**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**



**KHOA CÔNG NGHỆ THÔNG TIN**

****

**MÔN: ĐẶC TẢ PHẦN MỀM**

**BÁO CÁO ĐỀ TÀI:**

**ĐẶC TẢ PHẦN MỀM QUẢN LÝ CHI TIÊU CÁ NHÂN**

**Giảng viên hướng dẫn: Trần Văn Dũng**

**Danh sách sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Họ và tên** | **Mã sinh viên** |
| **1** | Nguyễn Thị Ngũ | 191200758 |

**Hà Nội - 2022**

**Mục lục**

[**I. Đặc tả yêu cầu** 3](#_Toc123245636)

[**1. Khảo sát** 3](#_Toc123245637)

[**2. Mục đích** 3](#_Toc123245638)

[**3. Phạm vi** 3](#_Toc123245639)

[**II. Đặc tả chức năng** 3](#_Toc123245640)

[**1. Đặc tả chức năng chính** 3](#_Toc123245641)

[**2. Đặc tả các chức năng cơ sở:** 4](#_Toc123245642)

[**3. Đặc tả cơ sở dữ liệu** 6](#_Toc123245643)

[**III. Cơ sở dữ liệu vật lý** 12](#_Toc123245644)

[**1. Câu lệnh tạo bảng** 12](#_Toc123245645)

[**2. Câu lệnh tạo dữ liệu** 14](#_Toc123245646)

[**3. Diagram** 17](#_Toc123245647)

[**IV. Tạo 2 form nhập liệu** 17](#_Toc123245648)

[**1. Form cập nhật ví\_nguồn thu** 17](#_Toc123245649)

[**2. Form thêm giao dịch mới.** 18](#_Toc123245650)

# **I. Đặc tả yêu cầu**

## **1. Khảo sát**

* Một số app dùng để quản lý chi tiêu các nhân đang được ưu chuộc như: Money love, Quản lý chi tiêu cá nhân Misa, Spendee, Quản lý chi tiêu cá nhân Mint, Pocket Guard, Money Mate, Fast Budget, Home Budget, Money Manage,….
* Đặc điểm chung của các app là là chạy trên nền tảng di động, cung cấp cho người dùng những tính năng như:

+ Ghi chép số liệu chi tiêu hàng ngày

+ Tạo báo cáo số tiền chi tiêu hàng tháng

+ Thiết lập quyền hạn mức chi từng danh mục

+ Thông kê tiền cho từng danh mục,….

## **2. Mục đích**

* Tài liệu này đưa ra các đặc tả chi tiết yêu cầu phần mềm cho hệ thống quản lý chi tiêu cá nhân. Hệ thống cho phép người dùng có thể quản lý chi tiêu của cá nhân hoặc chi tiêu của một hội nhóm. Có thể tạo ra các loại ví khác nhau, các danh mục cần chi khác nhau cũng mức hạn chi tiêu cho từng danh mục.
* Cung cấp cho người dùng một trang web có giao diện thân thiện, dễ dùng, đáp ứng được các nghiệp vụ cần có của một phần mềm quản lý chi tiêu cơ bản.

## **3. Phạm vi**

* Phần mềm nhắm tới các đối tượng khác nhau, từ sinh viên đến người đi làm, từ giám đốc đến công nhân.

# **II. Đặc tả chức năng**

## **1. Đặc tả chức năng chính**

1.1: Quản lý người dùng: Lưu trữ thông tin người dùng để phục vụ cho việc truy vấn các khoản chi tiêu.

1.2: Quản lý giao dịch: Cho biết khoản dư của người dùng, tra cứu lại thông tin các khoản chi cũng như cập nhật lại. Từ đó giúp cho người dùng có thể kiểm soát được chi tiêu của bản thân cũng như hộ nhóm

1.3: Quản lý danh mục: Tạo các danh mục cần thiết để quản lý chi tiêu một cách minh bạc, rõ ràng nhất.

1.4: Quản lý ví: Tra cứu thông tin ví để có thể điều chỉnh thông tin kịp thời.

* **Phân rã chức năng**
  1. : **Quản lý người dùng**: Cập nhật tài khoản, đăng ký, đăng nhập.
  2. : **Quản lý giao dịch**: Tra cứu thông tin giao dịch, cập nhật thông tin khoản chi, thống kê chi tiêu theo ngày, thống kê chi tiêu theo khoảng thời gian, thống kê chi tiêu theo ví.
  3. : **Quản lý danh mục**: Tra cứu thông tin danh mục, cập nhật thông tin danh mục.
  4. : **Quản lý ví**: Tra cứu ví, cập nhật ví.

## **2. Đặc tả các chức năng cơ sở:**

1.1: **Cập nhật tài khoản**: Thêm sửa xóa một tài khoản với các thông tin như: Tài khoản, mật khẩu, họ tên, giới tính, email, số điện thoại,…

1.2: **Đăng ký**: Mở form cho người đăng ký các thông tin cơ bản như: Tên tài khoản, mật khẩu, họ tên, giới tính, số điện thoại,…

1.3: **Đăng nhập**: Mở form cho người dùng đăng nhập vào hệ thống. Xác thực bằng tên tài khoản và mật khẩu.

2.1: **Tra cứu thông tin giao dịch**: Nhập loại danh mục hiển thị các khoản chi tương ứng.

2.2: **Cập nhật thông tin khoản chi**: Thêm sửa xóa các khoản chi.

2.3: **Thống kê chi tiêu theo ngày**: Nhập ngày cần thống kê, hệ thống thống kê các khoản chi tương ứng và hiển thị ra.

2.4: **Thống kê chi tiêu theo khoảng thời gian**: Nhập ngày bắt đầu và ngày kết thúc. Hệ thống hiển thị các thông tin cơ bản các khoản chi trong đó.

