

**APAMAN**

**Software Design Specification**

– Hanoi, February 2023 –

**Record of changeS**

| **Date** | **A\* M, D** | **In charge** | **Change Description** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

\*A - Added M - Modified D - Deleted

**Table of Contents**

[I. Overview 4](#_heading=h.gjdgxs)

[1. Code Packages 4](#_heading=h.30j0zll)

[2. Database Design 4](#_heading=h.1fob9te)

[a. Database Schema 4](#_heading=h.3znysh7)

[b. Table Description 4](#_heading=h.2et92p0)

[II. Code Designs 5](#_heading=h.tyjcwt)

[1. <Feature/Function Name1> 5](#_heading=h.3dy6vkm)

[a. Class Diagram 5](#_heading=h.1t3h5sf)

[b. Class Specifications 5](#_heading=h.4d34og8)

[c. Sequence Diagram(s) 5](#_heading=h.2s8eyo1)

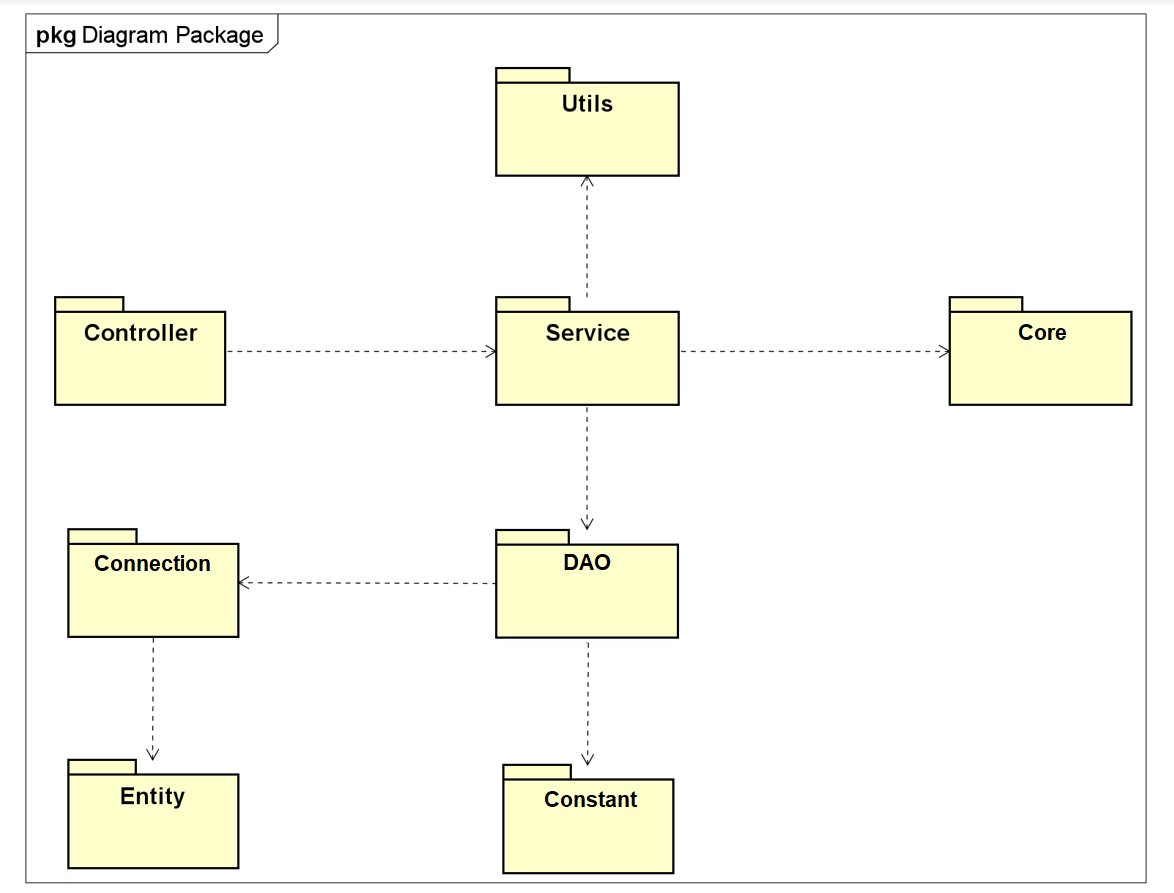
[d. Database queries 6](#_heading=h.17dp8vu)

[2. <Feature/Function Name2> 6](#_heading=h.3rdcrjn)

# I. Overview

## 1. Code Packages

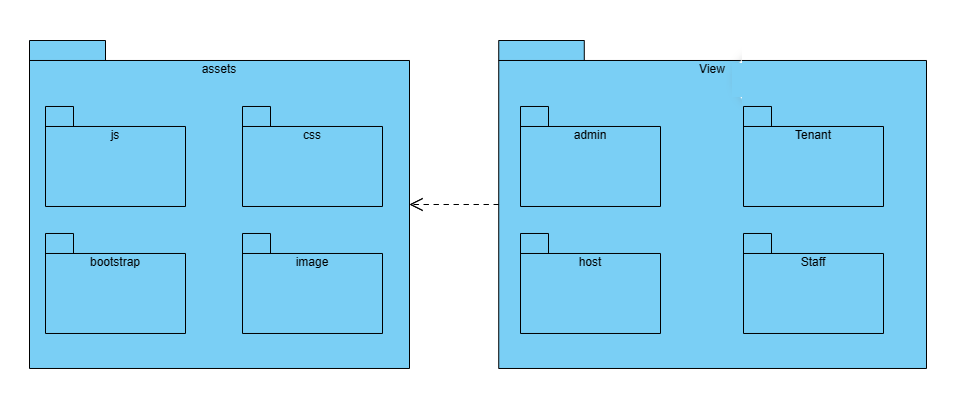
### 1.1 Back-end Package Diagram



***Package descriptions***

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | DAO | Contains interfaces which communicate with the database. |
| 02 | controller | Contains classes which handle incoming requests. |
| 03 | service | Contains classes which are used to perform business logic. These classes communicate with data tier, process data then return to controller classes. |
| 04 | entity | Contains classes represent real entities which can be persisted in the database and some additional helper classes. |
| 05 | utils | Contains validation, formatter classes. |
| 06 | core | Contains filter layer classes. |
| 07 | constant | Contain constant value |
| 08 | connection | Contain classes which help connecting to database |

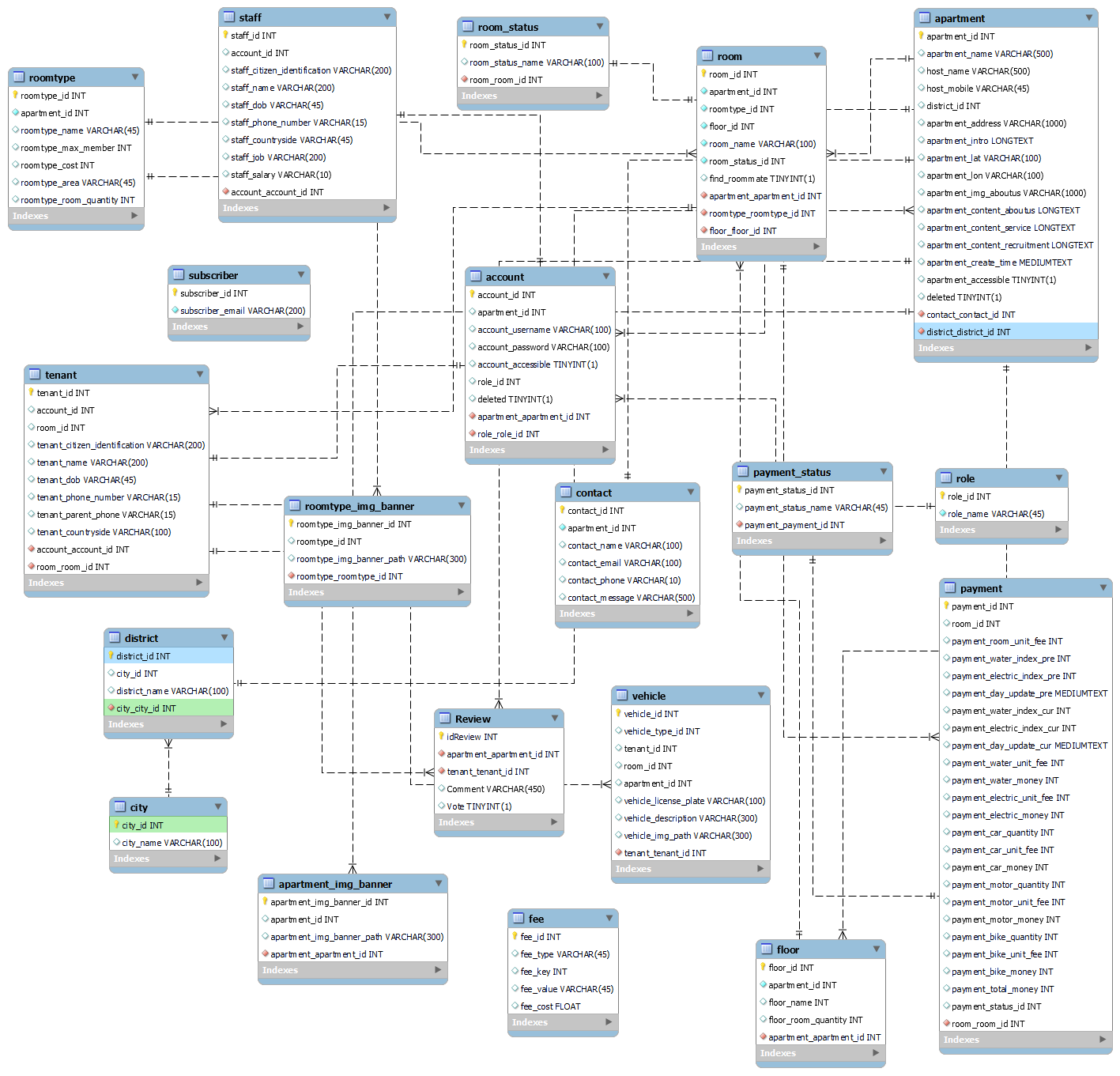
### 1.2 Front-end Package Diagram



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | assets | Package contains all resource file in front-end |
| 02 | js | Package contains all java script files which is not available in bootstrap |
| 03 | css | Package contains all css files which is not available in bootstrap |
| 04 | image | Package contains all images and icons appeared on website |
| 05 | views | Package contains all jsp files which include all-role webpages |
| 06 | admin | Package contains all admin jsp file |
| 07 | tenant | Package contains all tenant jsp file |
| 08 | host | Package contains all host jsp file |
| 09 | staff | Package contains all staff jsp file |

## 2. Database Design

### a. Database Schema

**

### b. Table Description

#### 1.Account

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 01 | Account | account\_id : **Int** (NOT NULL , AUTO\_INCREMENT)  apartment\_id : **Int**  account\_username : **Varchar**  account\_password : **Varchar**  account\_accesible : **tinyint**  role\_id : **int**  deleted t: **tinyint**  PRIMARY\_KEY : **account\_id**  FOREIGN KEY **: apartment\_apartment\_id , apartment\_account\_account\_id , role\_role\_id** |

#### 2.Apartment

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 02 | Apartment | apartment\_id : **Int** (NOT NULL , AUTO\_INCREMENT)  apartment\_name : **Varchar**  host\_name : **Varchar**  host\_mobile : **Varchar**  district\_id : **Int**  apartment\_address : **Varchar**  apartment\_intro : **longtext**  apartment\_latitude : **Varchar**  apartment\_longtitude : **Varchar**  apartment\_img\_aboutus : **Varchar**  apartment\_content\_aboutus : **longtext**  apartment\_content\_service: **longtext**  apartment\_content\_recruitment : **longtext**  apartment\_accessible : **tinyint**  account\_account\_id : **INT**  delete : **TINYINT**  PRIMARY\_KEY **: apartment\_id , account\_account\_id** |

#### 3. Apartment\_img

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 03 | Apartment\_img | apartment\_img\_banner\_id` : **Int** (NOT NULL , AUTO\_INCREMENT  apartment\_id` : **Int**  apartment\_img\_banner\_path` :  **Varchar**  PRIMARY\_KEY **: apartment\_img\_banner\_id**  FOREIGN\_KEY :  **apartment\_apartment\_id** |

#### 

#### ***4. City***

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 04 | City | city\_id : **Int** (NOT NULL , AUTO\_INCREMENT  city\_name :  **Varchar**  PRIMARY\_KEY **: city\_id**  FOREIGN\_KEY :  **apartment\_apartment\_id , apartment\_account\_account\_id** |

#### 5.District

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 05 | City | district\_id : **Int** (NOT NULL , AUTO\_INCREMENT)  city\_id : **Int**  district\_name :  **Varchar**  PRIMARY\_KEY **: district\_id**  FOREIGN\_KEY : **city\_city\_id** |

