TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

Viện Công nghệ thông tin và Truyền thông

Tài liệu đặc tả yêu cầu phần mềm

(Software Requirement Specification – SRS)

**Đề tài : Quản lý chi tiêu cá nhân**

**Smart Money**

Môn: Phát triển phần mềm theo kỹ năng ITSS

*Giáo viên hướng dẫn :Thầy Nguyễn Mạnh Tuấn*

*Nhóm số 4 - Danh sách sinh viên :*

*Bùi Thu Thủy Linh – 20184131*

*Nguyễn Hoàng Sơn – 20184187*

*Nguyễn Hữu Tuấn – 20184221*

*Phạm Thị Mai Tuyết – 20184227*

*Hà Nội, ngày 12 tháng  10 năm 2021*

Contents

[Giới thiệu đề tài](#_heading=h.8zqiumeacxzt) **3**

[Mục đích](#_heading=h.3znysh7) 4

[Đối tượng người dùng](#_heading=h.2et92p0) 4

[Khả năng phát triển](#_heading=h.tyjcwt) 5

[Từ điển thuật ngữ](#_heading=h.3dy6vkm) 5

[Tài liệu tham khảo](#_heading=h.1t3h5sf) 5

[Đặc tả usecase](#_heading=h.4d34og8) **5**

[Đặc tả chức năng](#_heading=h.2s8eyo1) 6

[Biểu đồ Usecase tổng quan](#_heading=h.3rdcrjn) 7

[Biểu đồ Usecase phân rã](#_heading=h.26in1rg) 8

[Phân rã use case “Quản lý người dùng”](#_heading=h.lnxbz9) 8

[Phân rã use case “CRUD của giao dịch”](#_heading=h.35nkun2) 9

[Phân rã use case “CRU của ví”](#_heading=h.1ksv4uv) 9

[Đặc tả use case](#_heading=h.44sinio) 10

[Usecase “Đăng nhập, đăng ký”](#_heading=h.2jxsxqh) 10

[Usecase “Quản lý ví”](#_heading=h.z337ya) 10

[Usecase “Quản lý giao dịch”](#_heading=h.3j2qqm3) 10

[Usecase “Quản lý ngân sách”](#_heading=h.hcvnrlr1u9ro) 11

[Usecase “Xem thống kê chi tiêu”](#_heading=h.yqslb2jugxv2) 11

[Usecase “Quản lý người dùng”](#_heading=h.1y810tw) 12

[Thiết kế hệ thống](#_heading=h.4i7ojhp) **13**

[Activity diagram](#_heading=h.2xcytpi) 13

[Quy trình tạo giao dịch mới](#_heading=h.1ci93xb) 13

[Quy trình chỉnh sửa một giao dịch](#_heading=h.3whwml4) 14

[Quy trình xóa một giao dịch](#_heading=h.2bn6wsx) 16

[Quy trình tạo một ngân sách](#_heading=h.qsh70q) 17

[Quy trình xóa một ngân sách](#_heading=h.3as4poj) 19

[Quy trình tạo ví](#_heading=h.1pxezwc) 20

[Tạo ví tiền mặt không liên kết tài khoản ngân hàng](#_heading=h.39mxlsgazgo) 20

[Quy trình người dùng thay đổi thông tin](#_heading=h.49x2ik5) 21

[Quy trình xem thống kê](#_heading=h.2p2csry) 22

[Quy trình quản lý người dùng (Admin)](#_heading=h.147n2zr) 24

[Authorize user account](#_heading=h.l5y8564y3jj) 24

[Browse user account](#_heading=h.gq0p1ylhjuhj) 25

[Search total expense](#_heading=h.jcv3ofv1asl7) 25

[Sequence diagram](#_heading=h.9thwmbxqzpxr) 26

[Sequence diagram User – Create budget](#_heading=h.kpu0e6yaosb9) 26

[Sequence diagram User – Create transaction](#_heading=h.lcfgdhfmurbu) 27

[Sequence diagram User – Update transaction](#_heading=h.64qra3ixy0wg) 28

[Sequence diagram User – Delete transaction](#_heading=h.k1jyi2j93zo6) 30

[Sequence diagram User – Create Wallet](#_heading=h.acgf7y1enr4e) 30

[Create Wallet without banking](#_heading=h.pvbxz6d1o2dc) 31

[Sequence diagram User – Update Information](#_heading=h.zbvxlsrvrhmg) 32

[Sequence diagram User – Brower expence](#_heading=h.i0itplgxkliv) 33

[Sequence diagram Admin](#_heading=h.q05lwe6ckn54) 34

[Sequence diagram Authorize user account](#_heading=h.29xup6g02wu7) 34

[Sequence diagram Browse user account](#_heading=h.2u0mqxrga71d) 34

[Sequence diagram Search total expense](#_heading=h.pbadyxdi2lxt) 35

[Class diagram](#_heading=h.cut0w0w7y8us) 35

[Yêu cầu khác](#_heading=h.k7pt4duwmofg) 36

[Chức năng](#_heading=h.23ckvvd) 36

[Tính dễ sử dụng](#_heading=h.ihv636) 36

[Các yêu cầu khác](#_heading=h.32hioqz) 36

# Giới thiệu đề tài

## Mục đích

Vấn đề tiền bạc hiện nay không chỉ là chuyện về những con số mà đang dần hình thành như một lối sống. Việc xây dựng một website giúp quản lý tài chính cá nhân sẽ có ích cho tất cả những bạn nào đang bắt đầu tập làm người lớn, hoặc cả những người đã lớn rồi mà vẫn hoang mang về chuyện tiền nong. Giúp mọi người dễ dàng quản lý tài chính của mình và đưa ra các quyết định về vấn đề tài chính một cách thông minh.

Quản lý tài chính cá nhân là một vấn đề khó, mỗi người có một cách tiếp cận và giải quyết riêng. Một số người xây dựng ngân sách, những người khác hạn chế dòng tiền tiêu, những người khác cố gắng tăng dòng tiền thu nhập hiện tại,... Có rất nhiều cách để tiết kiệm và kiếm tiền. Smart Money hoạt động dựa trên nguyên tắc rằng nếu bạn biết tiền của mình đang đi đến đâu, bạn có thể thiết lập một cách chi tiêu hợp lý để có thể tiết kiệm dựa trên số tiền kiếm được bằng cách theo dõi các khoản chi tiêu và thu nhập của mình.

## Đối tượng người dùng

Smart Money hướng đến những đối tượng người dùng đang bắt đầu tập rèn luyện cho bản thân những thói quen kế hoạch chi tiêu của mình , hay thậm chí những người có thói quen lập kế hoạch chi tiêu rồi nhưng vẫn muốn theo dõi lượng tiền hiện có của mình. Ngoài ra, những người đang hoang mang về chuyện tiền nong , về những chi tiêu của mình cũng có thể sử dụng Smart Money. Smart Money sẽ giúp các bạn.

Bạn có thể tạo và chỉnh sửa giao dịch ,tài khoản và cấp cho chúng ngân sách, danh mục. Bạn có thể tự động hóa một phần của việc nhập các giao dịch với các giao dịch định kỳ.Bạn có thể theo dõi các khoản nợ phải trả.Bạn có thể lưu trữ, chỉnh sửa và xóa các giao dịch rút tiền, gửi tiền và chuyển khoản. Điều này cho phép bạn quản lý tài chính đầy đủ.Bạn có thể quản lý các loại tài khoản khác nhau.Có thể lên kế hoạch chi tiêu bằng cách tạo, thay đổi và quản lý tiền trong ngân sách. Ngoài ra, bạn có thể xem báo cáo thu nhập và chi phí đã thanh toán cùng các biểu đồ thống kê.

## Khả năng phát triển

## Từ điển thuật ngữ

CRUD : Tạo, thêm, sửa, xóa

User : Người dùng

Admin : Quản trị viên

SM : Smart Money

Visitor : Khách

## Tài liệu tham khảo

SRS-UGMS-Sample-VN

# Đặc tả usecase

## Đặc tả chức năng

Khi chưa đăng nhập, khách có thể đăng ký tài khoản mới, đăng nhập, yêu cầu thiết lập lại mật khẩu khi quên mật khẩu, và thiết lập lại mật khẩu khi nhận được chỉ dẫn thiết lập qua email.

Khi khách đăng nhập thành công, hệ thống tạo ra menu chứa các chức năng tương ứng với nhóm người dùng mà người dùng đó thuộc về được mô tả chi tiết ở các phần tiếp theo.

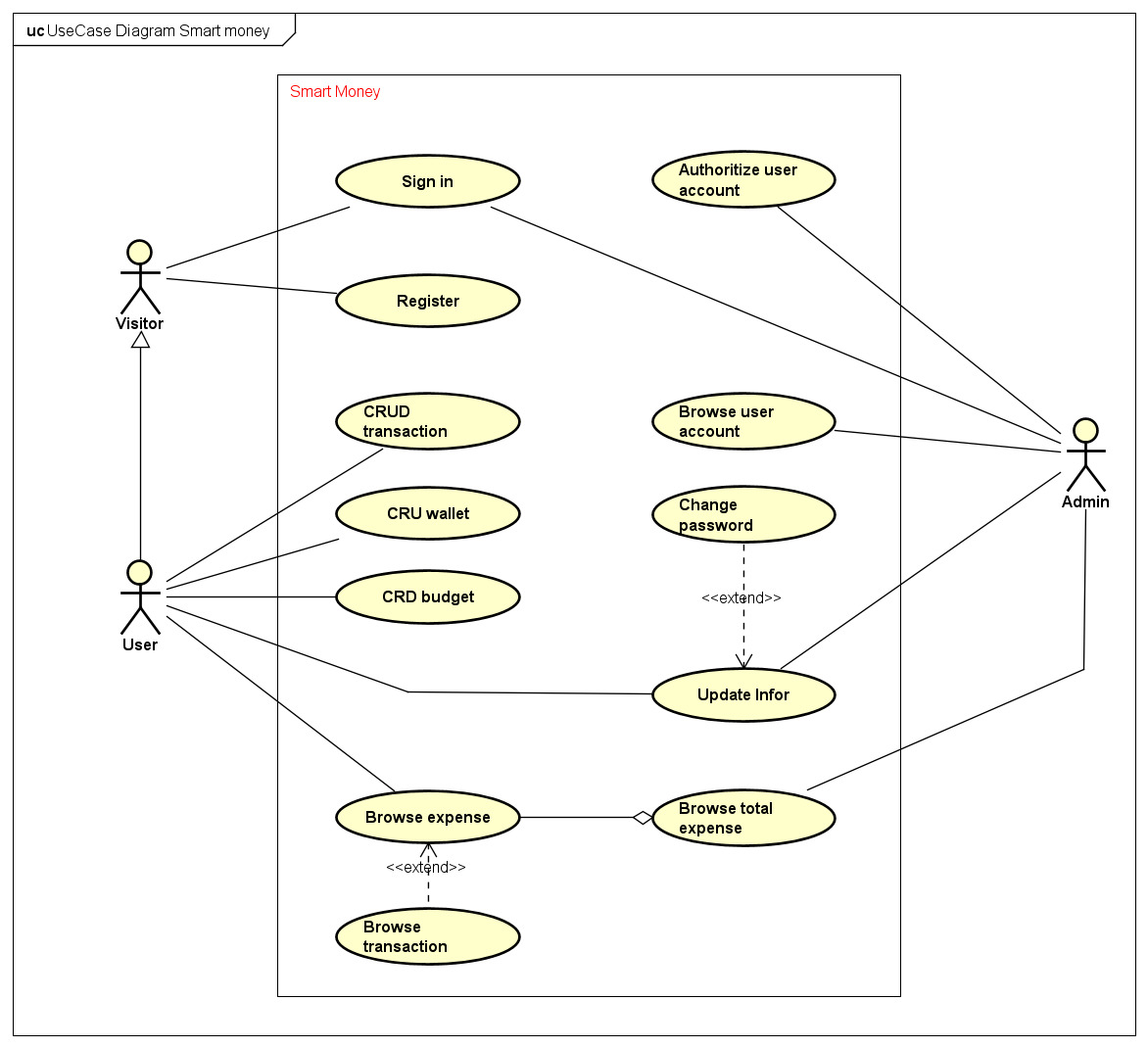
Sau khi đăng nhập, người dùng có thể xem và cập nhật thông tin cá nhân của mình, và có thể thay đổi thay đổi mật khẩu.