2.5: **Thống kê chi tiêu theo ví**: Chọn một loại ví. Hệ thống hiển thị biến động của ví

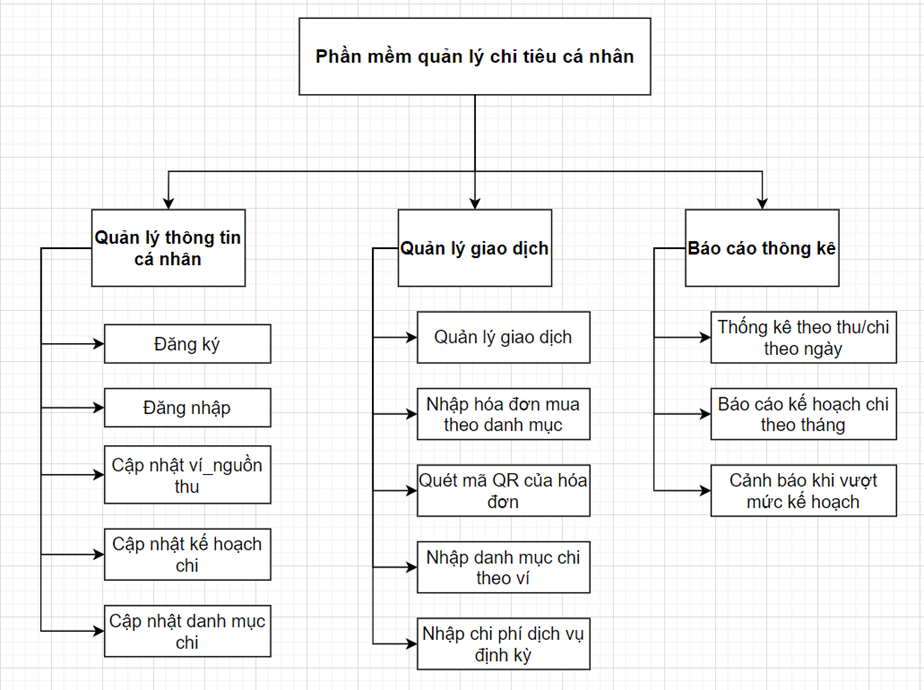
3.1: **Tra cứu thông tin danh mục**: Nhập tên danh mục, hiển thị thông tin cơ bản của danh mục

3.2: **Cập nhật thông tin danh mục**: Thêm sửa xóa thông tin danh mục

4.1: **Tra cứu ví**: Hiển thị các thông tin cơ bản của ví như số

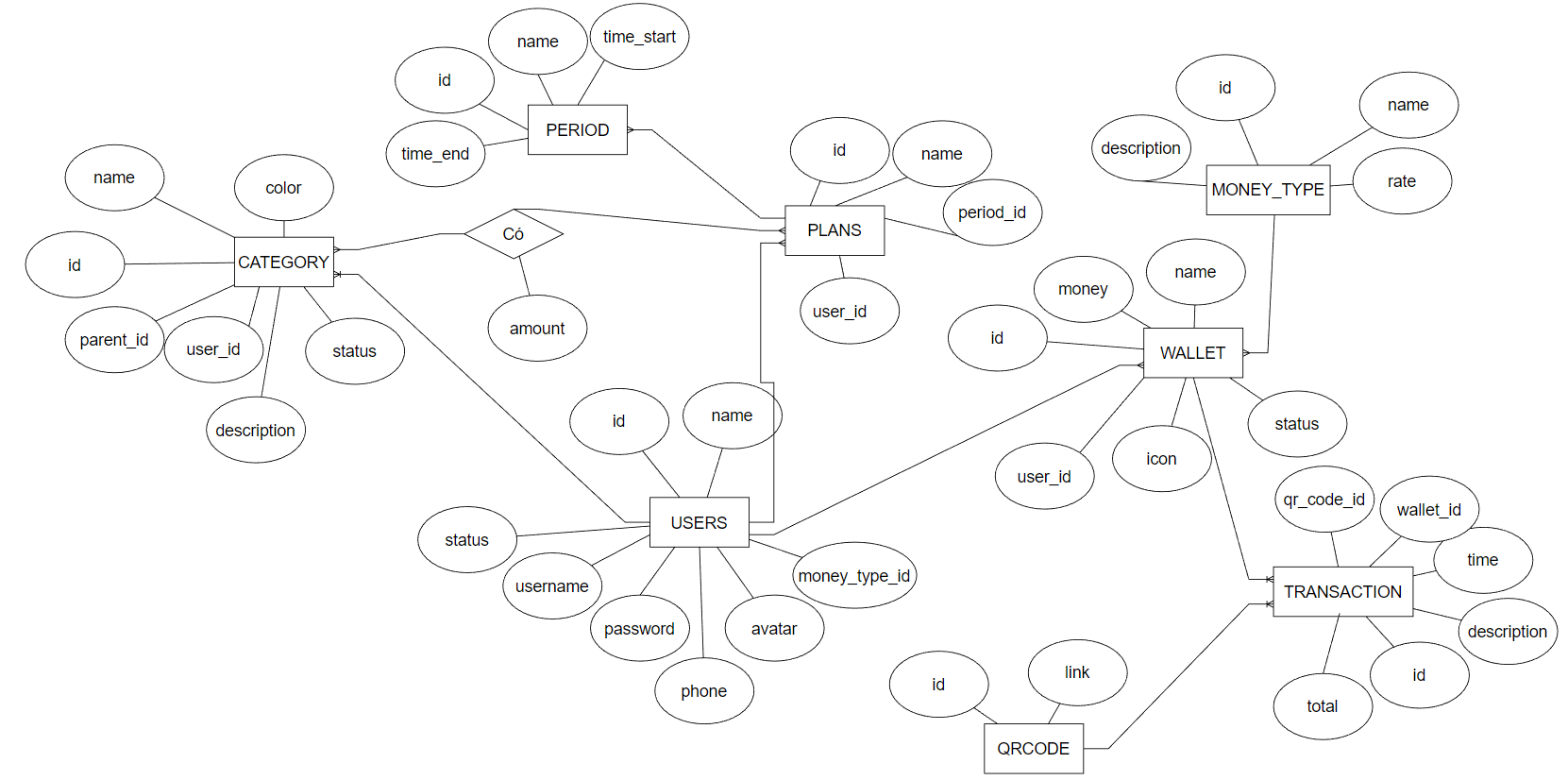
4.2: **Cập nhật ví**: Thêm sửa xóa thông tin một ví.

