|  |
| --- |
| TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI  KHOA CÔNG NGHỆ THÔNG TIN  Kết quả hình ảnh cho logo UTC  BÀI TẬP LỚN  **Chuyên đề: Công nghệ phần mềm**  **ĐỀ TÀI: Xây dựng Website bán sách Bookland**  Tên sinh viên: Lê Anh Phương.  Mã số sinh viên:191212555.  Lớp: CNTT5 – K60  Nhóm 10  Giảng viên hướng dẫn: Đào Thị Lệ Thủy    Hà Nội, tháng 8 năm 2022. |

**Mở đầu**

Ngày nay, Công nghệ thông tin đã có những bước phát triển mạnh mẽ theo cả chiều rộng và sâu. Máy tin hs điện tử không còn là một thứ phương tiện quý hiếm mà đang ngày càng trở lthành công cụ làm việc và giải trí thông dụng của con người, không chỉ ở nơi làm việc mà còn ngay cả trong gia đình. Và từ đó Website đã đóng một vai trò quan trọng đối với con người chúng ta từ giải trí cho đến quảng cáo, thương mại, quản lý…Website (thương mại điện tử) sẽ dần dần thay thế những phương thức kinh doanh cũ trong các doanh nghiệp bởi tính ưu việt mà Website mang lại như: nhanh hơn, rẽ hơn, tiện dụng hơn, hiệu quả hơn và không bị giới hạn không gian và thời gian.

Xuất phát từ những nhu cầu thực tế đó nhóm chúng em quyết định “Xây dựng Webstie bán giày” nhằm tạo ra hệ thống quản lý bán hàng có tính thiết thực và ứng dụng vào thực tế. Cửa hàng cón thể đưa các sản phẩm lên Website của mình và quản lý website đó, cửa hàng sẽ gửi dsanr phẩm đến tận tay khách hàng. Website là nơi của hàng qunagr bas tốt nhất các sản phẩm mình bán ra.

Quan đây chúng em cin gửi lời cảm ơn chân thành và bày tỏ lòng biết ơn đến sự tậm tình của các thầy cô trong khoa “Công nghệ thông tin”. Đặc biệt là cô Đào Thị Lệ Thuỷ, người đã giảng dạy, hướng dẫn chúng em trong thời gian thực hiện học phần này .

**Mục lục**

1. Tổng quan về đề tài……………………………………………………………. 2
2. Công nghệ áp dụng…………………………………………………………. 2
3. Tìm hiểu bài toán…………………………………………………………….6
   1. Mục đích
   2. Phạm vi
   3. Khảo sát các ứng dụng tương tự
4. Phân tích và thiết kế……………………………………………………………..7
   1. Phân tích và thiết kế chức năng……………………………………………...7
   2. Phân tích nghiệp vụ
   3. Biểu đồ phân cấp chức năng
   4. Biểu đồ luồng dữ liệu
   5. Biểu đồ usecase
   6. Biểu đồ tuần tự
   7. Phân tích và thiết kế dữ liệu …………………………………………………10
   8. Phân tích dữ liệu
   9. Các thực thể và mối quan hệ giữa các thực thể,code cài đặt
   10. Biểu đồ quan hệ
5. Xây dựng và cài đặt chương trình………………………………………………..27
6. Xây dựng trang quản trị cho người dùng, trang thành viên…………………..27
7. Xây dưng trang quản trị………………………………………………………34
8. Code xử lý…………………………………………………………………….41
9. Kết luận…………………………………………………………………………..42
10. **Tổng quan đề tài.**

1. *Công nghệ áp dụng*

* Nodejs: là một mã nguồn mở, đa nền tảng, chạy trên môi trường JavaSript, được xây dựng trên V8 JavaScript engine của Chrome - V8 thực thi mã JavaScript bên ngoài trình duyệt. Nó được tạo ra vào năm 2009 đi kèm với một lợi thế chính - NodeJS cho phép thực hiện lập trình bất đồng bộ.
* Ở chế độ đồng bộ thực thi từng dòng và tiến hành thực thi dòng tiếp theo khi dòng hiện tại đã thực thi xong.
* Khi bất đồng bộ thực thi tất cả dòng code cùng một lúc.
* NodeJS cũng cung cấp một thư viện bao gồm rất nhiều các module JavaScript khác nhau nhằm đơn giản hóa việc phát triển các ứng dụng web, qua đó giảm thiểu tình trạng sử dụng quá nhiều Node.js.
* Các tính năng: lập trình hướng sự kiện và không đồng bộ, cực kì nhanh chóng, đơn luồng/ single thread nhưng có khả năng mở rộng cao, không có buffer,..
* Ưu điểm:

+ IO hưóng điếm sự kiện không đồng bộ, cho phép xử lý nhiều yêu cầu đồng thời.

+ Sử dụng Javascript - một ngôn ngữ lập trình dễ đọc.

+ Chia sẻ cùng code ở cả phía client và server.

+ NPM(Node Package Manager) và module Node đang ngày càng phát triển mạnh mẽ.

+ Cộng đồng hỗ trợ tích cực.

+ Cho phép stream các file có kích thước lớn.

* Nhược điểm:

+ Không có khả năng mở rộng, vì vậy không thể tận dụng lợi thế mô hình đa lõi trong các phần cứng server hiện nay.

+ Khó thao tác với cơ sở dữ liệu quan hệ.

+ Mỗi callback sẽ đi kèm với rất nhiều callback lồng nhau khác.

+ Cần có kiến thức tốt về Javascript.

+ Không phù hợp với các tác vụ đòi hỏi nhiều CPU

* Cho phét develop nhanh và hiệu quả hơn.
* Khả năng mở rộng tốt,và dễ bảo trì ứng dụng
* Là framework Node.js phát triển mạnh mẽ nhất trong 3 năm trở lại đây.
* Cộng đồng hỗ trợ lớn, tích cực
* Kết hợp phát triển front-end và mid-tier, một đặc điểm vượt trội so với hầu hết các ngôn ngữ khác.
* Nguồn tài liệu hưóng dẫn phong phú, chi tiết.
* Quá trình unit testing trở lên đơn giản hơn
* Được xây dựng chuyên dùng cho các ứng dụng doanh nghiệp có quy mô lớn.
* Cung cấp các kiến trúc ứng dụng độc lập, cho phét các developer tạo ra những ứng dụng dễ test, dễ mở rộng và dễ bảo trì.
* Cho phép xây dựng ứng dụng Rest API, MVC, microservice, GrapQL, Web Socket hay CRON job
* Cấu trúc chủ yếu dưạ vào Angular – rất đơn giản và cho phét tập trung vào việc thiết kế endpoint thay vì cấu trúc của ứng dụng.
* Cung cấp các module, dịch vụ và controller giống Angular, cho phét ứng dụng các khả năng mở rộng và test tốt hơn so với express hay Koa
* Có tính chất kết nối lỏng, do đó những lỗi cũ trong dự án không ảnh hưởng dến cấu trúc codebase sau này.
* MySQL:
* MySQL đang tạo ra bảng để có thể lưu trữ dữ liệu và định nghĩa về sự liên quan giữa những bảng đó
* Client sẽ trực tiếp gửi yêu cầu SQL bằng 1 lệnh đặc biệt có trên MySQL.
* Ứng dụng tại server sẽ tiến hành phản hồi thông tin cũng như trả về những kết quả trên máy client.
* Ưu điểm:

**+ Nhanh chóng:**Nhờ vào việc đưa ra một số những tiêu chuẩn và cho phép MySQL làm việc hiệu quả cũng như tiết kiệm chi phí, giúp gia tăng tốc độ thực thi.

**+ Mạnh mẽ và khả năng mở rộng:** MySQL hoàn toàn có thể xử lý số lượng lớn dữ liệu và đặc biệt hơn thế nữa thì nó còn có thể mở rộng nếu như cần thiết.

**+ Đa tính năng:** Ưu điểm MySQL là gì? MySQL hiện đang hỗ trợ nhiều những chức năng SQL rất được mong chờ từ 1 hệ quản trị CSDL quan hệ cả gián tiếp cũng như trực tiếp.

**+ Độ bảo mật cao:** MySQL là gì? Hiện tại nó đang rất thích hợp cho những ứng dụng truy cập CSDL thông qua internet khi sở hữu rất nhiều những tính năng về bảo mật và thậm chí là đang ở cấp cao.

+ Dễ dàng sử dụng: MySQL đang là cơ sở dữ liệu dễ sử dụng, ổn định, tốc độ cao và hoạt động trên rất nhiều những hệ điều hành đang cung cấp 1 hệ thống lớn những hàm tiện ích rất mạnh.

* Nhược điểm:

**+ Dung lượng hạn chế:** Trong trường hợp nếu như số lượng bản ghi của bạn đang lớn dần lên thì khi đó quá trình truy xuất dữ liệu sẽ diễn ra vô cùng khó khăn. Như vậy cần phải áp dụng rất nhiều những biện pháp khác nhau để có thể gia tăng được tốc độ truy xuất những dữ liệu ví dụ như tạo cache MySQL hoặc chia tải database ra nhiều server.

**+ Độ tin cậy:**Nhược điểm MySQL là gì? Theo đó cách thức nhận chức năng cụ thể đang được xử lý cùng với MySQL (ví dụ như kiểm toán, những giao dịch, tài liệu tham khảo,…) khiến cho nó trở nên kém tin cậy hơn một số những hệ quản trị về cơ sở dữ liệu có quan hệ khác.

**+ Giới hạn:** MySQL là gì? Theo thiết kế thì MySQL không có ý định thực hiện toàn bộ và nó đang đi kèm cùng với những hạn chế liên quan tới chức năng mà một số ứng dụng có thể cần tới.

* HTML/CSS:
* ***HTML*, HyperText Markup Language**, cung cấp cấu trúc nội dung và ý nghĩa bằng cách xác định nội dung đó, ví dụ như tiêu đề, đoạn văn hoặc hình ảnh.
* ***CSS*, hay Cascading Style Sheets**, là ngôn ngữ trình bày được dùng để tạo kiểu cho sự xuất hiện của nội dung sử dụng, ví dụ như phông chữ hoặc màu sắc.
* Hai ngôn ngữ HTML và CSS độc lập với nhau và vẫn giữ nguyên như vậy. CSS không nên được viết bên trong một tài liệu HTML và ngược lại. Theo quy định, HTML sẽ luôn đại diện cho nội dung và CSS sẽ luôn thể hiện sự xuất hiện của nội dung đó.
* Javascrip/typescrip:
* **JavaScript** là ngôn ngữ lập trình  phổ biến dùng để tạo ra các trang web tương tác. Được tích hợp và nhúng vào HTML giúp website trở nên sống động hơn. JavaScript đóng vai trò như một phần của trang web, thực thi cho phép Client-Side Script từ phía người dùng cũng như phía máy chủ ([Nodejs](https://vietnix.vn/nodejs-la-gi/)) tạo ra các trang web động.

**+ JavaScript** là một **ngôn ngữ lập trình thông dịch** với khả năng hướng đến đối tượng. Là một trong 3 ngôn ngữ chính trong lập trình web và có mối liên hệ lẫn nhau để xây dựng một website sống động, chuyên nghiệp.

**+** JavaScript thường sẽ được nhúng trực tiếp vào một trang web hoặc được tham chiếu qua file .js riêng. JavaScript là ngôn ngữ từ phía client nên script sẽ được tải về máy client khi truy cập và được xử lý tại đó. Thay vì tải về máy server và sau khi xử lý xong mới phản hồi kết quả đến client.

**+ Ưu điểm:**

Chương trình JavaScript rất dễ học.

Lỗi JavaScript dễ phát hiện và sẽ giúp bạn sửa lỗi nhanh hơn.

Các trình duyệt web có thể dịch nó bằng HTML mà không cần một compiler.

JS hoạt động trên rất nhiều nền tảng và trình duyệt khác nhau.

Được đánh giá là ngôn ngữ lập trình nhẹ, nhanh so với các ngôn ngữ khác.

JS có thể được gắn trên một số element hoặc các events của trang web.

Khi website có sử dụng JS thì sẽ giúp cho trang web đó tương tác và tăng trải nghiệm người dùng khi truy cập.

Bạn có thể tận dụng JavaScript để kiểm tra các input thay vì kiểm tra thủ công thông qua việc truy xuất database.

Giao diện phong phú gồm các thành phần Drag and Drop, Slider để cung cấp một Rich Interface (Giao diện giàu tính năng).

+ Nhược điểm:

Dễ bị khai thác từ những hacker và scammer.

Có thể được dụng để thực thi mã độc trên máy tính của người dùng.

JS code snippet lớn.

Các thiết bị khác nhau có thể thực hiện JS khác nhau dẫn đến không đồng nhất.

Vì tính bảo mật nên client-side JavaScript không cho phép đọc và ghi các file.

JS không được hỗ trợ khi sử dụng trong kết nối mạng.

JavaScript không có khả năng đa luồng hoặc đa xử lý.

* Bootstrap: cho phép quá trình thiết kế website diễn ra nhanh chóng và dễ dàng hơn dựa trên những thành tố cơ bản sẵn có như typography, forms, buttons, tables, grids, navigation, image carousels…
* *Bootstrap* là một bộ sưu tập miễn phí của các *mã nguồn mở* và công cụ dùng để tạo ra một mẫu webiste hoàn chỉnh. Với các thuộc tính về giao diện được quy định sẵn như kích thước, màu sắc, độ cao, độ rộng…, các designer có thể sáng tạo nhiều sản phẩm mới mẻ nhưng vẫn tiết kiệm thời gian khi làm việc với framework này trong quá trình *thiết kế giao diện website*.
* Dưới đây là 3 File chính giúp quản lý các chức năng của Website và giao diện người dùng:

+ Bootstrap.CSS

+ Bootstrap.JS

+ Glyphicons

* Ajax: là chữ viết tắt của cụm từ Asynchronous Javascript and XML. AJAX là phương thức trao đổi dữ liệu với máy chủ và cập nhật một hay nhiều phần của trang web, hoàn toàn không reload lại toàn bộ trang.
* Ajax được viết bằng javascript chạy trên client, tức là mỗi browser sẽ chạy độc lập hoàn toàn không ảnh hưởng lẫn nhau. Về mặt kỹ thuật, nó đề cập đến việc sử dụng các đối tượng XmlHttpRequest để tương tác với một máy chủ web thông qua Javascript.
* AJAX là một trong những công cụ giúp chúng ta đem lại cho người dùng trải nghiệm tốt hơn. Khi cần một thay đổi nhỏ thì sẽ không cần load lại cả trang web, làm trang web phải tải lại nhiều thứ không cần thiết.
* Những lợi ích mà ajax mang lại:

+ AJAX được sử dụng để thực hiện một callback. Được dùng để thực hiện việc truy xuất dữ liệu hoặc lưu trữ dữ liệu mà không cần phải reload lại toàn bộ trang web. Với những server nhỏ thì việc này cũng tiết kiệm được băng thông cho chúng ta hơn.

+ Cần gì thì chỉ gửi dữ liệu phần đó, load lại 1 phần nhỏ để cập nhật thông tin chứ không load cả trang. Bằng cách này thì có thể giảm thiểu được tốc độ tải trang giúp người dùng có trải nghiệm tốt hơn.

+ Trang web bạn tạo ra cũng sẽ đa dạng và động hơn.

1. Tìm hiểu bài toán
   1. Mục đích

Xây dựng website demo bán giày giúp người dùng dễ tương tác với hệ thống, vừa giảm bớt các thao tác thủ công lại vừa đem lại tính chính xác và hiệu quả trong công tác quản lí hoạt động kinh doanh. Khách hàng hàng dễ tiếp cận đến các sản phẩm.

* 1. *Quy mô phạm vi:*

Quy mô xây dựng website demo cho các chức năng cơ bản của một cửa hàng vừa.

* 1. *Khảo sát các website tương tự:*
* Website shoppe
* Website Fahasa

+ Các website trên đều là các website thương mại điện tử lớn lên nó có đầy đủ tính năng cần thiết.

+ Dễ tương tác hệ thống với người dùng.

* Qua khảo sát thấy được một website bán hàng có chức năng như sau:
* Quản lý người dùng
* Quản lý sản phẩm nhập
* Quản lý loại/ hãng sản xuất sản phẩm
* Quản lý sản phẩm
* Quản lý giỏ hàng
* Quản lý voucher
* Quản lý hóa đơn bán
* Quản lý hóa đơn nhập

1. **Phân tích và thiết kế** 
   1. Phân tích và thiết kế chức năng
   2. Phân tích nghiệp vụ

Khách hàng truy cập vào website để tra cứu, tìm kiếm các loại sản phẩm sách theo tên sách , ISBN , tên tác giả. Thông tin mỗi loại sản phẩm bao gồm: mã loại sản phẩm, tên loại sản phẩm. Trong mỗi loại sản phẩm lại bao gồm nhiều sản phẩm khác nhau. Thông tin mỗi sản phẩm bao gồm: mã sản phẩm, tên sản phẩm, số lượng, thông tin, ảnh, màu,… Khách hàng có thể tìm kiếm theo tên,nhà xuất bản,loại. Khách hàng có thể xem các sản phẩm yêu thích của mình.

Khi khách hàng muốn thêm sản phẩm vào giỏ hàng của mình thì khách hàng phải đăng kí tài khoản và xác thực tài khoản của mình bằng cách nhận trong email. Thông tin mỗi tài khoản bao gồm mã khách hàng, tên khách hàng, email, password, địa chỉ, số điện thoại, … Sau đó tiến hàng đăng nhập vào hệ thống website theo email và password mà mình đã đăng kí hoặc người dùng có thể login tài khoản google. Nếu khách hàng quên mật khẩu của mình thì hệ thống yêu cầu nhập tài khoản email, hệ thống sẽ gửi link vào email khi đó người dùng click vào gửi yêu cầu đến hệ thống để tạo tài khoản mới.