Quản trị viên có thể quản lý người dùng và nhóm người, chức năng và loại chức năng có trong hệ thống.

Đồng thời, quản trị viên có thể phân nhóm cho người dùng, và phân quyền sử dụng chức năng cho nhóm người dùng.

Các use case của người dùng trong biểu đồ use case tổng quan này là use case phức hợp của một nhóm các use case. Chi tiết về các use case phức này được đưa ra trong các biểu đồ phân rã ở phần sau.

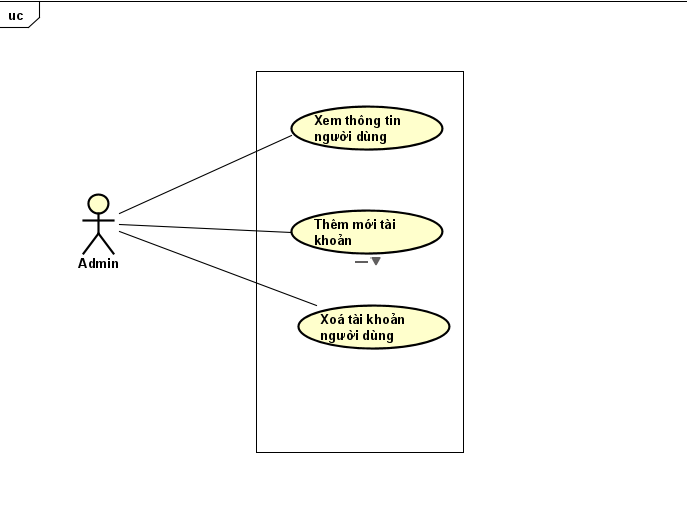
## Biểu đồ Usecase tổng quan



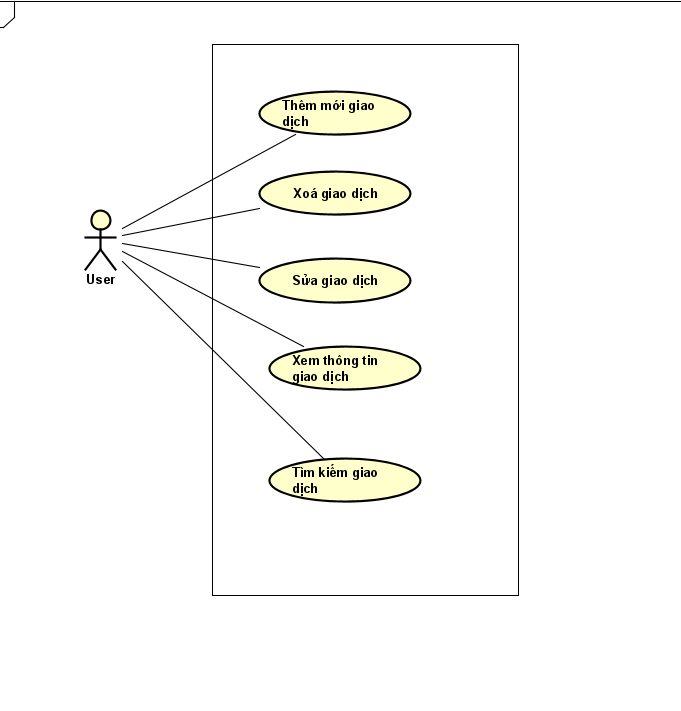


## Biểu đồ Usecase phân rã

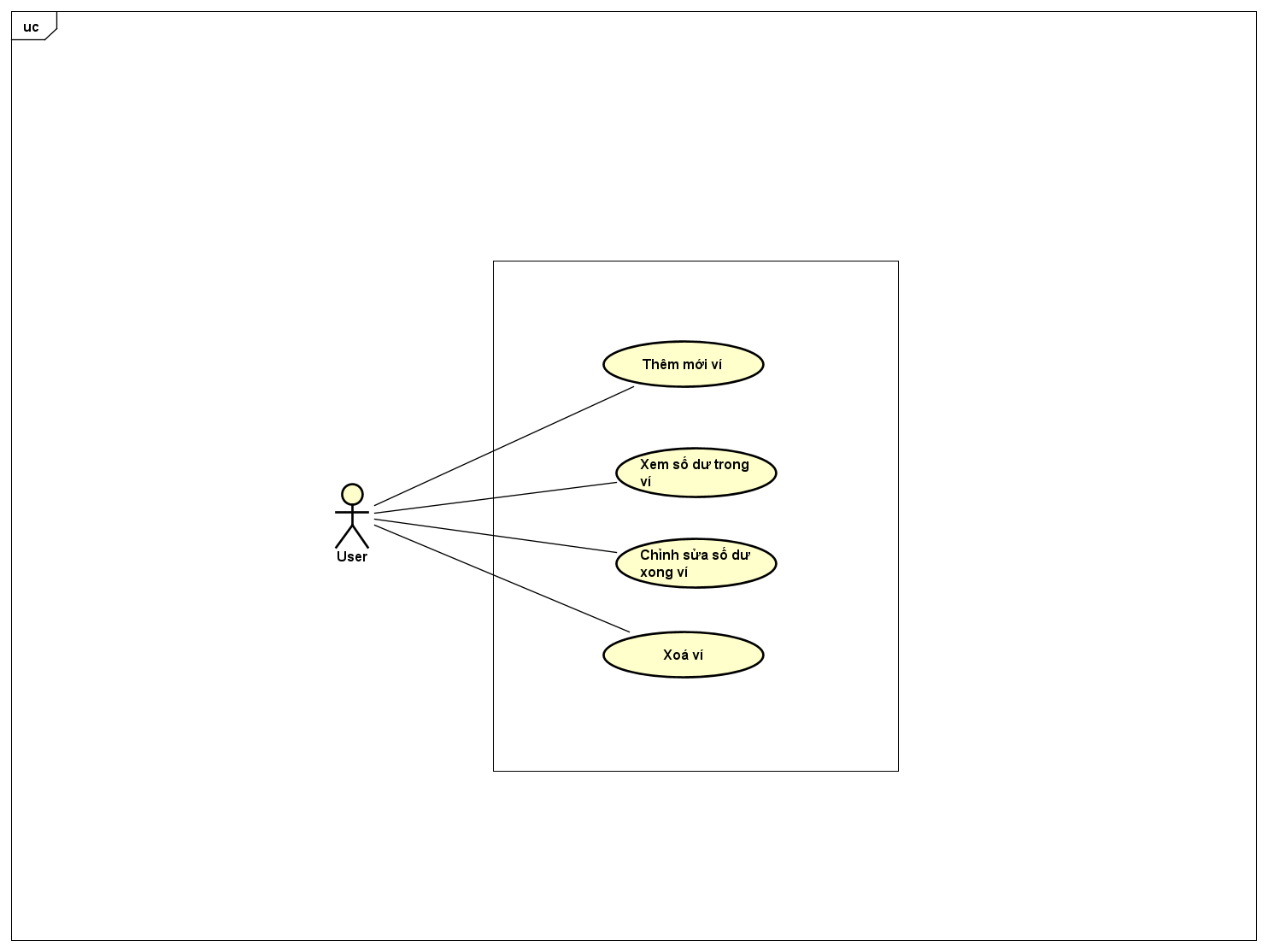
### Phân rã use case “Quản lý người dùng”



### Phân rã use case “CRUD của giao dịch”



### Phân rã use case “CRU của ví”



## Đặc tả use case

### Usecase “Đăng nhập, đăng ký”

Đăng ký : Khi khách hàng chọn đăng ký tài khoản . Hệ thống hiển thị giao diện đăng ký, khách hàng nhập email và mật khẩu .Sau đó hệ thống kiểm tra tài khoản đã tồn tại hay chưa, nếu chưa sẽ cho phép tạo tài khoản và chuyển về trang đăng nhập. Nếu đã tồn tại tài khoản email rồi thì thông báo lỗi cho khách hàng.

Đăng nhập : Khi khách hàng muốn đăng nhập hệ thống , chọn chức năng đăng nhập ->Hệ thống hiện giao diện đăng nhập, khách hàng nhập email và mật khẩu. Hệ thống kiểm tra xem khách hàng đăng ký thông tin chưa. Nếu sai , hệ thống gọi đến use case đăng ký, để khách hàng đăng ký. Nếu đúng, khách đăng nhập vào với tư cách người dùng.

Xóa tài khoản : Nếu đăng nhập là admin, hiển thị giao diện quản lý người dùng dành cho admin. Nếu admin thấy có những tài khoản người dùng lâu không sử dụng, bị bỏ , tài khoản ảo .. -> sẽ xóa tài khoản

### Usecase “Quản lý ví”

Người dùng ấn vào mục ví , hệ thống hiển thị giao diện quản lý ví . Tại giao diện này người dùng có thể tạo ví mới, xóa ví, cập nhật số tiền trong ví.

Nếu người dùng muốn tạo ví, ấn vào mục tạo ví. Hệ thống yêu cầu người dùng nhập vào tổng số tiền thu nhập người đó…..

Nếu người dùng muốn cập nhật số dư trong ví của mình ( muốn thêm thu nhập vào ví ) thì ấn mục thêm thu nhập, hệ thống hiện các ví mà người dùng có, sau đó người dùng chọn ví muốn thêm vào. Hệ thống tự cộng vào số dư trong ví đó.

