

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC THĂNG LONG**

**A red and blue logo

Description automatically generated**

**CHUYÊN ĐỀ TỐT NGHIỆP**

**QUẢN LÝ NGƯỜI NỘP THUẾ**

**TRÊN SÀN THƯƠNG MẠI ĐIỆN TỬ**

|  |  |
| --- | --- |
| **GIÁO VIÊN HƯỚNG DẪN** | **NHÓM SINH VIÊN THỰC HIỆN** |
| NGUYỄN THỊ HUYỀN CHÂU | A35973 - TRƯƠNG TUẤN HIỆP  A35090 - TÔ ĐỨC DŨNG |

**HÀ NỘI – 2023**

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC THĂNG LONG**

**A red and blue logo

Description automatically generated**

**CHUYÊN ĐỀ TỐT NGHIỆP**

**QUẢN LÝ NGƯỜI NỘP THUẾ**

**TRÊN SÀN THƯƠNG MẠI ĐIỆN TỬ**

|  |  |
| --- | --- |
| **GIÁO VIÊN HƯỚNG DẪN** | **NHÓM SINH VIÊN THỰC HIỆN** |
| NGUYỄN THỊ HUYỀN CHÂU | A35973 - TRƯƠNG TUẤN HIỆP  A35090 - TÔ ĐỨC DŨNG |

**HÀ NỘI – 2023**

**LỜI CẢM ƠN**

Lời đầu tiên chúng em xin được phép gửi lời cảm ơn chân thành nhất tới toàn bộ các thầy cô giáo trong khoa Công nghệ thông tin cũng như các thầy cô giảng dạy trong trường Đại học Thăng Long đã truyền đạt những kiến thức quý báu và bổ ích nhất cho chúng em trong những năm học vừa qua.

Đặc biệt, chúng em xin bày tỏ lòng biết ơn sâu sắc đến thầy Nguyễn Thị Huyền Châu, người đã tận tình hướng dẫn, động viên và trực tiếp hỗ trợ chúng em trong suốt quá trình triển khai và thực hiện chuyên đề tốt nghiệp (CĐTN).

Để có được kết quả như ngày hôm nay, chúng em rất biết ơn thầy cô, gia đình đã động viên, khích lệ, tạo mọi điều kiện thuận lợi nhất trong suốt quá trình học tập cũng như quá trình thực hiện CĐTN này.

Bên cạnh đó, chúng em xin chân thành cám ơn các bạn trong khoa Công nghệ thông tin trường Đại học Thăng Long đã ủng hộ, giúp đỡ, chia sẻ kiến thức, kinh nghiệm và các tài liệu có được cho chúng em trong suốt quá trình nghiên cứu và thực hiện CĐTN.

Vì kiến thức còn hạn chế, trong quá trình triển khai và thực hiện CĐTN này chúng em không tránh khỏi có những sai sót không mong muốn, chính vì vậy chúng em rất mong nhận được những ý kiến đóng góp từ các thầy cô cũng như các bạn trong khoa để hoàn thiện CĐTN này cũng như bổ sung những kiến thức còn thiếu sót.

Cuối cùng chúng em xin kính gửi lời chúc tới toàn thể các thầy cô trong khoa Công nghệ thông tin cũng như các thầy cô giảng dạy trong trường Đại học Thăng Long luôn luôn mạnh khỏe và tràn đầy nhiệt huyết để truyền đạt kiến thức cho các thế hệ mai sau.

*Nhóm em xin chân thành cảm ơn!*

*Hà Nội, ngày 23 tháng 07 năm 2023*

**NHÓM SINH VIÊN**

**LỜI CAM ĐOAN**

Chúng tôi xin cam đoan về số liệu sử dụng trong khóa luận là số liệu thật, kết quả nghiên cứu trình bày trong khóa luận là sản phẩm của chúng tôi, không sao chép, copy của người khác, các nội dung trích dẫn đều ghi đầy đủ thông tin nguồn, tuân thủ qui định của Luật Bản quyền.

Chúng tôi xin chịu trách nhiệm trước nhà trường và pháp luật nếu có bất kỳ hành vi vi phạm pháp luật nào được phát hiện.

**NHÓM SINH VIÊN**

**MỤC LỤC**

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 17](#_Toc152014731)

[1.1. Sơ lược về cơ sở dữ liệu sử dụng 17](#_Toc152014732)

[1.2. Kiến trúc công nghệ chủ đạo. 17](#_Toc152014733)

[1.2.1. Giới thiệu về API 18](#_Toc152014734)

[1.2.2. REST API 20](#_Toc152014735)

[1.3. Mô tả kiến trúc tổng thể 21](#_Toc152014736)

[1.4. Công nghệ và ngôn ngữ sử dụng 21](#_Toc152014737)

[1.4.1. MySQL 22](#_Toc152014738)

[1.4.2. Python 23](#_Toc152014739)

[CHƯƠNG 2. phân tích và đặc tả chức năng 25](#_Toc152014740)

[2.1. Mô tả bài toán 25](#_Toc152014741)

[2.1.1. Thực trạng 25](#_Toc152014742)

[2.1.2. Mô tả hệ thống 25](#_Toc152014743)

[2.2. Xác định yêu cầu 25](#_Toc152014744)

[2.2.1. Xác định và mô tả các tác nhân 25](#_Toc152014745)

[2.2.2. Xác định và mô tả các use case sử dụng 26](#_Toc152014746)

[2.2.3. Biểu đồ Usecase tổng quát 28](#_Toc152014747)

[2.3. Đặc tả chức năng 28](#_Toc152014748)

[2.3.1. Đăng nhập 29](#_Toc152014749)

[2.3.2. Tạo user đăng nhập 30](#_Toc152014750)

[2.3.3. Xem dữ liệu bảng fact 31](#_Toc152014751)

[2.3.4. Xóa dữ liệu 32](#_Toc152014752)

[2.3.5. Tạo biểu đồ 33](#_Toc152014753)

[CHƯƠNG 3. phân tích, thiết kế chức năng và cơ sở dữ liệu 35](#_Toc152014754)

[3.1. Quy tắc nghiệp vụ 35](#_Toc152014755)

[3.2. Các bảng nghiệp vụ 35](#_Toc152014756)

[3.3. Biểu đồ lớp thiết kế 48](#_Toc152014757)

[CHƯƠNG 4. CÀI ĐẶT VÀ TRIỂN KHAI 49](#_Toc152014758)

[4.1. Cài đặt môi trường 49](#_Toc152014759)

[4.1.1. My SQL 49](#_Toc152014760)

[4.1.2. FastAPI 49](#_Toc152014761)

[4.2. Các hàm thủ tục và api 49](#_Toc152014762)

[4.2.1. Thủ tục tổng hợp dự liệu bảng ecom\_sup\_ct 49](#_Toc152014763)

[4.2.2. Thủ tục tổng hợp dự liệu bảng ecom\_san\_ct 63](#_Toc152014764)

[4.2.3. Hàm api đăng nhập 72](#_Toc152014765)

[4.2.4. Hàm api tạo user đăng nhập 73](#_Toc152014766)

[4.2.5. Hàm api lấy dữ liệu ecom\_sup\_fct 74](#_Toc152014767)

[4.2.6. Hàm api lấy dữ liệu ecom\_san\_fct 75](#_Toc152014768)

[4.2.7. Hàm api lấy dữ liệu ecom\_sup\_fct với mã số thuế 76](#_Toc152014769)

[4.2.8. Hàm api xóa dữ liệu ecom\_sup\_fct 77](#_Toc152014770)

[4.2.9. Hàm api xóa dữ liệu ecom\_san\_fct 78](#_Toc152014771)

[4.2.10. Hàm api vẽ biểu đồ tròn thể hiện tỷ lệ loại hình cá nhân hay doanh nghiệp 79](#_Toc152014772)

[4.2.11. Hàm api Thống kê dữu liệu sàn có số giá trị tong\_so\_don\_hang lớn nhất 81](#_Toc152014773)

[4.2.12. Hàm api Thống kê dữu liệu sàn có số giá trị đơn hàng thành công > 60 82](#_Toc152014774)

[4.3. Hướng dẫn sử dụng 83](#_Toc152014775)

[CHƯƠNG 5. Kết luận 84](#_Toc152014776)

[5.1. Kết quả đạt được 84](#_Toc152014777)

[5.1.1. Ứng dụng làm được những gì 84](#_Toc152014778)

[5.1.2. Kiến thức đã học: 84](#_Toc152014779)

[5.1.3. Kiến thức về nghiệp vụ 84](#_Toc152014780)

[5.1.4. Hướng phát triển 84](#_Toc152014781)

[TÀI LIỆU THAM KHẢO 86](#_Toc152014782)

**DANH MỤC MINH HỌA**

[Hình 1.1. Sơ đồ kiến trúc tổng thể 11](#_Toc159786383)

[Hình 1.2.API 13](#_Toc159786384)

[Hình 1.3. Open API 13](#_Toc159786385)

[Hình 1.4. REST API 14](#_Toc159786386)

[Hình 1.5. Công nghệ MySQL 15](#_Toc159786387)

[Hình 1.6. Công nghệ Python 16](#_Toc159786388)

[Hình 2.1. Biểu đồ Usecase tổng quát 21](#_Toc159786389)

[Bảng 2.1. Bảng danh sách các tác nhân của hệ thống 19](#_Toc159786795)

[Bảng 2.2. Bảng danh sách các use case sử dụng 20](#_Toc159786796)

**DANH MỤC THUẬT NGỮ SỬ DỤNG**

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Ý nghĩa** |
| 1 | API | Application Programming Interface |
| 2 | REST API | REpresentational State Transfer Application Programming Interface |
| 3 | OPEN API | Open Application Programming Interface |
| 4 | SQL | Structured Query Language |
| 5 | UC | Use Case |
| 6 | XML | Extensible Markup Language |
| 7 | URL | Uniform Resource Locator |
| 8 | JSON | JavaScript Object Notation |
| 9 | HTTP | Hypertext Transfer Protocol |

**BẢNG PHÂN CÔNG CÔNG VIỆC**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Thành viên** | **Công việc** | **Hoàn thành** |
| 3 | **Trương Tuấn Hiệp** | - Viết tài liệu phân tích  - Viết tài liệu đặc tả  - Viết tài liệu thiết kế CSDL  - Vẽ sơ đồ | 100% |
| 4 | **Tô Đức Dũng** | - Viết lời giới thiệu  - Viết tài liệu triển khai  - Làm slide  - Viết tài liệu công nghệ sử dụng | 100% |

**LỜI GIỚI THIỆU**

Trong thời đại số hóa và sự phát triển mạnh mẽ của thương mại điện tử, quản lý người nộp thuế đã trở thành một yếu tố quan trọng trong hệ thống thuế của nhiều quốc gia. Việc đảm bảo rằng các doanh nghiệp và cá nhân tham gia hoạt động kinh doanh trên sàn thương mại điện tử đăng ký và tuân thủ các quy định thuế là một thách thức quan trọng đối với các cơ quan thuế và người nộp thuế.

Chuyên đề nhằm tìm hiểu và phân tích quá trình dữ liệu kê khai của người nộp thuế trên sàn thương mại điện tử. Ngoài ra chuyên đề này sẽ xem xét các yếu tố quan trọng như việc thu thập thông tin, xác minh dữ liệu . Từ đó để đảm bảo tính hiệu quả, minh bạch, và công bằng trong thu thuế.

Bằng cách nghiên cứu cận thận về quản lý thông tin người nộp thuế trên sàn thương mại điện tử, Chuyên đề này hy vọng sẽ đóng góp kiến thức quý báu cho các quốc gia và tổ chức quản lý thuế trong việc phát triển các chính sách và quy trình hiệu quả để quản lý thuế trong môi trường số hóa ngày càng phức tạp..

Cấu trúc chuyên đề:

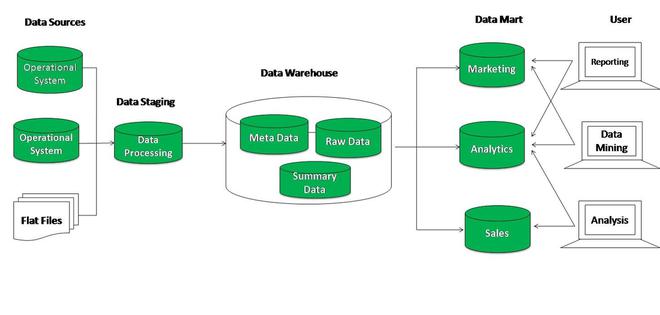
* Chương 1: Cơ sở lý thuyết: các công nghệ, kỹ thuật được áp dụng vào trong ứng dụng;
* Chương 2: Phân tích và đặc tả chức năng: bao gồm đặt vấn đề cùng với mô tả các tác nhân và use case, rồi sau đó đưa ra các đặc tả chức năng;
* Chương 3: Phân tích và thiết kế dữ liệu bao gồm các sơ đồ tuần tự cho chức năng và các lược đồ cho cơ sở dữ liệu;
* Chương 4: Cài đặt và triển khai gồm có mô tả cách thiết lập môi trường, cài đặt ứng dụng và một kịch bản hướng dẫn sử dụng;
* Chương 5: Kết luận đưa ra kết quả đạt được sau quá trình làm, về những điều học được, những thứ còn thiếu sót và những định hướng phát triển trong tương lai.

