

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



BÀI BÁO CÁO CUỐI KỲ MÔN WEB NÂNG CAO

POINT OF SALE BẰNG NODEJS

Người hướng dẫn: **GV. VŨ ĐÌNH HỒNG**

Người thực hiện:

HUỲNH NHẬT HẢO - 520H0531

TRẦN LÊ MINH TRÍ - 520H0500

Lớp : F512

Khoá : 24

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



BÀI BÁO CÁO CUỐI KỲ MÔN WEB NÂNG CAO

POINT OF SALE BẰNG NODEJS

Người hướng dẫn: **GV. VŨ ĐÌNH HỒNG**

Người thực hiện:

HUỲNH NHẬT HÀO - 520H0531

TRẦN LÊ MINH TRÍ- 520H0500

Lớp : **F512**

Khoá : **24**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

LỜI CẢM ƠN

Bài báo cáo lần này được nhóm hoàn thành với nhiều sự thuận lợi đến từ nhiều mặt nhưng quan trọng nhất là từ sự định hướng trong hướng dẫn của thầy Vũ Đình Hồng. Ngoài ra, cả nhóm rất cảm kích những giờ dạy và học nhiệt tình trên cả lớp lý thuyết và thực hành của thầy. Các yếu tố được nêu ở trên đã góp phần không nhỏ trong việc hoàn thành được báo cáo lần này. Vì thế nên nhóm rất chân thành cảm kích sự giúp đỡ cả trực tiếp và gián tiếp đến từ thầy. Bài báo cáo cũng như sản phẩm đã được nhóm hoàn thiện một cách cẩn thận và có kế hoạch nhất. Tuy nhiên mọi sản phẩm đều sẽ có những ưu và khuyết điểm khác nhau. Nên nhóm mong nhận được sự góp ý đến từ thầy để nhóm có thể cải thiện được trong tương lai, cụ thể hơn là các bài tập môn sau này. Cả nhóm xin một lần nữa được gửi lời cảm ơn ạ!

ĐỒ ÁN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Nhóm tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của GV. Vũ Đình Hồng. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào nhóm tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do nhóm tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 10 tháng 12 năm 2023

Tác giả

(ký tên và ghi rõ họ tên)

Huỳnh Nhật Hào

Nguyễn Minh Trí

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

Phần đánh giá của GV chấm bài

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

TÓM TẮT

Bài tập cuối kỳ lần này nhóm chọn thực hiện đề tài là tạo một ứng dụng point of sale bằng nodejs. Đây là một tập sử dụng ReactJS hỗ trợ trong việc viết một ứng dụng có giao diện. Nhóm đã sử dụng các tính năng đến từ React js như là chia các component và các thư viện đi kèm để phát triển được ứng dụng POS lần này. Và cũng vì thế nên ứng dụng ngoài chức năng bán hàng thì nó còn có đồ họa bắt mắt. Phía Server mô hình nhóm phân chia được nhóm áp dụng để phát triển lần này là khá giống với MVC. Cũng vì thế mà nhóm gặp thuận lợi trong việc làm việc nhóm nhờ có sự phân biệt rõ ràng giữa các chức năng. Ngoài ra, việc tạo các Router Controller và Services tương ứng của nó giúp nhóm tập trung vào việc viết code để xử lý event của người dùng. Tuy nhiên nhóm cũng gặp phải những khó khăn trong việc phát triển ứng dụng. Khó khăn đầu tiên là việc thiếu về trí lực và nhân lực trong nhóm nên một số chức năng ban đầu không thể được thực hiện hóa. Thứ hai là nhóm nhận ra mình vẫn chưa phát triển được khả năng thiết kế một phần mềm mà có thể được đánh giá là tối ưu và phù hợp với đa số người dùng. Khó khăn là vậy nhưng các thành viên trong nhóm đã vận dụng tốt những kiến thức từ cơ sở dữ liệu để mô hình hóa các đối tượng và thiết kế giao diện để cho một sản phẩm bắt mắt và dễ nhìn. Nhìn chung, bài tập lần này nhóm đã học thêm được một công nghệ Node js để phát triển ứng dụng , tập luyện thêm khả năng thiết kế hệ thống và nhận ra những kiến thức cần thiết để phát triển phần mềm tốt hơn.

MỤC LỤC

LỜI CẢM ƠN	1
PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN.....	3
TÓM TẮT	4
MỤC LỤC.....	1
CÁC CHỮ VIẾT TẮT	3
DANH MỤC CÁC HÌNH ẢNH	4
CHƯƠNG 1 – GIỚI THIỆU ĐỀ TÀI	5
1.1. Mục đích chọn ứng dụng POS	5
1.2. Sản phẩm cuối của đề tài.....	5
1.2.1. Các vấn đề chính mà nhóm cần phải giải quyết	5
1.2.2. Kế hoạch xây dựng ứng dụng	5
1.2.3 Yêu cầu Đề Tài Cần Thực Hiện.....	6
CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT.....	8
2.1 Lý thuyết về Express JS	8
2.2 Lý Thuyết về ReactJS	9
2.2.1 Ưu Điểm của ReactJS	9
2.3 Lý Thuyết về Cơ Sở Dữ Liệu Mongo	9
CHƯƠNG 3 – PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ	11
3.1. Yêu cầu của người dùng	11
3.1.1. Tìm hiểu về người dùng và ý tưởng	11
3.1.2. Functional Requirement.....	11
3.1.3. System requirement	14
3.2. Giao diện chính của ứng dụng	14
3.2.1 .Giao Diện Chính của User	14
3.2.1 .Giao Diện Chính của Admin.....	15
3.3. Kiến trúc của ứng dụng	16

3.1.1 quản lý khách hàng	18
3.1.2 quản lý sản phẩm	19
3.1.3 Quá Trình Giao Dịch	20
3.1.4 Reporting and Analytics	21
3.4 Sơ đồ ERD và định nghĩa các model	22
3.1.4 Sơ Đồ ERD	22
CHƯƠNG 4 – Hiện Thực Hệ Thống	23
4.1. Tổng quát về ứng dụng	23
CHƯƠNG 5 – TỔNG KẾT	23
5.1 Kết quả đạt được	23
5.2 Những hạn chế cần khắc phục	25
5.3 Hướng phát triển trong tương lai	26

CÁC CHỮ VIẾT TẮT

UI	Giao diện người dùng
CSDL	Cơ sở dữ liệu
POS	Point of sale
ERD	Entity Relational Diagram
MVC	Model View Controller

DANH MỤC CÁC HÌNH ẢNH

Hình 2.1 Tạo server với express	6
Hình 2.2 Kết nối với mongoDB	8
Hình 3.1 Sơ đồ Use Case admin	10
Hình 3.2 Sơ đồ Use Case user.....	11
Hình 3.3 Giao diện chính của user	13
Hình 3.4 Giao diện chính của admin.....	13
Hình 3.5 Kiến trúc server	14
Hình 3.6 Sơ đồ các chức năng.....	15
Hình 3.7 Sơ đồ luồng đăng nhập	15
Hình 3.8 Sơ đồ ERD.....	20
Hình 4.1 Tổng quát ứng	36
Hình 5.1 Danh sách các bài hát được lưu trong ứng dụng.....	21
Hình 5.2 Giao diện thanh toán	22
Hình 5.3 Giao diện chưa responsive	23

CHƯƠNG 1 – GIỚI THIỆU ĐỀ TÀI

Chương này, nhóm sẽ nói về các lí do chọn đề tài, các vấn đề chính cần phải giải quyết và kế hoạch làm việc của nhóm.