### Usecase “Quản lý giao dịch”

Thêm giao dịch : Khi người dùng ấn vào nút thêm giao dịch, hệ thống hiển thị giao diện:

Thêm khoản chi tiêu : Nếu người dùng chọn chức năng này hiện giao diện bao gồm chọn ví, chọn loại hàng chi tiêu, số tiền chi tiêu bao nhiêu, ngày tháng nào. Khi người dùng điền thông tin đó và chọn lưu, hệ thống lưu và tự động trừ số dư trong ví. Sau đó hệ thống hiển thị giao diện thông tin giao dịch bao gồm số dư trong ví, tiền chi tiêu. Nếu người dùng muốn xem chi tiết thì có thể gọi đến use case tìm kiếm chi tiêu theo thời gian, hoặc use case thống kê để xem chi tiết.

Thêm thu nhập : Hiển thị giao diện yêu cầu người dùng ghi lại thu nhập ( lương). Sau đó hệ thống tự động cộng khoản thu nhập này vào trong ví.

Thêm khoản vay nợ : Người dùng được phép vay từ các khoản khác và hệ thống tự động trừ trong ví.

Tìm kiếm giao dịch : Hệ thống hiển thị giao diện gồm các chuyên mục, ngày, số lượng . Người dùng có thể ấn vào các thời gian mà mình đã chi tiêu, thu nhập , hệ thống hiển thị các thông tin chi tiêu, thu nhập trong thời gian đó.

Xem thông tin giao dịch : Khi người dùng ấn vào mục thống kê ( xem báo cáo) ->Hệ thống hiển thị giao diện đồ thị gồm số tiền đã chi tiêu hoặc thu thập từng ngày trong tháng , và có thể xem chi tiết những ngày đó đã chi tiêu những gì và thu thập bao nhiêu.

Xóa giao dịch : Những giao dịch mà ở thời gian lâu , người dùng muốn xóa có thể ẩn mục xóa giao dịch hiển thị trên giao diện của các giao dịch, hệ thống sẽ xóa các thông tin liên quan đến giao dịch đó .

### Usecase “Quản lý ngân sách”

Người dùng có thể thực hiện các thao tác với budget như là thêm, xóa budget. Mỗi budget sẽ được gán với 1 category xác định. Khi người dùng thực hiện 1 giao dịch ( transaction) vào một danh mục ( category) tương ứng với category của budget, số tiền sẽ được cộng/ trừ vào số dư của budget. Budget sẽ giúp người dùng quản lý chi tiêu chi tiết từng danh mục và các khoản tiền sử dụng cho nó.

### Usecase “Xem thống kê chi tiêu”

Tại giao diện chính của trang web , có hiển thị các thông tin thống kê chi tiêu của người dùng . Người cùng có thể xem số dư hiện có, các khoản chi tiêu thông qua các biểu đồ thống kê. Ngoài ra, người dùng muốn xem chi tiết khoản chi tiêu ấn vào xem chi tiết, hệ thống hiển thị chi tiết khoản chi tiêu đó.

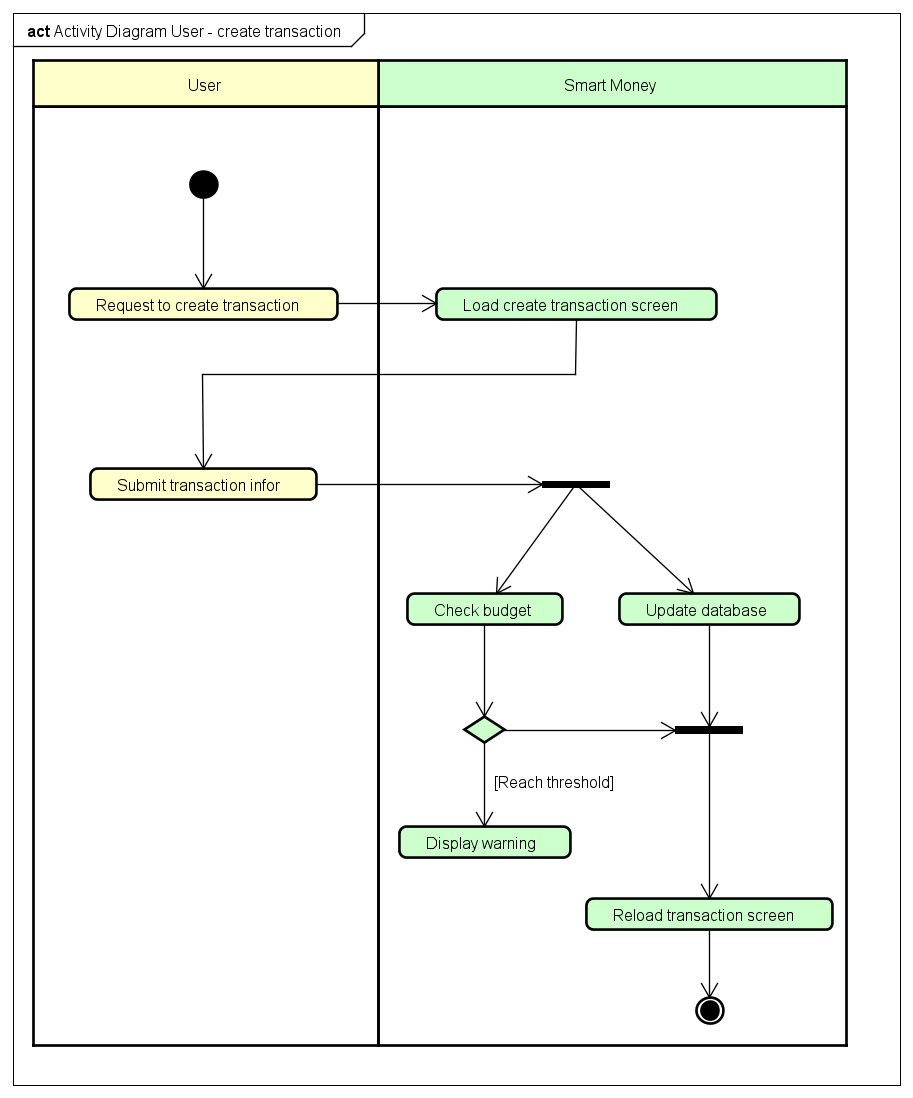
### Usecase “Quản lý người dùng”

Chức năng này dành cho admin. Khi đăng nhập thành công với vai trò admin.Tại giao diện dành cho admin, chọn xem thông tin người dùng. Hệ thống sẽ lấy danh sách các người dùng và hiển thị giao diện có các thông tin người dùng. Trên giao diện này, admin sẽ được phép xóa những tài khoản lâu không sử dụng ( 1-2 năm nếu tài khoản không dùng chưa đến 1 năm không được phép xóa... ) , tuy nhiên admin không có quyền chỉnh sửa thông tin người dùng ( nếu chỉnh sửa hệ thống báo lỗi).

# Thiết kế hệ thống

## Activity diagram

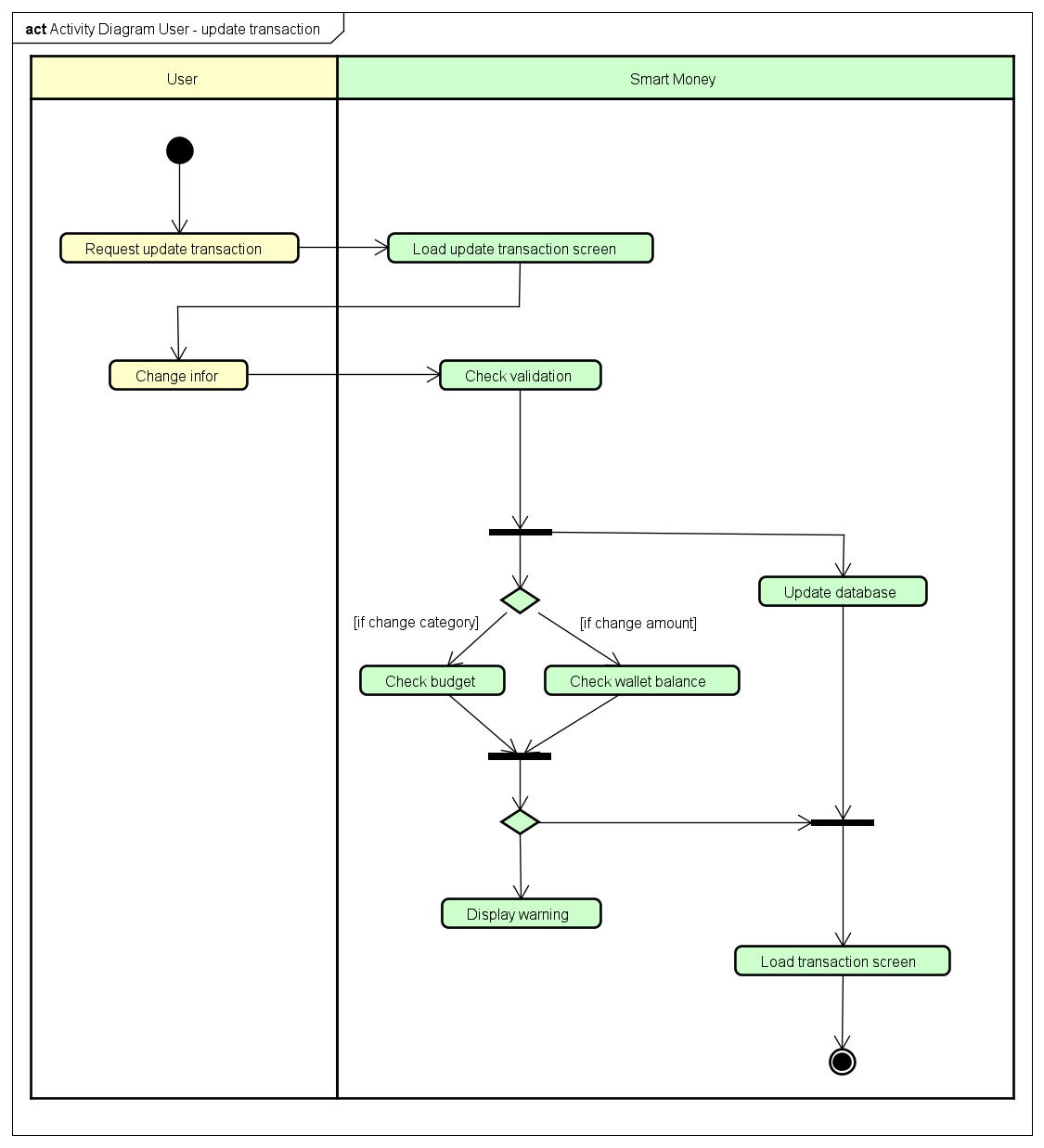
### Quy trình tạo giao dịch mới



Luồng hoạt động:

1. Người dùng nhấn vào button “Create Transaction” gửi yêu cầu tạo một giao dịch mới đến hệ thống.
2. Hệ thống chuyển sang màn hình tạo giao dịch mới gồm các trường thông tin để người dùng nhập.
3. Người dùng thực hiện nhập các thông tin: Tên giao dịch, Danh mục chi tiêu, Ghi chú, Ngày giao dịch, Nhãn, Số tiền, Đơn vị tiền. Trong đó, trường Danh mục chi tiêu là trường select, trường Tên giao dịch, Số tiền, Đơn vị tiền, Ngày giao dịch là các trường bắt buộc điền. Các trường còn lại có thể bỏ trống. Sau đó, nhấn Submit.
4. Với thông tin được nhập từ người dùng, hệ thống sẽ kiểm tra nó với số tiền và tên danh mục của các Ngân sách người dùng đã tạo trước đó. Nếu có tồn tại ngân sách trùng với tên danh mục sẽ tiến hành kiểm tra số dư của ngân sách còn đủ hay không. Nếu đủ, thực hiện lưu vào Database và thông báo tạo giao dịch thành công. Nếu không đủ, hiển thị thông báo với người dùng và trở lại màn hình các giao dịch.