#### 6. Floor

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 06 | Floor | floor\_id : **Int** (NOT NULL , AUTO\_INCREMENT)  apartment\_id : **Int**  floor\_name : **Varchar**  PRIMARY\_KEY : **floor\_id** |

#### 7. Role

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 07 | Role | role\_id : **Int (**NOT NULL , AUTO\_INCREMENT**)**  role\_name : Varchar  PRIMARY\_KEY : **role\_id** |

#### ***8. RoomType***

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 08 | RoomType | roomtype\_id :  **Int (**NOT NULL , AUTO\_INCREMENT**)**  apartment\_id :  **int**  roomtype\_name :  **varchar(45)**  roomtype\_max\_member : **int**  roomtype\_cost :  **int**  roomtype\_area : **varchar**  PRIMARY KEY : **roomtype\_id** |

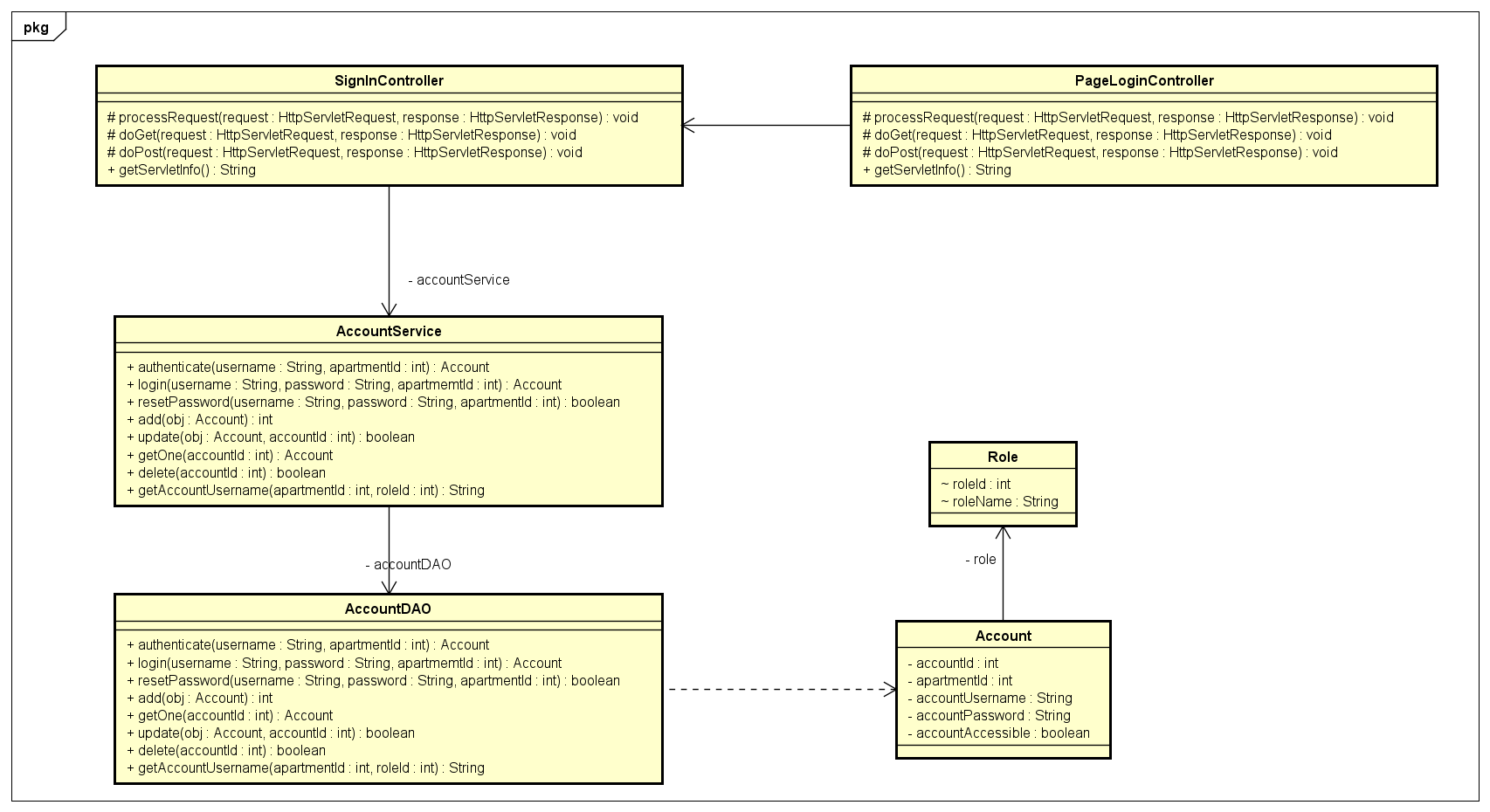
***9. Staff***

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 09 | Staff | staff\_id : **int**  account\_id : **int**  staff\_citizen\_identification : **varchar**  staff\_name : **varchar**  staff\_dob : **varchar**  staff\_phone\_number : **varchar**  staff\_countryside : **varchar**  staff\_job : **varchar**  staff\_salary : **varchar**  PRIMARY KEY : **staff\_id**  FOREIGN\_KEY : **account\_account\_id** |