1.1. Mục đích chọn ứng dụng POS

Như đã được phổ biến trên lớp về các dự án được chọn cho bài cáo cáo lần này, nhóm em quyết định chọn thực hiện đề tài là tạo một ứng dụng POS. Tại sao nhóm lại chọn thực hiện đề tài này thì có lí do chính là nhóm muốn hiểu về quá trình bán hàng tại các cửa hàng bán lẻ hoạt động được ánh xạ qua lập trình sẽ như thế nào

1.2. Sản phẩm cuối của đề tài

1.2.1. Các vấn đề chính mà nhóm cần phải giải quyết

Với đề tài này nhóm sẽ phát triển một trang web cung cấp chức năng Point of Sale (POS) để bán hàng tại cửa hàng bán lẻ điện thoại và phụ kiện. Người sử dụng ứng dụng web này là nhân viên bán hàng và quản trị viên tại cửa hàng điện thoại. Người mua điện thoại không phải là người sử dụng của ứng dụng web này. Đơn giản, ứng dụng web này cần cung cấp các chức năng như: giao dịch bán hàng, quản lý sản phẩm, quản lý nhân viên, xem báo cáo và thống kê. Loại ứng dụng này thường được sử dụng tại các cửa hàng bán lẻ ở Việt Nam như cửa hàng điện tử, quần áo, thực phẩm và đồ uống hoặc siêu thị.

1.2.2. Kế hoạch xây dựng ứng dụng

Để giải quyết các vấn đề trên, nhóm với số lượng thành viên là 2 người sẽ bắt đầu xây dựng theo kế hoạch theo trình tự sau:

- Thiết kế màn hình giao diện chính bằng Reactjs : Để có thể dễ dàng trong việc chạy thử và phát triển các chức năng.
- Thiết kế server bằng node js: Để giao diện có dữ liệu động phía server sẽ viết các api cho client
- Thiết kế cơ sở dữ liệu : Nhóm đã lựa chọn một cơ sở dữ liệu không có quan hệ đó là mongo

Với kế hoạch như trên thì nhóm sẽ bắt đầu thực hiện và cải thiện sản phẩm theo từng phiên bản khác nhau để dễ dàng cải thiện thêm chức năng.

1.2.3 Yêu cầu Đề Tài Cần Thực Hiện

1.2.3.1 Quản lý tài khoản

- Tạo sẵn tài khoản quản trị viên với thông tin đăng nhập là admin/admin.
- Nhân viên bán hàng không thể tự tạo tài khoản, chỉ quản trị viên mới có thể tạo.
- Quản trị viên cần cung cấp thông tin nhân viên khi tạo tài khoản, và gửi email thông báo với liên kết đăng nhập trong 1 phút.
- Nhân viên mới phải sử dụng liên kết để đăng nhập lần đầu, không thể đăng nhập trực tiếp bằng form đăng nhập.
- Mọi người dùng đăng nhập bằng tên người dùng là phần đầu của email. Nhân viên mới có mật khẩu tạm thời giống với tên người dùng.
- Sau lần đăng nhập đầu tiên, nhân viên phải tạo mật khẩu mới.
- Tất cả nhân viên có thể xem và cập nhật thông tin cá nhân, bao gồm ảnh đại diện và tên đầy đủ.
- Quản trị viên có thể thực hiện các chức năng liên quan đến nhân viên như xem danh sách, khóa/mở khóa tài khoản, xem thông tin bán hàng

1.2.3.2 Quản lý Danh mục Sản phẩm (chỉ dành cho quản trị viên)

- Quản trị viên có thể xem, thêm mới, cập nhật và xóa sản phẩm.
- Sản phẩm chỉ có thể bị xóa khi không có trong đơn hàng nào.
- Nhân viên bán hàng chỉ có quyền xem danh sách sản phẩm, không thể xóa hoặc xem giá gốc.

1.2.3.3 Quản lý khách hàng

- Nhân viên thanh toán sẽ yêu cầu số điện thoại của khách hàng.
- Hệ thống tự động hiển thị thông tin nếu khách hàng đã mua trước đó.
- Nếu là lần mua đầu tiên, nhân viên nhập thông tin để tạo tài khoản khách hàng tự động

1.2.3.4 Xử lý giao dịch

- Chức năng chính của POS, cho phép nhập sản phẩm thông qua tìm kiếm hoặc mã vạch và tên sản phẩm.
- Sản phẩm được hiển thị ngay sau khi thêm vào, thông số cập nhật tự động.
- Tại giao diện thanh toán, nhân viên nhập thông tin khách hàng và hoàn thành thanh toán.

1.2.3.5 Báo Cáo và Phân Tích

- Hiển thị kết quả bán hàng theo thời gian, có thể chọn các khoảng thời gian khác nhau.
- Cho biết tổng số tiền nhận được, số đơn hàng và sản phẩm, cùng với danh sách đơn hàng theo thứ tự thời gian.
- Cả quản trị viên và nhân viên bán hàng có thể xem thông tin này, trong khi quản trị viên có thể xem thêm về lợi nhuận tổng cộng.

CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT

Để có thể phát triển thành công ứng dụng này, nhóm sẽ cần phải tìm hiểu về cơ sở lý thuyết Express JS ReactJS .

2.1 Lý thuyết về Express JS

Express.js là một framework web dựa trên Node.js, giúp xây dựng ứng dụng web và API một cách nhanh chóng và dễ dàng. Nó sử dụng cú pháp đơn giản và mô hình middleware để xử lý các yêu cầu HTTP. Express.js giúp quản lý định tuyến, xử lý middleware, và tạo các ứng dụng web hiệu quả. Được biết đến với tính nhẹ, linh hoạt, và cộng đồng lớn, Express.js thường được sử dụng làm lựa chọn chính để phát triển ứng dụng web Node.js.



```

1  const express = require('express')
2  const { PORT } = require('./config')
3  const expressApp = require('./express-app')
4  const { connectDB } = require('./database')
5  const http = require('http')
6
7  const startServer = async () => {
8      const app = express()
9      const server = http.createServer(app)
10     // set up middlewares
11     await expressApp(app)
12     // connect to db
13     await connectDB()
14
15     // run server
16     server.listen(PORT, () => {
17         console.log(`Server is running on port ${PORT}`)
18     })
19     // close app when it error
20     app.on('error', (err) => {
21         console.log(err)
22         process.exit()
23     })
24 }
25
26 startServer()
27

```

Hình 2.1 Tạo server với express

2.2 Lý Thuyết về ReactJS

React.js là một thư viện JavaScript mã nguồn mở được phát triển bởi Facebook, chủ yếu được sử dụng để xây dựng giao diện người dùng hiện đại và linh hoạt trong các ứng dụng web. Nó tập trung vào việc tạo ra các thành phần tái sử dụng và quản lý trạng thái ứng dụng một cách hiệu quả, giúp cải thiện hiệu suất và sự duyệt.

2.2.1 Ưu Điểm của ReactJS

- React sử dụng cú pháp JSX, một mở rộng của JavaScript, để viết các thành phần giao diện một cách dễ đọc và dễ hiểu.
- Các thành phần React giúp tách biệt giao diện thành các khối tái sử dụng, giúp quản lý mã nguồn và bảo trì một cách dễ dàng.
- React sử dụng mô hình Virtual DOM để cải thiện hiệu suất bằng cách cập nhật chỉ những phần thay đổi thực sự trong DOM thực tế.