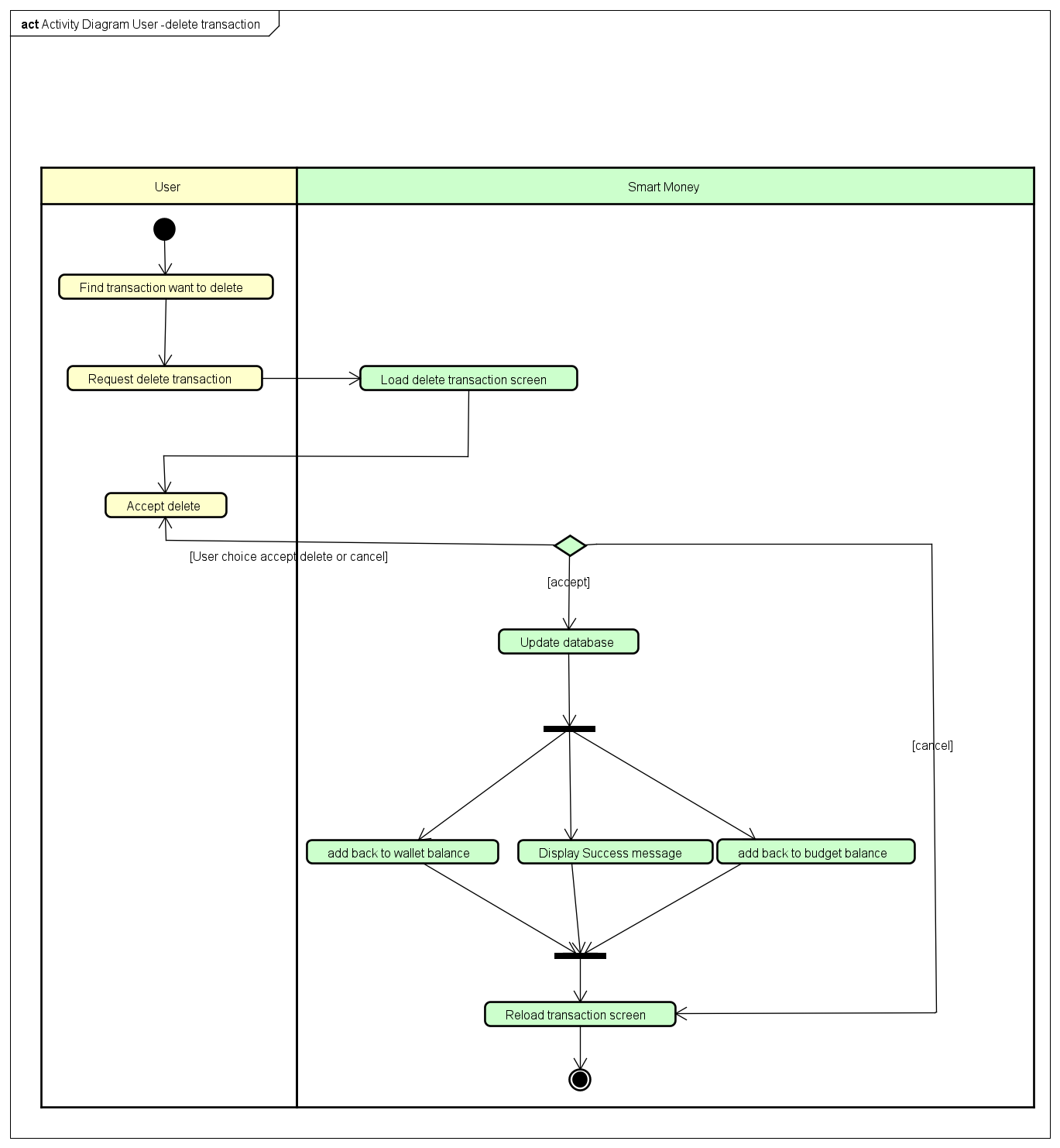
### Quy trình chỉnh sửa một giao dịch



Luồng hoạt động:

1. Người dùng tìm đến giao dịch muốn sửa, nhấn vào button “Update Transaction” gửi yêu cầu chỉnh sửa một giao dịch đến hệ thống.
2. Hệ thống chuyển sang màn hình chỉnh sửa giao dịch gồm các trường thông tin để người dùng nhập.
3. Người dùng thực hiện chỉnh sửa các thông tin: Tên giao dịch, Danh mục chi tiêu, Ghi chú, Ngày giao dịch, Nhãn, Số tiền, Đơn vị tiền. Sau đó, nhấn Submit.
4. Hệ thống kiểm tra thông tin chỉnh sửa của người dùng, nếu người dùng thực hiện chỉnh sửa danh mục có tạo ngân sách, cần thực hiện cập nhật thông tin của ngân sách tương ứng. Kiểm tra số tiền người dùng nhập với số tiền hiện có trong ví tương ứng, nếu phù hợp cập nhật database và thông báo cập nhật thành công. Nếu không thông báo lỗi đến người dùng và quay lại trang các giao dịch.

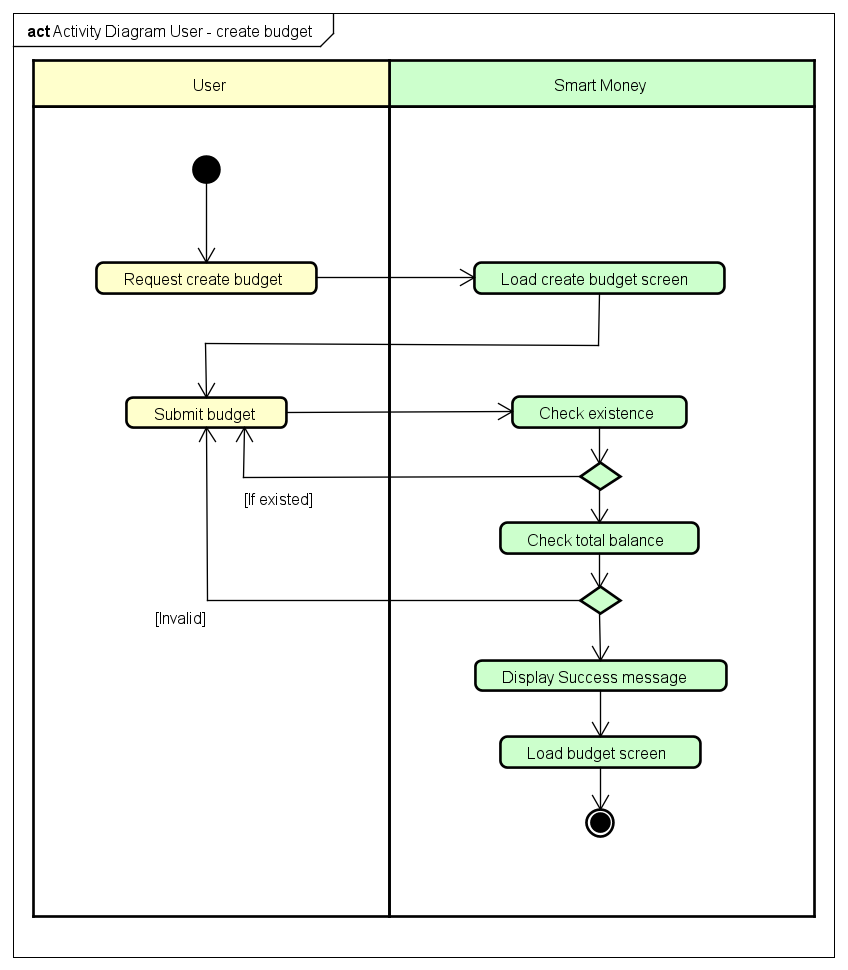
### Quy trình xóa một giao dịch



Luồng hoạt động:

1. Người dùng tìm đến giao dịch muốn xóa, nhấn vào button “Delete Transaction” gửi yêu cầu xóa một giao dịch đến hệ thống.
2. Hệ thống chuyển sang màn hình xác nhận xóa giao dịch gồm các button Accept và Cancel.
3. Nếu Accept, hệ thống xóa database của giao dịch tương ứng, cộng tiền trả lại vào ví và ngân sách tương ứng nếu có. Hiển thị thông báo xóa thành công.
4. Nếu Cancel, hệ thống hiển thị lại trang các giao dịch.