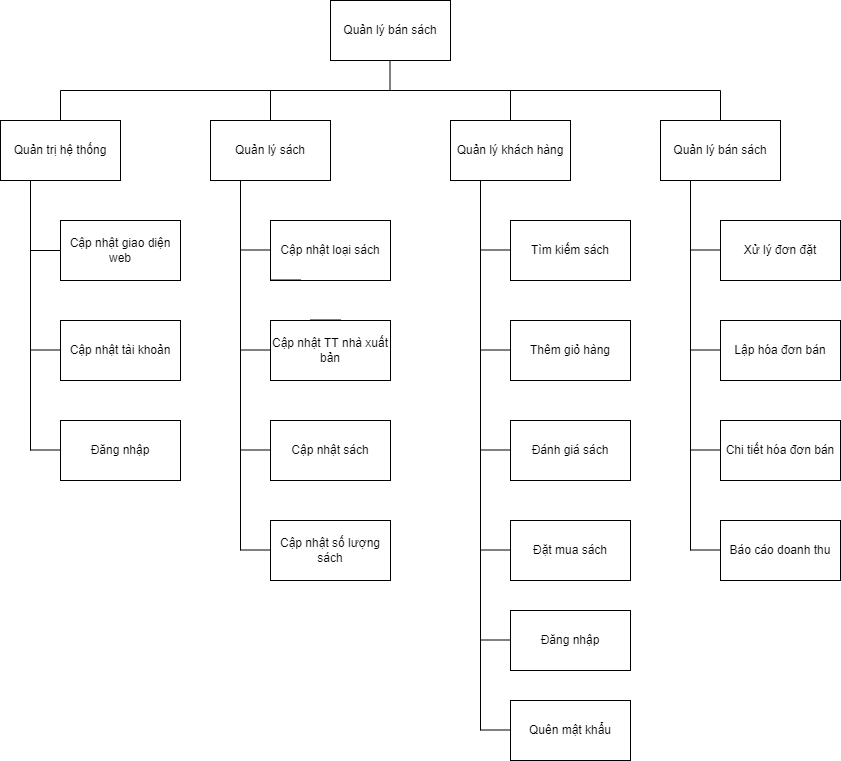
Khách hàng chọn mặt hàng mà mình mong muốn bằng cách ấn nút mua mặt hàng. Nếu muốn xóa sản phẩm trong giỏ hàng thì nhấn dấu X. Khi chọn xong, người dùng ấn nút thanh toán thì hệ thống thiết lập hóa đơn và chi tiết hóa đơn. Thông tin hóa đơn gồm mã hóa đơn ngày đặt, địa chỉ, số điện thoại, note, tổng tiền, trạng thái,… Còn đối với chi tiết hóa đơn bao gồm: mã chi tiết hóa đơn, mã sản phẩm, mã hóa đơn, số lượng, giá tiền mỗi loại,…Người quản trị sẽ cập nhật trạng thái hóa đơn và tiến hành giao hàng theo yêu cầu của khách hàng.

Khách hàng có thể nhập voucher nếu voucher tồn tại hợp lệ. Voucher gồm: mã voucher, tên voucher, số lượng, thời gian bắt đầu, thời gian kết thúc,…

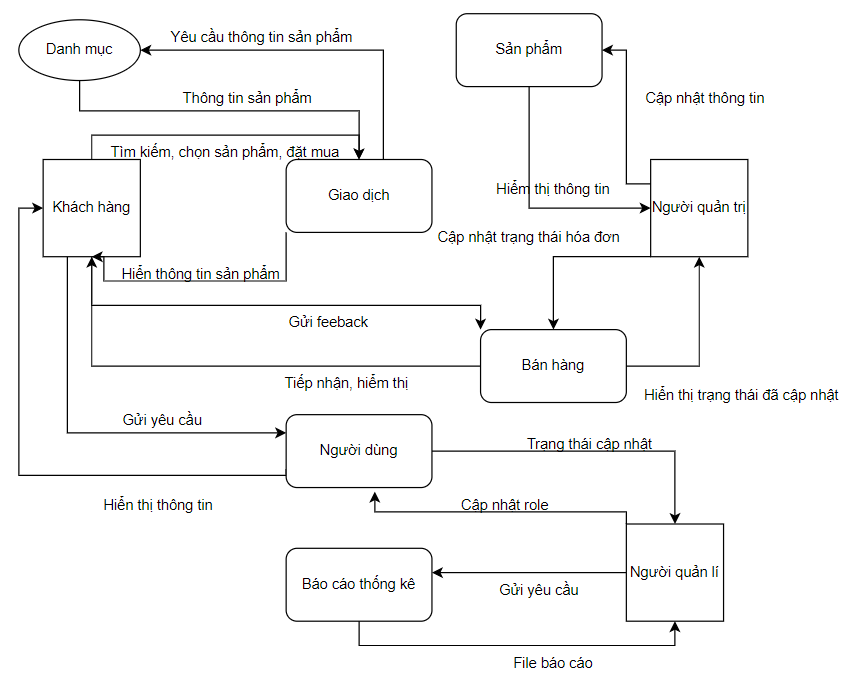
Người quản trị có chức năng CRUD sản phẩm, danh mục sản phẩm, tên tác giả,tên nhà xuất bản , voucher, cập nhật trạng thái cho sản phẩm.

Manager sẽ thực hiện xem báo cáo và cấp quyền cho các tài khoản quản trị.

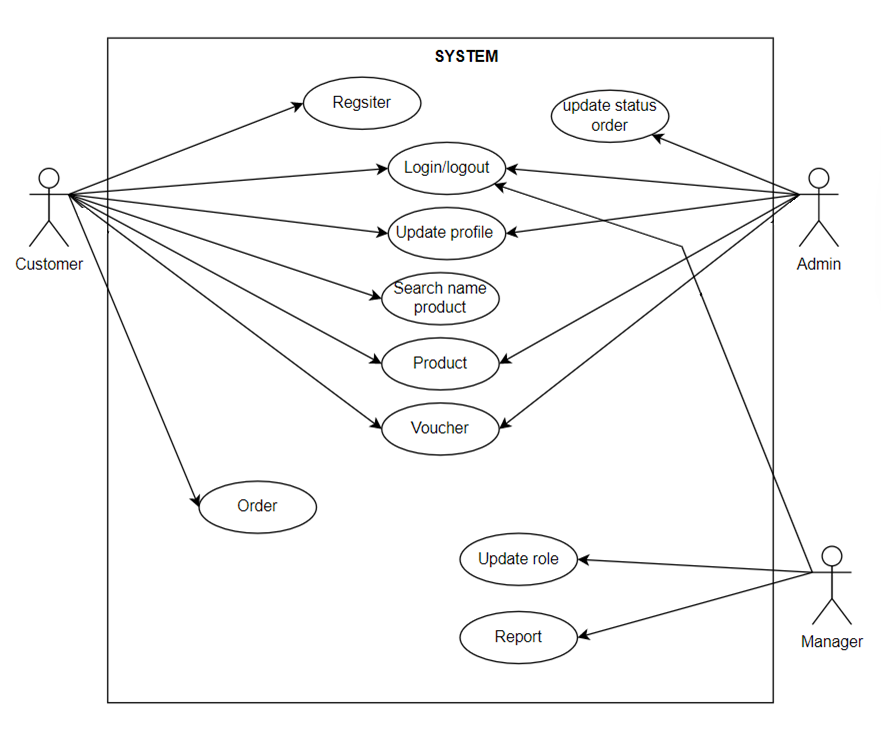
* 1. Biểu đồ phân cấp chức năng



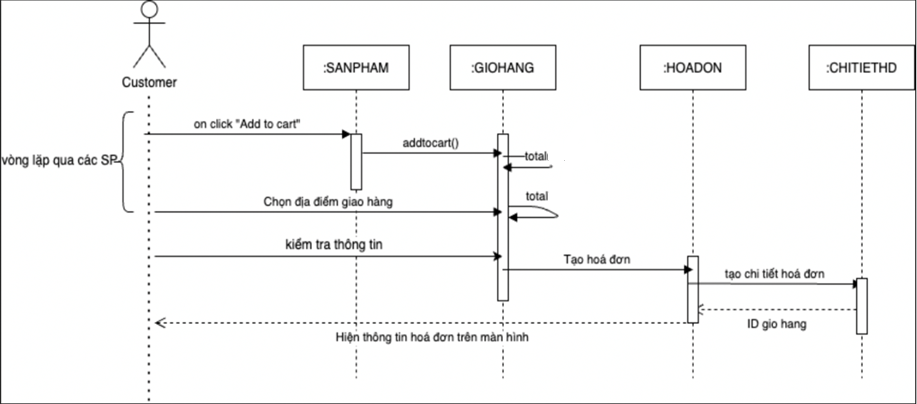
* 1. Biểu đồ luồng dữ liệu



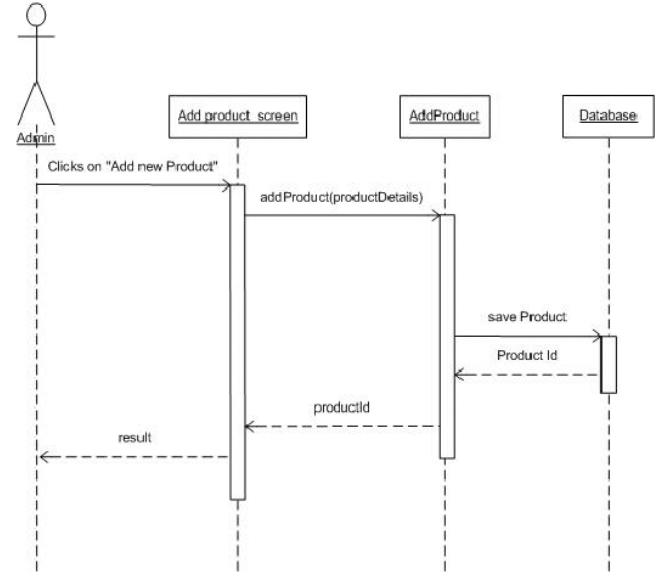
* 1. Biểu đồ usecase



* 1. Biểu đồ tuần tự
* Biểu đồ tuần tự mua hàng.



* Biểu đồ cập nhật thông tin sản phẩm.



* 1. Phân tích và thiết kế dữ liệu
  2. Phân tích dữ liệu

+ Đăng ký: Mở form cho người dùng sử dụng đăng ký Tài khoản, điến các thông tin cần thiết của bản thân như username, số điện thoại, email, mật khẩu, …

+ Đăng nhập: Xác thực vào hệ thống với tư cách một tài khoản bằng email và mật khẩu.

+ Cập nhật tài khoản: cho phép người quản trị viên sửa và xóa thông tin một tài khoản ( trừ tài khoản của quản lý). Người quản lý có quyền gán vai trò cho một tài khoản. Người dùng được cập cập nhật lại một số thông tin như địa chỉ, số điện thoại,….

+ Phân quyền người dùng: cấp các quyền cho một vai trò.

* *Quản lý sản phẩm:*

+ Cập nhật : thêm, sửa, xóa nhà xuất bản , loại sách , tên tác giả …

+ Cập nhật sản phẩm: thêm, sửa, xóa một sản phẩm với các thông tin liên quan như: tên, mô tả, hình ảnh, giá bán,giá nhập, ngày xuất bản,ngôn ngữ …

+ Lập phiếu nhập kho: khi nhập hàng, nhân viên sẽ lập phiếu nhập kho với các thông tin như ngày lập, người lập, tổng tiền, …

+ Cập nhật chi tiết phiếu nhập kho: đi kèm với phiếu nhập kho và mỗi sản phẩm là chi tiết phiếu nhập kho bao gồm thông tin như mã phiếu nhập kho, mã sản phẩm, số lượng, đơn giá nhập, …

+ Cập nhật chi tiết sản phẩm nhập: đi kèm với một chi tiết phiếu nhập kho với số lượng tương ứng cho mỗi sản phẩm vật lý một bản ghi để lưu lại của nó.

* *Giao dịch khách hàng*:

+ Tìm kiếm sản phẩm: hỗ trợ người sử dụng tìm kiếm sản phẩm theo nhà xuất bản và mã isbn , tên tác giả , tên sách

+ Chọn giỏ hàng: hỗ trợ một tài khoản lựa chọn một số sản phẩm muốn mua với số lượng muốn

+ Đặt mua hàng: sau khi đã chọn giỏ hàng, khách hàng có thế nhấn nút “Đặt mua hàng” và chọn thông tin nơi nhận hàng.

+Theo dõi trạng thái đơn hàng: khách hàng có thể theo dõi quá trình xử lý đơn mua hàng của mình thông qua các trạng thái: chờ xác nhận - đã thanh toán hoặc chưa , đã xử lý, đang chờ giao hàng, đã giao hàng.

* *Quản lý bán hàng:*

+ Duyệt đơn đặt hàng: xem xét các đơn đặt về tính hợp lệ, mặt hàng, số lượng,địa điểm nhận hàng để duyệt đơn hàng.

+ Lập hóa đơn bán:

+ In hóa hơn bán: Xuất hóa đơn bán cho một đơn hàng.

+ Phản hồi của khách hàng: trả lời phản hồi của khách hàng sau khi duyệt đơn.

* *Báo cáo thống kê:*

+ Báo cáo tình trạng sản phẩm: xuất báo cáo về tình trạng còn hàng, hết hàng, số lượng còn lại bao nhiêu để làm cơ sở cho xử lý đơn hàng và lập phiếu nhập hàng.

+ Báo cáo doanh thu định kỳ: đưa ra báo cáo đoanh thu theo nhóm hàng định kỳ từ ngày bắt đầu đến ngày kết thúc.

+ Lập kế hoạch nhập hàng: Đưa ra báo cáo đề xuất nhập hàng để các cấp có thẩm quyền duyệt kế hoạch nhập hàng.

* 1. Các thực thể, mối quan hệ giữa các thực thể và quan hệ dữ liệu
* ***Author***: tác giả

+ Quan hệ n-n với bảng *Item*.

*CREATE TABLE IF NOT EXISTS `book\_store`.`author` (*

*`author\_id` INT NOT NULL AUTO\_INCREMENT,*

*`author\_name` VARCHAR(45) NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`status` INT NOT NULL,*

*PRIMARY KEY (`author\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Author\_id | Mã tác giả | T | int |
| Author\_name | Tên của tác giả | F | nvarchar(45) |

* ***Products\_has\_author***: bảng trung gian giữa bảng author và product

+ Quan hệ n-1 với bảng *author*

*+* Quan hệ n-1 với bảng *products*

*CREATE TABLE IF NOT EXISTS `book\_store`.`products\_has\_author` (*

*`product\_id` INT NOT NULL,*

*`author\_id` INT NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*PRIMARY KEY (`product\_id`, `author\_id`),*

*INDEX `fk\_products\_has\_author\_author1\_idx` (`author\_id` ASC) VISIBLE,*

*INDEX `fk\_products\_has\_author\_products1\_idx` (`product\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_products\_has\_author\_author1`*

*FOREIGN KEY (`author\_id`)*

*REFERENCES `book\_store`.`author` (`author\_id`),*

*CONSTRAINT `fk\_products\_has\_author\_products1`*

*FOREIGN KEY (`product\_id`)*

*REFERENCES `book\_store`.`products` (`product\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Product\_id | Mã sản phẩm | FK | int |
| Author\_id | Mã tác giả | FK | int |
| Status | Trạng thái | F | int |

* ***Products***: sản phẩm

+ Quan hệ 1-n với bảng *Feedback*

+ Quan hệ n-1 với bảng *Publish* thông qua khóa ngoại - *publish\_id*

+ Quan hệ n-1 với bảng *Genre* thông qua khóa ngoại - *genre\_id*

+ Quan hệ n-n với bảng *Author* thông qua khóa bảng trung gian – *products\_has\_author*

+ Quan hệ 1-n với bảng *order\_detail*

+ Quan hệ 1-n với bảng *order\_input\_detail*

*CREATE TABLE IF NOT EXISTS `book\_store`.`products` (*

*`product\_id` INT NOT NULL AUTO\_INCREMENT,*

*`isbn` VARCHAR(45) NOT NULL,*

*`product\_name` VARCHAR(150) NOT NULL,*

*`buy\_turn` INT NOT NULL,*

*`quantity` INT NOT NULL,*

*`status` INT NOT NULL,*

*`description` VARCHAR(2000) CHARACTER SET 'utf8mb3' NOT NULL,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`image` VARCHAR(200) NOT NULL,*

*`price\_display` FLOAT NOT NULL,*

*`percent\_sale` FLOAT NOT NULL,*

*`price\_output` FLOAT NOT NULL,*

*`price\_input` FLOAT NOT NULL,*

*`publish\_id` INT NOT NULL,*

*`genre\_id` INT NOT NULL,*

*`pages` INT NOT NULL,*

*`time\_public` DATE NOT NULL,*

*`language` VARCHAR(45) NOT NULL,*

*PRIMARY KEY (`product\_id`, `isbn`),*

*INDEX `fk\_products\_publish1\_idx` (`publish\_id` ASC) VISIBLE,*

*INDEX `fk\_products\_genre1\_idx` (`genre\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_products\_genre1`*

*FOREIGN KEY (`genre\_id`)*

*REFERENCES `book\_store`.`genre` (`genre\_id`),*

*CONSTRAINT `fk\_products\_publish1`*

*FOREIGN KEY (`publish\_id`)*

*REFERENCES `book\_store`.`publish` (`publish\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Product\_id | Mã sản phẩm | T | int |
| Isbn | Mã Isbn sản phẩm | T | Nvarchar(45) |
| Product\_name | Tên sản phẩm | F | nvarchar(45) |
| Buy\_turn | Số lượt mua | F | int |
| Status | Trạng thái sản phẩm | F | int |
| Describe | Mô tả sản phẩm | F | varchar(1000) |
| Image | Ảnh sản phẩm | F | Varchar(200) |
| Price\_input | Giá nhập | F | float |
| Price\_display | Giá hiện giảm | F | float |
| Percent\_sale | Phần trăm giảm giá | F | float |
| Price\_output | Giá xuất | F | float |
| Publish\_id | Mã nhà xuất bản | Fk | int |
| Genre\_id | Mã thể loại sách | Fk | iny |
| quantity | Số lượng sản phẩm | F | int |
| pages | Số trang của sách | F | int |
| Time\_public | Thời gian xuất bản | F | Date time |
| Language | Ngôn ngữ | F | Varchar(45) |

* ***Publish***: Nhà xuất bản

+ Quan hệ 1-n với bảng *Product*

*CREATE TABLE IF NOT EXISTS `book\_store`.`publish` (*

*`publish\_id` INT NOT NULL AUTO\_INCREMENT,*

*`publish\_name` VARCHAR(45) NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`status` INT NOT NULL,*

*PRIMARY KEY (`publish\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Publish\_id | Mã nhà xuất bản | T | int |
| Publish\_name | Tên của nhà xuất bản | F | nvarchar(255) |
| Status | Trạng thái | F | int |

* ***Genre***: Thể loại sách

+ Quan hệ 1-n với bảng product

+ Quan hệ n-1 với bảng category qua khóa ngoaki category\_id

*CREATE TABLE IF NOT EXISTS `book\_store`.`genre` (*

*`genre\_id` INT NOT NULL AUTO\_INCREMENT,*

*`genre\_name` VARCHAR(100) NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`category\_id` INT NOT NULL,*

*`status` INT NOT NULL,*

*PRIMARY KEY (`genre\_id`),*

*INDEX `fk\_genre\_category1\_idx` (`category\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_genre\_category1`*

*FOREIGN KEY (`category\_id`)*

*REFERENCES `book\_store`.`category` (`id\_category`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Genre\_id | Mã thể loại sách | T | int |
| Genre\_name | Tên của thể loại sách | F | nvarchar(255) |
| Category\_id | Mã loại sách | Fk | int |
| Status | Trạng thái | F | int |

* ***Category***: Thể loại sách

+ Quan hệ 1-n với bảng product

+ Quan hệ 1-n với bảng genre

*CREATE TABLE IF NOT EXISTS `book\_store`.`category` (*

*`id\_category` INT NOT NULL AUTO\_INCREMENT,*

*`name\_category` VARCHAR(100) NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`update\_at` TIMESTAMP NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`status` INT NOT NULL,*

*PRIMARY KEY (`id\_category`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| category\_id | Mã loại sách | T | int |
| category\_name | Tên của loại sách | F | nvarchar(255) |
| Status | Trạng thái | F | int |