2.3 Lý Thuyết về Cơ Sở Dữ Liệu Mongo

MongoDB là một hệ quản trị cơ sở dữ liệu thuộc dạng NoSQL, nổi tiếng với việc không sử dụng SQL và bảng như trong các hệ quản trị cơ sở dữ liệu quan hệ. Thay vào đó, MongoDB sử dụng JSON-like BSON (Binary JSON) để lưu trữ dữ liệu, giúp làm việc với dữ liệu không có cấu trúc hoặc cấu trúc dữ liệu thay đổi trở nên thuận lợi. Một trong những ưu điểm quan trọng của MongoDB là tính nhẹ và linh hoạt. Với việc thuộc dạng cơ sở dữ liệu NoSQL, MongoDB giúp linh hoạt khi làm việc với dữ liệu không có cấu trúc đặc biệt hoặc khi cần thay đổi cấu trúc dữ liệu một cách dễ dàng. Dữ liệu trong MongoDB được lưu trữ dưới dạng BSON, một dạng nhị phân của JSON, tạo ra cấu trúc dữ liệu dễ đọc và hiểu. MongoDB không yêu cầu định rõ cấu trúc của dữ liệu trước, điều này giúp giảm sự phức tạp khi phát triển và làm cho quá trình bảo trì dễ dàng hơn. MongoDB cũng giữ tính mở rộng và linh hoạt thông qua khả năng mở rộng cơ sở dữ liệu theo chiều ngang. Điều này giúp ứng dụng mở rộng linh hoạt khi cần thiết, giảm thiểu sự ràng buộc từ các quy định cứng nhắc. Ngoài ra, MongoDB hỗ trợ indexing rất phù hợp cho việc quét mã vạch trong POS và tra cứu hiệu quả, giúp tìm kiếm dữ liệu nhanh chóng trong các tập dữ liệu lớn.

```
1  const mongoose = require('mongoose')
2  const { MONGODB_URL } = require('../config')
3
4  class Database {
5    async connect() {
6      await mongoose
7        .connect(MONGODB_URL)
8        .then(_) =>
9          console.log(
10            `Connected to mongooseDB ${_.connection.db.databaseName} successful`,
11          ),
12    }
13    .catch(_) => console.log('Connect to mongoDB failed'))
14  }
15
16  static async getInstance() {
17    if (!this.instance) {
18      this.instance = new Database()
19      await this.instance.connect()
20    }
21    return this.instance
22  }
23 }
24
25 module.exports = {
26   connectDB: async () => {
27     return await Database.getInstance()
28   },
29 }
30
```

Hình 2.2 Kết nối với mongoDB

CHƯƠNG 3 – PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ

3.1. Yêu cầu của người dùng

3.1.1. *Tìm hiểu về người dùng và ý tưởng*

Các ứng dụng POS không thiếu mà thực sự có khá nhiều và một số là ứng dụng cung cấp sẵn nhiều chức, vậy thì sự khác biệt cần có của ứng dụng được xây dựng lần này là khả năng cho phép tạo thanh toán nhanh chóng, dễ dàng thao tác thân thiện với người dùng.

Theo quan sát trên thực tiễn thì người dùng thường sẽ có những đặc điểm sau:

- Thực hiện chọn các mặt hàng sau đó là thanh toán.
- Người dùng có thể xem lịch sử mua hàng của khách hàng cũng như thông tin khách hàng.
- Thực hiện in hóa đơn sau khi hoàn thành quá trình thanh toán.

Theo quan sát trên thực tiễn thì admin thường sẽ có những đặc điểm sau:

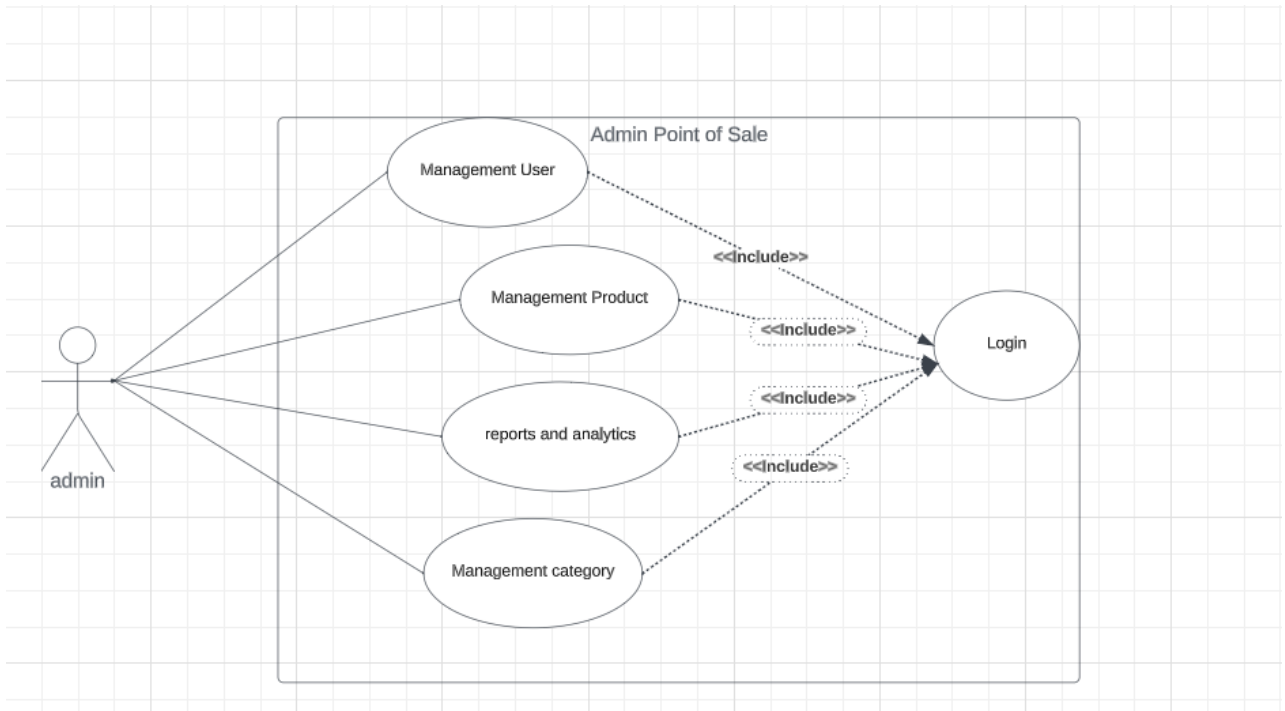
- Thực hiện quản lý danh mục sản phẩm.
- Quản lý các loại danh mục sản phẩm(category).
- Quản lý tài khoản của người dùng (nhân viên).
- Xem báo cáo cũng như phân tích của hệ thống.

Với ý tưởng cơ bản như vậy, nhóm nhận thấy để truyền đạt ý tưởng tốt hơn thì một sơ đồ Use case là cần thiết.

3.1.2. *Functional Requirement*

3.1.2.1 Functional requirement Admin

Functional requirement của admin là các yêu cầu về chức năng mà ứng dụng phải cung cấp là vừa quản lý sản phẩm, người dùng và báo cáo, phân tích. Nó sẽ có sơ đồ use case như sau:

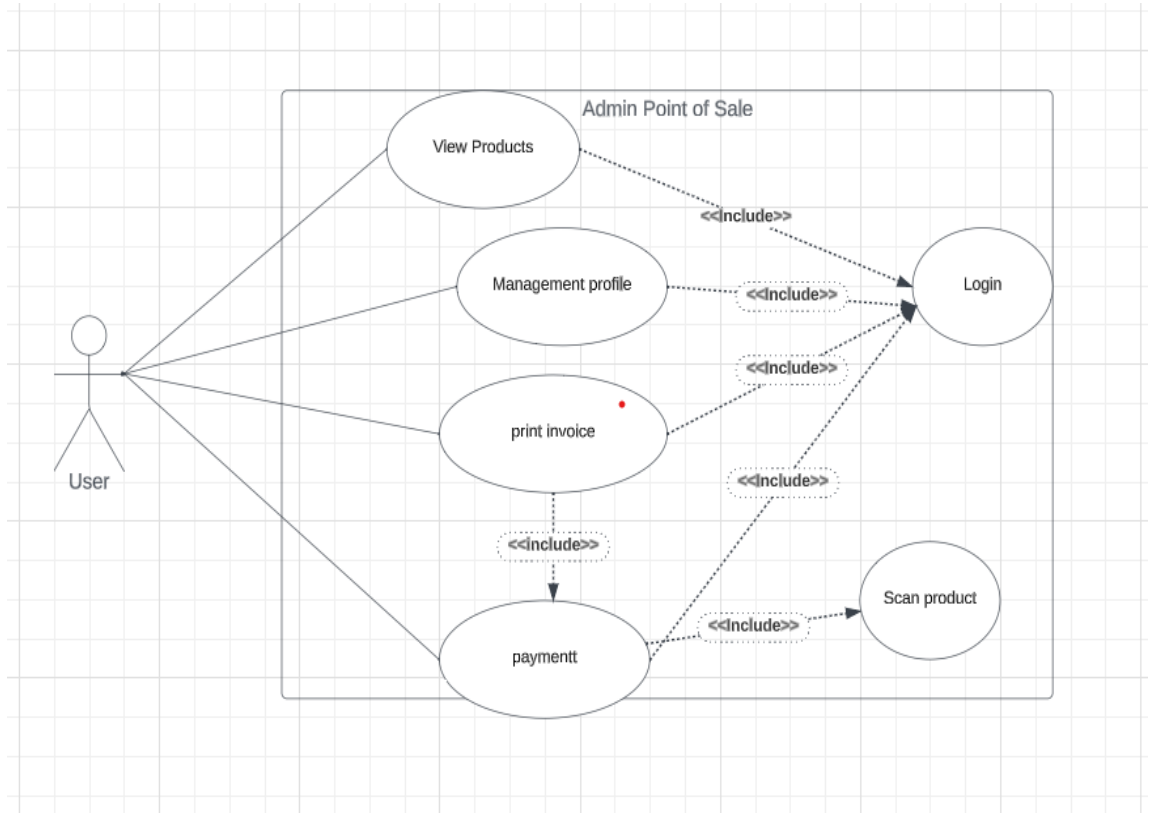


Hình 3.1 Sơ đồ Use Case Admin

Từ sơ đồ trên ta thấy ứng dụng sẽ cung cấp cho admin bốn chức năng chính là quản lý người dùng, trong quản lý user còn có các chức năng khác như là thêm, xóa, xem thông tin chi tiết của một người dùng. Tiếp theo đó là chức năng quản lý danh mục sản phẩm đó như là thêm xóa và sửa sản phẩm. Tiếp theo đó là chức năng quản lý loại sản phẩm cũng sẽ có các chức năng cơ bản như thêm xóa sửa. Cuối cùng là chức năng báo cáo và phân tích như là xem số lượng hóa đơn, tổng lợi nhuận, tổng thu nhập. Và có điều cần lưu ý là tất các chức năng trên bắt buộc admin phải đăng vào hệ thống.

3.1.2.2 Functional Requirement User

Functional requirement của user là các yêu cầu về chức năng mà ứng dụng phải cung cấp là tạo đơn đặt hàng cho khách hàng, in hóa đơn cho khách, xem các danh sách sản phẩm, thêm các sản phẩm vào order bằng scan barcode hoặc tìm kiếm theo tên sản phẩm, update profile. Nó sẽ có sơ đồ use case như sau:



Hình 3.2 Sơ đồ Use Case User

Từ sơ đồ trên ta thấy ứng dụng sẽ cung cấp cho người dùng bốn chức năng chính là xem các danh mục sản phẩm. Tiếp theo đó là chức năng quản lý profile như là xem hoặc sửa các thông tin người dùng. Tiếp theo đó là chức năng quản in hóa đơn, chức năng in hóa chỉ được khi có payment hoàn thành . Cuối cùng là chức năng payment, chức năng này bắt buộc nhân viên đã scan sản phẩm cho khách hàng. Và có điều cần lưu ý là tất các chức năng trên bắt buộc người dùng phải đăng vào hệ thống.

3.1.3. System requirement

Yêu cầu và chức năng đều đã được xác định. Bước tiếp theo là từ những yêu cầu trên ta sẽ triển khai ra theo từng bước nhỏ cụ thể mà ứng dụng phải thực hiện để đạt được chức năng yêu cầu. Để phát triển được một ứng dụng đáp ứng được yêu cầu đó thì system requirement cần được xác định rõ.

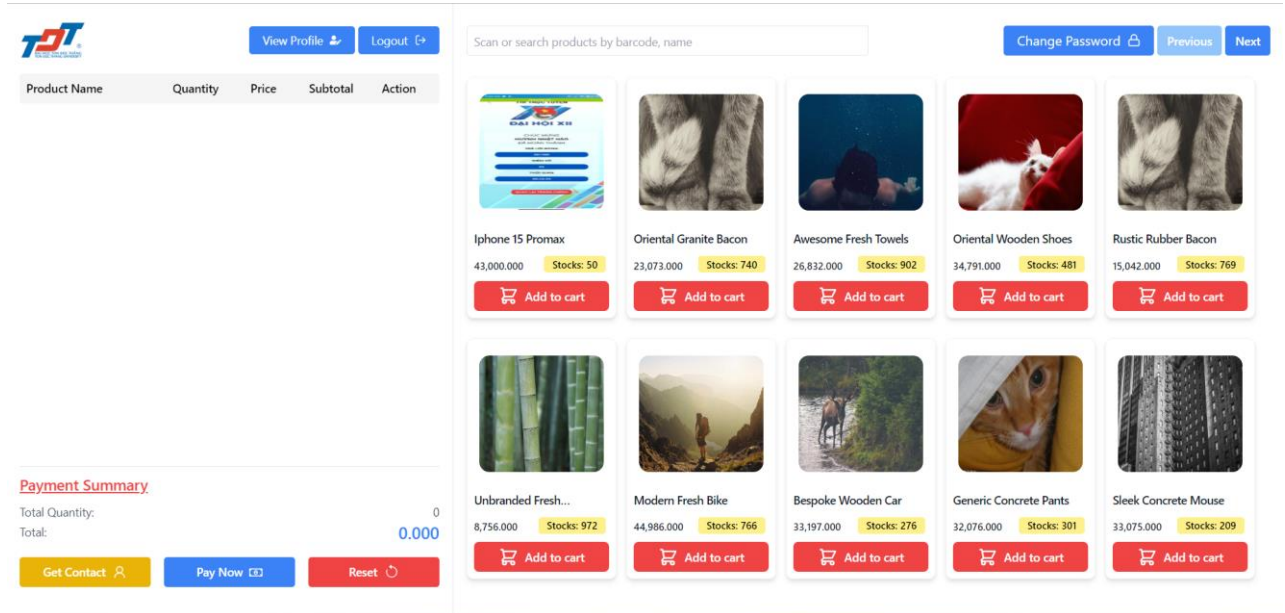
1. Không cho phép nhân viên tạo tài khoản, chỉ admin mới có quyền tạo tài khoản cho nhân viên. Nhân viên xác thực tài khoản qua email với thời hạn 1 phút. Bắt buộc lần đầu tiên đăng nhập phải qua liên kết của email sau khi đăng nhập phải đổi mật khẩu.
2. Admin thực hiện hành động gửi lại email trong vòng 1 phút sau khi đã gửi.
3. Yêu cầu kiểm tra trạng thái của sản phẩm có đang tồn tại trong đơn đặt hàng nào không nếu có thì sẽ không cho xóa.
4. Yêu cầu nhập thông tin khách hàng lần đầu tiên mua hàng, và tự động tạo lưu thông tin khách hàng. Lần sau khách hàng đến sẽ tự động lấy thông tin khách hàng.
5. Tự động hiển thị danh sách sản phẩm tổng tiền ngay khi thêm sản phẩm vào giỏ hàng. Tự động tính tổng tiền và tiền thối lại

System Requirement là các đặc tả yêu cầu mà mô tả ứng dụng, hệ thống sẽ phải làm gì để đạt mục tiêu, các dịch vụ và chức năng cần lập trình. Với các đặc tả yêu cầu trên mà mọi người trong nhóm có thể hình dung rõ các bước cần làm. Với việc xác định đầy đủ các yêu cầu, nhóm đã có đủ thông tin để lên thiết kế vào viết chương trình.