# CƠ SỞ LÝ THUYẾT

## Sơ lược về cơ sở dữ liệu sử dụng

Những công nghệ mà nhóm em ứng dụng để làm ra sản phẩm bao gồm: FastAPI, Python, MySQL. Là những công nghệ được ứng dụng phổ biến để thiết kế và phát triển phần mềm. Công nghệ này giúp việc thu thập, lưu trữ thông tin người sử dụng một cách khoa học, dễ dàng và chính xác. Tạo ra giao diện dễ sử dụng, thuận tiện và phù hợp với nhiều thiết bị khác nhau, đảm bảo cung cấp các chức năng xem, tìm kiếm và chỉnh sửa khoa học.

## Kiến trúc công nghệ chủ đạo.



Hình .. Sơ đồ kiến trúc tổng thể

Cách thức hoạt động của Data warehouse của sơ đồ:

1. Operational System: Operational System là một phương pháp được sử dụng trong kho dữ liệu để chỉ một hệ thống được sử dụng để xử lý các giao dịch hàng ngày của một tổ chức

2. Flat Files: Flat Files là một hệ thống tệp trong đó dữ liệu giao dịch được lưu trữ và mọi tệp trong hệ thống phải có một tên khác.

3. Metadata: Tập hợp dữ liệu xác định và cung cấp thông tin về dữ liệu khác. Dữ liệu meta được sử dụng trong Kho dữ liệu cho nhiều mục đích khác nhau, bao gồm: Dữ liệu meta tóm tắt thông tin cần thiết về dữ liệu, có thể giúp việc tìm kiếm và làm việc với các trường hợp dữ liệu cụ thể dễ tiếp cận hơn. Siêu dữ liệu được sử dụng để hướng một truy vấn đến nguồn dữ liệu thích hợp nhất.

4. Dữ liệu tóm tắt nhẹ nhàng và cao: Khu vực của kho dữ liệu lưu tất cả các dữ liệu được xác định trước (tổng hợp) nhẹ và cao do người quản lý kho tạo ra. Mục tiêu của thông tin tóm tắt là để tăng tốc hiệu suất truy vấn. Bản ghi tóm tắt được cập nhật liên tục khi thông tin mới được tải vào kho.

5. End-User access Tools: Mục đích chính của kho dữ liệu là cung cấp thông tin cho các nhà quản lý doanh nghiệp để ra quyết định chiến lược. Những khách hàng này tương tác với nhà kho bằng các công cụ truy cập khách hàng cuối.

Ví dụ về một số công cụ truy cập người dùng cuối có thể là:

* Reporting and Query Tools
* Online Analytical Processing Tools
* Data Mining Tools

## Mô tả kiến trúc tổng thể

Hệ thống này được chia làm 2 phần: Client và Server. Mỗi phần đều có những chức năng và đặc điểm riêng.

**Client:**

* Đây là nơi giúp người dùng có thể nhìn thấy, nhằm mục đích tương tác, giao tiếp với Server qua giao diện người dùng.
* Ứng dụng sử dụng: VS Studio Code.
* Thiết bị sử dụng:
* Laptop window: sử dụng để code API.

**Server:**

* Server là nơi trực tiếp xử lý mọi request của client và trả lại những thông tin nếu cần thiết.
* Các endpoints (APIs) của server sẽ được thiết kế theo phong cách REST, nhằm mục đích tạo nơi giao tiếp giữa các những server và giao diện người dùng.
* Công nghệ sử dụng: Python, FastAPI, MySQL.

## Công nghệ và ngôn ngữ sử dụng

### API

#### Khái niệm

A blue and black logo

Description automatically generated

Hình ..API

API (Application Programming Interface) là một giao diện lập trình ứng dụng mà một hệ thống máy tính hay ứng dụng cung cấp để cho phép các yêu cầu dịch vụ có thể được tạo ra từ các chương trình máy tính khác và cho phép dữ liệu có thể được trao đổi qua lại giữa chúng.

OpenAPI cho phép người dùng tích hợp các phần mềm và ứng dụng thông qua nhiều ngôn ngữ lập trình khác nhau. Open API với khả năng xử lý các ngôn ngữ lập trình cũ sẽ giảm bớt rào cản và hạn chế tối đa rủi ro mất dữ liệu từ những ứng dụng không còn được hỗ trợ.

A diagram of a software development process

Description automatically generated

Hình .. Open API

FastAPI là một framework phát triển ứng dụng web nhanh chóng và hiệu quả với Python. Nó được xây dựng dựa trên tiêu chuẩn OpenAPI và ASGI, giúp viết các ứng dụng web chất lượng cao với mã nguồn mở. FastAPI có khả năng xử lý đồng thời, hỗ trợ validation và serialization tự động đối với các dữ liệu đầu vào và đầu ra. Nó cung cấp khả năng tự tạo tài liệu API, sử dụng các công cụ như Swagger UI, và hỗ trợ autocomplete trong các trình biên dịch. FastAPI cũng tương thích với các công nghệ phổ biến khác của Python như Pedantic, SQLAlchemy, OAuth, JWT và nhiều công nghệ khác.

#### Những đặc điểm nổi bật của API

* API sử dụng mã nguồn mở, dùng được với mọi client hỗ trợ XML, JSON.
* API có khả năng đáp ứng đầy đủ các thành phần HTTP: URI, request/response headers, caching, versioning, content format…
* Mô hình web API dùng để hỗ trợ MVC như: unit test, injection, ioc container, model binder, action result, filter, routing, controller. Ngoài ra, nó cũng hỗ trợ RESTful đầy đủ các phương thức như: GET, POST, PUT, DELETE các dữ liệu.

#### REST API

A diagram of a computer process

Description automatically generated

Hình .. REST API

REST API (còn được gọi là RESTful API) là một giao diện lập trình ứng dụng (API) tuân theo các ràng buộc và quy ước kiến trúc REST trong việc giao tiếp giữa client và server.

Trong REST API, chúng ta có hai thành phần chính:

* **API** (Application Programming Interface): Đây là một bộ quy tắc và cơ chế cho phép một ứng dụng hoặc thành phần tương tác với một ứng dụng hoặc thành phần khác. API thường trả về dữ liệu trong các định dạng phổ biến như JSON hoặc XML để ứng dụng có thể sử dụng.
* **REST** (Representational State Transfer): REST là một kiểu kiến trúc cho việc thiết kế API. Nó sử dụng các phương thức HTTP để tạo ra tương tác giữa các máy chủ và máy khách thông qua các URL. REST quy định cách sử dụng các phương thức HTTP và định dạng URL cho ứng dụng web.

REST API cung cấp bốn phương thức chính để thao tác với dữ liệu:

* **GET**: Sử dụng để yêu cầu dữ liệu từ máy chủ.
* **POST**: Sử dụng để gửi dữ liệu từ máy khách đến máy chủ, thường được sử dụng khi bạn muốn thêm dữ liệu mới vào máy chủ.
* **PUT**: Sử dụng để cập nhật hoặc thêm dữ liệu vào thông tin hiện có trên máy chủ.
* **DELETE**: Sử dụng để xóa thông tin hiện có trên máy chủ.

Các phương thức này cho phép tương tác linh hoạt với dữ liệu trên máy chủ thông qua REST API.

### MySQL

A logo with a dolphin

Description automatically generated

Hình .. Công nghệ MySQL

* MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở, được phát triển bởi Oracle Corporation. Nó được sử dụng phổ biến trong việc lưu trữ và truy xuất dữ liệu trong các ứng dụng web. MySQL hỗ trợ ngôn ngữ truy vấn SQL để truy vấn và quản lý cơ sở dữ liệu. Nó cung cấp tính năng bảo mật, kiểm soát đồng thời, và khả năng mở rộng linh hoạt giúp cho việc xử lý và truy xuất dữ liệu trở nên hiệu quả.
* Ưu điểm của Mysql:
* Tốc độ xử lý nhanh: MySQL được thiết kế để tối ưu hiệu suất xử lý truy vấn, cho phép xử lý một lượng lớn dữ liệu một cách nhanh chóng.
* Độ tin cậy cao: MySQL có tính năng khôi phục dữ liệu trong trường hợp xảy ra sự cố như mất điện, crash hệ thống hoặc lỗi phần cứng.
* Hỗ trợ cho nhiều hệ điều hành: MySQL có sẵn cho hầu hết các hệ điều hành phổ biến như Windows, Linux và macOS.
* Bảo mật cao: MySQL cung cấp các tính năng bảo mật như mã hóa dữ liệu, kiểm soát quyền truy cập và giám sát hoạt động của người dùng.
* Dễ sử dụng và linh hoạt: MySQL có giao diện quản lý đơn giản và dễ sử dụng, cho phép người dùng tùy chỉnh cấu hình và thực hiện các truy vấn phức tạp.
* Hỗ trợ cho các ứng dụng phổ biến: MySQL được tích hợp sẵn trong nhiều ngôn ngữ lập trình và các ứng dụng phát triển web như PHP, Python, Java và Ruby.
* Tuy nhiên, cần lưu ý rằng các ưu điểm này có thể thay đổi tuỳ theo mục đích sử dụng và môi trường triển khai của từng dự án.

### Python

A yellow and grey logo

Description automatically generated

Hình .. Công nghệ Python

Python là một ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học. Các nhà phát triển sử dụng Python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau. Phần mềm Python được tải xuống miễn phí, tích hợp tốt với tất cả các loại hệ thống và tăng tốc độ phát triển.

* Lợi ích python mang lại:
* Dễ dàng đọc và hiểu một chương trình Python vì ngôn ngữ này có cú pháp cơ bản giống tiếng Anh.
* Python giúp cải thiện năng suất làm việc của các nhà phát triển vì so với những ngôn ngữ khác, họ có thể sử dụng ít dòng mã hơn để viết một chương trình Python.
* Python có một thư viện tiêu chuẩn lớn, chứa nhiều dòng mã có thể tái sử dụng cho hầu hết mọi tác vụ.
* Cộng đồng Python tích cực hoạt động bao gồm hàng triệu nhà phát triển nhiệt tình hỗ trợ trên toàn thế giới.
* Trên Internet có rất nhiều tài nguyên hữu ích nếu bạn muốn học Python. Ví dụ: bạn có thể dễ dàng tìm thấy video, chỉ dẫn, tài liệu và hướng dẫn dành cho nhà phát triển.
* Python có thể được sử dụng trên nhiều hệ điều hành máy tính khác nhau, chẳng hạn như Windows, macOS, Linux và Unix.
* Đặc điểm của python:
* Là một ngôn ngữ thông dịch
* Là một ngôn ngữ dễ sử dụng
* Là một ngôn ngữ linh hoạt
* Là một ngôn ngữ lập trình hướng đối tượng
* Là một ngôn ngữ cấp cao

# phân tích và đặc tả chức năng

## Mô tả bài toán

### Thực trạng

Sự bùng nổ của sàn thương mại điện tử (e-commerce) hiện nay đã đặt ra nhiều thách thức cho quản lý thuế. E-commerce đã trở thành một phần quan trọng của nền kinh tế toàn cầu và đang ngày càng phát triển mạnh mẽ. Điều này đã tạo nên một thực trạng phức tạp trong việc thu thuế và quản lý thuế trên các giao dịch trực tuyến.

Nhằm tăng cường công tác quản lý thuế đối với thương mại điện tử, ngành thuế đã có các chỉ đạo triển khai trên toàn quốc. Do đó, các cơ quan thuế địa phương đã có yêu cầu các doanh nghiệp là chủ sở hữu sàn giao dịch thương mại điện tử (sau đây gọi tắt là Công ty) cung cấp thông tin theo các yêu cầu và hình thức cung cấp thông tin không thống nhất, do đó sẽ không thuận lợi cho việc cung cấp thông tin của các Sàn.

Để tạo điều kiện thuận lợi cho các Sàn trong việc cung cấp thông tin cho cơ quan thuế theo quy định tại khoản 1 Điều 29 Luật Quản lý thuế số 38/2019/QH14; khoản 5 Điều 1 Thông tư số 100/2021/TT-BTC ngày 15/11/2021 của Bộ Tài chính.

Việc cung cấp thông tin về các tổ chức, cá nhân cung cấp hàng hóa, dịch vụ thông qua sàn giao dịch thương mại điện tử do Công ty sở hữu gửi về Tổng cục Thuế bằng hình thức điện tử qua Cổng thông tin điện tử của Tổng cục Thuế.

Để giải quyết thực trạng này, chúng em đã tìm hiểu và thực hiện đề tài “Quản lý người nộp thuế trên sàn thương mại điện tử”. Từ đó đưa ra giải pháp giúp nhà nước có thể quản lý hoạt động kinh doanh trên sàn thương mại một cách dễ dàng hơn.