# II. Code Designs

## 1. Feature 1: Authentication and Authorization

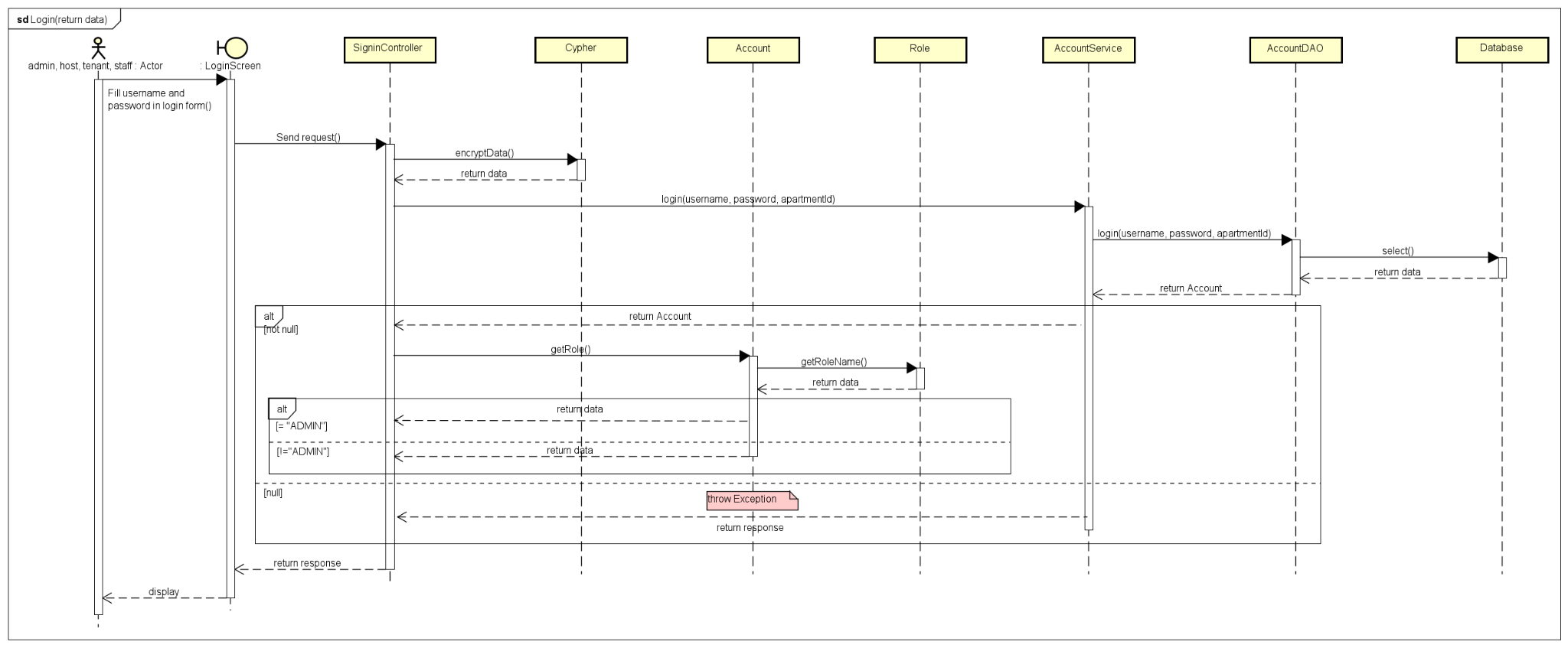
### a. Class Diagram

**

### b. Class Specifications

### c. Sequence Diagram(s)

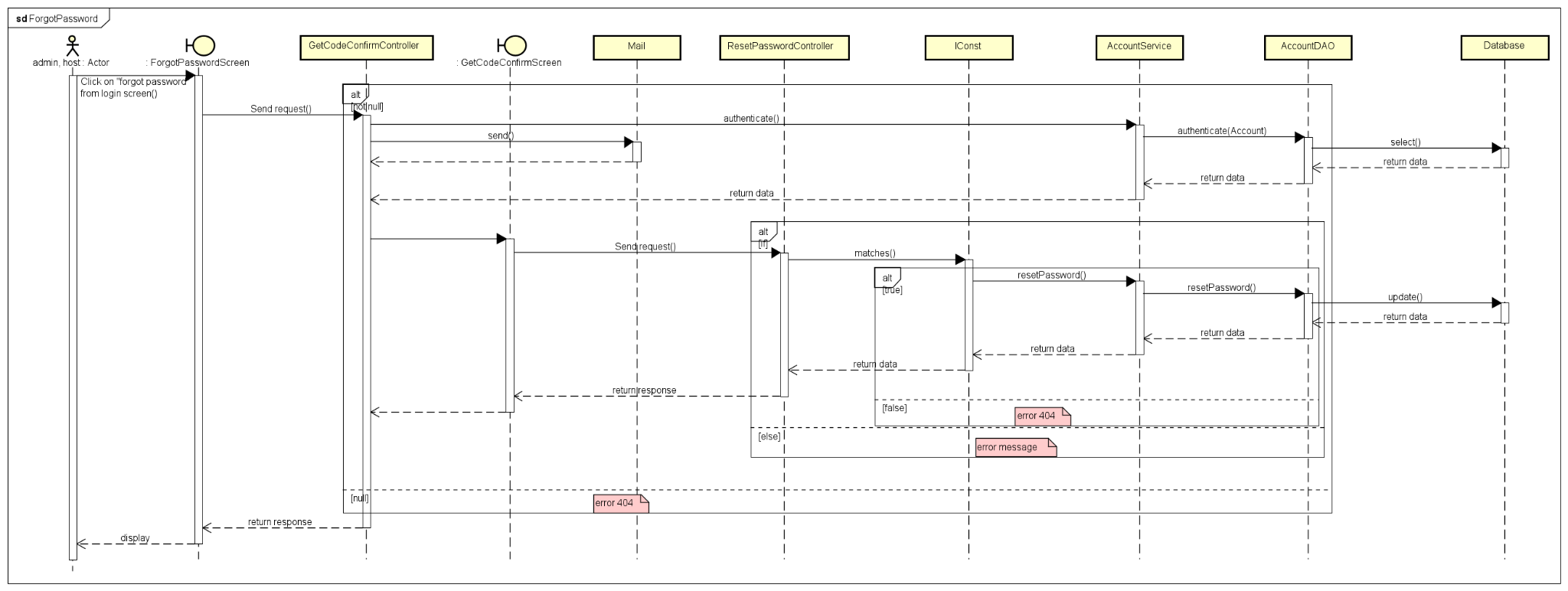
#### Function: Login



#### Function: Logout



#### Function: Forgot and Reset Password



### d. Database Queries

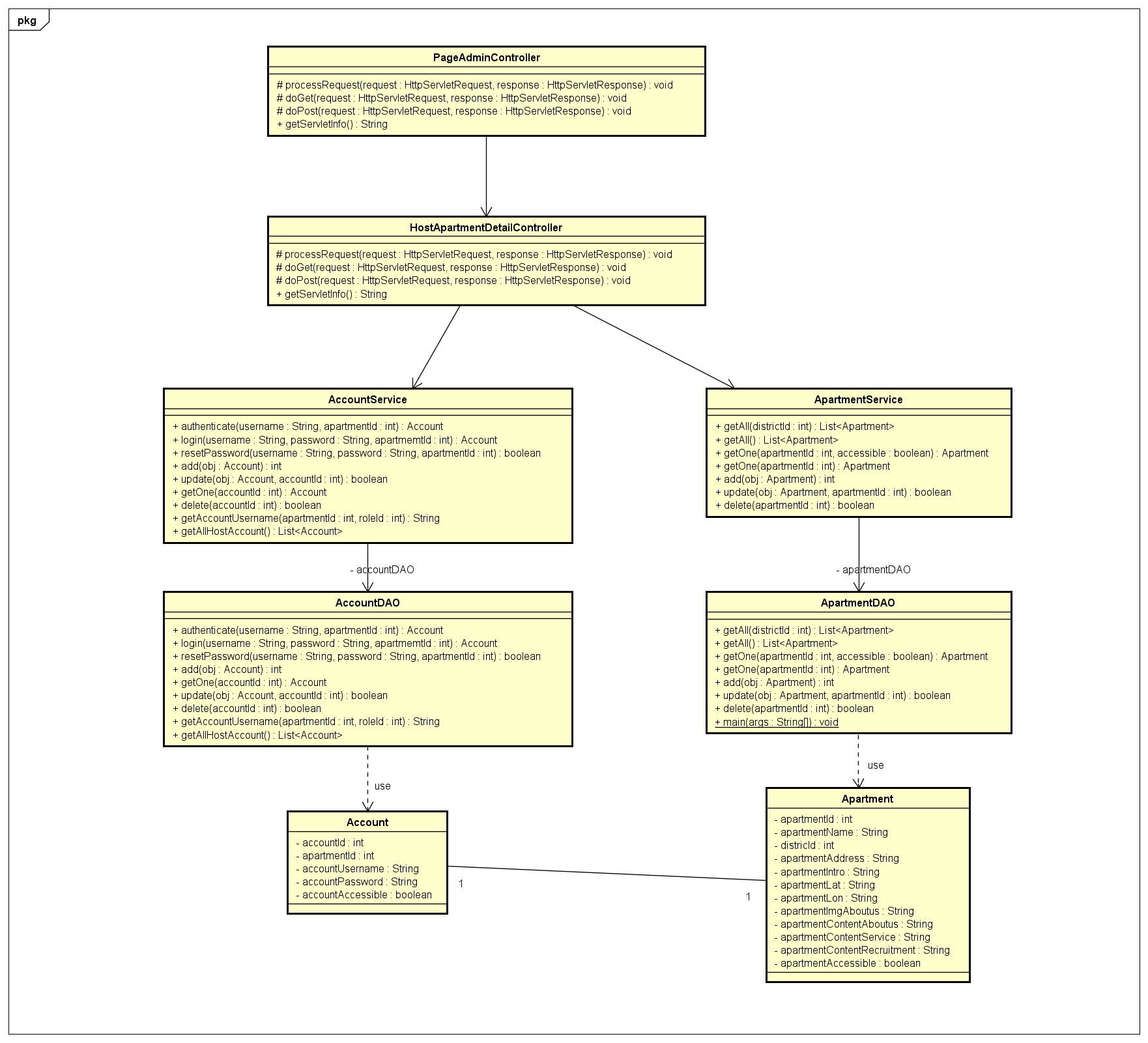
* ***Login function queries:***

*SELECT a.account\_id, a.apartment\_id, a.account\_username, a.account\_password, r.role\_name FROM apamandb.`account` a join `role` r ON a.role\_id = r.role\_id\n"*

*+ "Where a.account\_username = ? AND a.account\_password = ? AND a.account\_accessible = true AND deleted = false AND (a.apartment\_id = 0 OR a.apartment\_id = ?)"*

## 2. Feature 2: Manage Apartment

### a. Class Diagram

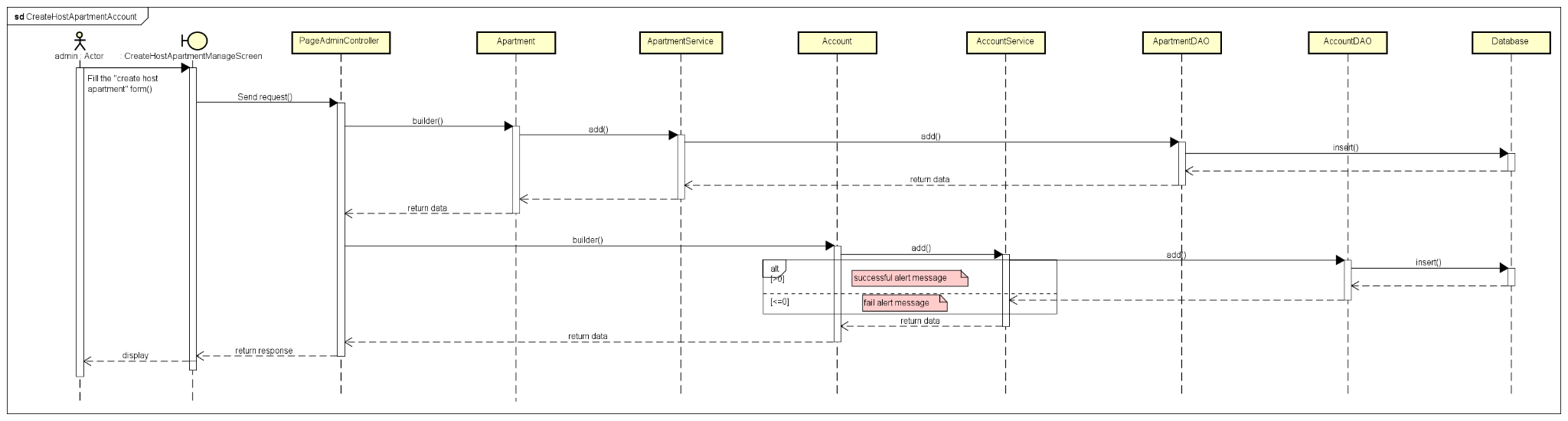


### b. Class Specifications

#### 

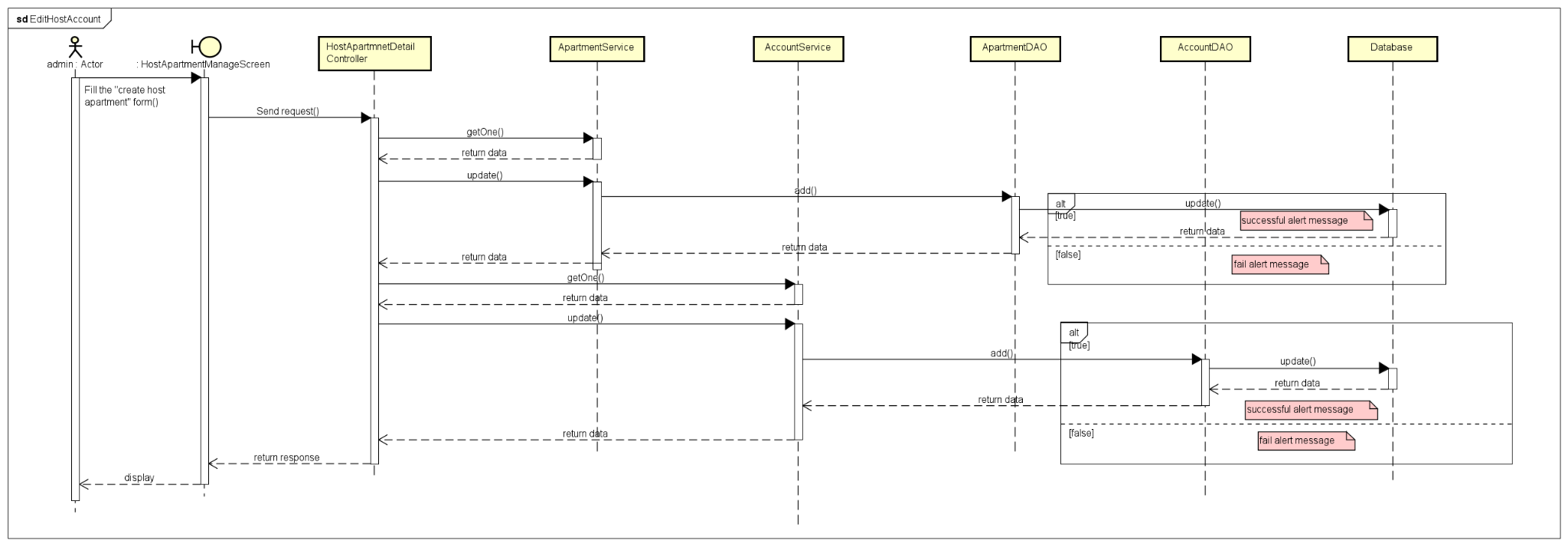
### c. Sequence Diagram(s)

#### Function: Create Host Account



## 

#### Function: Edit Host Account



#### Function: Delete Host Account

## 

### d. Database Queries

* ***Create host account function queries:***

*"INSERT INTO `account`(apartment\_id, account\_username, account\_password, account\_accessible, role\_id, deleted)" + " VALUES(?, ?, ?, ?, ?, ?)"*

* ***Edit host account function queries:***

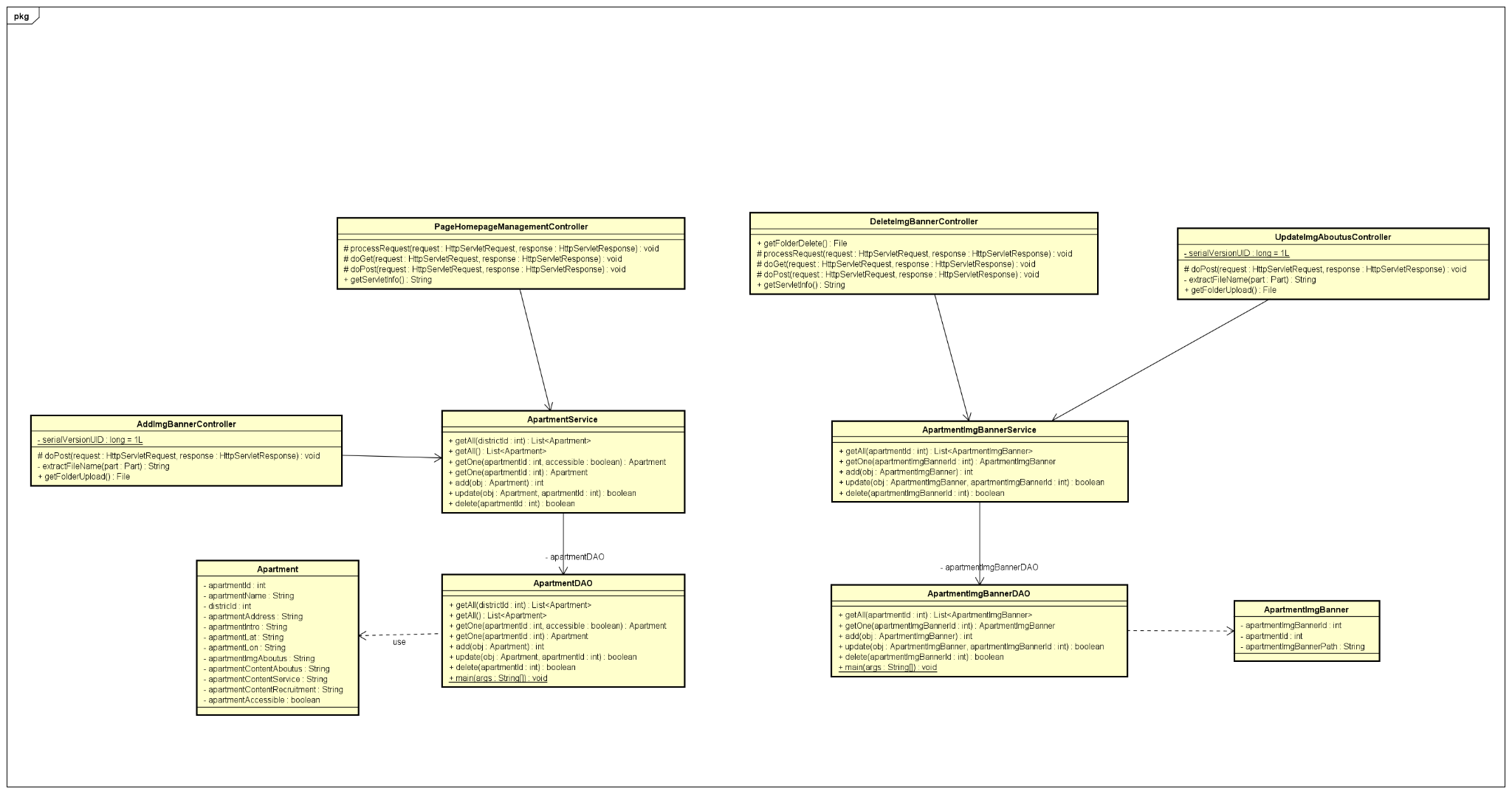
*"UPDATE apamandb.`account` SET apartment\_id = ?, account\_username = ?, account\_password = ?, account\_accessible = ?, role\_id = ? WHERE account\_id = ?"*