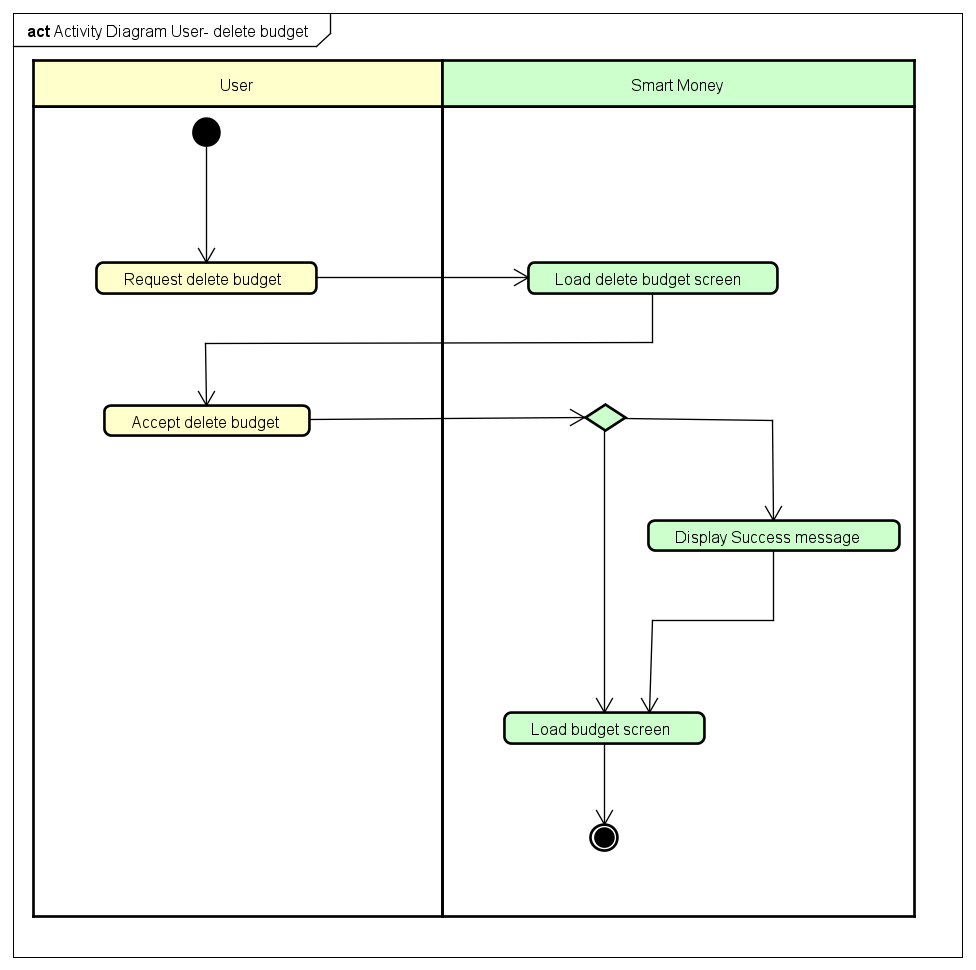
### Quy trình tạo một ngân sách



Luồng hoạt động:

1. Người dùng nhấn vào button “Create Budget” gửi yêu cầu tạo một ngân sách mới đến hệ thống.
2. Hệ thống chuyển sang màn hình tạo ngân sách mới gồm các trường thông tin để người dùng nhập.
3. Người dùng thực hiện nhập các thông tin: Tên ngân sách, Danh mục chi tiêu, Số tiền, Đơn vị tiền. Trong đó, trường Danh mục chi tiêu là trường select, trường Tên giao dịch, Số tiền, Đơn vị tiền,là các trường bắt buộc điền. Sau đó, nhấn Submit.
4. Với thông tin được nhập từ người dùng, hệ thống sẽ kiểm tra nó với tên danh mục của các Ngân sách người dùng đã tạo trước đó. Nếu không tồn tại ngân sách trùng với tên danh mục sẽ tiến hành kiểm tra số dư của ví còn đủ hay không. Nếu đủ, thực hiện lưu vào Database và thông báo tạo ngân sách thành công. Nếu lỗi, hiển thị thông báo với người dùng và trở lại màn hình các giao dịch.

### Quy trình xóa một ngân sách

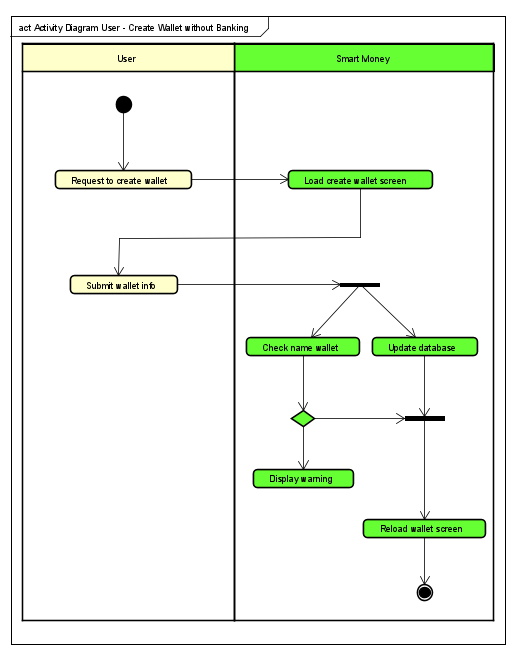


Luồng hoạt động :

1. Người dùng tìm đến ngân sách muốn xóa, nhấn vào button “Delete Budget” gửi yêu cầu xóa một ngân sách đến hệ thống.
2. Hệ thống chuyển sang màn hình xác nhận xóa ngân sách gồm các button Accept và Cancel.
3. Nếu Accept, hệ thống xóa database của ngân sách tương ứng. Hiển thị thông báo xóa thành công.
4. Nếu Cancel, hệ thống hiển thị lại trang ngân sách.

### Quy trình tạo ví

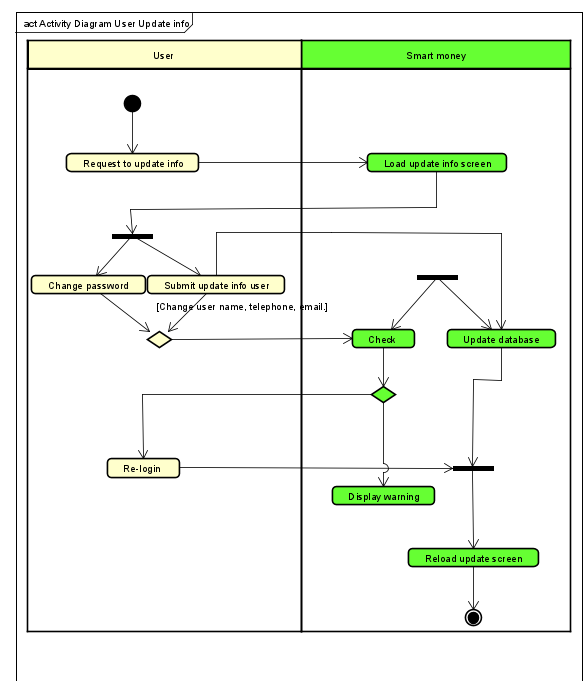
#### Tạo ví tiền mặt không liên kết tài khoản ngân hàng



Luồng hoạt động :

1. Người dùng ấn vào nút “Create Wallet”, chọn tạo “Create cash wallet”
2. Hệ thống sẽ hiển thị giao diện tạo ví mới cho người dùng.
3. Sau đó người dùng nhập các thông tin liên quan đến ví ( tên ví, số dư, đơn vị tiền). Người dùng ấn “submit” , các thông tin được gửi lên hệ thống.
4. Hệ thống sẽ kiểm tra tên ví đã tồn tại hay chưa, nếu tồn tại rồi sẽ báo lỗi cho người dùng, nếu chưa hệ thống lưu dữ liệu. Và gửi về cho người dùng tạo ví thành công.

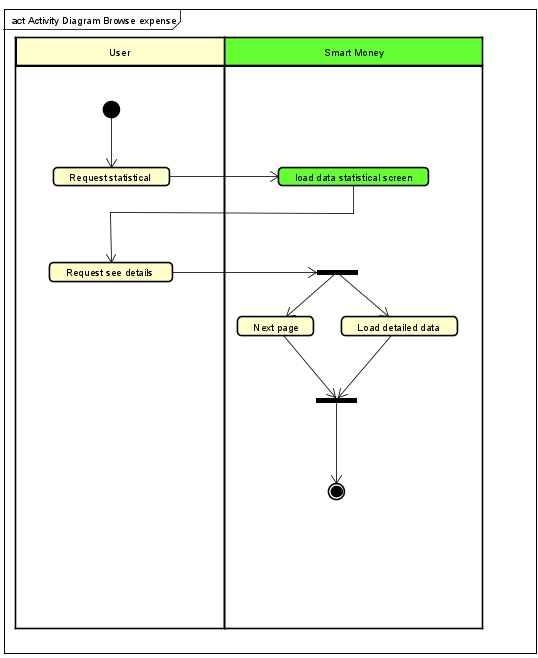
### Quy trình người dùng thay đổi thông tin



Luồng hoạt động :

1. Người dùng ấn vào nút “Setting” chọn “Update information” .
2. Hệ thống hiển thị giao diện thay đổi thông tin.
3. Tại đây người dùng sẽ chọn “Change password” hoặc “Account”. Nếu chọn “Account” , hệ thống hiển thị giao diện các thông tin : Ảnh đại diện, tên, email . Sau khi người dùng thay đổi thông tin ấn “Update”.
4. Hệ thống sẽ lưu các dữ liệu thay đổi.
5. Nếu người dùng chọn “Change password” , hệ thống yêu cầu người dùng nhập mật khẩu cũ , mật khẩu mới.
6. Kiểm tra đúng hệ thống yêu cầu người dùng đăng nhập lại vào hệ thống.

### Quy trình xem thống kê

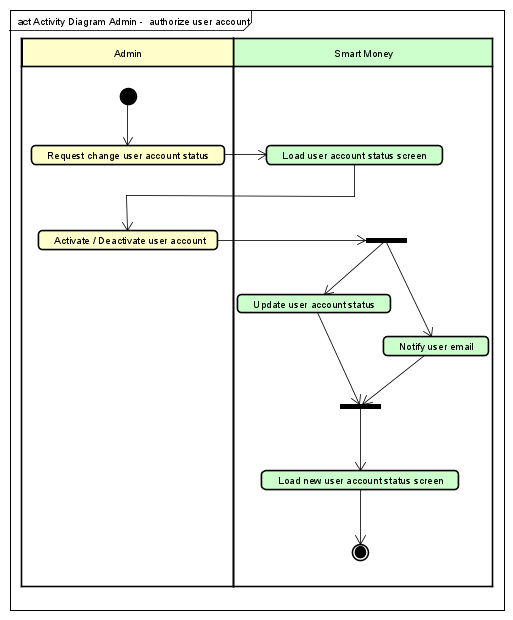


Luồng hoạt động:

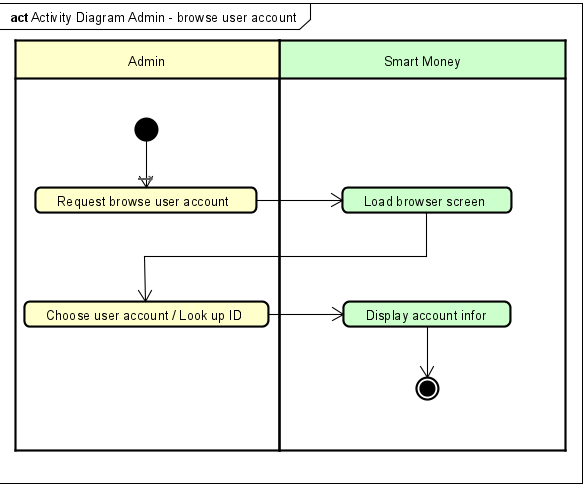
1. Người dùng ấn trang chủ, yêu cầu xem chi tiết thống kê .
2. Hệ thống sẽ load dữ liệu thống kê và chuyển trang mới.
3. Đưa ra màn hình thống kê cho người dùng.