### Mô tả hệ thống

Hệ thống quản lý người nộp thuế hoạt động trên sàn thương mại điện giúp cán bộ thuế có thể dễ dàng quản lý thông tin dễ dàng hơn. Cán bộ thuế khai thác dữ liệu theo nhu cầu từ các văn bản báo cáo đã ban hành. Người quản trị cơ sở dữ liệu có nhiệm tổng hợp dữ liệu theo yêu cầu của báo cáo.

## Xác định yêu cầu

### Xác định và mô tả các tác nhân

| **STT** | **Tác nhân** | **Mô tả** |
| --- | --- | --- |
| 1 | Người quản lý | Đây là actor đại diện cho cán bộ thuế người có nhu cầu sử dụng ứng dụng để khai thác dữ liệu, …. |

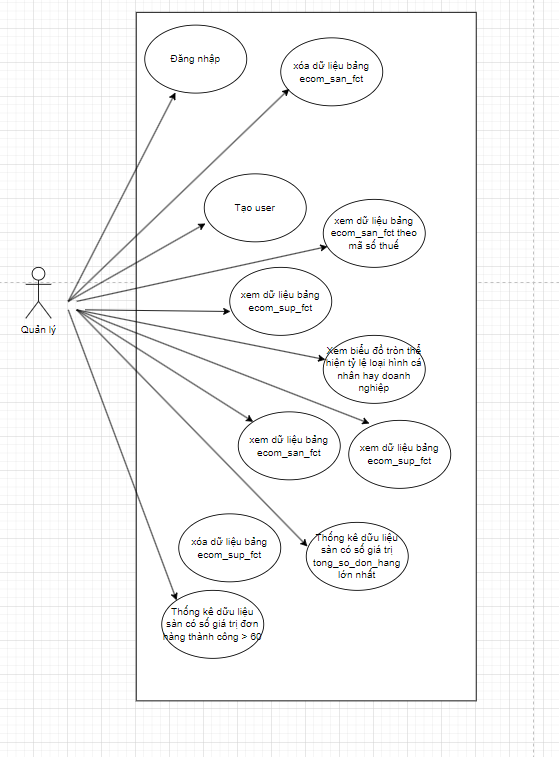
Bảng .. Bảng danh sách các tác nhân của hệ thống

### Xác định và mô tả các use case sử dụng

| **Use Case** | **Chức năng** |
| --- | --- |
| UC#01: Đăng nhập | Người dùng thực hiện chức năng đăng nhập để sử dụng hệ thống. |
| UC#02: Tạo user | Chức năng này cho quản lý thêm user đăng nhập vào hệ thống |
| UC#03: xem dữ liệu bảng ecom\_sup\_fct | Chức năng này cho phép người dùng xem dữ liệu trong bảng ecom\_sup\_fct |
| UC#04: xem dữ liệu bảng ecom\_san\_fct | Chức năng này cho phép người dùng xem dữ liệu trong bảng ecom\_san\_fct |
| UC#05: xóa dữ liệu bảng ecom\_sup\_fct | Chức năng này cho phép người dùng xóa dữ liệu trong abnrg ecom\_sup\_fct theo kỳ |
| UC#06: xóa dữ liệu bảng ecom\_sup\_fct | Chức năng này cho phép người dùng xóa dữ liệu trong abnrg ecom\_san\_fct theo kỳ |
| UC#07: xem dữ liệu bảng ecom\_san\_fct theo mã số thuế | Chức năng này cho phép người dùng xem dữ liệu trong bảng ecom\_sup\_fct theo mst |
| UC#8: Xem biểu đồ tròn thể hiện tỷ lệ loại hình cá nhân hay doanh nghiệp | Chức năng này cho phép người dùng xem được biểu đồ tròn thể hiện tỷ lệ loại hình cá nhân hay doanh nghiệp theo kỳ kê khai |
| UC#9: Thống kê dữu liệu sàn có số giá trị tong\_so\_don\_hang lớn nhất | Chức năng này cho phép người dùng xem được dữ liệu của sàn có có số giá trị tong\_so\_don\_hang lớn nhất |
| UC#10: Thống kê dữu liệu sàn có số giá trị đơn hàng thành công > 60 | Chức năng này cho phép người dùng xem được dữu liệu sàn có số giá trị đơn hàng thành công > 60% |

Bảng .. Bảng danh sách các use case sử dụng

### Biểu đồ Usecase tổng quát



Hình .. Biểu đồ Usecase tổng quát

## Đặc tả chức năng

### Đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#01** | | **Đăng nhập** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng đăng nhập vào tài khoản | |
| **Tác nhân** | | Nguời dùng | |
| **Tiền điều kiện** | | Người dùng đăng nhập vào hệ thống | |
| **Hậu điều kiện** | **Thành công** | Đăng nhập thành công | |
| **Lỗi** | Đăng nhập không thành công | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn Đăng nhập để sử dụng các chức năng của ứng dụng.   * Hệ thống hiển thị bảng thông tin Đăng nhập gồm: Tên đăng nhập, mật khẩu * Nhập tên đăng nhập, mật khẩu * Nhấn nút Đăng nhập * Sau khi đăng nhập thành công, hệ thống hiển thị tài khoản của người dùng | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Nếu người dùng nhập sai tên mật khẩu: hệ thống sẽ hiển thị đăng nhập không thành công. * Nếu người dùng không nhập tên đăng nhập hoặc mật khẩu: hệ thống sẽ yêu cầu điền đầy đủ thông tin. | | | |
| **Giao diện minh họa** | | | |
|  | | | |

### Tạo user đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#02** | | **Tạo user đăng nhập** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép quản lý tạo thêm user đăng nhập | |
| **Tác nhân** | | Quản lý | |
| **Tiền điều kiện** | | Người dùng đăng nhập vào hệ thống | |
| **Hậu điều kiện** | **Thành công** |  | |
| **Lỗi** |  | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn Đăng xuất khỏi tài khoản   * Vào mục Tài khoản * Chọn dòng Đăng xuất * Hiện thông báo xác nhận đăng xuất * Nhấn nút Đăng xuất * Sau khi đăng xuất thành công, hệ thống đăng xuất khỏi tài khoản của người dùng. | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| **Giao diện minh họa** | | | |
|  | | | |

### Xem dữ liệu bảng fact

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#04** | | **Xem dữ diệu trong bảng fact** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng xem dữ liệu | |
| **Tác nhân** | | Quản lý | |
| **Tiền điều kiện** | | chọn các api get dữ liệu | |
| **Hậu điều kiện** | **Thành công** | Hiển thị bảng dữ liệu | |
| **Lỗi** | Không hiện dữ liệu | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn xem dữ liệu bảng fact   * Chọn kỳ * Chọn cơ quan thuế | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Hệ thống không hiển thị dữ liệu * Hệ thống báo lỗi câu sql | | | |
| **Giao diện minh họa** | | | |
|  | | | |

### Xóa dữ liệu

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#03** | | **Xóa dữ liệu** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Xóa dữ liệu của các kỳ báo cáo | |
| **Tác nhân** | | Quản lý | |
| **Tiền điều kiện** | | Đăng nhập vào hệ thống | |
| **Hậu điều kiện** | **Thành công** | Dữ liệu đã được xóa thành công | |
| **Lỗi** | Không tìm thấy dữ liệu để xóa | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn xóa dữ liệu   * Chọn kỳ kê khai muốn xóa | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Hệ thống báo không có dữ liệu để xóa | | | |
| **Giao diện minh họa** | | | |
|  | | | |

### Tạo biểu đồ

|  |  |  |  |
| --- | --- | --- | --- |
| **UC#05** | | **Tạo biểu đồ** | **Độ phức tạp: đơn giản** |
| **Mô tả** | | Cho phép người dùng tạo biểu đồ có sẵn trong api | |
| **Tác nhân** | | Sinh viên | |
| **Tiền điều kiện** | | chọn api in biểu đồ | |
| **Hậu điều kiện** | **Thành công** | Hiển thị biểu đồ | |
| **Lỗi** | Không hiện biểu đồ | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| Use case bắt đầu khi Tác nhân muốn in biểu đồ   * Chọn kỳ kê khai * In biểu đồ | | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | | |
| * Hệ thống không hiển thị biểu đồ và báo lỗi | | | |
| **Giao diện minh họa** | | | |
|  | | | |

# phân tích, thiết kế chức năng và cơ sở dữ liệu

## Quy tắc nghiệp vụ

Quy trình nghiệp vụ:

* Các sàn thương mại điện tử gửi các thông tin của sàn mình đã được yêu cầu trước dựa trên công văn 3434/TCT-DNNCN theo dạng mã hóa gói tin.
* Tổng cục thuế yêu cầu tổng hợp báo cáo theo các biểu mẫu đã cho trước. Từ các thông tin từ nghiệp vụ phân tích người quản trị dữ liệu tiến hành sàng lọc dữ liệu và tổng hợp lại theo yêu cầu.
* Sau khi tổng hợp dữ liệu từ các bảng nghiệp vụ. Người quản trị tạo một bảng phân tích mới cho cán bộ thuế thực hiện khai thác dữ liệu.

Nghiệp vụ dữ liệu quản lý:

* Mỗi sàn thương mại trong hệ thống được định danh bằng một mã gói tin duy nhất. Một sàn thì có nhiều nhà cung cấp, một nhà cung cấp có thể kinh doanh trên nhiều sàn.
* Mỗi nhà cung cấp sẽ cung cấp các thông tin cơ bản chung gồm email, số điện thoại, là cá nhân hay tổ chức (công ty), địa chỉ chi tiết, thành phố, quốc gia, và các thông tin liên quan khác và các thông tin khác.
* Mỗi nhà cung cấp khai báo một mã số thuế duy nhất để đảm bảo tính bảo mật và quản lý dữ liệu gồm tên, mã số thuế, địa chỉ nộp thuế, căn cước, hiệu lực.
* Nhà cung cấp khai báo thêm mặt hàng kinh doanh và tài khoản thanh toán trên sàn
* Một chi cục thuế quản lý nhiều sàn thương mại điện tử khác nhau. Một cục thuế quản lý nhiều chi cục.
* Nhà cung cấp hoạt động trên sàn thương mại điện tử được quản lý bởi chi cục địa phương sinh sống.

## Các bảng nghiệp vụ

### Bảng: NHA\_CUNGCAP

* Mục đích: Bảng dữ liệu chi tiết nhà cung cấp trên sàn thương mại điện tử
* Các trường trong bảng (trường thêm mới):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
|  | MSG\_ID | VARCHAR(13) | Mã gói tin |  |
|  | MAQUANLYTRENSAN | VARCHAR(50) | Mã quản lý trên sàn |  |
|  | TENNHACUNGCAP | VARCHAR(168) | Tên nhà cung cấp |  |
|  | MALOAIHINH | INTEGER | Mã loại hình |  |
|  | TENLOAIHINH | VARCHAR(47) | Tên loại hình |  |
|  | MA\_LOAIGIAYTO | VARCHAR(5) | Mã loại giấy tờ |  |
|  | TEN\_LOAIGIAYTO | VARCHAR(8) | Tên loại giấy tờ |  |
|  | SO | VARCHAR(18) | Số giấy tờ tùy thân |  |
|  | SO\_GPKD | VARCHAR(44) | Số giấy phép kinh doanh |  |
|  | MST | VARCHAR(14) | Mã số thuế |  |
|  | TONG | VARCHAR(15) | Tổng doanh thu |  |
|  | CACKHOANGIAMTRU | VARCHAR(14) | Các khoản khuyến mãi |  |
|  | TONG\_DONHANGTHANHCONG | VARCHAR(13) | Tổng đơn hàng thành công |  |
|  | TONG\_DONHANGKHONGTHANHCONG | VARCHAR(7) | Tổng đơn hàng không thành công |  |
|  | MA\_TRANGTHAI | INTEGER | Mã trạng thái |  |
|  | TEN\_TRANGTHAI | VARCHAR(8) | Tên trạng thái |  |

### Bảng: SAN\_TMDT

* Mục đích: Bảng dữ liệu thông tin về sàn thương mại điện tử

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
|  | MSG\_ID | VARCHAR(13) | Mã gói tin |  |
|  | KIEUKY | VARCHAR(1) | Kiểu kỳ |  |
|  | KYKKHAI | VARCHAR(6) | Kỳ kê khai |  |
|  | KYKKHAITUNGAY | VARCHAR(19) | Kỳ khai từ ngày |  |
|  | KYKKHAIDENNGAY | VARCHAR(19) | Kỳ khai đến ngày |  |
|  | MACQTNOINOP | INTEGER | Mã cơ quan thuế nơi nộp |  |
|  | TENCQTNOINOP | VARCHAR(61) | Tên cơ quan thuế nơi nộp |  |
|  | NGAYLAPTKHAI | VARCHAR(19) | Ngày lập tờ khai |  |
|  | NGANHNGHEKD | VARCHAR(191) | Ngành nghề kinh doanh |  |
|  | MST | VARCHAR(14) | Mã số thuế |  |
|  | TENNNT | VARCHAR(74) | Tên chủ sở hữu sàn |  |
|  | DCHINNT | VARCHAR(154) | Địa chỉ |  |
|  | PHUONGXA | VARCHAR(30) | Phường xã |  |
|  | MAHUYENNNT | INTEGER | Mã huyện |  |
|  | TENHUYENNNT | VARCHAR(22) | Tên Huyện |  |
|  | MATINHNNT | INTEGER | Mã tỉnh |  |
|  | TENTINHNNT | VARCHAR(21) | Tên tỉnh |  |
|  | DTHOAINNT | VARCHAR(20) | Số điện thoại |  |
|  | FAXNNT | VARCHAR(17) | Fax |  |
|  | EMAILNNT | VARCHAR(28) | Email |  |
|  | NGAY\_NOP\_TK | VARCHAR(19) | Ngày nộp tờ khai |  |