* ***Delete host account function queries:***

*UPDATE `account` SET deleted = 1 WHERE account\_id = ?*

## 3. Feature 3: Manage HomePage

### a. Class Diagram



### b. Class Specifications

#### 

***Class Methods***

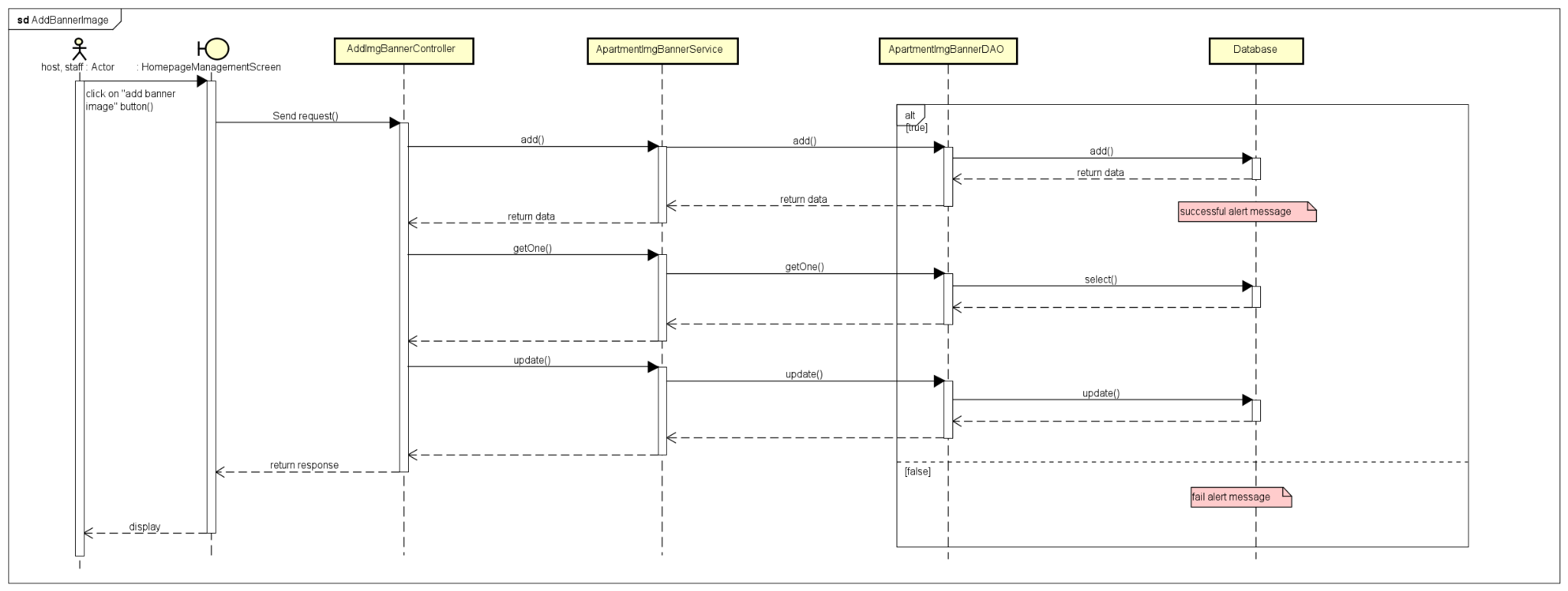
| **No** | **Method** | **Description** |
| --- | --- | --- |
|  |  |  |
|  |  |  |

### c. Sequence Diagram(s)

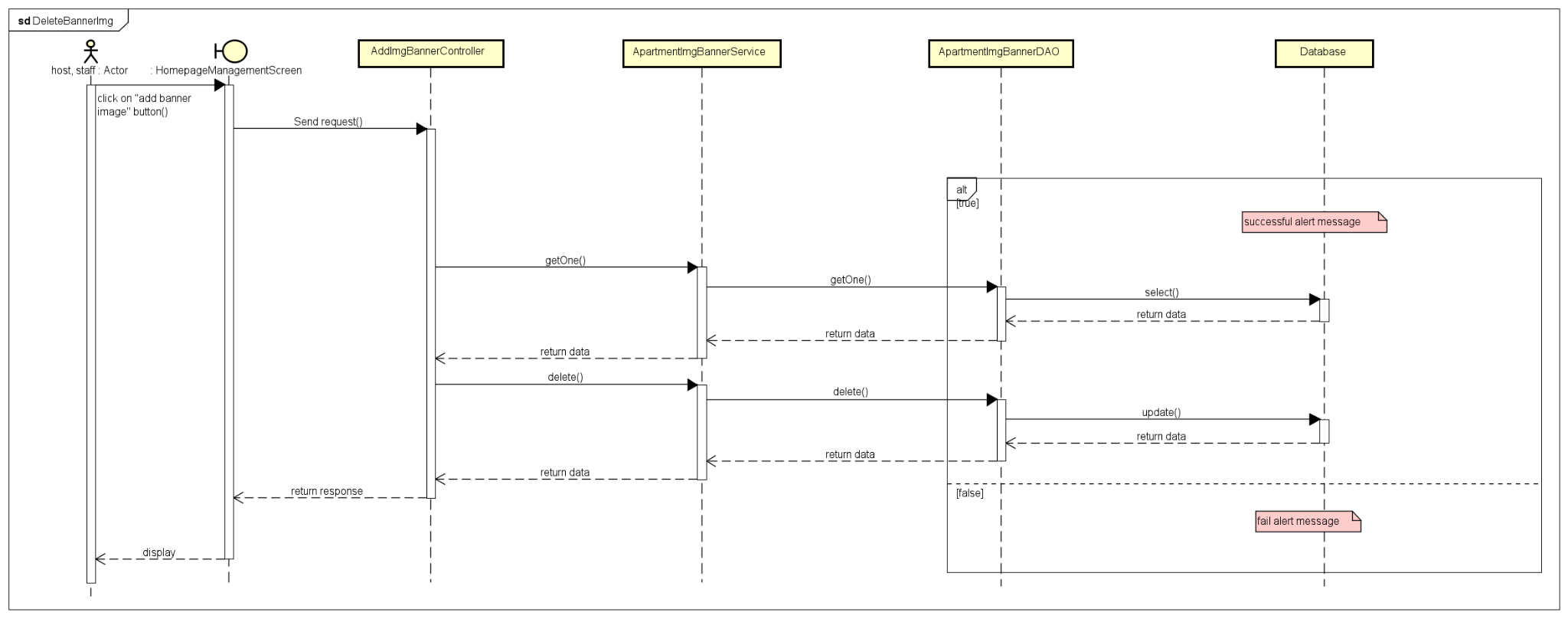
#### Function: Edit HomePage

#### 

#### Function: Add Image Banner



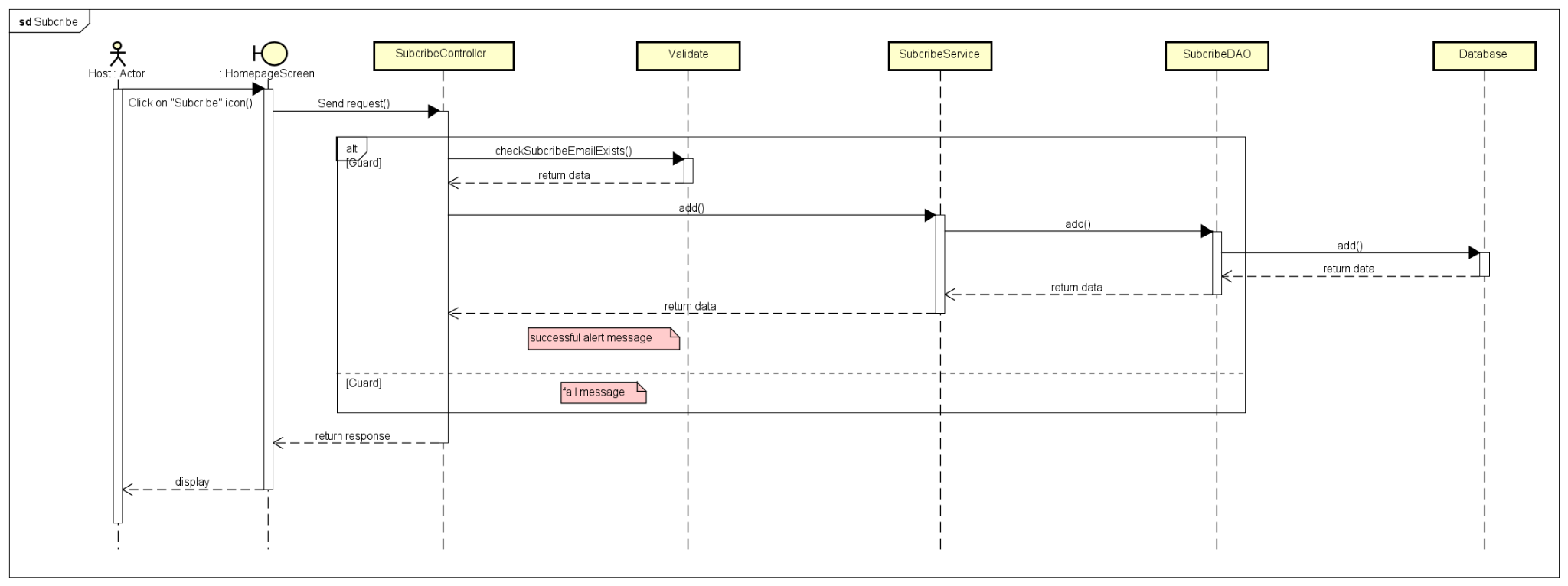
#### Function: Delete Image Banner



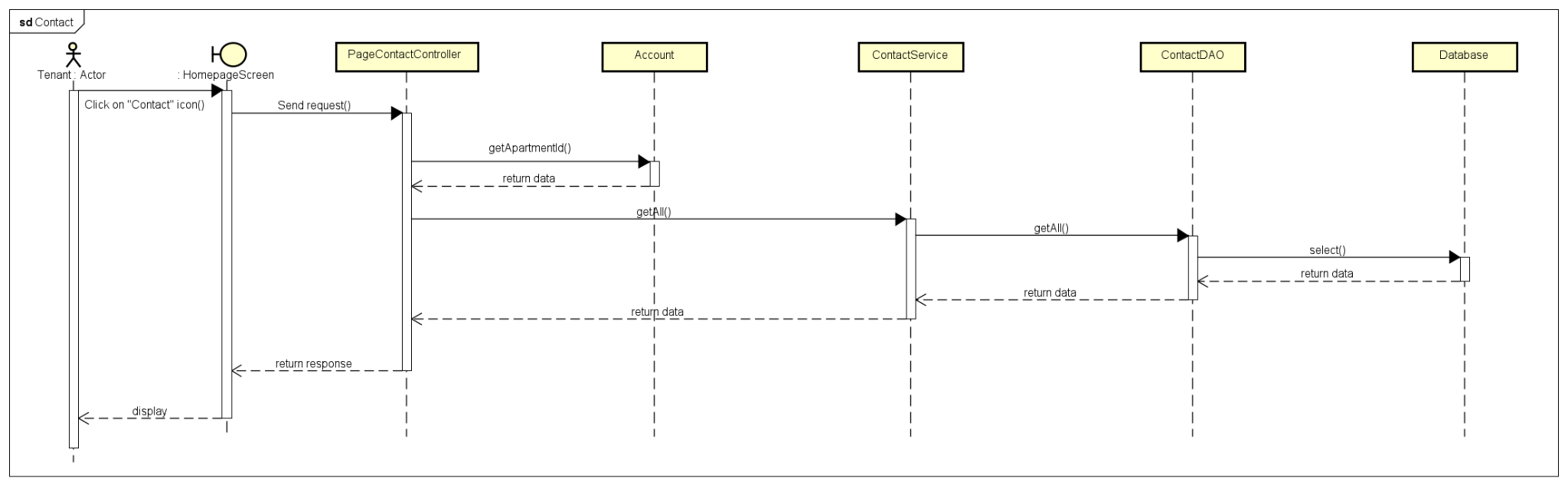
#### Function: Filter Apartment

#### 

#### Function: Subcribe



#### Function: Contact



### d. Database Queries

* ***Create host account function queries:***

*"INSERT INTO `account`(apartment\_id, account\_username, account\_password, account\_accessible, role\_id, deleted)" + " VALUES(?, ?, ?, ?, ?, ?)"*