* ***order\_input\_detail***: chi tiết hóa đơn nhập

+ Quan hệ n-1 với bảng *product* thông qua khóa ngoại *product\_id*

+ Quan hệ n-1 với bảng *order\_input* thông qua khóa ngoại *order\_input\_id*

*CREATE TABLE IF NOT EXISTS `book\_store`.`order\_input\_detail` (*

*`quantity` INT NOT NULL,*

*`create\_at` TIMESTAMP NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`update\_at` TIMESTAMP NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`order\_input\_id` INT NOT NULL,*

*`product\_id` INT NOT NULL,*

*INDEX `fk\_order\_input\_detail\_order\_input1\_idx` (`order\_input\_id` ASC) VISIBLE,*

*INDEX `fk\_order\_input\_detail\_products1\_idx` (`product\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_order\_input\_detail\_order\_input1`*

*FOREIGN KEY (`order\_input\_id`)*

*REFERENCES `book\_store`.`order\_input` (`id\_order\_input`),*

*CONSTRAINT `fk\_order\_input\_detail\_products1`*

*FOREIGN KEY (`product\_id`)*

*REFERENCES `book\_store`.`products` (`product\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Quantity | Số lượng sản phẩm | F | int |
| Order\_input\_id | Mã hóa đơn nhập | Fk | int |
| Product\_id | Mã sản phẩm | Fk | int |

* ***Order\_input***: hóa đơn nhập

+ Quan hệ 1-n với bảng order\_input\_detail

*CREATE TABLE IF NOT EXISTS `book\_store`.`order\_input` (*

*`id\_order\_input` INT NOT NULL AUTO\_INCREMENT,*

*`total` INT NOT NULL,*

*`note` VARCHAR(500) NULL DEFAULT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*PRIMARY KEY (`id\_order\_input`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Id\_order\_input | Mã hóa đơn nhập | T | int |
| Total | Tổng tiền | F | int |
| Note | Ghi chú | F | Varchar(500) |

* ***Shopping\_cart***: giỏ hàng

+ Quan hệ 1-1 với bảng *account* thông qua khóa ngoại *account\_id*

+ Quan hệ n-n với bảng *products* thông qua bảng trung gian *shopping\_cart\_has\_product*

*CREATE TABLE IF NOT EXISTS `book\_store`.`shopping\_cart` (*

*`Shopping\_cart\_id` INT NOT NULL AUTO\_INCREMENT,*

*`account\_id` INT NOT NULL,*

*PRIMARY KEY (`Shopping\_cart\_id`),*

*INDEX `fk\_shopping\_cart\_account1\_idx` (`account\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_shopping\_cart\_account1`*

*FOREIGN KEY (`account\_id`)*

*REFERENCES `book\_store`.`account` (`account\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Shopping\_cart\_id | Mã giỏ hàng | T | int |
| Account\_id | Mã tài khoản | Fk | int |

* ***Shopping\_cart\_has\_product***: Bảng trung gian giữa bảng *Products* vs bảng *Shopping\_cart*

+ Quan hệ n-1 với bảng *Shopping\_cart* thông qua khóa ngoại *shopping\_cart\_id*

+ Quan hệ n-1 với bảng *Products* thông qua khóa ngoại *product\_id*

*CREATE TABLE IF NOT EXISTS `book\_store`.`shopping\_cart\_has\_products` (*

*`Shopping\_cart\_id` INT NOT NULL,*

*`product\_id` INT NOT NULL,*

*`product\_quantity` INT NULL DEFAULT NULL,*

*PRIMARY KEY (`Shopping\_cart\_id`, `product\_id`),*

*INDEX `fk\_shopping\_cart\_has\_products\_products1\_idx` (`product\_id` ASC) VISIBLE,*

*INDEX `fk\_shopping\_cart\_has\_products\_shopping\_cart1\_idx` (`Shopping\_cart\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_shopping\_cart\_has\_products\_products1`*

*FOREIGN KEY (`product\_id`)*

*REFERENCES `book\_store`.`products` (`product\_id`),*

*CONSTRAINT `fk\_shopping\_cart\_has\_products\_shopping\_cart1`*

*FOREIGN KEY (`Shopping\_cart\_id`)*

*REFERENCES `book\_store`.`shopping\_cart` (`Shopping\_cart\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Shopping\_cart\_id | Mã giỏ hàng | Fk | int |
| quantity | Số lượng sản phẩm | F | int |
| Product\_id | Mã sẩn phẩm | Fk | int |

* **Account**: tài khoản

+ Quan hệ 1-n với bảng Feedback

+ Quan hệ 1-n với bảng Rep\_Feedback

+ Quan hệ 1-1 với bảng Shopping\_cart

+ Quan hệ 1-n với bảng Order

+ Quan hệ 1-n với bảng Account\_address

*CREATE TABLE IF NOT EXISTS `book\_store`.`account` (*

*`account\_id` INT NOT NULL AUTO\_INCREMENT,*

*`name` VARCHAR(45) NOT NULL,*

*`pass` VARCHAR(150) NULL DEFAULT NULL,*

*`email` VARCHAR(45) NOT NULL,*

*`phone` VARCHAR(45) NOT NULL,*

*`avatar` VARCHAR(45) NOT NULL,*

*`role` INT NOT NULL,*

*`status` INT NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`genre` VARCHAR(45) NOT NULL,*

*`verify` VARCHAR(45) NULL DEFAULT NULL,*

*PRIMARY KEY (`account\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Account\_id | Mã tài khoản | T | int |
| name | Tên người dùng | F | Nvarchar(45) |
| pass | Mật khẩu | F | Varchar(150) |
| email | Email | F | Varchar(45) |
| phone | Số điện thoại | F | Varchar(45) |
| avatar | Ảnh đại diện | F | Varchar(45) |
| role | Quyền | F | int |
| status | Trạng thái | F | int |
| Genre | Loại tài khoản | F | int |
| Verify | Xác thực | F | Varchar(45) |

* ***Account\_address***: địa chỉ tài khoản

+ Quan hệ n-1 với bảng *account* thông qua khóa ngoại *account\_id*

+ Quan hệ 1-n với bảng *order*

*CREATE TABLE IF NOT EXISTS `book\_store`.`account\_address` (*

*`account\_address\_id` INT NOT NULL AUTO\_INCREMENT,*

*`content` VARCHAR(1000) CHARACTER SET 'utf8mb3' NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`account\_id` INT NOT NULL,*

*`address` VARCHAR(200) NULL DEFAULT NULL,*

*`id\_province` INT NOT NULL,*

*`id\_district` INT NOT NULL,*

*`id\_town` INT NOT NULL,*

*`phone` VARCHAR(45) NOT NULL,*

*`name` VARCHAR(50) NOT NULL,*

*`default\_address` INT NOT NULL,*

*`status` INT NULL DEFAULT NULL,*

*PRIMARY KEY (`account\_address\_id`),*

*INDEX `fk\_account\_address\_account1\_idx` (`account\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_account\_address\_account1`*

*FOREIGN KEY (`account\_id`)*

*REFERENCES `book\_store`.`account` (`account\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Account\_address\_id | Mã tài khoản | T | int |
| content | Địa chỉ chi tiết | F | varchar(1000) |
| Account\_id | Mã tài khoản | Fk | int |
| address | Các thành phố , tỉnh huyện | F | Varchar(200) |
| Id\_province | Mã thành phố | F | int |
| Id\_district | Mã huyện , quận | F | int |
| Id\_town | Mã xã | F | int |
| phone | Số điện thoại người nhận | F | Varchar(45) |
| name | Tên người nhận | F | Varchar(50 |
| Default\_address | Địa chỉ mặc định | F | int |
| Status | Trạng thái | F | int |

* ***Order***: hóa đơn xuất

+ Quan hệ n-1với bảng *account\_address* thông qua khóa ngoại *account\_address\_id*

+ Quan hệ n-1với bảng *account* thông qua khóa ngoại *account\_id*

+ Quan hệ 1-n với bảng *order\_detail*

+ Quan hệ n-1 với bảng *discount* thông qua khóa ngoại díscount\_id

*CREATE TABLE IF NOT EXISTS `book\_store`.`order` (*

*`order\_id` INT NOT NULL AUTO\_INCREMENT,*

*`total` INT NOT NULL,*

*`status` INT NOT NULL,*

*`note` VARCHAR(1000) NULL DEFAULT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`account\_id` INT NOT NULL,*

*`account\_address\_id` INT NOT NULL,*

*`discount\_id` INT NULL DEFAULT NULL,*

*`delivery` VARCHAR(100) NOT NULL,*

*`payment` VARCHAR(45) NOT NULL,*

*PRIMARY KEY (`order\_id`),*

*INDEX `fk\_order\_account1\_idx` (`account\_id` ASC) VISIBLE,*

*INDEX `fk\_order\_account\_address1\_idx` (`account\_address\_id` ASC) VISIBLE,*

*INDEX `fk\_order\_discount1\_idx` (`discount\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_order\_account1`*

*FOREIGN KEY (`account\_id`)*

*REFERENCES `book\_store`.`account` (`account\_id`),*

*CONSTRAINT `fk\_order\_discount1`*

*FOREIGN KEY (`discount\_id`)*

*REFERENCES `book\_store`.`discount` (`discount\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Order\_id | Hóa đơn xuất | T | int |
| total | Tổng tiền hóa đơn | F | int |
| status | Trạng thái đơn hàng | F | Varchar(200) |
| note | Ghi chú | F | Varchar(1000) |
| Account\_id | Mã tài khoản | FK | int |
| Account\_address\_id | Tên người nhận | FK | int |
| Discount\_id | Địa chỉ mặc định | FK | int |
| Delivery | Hãng vận chuyển | F | Varchar(100) |
| Payment | Phương thức thanh toán | F | Varchar(100) |

* ***order\_detail***: chi tiết hóa đơn

+ Quan hệ n-1 với bảng *order* thông qua khóa ngoại *order\_id*

+ Quan hệ n-1 với bảng *products* thông qua khóa ngoại *product\_id*

*CREATE TABLE IF NOT EXISTS `book\_store`.`order\_detail` (*

*`quantity` INT NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`order\_id` INT NOT NULL,*

*`product\_id` INT NOT NULL,*

*INDEX `fk\_order\_detail\_order1\_idx` (`order\_id` ASC) VISIBLE,*

*INDEX `fk\_order\_detail\_products1\_idx` (`product\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_order\_detail\_order1`*

*FOREIGN KEY (`order\_id`)*

*REFERENCES `book\_store`.`order` (`order\_id`),*

*CONSTRAINT `fk\_order\_detail\_products1`*

*FOREIGN KEY (`product\_id`)*

*REFERENCES `book\_store`.`products` (`product\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| quantity | Số lượng sản phẩm | F | int |
| order\_id | Mã hóa đơn | FK | int |
| Product\_id | Mã sản phẩm | FK | int |

* ***Discount***: Mã giảm giá

+ Quan hệ 1-n với bảng *order*

*CREATE TABLE IF NOT EXISTS `book\_store`.`discount` (*

*`discount\_id` INT NOT NULL AUTO\_INCREMENT,*

*`discount\_name` VARCHAR(45) NOT NULL,*

*`discount\_start` DATETIME NOT NULL,*

*`discount\_end` DATETIME NOT NULL,*

*`discount\_price` INT NOT NULL,*

*`discount\_code` VARCHAR(45) NOT NULL,*

*`discount\_quantity` INT NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*PRIMARY KEY (`discount\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Discount\_id | Mã giảm giá | T | int |
| Discount\_name | Tên mã giảm giá | F | Varchar(45) |
| Discount\_start | Thời gian bắt đầu giảm giá | F | datetime |
| Discount\_end | Thời gian kết thúc giảm giá | F | datetime |
| Discount\_price | Giá trị mã giảm giá | F | int |
| Discount\_quantity | Số lượng mã giảm giá | F | int |
| Discount\_code | Code mã giảm giá | F | Varchar(45) |

* ***Feedback***: Phản hồi về sản phẩm

+ Quan hệ n-1 với bảng *account* thông qua khóa ngoại *account\_id*

+ Quan hệ n-1 với bảng *products* thông qua khóa ngoại *product\_id*

+ Quan hệ 1-n với bảng *rep\_feedback*

*CREATE TABLE IF NOT EXISTS `book\_store`.`feedback` (*

*`feedback\_id` INT NOT NULL AUTO\_INCREMENT,*

*`rate\_star` INT NOT NULL,*

*`content` VARCHAR(1000) CHARACTER SET 'utf8mb3' NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`product\_id` INT NOT NULL,*

*`account\_id` INT NOT NULL,*

*PRIMARY KEY (`feedback\_id`),*

*INDEX `fk\_feedback\_products1\_idx` (`product\_id` ASC) VISIBLE,*

*INDEX `fk\_feedback\_account1\_idx` (`account\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_feedback\_account1`*

*FOREIGN KEY (`account\_id`)*

*REFERENCES `book\_store`.`account` (`account\_id`),*

*CONSTRAINT `fk\_feedback\_products1`*

*FOREIGN KEY (`product\_id`)*

*REFERENCES `book\_store`.`products` (`product\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| feedback\_id | Mã phản hồi | T | int |
| Rate\_star | Số sao đánh giá | F | int |
| content | Nội dung đánh giá | F | Varchar(1000) |
| Product\_id | Mã sản phẩm | FK | int |
| Account\_id | Mã tài khoản | FK | int |

* ***Reply\_Feedback***: Phản hồi về sản phẩm

+ Quan hệ n-1 với bảng *account* thông qua khóa ngoại *account\_id*

+ Quan hệ n-1 với bảng *feedback* thông qua khóa ngoại *feedback\_id*

*CREATE TABLE IF NOT EXISTS `book\_store`.`rep\_feedback` (*

*`rep\_feedback\_id` INT NOT NULL AUTO\_INCREMENT,*

*`content` VARCHAR(1000) CHARACTER SET 'utf8mb3' NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*`account\_id` INT NOT NULL,*

*`feedback\_id` INT NOT NULL,*

*PRIMARY KEY (`rep\_feedback\_id`),*

*INDEX `fk\_rep\_feedback\_account1\_idx` (`account\_id` ASC) VISIBLE,*

*INDEX `fk\_rep\_feedback\_feedback1\_idx` (`feedback\_id` ASC) VISIBLE,*

*CONSTRAINT `fk\_rep\_feedback\_account1`*

*FOREIGN KEY (`account\_id`)*

*REFERENCES `book\_store`.`account` (`account\_id`),*

*CONSTRAINT `fk\_rep\_feedback\_feedback1`*

*FOREIGN KEY (`feedback\_id`)*

*REFERENCES `book\_store`.`feedback` (`feedback\_id`))*

*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| Rep\_feedback\_id | Mã trả lời phản hồi | T | int |
| content | Nội dung trả lời | F | Varchar(1000) |
| feedback\_id | Mã phản hồi | FK | int |
| Account\_id | Mã tài khoản | FK | int |