**Sơ đồ phân rã chức năng:**



*Hình 1: Sơ đồ phân rã chức năng.*

**Sơ đồ mô hình thực thể:**

****

*Hình 2: Sơ đồ mô hình thực thể*

## **3. Đặc tả cơ sở dữ liệu**

3.1 Xác định thực thể

- CATEGORY: Lưu thông tin các loại danh mục thu/chi

- MONEY\_TYPE: Lưu danh mục loại tiền

- PERMISSIONS: Lưu thông tin các quyền hạn

- ROLES: Lưu thông tin vai trò của các tài khoản

- ROLES\_PERMISSIONS: Bảng phụ liên kết giữa role và permissions(1 tài khoản có thể có nhiều quyền hạn)

- TRANSACTION: Lưu thông tin giao dịch

- USERS: Lưu thông tin người dùng

- USER\_ROLE: Bảng phụ liên kết giữa users và role(1 tài khoản có thể có nhiều vai trò)

- WALLET: Lưu thông tin ví

- WALLET\_USERS: Lưu thông tin ví của người mỗi người dùng

- EDIT\_HISTORY: Lưu thông tin các lần đã từng sửa

3.2 Xác định các thuộc tính và quan hệ

* CATEGORY: category\_id, color(red, blue, …), description, name, status(hoạt động, chưa hoạt động, xóa), user\_id, parent\_id

+ unique keys (#name, #user\_id)

* MONEY\_TYPE: money\_type\_id, name(dolla, VND, …), rate
* PERMISSIONS: permission\_id, name(get\_users, get\_user, create\_user, update\_user, delete\_user, get\_role, get\_roles, create\_role, update\_role, delete\_role)

+ unique keys(#name)

* ROLES: role\_id, name(admin, user)
* ROLES\_PERMISSIONS: role\_id, permission\_id
* TRANSACTIONS: transation\_id, description, time, total, wallet\_id, user\_id

+ unique keys(#time, #user\_id)

* USERS: user\_id, avatar, name, username, password, phone, status

+ unique keys(#username, #password)

* USERS\_ROLE: user\_role\_id, user\_id, role\_id

+ unique keys(#user\_id, #role\_id)

* WALLET: wallet\_id, icon, money, name, money\_type\_id, status

+ unique keys(#user\_id, #name)

* WALLET\_USERS: wallet\_user\_id, wallet\_id, user\_id

+ unique keys(#wallet\_id, #user\_id)

* EDIT\_HISTORY: edit\_history\_id, transaction\_id, user\_id, time, old\_money, new\_money

+ unique keys(#user\_id, #time)

3.3 Mô tả các bảng

* Bảng “CATEGORY”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| category\_id | Int | Not null | PRIMARY KEY – mã danh mục |
| name | Varchar(20) | Not null | Tên danh mục |
| color | Varchar(10) | Null | Màu thể hiện độ quan trọng của danh mục |
| description | LongText | Null | Mô tả cụ thể của danh mục |
| status | Int | Not null | Trạng thái |
| user\_id | Int | Null | Mã người tạo |
| Parent\_id | Int | Null | Mã danh mục cha |

* Bảng “MONEY\_TYPE”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| Money\_type\_id | Int | Not null | PRIMARY KEY – mã loại tiền |
| Name | Varchar(10) | Not null | Tên loại tiền |
| Rate | Double | Null | Tỷ giá quy đổi |

* Bảng ” PERMISSIONS”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| Permission\_id | Int | Not null | PRIMARY KEY – mã quyền |
| Name | Varchar(20) | Not null | Tên quyền |

* Bảng “ROLE”:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| Role\_id | Int | Not null | PRIMARY KEY – mã vai trò |
| Name | Varchar(20) | Not null | Tên vai trò |

* Bảng “ROLES\_PERMISSIONS: role\_id, permission\_id”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| Role\_permission\_id | Int | Not null | PRIMARY KEY – số thứ tự |
| Role\_id | Int | Not null | Mã vai trò |
| Permission\_id | Int | Not null | Mã quyền |

* Bảng “TRANSACTIONS”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| transaction \_id | Int | Not null | PRIMARY KEY – mã giao dịch |
| Description | Int | Null | Chú thích của giao dịch |
| Time | Date | Not Null | Thời gian tạo giao dịch |
| Total | Double | Null | Tổng tiền |
| User\_id | Int | Not null | Người lập giao dịch |
| Wallet\_id | Int | Not null | Mã ví |
| User\_id | Int | Not null | Mã tài khoản |

* Bảng “Users”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| User\_id | Int | Not null | PRIMARY KEY – mã tài khoản |
| Name | Varchar(50) | Null | Họ và tên đầy đủ của tài khoản |
| Username | Varchar(50) | Not null | Tên đăng nhập |
| Password | Varchar(20) | Not null | Mật khẩu |
| Avatar | Varchar(100) | Null | Ảnh đại diện |
| Phone | Varchar(10) | Null | Số điện thoại |
| Status | Bit | Defaut(1) | Trạng thái hoạt động |