3.2. Giao diện chính của ứng dụng

Trước khi làm rõ cơ chế hoạt động của ứng dụng thì chúng ta nên nhìn qua về giao diện chính của admin và user và làm quen với một số thành phần quan trọng.

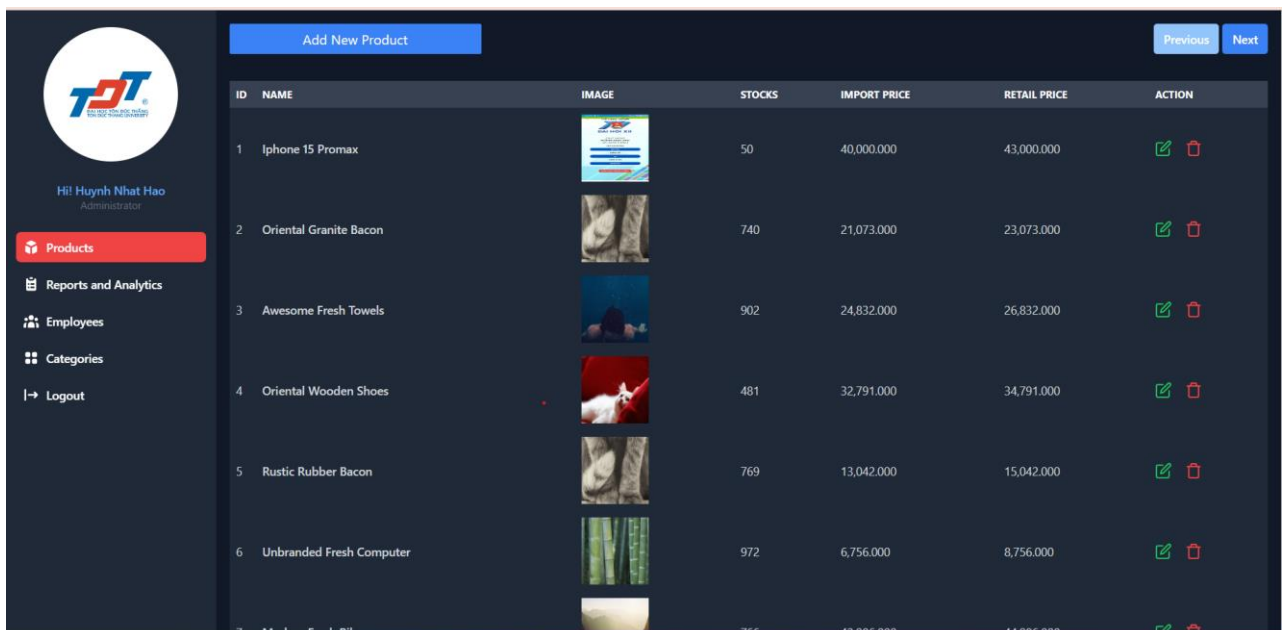
3.2.1 .Giao Diện Chính của User



Hình 3.3 Giao diện chính của user

Hình trên như ta có thể thấy chính là giao diện chính của ứng dụng gồm phần thanh toán và phần danh mục sản phẩm gồm có phân trang cho sản phẩm. Phần thanh toán gồm có chức năng hiển các sản phẩm được khách hàng order và có chức năng thanh toán cho khách hàng.

3.2.1 .Giao Diện Chính của Admin

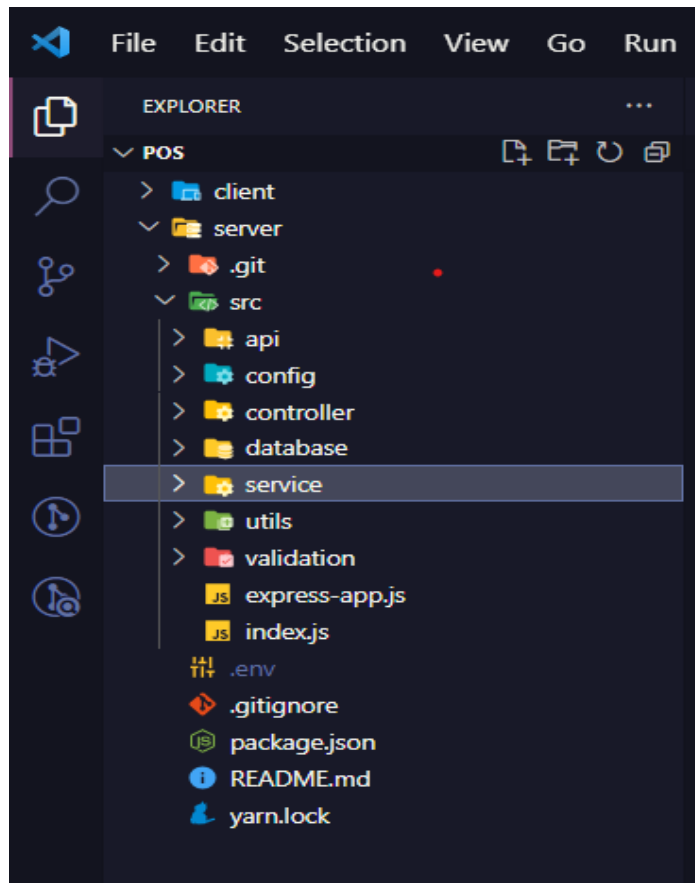


Hình 3.4 Giao diện chính của user

Hình trên như ta có thể thấy chính là giao diện chính của ứng dụng gồm phần điều hướng của trang admin và bên phải là phần động để hiển thị dữ liệu liên quan đến phần điều hướng .

3.3. Kiến trúc của ứng dụng

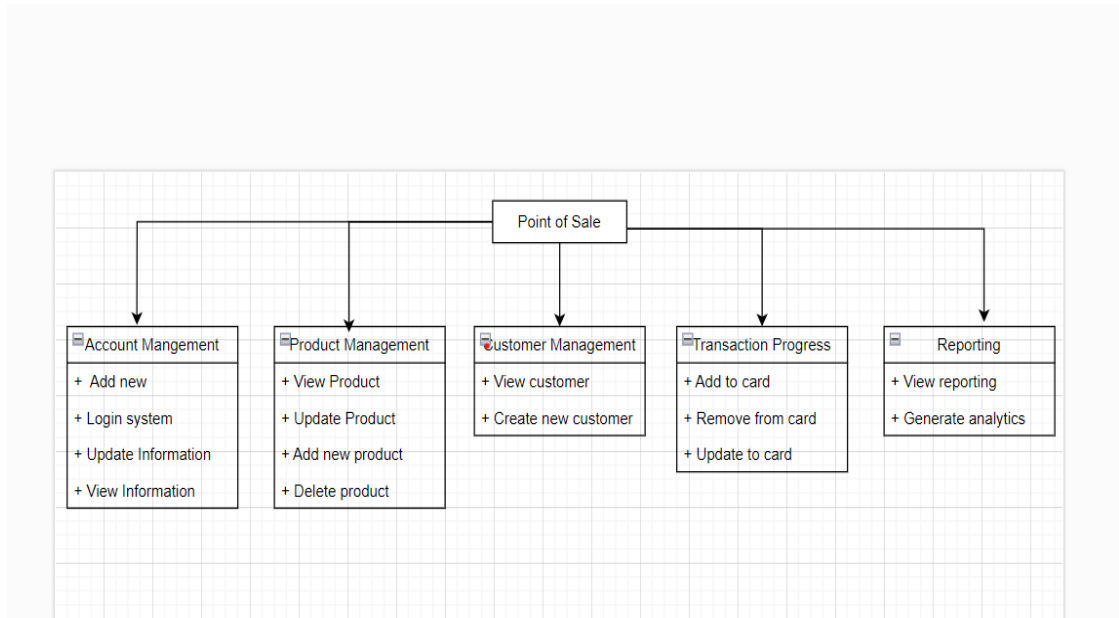
Ứng dụng được nhóm phát triển dựa trên mô hình thiết kế khá tương tự MVC. Thư mục code vì thế mà cũng được chia theo 3 mục này là chính