* ***Edit host account function queries:***

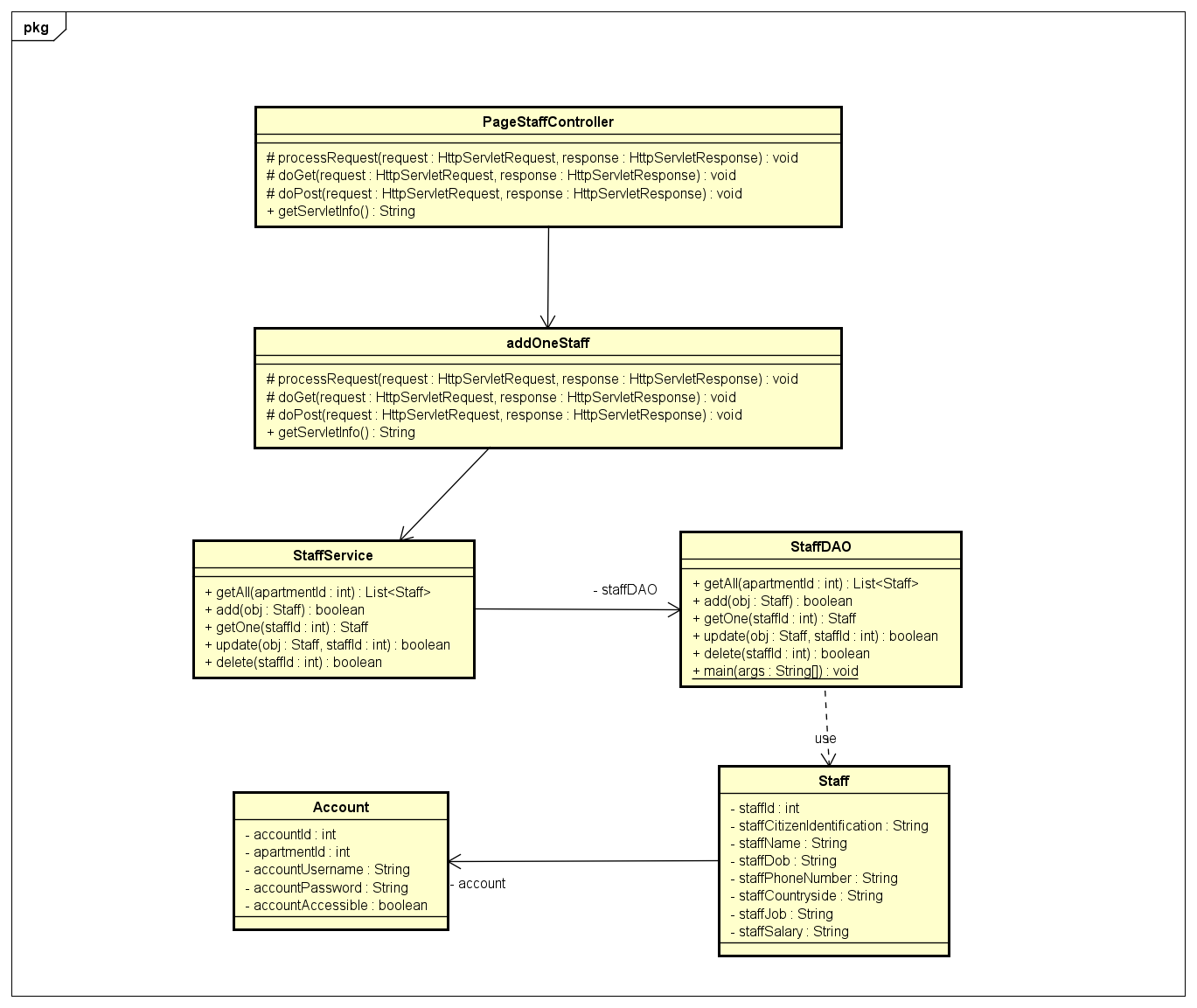
*"UPDATE apamandb.`account` SET apartment\_id = ?, account\_username = ?, account\_password = ?, account\_accessible = ?, role\_id = ? WHERE account\_id = ?"*

* ***Delete host account function queries:***

*UPDATE `account` SET deleted = 1 WHERE account\_id = ?*

## 4. Feature 4: Manage Staff

### a. Class Diagram



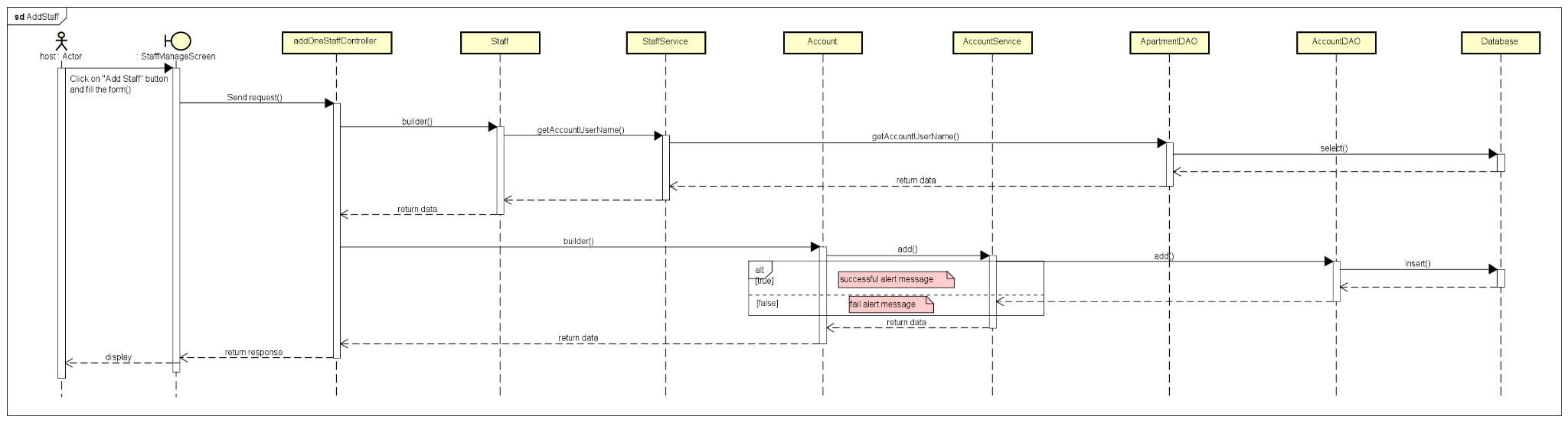
### b. Class Specifications

#### 

#### 

### c. Sequence Diagram(s)

#### Function: Add Staff



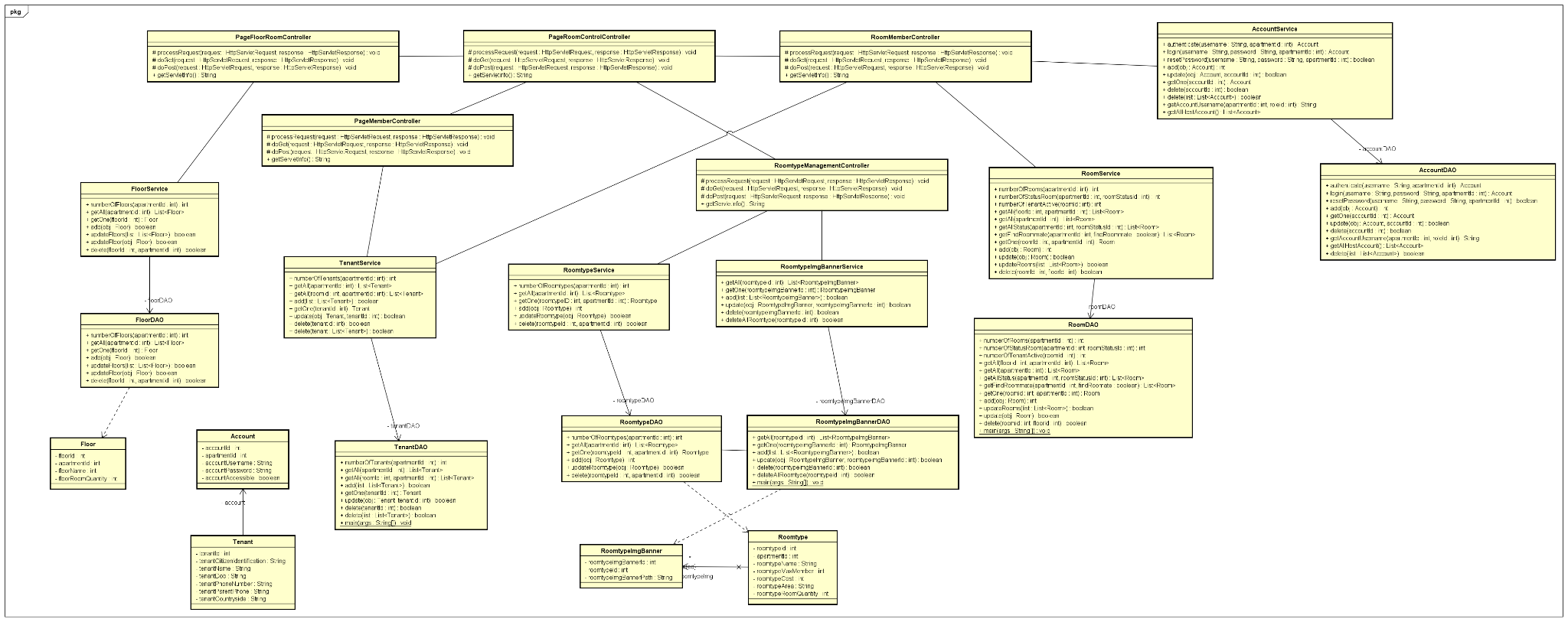
### d. Database Queries

* ***Add staff function queries:***

*"INSERT INTO staff(account\_id, staff\_citizen\_identification, staff\_name, staff\_dob, staff\_phone\_number, staff\_countryside, staff\_job, staff\_salary)" + " VALUES(?, ?, ?, ?, ?, ?, ?, ?)"*

## 5. Feature 5: Manage Room

### a. Class Diagram



### b. Class Specifications

RoomMemberController

| No | Method | Description |
| --- | --- | --- |
| 1 | doGet | displayed all tenant information in one room |
| 2 | doPost | update and delete tenant in room |

PageMemberController

| No | Method | Description |
| --- | --- | --- |
| 1 | doGet | View tenant vehicle \*\* |
| 2 | doPost | N/A |

RoomTypeManagementController

| No | Method | Description |
| --- | --- | --- |
| 1 | doGet | View room type detail by apartmentID and roomtypeID |
| 2 | doPost | Update and Delete room type of apartment |

PageRoomControlController

| No | Method | Description |
| --- | --- | --- |
| 1 | processRequest | View menu control and count floor, room, room type, tenant and staff of apartment |

FloorService

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get one floor by floor\_id |
| 2 | add | add new floor into apartment |
| 3 | update | edit floor information |
| 4 | delete | delete one floor |
| 5 | getAll | get all floor by apartment\_id |

TenantService

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get one tenant information by tenant\_id |
| 2 | add | add one tenant into room |
| 3 | update | edit tenant information |
| 4 | delete | delete tenant |
| 5 | getAll | get all tenant of room |

RoomTypeService

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get one room type by roomtypeId and apartmentId |
| 2 | add | add new room type |
| 3 | update | edit room type information |
| 4 | delete | delete a room type exist in apartment |
| 5 | getAll | get all room type of apartment |

RoomTypeImgBannerService

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get a Roomtype image banner by roomtypeImgBannerId |
| 2 | add | add room type image |
| 3 | update | edit room type image |
| 4 | delete | delete room type image |
| 5 | getAll | get all room type image of room type |

RoomService

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get a room by roomId and apartmentId |
| 2 | add | add new room into apartment |
| 3 | update | update room information in apartment |
| 4 | delete | delete one room in apartment |
| 5 | getAll | get all room exist in apartment by apartmentId |

AccountService

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get one account by accountId |
| 2 | add | add new account |
| 3 | update | update account exist in database |
| 4 | delete | delete account exist in database |
| 5 | getAll | get all account in database |

FloorDAO

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get a floor by floorId |
| 2 | add | add new floor into apartment |
| 3 | update | edit floor information |
| 4 | delete | delete a floor in apartment |
| 5 | getAll | get all floor in apartment by apartmentId |

TenantDAO

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get one Tenant by tenantId |
| 2 | add | add new tenant |
| 3 | update | edit tenant information |
| 4 | delete | delete tenant of room |
| 5 | getAll | get all tenant in room by roomId |

RoomTypeDAO

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get a room type by roomtypeId |
| 2 | add | add new room type into apartment |
| 3 | update | edit room type information |
| 4 | delete | delete a room type |
| 5 | getAll | get all room type of apartment by apartmentId |

RoomTypeImgBannerDAO

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get a Roomtype image banner by roomtypeImgBannerId |
| 2 | add | add new room type image banner |
| 3 | update | edit image banner of room type |
| 4 | delete | delete room type image banner |
| 5 | getAll | get all room type image banner of room type by roomtypeId |

RoomDAO

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get a room of apartment by roomId and apartmentId |
| 2 | add | add new room into apartment |
| 3 | update | edit room information |
| 4 | delete | delete a room in apartment |
| 5 | getAll | get all room in apartment by apartmentId |

