.

5.5. Prisma

Prisma là một ORM thế hệ mới giúp đơn giản hóa quy trình làm việc với cơ sở dữ liệu và thay thế các ORM truyền thống. Nó cung cấp khả năng truy cập cơ sở dữ liệu an toàn kiểu (type-safe) với Prisma Client, di chuyển (migrate) dữ liệu với Prisma Migrate cũng như quản lý dữ liệu trực quan với Prisma

Studio. Prisma Client có thể được sử dụng để xây dựng các API GraphQL, REST, gRPC và hơn thế nữa. Prisma hiện hỗ trợ nhiều hệ thống quản lý cơ sở dữ liệu như MySQL, PostgreSQL và MongoDB.

Một số lợi ích của Prisma có thể kể đến như an toàn kiểu, API hiện đại và đọc/ghi dữ liệu quan hệ một cách linh hoạt. Prisma thống nhất quyền truy cập vào nhiều cơ sở dữ liệu cùng một lúc và từ đó giảm đáng kể độ phức tạp trong các quy trình làm việc trên nhiều cơ sở dữ liệu. Nó cũng có hệ thống sự kiện và streaming theo thời gian thực cho cơ sở dữ liệu, đảm bảo người dùng nhận được cập nhật cho tất cả các sự kiện quan trọng xảy ra trong cơ sở dữ liệu của mình.

5.6. PostgreSQL

PostgreSQL, được phát triển và duy trì bởi một nhóm các nhà phát triển phần mềm PostgreSQL Global Development Group, là một hệ thống cơ sở dữ liệu quan hệ đối tượng nguồn mở sử dụng và mở rộng ngôn ngữ SQL kết hợp với nhiều tính năng lưu trữ và mở rộng quy mô một cách an toàn cho các khối lượng dữ liệu phức tạp nhất. PostgreSQL nổi tiếng bởi kiến trúc đã được chứng minh, độ tin cậy, tính toàn vẹn dữ liệu, bộ tính năng mạnh mẽ, khả năng mở rộng và cộng đồng nguồn mở đẳng cấp sau cung cấp nhất quán các giải pháp sáng tạo và hiệu quả.

PostgreSQL đi kèm với nhiều tính năng nhằm giúp các lập trình viên xây dựng ứng dụng, các quản trị viên bảo vệ tính toàn vẹn của dữ liệu và xây dựng môi trường có khả năng đối phó với lỗi, đồng thời giúp người dùng quản lý dữ liệu của mình dù lớn hay nhỏ. Ngoài việc là nguồn mở và miễn phí, PostgreSQL còn có khả năng mở rộng cao. Chẳng hạn như người dùng có thể tạo loại dữ liệu của riêng mình, xây dựng các hàm tùy chỉnh, thậm chí viết mã từ các ngôn ngữ lập trình khác nhau mà không cần biên dịch lại cơ sở dữ liệu.



>>>>>> d352de4ae52d5400df45adf373785c58b348ffad

Tài liệu tham khảo