* Bảng “USERS\_ROLE”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| User\_role\_id | Int | Not null | PRIMARY KEY – số thứ tự |
| User\_id | Int | Not null | Mã tài khoản |
| Role\_id | Int | Not null | Mã vai trò |

* Bảng “WALLET”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| Wallet\_id | Int | Not null | PRIMARY KEY – mã ví |
| Name | Varchar(100) | Null | Tên ví |
| Icon | Varchar(100) | Null | Icon ví |
| Money | Double | Null | Tiền của ví |
| Money\_type\_id | Int | Null | Mã loại tiền |
| Status | Bit | Defaut(1) | Trạng thái |

* Bảng “WALLET\_USERS”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| Wallet\_user\_id | Int | Not null | PRIMARY KEY – số thứ tự |
| Wallet\_id | Int | Not null | Mã ví |
| User\_id | Int | Not null | Mã tài khoản |

* Bảng “EDIT\_HISTORY”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Nullable** | **Mô tả** |
| Edit\_history\_id | Int | Not null | PRIMARY KEY – số thứ tự |
| Transaction\_id | Int | Not null | Mã giao dịch |
| User\_id | Int | Not null | Mã tài khoản |
| Time | Date | Not null | Thời gian sửa |
| Old\_money | Double | Null | Số tiền cũ |
| New\_money | Double | Null | Số tiền mới |

# **III. Cơ sở dữ liệu vật lý**

## **1. Câu lệnh tạo bảng**

CREATE TABLE `users` (

`id` bigint(20) NOT NULL AUTO\_INCREMENT,

`avatar` varchar(255) DEFAULT NULL,

`email` varchar(50) NOT NULL,

`name` varchar(50) DEFAULT NULL,

`password` varchar(255) NOT NULL,

`phone` varchar(10) DEFAULT NULL,

`status` int(11) DEFAULT 1,

`username` varchar(50) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `UK\_6dotkott2kjsp8vw4d0m25fb7` (`email`),

UNIQUE KEY `UK\_r43af9ap4edm43mmtq01oddj6` (`username`)

) ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=utf8mb4;

CREATE TABLE `wallet` (

`id` bigint(20) NOT NULL AUTO\_INCREMENT,

`icon` varchar(255) DEFAULT NULL,

`money` double DEFAULT NULL,

`name` varchar(50) DEFAULT NULL,

`status` int(11) DEFAULT 1,

`id\_user` bigint(20) DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `category` (

`id` bigint(20) NOT NULL AUTO\_INCREMENT,

`color` varchar(255) DEFAULT NULL,

`description` longtext DEFAULT NULL,

`name` varchar(50) DEFAULT NULL,

`status` int(11) DEFAULT 1,

`user\_id` bigint(20) DEFAULT NULL,

`id\_parent` bigint(20) DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `FK7ffrpnxaflomhdh0qfk2jcndo` (`user\_id`),

KEY `FK\_idx` (`id\_parent`),

KEY `FK\_idx1` (`id\_parent`),

CONSTRAINT `FK1` FOREIGN KEY (`id\_parent`) REFERENCES `category` (`id`),

CONSTRAINT `FK7ffrpnxaflomhdh0qfk2jcndo` FOREIGN KEY (`user\_id`) REFERENCES `users` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `period` (

`id` bigint(20) NOT NULL,

`name` varchar(20) DEFAULT NULL,

`id\_parent` bigint(20) DEFAULT NULL,

`start\_time` date DEFAULT NULL,

`end\_time` date DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `FK\_idx` (`id\_parent`),

CONSTRAINT `FK` FOREIGN KEY (`id\_parent`) REFERENCES `period` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `plan` (

`id` bigint(20) NOT NULL,

`name` varchar(20) DEFAULT NULL,

`period\_id` bigint(20) DEFAULT NULL,

`user\_id` bigint(20) DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `FK\_Plan\_User\_idx` (`user\_id`),

KEY `FK\_Plan\_Period\_idx` (`period\_id`),

CONSTRAINT `FK\_Plan\_Period` FOREIGN KEY (`period\_id`) REFERENCES `period` (`id`),

CONSTRAINT `FK\_Plan\_User` FOREIGN KEY (`user\_id`) REFERENCES `users` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `plan\_category` (

`id` bigint(20) NOT NULL,

`plan\_id` bigint(20) DEFAULT NULL,

`category\_id` bigint(20) DEFAULT NULL,

`amount` double DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `FK\_Plan\_idx` (`plan\_id`),

KEY `FK\_Category\_idx` (`category\_id`),

CONSTRAINT `FK\_Category` FOREIGN KEY (`category\_id`) REFERENCES `category` (`id`),

CONSTRAINT `FK\_Plan` FOREIGN KEY (`plan\_id`) REFERENCES `plan` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `qr\_code` (

`id` bigint(20) NOT NULL AUTO\_INCREMENT,

`link` varchar(200) DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `transaction` (

`id` bigint(20) NOT NULL AUTO\_INCREMENT,

`description` longtext DEFAULT NULL,

`time` datetime DEFAULT NULL,

`total` double DEFAULT NULL,

`wallet\_id` bigint(20) DEFAULT NULL,

`category\_id` bigint(20) DEFAULT NULL,

`id\_qr` bigint(20) DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `FKtfwlfspv2h4wcgc9rjd1658a6` (`wallet\_id`),

KEY `FK\_Transaction\_Category\_idx` (`category\_id`),

KEY `id\_qr` (`id\_qr`),

CONSTRAINT `FK\_Transaction\_Category` FOREIGN KEY (`category\_id`) REFERENCES `category` (`id`),