* ***News***: Tin tức

*CREATE TABLE IF NOT EXISTS `book\_store`.`news` (*

*`news\_id` INT NOT NULL AUTO\_INCREMENT,*

*`news\_name` VARCHAR(100) NOT NULL,*

*`content` VARCHAR(2000) NOT NULL,*

*`image` VARCHAR(100) NOT NULL,*

*`create\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,*

*`create\_by` VARCHAR(45) NOT NULL,*

*`update\_at` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,*

*`update\_by` VARCHAR(45) NULL DEFAULT NULL,*

*PRIMARY KEY (`news\_id`))*

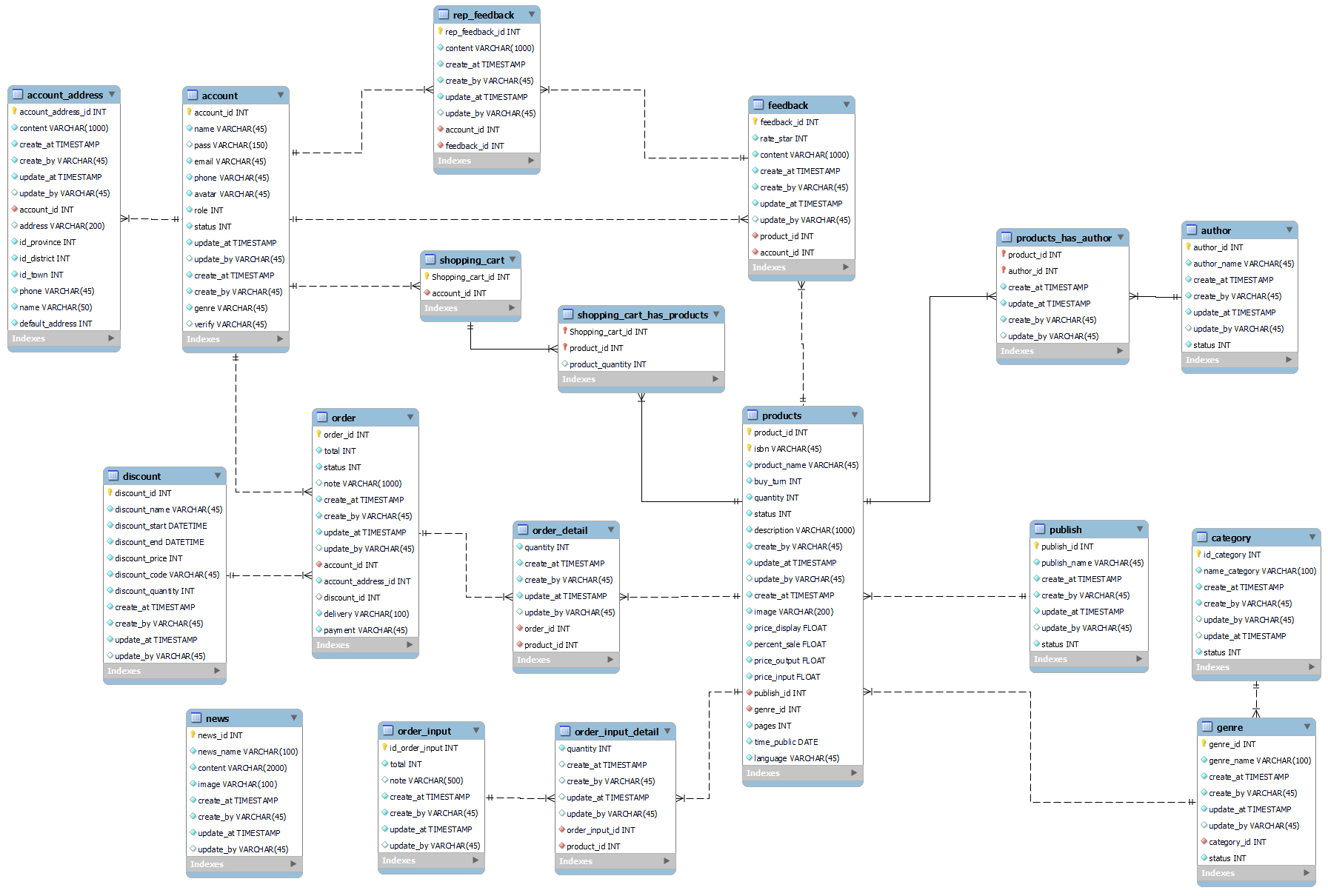
*ENGINE = InnoDB*

*DEFAULT CHARACTER SET = utf8mb4*

*COLLATE = utf8mb4\_0900\_ai\_ci*

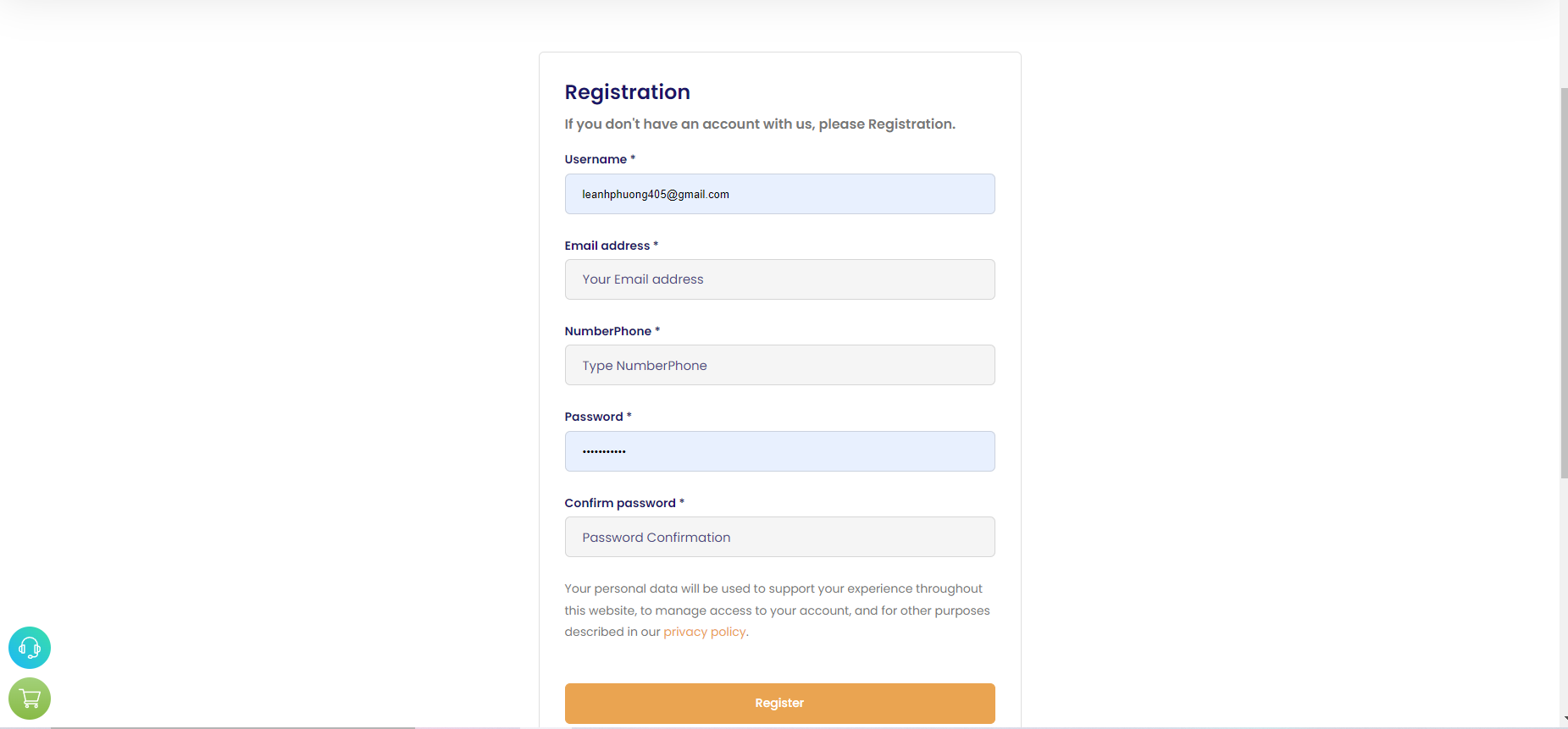
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thực thể** | **Mô tả** | **Unique keys** | **Kiểu dữ liệu – Độ dài** |
| News\_id | Mã tin tức | T | int |
| content | Nội dung tin tức | F | Varchar(2000) |
| News\_name | Tên tin tức | F | Varchar(100) |
| Image | Ảnh | F | Varchar(100) |

* 1. Biểu đồ quan hệ

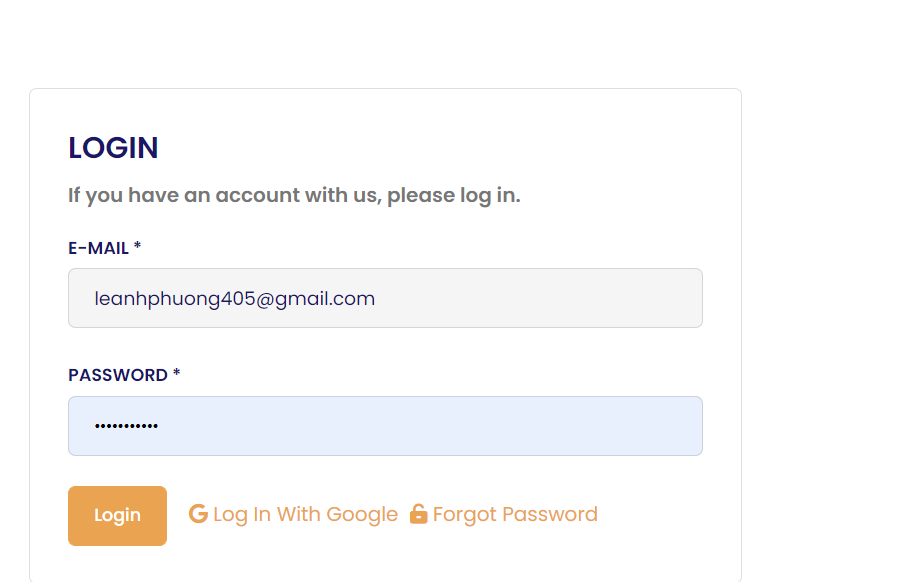


1. Xây dựng và cài đặt chương trình
2. Xây dựng trang quản trị cho người dùng.

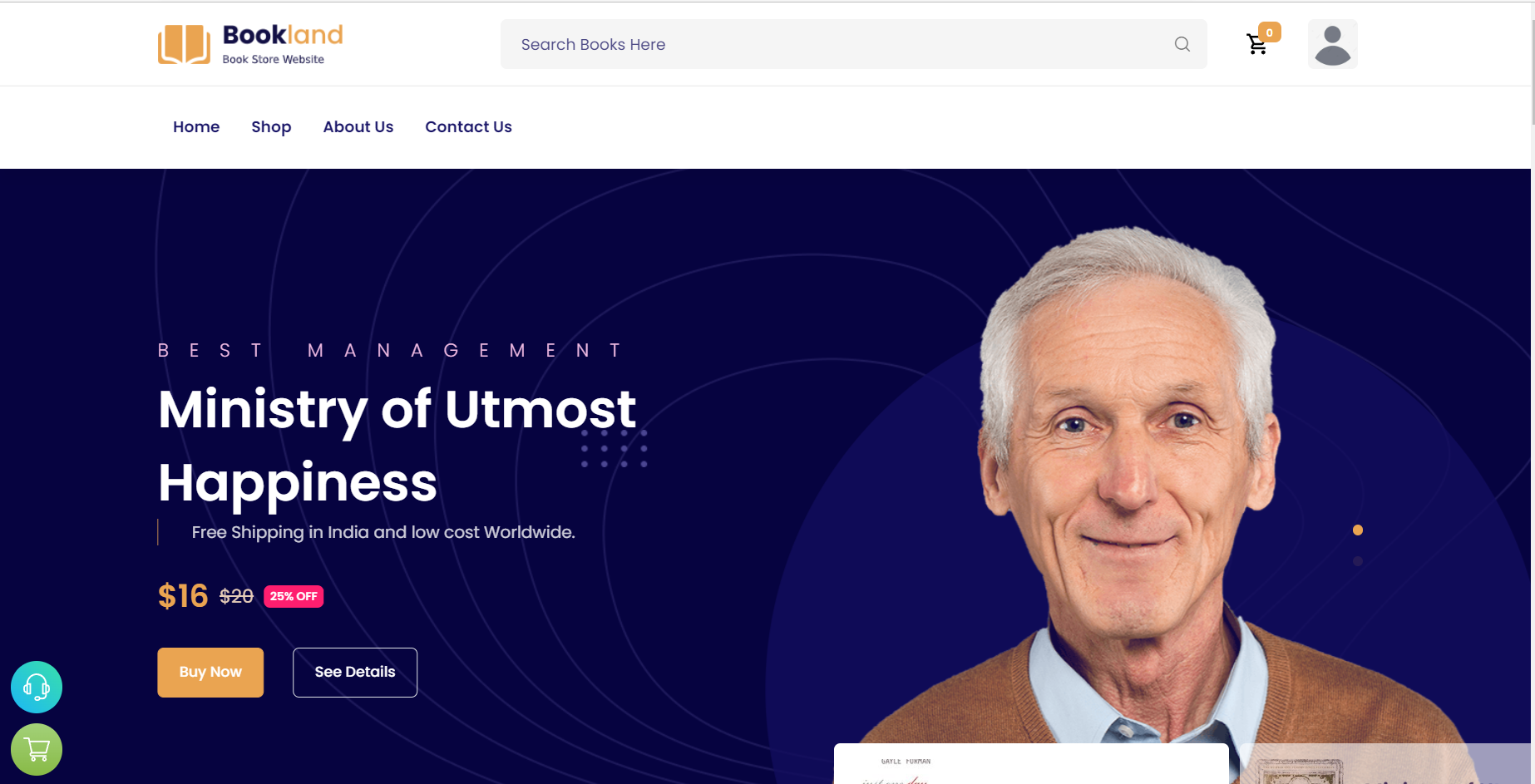
* Trang đăng ký:



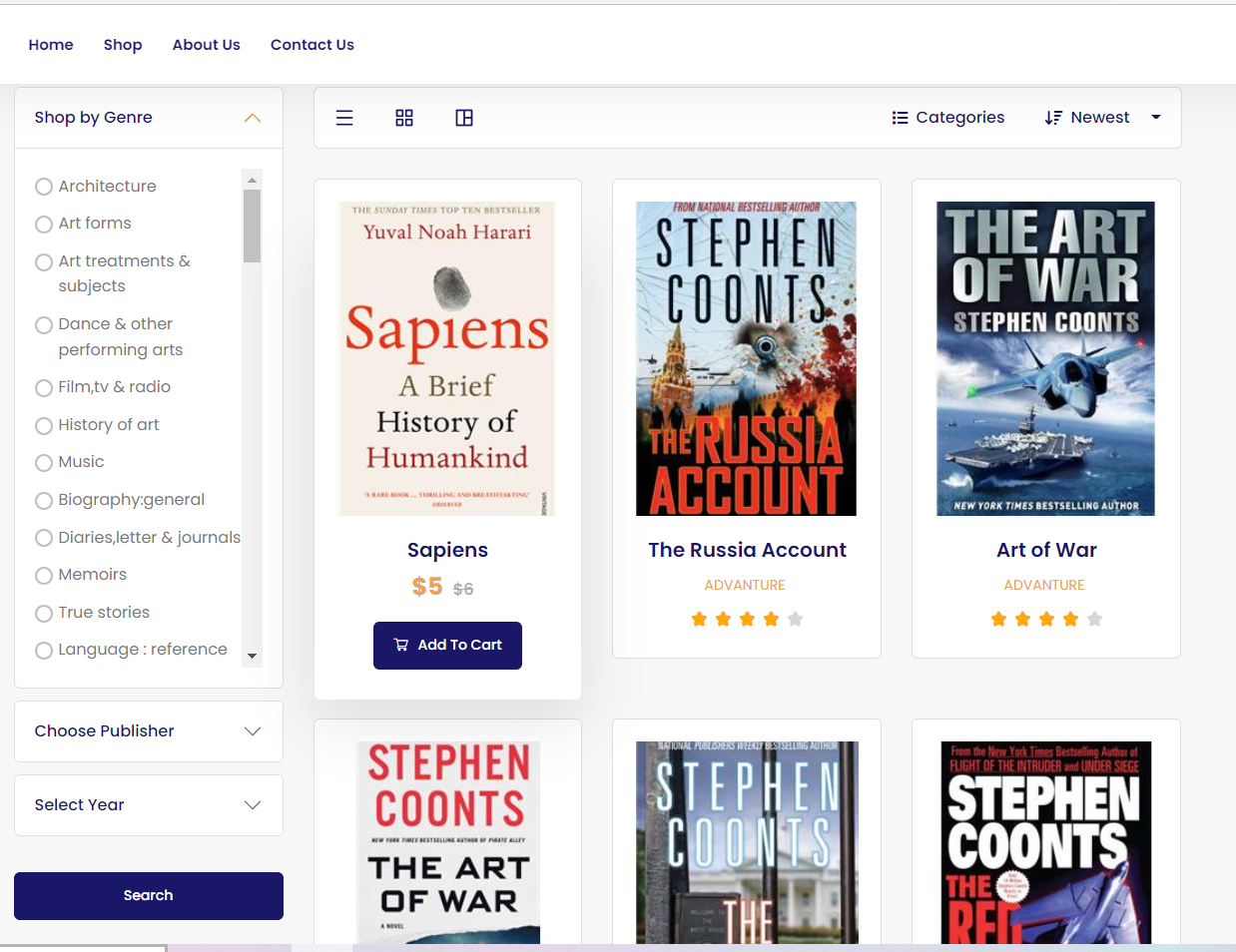
* Trang đăng nhập:



* Trang chủ:



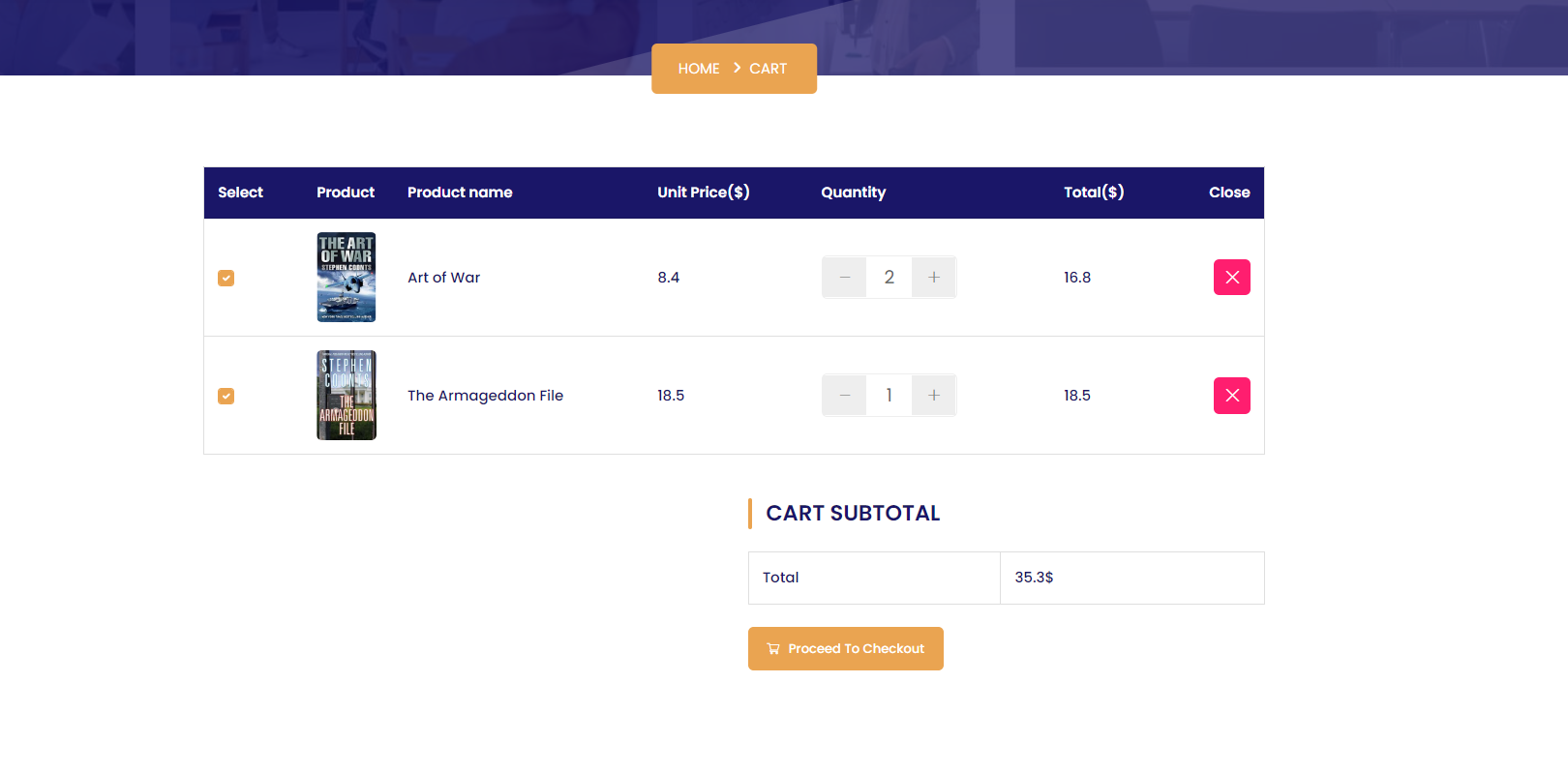
* Trang sản phẩm:



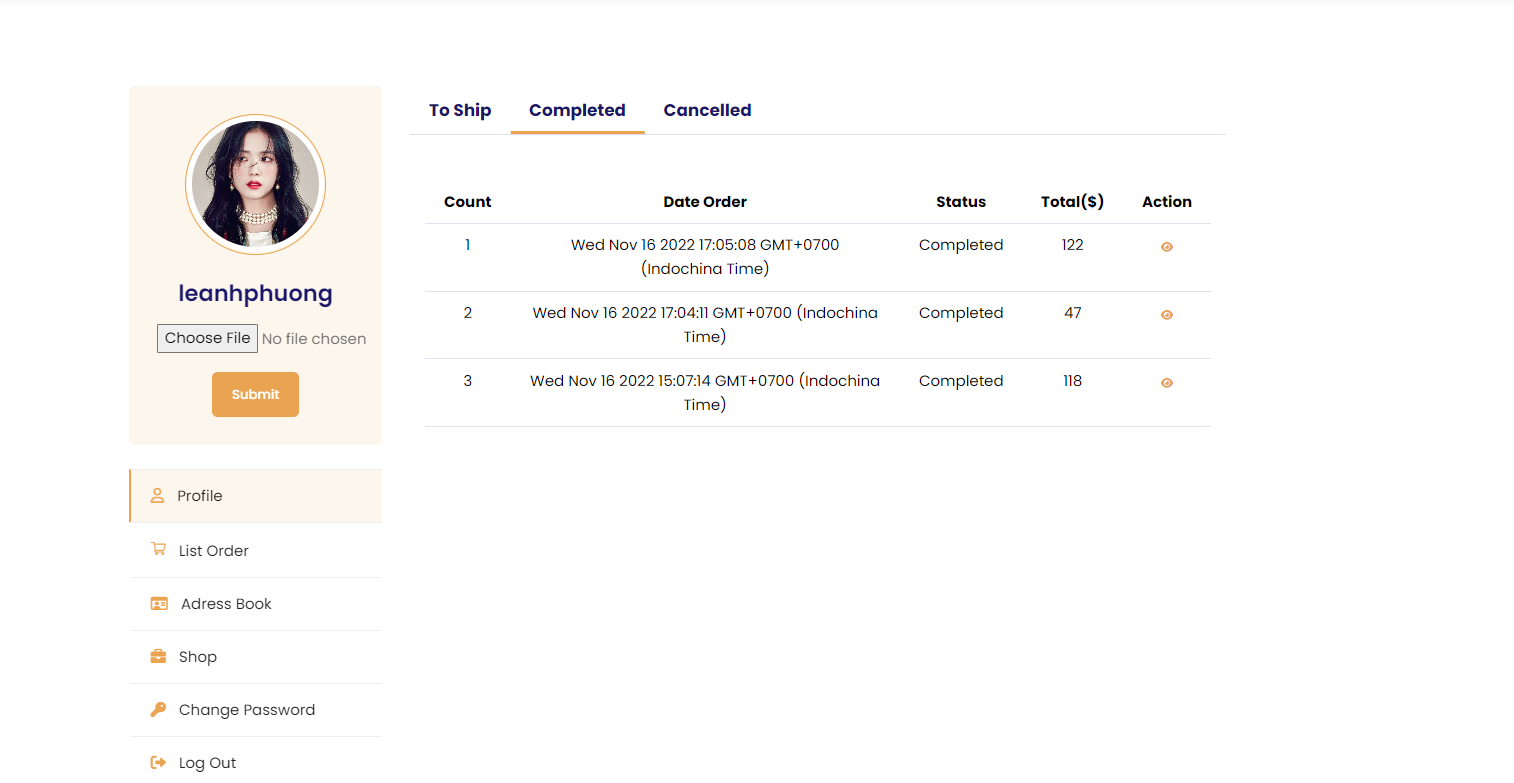
* Chi tiết sản phẩm:



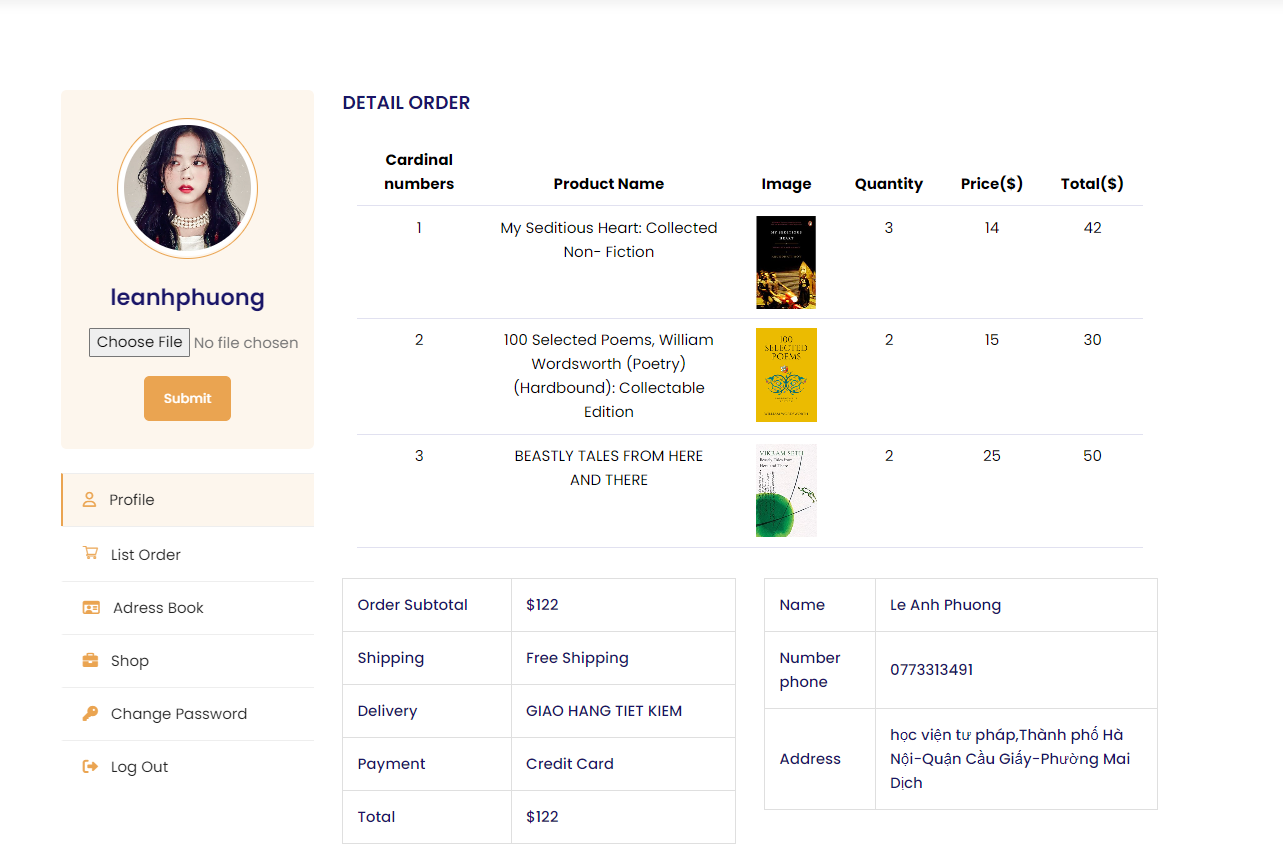
* Giỏ hàng:



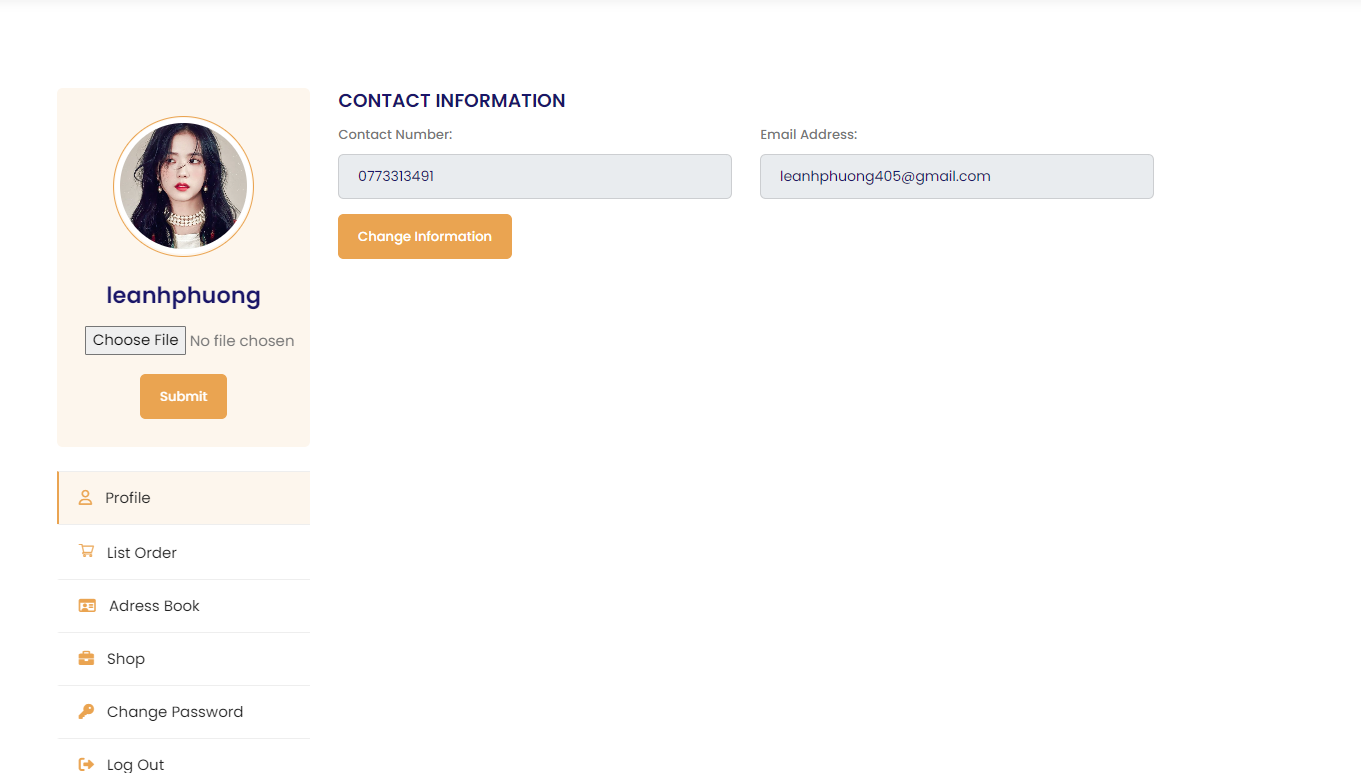
* Quản lí hoá đơn:



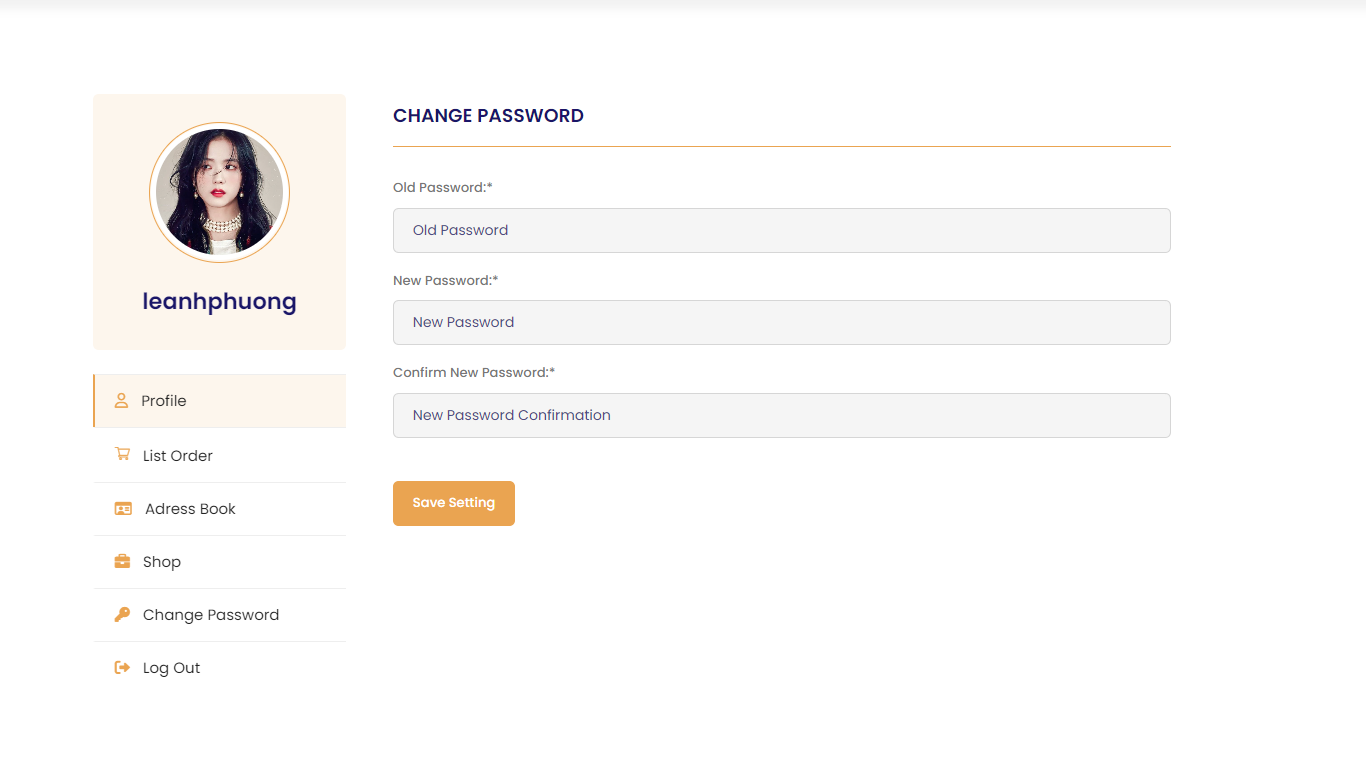
* Chi tiết hóa đơn:



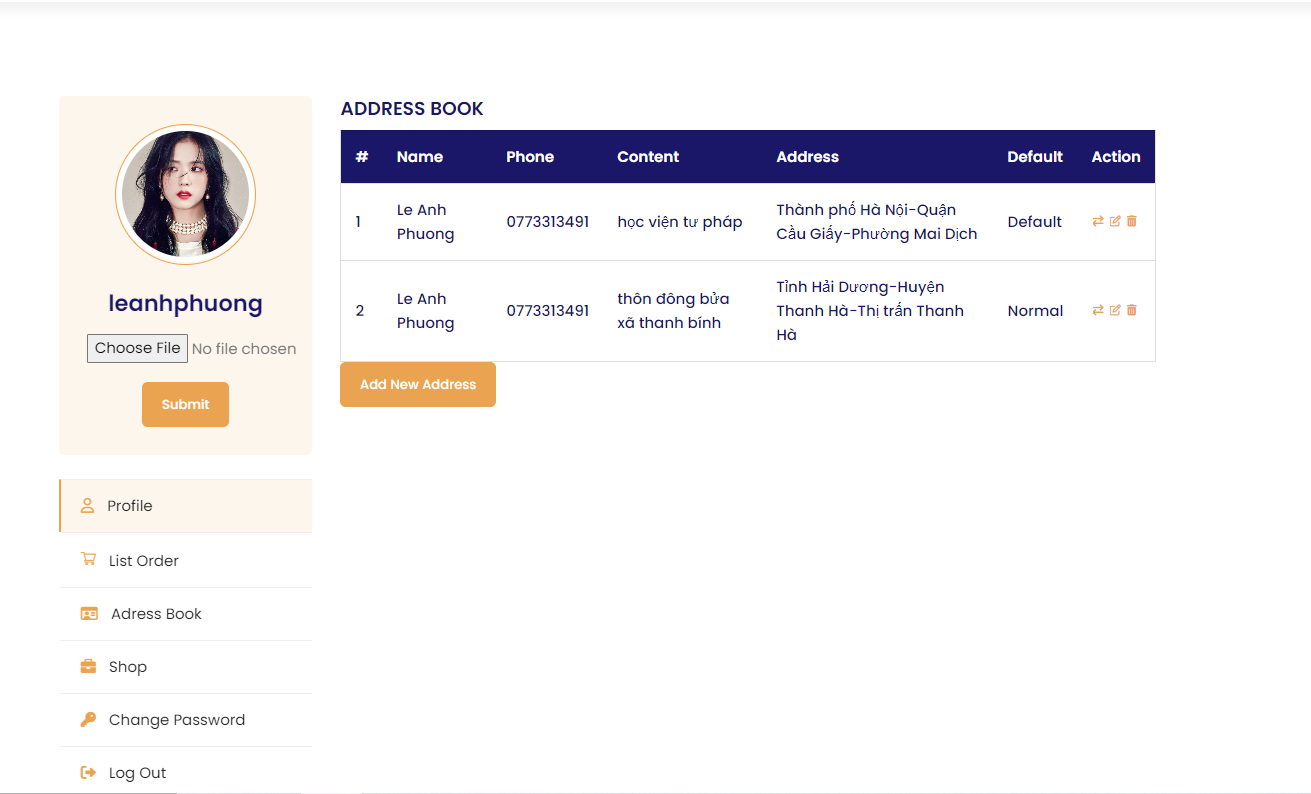
* Quản lí thông tin:



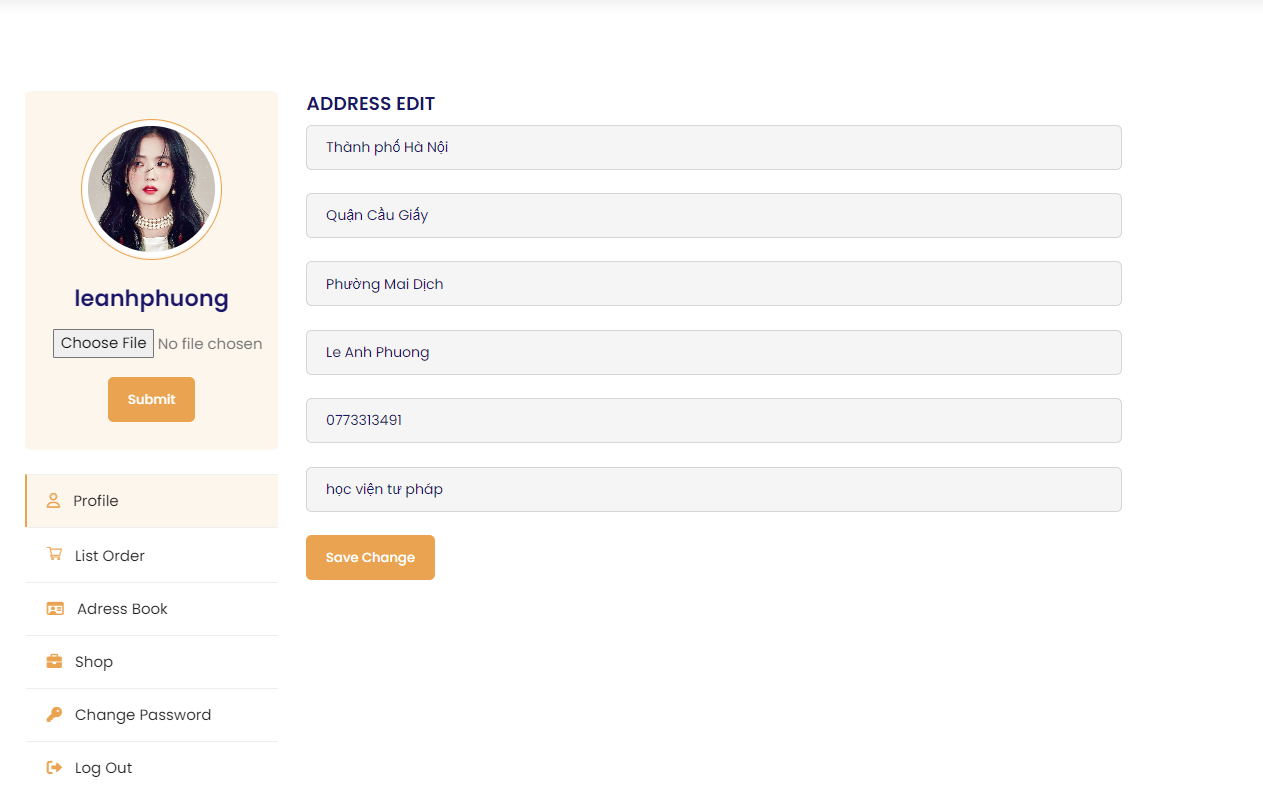
* Đổi mật khẩu



* Sổ địa chỉ



* Form sửa địa chỉ

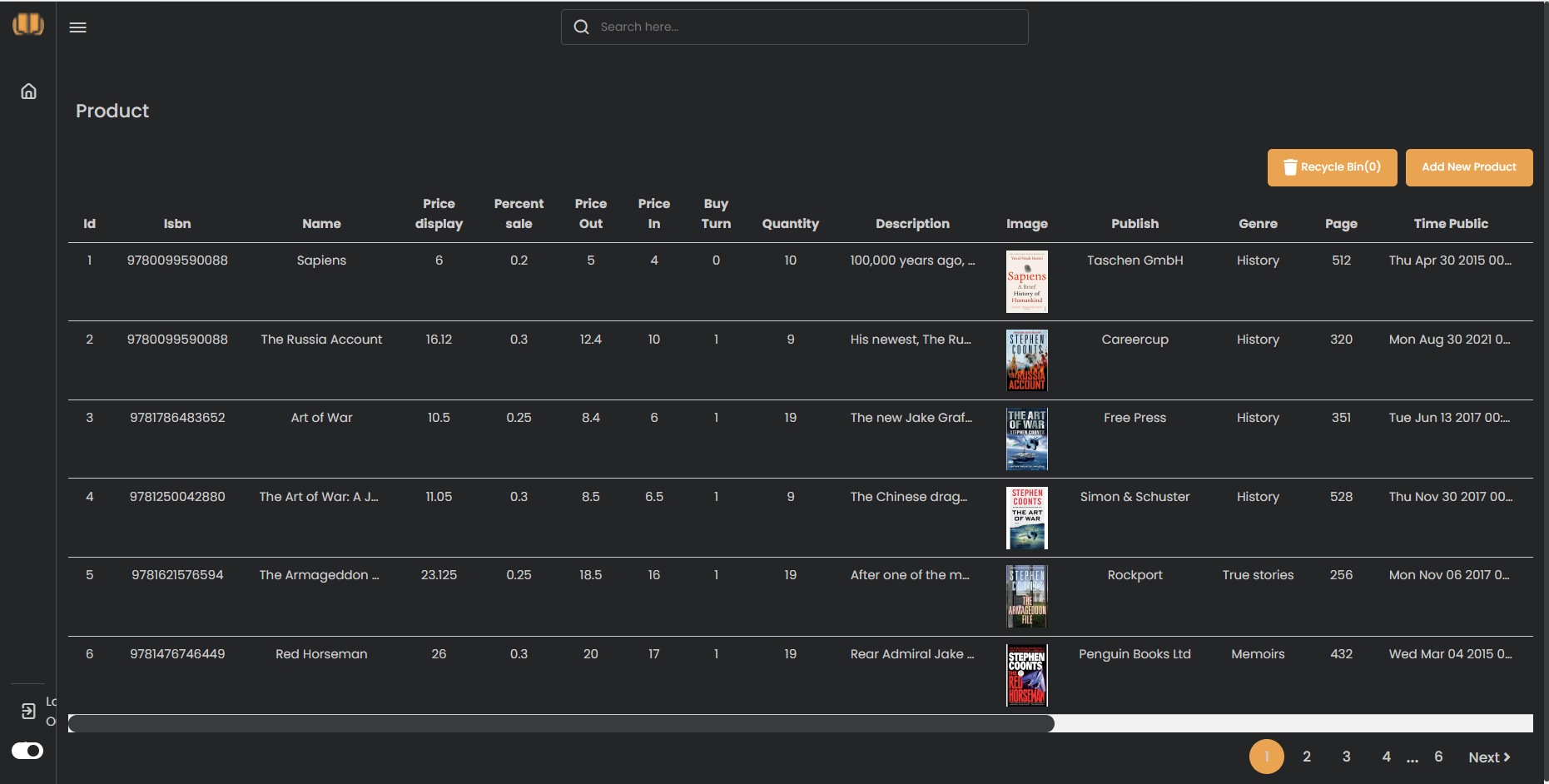


* Form thanh toán

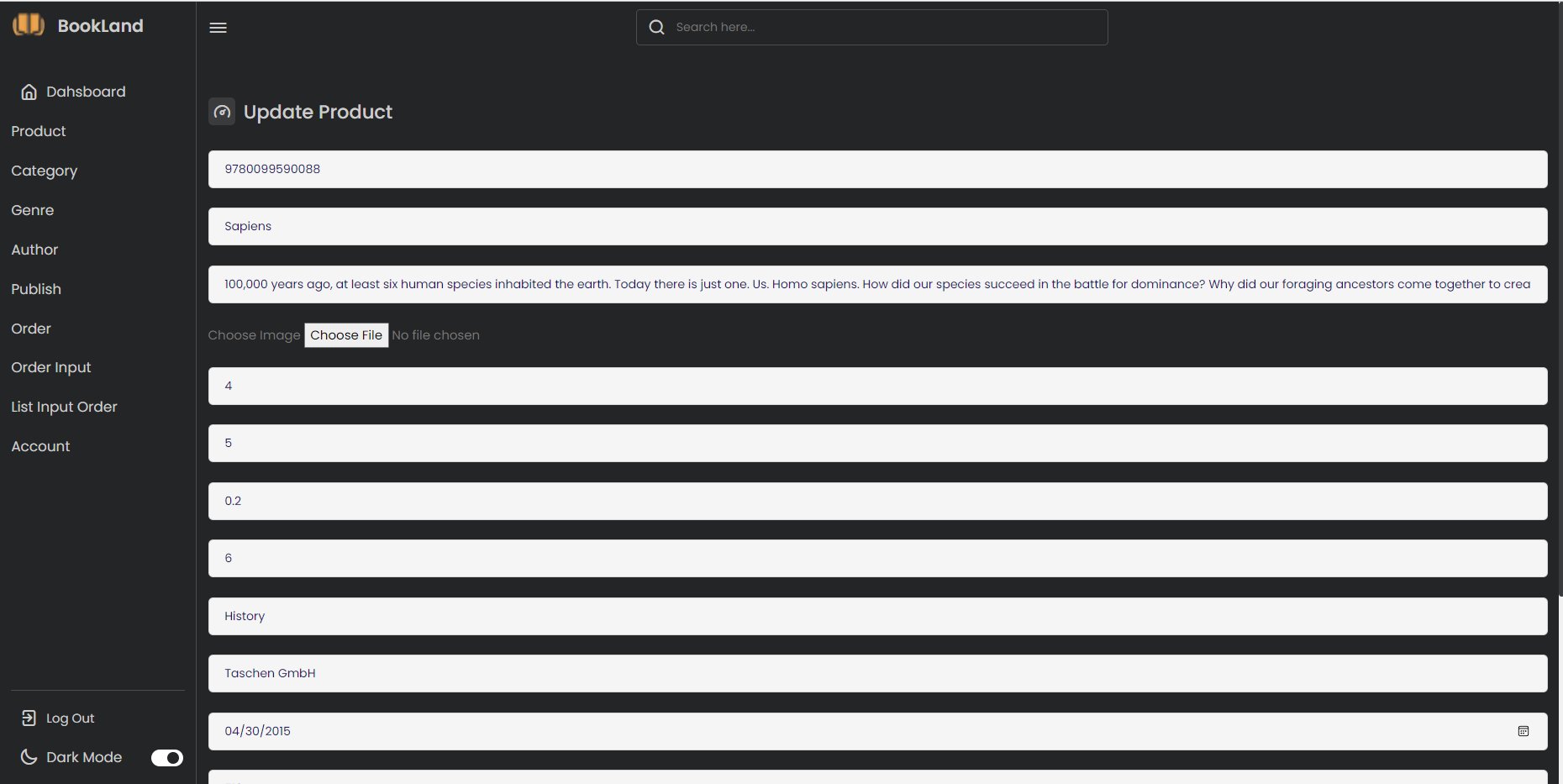


1. Xây dựng trang quản trị cho quản lí.

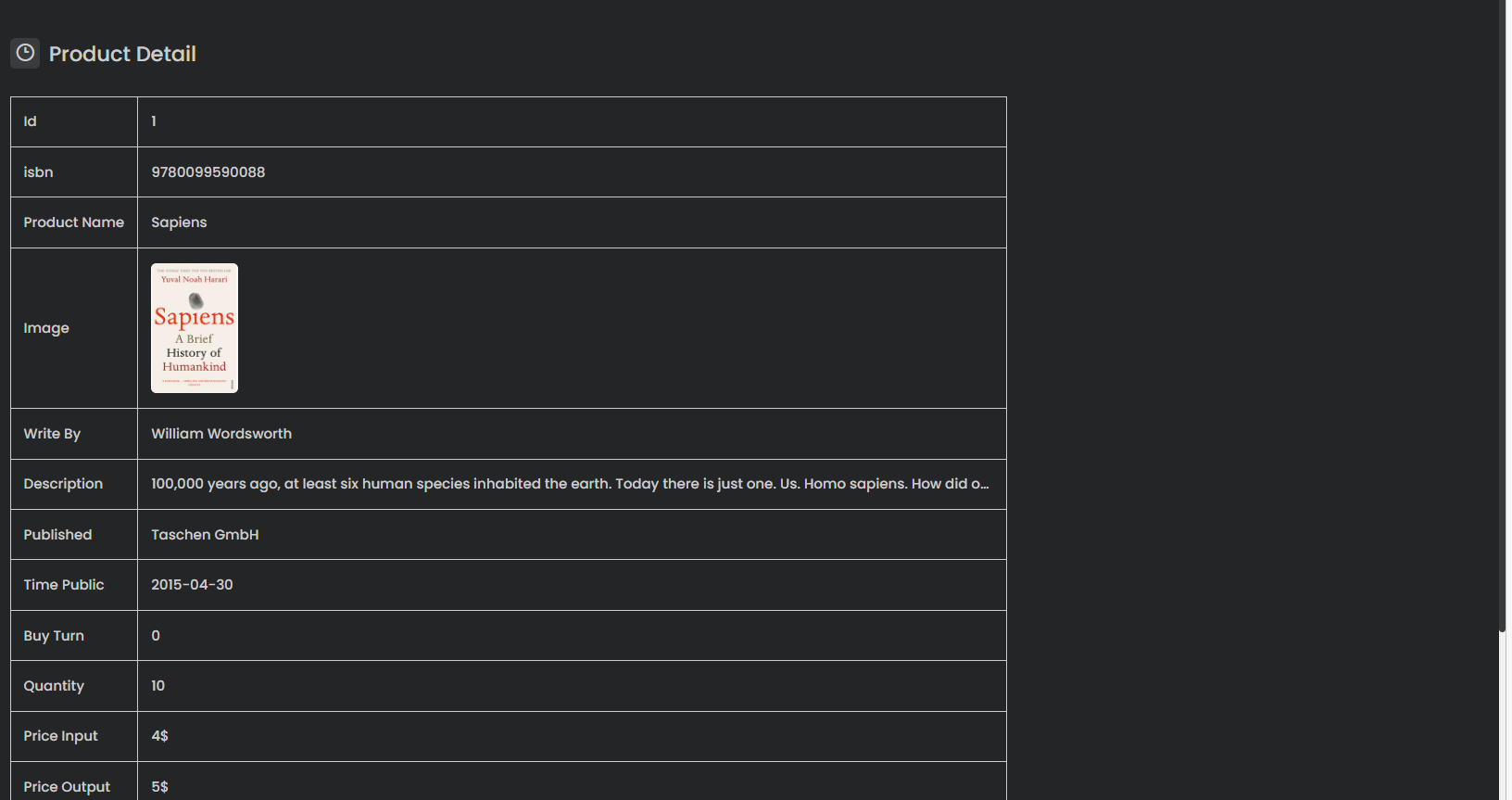
* Trang quản lí thông tin sản phẩm, danh mục sản phẩm, tác giả và nhà xuất bản ,..