### Bảng: NGUOINOPTHUE

* Mục đích: Bảng thông tin người nộp thuế

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
|  | ID | NUMBER(19) | ID người nộp thuế |  |
|  | TIN | VARCHAR2(60 BYTE) | Mã số thuế |  |
|  | TAXO | VARCHAR2(36 BYTE) | Mã chi cục quản lý |  |
|  | STATUS | VARCHAR2(6 BYTE) | Trạng thái |  |
|  | NAME | VARCHAR2(300 BYTE) | Tên người nộp thuế |  |
|  | ONAME | VARCHAR2(300 BYTE) | Tên tổ thức |  |
|  | NAME\_FIRST | VARCHAR2(300 BYTE) | Tên Họ |  |
|  | IDENTIFICATION | VARCHAR2(180 BYTE) | Số giấy tờ tùy thân |  |
|  | IND | VARCHAR2(1 BYTE) | Hiệu lực |  |

### Bảng: COQUANTHUE

* Mục đích: Bảng dữ liệu thông tin cơ quan thuế

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
|  | MA\_CQT | VARCHAR(13) | Mã chi cục thuế |  |
|  | TEN\_CQT\_NGAN | VARCHAR(50) | Tên cơ quan thuế viết tắt |  |
|  | TEN\_CQT\_DAI | VARCHAR(168) | Tên cơ quan thế |  |
|  | MA\_CHA\_4 | INTEGER | Mã cục thuế |  |
|  | MA\_QLT | VARCHAR(47) | Mã quản lý thuế |  |

### Bảng: TAIKHOAN

* Mục đích: Bảng dữ liệu chi tiết thông tin tài khoản nhà cung cấp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
|  | MSG\_ID | VARCHAR(13) | Mã gói tin |  |
|  | MAQUANLYTRENSAN | VARCHAR(50) | Mã quản lý trên sàn |  |
|  | SOTAIKHOAN | VARCHAR(43) | Mã ngân hàng |  |
|  | CHUTAIKHOAN | VARCHAR(78) | Chủ tài khoản |  |
|  | MA\_NGANHANG | VARCHAR(18) | Mã ngân hàng |  |
|  | TEN\_NGANHANG | VARCHAR(93) | Tên ngân hàng |  |
|  | MA\_TRANGTHAI | INTEGER | Mã trạng thái |  |
|  | TEN\_TRANGTHAI | VARCHAR(8) | Tên trạng thái |  |
|  | IS\_TRONGNUOC | BIT | Mã xác định tài khoản nước ngoài |  |

### Bảng: HANG\_HOA

* Mục đích: Bảng dữ liệu chi tiết hàng hóa của nhà cung cấp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
|  | MSG\_ID | VARCHAR(13) | Mã gói tin |  |
|  | MAQUANLYTRENSAN | VARCHAR(50) | Mã quản lý trên sàn |  |
|  | MA\_NGANHHANG | VARCHAR(168) | Mã ngành hàng |  |
|  | LOAI\_HANGHOA | INTEGER | Loại hàng hóa |  |
|  | MA\_TRANGTHAI | VARCHAR(47) | Mã trạng thái |  |
|  | TEN\_TRANGTHAI | VARCHAR(5) | Tên trạng thái |  |

### Bảng: THONGTIN

* Mục đích: Bảng dữ liệu chi tiết nhà cung cấp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
|  | MSG\_ID | VARCHAR(13) | Mã gói tin |  |
|  | MAQUANLYTRENSAN | VARCHAR(50) | Mã quản lý trên sàn |  |
|  | MALOAIHINH | INTEGER | Mã loại hình |  |
|  | TENLOAIHINH | VARCHAR(47) | Tên loại hình |  |
|  | EMAIL | VARCHAR(200) | Email |  |
|  | SODIENTHOAI | VARCHAR(48) | Số điện thoại |  |
|  | TRUSO\_QUOCGIA\_MA | VARCHAR(2) | Mã trụ sở quốc gia |  |
|  | TRUSO\_QUOCGIA\_TEN | VARCHAR(11) | Tên trụ sở quốc gia |  |
|  | TRUSO\_TINH\_TP\_MA | INTEGER | Mã trụ sở tỉnh |  |
|  | TRUSO\_TINH\_TP\_TEN | VARCHAR(18) | Tên trụ sở tỉnh |  |
|  | TRUSO\_QUAN\_HUYEN\_MA | INTEGER | Mã trụ sở quận huyện |  |
|  | TRUSO\_QUAN\_HUYEN\_TEN | VARCHAR(29) | Tên trụ sở quận huyện |  |
|  | TRUSO\_PHUONG\_XA\_MA | INTEGER | Mã trụ sở phường xã |  |
|  | TRUSO\_PHUONG\_XA\_TEN | VARCHAR(44) | Tên trụ sở phường xã |  |
|  | TRUSO\_SONHA | VARCHAR(200) | Số nhà trụ sở |  |
|  | THUONGTRU\_QUOCGIA\_MA | VARCHAR(2) | Mã thường trú quốc gia |  |
|  | THUONGTRU\_QUOCGIA\_TEN | VARCHAR(11) | Tên thường trú quốc gia |  |
|  | THUONGTRU\_TINH\_TP\_MA | INTEGER | Mã thường trú tỉnh |  |
|  | THUONGTRU\_TINH\_TP\_TEN | VARCHAR(18) | Tên thường trú tỉnh |  |
|  | THUONGTRU\_QUAN\_HUYEN\_MA | INTEGER | Mã thường trú quận huyện |  |
|  | THUONGTRU\_QUAN\_HUYEN\_TEN | VARCHAR(29) | Tên thường trú quận huyện |  |
|  | THUONGTRU\_PHUONG\_XA\_MA | INTEGER | Mã thường trú phường xã |  |
|  | THUONGTRU\_PHUONG\_XA\_TEN | VARCHAR(44) | Tên thường trú phường xã |  |
|  | THUONGTRU\_SONHA | VARCHAR(169) | Số nhà thường trú |  |
|  | MA\_TRANGTHAI | VARCHAR(8) | Mã trạng thái |  |
|  | TEN\_TRANGTHAI | VARCHAR(8) | Tên trạng thái |  |
|  | IS\_TRONGNUOC | BIT | Mã xác định nhà cung cấp nước ngoài |  |

### Bảng: ECOM\_SUP\_FACT

* Mục đích: Bảng dữ liệu tổng hợp cho báo cáo

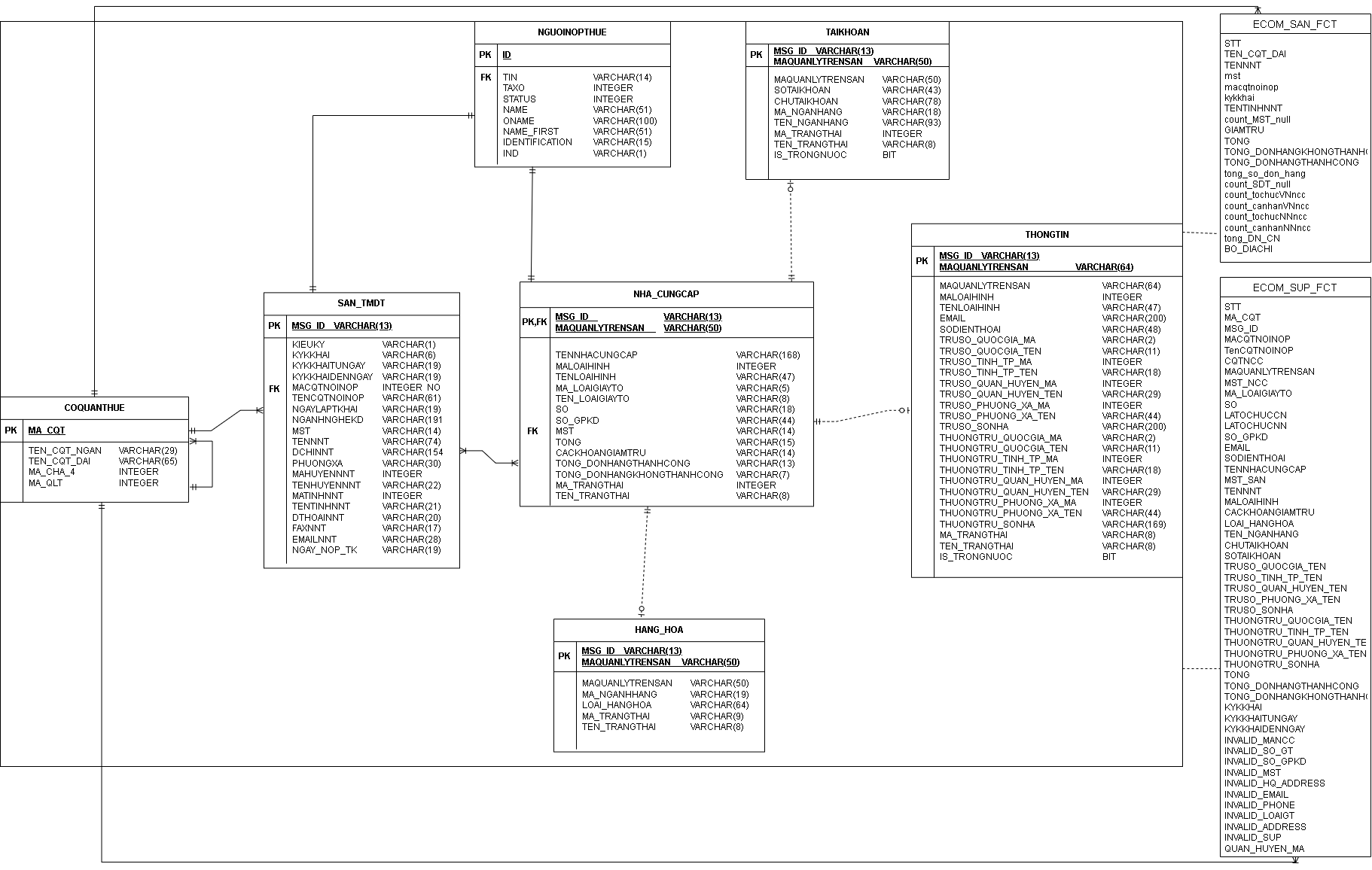
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
| --- | --- | --- | --- | --- |
| 1 | STT | NUMBER | Số thứ tự |  |
| 2 | MA\_CQT | VARCHAR | Mã cơ quan thuế |  |
| 3 | MSG\_ID | VARCHAR | Mã gói tin |  |
| 4 | MACQTNOINOP | VARCHAR | Mã cơ quan thuế nơi nộp |  |
| 5 | MATENCQTNOINOP | VARCHAR | Tên cơ quan thuế nơi nộp |  |
| 6 | MACQTNCC | VARCHAR | Mã cơ quan thuế nhà cung cấp |  |
| 7 | MAQUANLYTRENSAN | VARCHAR | Mã quản lý trên sàn |  |
| 8 | MST\_NCC | VARCHAR | Mã số thuế nhà cung cấp |  |
| 9 | MA\_LOAIGIAYTO | VARCHAR | Mã loại giấy tờ tùy thân |  |
| 10 | SO | VARCHAR | Số giấy tờ tùy thân |  |
| 11 | LATOCHUCCN | VARCHAR | Nhà cung cấp là tổ chức cá nhân |  |
| 12 | LATOCHUCNN | VARCHAR | Nhà cung cấp là tổ chức nước ngoài |  |
| 13 | SO\_GPKD | VARCHAR | Số giấy phép kinh doanh |  |
| 14 | EMAIL | VARCHAR | Email |  |
| 15 | SODIENTHOAI | VARCHAR | Số điện thoại |  |
| 16 | TENNHACUNGCAP | VARCHAR | Tên nhà cung cấp |  |
| 17 | MST\_SAN | VARCHAR | Mã số thuế sàn |  |
| 18 | TENNNT | VARCHAR | Tên người nộp thuế |  |
| 19 | MALOAIHINH | VARCHAR | Mã loại hình |  |
| 20 | CACKHOANGIAMTRU | NUMBER | Các khoản giảm trừ |  |
| 21 | LOAI\_HANGHOA | VARCHAR | Loại hàng hóa |  |
| 22 | TEN\_NGANHANG | VARCHAR | Tên ngân hàng |  |
| 23 | CHUTAIKHOAN | VARCHAR | Chủ tài khoản |  |
| 24 | SOTAIKHOAN | VARCHAR | Số tài khoản |  |
| 25 | TRUSO\_QUOCGIA\_TEN | VARCHAR | Tên quốc gia của trụ sở |  |
| 26 | TRUSO\_TINH\_TP\_TEN | VARCHAR | Tỉnh/Thành phố của trụ sở |  |
| 27 | TRUSO\_QUAN\_HUYEN\_TEN | VARCHAR | Quận/Huyện của trụ sở |  |
| 28 | TRUSO\_PHUONG\_XA\_TEN | VARCHAR | Phường/Xã của trụ sở |  |
| 29 | TRUSO\_SONHA | VARCHAR | Số nhà trụ sở |  |
| 30 | THUONGTRU\_QUOCGIA\_TEN | VARCHAR | Quốc gia thường trú |  |
| 31 | THUONGTRU\_TINH\_TP\_TEN | VARCHAR | Tỉnh/Thành phố thường trú |  |
| 32 | THUONGTRU\_QUAN\_HUYEN\_TEN | VARCHAR | Quận/Huyện thường trú |  |
| 33 | THUONGTRU\_PHUONG\_XA\_TEN | VARCHAR | Phường/Xã thường trú |  |
| 34 | THUONGTRU\_SONHA | VARCHAR | Số nhà thường trú |  |
| 35 | TONG | NUMBER | Tổng số tiền |  |
| 36 | TONG\_DONHANGTHANHCONG | NUMBER | Tổng đơn hàng thành công |  |
| 37 | TONG\_DONHANGKHONGTHANHCONG | NUMBER | Tổng đơn hàng không thành công |  |
| 38 | KYKKHAI | VARCHAR | Kỳ kê khai |  |
| 39 | KYKKHAITUNGAY | VARCHAR | Kỳ kê khai từ ngày |  |
| 40 | KYKKHAIDENNGAY | VARCHAR | Kỳ kê khai đến ngày |  |
| 41 | INVALID\_MANCC | NUMBER | Mã nhà cung cấp không hợp lệ |  |
| 42 | INVALID\_SO\_GT | NUMBER | Số giấy tờ không hợp lệ |  |
| 43 | INVALID\_SO\_GPKD | NUMBER | Số giấy phép kinh doanh không hợp lệ |  |
| 44 | INVALID\_MST | NUMBER | Mã số thuế không hợp lệ |  |
| 45 | INVALID\_HQ\_ADDRESS | NUMBER | Địa chỉ trụ sở không hợp lệ |  |
| 46 | INVALID\_EMAIL | NUMBER | Email không hợp lệ |  |
| 47 | INVALID\_PHONE | NUMBER | Số điện thoại không hợp lệ |  |
| 48 | INVALID\_LOAIGT | NUMBER | Loại giấy từ không hợp lệ |  |
| 49 | INVALID\_ADDRESS | NUMBER | Địa chỉ thường trú không hợp lệ |  |
| 50 | INVALID\_SUP | NUMBER | Thông tin nhà cung cấp không hợp lệ |  |