Hình 3.5 kiến trúc server

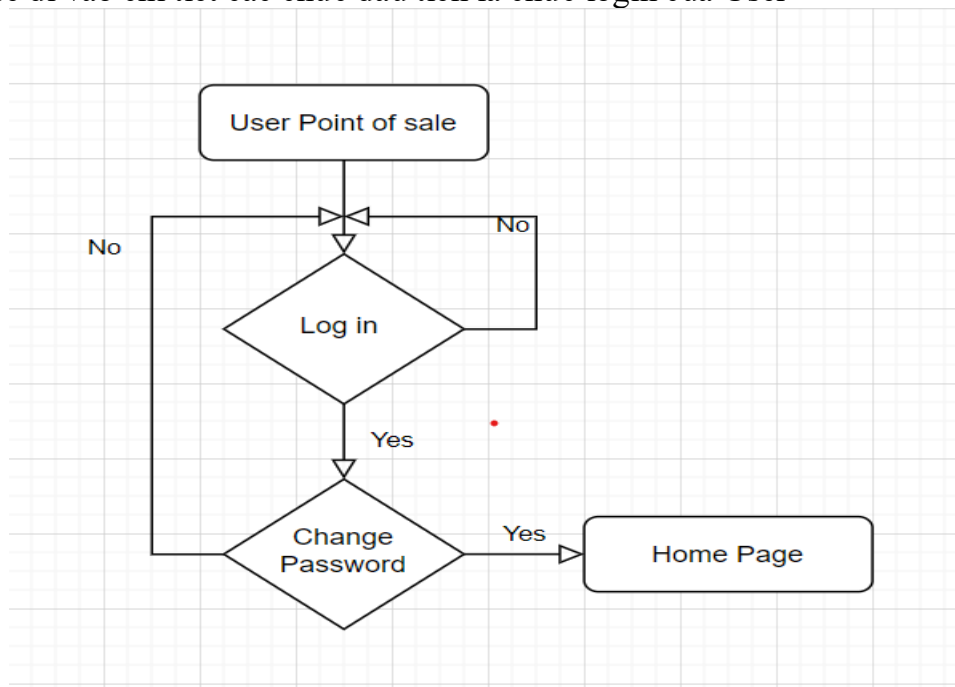
Mô hình thiết kế của nhóm em đó là thư mục API sẽ đảm nhận nhiệm vụ định nghĩa các router cho ứng dụng thư mục controller sẽ đảm nhận việc kết nối thư mục API và Service và sau đó controller sẽ trả về response. Trong folder Service em sẽ viết những logic và việc gọi đến database sẽ có 1 folder repository trong folder database đảm nhận. Và ứng của em sẽ được validation tầng controller và việc validation được thực đầu tiên. Ngoài ra trong folder api còn có folder middleware đảm nhận việc trung gian cho các router như là verify token or verify admin

Mọi ứng dụng dù nhỏ hay lớn đều sở hữu một kiến trúc. Đây chính là giải pháp mà ứng dụng đó đang nhắm đến. Đối với ứng dụng lần này thì nhóm nhận thấy kiến trúc của mình phải giải quyết các vấn đề sau đây



Hình 3.6 Sơ đồ các chức năng

Tiếp theo ta sẽ đi vào chi tiết các chức đầu tiên là chức login của User



Hình 3.7 Sơ đồ luồng đăng nhập

Tiếp nhóm em sẽ nói qua các chức còn lại

3.1.1 quản lý khách hàng

Hiển thị Danh sách Khách hàng:

- Salesperson/Admin yêu cầu hệ thống hiển thị danh sách khách hàng.
- Hệ thống truy xuất dữ liệu từ CSDL về thông tin của tất cả các khách hàng.
- Dữ liệu được trả về và hiển thị cho Salesperson/Admin.

Thêm mới Khách hàng:

- Salesperson/Admin nhập thông tin mới cho khách hàng (tên, số điện thoại, địa chỉ).
- Hệ thống lưu thông tin mới vào CSDL.
- Hệ thống trả về thông báo xác nhận hoặc thông báo lỗi.

Xem thông tin chi tiết Khách hàng:

- Salesperson/Admin chọn một khách hàng từ danh sách.
- Hệ thống truy xuất chi tiết của khách hàng từ CSDL.
- Chi tiết khách hàng được hiển thị cho Salesperson/Admin.

Xem Lịch sử Mua hàng của Khách hàng:

- Salesperson/Admin chọn xem lịch sử mua hàng của một khách hàng.
- Hệ thống truy xuất dữ liệu từ CSDL về các đơn đặt hàng của khách hàng đó.
- Dữ liệu được hiển thị với thông tin như tổng số tiền, sản phẩm mua, ngày mua, và các chi tiết khác.

-

3.1.2 quản lý sản phẩm

Hiện thị Danh sách Sản phẩm:

- Admin yêu cầu hệ thống hiển thị danh sách sản phẩm.
- Hệ thống truy xuất dữ liệu từ CSDL liên quan đến thông tin của tất cả các sản phẩm.
- Dữ liệu được trả về và hiển thị cho Admin.

Thêm mới Sản phẩm:

- Admin nhập thông tin mới cho sản phẩm (mã vạch, tên sản phẩm, giá nhập, giá bán, danh mục, ngày tạo).
- Hệ thống lưu thông tin mới vào CSDL.
- Hệ thống trả về thông báo xác nhận hoặc thông báo lỗi.

Xem chi tiết Sản phẩm:

- Admin chọn một sản phẩm từ danh sách.
- Hệ thống truy xuất chi tiết của sản phẩm đó từ CSDL.
- Chi tiết sản phẩm được hiển thị cho Admin.

Cập nhật thông tin Sản phẩm:

- Admin cập nhật thông tin của một sản phẩm.
- Hệ thống lưu thông tin cập nhật vào CSDL.
- Hệ thống trả về thông báo xác nhận hoặc thông báo lỗi.

Xóa Sản phẩm:

Admin chọn xóa một sản phẩm.

- Hệ thống kiểm tra xem sản phẩm có trong đơn đặt hàng nào đó không.

- Nếu không có, hệ thống xóa thông tin sản phẩm khỏi CSDL.
- Nếu có, hệ thống trả về thông báo lỗi.

Xem Danh mục Sản phẩm:

- Admin yêu cầu hệ thống hiển thị danh mục sản phẩm.
- Hệ thống truy xuất danh mục từ CSDL và hiển thị cho Admin

3.1.3 Quá Trình Giao Dịch

Nhập Sản phẩm:

- Nhân viên bán hàng nhập thông tin sản phẩm vào hệ thống thông qua các cách như tìm kiếm hoặc quét mã vạch.
- Sản phẩm được thêm vào danh sách giao dịch ngay lập tức với các thông tin như số lượng, giá bán, và tổng giá trị được cập nhật tự động.

Hiển thị Danh sách Sản phẩm Đang Mua:

- Danh sách sản phẩm đang được mua sẽ hiển thị trên giao diện người dùng, bao gồm các thông tin như tên sản phẩm, số lượng, giá bán, và tổng cộng.

Tính Toán Tổng Giá Trị Đơn Hàng:

- Hệ thống tự động tính toán và hiển thị tổng giá trị của toàn bộ đơn hàng, bao gồm cả số lượng sản phẩm và các khoản thanh toán.

Nhập Thông Tin Khách Hàng:

- Nhân viên bán hàng nhập thông tin khách hàng, thường là số điện thoại, để hệ thống có thể hiển thị thông tin liên quan nếu khách hàng đã từng mua hàng trước đó.