CONSTRAINT `FKtfwlfspv2h4wcgc9rjd1658a6` FOREIGN KEY (`wallet\_id`) REFERENCES `wallet` (`id`),

CONSTRAINT `transaction\_ibfk\_1` FOREIGN KEY (`id\_qr`) REFERENCES `qr\_code` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

## **2. Câu lệnh tạo dữ liệu**

INSERT INTO `` (`id`,`avatar`,`email`,`name`,`password`,`phone`,`status`,`username`) VALUES (1,'avt.meo.cute','meocon@gmail.com','Mèo','Meo@123','0362871301',1,'MeoCon');

INSERT INTO `` (`id`,`avatar`,`email`,`name`,`password`,`phone`,`status`,`username`) VALUES (2,'avt.cho.ngao','chongao@gmail.com','Chó','Cho@123', 0394229171, 1,'ChoNgao');

INSERT INTO `` (`id`,`avatar`,`email`,`name`,`password`,`phone`,`status`,`username`) VALUES (3,'avt.heo.map','heomap@gmail.com','Heo','Heo@123','0987654321',0,'HeoMap');

INSERT INTO `` (`id`,`avatar`,`email`,`name`,`password`,`phone`,`status`,`username`) VALUES (4,'avt.khi.buon','khibuon@gmail.com','Khỉ','Khi@123','0764384197',1,'KhiBuon');

INSERT INTO `` (`id`,`avatar`,`email`,`name`,`password`,`phone`,`status`,`username`) VALUES (5,'avt.gau.co.don','gaucodon@gmail.com','Gấu ','Gaucodon@123','0864329874',1,'GauCoDon');

INSERT INTO `wallet`.`qr\_code` (`link`) VALUES ('qr.code.1');

INSERT INTO `wallet`.`qr\_code` (`link`) VALUES ('qr.code.2');

INSERT INTO `wallet`.`qr\_code` (`link`) VALUES ('qr.code.3');

INSERT INTO `wallet`.`qr\_code` (`link`) VALUES ('qr.code.4');

INSERT INTO `wallet`.`qr\_code` (`link`) VALUES ('qr.code.5');

INSERT INTO `wallet`.`qr\_code` (`link`) VALUES ('qr.code.6');

INSERT INTO `wallet`.`qr\_code` ( `link`) VALUES ('qr.code.7');

INSERT INTO `wallet`.`qr\_code` (`link`) VALUES ('qr.code.8');

INSERT INTO `wallet`.`wallet` (`id`, `icon`, `money`, `name`, `status`, `id\_user`) VALUES ('1', 'icon1', '10000000', 'BIDV', '1', '1');

INSERT INTO `wallet`.`wallet` (`id`, `icon`, `money`, `name`, `status`, `id\_user`) VALUES ('2', 'icon2', '1500000', 'TPBank', '1', '1');

INSERT INTO `wallet`.`wallet` (`id`, `icon`, `money`, `name`, `status`, `id\_user`) VALUES ('3', 'icon3', '2000000', 'Viettel money', '1', '1');

INSERT INTO `wallet`.`wallet` (`id`, `icon`, `money`, `name`, `status`, `id\_user`) VALUES ('4', 'icon4', '5000000', 'HDBank', '1', '1');

INSERT INTO `wallet`.`wallet` (`id`, `icon`, `money`, `name`, `status`, `id\_user`) VALUES ('5', 'icon2', '2500000', 'TPBank', '1', '2');

INSERT INTO `wallet`.`wallet` (`id`, `icon`, `money`, `name`, `status`, `id\_user`) VALUES ('6', 'icon3', '5000000', 'Viettel money', '0', '2');

INSERT INTO `wallet`.`wallet` (`id`, `icon`, `money`, `name`, `status`, `id\_user`) VALUES ('7', 'icon5', '10000000', 'VPBank', '1', '3');

INSERT INTO `wallet`.`wallet` (`id`, `icon`, `money`, `name`, `status`, `id\_user`) VALUES ('8', 'icon2', '23000000', 'TPBank', '1', '5');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`) VALUES ('1', 'red', 'Ăn sáng, trưa, tối', 'Ăn uống', '1', '1');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`) VALUES ('2', 'bule', 'Mua các đồ dùng cần thiết cho sinh hoạt', 'Mua sắm', '1', '1');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`, `id\_parent`) VALUES ('3', 'gray', 'Mỹ phẩm, son phấn, dưỡng da', 'Mỹ phẩm', '1', '1', '2');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`, `id\_parent`) VALUES ('4', 'yellow', 'Quấn áo, giày dép', 'Quần áo', '0', '1', '2');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`) VALUES ('5', 'black', 'Tiền cho các cuộc vui chơi', 'Vui chơi', '1', '1');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`) VALUES ('6', 'green', 'Đổ xăng xe, bảo dưỡng xe, lộ phí đi lại', 'Đi lại', '1', '1');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`) VALUES ('7', 'pink', 'Các khoản chi tiêu khác', 'Khác', '1', '1');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`, `id\_parent`) VALUES ('8', 'gray', 'Đi uống cafe', 'Cafe', '0', '1', '5');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`, `id\_parent`) VALUES ('9', 'red', 'Uống trà sữa', 'Trà sữa', '1', '1', '1');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`, `id\_parent`) VALUES ('10', 'green', 'Đi xe ôm công nghệ', 'Grab', '1', '3', '6');

INSERT INTO `wallet`.`category` (`id`, `color`, `description`, `name`, `status`, `user\_id`, `id\_parent`) VALUES ('11', 'pink', 'Vui chơi, sn của bạn bè, người yêu', 'Bạn bè', '1', '4', '5');

INSERT INTO `wallet`.`period` (`id`, `name`, `start\_time`, `end\_time`) VALUES ('1', '1 tháng', '2022-01-01', '2022-02-01');