### Bảng: ECOM\_SAN\_FACT

* Mục đích: Bảng dữ liệu tổng hợp cho báo cáo

| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ghi chú** |
| --- | --- | --- | --- | --- |
| 1 | STT | NUMBER | Số thứ tự |  |
| 2 | TEN\_CQT\_DAI | VARCHAR | Tên cơ quan thuế |  |
| 3 | TENNNT | VARCHAR | Tên chủ sở hữu sàn |  |
| 4 | MST | VARCHAR | Mã số thuế |  |
| 5 | MACQTNOINOP | VARCHAR | Mã cơ quan thuế nhà cung cấp |  |
| 6 | KYKKHAI | VARCHAR | Kỳ kê khai |  |
| 7 | TENTINHNNT | VARCHAR | Mã quản lý trên sàn |  |
| 8 | COUNT\_MST\_NULL | NUMBER | Đếm số lượng nhà cung cấp trên sàn không có mã số thuế |  |
| 9 | GIAMTRU | NUMBER | Tổng khoản khuyến mại |  |
| 10 | TONG | NUMBER | Tổng doanh thu |  |
| 11 | TONG\_DONHANGKHONGTHANHCONG | NUMBER | Số lượng đơn hàng không thành công |  |
| 12 | TONG\_DONHANGTHANHCONG | NUMBER | Số lượng đơn hàng thành công |  |
| 13 | TONG\_SO\_DON\_HANG | NUMBER | Tổng số đơn hàng |  |
| 14 | COUNT\_SDT\_NULL | NUMBER | Đếm số lượng nhà cung cấp trên sàn không có số điện thoại |  |
| 15 | COUNT\_TOCHUCVNNCC | NUMBER | Đếm số lượng nhà cung cấp trên sàn là tổ chức ở Việt Nam |  |
| 16 | COUNT\_CANHANVNNCC | NUMBER | Đếm số lượng nhà cung cấp trên sàn là cá nhân ở Việt Nam |  |
| 17 | COUNT\_TOCHUCNNNCC | NUMBER | Đếm số lượng nhà cung cấp trên sàn là tổ chức nước ngoài |  |
| 18 | COUNT\_CANHANNNNCC | NUMBER | Đếm số lượng nhà cung cấp trên sàn là cá nhân nước ngoài |  |
| 19 | TONG\_DN\_CN | NUMBER | Tổng số lượng cá nhân và tổ chức có trên sàn |  |
| 20 | BO\_DIACHI | NUMBER | Bộ kê khai địa chỉ |  |

## Biểu đồ lớp thiết kế



*Hình 3.11. Biểu đồ lớp thiết kế*

# CÀI ĐẶT VÀ TRIỂN KHAI

## Cài đặt môi trường

### My SQL

* Cài đặt MySQL
* Tải workbench phiên bản 8.0
* Cài đặt sever và tài khoản root
* Kết nối MySQL vs fast api

### FastAPI

* Cài đặt Visual Studio Code
* Tải Visual Studio Code
* Cài các extension của Python như Python, Pylance
* Tạo folder chứa project
* Cài đặt python
* Tải python từ web và cài đặt
* Cài đặt FastAPI: Thông thường khi cài đặt VS code sẽ tự động có pip
* Trong VS Code, mở terminal
* Gõ py -m pip install --upgrade pip
* Sau khi update pip, gõ pip install fastapi[all], uvicorn
* Cài đặt sqlalchemy
* Sau khi update pip, gõ pip install sqlalchemy[all]
* Cài đặt các thư viên liên quan:
* pip install pandas, numpy, matplotlib

## Các hàm thủ tục và api

### Thủ tục tổng hợp dự liệu bảng ecom\_sup\_ct

Code xử lý:

CREATE DEFINER=`root`@`localhost` PROCEDURE `prc\_ECOM\_SUP\_FCT`(IN p\_etl\_date VARCHAR(8))

BEGIN

DELETE FROM tmdt.ECOM\_SUP\_FCT

WHERE CONCAT(SUBSTR(p\_etl\_date, 1, 4), 'Q',CEIL(SUBSTR(p\_etl\_date, 5, 2)/3)) =KYKKHAI;

INSERT INTO tmdt.ECOM\_SUP\_FCT(

STT ,

MA\_CQT ,

MSG\_ID ,

MACQTNOINOP ,

TenCQTNOINOP ,

CQTNCC ,

MAQUANLYTRENSAN ,

MST\_NCC ,

MA\_LOAIGIAYTO ,

SO ,

LATOCHUCCN ,

LATOCHUCNN ,

SO\_GPKD ,

EMAIL ,

SODIENTHOAI ,

TENNHACUNGCAP ,

MST\_SAN ,

TENNNT ,

MALOAIHINH ,

CACKHOANGIAMTRU ,

LOAI\_HANGHOA ,

TEN\_NGANHANG ,

CHUTAIKHOAN ,

SOTAIKHOAN ,

TRUSO\_QUOCGIA\_TEN ,

TRUSO\_TINH\_TP\_TEN ,

TRUSO\_QUAN\_HUYEN\_TEN ,

TRUSO\_PHUONG\_XA\_TEN ,

TRUSO\_SONHA ,

THUONGTRU\_QUOCGIA\_TEN ,

THUONGTRU\_TINH\_TP\_TEN ,

THUONGTRU\_QUAN\_HUYEN\_TEN ,

THUONGTRU\_PHUONG\_XA\_TEN ,

THUONGTRU\_SONHA ,

TONG ,

TONG\_DONHANGTHANHCONG ,

TONG\_DONHANGKHONGTHANHCONG ,

KYKKHAI ,

KYKKHAITUNGAY ,

KYKKHAIDENNGAY ,

INVALID\_MANCC ,

INVALID\_SO\_GT ,

INVALID\_SO\_GPKD ,

INVALID\_MST ,

INVALID\_HQ\_ADDRESS ,

INVALID\_EMAIL ,

INVALID\_PHONE ,

INVALID\_LOAIGT ,

INVALID\_ADDRESS ,

INVALID\_SUP ,

QUAN\_HUYEN\_MA

)

with pr as (select case when substr(p\_etl\_date,5,2) in ('01','02','03') then 1

when substr(p\_etl\_date,5,2) in ('04','05','06') then 2

when substr(p\_etl\_date,5,2) in ('07','08','09') then 3

when substr(p\_etl\_date,5,2) in ('10','11','12') then 4 end tu,

case when substr(p\_etl\_date,5,2) in ('01','02','03') then 1

when substr(p\_etl\_date,5,2) in ('04','05','06') then 2

when substr(p\_etl\_date,5,2) in ('07','08','09') then 3

when substr(p\_etl\_date,5,2) in ('10','11','12') then 4 end den,

substr(p\_etl\_date,1,4) nam

from dual),

-- chi lay to khai co ngay nop gan nhat

max\_status as (

SELECT aa.MST, MAQUANLYTRENSAN, KYKKHAITUNGAY, KYKKHAIDENNGAY , MAX(NGAY\_NOP\_TK) AS NGAY\_NOP\_TK

FROM tmdt.ETA\_THONGTINCHUNG aa

JOIN tmdt.ETA\_BANGTHONGTINCHITIET bb ON aa.MSG\_ID = bb.MSG\_ID

JOIN pr ON 1=1

WHERE MA\_TRANGTHAI <> '03'

AND aa.KYKKHAI IN (

SELECT CONCAT(pr.tu - 1 + numbers.n, '/', pr.nam) AS KYKKHAI

FROM (

SELECT 1 AS n

UNION ALL SELECT 2 UNION ALL SELECT 3

-- Thêm các số cần thiết cho dãy giá trị

) AS numbers

JOIN pr ON numbers.n <= pr.den - pr.tu + 1

)

GROUP BY aa.MST, MAQUANLYTRENSAN, KYKKHAITUNGAY, KYKKHAIDENNGAY

),

dr as (

select aa.MSG\_ID

, aa.MACQTNOINOP

, bb.MAQUANLYTRENSAN

, bb.MST MST\_NCC

, bb.MA\_LOAIGIAYTO

, bb.SO

, bb.SO\_GPKD

, cc.EMAIL

, cc.SODIENTHOAI

, bb.CACKHOANGIAMTRU

, dd.LOAI\_HANGHOA

, ee.TEN\_NGANHANG, ee.CHUTAIKHOAN, ee.SOTAIKHOAN

, bb.TENNHACUNGCAP TENNHACUNGCAP

, aa.MST MST\_SAN

, aa.TENNNT TENNNT

, bb.MALOAIHINH

, cc.TRUSO\_QUOCGIA\_TEN TRUSO\_QUOCGIA\_TEN

, cc.TRUSO\_TINH\_TP\_TEN TRUSO\_TINH\_TP\_TEN

, cc.TRUSO\_QUAN\_HUYEN\_TEN TRUSO\_QUAN\_HUYEN\_TEN

, cc.TRUSO\_PHUONG\_XA\_TEN TRUSO\_PHUONG\_XA\_TEN

, cc.TRUSO\_SONHA TRUSO\_SONHA

, cc.THUONGTRU\_QUOCGIA\_TEN THUONGTRU\_QUOCGIA\_TEN

, cc.THUONGTRU\_TINH\_TP\_TEN THUONGTRU\_TINH\_TP\_TEN

, cc.THUONGTRU\_QUAN\_HUYEN\_TEN THUONGTRU\_QUAN\_HUYEN\_TEN

, cc.THUONGTRU\_PHUONG\_XA\_TEN THUONGTRU\_PHUONG\_XA\_TEN

, cc.THUONGTRU\_SONHA THUONGTRU\_SONHA -- ,bb.TONG, bb.TONG\_DONHANGTHANHCONG, bb.TONG\_DONHANGKHONGTHANHCONG

, SUM(case when max\_status.MST is not null then COALESCE(bb.TONG) else 0 end) TONG

, SUM(case when max\_status.MST is not null then COALESCE(bb.TONG\_DONHANGTHANHCONG) else 0 end) TONG\_DONHANGTHANHCONG

, SUM(case when max\_status.MST is not null then COALESCE(bb.TONG\_DONHANGKHONGTHANHCONG) else 0 end) TONG\_DONHANGKHONGTHANHCONG