### Quy trình quản lý người dùng (Admin)

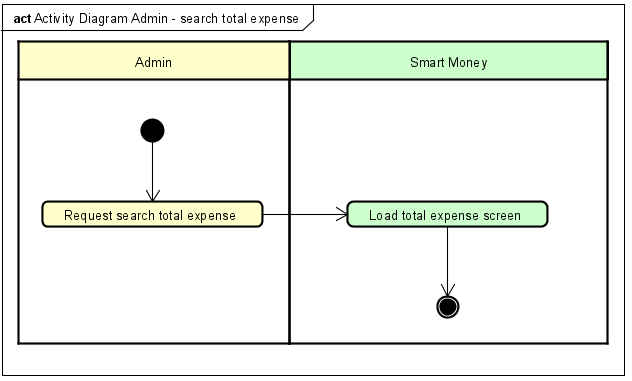
#### Authorize user account



#### Browse user account

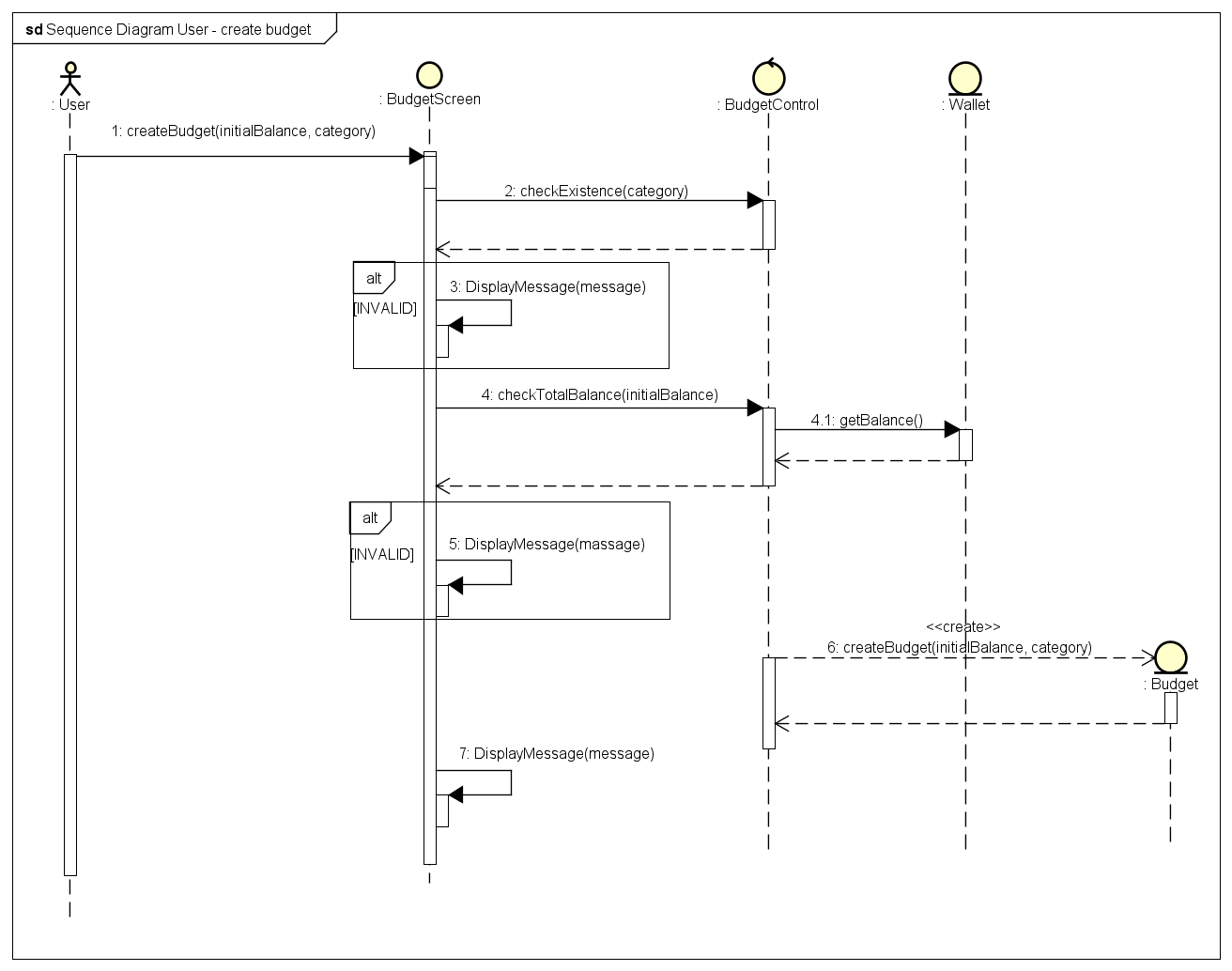


#### Search total expense

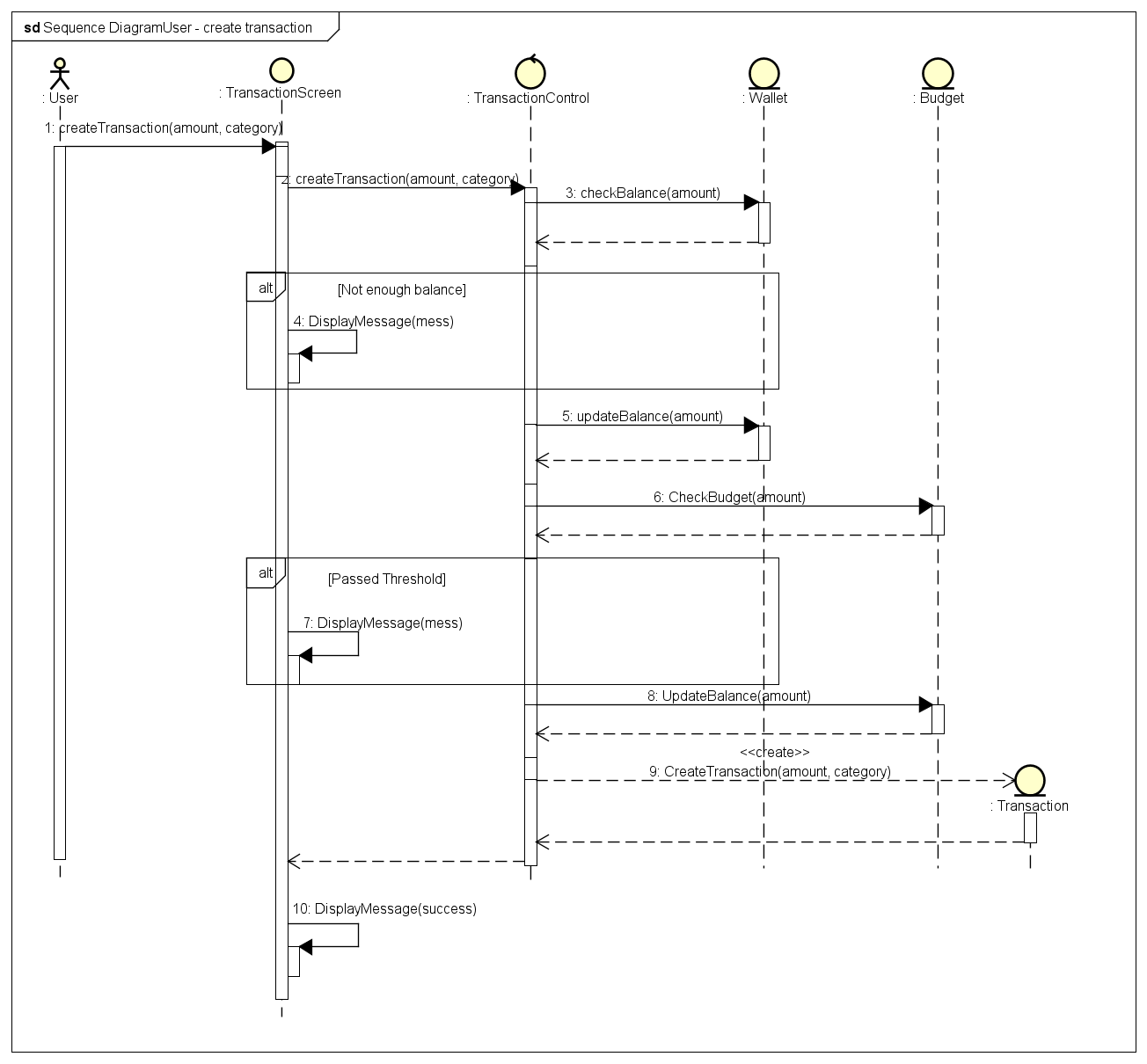


## Sequence diagram

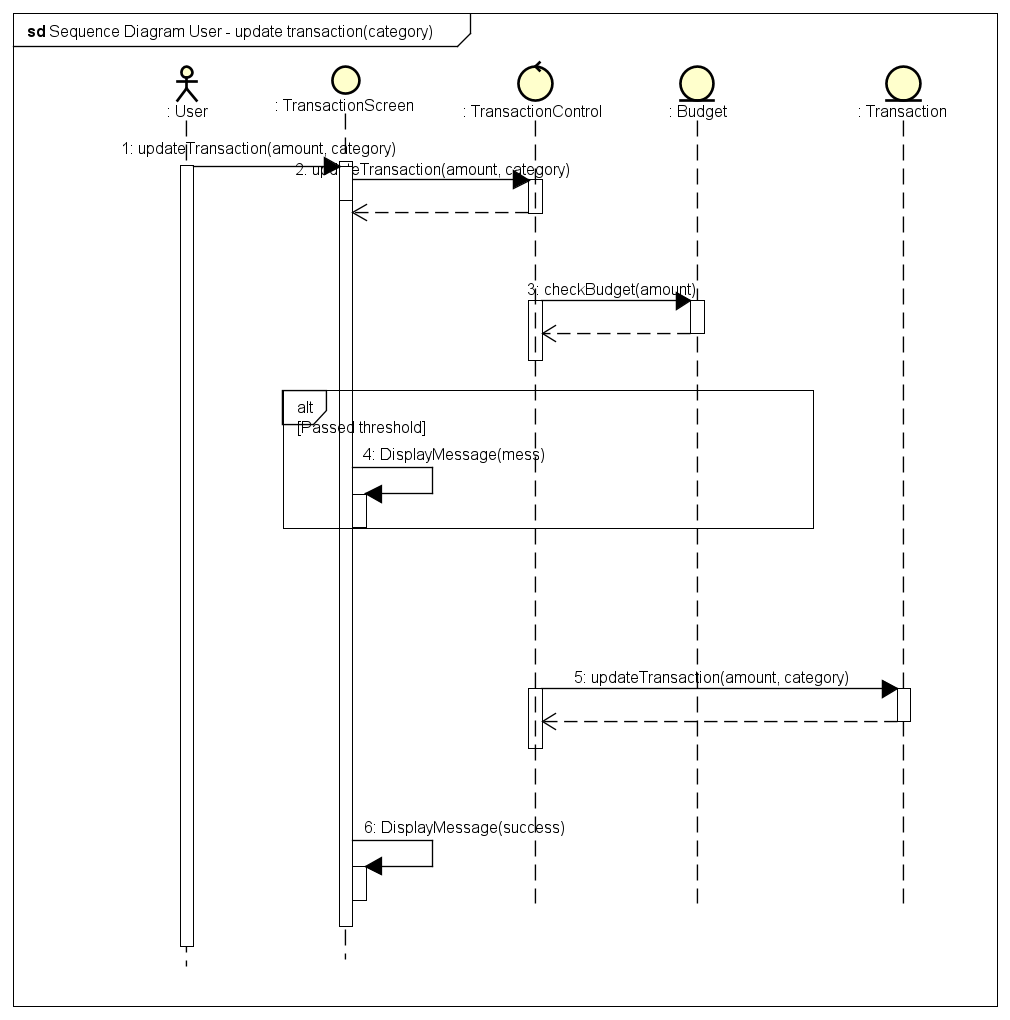
### Sequence diagram User – Create budget