INSERT INTO `wallet`.`period` (`id`, `name`, `start\_time`, `end\_time`) VALUES ('2', '3 tháng', '2022-01-01', '2022-04-01');

INSERT INTO `wallet`.`period` (`id`, `name`, `start\_time`, `end\_time`) VALUES ('3', '6 tháng', '2022-01-01', '2022-07-01');

INSERT INTO `wallet`.`period` (`id`, `name`, `start\_time`, `end\_time`) VALUES ('4', '12 tháng', '2022-01-01', '2023-01-01');

INSERT INTO `wallet`.`period` (`id`, `name`, `start\_time`, `end\_time`) VALUES ('5', '10 năm', '2022-01-01', '2032-01-01');

INSERT INTO `wallet`.`plan` (`id`, `name`, `user\_id`) VALUES ('1', 'Mua điện thoại ', '1');

INSERT INTO `wallet`.`plan` (`id`, `name`, `user\_id`) VALUES ('2', 'Tương lai', '1');

INSERT INTO `wallet`.`plan` (`id`, `name`, `user\_id`) VALUES ('3', 'Cố định', '2');

INSERT INTO `wallet`.`plan` (`id`, `name`, `period\_id`, `user\_id`) VALUES ('4', 'Mua xe', '3', '3');

INSERT INTO `wallet`.`plan` (`id`, `name`, `period\_id`, `user\_id`) VALUES ('5', 'Cưới vợ', '2', '5');

INSERT INTO `wallet`.`transaction` (`id`, `time`, `total`, `wallet\_id`, `category\_id`, `id\_qr`) VALUES ('1', '2022-19-10', '10000', '1', '1', '1');

INSERT INTO `wallet`.`transaction` (`id`, `description`, `time`, `total`, `wallet\_id`, `category\_id`, `id\_qr`) VALUES ('2', 'Ăn xiên bẩn', '2022-19-10', '20000', '2', '5', '2');

INSERT INTO `wallet`.`transaction` (`id`, `description`, `time`, `total`, `wallet\_id`, `category\_id`, `id\_qr`) VALUES ('3', 'Mua gói mạng', '2022-19-10', '56000', '3', '7', '3');

INSERT INTO `wallet`.`transaction` (`id`, `description`, `time`, `total`, `wallet\_id`, `category\_id`, `id\_qr`) VALUES ('4', 'Ăn trưa', '2022-19-10', '15000', '4', '1', '2');

INSERT INTO `wallet`.`transaction` (`id`, `time`, `total`, `wallet\_id`, `category\_id`, `id\_qr`) VALUES ('5', '2022-19-10', '70000', '2', '6', '3');

INSERT INTO `wallet`.`transaction` (`id`, `description`, `time`, `total`, `wallet\_id`, `category\_id`, `id\_qr`) VALUES ('6', 'Ăn sáng', '2022-19-10', '10000', '1', '1', '4');

INSERT INTO `wallet`.`plan\_category` (`id`, `plan\_id`, `category\_id`, `amount`) VALUES ('1', '1', '1', '1000000');

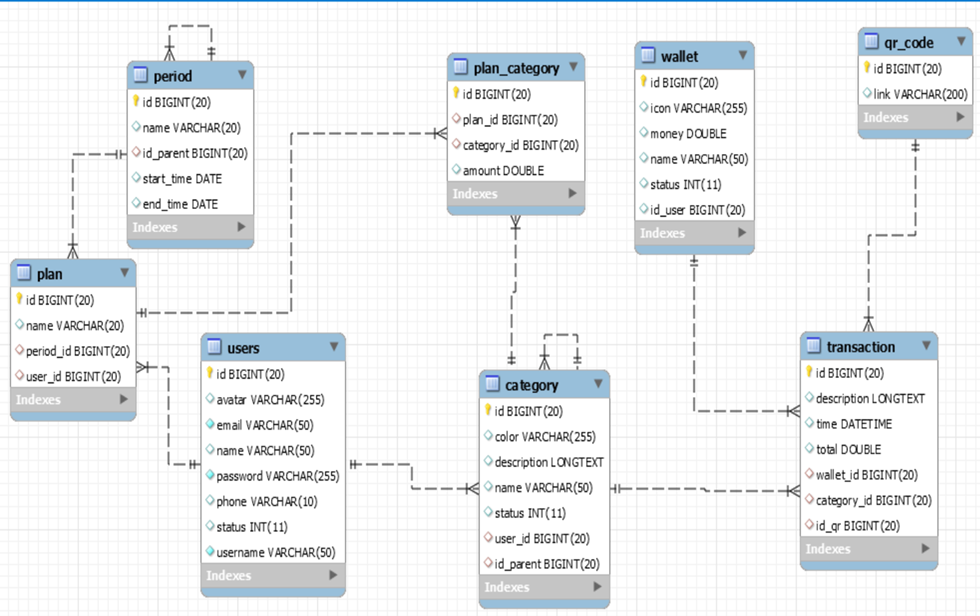
INSERT INTO `wallet`.`plan\_category` (`id`, `plan\_id`, `category\_id`, `amount`) VALUES ('2', '2', '5', '2500000');

INSERT INTO `wallet`.`plan\_category` (`id`, `plan\_id`, `category\_id`, `amount`) VALUES ('3', '3', '2', '2000000');

INSERT INTO `wallet`.`plan\_category` (`id`, `plan\_id`, `category\_id`, `amount`) VALUES ('4', '4', '7', '2000000');

INSERT INTO `wallet`.`plan\_category` (`id`, `plan\_id`, `category\_id`, `amount`) VALUES ('5', '5', '3', '2000000');

## **3. Diagram**

****

# **IV. Tạo 2 form nhập liệu**

## **1. Form cập nhật ví\_nguồn thu**

1. Thông tin chung

- Mục đích: cập nhật thông tin ví mới (thêm, sửa, xóa ví).