, aa.KYKKHAI

,(STR\_TO\_DATE(aa.KYKKHAITUNGAY, '%m/%d/%Y %H:%i:%s'))KYKKHAITUNGAY

,(STR\_TO\_DATE(aa.KYKKHAIDENNGAY, '%m/%d/%Y %H:%i:%s'))KYKKHAIDENNGAY

, case when bb.MAQUANLYTRENSAN is null then 1 else 0 end invalid\_maNCC

, case when bb.SO is null and MA\_LOAIGIAYTO is not null then 1 else 0 end invalid\_so\_gt

, case when bb.MALOAIHINH = '01' and SO\_GPKD is null then 1 else 0 end invalid\_so\_GPKD

, case when bb.MALOAIHINH = '01' and bb.MST is null then 1 else 0 end invalid\_MST

, case when bb.MALOAIHINH = '01' and (cc.TRUSO\_QUOCGIA\_TEN is null

or cc.TRUSO\_TINH\_TP\_TEN is null

or cc.TRUSO\_QUAN\_HUYEN\_TEN is null

or cc.TRUSO\_PHUONG\_XA\_TEN is null) then 1 else 0 end invalid\_hq\_address

, case when bb.MALOAIHINH in ('01','02') and cc.EMAIL is null then 1 else 0 end invalid\_email

, case when bb.MALOAIHINH = '02' and cc.SODIENTHOAI is null then 1 else 0 end invalid\_phone

, case when bb.MALOAIHINH = '02' and bb.MA\_LOAIGIAYTO is null then 1 else 0 end invalid\_loaiGT

, case when bb.MALOAIHINH in ('02','03') and (cc.THUONGTRU\_QUOCGIA\_TEN is null

or cc.THUONGTRU\_TINH\_TP\_TEN is null

or cc.THUONGTRU\_QUAN\_HUYEN\_TEN is null

or cc.THUONGTRU\_PHUONG\_XA\_TEN is null) then 1 else 0 end invalid\_address

, case when bb.MAQUANLYTRENSAN is null then 1

when bb.SO is null and MA\_LOAIGIAYTO is not null then 1

when bb.MALOAIHINH = '01' and (SO\_GPKD is null

or bb.MST is null

or cc.TRUSO\_QUOCGIA\_TEN is null

or cc.TRUSO\_TINH\_TP\_TEN is null

or cc.TRUSO\_QUAN\_HUYEN\_TEN is null

or cc.TRUSO\_PHUONG\_XA\_TEN is null

or cc.EMAIL is null

or cc.SODIENTHOAI is null) then 1

when bb.MALOAIHINH = '02' and (bb.MA\_LOAIGIAYTO is null

or cc.EMAIL is null

or cc.SODIENTHOAI is null

or cc.THUONGTRU\_QUOCGIA\_TEN is null

or cc.THUONGTRU\_TINH\_TP\_TEN is null

or cc.THUONGTRU\_QUAN\_HUYEN\_TEN is null

or cc.THUONGTRU\_PHUONG\_XA\_TEN is null) then 1

when bb.MALOAIHINH = '03' and (cc.THUONGTRU\_QUOCGIA\_TEN is null

or cc.THUONGTRU\_TINH\_TP\_TEN is null

or cc.THUONGTRU\_QUAN\_HUYEN\_TEN is null

or cc.THUONGTRU\_PHUONG\_XA\_TEN is null) then 1

else 0 end invalid\_sup

, CASE WHEN cc.TRUSO\_QUAN\_HUYEN\_MA is null THEN cc.THUONGTRU\_QUAN\_HUYEN\_MA

WHEN cc.THUONGTRU\_QUAN\_HUYEN\_MA is null then cc.TRUSO\_QUAN\_HUYEN\_MA

when cc.THUONGTRU\_QUAN\_HUYEN\_MA is null and cc.TRUSO\_QUAN\_HUYEN\_MA is null then 'undefind'

else cc.TRUSO\_QUAN\_HUYEN\_MA

END QUAN\_HUYEN\_MA

from tmdt.ETA\_THONGTINCHUNG aa

join pr on 1=1

join tmdt.ETA\_BANGTHONGTINCHITIET bb on aa.MSG\_ID = bb.MSG\_ID

and bb.MALOAIHINH in ('01','02','03','04')

and aa.KYKKHAI in (SELECT CONCAT(pr.tu - 1 + numbers.n, '/', pr.nam) AS KYKKHAI

FROM (

SELECT 1 AS n

UNION ALL SELECT 2 UNION ALL SELECT 3

-- Thêm các số cần thiết cho dãy giá trị

) AS numbers

JOIN pr ON numbers.n <= pr.den - pr.tu + 1)

and bb.MA\_TRANGTHAI <> '03'

join max\_status on aa.MST = max\_status.MST and bb.MAQUANLYTRENSAN = max\_status.MAQUANLYTRENSAN

and aa.NGAY\_NOP\_TK = max\_status.NGAY\_NOP\_TK

and aa.KYKKHAITUNGAY = max\_status.KYKKHAITUNGAY

and aa.KYKKHAIDENNGAY = max\_status.KYKKHAIDENNGAY

left join tmdt.ETA\_PHULUC\_01 cc on aa.MSG\_ID = cc.MSG\_ID and bb.MAQUANLYTRENSAN = cc.MAQUANLYTRENSAN and (cc.maloaihinh is not null or cc.maquanlytrensan is not null)

left join tmdt.ETA\_PHULUC\_02 dd on aa.MSG\_ID = dd.MSG\_ID and bb.MAQUANLYTRENSAN = dd.MAQUANLYTRENSAN

left join (select distinct MAQUANLYTRENSAN,SOTAIKHOAN,CHUTAIKHOAN,MA\_NGANHANG,TEN\_NGANHANG,MSG\_ID from tmdt.ETA\_PHULUC\_03) ee on aa.MSG\_ID = ee.MSG\_ID and bb.MAQUANLYTRENSAN = ee.MAQUANLYTRENSAN

-- where bb.mst = '0102289856'

group by aa.MSG\_ID

, aa.MACQTNOINOP

, bb.MAQUANLYTRENSAN

, bb.MST

, bb.MA\_LOAIGIAYTO

, bb.SO

, bb.SO\_GPKD

, cc.EMAIL

, cc.SODIENTHOAI

, bb.TENNHACUNGCAP

, aa.MST

, aa.TENNNT

, bb.MALOAIHINH

, bb.CACKHOANGIAMTRU

, dd.LOAI\_HANGHOA

, ee.TEN\_NGANHANG, ee.CHUTAIKHOAN, ee.SOTAIKHOAN

, cc.TRUSO\_QUOCGIA\_TEN

, cc.TRUSO\_TINH\_TP\_TEN

, cc.TRUSO\_QUAN\_HUYEN\_TEN

, cc.TRUSO\_PHUONG\_XA\_TEN

, cc.TRUSO\_SONHA

, cc.THUONGTRU\_QUOCGIA\_TEN

, cc.THUONGTRU\_TINH\_TP\_TEN

, cc.THUONGTRU\_QUAN\_HUYEN\_TEN

, cc.THUONGTRU\_PHUONG\_XA\_TEN

, cc.THUONGTRU\_SONHA

, aa.KYKKHAI, aa.KYKKHAITUNGAY, aa.KYKKHAIDENNGAY

,cc.TRUSO\_QUAN\_HUYEN\_MA,cc.THUONGTRU\_QUAN\_HUYEN\_MA)

,

d as (select

row\_number() over (partition by MAQUANLYTRENSAN,MST\_SAN,KYKKHAI order by msg\_id desc) stt -- lay to khai tu goi tin moi nhat

, MSG\_ID

, MACQTNOINOP

, MAQUANLYTRENSAN

, MST\_NCC

, MA\_LOAIGIAYTO

, SO

, SO\_GPKD

, EMAIL

, SODIENTHOAI

, TENNHACUNGCAP

, MST\_SAN

, TENNNT

, MALOAIHINH

, CACKHOANGIAMTRU

, LOAI\_HANGHOA

, TEN\_NGANHANG, CHUTAIKHOAN, SOTAIKHOAN

, TRUSO\_QUOCGIA\_TEN

, TRUSO\_TINH\_TP\_TEN

, TRUSO\_QUAN\_HUYEN\_TEN

, TRUSO\_PHUONG\_XA\_TEN

, TRUSO\_SONHA

, THUONGTRU\_QUOCGIA\_TEN

, THUONGTRU\_TINH\_TP\_TEN

, THUONGTRU\_QUAN\_HUYEN\_TEN

, THUONGTRU\_PHUONG\_XA\_TEN

, THUONGTRU\_SONHA -- ,bb.TONG, bb.TONG\_DONHANGTHANHCONG, bb.TONG\_DONHANGKHONGTHANHCONG

, TONG

, TONG\_DONHANGTHANHCONG

, TONG\_DONHANGKHONGTHANHCONG

, CONCAT(substr(KYKKHAI,3,4),'Q',substr(KYKKHAI,1,1)) KYKKHAI

, KYKKHAITUNGAY

, KYKKHAIDENNGAY

, invalid\_maNCC

, invalid\_so\_gt

, invalid\_so\_GPKD

, invalid\_MST

, invalid\_hq\_address

, invalid\_email

, invalid\_phone

, invalid\_loaiGT

, invalid\_address

, invalid\_sup

,QUAN\_HUYEN\_MA

from dr)

select stt

, s.MA\_CQT

, MSG\_ID

, MACQTNOINOP

, s.TEN\_CQT\_DAI TenCQTNOINOP

, s.TEN\_CQT\_DAI CQTNCC

, MAQUANLYTRENSAN

, case when d.MST\_NCC is null then tts2.tin else d.mst\_ncc end mst\_ncc

, MA\_LOAIGIAYTO

, case when d.SO is null then tts1.IDENTIFICATION else d.SO end SO

, (case when MALOAIHINH = '04' then 'x' else ' 'end) LATOCHUCCN

, (case when MALOAIHINH = '03' then 'x' else ' 'end) LATOCHUCNN

, SO\_GPKD

, EMAIL

, SODIENTHOAI

, TENNHACUNGCAP

, MST\_SAN

, TENNNT

, MALOAIHINH

, CACKHOANGIAMTRU

, LOAI\_HANGHOA

, TEN\_NGANHANG, CHUTAIKHOAN, SOTAIKHOAN

, TRUSO\_QUOCGIA\_TEN

, TRUSO\_TINH\_TP\_TEN

, TRUSO\_QUAN\_HUYEN\_TEN

, TRUSO\_PHUONG\_XA\_TEN

, TRUSO\_SONHA

, THUONGTRU\_QUOCGIA\_TEN

, THUONGTRU\_TINH\_TP\_TEN

, THUONGTRU\_QUAN\_HUYEN\_TEN

, THUONGTRU\_PHUONG\_XA\_TEN

, THUONGTRU\_SONHA -- ,bb.TONG, bb.TONG\_DONHANGTHANHCONG, bb.TONG\_DONHANGKHONGTHANHCONG

, TONG

, TONG\_DONHANGTHANHCONG

, TONG\_DONHANGKHONGTHANHCONG

, KYKKHAI

, KYKKHAITUNGAY

, KYKKHAIDENNGAY

, invalid\_maNCC

, invalid\_so\_gt

, invalid\_so\_GPKD

, invalid\_MST

, invalid\_hq\_address

, invalid\_email

, invalid\_phone

, invalid\_loaiGT

, invalid\_address

, invalid\_sup

, QUAN\_HUYEN\_MA

from d

left join tmdt.ztb\_map\_cqt s on d.MACQTNOINOP = s.MA\_QLT

left join tmdt.tin\_taxo\_status tts1 on tts1.TIN = d.MST\_NCC and tts1.IND= 'Y' and tts1.IDENTIFICATION not in ('000000000','000000000000')

left join tmdt.tin\_taxo\_status tts2 on d.MST\_NCC is null and d.so is not null and d.so = tts2.IDENTIFICATION and tts2.ind = 'Y' and tts1.IDENTIFICATION not in ('000000000','000000000000')

where stt = 1;

END

### Thủ tục tổng hợp dự liệu bảng ecom\_san\_ct

Code xử lý:

CREATE DEFINER=`root`@`localhost` PROCEDURE `prc\_ECOM\_SAN\_FCT`(IN p\_etl\_date VARCHAR(8))

BEGIN

DELETE FROM tmdt.ECOM\_SAN\_FCT

WHERE CONCAT(SUBSTR(p\_etl\_date, 1, 4), 'Q',CEIL(SUBSTR(p\_etl\_date, 5, 2)/3)) =KYKKHAI;