AccountDAO

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get one account by accountId |
| 2 | add | add new account |
| 3 | update | update account exist in database |
| 4 | delete | delete account exist in database |
| 5 | getAll | get all account in database |

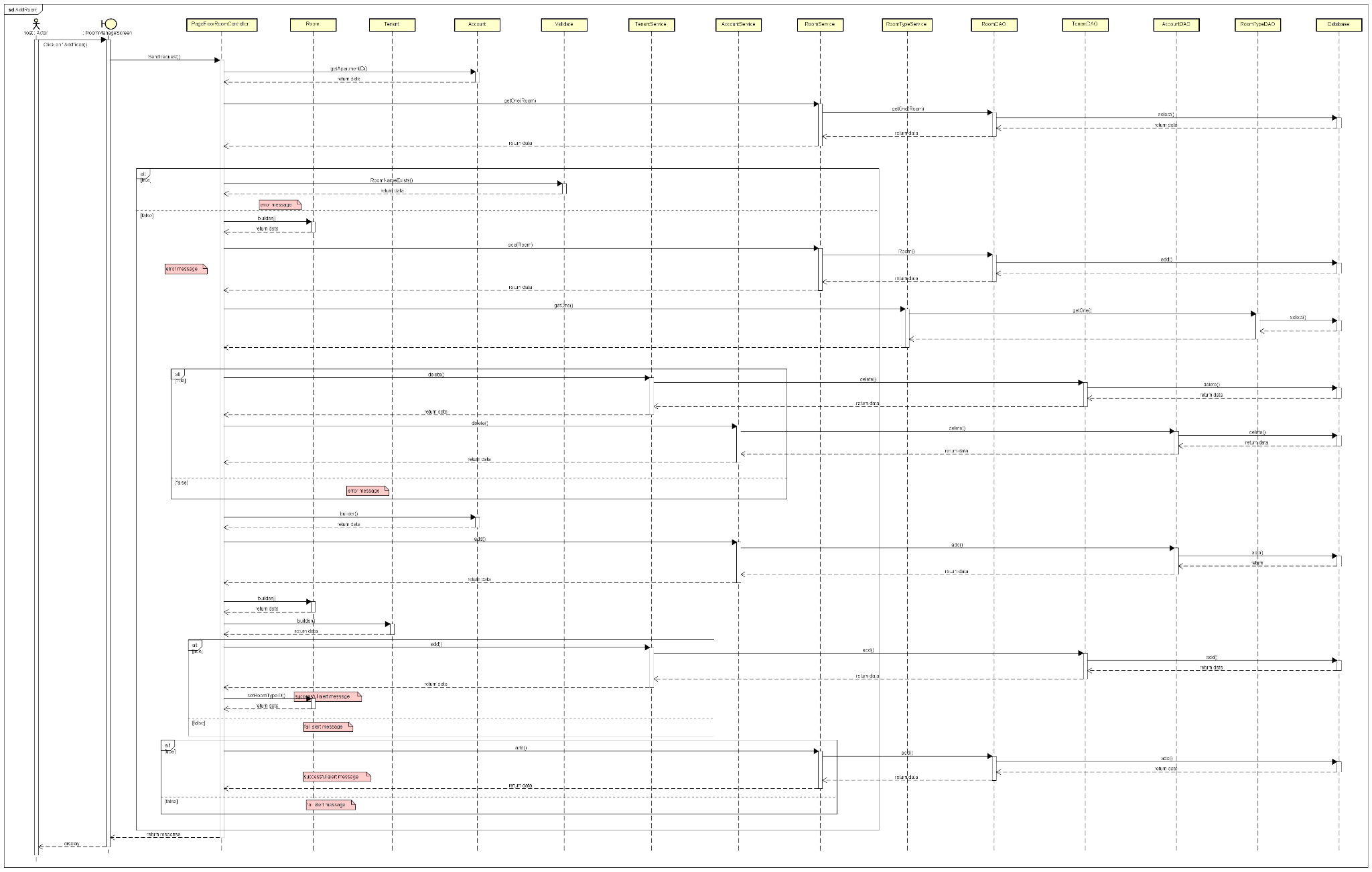
Floor

| No | Method | Description |
| --- | --- | --- |
| 1 | getOne | get a floor by floorId |
| 2 | add | add new floor into apartment |
| 3 | update | edit floor information |
| 4 | delete | delete a floor in apartment |
| 5 | getAll | get all floor in apartment by apartmentId |