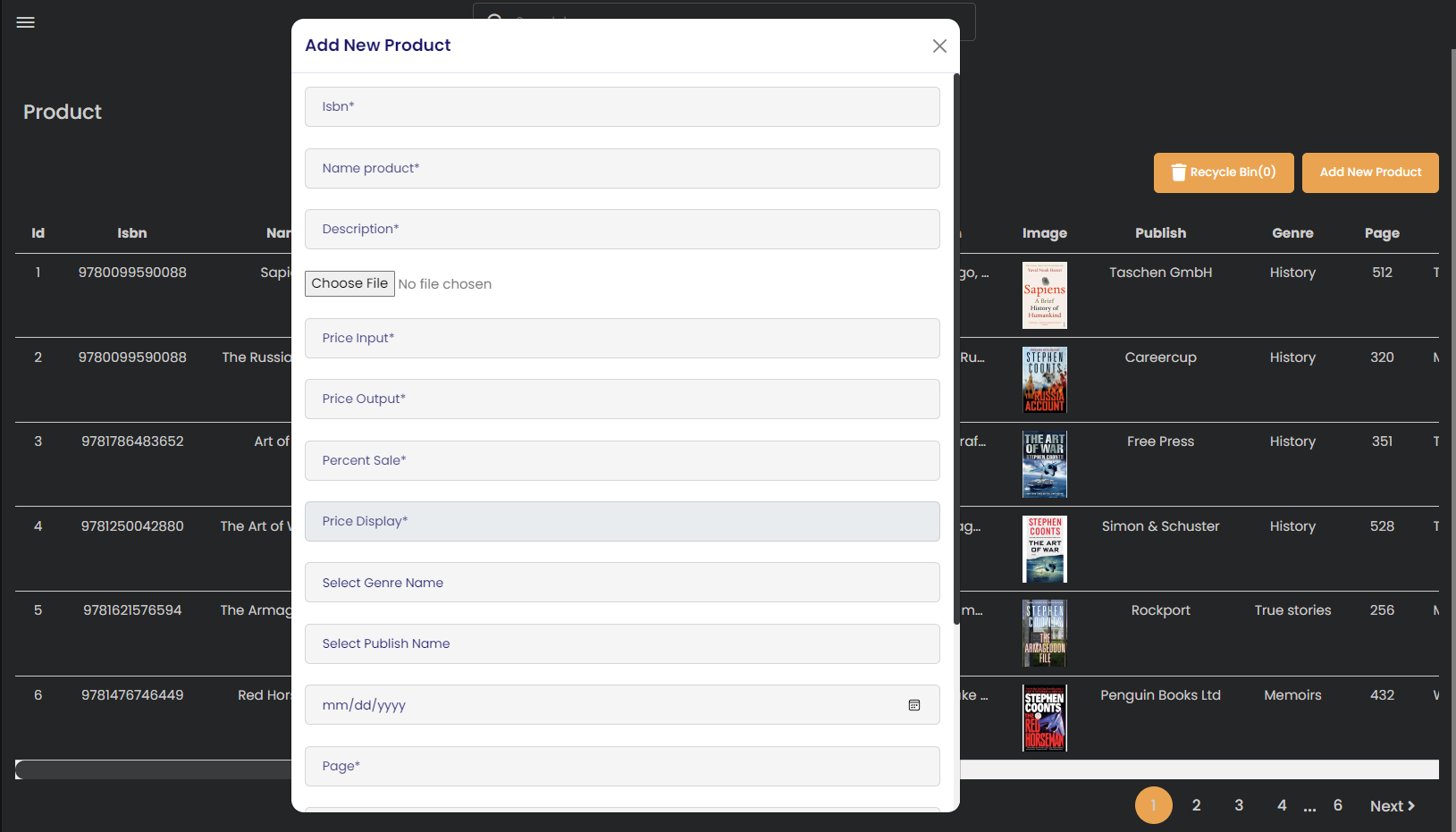
* Sửa sản phẩm



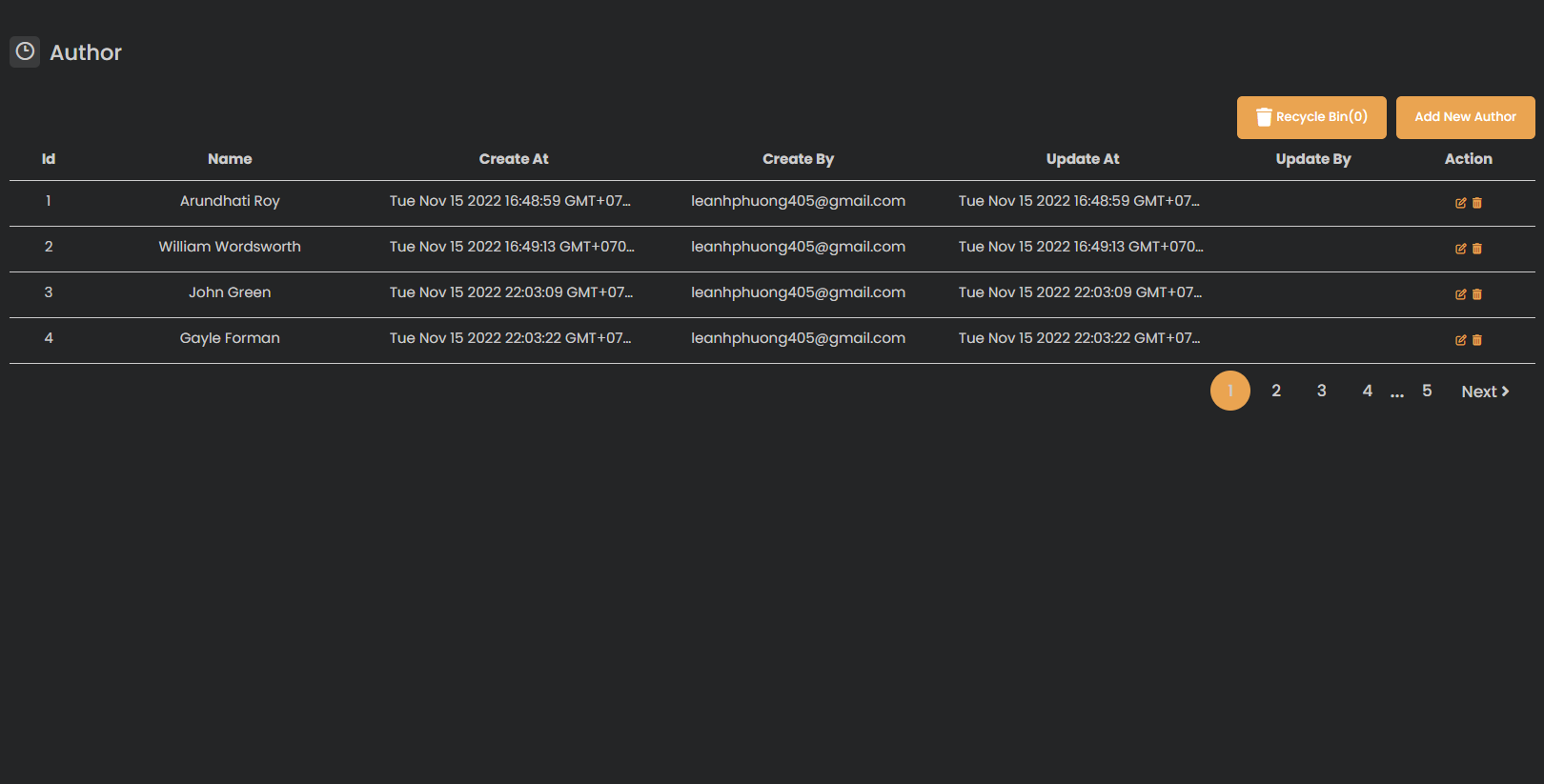
* Chi tiết sản phẩm



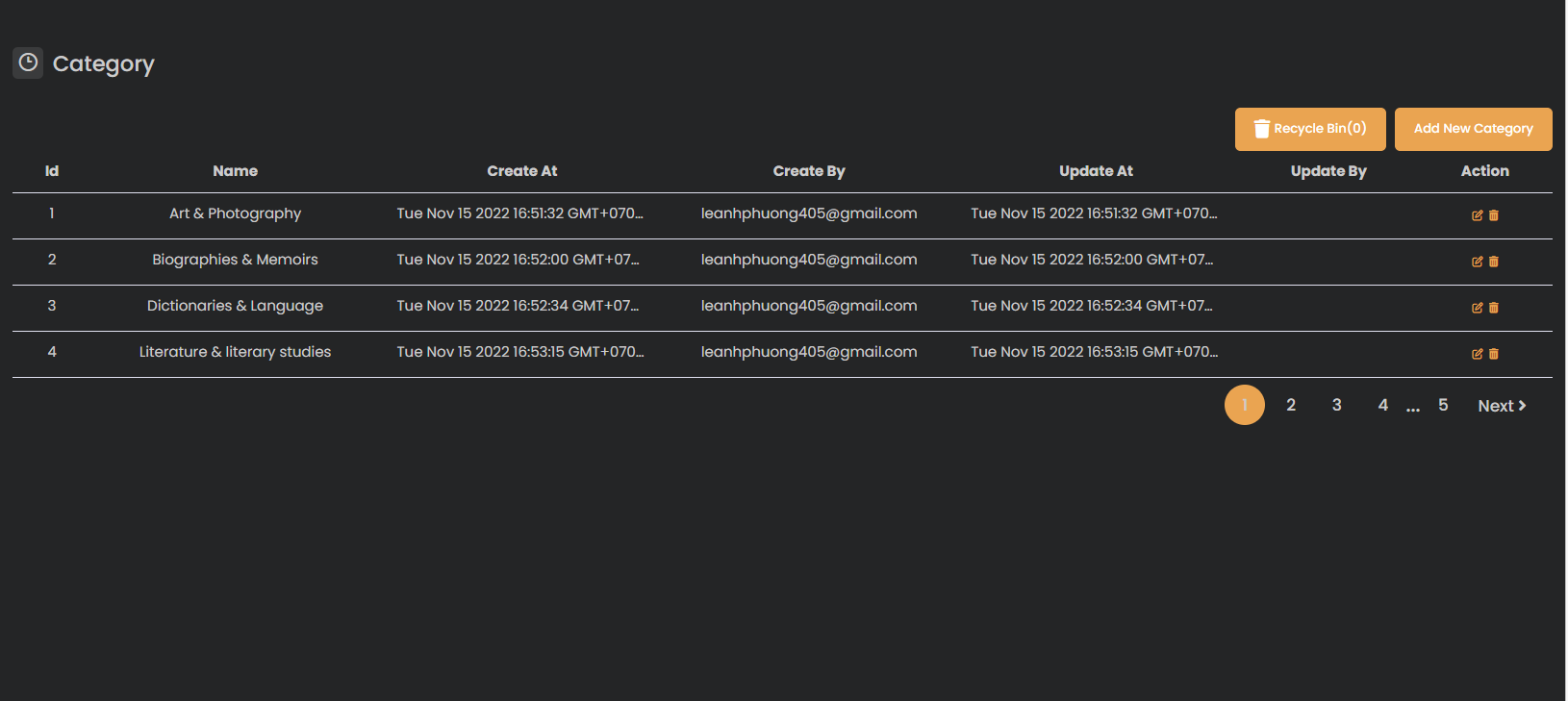
* Form tạo thông tin sản phẩm



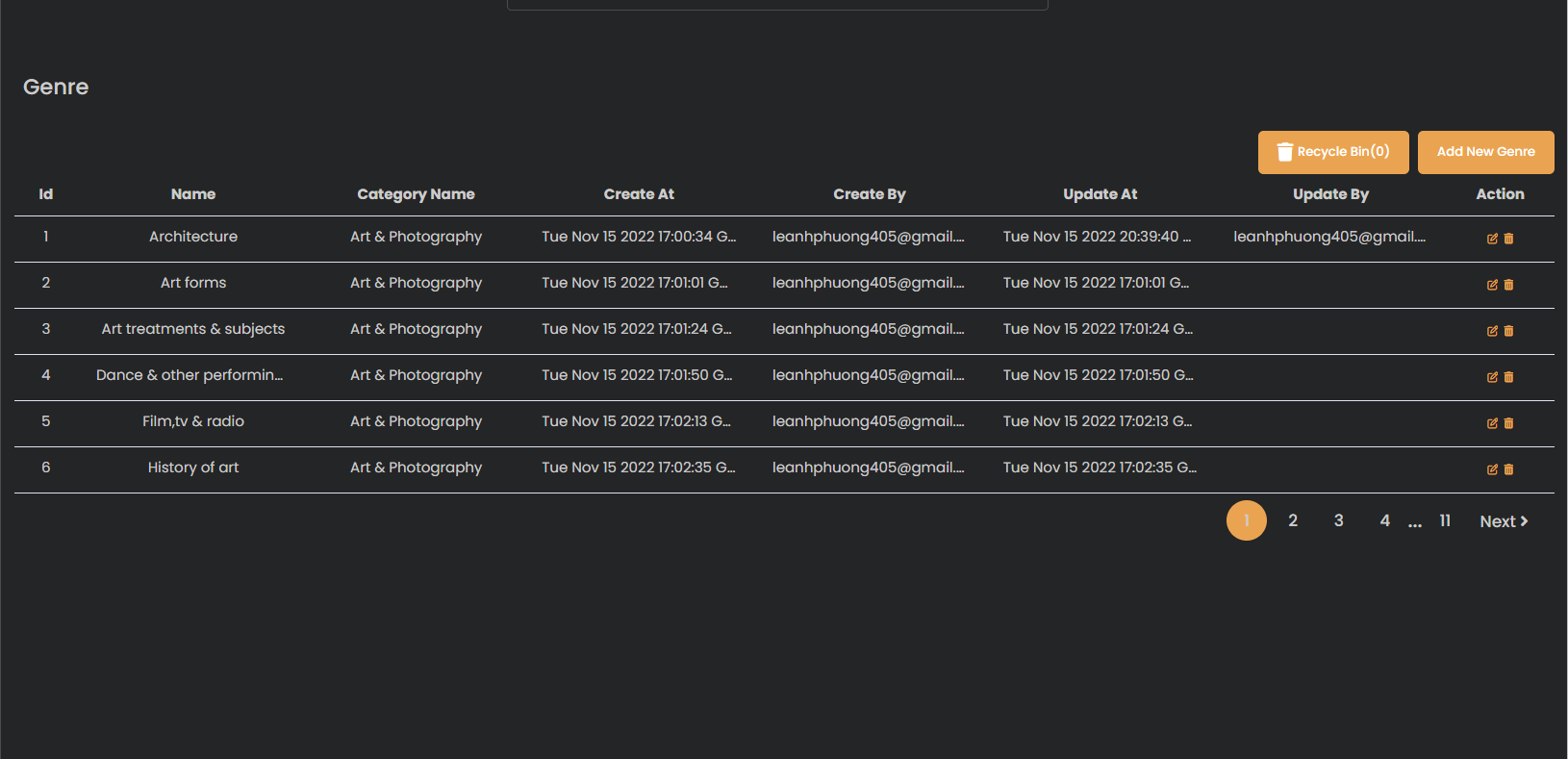
* Form quản lý tác giả



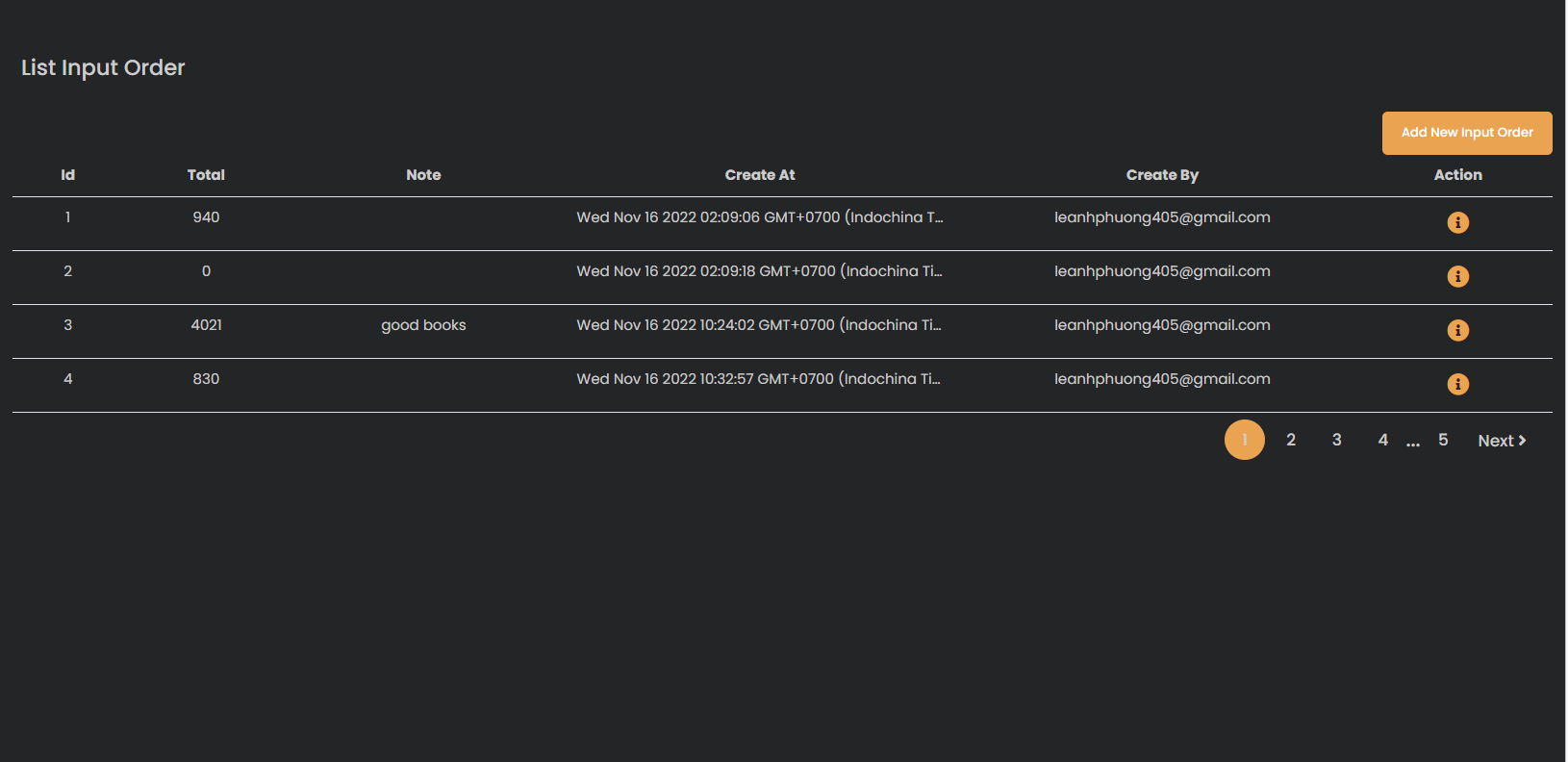
* Form quản lý Loại sách:



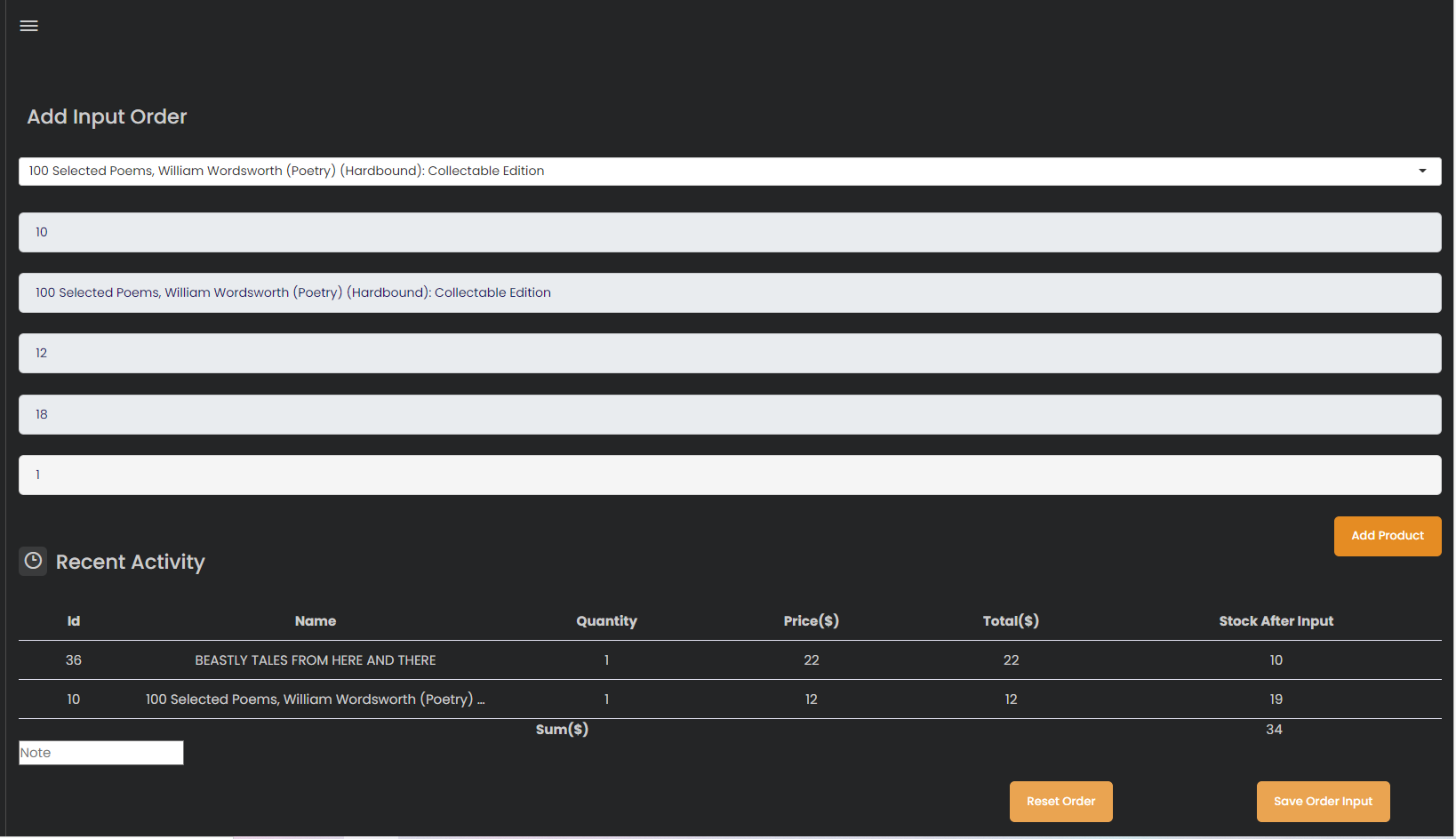
* Form quản lý Thể loại sách:



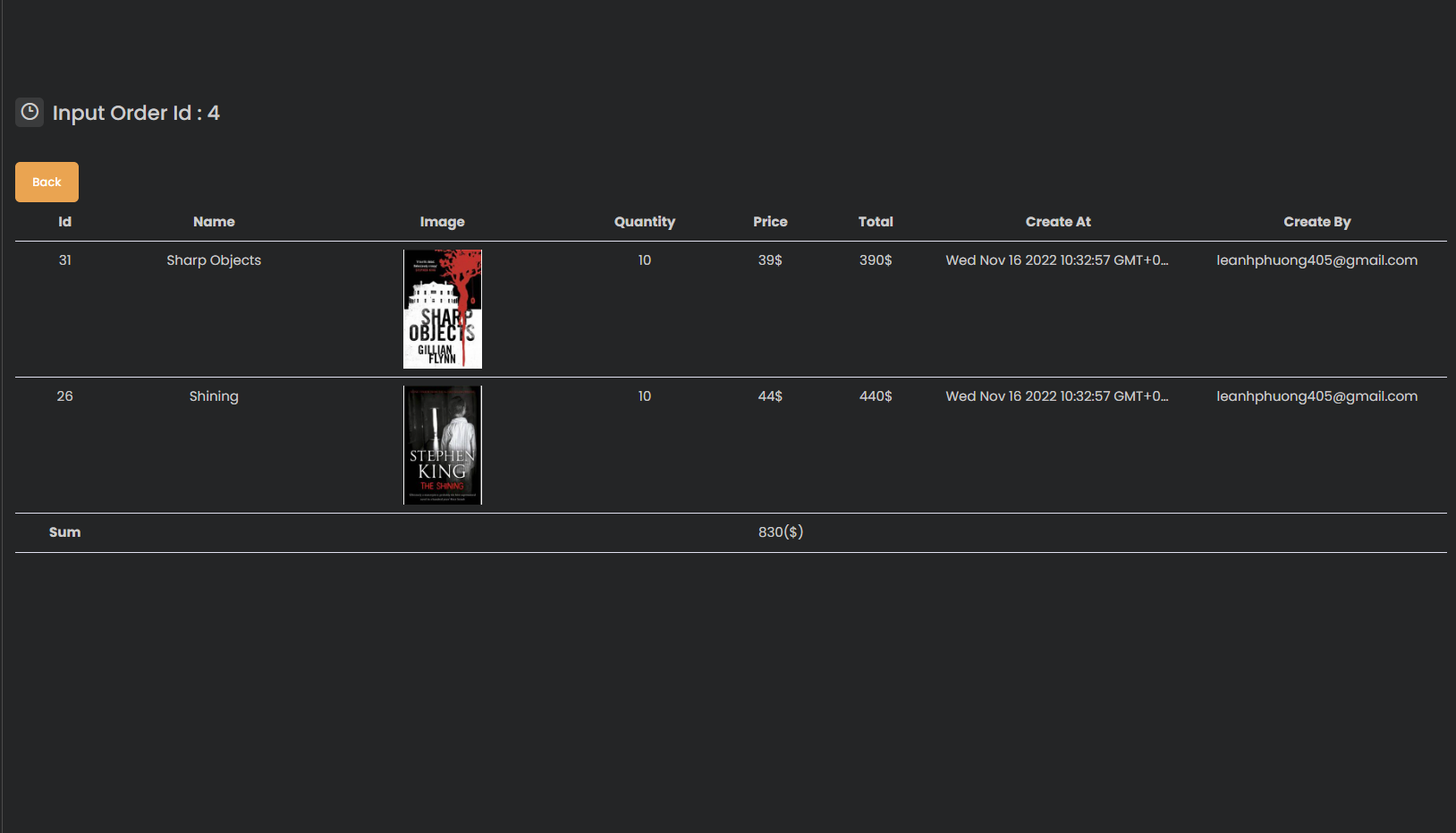
* Quản lý hóa đơn nhập:



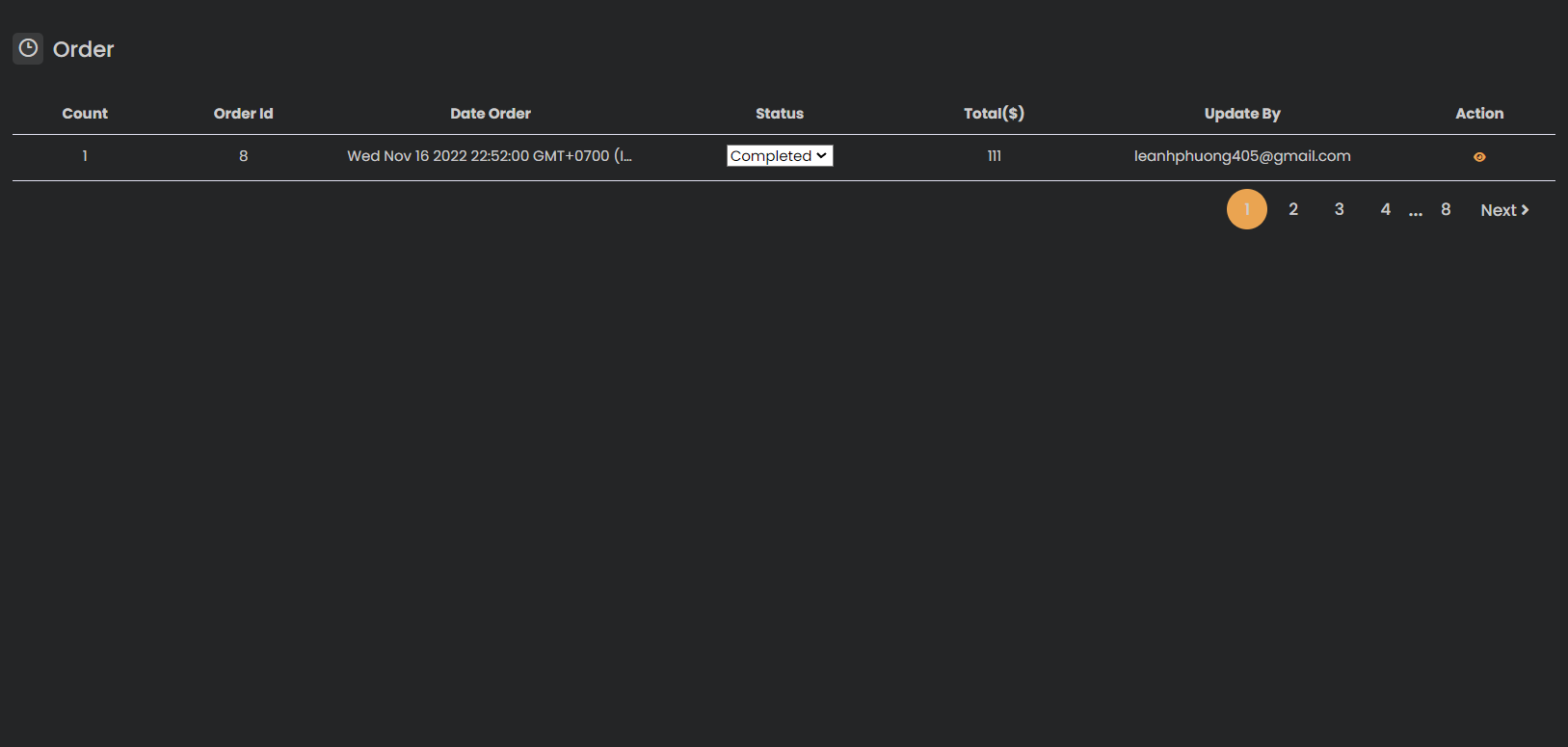
* Form hoá đơn nhập:



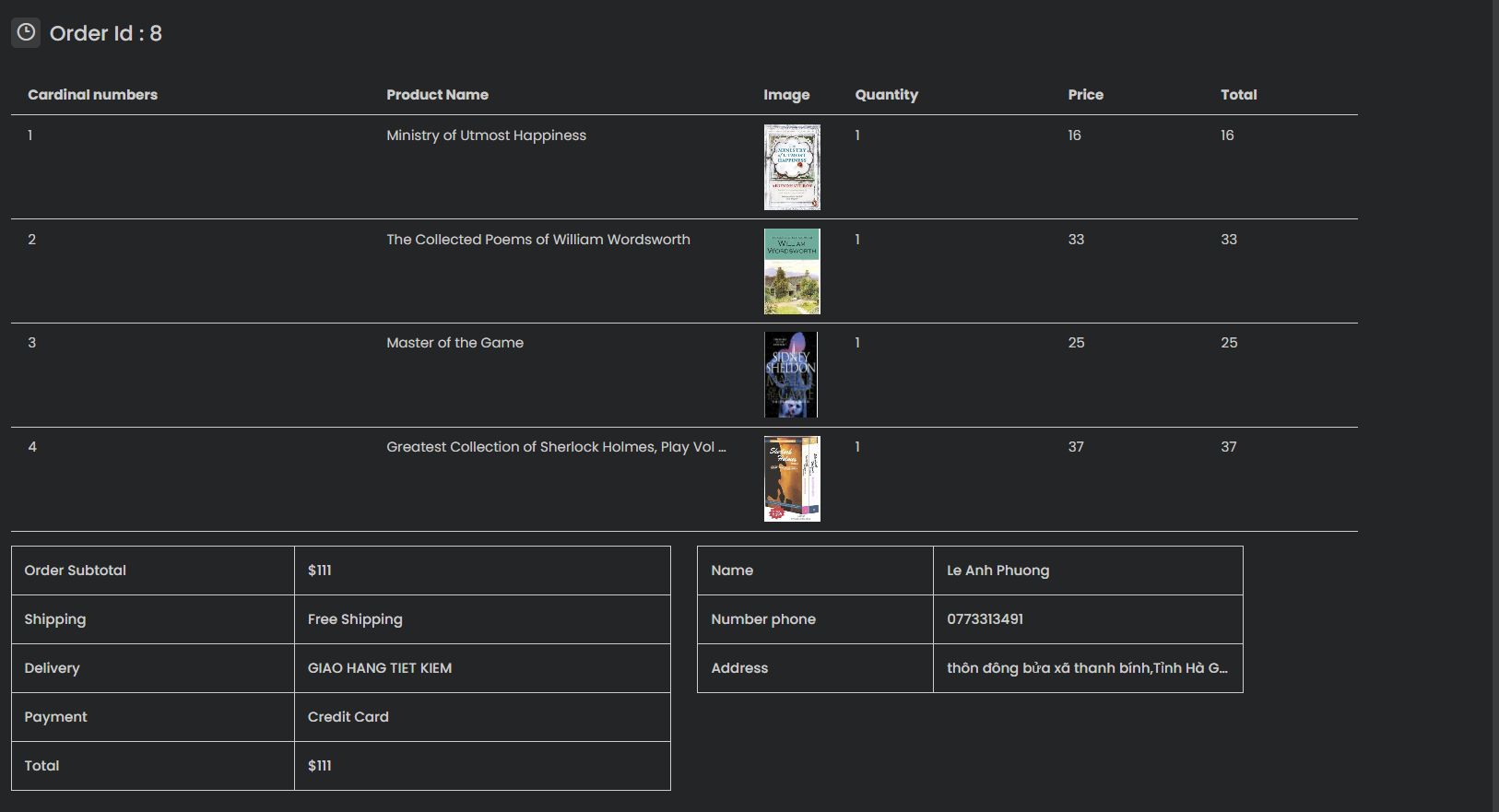
* Chi tiết hoá đơn nhập



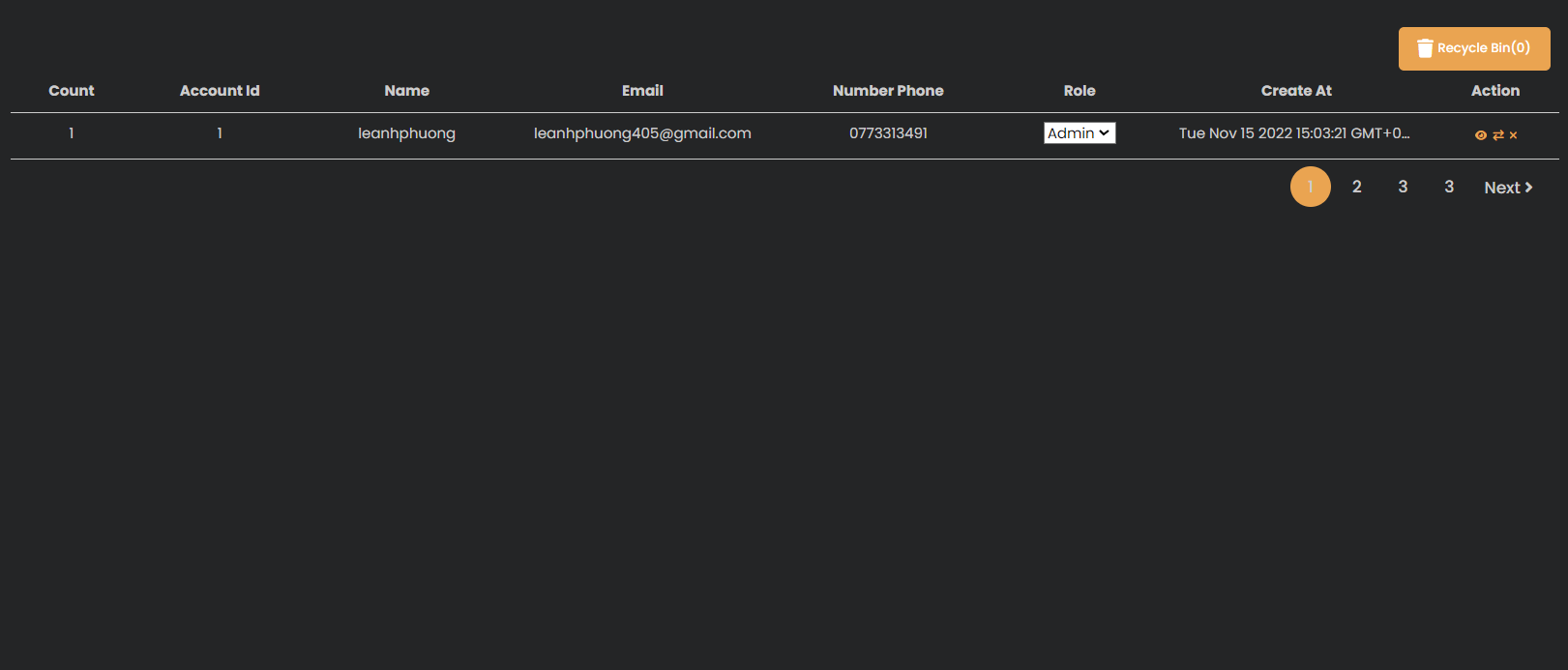
* Quản lí hoá đơn bán:



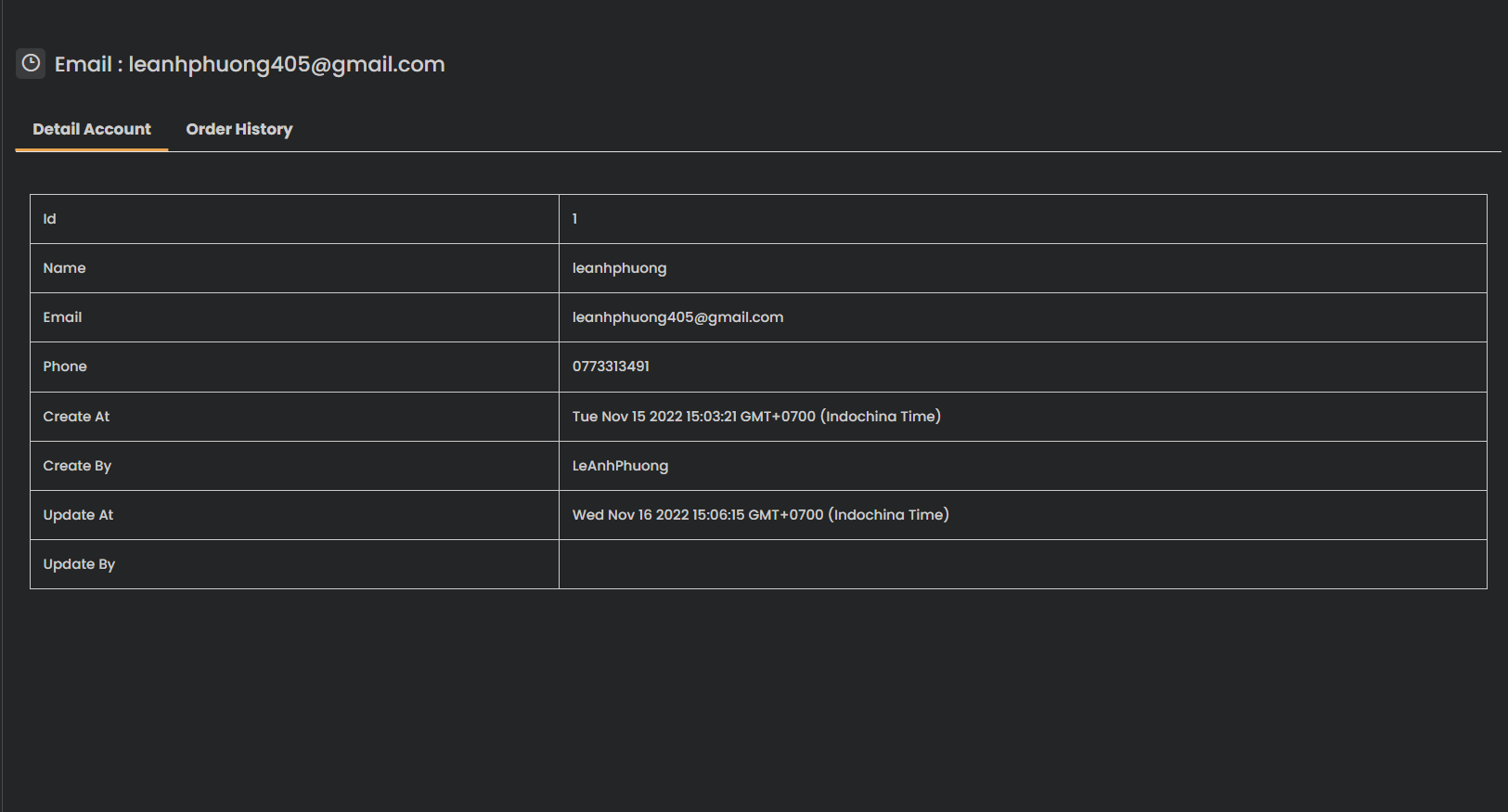
* Chi tiết hóa đơn bán



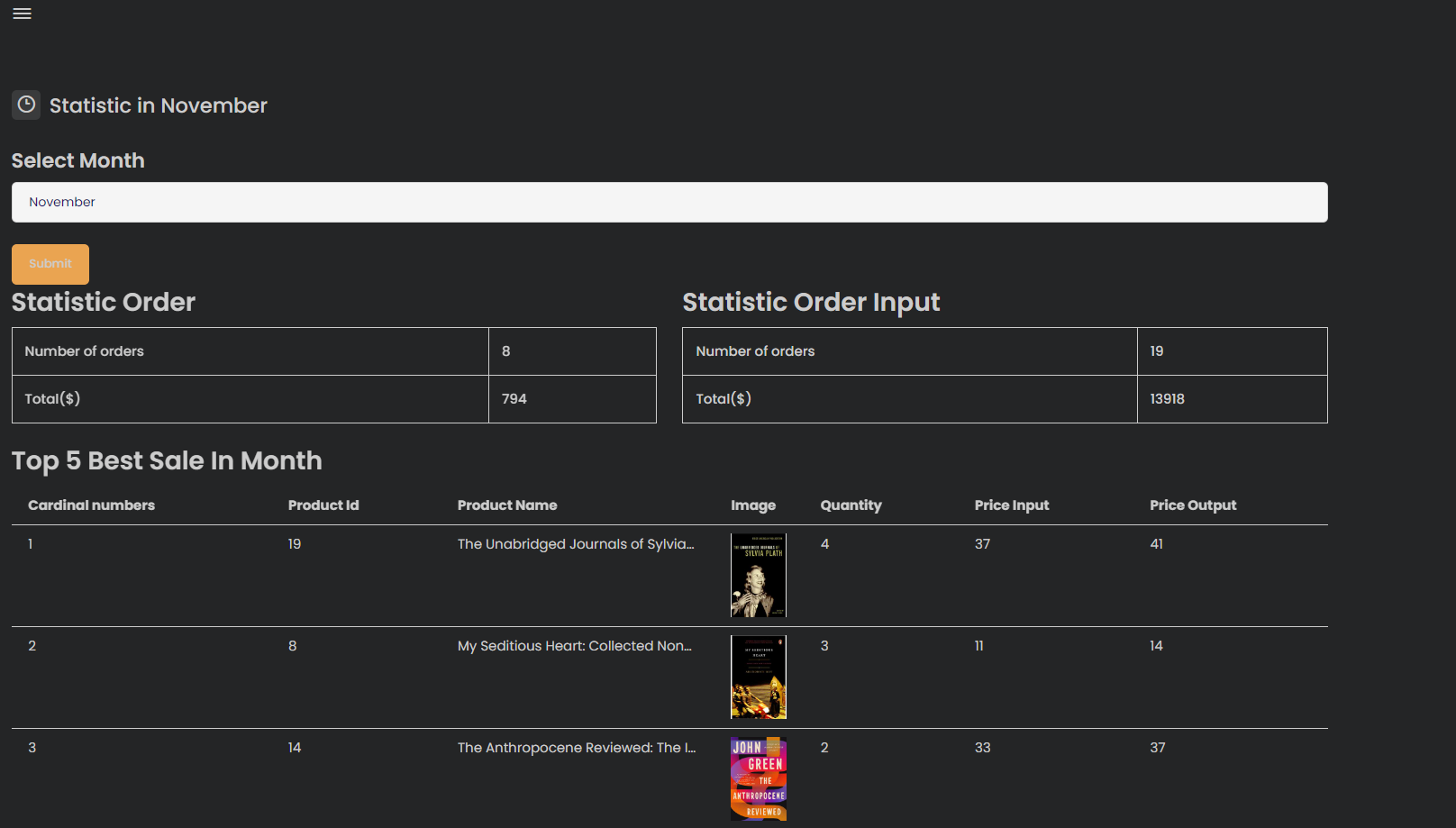
* Quản lí người dùng



* Chi tiết người dùng



* Trang thống kê



1. Code xử lí (tệp đính kèm)

* Kết quả đạt được:
* Hướng phát triển cho dự án:

+ Tính năng sản phẩm yêu thích của khách hàng.

+ Tính năng đánh giá sản phẩm khi đã mua sản phẩm.

+ Thanh toán online thông qua ví điện tử vnpay.

1. Kiểm thử.
2. Kết luận

Qua quá trình thục hiện đề tài, em đã tự tìm hiểu các công nghệ và áp dụng các kiến thức trong thời gian học tại trường để hoàn thành bài tập kết thúc môn học. Đồng thời, em đã tìm hiểu và nắm được quy trình xây dựng hệ thống, nâng cao kỹ năng lập trình, có thể chủ động trong việc tìm kiếm các thông tin để phục vụ công việc.

Sau khi hoàn thành bài tập lớn, đề tài tập trung xử lí các kĩ thuật liên quan đến thương mại điện tử. Đề tài đã xử lí hết các chức năng cơ bản của một trang website bán hàng và đã kiểm thử cho ra kết quả như mong muốn.

Hướng phát triển tiếp theo là tích hợp thanh toán của các bên thứ ba như VNPay, momo,….