INSERT INTO tmdt.ECOM\_SAN\_FCT (

STT

,TEN\_CQT\_DAI

,TENNNT

,mst

,macqtnoinop

,kykkhai

,TENTINHNNT

,count\_MST\_null

,GIAMTRU

,TONG

,TONG\_DONHANGKHONGTHANHCONG

,TONG\_DONHANGTHANHCONG

,tong\_so\_don\_hang

,count\_SDT\_null

,count\_tochucVNncc

,count\_canhanVNncc

,count\_tochucNNncc

,count\_canhanNNncc

,tong\_DN\_CN

,BO\_DIACHI

)

with pr as (select

case when substr(p\_etl\_date,5,2) in ('01','02','03') then 1

when substr(p\_etl\_date,5,2) in ('04','05','06') then 2

when substr(p\_etl\_date,5,2) in ('07','08','09') then 3

when substr(p\_etl\_date,5,2) in ('10','11','12') then 4 end tu,

case when substr(p\_etl\_date,5,2) in ('01','02','03') then 1

when substr(p\_etl\_date,5,2) in ('04','05','06') then 2

when substr(p\_etl\_date,5,2) in ('07','08','09') then 3

when substr(p\_etl\_date,5,2) in ('10','11','12') then 4 end den,

substr(p\_etl\_date,1,4) nam

from dual)

,

MAX\_NGAY\_NOP\_TK AS (

SELECT MAX(NGAY\_NOP\_TK) NGAY\_NOP\_TK , aa.MST

FROM tmdt.ETA\_THONGTINCHUNG AA

join pr on 1 =1

WHERE 1=1

-- and bb.maquanlytrensan = '0108733570' and aa.MST='0108733570'

AND aa.KYKKHAI in (SELECT CONCAT(pr.tu - 1 + numbers.n, '/', pr.nam) AS KYKKHAI

FROM (

SELECT 1 AS n

UNION ALL SELECT 2 UNION ALL SELECT 3

-- Thêm các số cần thiết cho dãy giá trị

) AS numbers

JOIN pr ON numbers.n <= pr.den - pr.tu + 1) -- P\_kykkhai

GROUP BY aa.MST

)

,

max\_msgid\_by\_mst as (select max(msg\_id) max\_msg\_id,aa.mst

FROM tmdt.ETA\_THONGTINCHUNG aa

left JOIN MAX\_NGAY\_NOP\_TK ON MAX\_NGAY\_NOP\_TK.MST = aa.MST

AND MAX\_NGAY\_NOP\_TK.NGAY\_NOP\_TK = AA.NGAY\_NOP\_TK

join pr on 1 =1

WHERE 1=1

AND aa.KYKKHAI in (SELECT CONCAT(pr.tu - 1 + numbers.n, '/', pr.nam) AS KYKKHAI

FROM (

SELECT 1 AS n

UNION ALL SELECT 2 UNION ALL SELECT 3

-- Thêm các số cần thiết cho dãy giá trị

) AS numbers

JOIN pr ON numbers.n <= pr.den - pr.tu + 1)

group by aa.mst

)

,

ad as(

select distinct msg\_id,maquanlytrensan from ETA\_PHULUC\_01 x where 1=1

and(x.TRUSO\_QUOCGIA\_TEN is not null

or x.TRUSO\_TINH\_TP\_TEN is not null

or x.TRUSO\_PHUONG\_XA\_TEN is not null

or x.TRUSO\_QUAN\_HUYEN\_TEN is not null

or x.TRUSO\_SONHA is not null

or x.THUONGTRU\_QUOCGIA\_TEN is not null

or x.THUONGTRU\_TINH\_TP\_TEN is not null

or x.THUONGTRU\_PHUONG\_XA\_TEN is not null

or x.THUONGTRU\_QUAN\_HUYEN\_TEN is not null

or x.THUONGTRU\_SONHA is not null)),

xad as

(select distinct msg\_id,maquanlytrensan from ETA\_PHULUC\_01 x where 1=1

and (maloaihinh is not null or maquanlytrensan is not null)

and (msg\_id,maquanlytrensan) not in (select msg\_id,maquanlytrensan from ad)

),

mb as (

select distinct msg\_id,maquanlytrensan from ETA\_PHULUC\_01 x where 1=1

and x.sodienthoai is not null),

xmb as

(select distinct msg\_id,maquanlytrensan from ETA\_PHULUC\_01 x where 1=1

and (maloaihinh is not null or maquanlytrensan is not null)

and (msg\_id,maquanlytrensan) not in (select msg\_id,maquanlytrensan from mb)

)

,

-- thong tin chi tiet ncc

DATARAW\_ncc AS (

SELECT ROW\_NUMBER() OVER (partition by aa.mst,bb.MAQUANLYTRENSAN order by COALESCE(bb.MSG\_ID,0) desc) STT ,aa.MSG\_ID

-- 1 san gui 2 goi msg -> goi truoc co thong tin ncc, goi sau khong co thong tin nha cung cap thi lay goi truoc

, aa.MACQTNOINOP

, aa.TENNNT

, aa.TENTINHNNT

, aa.MST

, case when bb.MST is null then 1 else 0 end MST\_null

, bb.MAQUANLYTRENSAN

, bb.MALOAIHINH

, bb.CACKHOANGIAMTRU

, bb.TONG

, bb.TONG\_DONHANGTHANHCONG

, bb.TONG\_DONHANGKHONGTHANHCONG

, AA.KYKKHAI

FROM tmdt.ETA\_THONGTINCHUNG aa

JOIN MAX\_NGAY\_NOP\_TK ON MAX\_NGAY\_NOP\_TK.MST = aa.MST

AND MAX\_NGAY\_NOP\_TK.NGAY\_NOP\_TK = AA.NGAY\_NOP\_TK

LEFT JOIN tmdt.ETA\_BANGTHONGTINCHITIET BB

ON BB.MSG\_ID = AA.MSG\_ID AND (BB.MA\_TRANGTHAI <> '03' or BB.MA\_TRANGTHAI is null)

join pr on 1 =1

WHERE 1=1

AND (BB.MA\_TRANGTHAI <> '03' or BB.MA\_TRANGTHAI is null)

-- and aa.mst = '0308808576'

AND aa.KYKKHAI in (SELECT CONCAT(pr.tu - 1 + numbers.n, '/', pr.nam) AS KYKKHAI

FROM (

SELECT 1 AS n

UNION ALL SELECT 2 UNION ALL SELECT 3

-- Thêm các số cần thiết cho dãy giá trị

) AS numbers

JOIN pr ON numbers.n <= pr.den - pr.tu + 1) -- P\_kykkhai

)

,

DATARAW\_san as (

SELECT a.MACQTNOINOP

,a.MST

,a.KYKKHAI

,a.TENNNT

,a.TENTINHNNT

,sum(COALESCE(a.MST\_null)) count\_MST\_null

,sum(COALESCE(a.CACKHOANGIAMTRU)) GIAMTRU

,sum(COALESCE(a.TONG)) TONG

,sum(COALESCE(a.TONG\_DONHANGTHANHCONG)) TONG\_DONHANGTHANHCONG

,sum(COALESCE(a.TONG\_DONHANGKHONGTHANHCONG)) TONG\_DONHANGKHONGTHANHCONG

,sum(COALESCE(a.TONG\_DONHANGTHANHCONG)) + sum(COALESCE(a.TONG\_DONHANGKHONGTHANHCONG)) tong\_so\_don\_hang

,SUM(case when a.MALOAIHINH= '01' then 1 else 0 end) count\_tochucVNncc

,SUM(case when a.MALOAIHINH= '02' then 1 else 0 end) count\_canhanVNncc

,SUM(case when a.MALOAIHINH= '03' then 1 else 0 end) count\_tochucNNncc

,SUM(case when a.MALOAIHINH= '04' then 1 else 0 end) count\_canhanNNncc

,SUM(case when xmb.MAQUANLYTRENSAN is null then 0 else 1 end) count\_SDT\_null

,SUM(case when xad.MAQUANLYTRENSAN is null then 0 else 1 end) BO\_DIACHI

FROM DATARAW\_ncc a

left join xad on xad.MAQUANLYTRENSAN = a.MAQUANLYTRENSAN and xad.MSG\_ID =a.MSG\_ID

left join xmb on xmb.MAQUANLYTRENSAN = a.MAQUANLYTRENSAN and xmb.MSG\_ID =a.MSG\_ID

where 1=1

and a.stt = 1

GROUP BY a.MACQTNOINOP

,a.MST

,a.KYKKHAI

,a.TENNNT

,a.TENTINHNNT

)

-- select \* from DATARAW\_san;

,

san as (

select aa.MACQTNOINOP

,aa.TENNNT

,aa.TENTINHNNT

,aa.MST

,aa.KYKKHAI

,row\_number() over (partition by aa.MACQTNOINOP,aa.TENNNT,aa.TENTINHNNT,aa.MST,aa.KYKKHAI order by msg\_id) stt

FROM tmdt.ETA\_THONGTINCHUNG aa

JOIN MAX\_NGAY\_NOP\_TK ON MAX\_NGAY\_NOP\_TK.MST = aa.MST

AND MAX\_NGAY\_NOP\_TK.NGAY\_NOP\_TK = AA.NGAY\_NOP\_TK

join pr on 1 =1

WHERE 1=1

AND aa.KYKKHAI in (SELECT CONCAT(pr.tu - 1 + numbers.n, '/', pr.nam) AS KYKKHAI

FROM (

SELECT 1 AS n

UNION ALL SELECT 2 UNION ALL SELECT 3

-- Thêm các số cần thiết cho dãy giá trị

) AS numbers

JOIN pr ON numbers.n <= pr.den - pr.tu + 1) -- P\_kykkhai

), dt as (

select aa.MACQTNOINOP

,aa.TENNNT

,aa.TENTINHNNT

,aa.MST

, CONCAT(substr(aa.KYKKHAI,3,4),'Q',substr(aa.KYKKHAI,1,1)) KYKKHAI

,bb.count\_MST\_null

,bb.GIAMTRU

,bb.TONG

,bb.TONG\_DONHANGTHANHCONG

,bb.TONG\_DONHANGKHONGTHANHCONG

,bb.tong\_so\_don\_hang

,bb.count\_tochucVNncc

,bb.count\_canhanVNncc

,bb.count\_tochucNNncc

,bb.count\_canhanNNncc

,bb.count\_SDT\_null

,bb.BO\_DIACHI

FROM san aa

left join DATARAW\_san bb on bb.mst = aa.MST and bb.MACQTNOINOP = aa.MACQTNOINOP and bb.KYKKHAI =AA.KYKKHAI

join pr on 1 =1

WHERE 1=1

and stt = 1)

select ROW\_NUMBER() OVER (partition by null order by null) STT

,s.TEN\_CQT\_DAI

,dt.TENNNT

,dt.mst

,dt.macqtnoinop

,dt.kykkhai

,dt.TENTINHNNT

,dt.count\_MST\_null

,dt.GIAMTRU

,dt.TONG

,dt.TONG\_DONHANGKHONGTHANHCONG

,dt.TONG\_DONHANGTHANHCONG

,dt.tong\_so\_don\_hang

,dt.count\_SDT\_null

,count\_tochucVNncc

,count\_canhanVNncc

,count\_tochucNNncc

,count\_canhanNNncc

,(count\_tochucVNncc + count\_canhanVNncc + count\_tochucNNncc + count\_canhanNNncc) tong\_DN\_CN

, dt.BO\_DIACHI

from dt

left join tmdt.ztb\_map\_cqt s on dt.MACQTNOINOP = s.MA\_QLT

where 1=1

-- and dt.mst= '0108824877 '

order by dt.mst,macqtnoinop ;

END

### Hàm api đăng nhập

-    Chức năng Đăng nhập

    + NHập 2 trường user và password rồi so sánh dữ liệu vs bảng user trong db xem có dữ liệu dó không

    + cqt: tên đăng nhập

    + password: mật khẩu đăng nhập

Code xử lý:

@app.post("/login")

async def Login(cqt: str, password: str, db: Session = Depends(get\_db)):

    user = db.query(User).filter(User.cqt == cqt, User.password == password).first()

    if user is None:

        raise HTTPException(status\_code=401, detail="Đăng nhập không thành công")

    return user

### Hàm api tạo user đăng nhập

- Tạo thêm dữ liệu vào cơ sở dữ liệu

- user: tạo một đối tượng user từ dữ liệu được gửi lên và sử dụng

- db.add(usser): thêm đối tượng data\_to\_insert vào phiên làm việc SQLAlchemy

- db.commit(): thực hiện việc chèn dữ liệu đã được thêm vào phiên làm việc vào cơ sở dữ liệu

Code xử lý:

@app.post("/create\_user", status\_code=status.HTTP\_201\_CREATED)

async def create\_user(data: userCreate, db: Session = Depends(get\_db)):

user = User(id=data.id, cqt=data.cqt,password=data.password)

db.add(user)

db.commit()

db.refresh(user)

return user

### Hàm api lấy dữ liệu ecom\_sup\_fct

- Thống kê dữu liệu nhà cung cấp

    - MA\_CQT: tên đại chỉ cơ quan thuế muốn in

    - KYKKHAI: Kỳ kê khai muốn in

    - sau khi thực hiện xong sẽ trả kết quả dưới dạng json

Code xử lý:

@app.get("/get\_data\_ECOM\_SUP\_FCT")

def get\_data( MA\_CQT: str,KYKKHAI:  str):

    db = SessionLocal()

    data\_entry = db.query(ECOM\_SUP\_FCT).filter( ECOM\_SUP\_FCT.MA\_CQT == MA\_CQT,ECOM\_SUP\_FCT.KYKKHAI == KYKKHAI).first()

    db.close()

    if not data\_entry:

        raise HTTPException(status\_code=404, detail="Không tìm thấy dữ liệu")

    return data\_entry

### Hàm api lấy dữ liệu ecom\_san\_fct

- Thống kê dữu liệu sàn thương mại điện tử

    - KYKKHAI: Kỳ kê khai muốn in

    - sau khi thực hiện xong sẽ trả kết quả dưới dạng json

Code xử lý:

@app.get("/get\_data\_ECOM\_SAN\_FCT")

def get\_data(KYKKHAI: str):

    db = SessionLocal()

    data\_entry = db.query(ECOM\_SUP\_FCT).filter(ECOM\_SUP\_FCT.KYKKHAI == KYKKHAI).first()

    db.close()

    if not data\_entry:

        raise HTTPException(status\_code=404, detail="Không tìm thấy dữ liệu")

    return data\_entry

### Hàm api lấy dữ liệu ecom\_sup\_fct với mã số thuế

  - Thống kê dữu liệu sàn thương mại điện tử

    - KYKKHAI: Kỳ kê khai muốn in

    - MST\_NCC: Mã số thuế nhà cung cấp

    - sau khi thực hiện xong sẽ trả kết quả dưới dạng json

Code xử lý:

@app.get("/get\_data\_ECOM\_SUP\_FCT\_with\_mst")

def get\_data(KYKKHAI: str='2022Q4', MST\_NCC:str='8033509894'):

    db = SessionLocal()

    data\_entry = db.query(ECOM\_SUP\_FCT).filter(ECOM\_SUP\_FCT.KYKKHAI == KYKKHAI,ECOM\_SUP\_FCT.MST\_NCC == MST\_NCC).first()

    db.close()

    if not data\_entry:

        raise HTTPException(status\_code=404, detail="Không tìm thấy dữ liệu")

    return data\_entry

### Hàm api xóa dữ liệu ecom\_sup\_fct

- Xóa dữ liệu dựa theo mã Code và năm được chỉ định

- record\_to\_delete: thực hiện truy vấn cơ sở dữ liệu để tìm bản ghi cần xóa theo các biến đầu vào và lấy bản ghi đầu tiên

- Kiểm tra xem có tồn tại dữ liệu nếu không sẽ trả về lỗi

Code xử lý:

@app.delete("/delete\_data\_ECOM\_SUP\_FCT/{KYKKHAI}", status\_code = status.HTTP\_200\_OK)

async def delete\_data( KYKKHAI: str):

db = SessionLocal()

record\_to\_delete = db.query(ECOM\_SUP\_FCT).filter(ECOM\_SUP\_FCT.KYKKHAI == KYKKHAI).first()

if record\_to\_delete:

db.delete(record\_to\_delete)

db.commit()

return {"message": "Dữ liệu đã được xóa thành công."}

else:

raise HTTPException(status\_code=404, detail="Không tìm thấy dữ liệu để xóa.")

### Hàm api xóa dữ liệu ecom\_san\_fct

- Xóa dữ liệu dựa theo KYKKHAI được chỉ định

- record\_to\_delete: thực hiện truy vấn cơ sở dữ liệu để tìm bản ghi cần xóa theo các biến đầu vào và lấy bản ghi đầu tiên

- Kiểm tra xem có tồn tại dữ liệu nếu không sẽ trả về lỗi

Code xử lý:

@app.delete("/delete\_data\_ECOM\_SAN\_FCT/{KYKKHAI}", status\_code = status.HTTP\_200\_OK)

async def delete\_data( KYKKHAI: str):

db = SessionLocal()

record\_to\_delete = db.query(ECOM\_SAN\_FCT).filter(ECOM\_SAN\_FCT.KYKKHAI == KYKKHAI).first()

if record\_to\_delete:

db.delete(record\_to\_delete)

db.commit()

return {"message": "Dữ liệu đã được xóa thành công."}

else:

raise HTTPException(status\_code=404, detail="Không tìm thấy dữ liệu để xóa.")

### Hàm api vẽ biểu đồ tròn thể hiện tỷ lệ loại hình cá nhân hay doanh nghiệp

- Vẽ biểu đồ tròn tỷ lệ loại hình cá nhân hoạc doanh nghiệp trong kỳ khai được chỉ định

- data: lọc dữ liệu và lấy count bản ghi maloaihinh của kykhai

- labels: tạo danh sách nhãn và kích thước cho các phần của biểu đồ tròn dựa trên dữ liệu vừa tính toán

Code xử lý:

@app.get("/pie\_chart/{KYKKHAI}")

async def pie\_chart(KYKKHAI: str, db: Session = Depends(get\_db)):

try:

# Modify the query to filter by KYKKHAI and group by MALOAIHINH with count

data = (

db.query(ECOM\_SUP\_FCT.MALOAIHINH, func.count(ECOM\_SUP\_FCT.MALOAIHINH).label('SoLuong'))

.filter(ECOM\_SUP\_FCT.KYKKHAI == KYKKHAI)

.group\_by(ECOM\_SUP\_FCT.MALOAIHINH)

.all()

)

if not data:

raise HTTPException(status\_code=404, detail=f"No data found for {KYKKHAI}")

labels = [f"Loại {record.MALOAIHINH}" for record in data]

sizes = [record.SoLuong for record in data]

plt.figure(figsize=(6, 6))

plt.pie(sizes, labels=labels, autopct='%1.1f%%', startangle=140)

plt.title(f"Số loại hình nộp thuế trong {KYKKHAI}")

img = BytesIO()

plt.savefig(img, format="png")

plt.close()

return Response(content=img.getvalue(), media\_type="image/png")

except Exception as e:

raise HTTPException(status\_code=500, detail=f"Error processing data: {str(e)}")

### Hàm api Thống kê dữu liệu sàn có số giá trị tong\_so\_don\_hang lớn nhất

- Thống kê dữu liệu sàn có số giá trị tong\_so\_don\_hang lớn nhất

- subquery: tổng hợp dữ liệu sàn có giá trị giá trị tong\_so\_don\_hang lớn nhất

- query: từ số đươn hàng lớn nhất tìm thông tin chi tiết sàn đó

- sau khi thực hiện xong sẽ trả kết quả dưới dạng json

Code xử lý:

@app.get("/max\_orders\_ECOM\_SAN\_FCT")

async def max\_orders(db: Session = Depends(get\_db)):

try:

# Using a subquery to find the max TONG\_SO\_DON\_HANG

subquery = (

db.query(func.max(ECOM\_SAN\_FCT.tong\_so\_don\_hang))

.filter(ECOM\_SAN\_FCT.tong\_so\_don\_hang.isnot(None))

.subquery()

)

# Use the subquery to filter the main query

query = (

db.query(

ECOM\_SAN\_FCT.TENNNT,

ECOM\_SAN\_FCT.mst,

ECOM\_SAN\_FCT.macqtnoinop,

ECOM\_SAN\_FCT.tong\_so\_don\_hang,

)

.filter(ECOM\_SAN\_FCT.tong\_so\_don\_hang == subquery)

.all()

)

if not query:

raise HTTPException(status\_code=404, detail="No data found for the maximum orders." )

result = []

for record in query:

result.append(

{

"TENNNT": record.TENNNT,

"mst": record.mst,

"macqtnoinop": record.macqtnoinop,

"tong\_so\_don\_hang": record.tong\_so\_don\_hang,

}

)

return JSONResponse(content=result)

except Exception as e:

raise HTTPException(status\_code=500, detail=f"Error processing data: {str(e)}")

### Hàm api Thống kê dữu liệu sàn có số giá trị đơn hàng thành công > 60

- Thống kê dữu liệu sàn có số giá trị đơn hàng thành công > 60

- data: tổng hợp dữ liệu sàn có giá trị đơn hàng thành công >60

- KYKKHAI: Kỳ kê khai muốn in

- sau khi thực hiện xong sẽ trả kết quả dưới dạng json

Code xử lý:

@app.get("/value\_per\_ECOM\_SAN\_FCT")

async def value\_per(kykkhai: str = '2022Q4', db: Session = Depends(get\_db)):

try:

# Construct the SQLAlchemy query

data = (

db.query(

ECOM\_SAN\_FCT.TENNNT,

ECOM\_SAN\_FCT.mst,

ECOM\_SAN\_FCT.macqtnoinop,

((ECOM\_SAN\_FCT.TONG\_DONHANGTHANHCONG - ECOM\_SAN\_FCT.TONG\_DONHANGKHONGTHANHCONG) / func.NULLIF(ECOM\_SAN\_FCT.TONG\_DONHANGTHANHCONG, 0)) \* 100

)

.filter(ECOM\_SAN\_FCT.KYKKHAI == kykkhai)

.filter(((ECOM\_SAN\_FCT.TONG\_DONHANGTHANHCONG - ECOM\_SAN\_FCT.TONG\_DONHANGKHONGTHANHCONG) / func.NULLIF(ECOM\_SAN\_FCT.TONG\_DONHANGTHANHCONG, 0)) \* 100 > 60)

.all()

)

if not data:

raise HTTPException(status\_code=404, detail="No data found for the specified conditions.")

# Convert the result to a list of dictionaries

result = [

{

"Tennnt": record[0],

"mst": record[1],

"macqtnoinop": record[2],

"per": float(record[3]) if record[3] is not None else None,

}

for record in data

]

return JSONResponse(content=result)

except Exception as e:

raise HTTPException(status\_code=500, detail=f"Error processing data: {str(e)}")

## Hướng dẫn sử dụng

- Đăng nhập vào ứng dụng qua user và password được cung cấp

- Sau khi đăng nhập sẽ xuất hiện ở một trang dashboard. Đây là nơi chọn các mẫu báo cáo cần in.

- Chọn vào mẫu báo cáo cần in, nhập các tham số cần in, bấm in báo cáo, có thể chọn in trên web hoặc xuất excel

- Sau khi in báo cáo thì bấm đăng xuất.

# Kết luận

## Kết quả đạt được

### Ứng dụng làm được những gì

* Về phía người khai thác: Ứng dụng quản lý thuế giúp cán bộ thuế dễ dàng quản lý tra cứu thông tin một cách dễ dàng, cung cấp thông tin đáng tin cậy.
* Về phía người quản trị: Tổng hợp thông tin theo yêu cầu và đảm bảo thông tin được cập nhật liên tục.

### Kiến thức đã học:

* Phân tích thiết kế: Vẽ biểu đồ lớp, use case, biểu đồ tuần tự, ...
* MySQL: Phân tích và thiết kế dữ liệu, quan hệ giữa các lớp, ...
* Python: Hiểu về hướng đối tượng, truy vấn trên cơ sở dữ liệu và API

### Kiến thức về nghiệp vụ

* Phân tích và thiết kế hệ thống: Nghiên cứu và phân tích yêu cầu của khách hàng, sau đó thiết kế hệ thống phần mềm hoặc phần cứng để đáp ứng các yêu cầu đó.
* Quản lý dự án: Quản lý và điều phối các dự án CNTT. Các nhiệm vụ bao gồm lập kế hoạch, điều phối công việc và giám sát tiến độ của dự án.
* Quản trị hệ thống: Quản lý và duy trì hệ thống mạng, máy chủ và phần cứng CNTT. Nhiệm vụ bao gồm cấu hình, bảo mật và giám sát hệ thống.
* Hỗ trợ người dùng: Giúp đỡ người dùng sử dụng và khắc phục sự cố trong việc sử dụng các ứng dụng.
* An toàn thông tin: Đảm bảo bảo mật thông tin và bảo vệ dữ liệu không để bị rò rỉ. Các nhiệm vụ bao gồm phân tích rủi ro, triển khai biện pháp bảo mật và xử lý các sự cố bảo mật.

### Hướng phát triển

* Phát triển thêm các nhóm báo cáo khác tận dụng từ những bảng đã có.
* Nâng cấp phân quyền cho từng cán bộ vào khai thác dữ liệu.
* Cài đặt thêm phần mềm có thể giúp người dùng dễ nhìn, dễ thao tác hơn như Oracle Business Intelligence, IBM Cognos Analytics, …
* Cài đặt phần mềm hỗ trợ người quản trị dữ liệu như IBM DataStage, Oracle Data Integrator (ODI), Apache Spark, …Giúp người quản trị dễ dàng hơn trong việc xử lý các luồng dữ liệu, đặt lịch chạy.

# TÀI LIỆU THAM KHẢO

1. <https://fastapi.tiangolo.com/reference/status/>
2. https://thuvienphapluat.vn/cong-van/Thuong-mai/Cong-van-3434-TCT-DNNCN-2022-khao-sat-du-lieu-cung-cap-thong-tin-San-thuong-mai-530712.aspx
3. <https://fptcloud.com/mysql-la-gi/>
4. <https://fastapi.tiangolo.com/tutorial/security/oauth2-jwt/>
5. <https://www.mulesoft.com/resources/api/what-is-an-api>