- Ngữ cảnh thực hiện: Khi lần đầu tạo nick và muốn tạo một ví mới để sử dụng. Hoặc muốn sửa thông tin của ví.

b. Thông tin chi tiết

- Điều kiện thực hiện: Khi đã đăng nhập, nếu tạo ví mới thì số lượng ví phải chưa đạt giới hạn. Nếu sửa thông tin ví thì tên không được trùng với các ví đã tồn tại.

- Trình thực hiện:

+ Đăng nhập vào hệ thống.

+ Chọn chức năng cập nhật ví. Hệ thống hiển thị tất cả các ví và có các nút thêm, sửa, xóa.

+ Chọn xóa thì cần xác nhận từ phía người dùng.

+ Chọn sửa thì hiển thị thông tin cũ lên và sửa. Các thông tin có thể sửa như: Tên ví, hình ảnh, số dư của ví, loại tiền.

+ Chọn mới thì hiển thị form cho người dùng nhập. Hệ thống kiểm tra thông tin xem có bị trùng thông tin không và số lượng ví có hợp lệ không.

c. Các thông tin chi tiết

- Thông tin về các ví đã tồn tại.

+ Tên ví

+ Ảnh minh họa

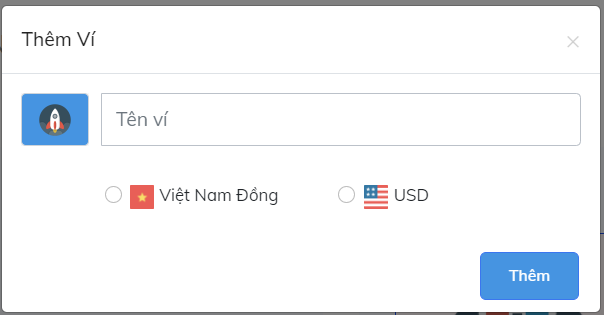
+ Số tiền hiện có trong ví

+ Đơn vị tiền tệ

* Thông tin về loại tiền.

+ Tên loại tiền

+ Ảnh minh họa



## **2. Form thêm giao dịch mới.**

1. Thông tin chung

* Mục đích: Module này thực hiện chức năng thêm giao dịch.
* Ngữ cảnh thực hiện: Khi người dùng cần lưu trữ lại thông tin giao dịch.

1. Thông tin chi tiết

* Điều kiện thực hiện: Đã đăng nhập vào hệ thông và ví tiền có số dư hơn hơn 0.
* Trình tự thực hiện:

+ Đầu tiên, người dùng phải đăng nhập vào hệ thống.

+ Chọn chức năng thêm giao dịch.

+ Người dùng điền form giao dịch bao gồm: Thời gian, dùng ví nào, mô tả thêm, tổng tiền, ảnh minh họa hoặc mã qr.

+ Nếu số tiền giao dịch nhỏ hơn số dư hiện tại của ví thì thêm thông tin giao dịch mới vào bảng, đồng thời cập nhật lại số dư hiện tại của ví. Thông báo cho người dùng “Thêm giao dịch thành công”. Ngược lại thì thông báo “Số dư không đủ” và hủy giao dịch.