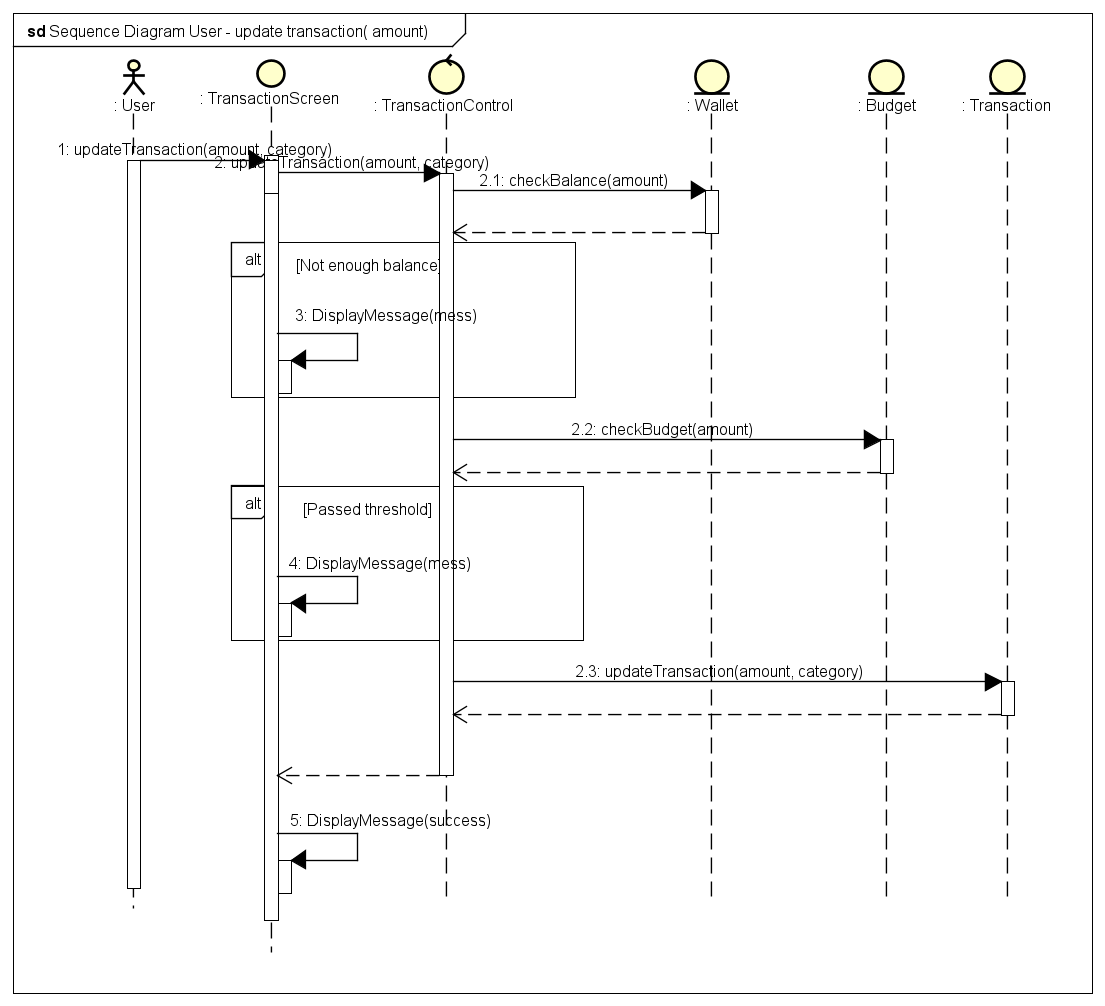


### Sequence diagram User – Create transaction

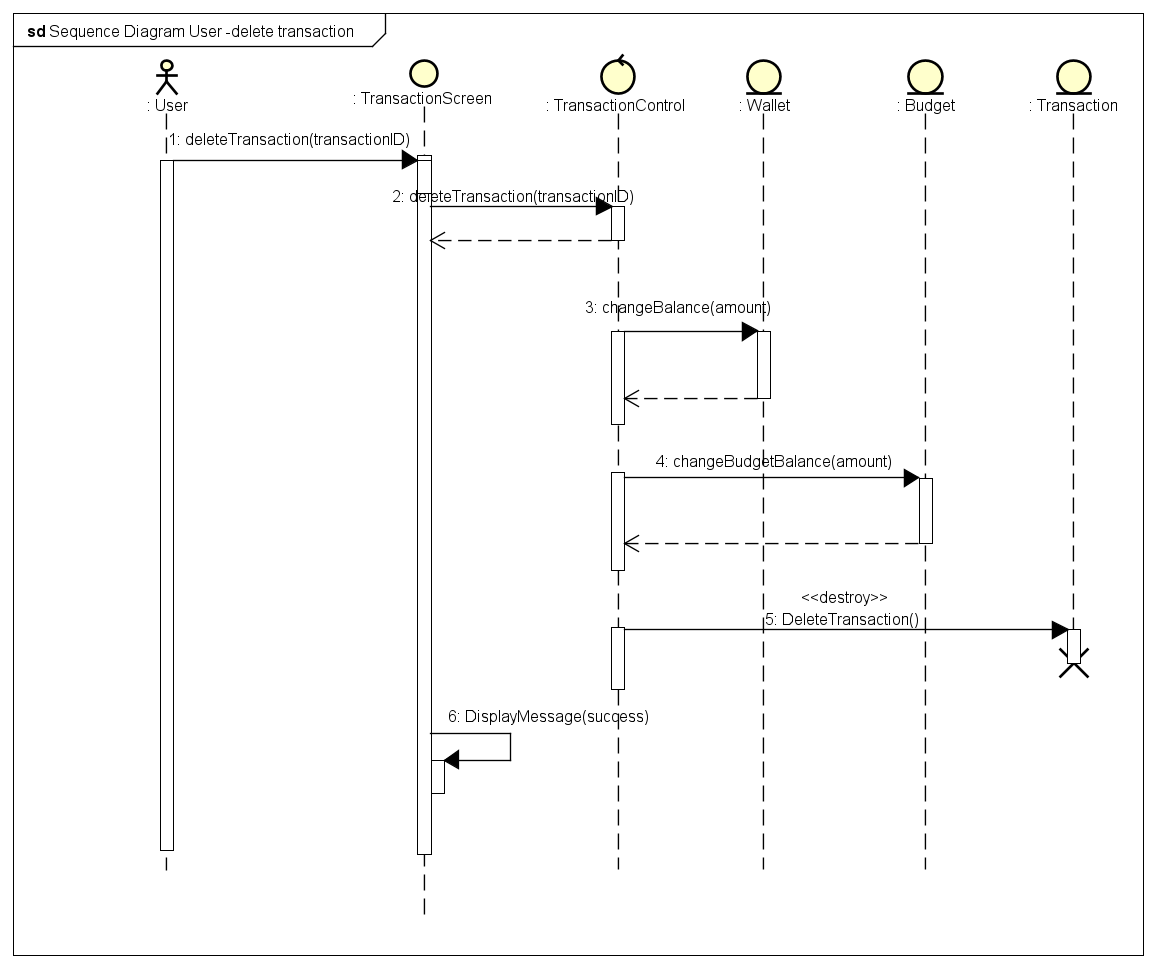


### Sequence diagram User – Update transaction



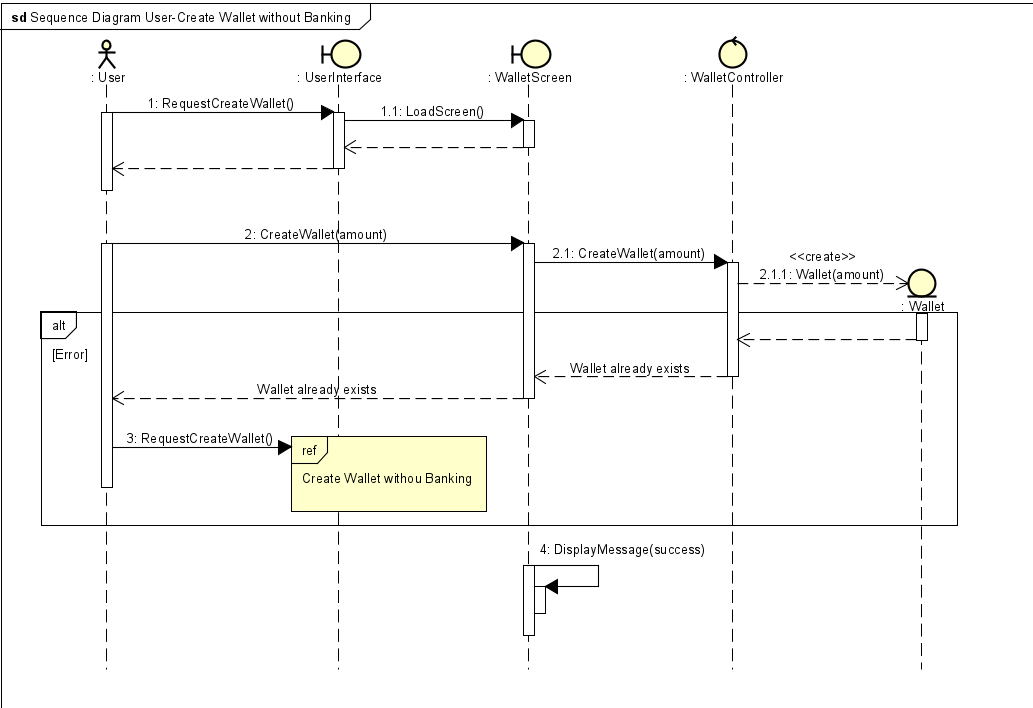


### Sequence diagram User – Delete transaction

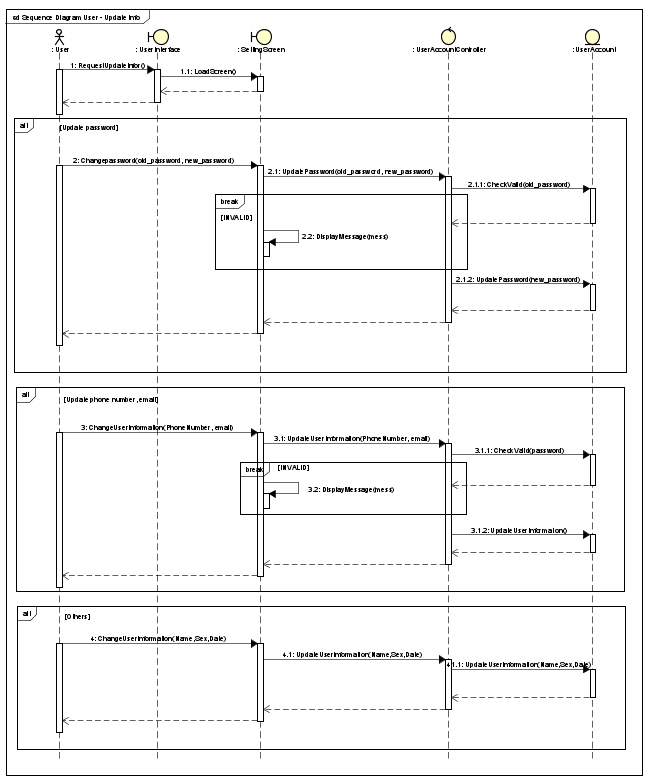


### Sequence diagram User – Create Wallet

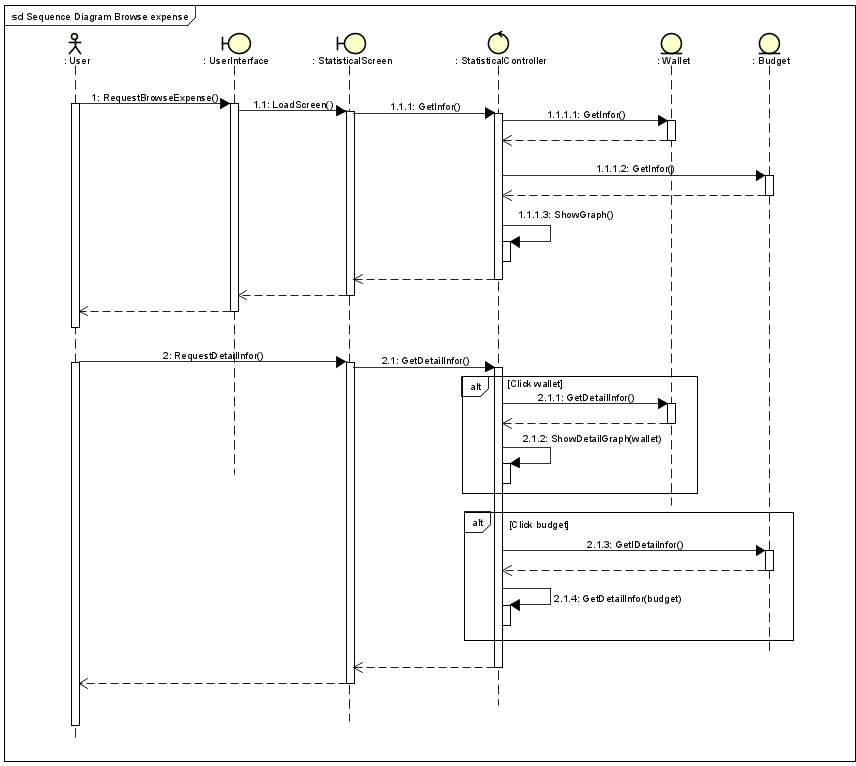
#### Create Wallet without banking



### Sequence diagram User – Update Information

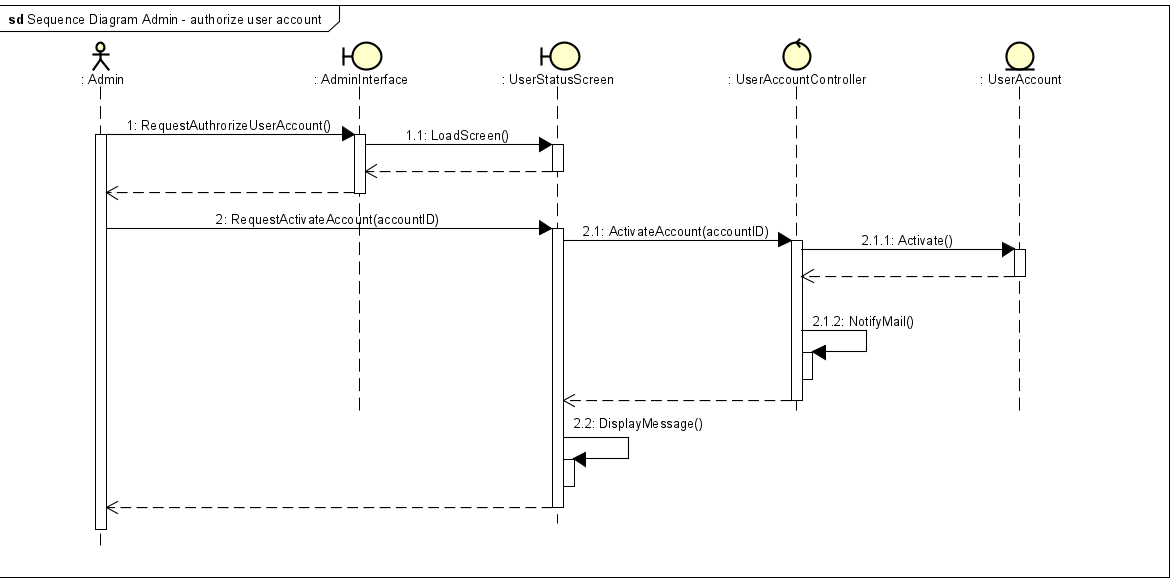


### Sequence diagram User – Brower expence

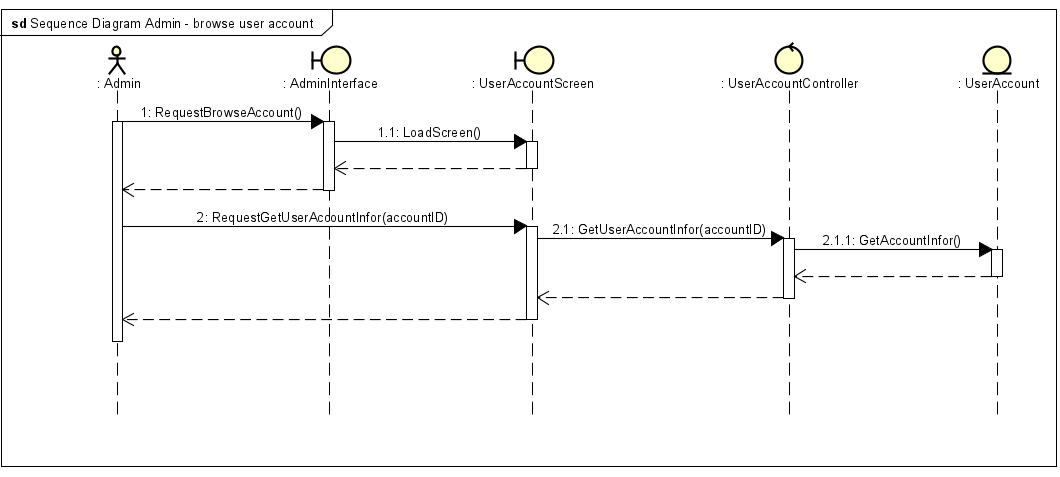


### Sequence diagram Admin

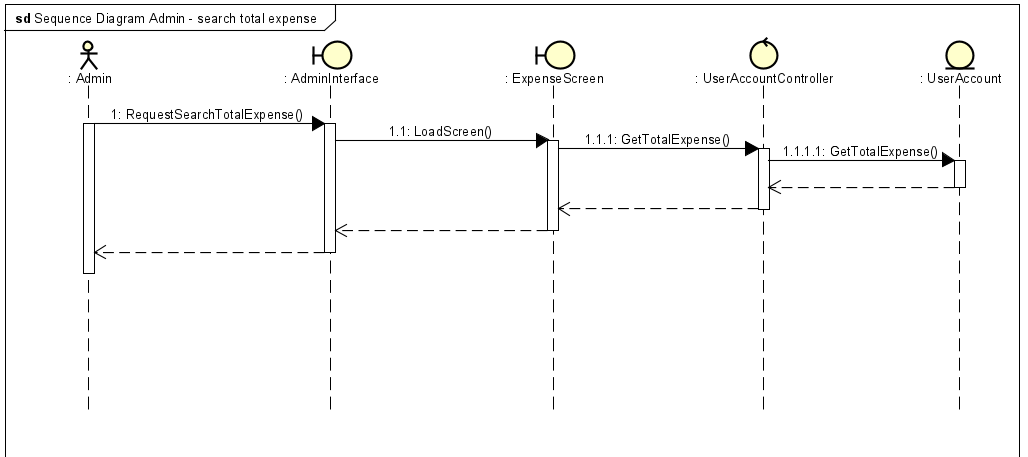
#### Sequence diagram Authorize user account



#### Sequence diagram Browse user account



#### Sequence diagram Search total expense



## Class diagram

## 

## Yêu cầu khác

## Chức năng

* Trong các chuỗi sự kiện của các use case, tất cả các bước có thao tác với CSDL, nếu có lỗi trong quá trình kết nối hoặc thao tác, cần có thông báo lỗi tương ứng để tác nhân biết là lỗi liên quan đến CSDL chứ không liên quan tới lỗi của người dung
* Các use case do Quản trị viên và Người dùng sử dụng thì Khách cần đăng nhập với vai trò tương ứng
* Định dạng hiển thị chung như sau:
* Số căn phải
* Chữ căn trái
* Font: Arial 14, màu đen
* Nền trắng

## Tính dễ sử dụng

Các chức năng được thiết kế sao cho dễ thao tác. Có hướng dẫn cụ thể lỗi sai của người dùng để người dùng biết định vị lỗi, biết lỗi gì và biết cách sửa lỗi.

## Các yêu cầu khác

* Thiết kế trang web có tính tin cậy : đảm bảo được những thông tin người dùng không bị lộ.
* Về giao diện thiết kế sao cho người dùng dễ dàng sử dụng, hình ảnh bắt mắt….
* Thiết kế trang web để dễ dàng chỉnh sửa.