Hoàn Tất Giao Dịch:

- Nhân viên bán hàng nhập thông tin thanh toán, bao gồm số tiền khách hàng trả và số tiền trả lại (nếu có).
- Hệ thống xác nhận hoàn tất giao dịch và cập nhật dữ liệu vào CSDL.

In Hóa Đơn:

- Sau khi giao dịch hoàn tất, hệ thống tạo hóa đơn và cung cấp tùy chọn in hóa đơn, thường được thực hiện bằng cách tạo một tệp PDF hoặc in trực tiếp.

3.1.4 Reporting and Analytics

Chọn Thời Gian Báo Cáo:

- Người dùng (bạn có thể là quản trị viên hoặc nhân viên bán hàng) chọn khoảng thời gian cụ thể hoặc sử dụng các tùy chọn mặc định như "hôm nay," "hôm qua," "trong 7 ngày qua," "tháng này," và tùy chọn thời gian khác.

Xử Lý và Truy Vấn Dữ Liệu:

- Hệ thống sẽ xử lý yêu cầu và truy vấn dữ liệu từ cơ sở dữ liệu liên quan đến các giao dịch trong khoảng thời gian đã chọn.

Hiển Thị Thông Tin Báo Cáo:

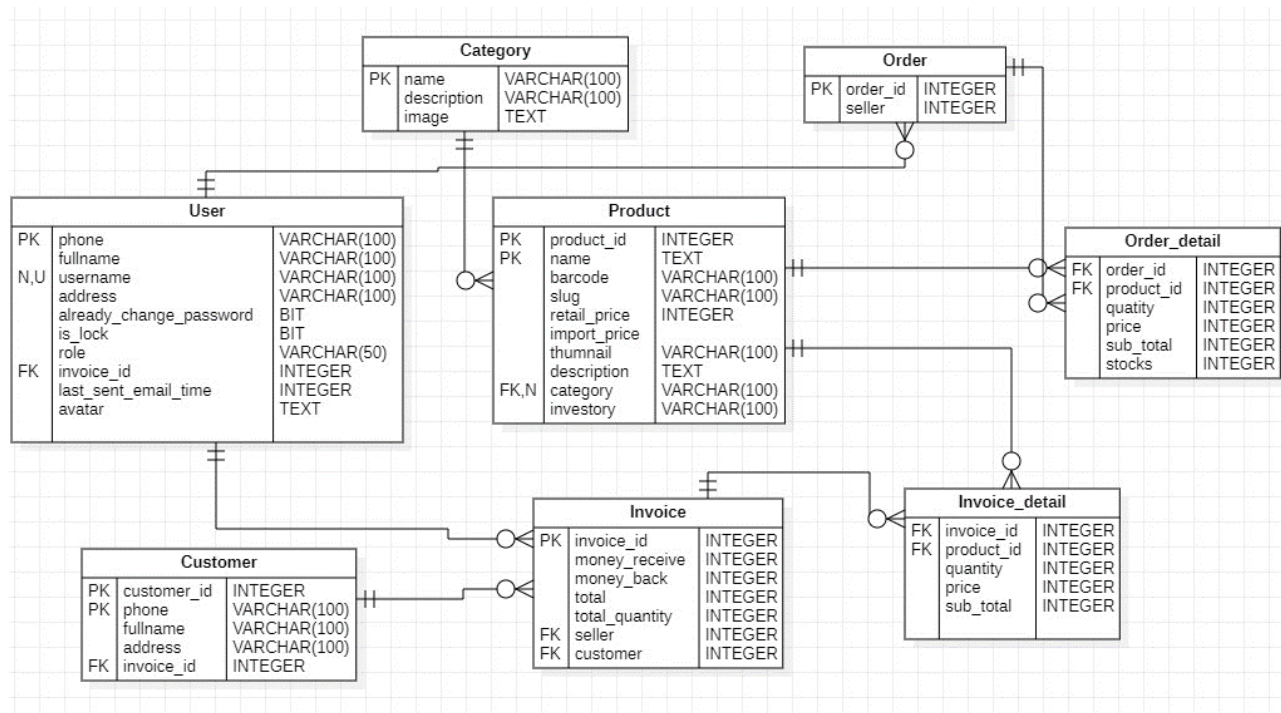
- Dữ liệu báo cáo sẽ được hiển thị trên giao diện người dùng theo các chỉ số kinh doanh như tổng doanh thu, số lượng đơn hàng, số lượng sản phẩm được bán, và các thông tin khác liên quan.

Liệt Kê Đơn Hàng:

- Danh sách các đơn hàng sẽ được liệt kê theo thứ tự thời gian, cung cấp cái nhìn tổng quan về các giao dịch đã diễn ra.

3.4 Sơ đồ ERD và định nghĩa các model

3.1.4 Sơ Đồ ERD

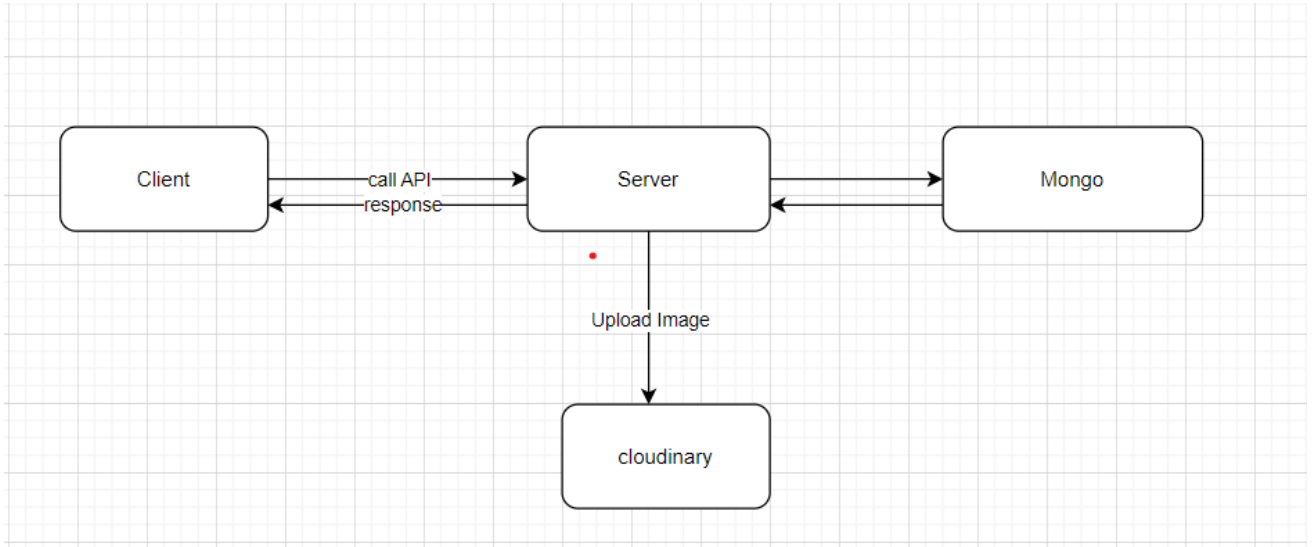


Hình 3.8 Sơ đồ ERD

CHƯƠNG 4 – Hiện Thực Hệ Thống

Chương này nhóm sẽ cho thấy sản phẩm mà nhóm đã hoàn thành được, các phần mà nhóm đã hiện thực bằng ý tưởng ở chương 3

4.1. Tổng quát về ứng dụng



Hình 4.1 Tổng quát ứng dụng

Tổng quát ứng dụng của chúng sẽ có bốn phần chính đó là client, server, database và service cloudinary để lưu hình ảnh khi. Do cloudinary là service bên thứ ba nên chúng không cần hiện thực gì nhiều. Do vậy nhóm chúng em chỉ cần hiện thực 3 phần đó là database, client, server. Database và server do phần code quá nhiều nhóm em quyết định sẽ nói chúng trong video. Ngoài ra phần client em cũng đã đưa ra hình ảnh chương 3

CHƯƠNG 5 – TỔNG KẾT

Chương này nhóm sẽ tóm tắt lại sản phẩm mà nhóm đã hoàn thành được, các vấn đề đã được giải quyết và những hạn chế, thiếu sót nhóm còn cần phải cải thiện.