1. Các thông tin chi tiết

* Thông tin cần truy vấn để hiển thị

+ Tên ví

+ Ảnh minh họa

+ Số tiền hiện có trong ví

+ Đơn vị tiền tệ

+ Tên danh mục

+ Ảnh danh mục

* Thông tin mong muốn đầu ra

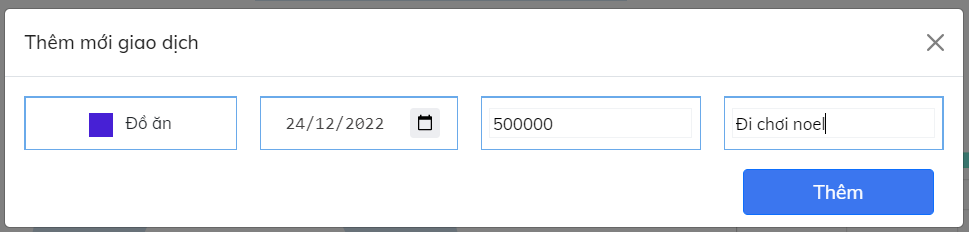
+ Tên giao dịch

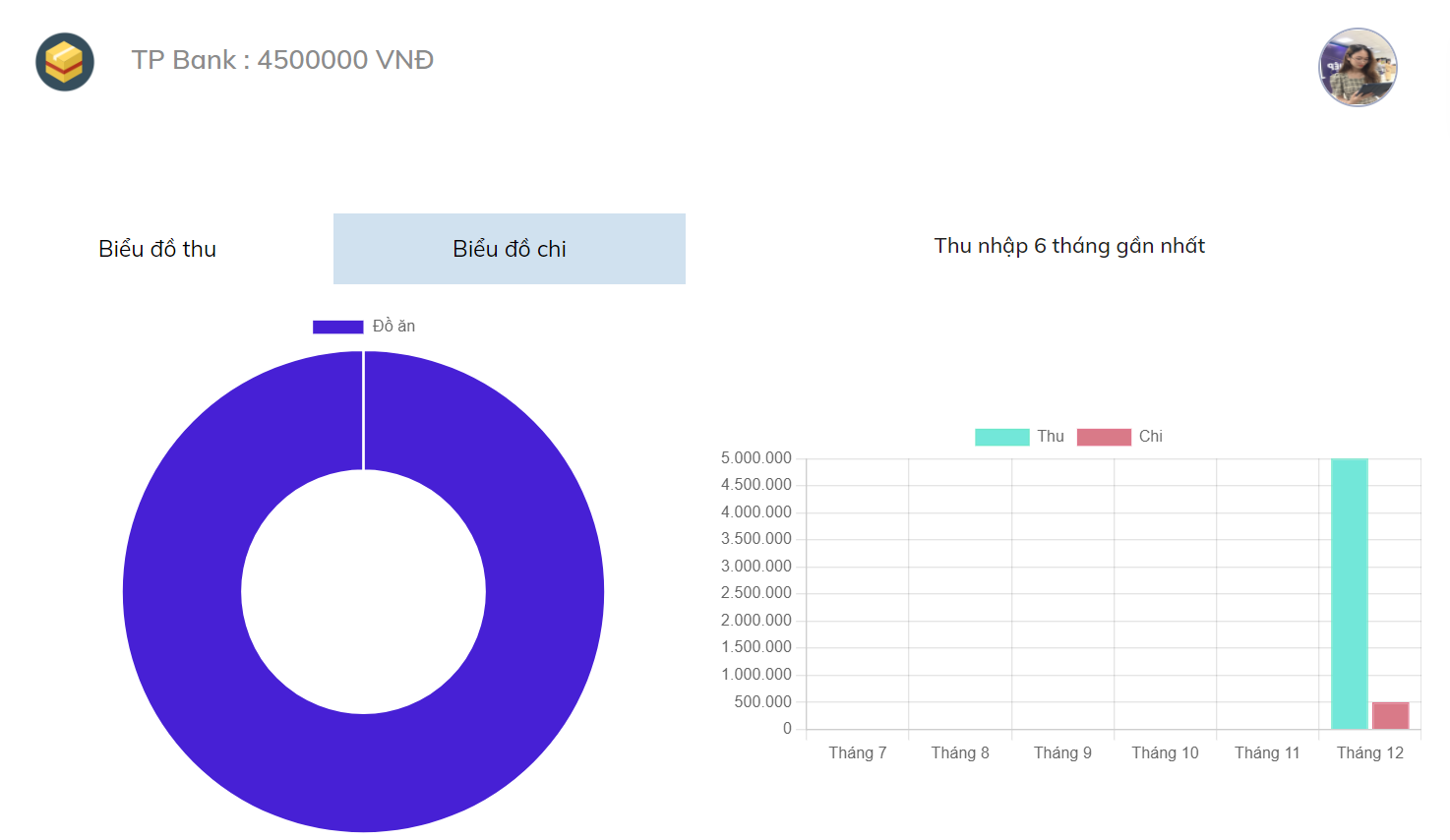
+ Hình ảnh minh họa của giao dịch

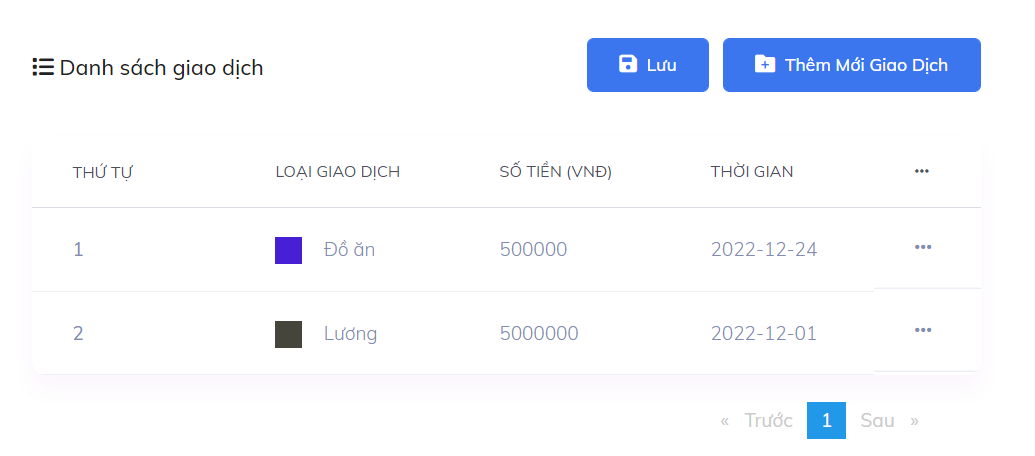
+ Số tiền giao dịch

+ Thời gian giao dịch

+ Mô tả của giao dịch

+ Thông tin số dư của ví





**3. From thay đổi loại tiền**

a. Thông tin chung

* Mục đích: Module này thực hiện chức năng thay đổi loại tiền trong ví.
* Ngữ cảnh thực hiện: Khi người dùng muốn chuyển đổi loại tiền để tiện sử dụng trong nhiều quốc gia khác nhau.

b. Thông tin chi tiết

* Điều kiện thực hiện: Đã đăng nhập vào hệ thông.
* Trình tự thực hiện:

+ Đầu tiên, người dùng phải đăng nhập vào hệ thống.

+ Mở ví, sau đó vào “sửa ví” và chọn loại tiền mong muốn.

c. Các thông tin chi tiết

* Thông tin cần truy vấn để hiển thị

+ Tên ví

+ Ảnh minh họa

+ Số tiền hiện có trong ví

+ Đơn vị tiền tệ

+ Tên danh mục

+ Ảnh danh mục

* Thông tin mong muốn đầu ra
* Ví

+ Đơn vị tiền tệ của ví đã được thay đổi

+ Số tiền trong ví được thay đổi tương ứng

* Giao dịch

+ Tên giao dịch

+ Hình ảnh minh họa của giao dịch

+ Số tiền giao dịch đã được chuyển sang đơn vị tiền tệ mong muốn

+ Thời gian giao dịch

+ Mô tả của giao dịch