5.1 Kết quả đạt được

Sau khi cùng nhau làm việc thì sản phẩm cuối cùng của nhóm là ứng dụng POS

CONFIRM BILLING

Customer Name
Huỳnh Nhật Hào

Customer Address
458/63 Huỳnh Tấn Phát Quận 7

Customer Phone
0399331472

Product Items
 Iphone 15 Promax - Quantity: 1 - Price: 43,000.000 - Subtotal: 43,000.000
 Oriental Granite Bacon - Quantity: 1 - Price: 23,073.000 - Subtotal: 23,073.000
 Awesome Fresh Towels - Quantity: 1 - Price: 26,832.000 - Subtotal: 26,832.000

Order Details
 Total Quantity: 3
 Total Cost: 92,905.000
 Order ID: 65753a5abd4786522ebef98
 Seller ID: 65742784667e9807429601ef

Money Receive

Money Back

Complete Purchase

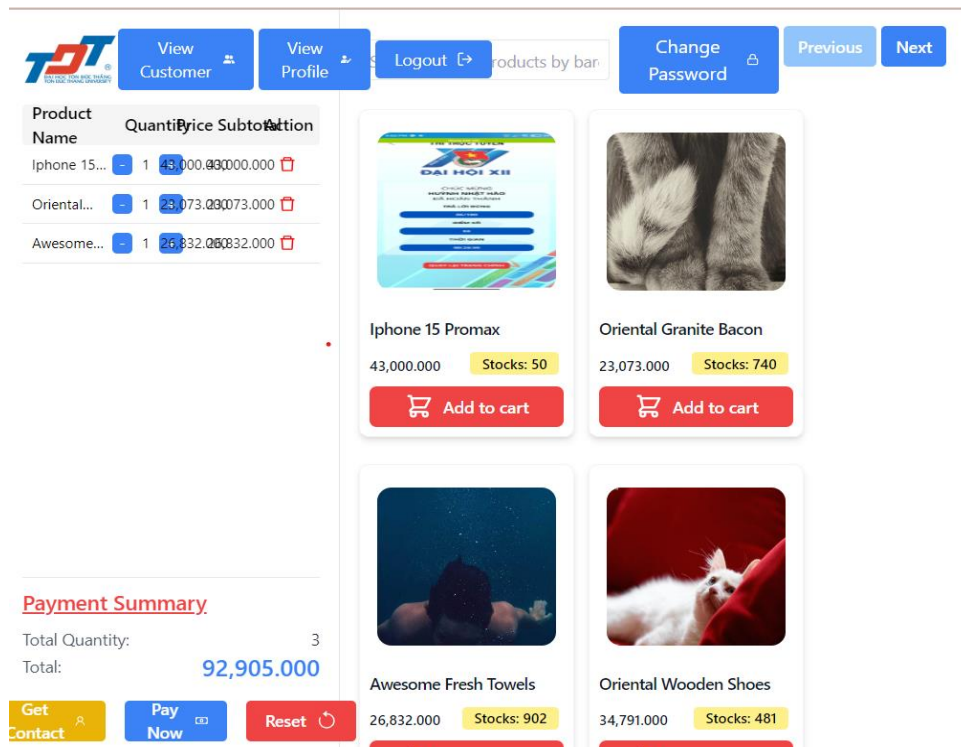
Hình 5.1 Giao diện thanh toán

Hình trên đang mô tả nội dung của một order cụ thể. Ngoài việc thanh toán thì người dùng sẽ xem được hóa đơn sau khi nhấn nút complete purchase bên cạnh đó thầy có thể xem qua video demo đây chỉ là một trong kết quả nhóm em thực hiện được thôi ạ. Ứng dụng được xây dựng bằng ExpressJS và ReactJS nên sẽ được một số lợi thế như là giao diện đẹp và xử lý hiệu ứng tốt, xử lý các thao tác nhanh chóng. Ứng dụng này có thể được sử dụng bởi người dùng có nhu cầu.

Nhìn chung cả nhóm thấy mình đã hiểu thêm về các hoạt động của Express và ReactJs. Ngoài các kiến thức này ra thì nhóm còn phát triển được cho mình khả năng lên kế hoạch, làm việc nhóm và các kỹ năng trong software engineering.

5.2 Những hạn chế cần khắc phục

Một trong những hạn chế đầu tiên mà nhóm cần phải khắc phục đến từ ứng dụng. Tuy ứng dụng lần này được phát triển cũng khá tốt nhưng nhóm nhận thấy mình vẫn hạn chế ở mặt lên ý tưởng cho ứng dụng. Cụ thể thì ứng dụng lần này sẽ không giúp giải quyết các vấn đề thời gian thực trong việc tạo hóa đơn của user và admin tự động cập nhật trong phần quản lý của admin. Thứ hai, nhóm nhận thấy mình vẫn chưa thực sự phát triển khả năng tối ưu các chức năng có liên quan đến xử lý phản hồi. Cụ thể là khi người dùng thực hiện các thao tác thanh toán họ chờ thời gian khoảng 3s tới 4s mới hoàn thành. Theo em thì thời gian như này là quá chậm làm giảm trải nghiệm người dùng. Tiếp theo đó là về mặt giao diện ứng dụng chưa được responsive nên sẽ không hiển thị đẹp trên nhiều loại màn hình nhỏ hơn hoặc lớn hơn.



Hình 5.2 Giao diện chưa responsive

Cuối cùng là về database em cảm nhận do quá trình tạo có nhiều sự gòmrà còn có thể tuổi ưu và hiện tại chưa có back up dữ liệu chưa có chức năng log cho phía server, Ứng dụng chưa đo lường được số lượng người dùng đồng thời. Đặc biệt ngoài ra nhóm chưa thực hiện tạo cho ứng dụng một server đệm để caching các dữ liệu thường xuyên không thay giúp giảm tải cho server thường sẽ sử dụng redis

Tóm tắt lại thì nhóm nhận thấy để tạo một phần mềm chất lượng không phải là việc có thể thực hiện được ngay lập tức, mà đòi hỏi quy trình gồm các bước để mọi người trong nhóm cùng nhau phát triển.

5.3 Hướng phát triển trong tương lai

Sản phẩm lần này được phát triển dựa ý tưởng micro microservices nhưng không đủ do thời gian và nhóm em chưa có kiến thức và phần này. Microservice nghĩa là sẽ chia nhỏ server ra để quản lý và có tính scalale và flexiable tốt hơn ngoài ra nhóm cũng có ý tưởng là mở rộng database ra dùng Mysql để ghi và dùng mongo để đọc dữ liệu và dùng redis để caching dữ liệu. Nếu trong tương lai có khả năng nhóm sẽ tạo thêm Message Queue để giúp ứng dụng hoạt động tốt hơn. Ngoài ra nhóm còn phải phát triển thêm phần giao cho thân thiện với người dùng hơn. Ý tưởng lúc đầu của nhóm được lấy cảm hứng từ ứng dụng này các máy POS trong cửa hàng , tuy nhiên do nhóm vẫn chưa đủ về nhân lực lẫn trí lực nên không thể thực hiện được ý tưởng. Nên bước phát triển hiện tại mà nhóm nhận thấy mình có thể làm là tìm hiểu cách cơ bản nhất. Đồng thời thông qua đó nhóm cũng muốn học tập ở họ cách phát triển một hệ thống và lên kế.

Đây là bước phát triển tiếp theo của nhóm do bản thân từng thành viên nhận thấy thứ làm nên những ứng dụng công nghệ này chính là cách giải quyết một vấn đề theo hệ thống. Cách tạo một hệ thống hiệu quả mà đồng thời có thể hữu dụng cho người khác.