#### 

### c. Sequence Diagram(s)

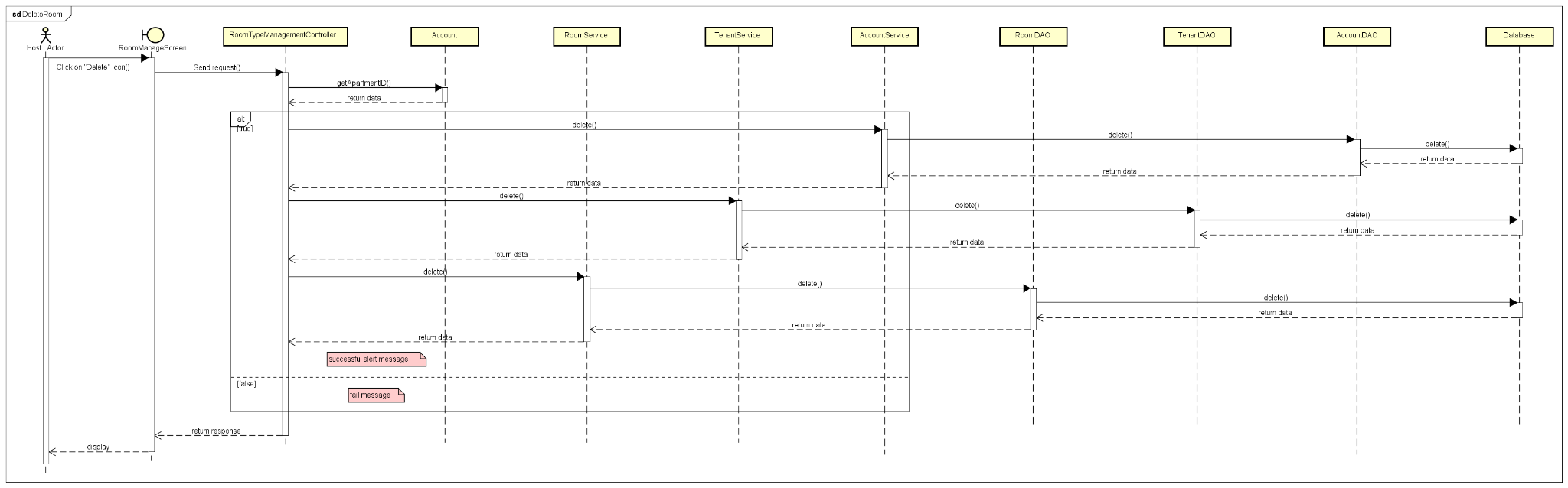
#### Function: Add Room



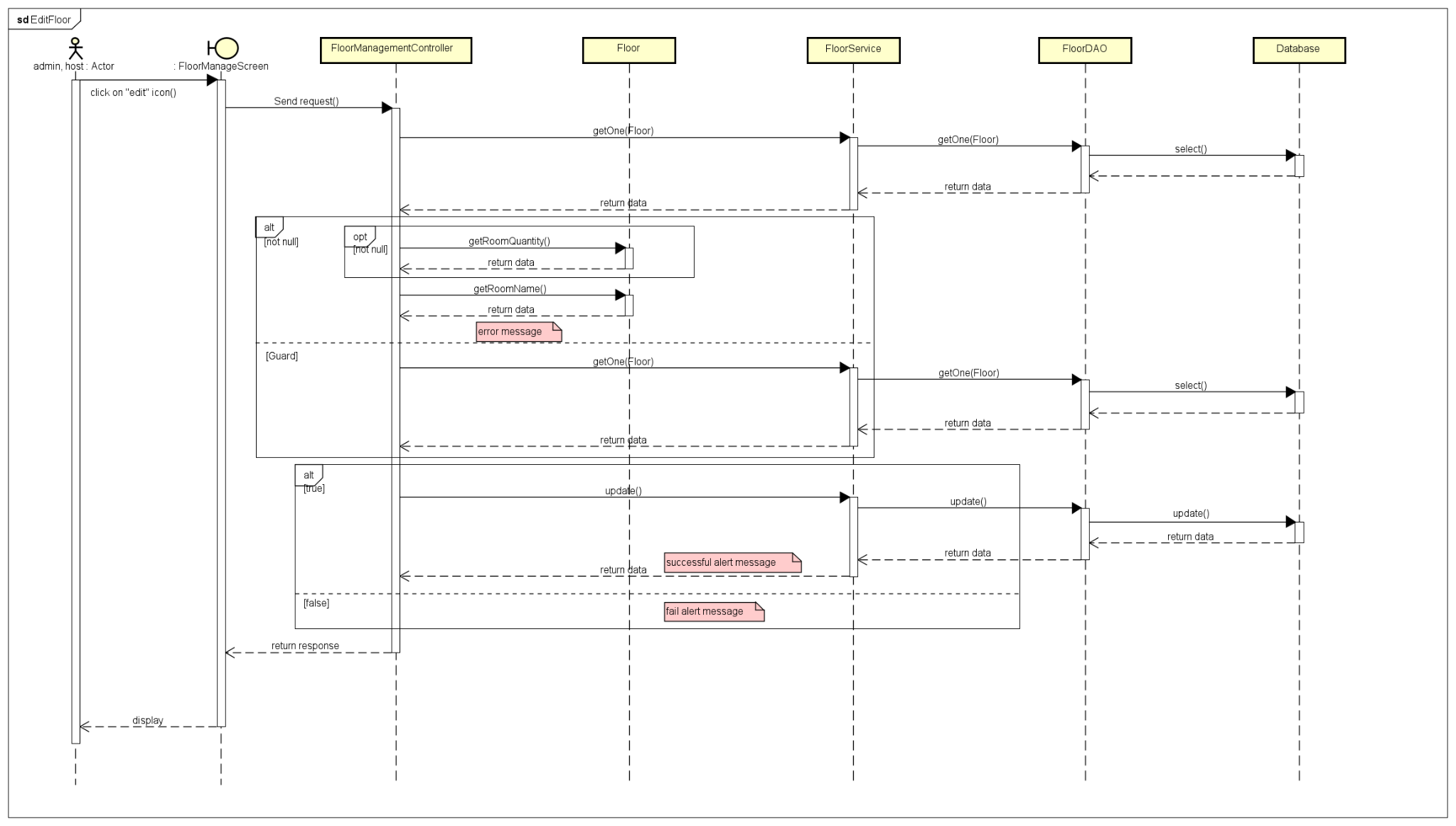
#### Function: Edit Room

#### 

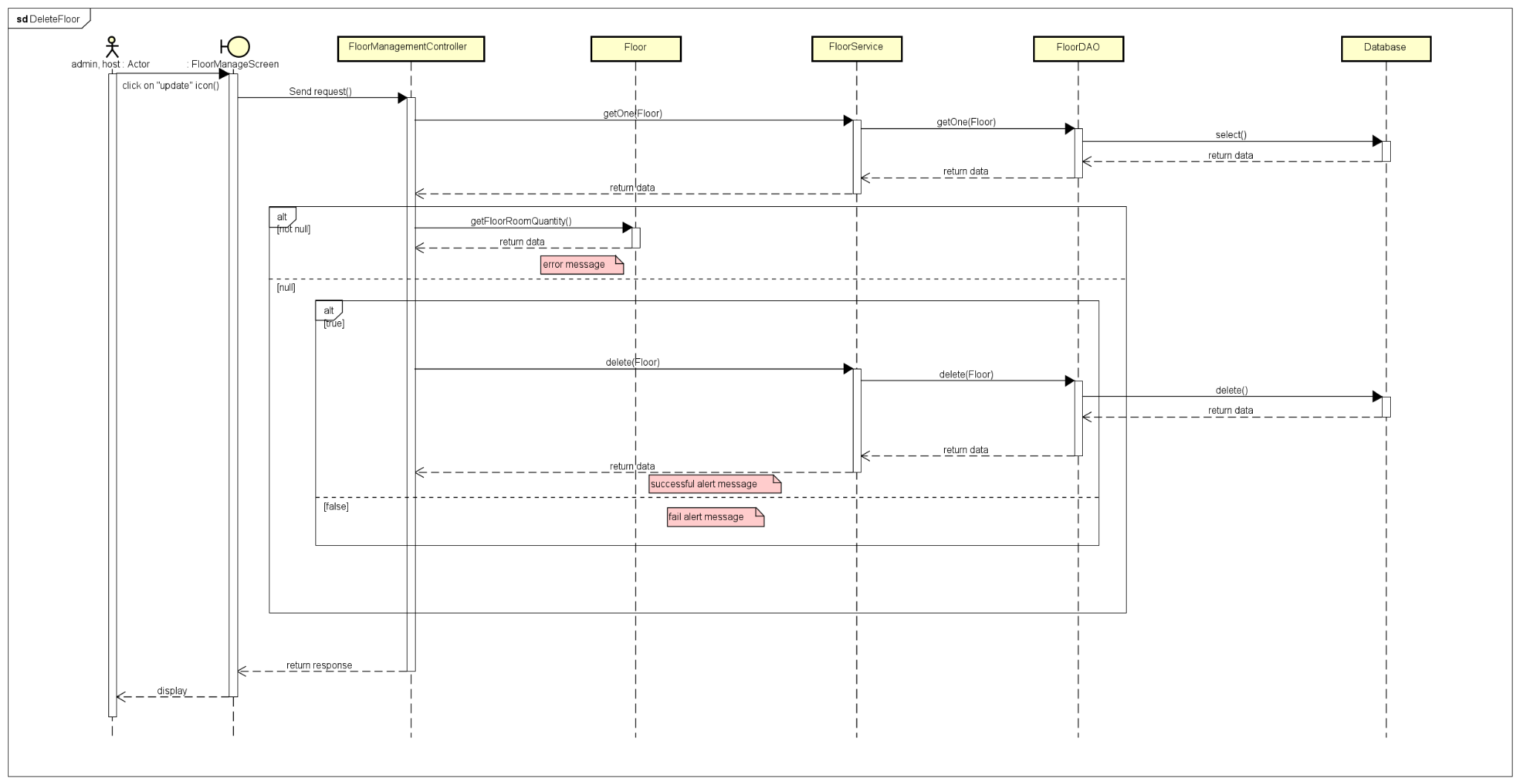
#### Function: Delete Room



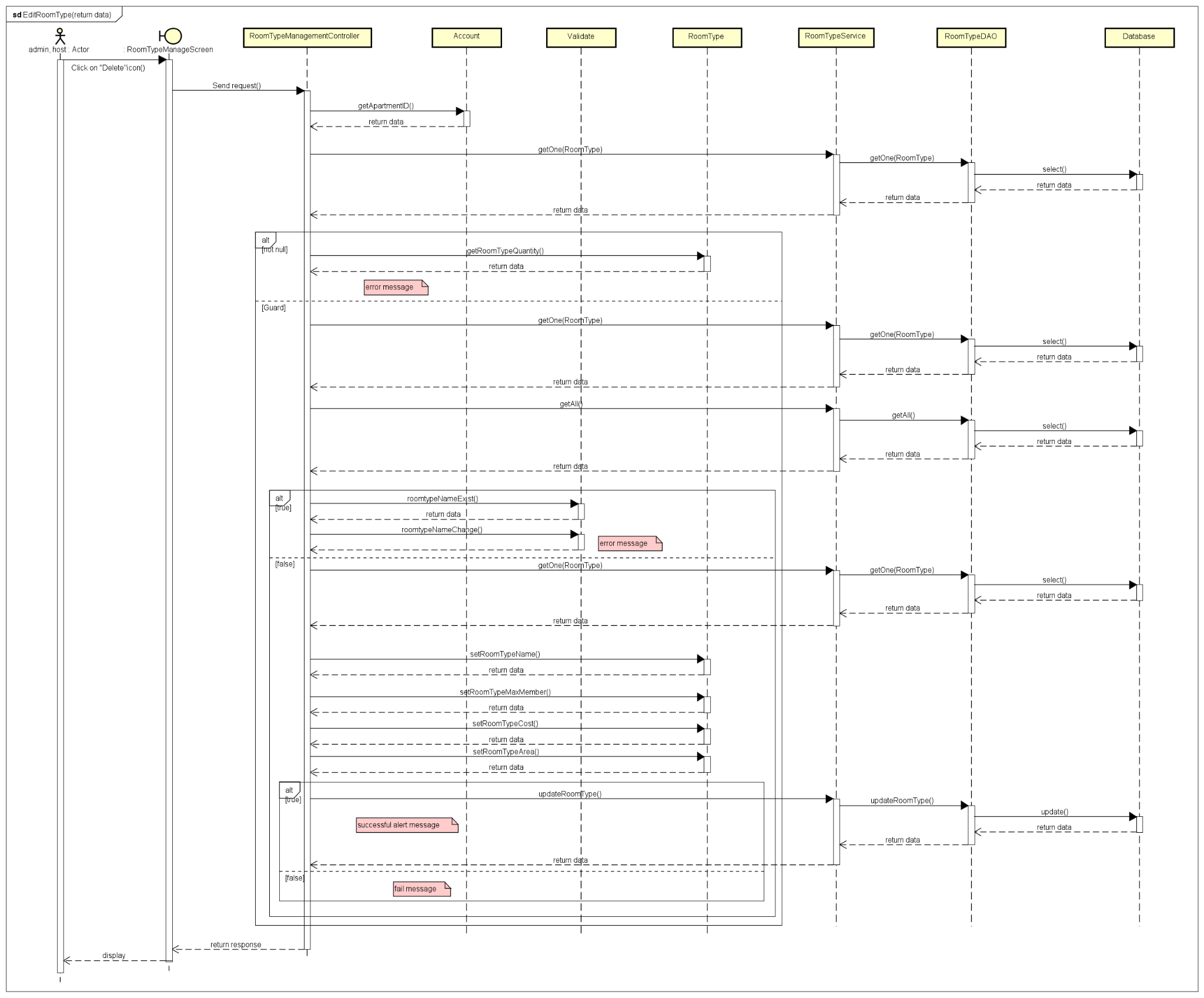
#### Function: Edit Floor



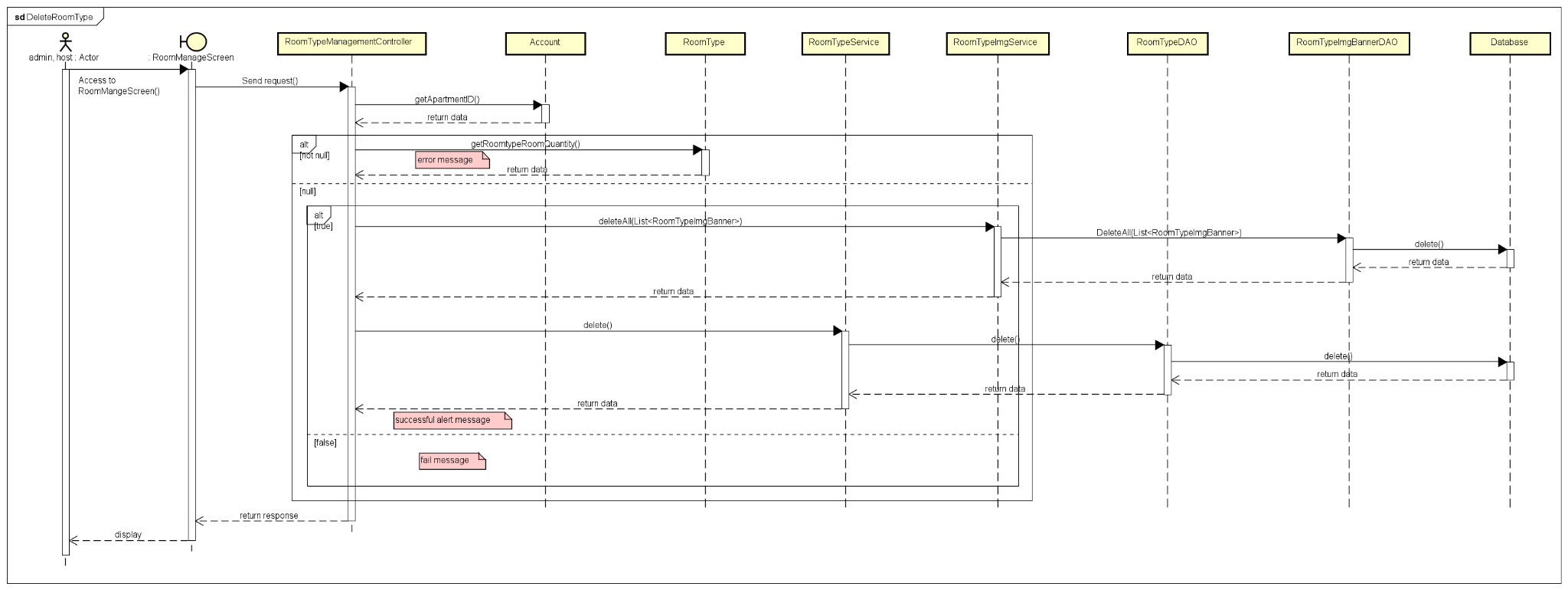
#### Function: Delete Floor



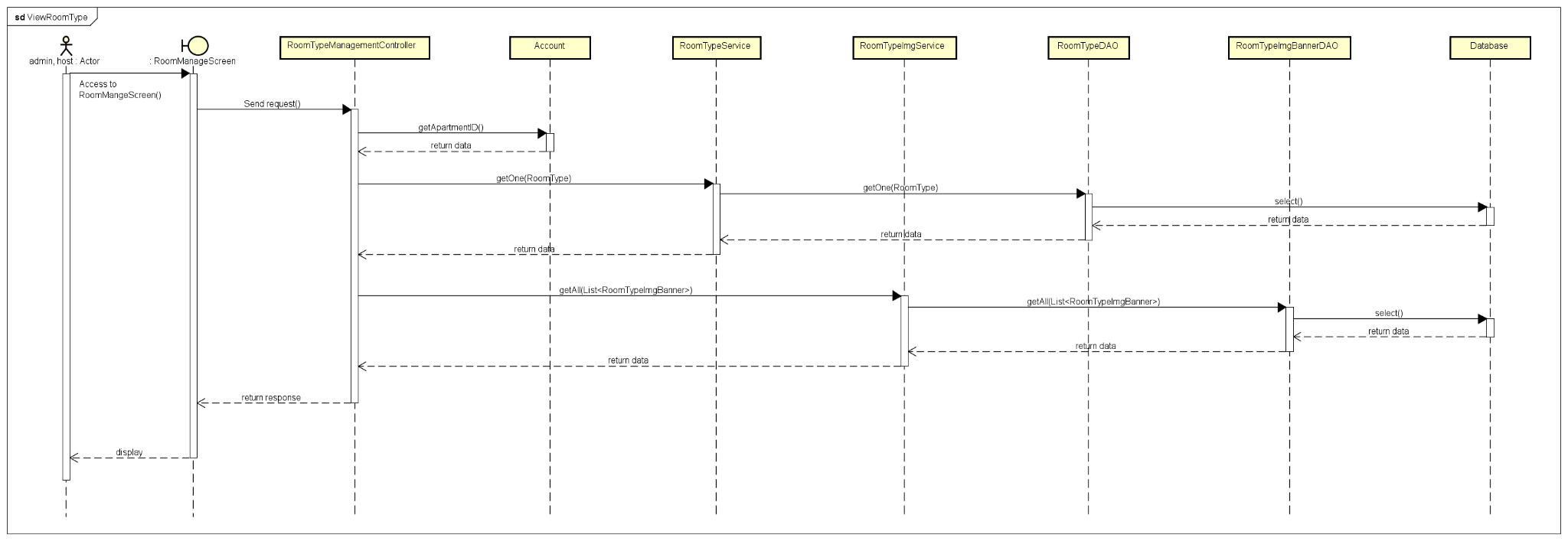
#### Function: Edit RoomType



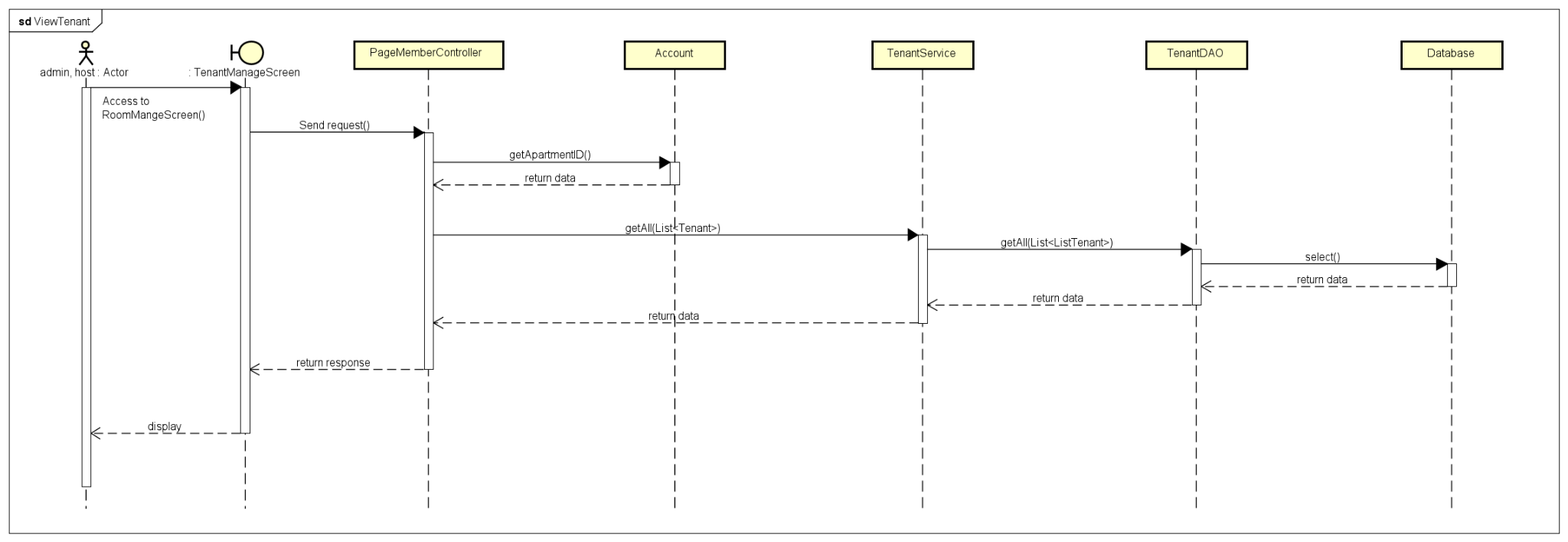
#### Function: Delete RoomType



#### Function: View RoomType

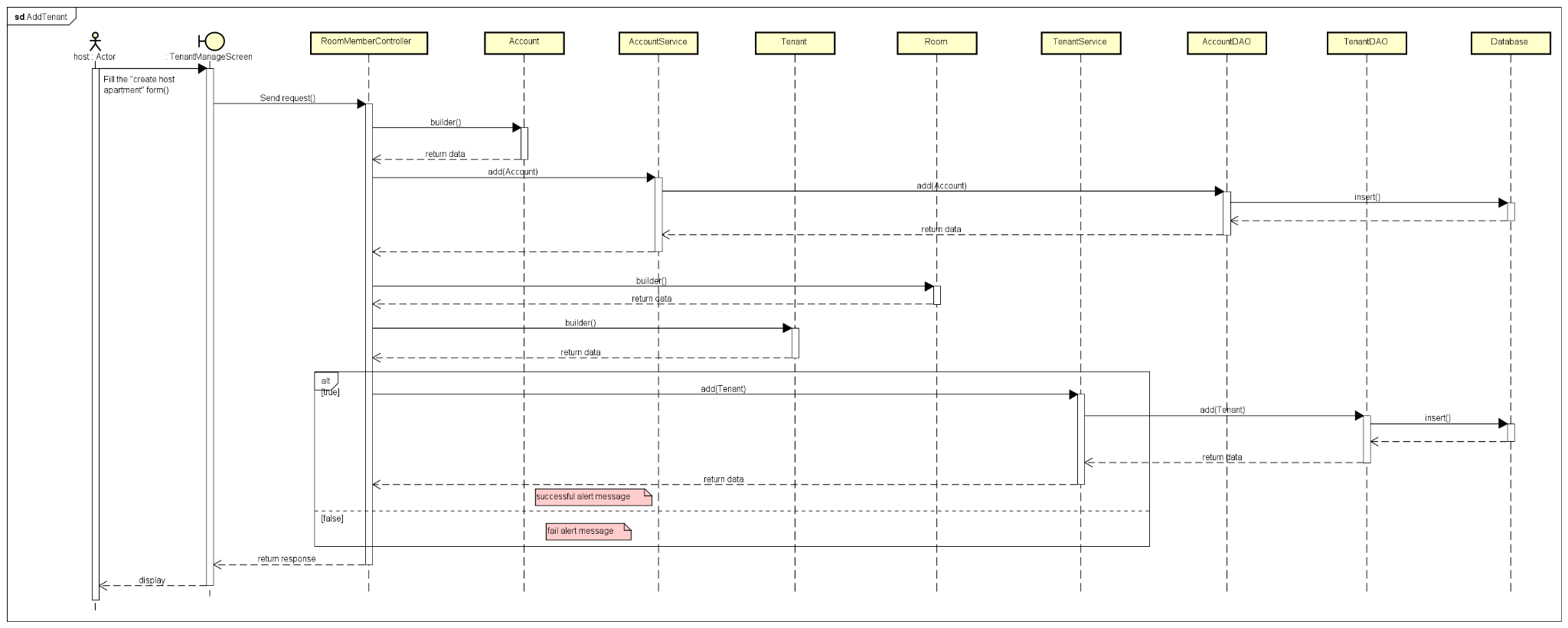


#### Function: View Tenant

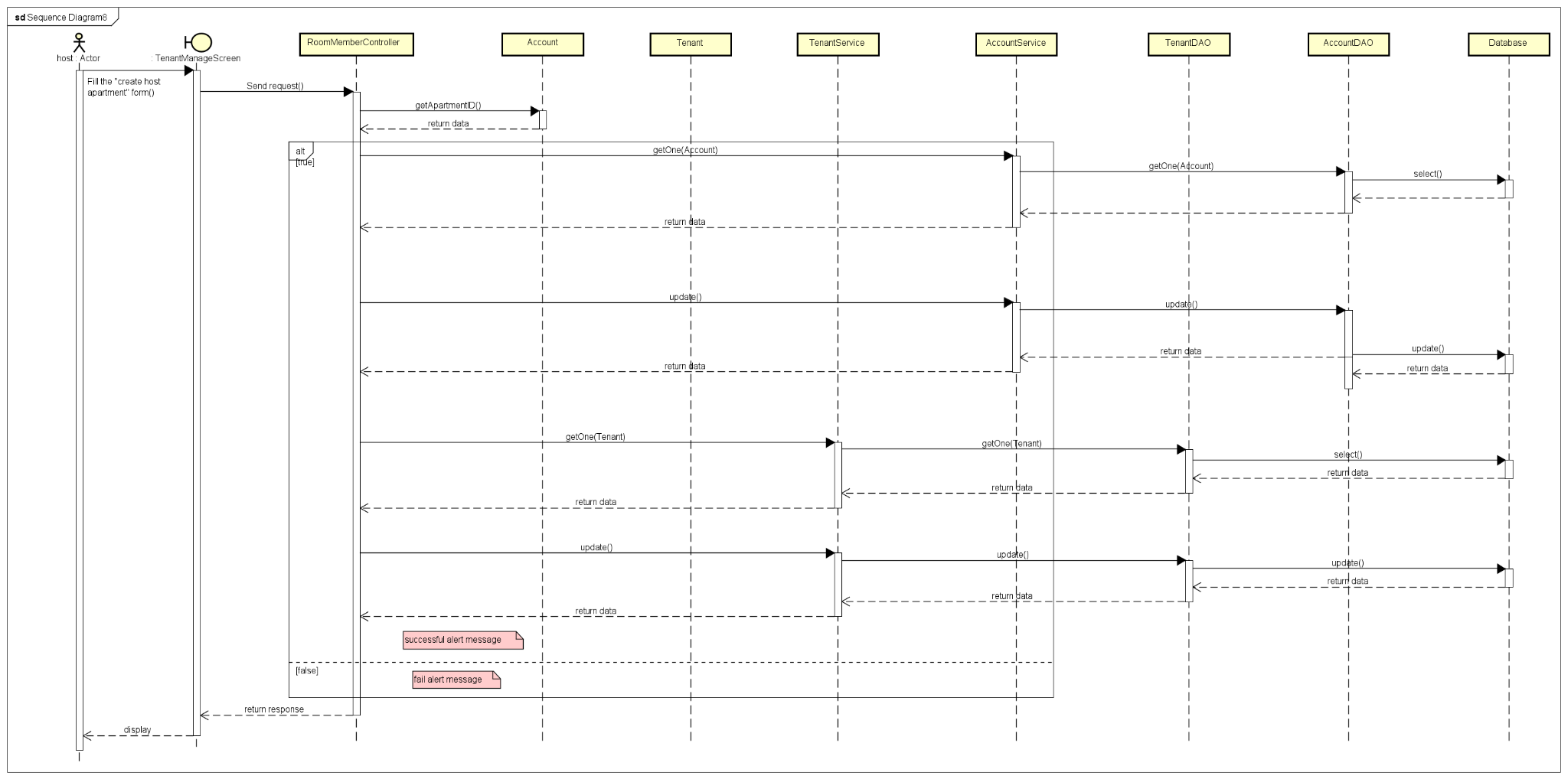


#### 

#### Function: Add Tenant



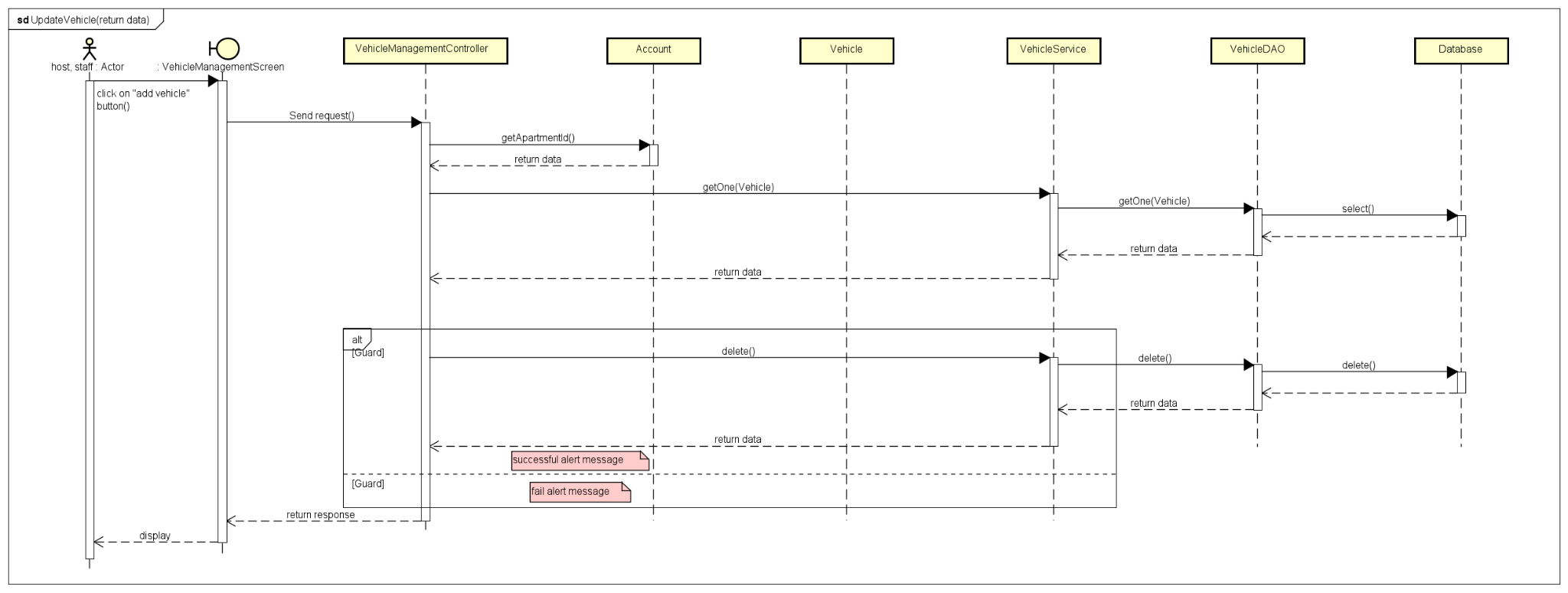
#### Function: Update Tenant



#### Function: Add Vehicle



#### Function: Delete Vehicle



### d. Database Queries

***- Add Room function queries:***

*String sql = "INSERT INTO room(room\_name, roomtype\_id, floor\_id, apartment\_id , find\_roommate, room\_status\_id)"*

*+ " VALUES(?, ?, ?, ? , ? , ?)";*

*-* ***Edit Room function queries:***

*"UPDATE room Set room\_name = ?,roomtype\_id = ?, floor\_id = ?, apartment\_id = ?, room\_status\_id = ?, find\_roommate = ? WHERE room\_id = ?";*

*-* ***Delete Room function queries:***

***“****DELETE FROM room Where room\_id = ? AND floor\_id = ?"*

*-* ***Edit Floor***

*“UPDATE floor Set floor\_name = ?, floor\_room\_quantity = ? WHERE floor\_id = ?";”*

***- Delete Floor***

***“****DELETE FROM floor Where floor\_id = ? AND apartment\_id = ?”*

***- Edit RoomType***

*"UPDATE roomtype Set roomtype\_name = ?, roomtype\_max\_member = ?, roomtype\_cost = ?, roomtype\_area = ?, roomtype\_room\_quantity = ? WHERE roomtype\_id = ?"*

***- Delete RoomType***

*"DELETE FROM roomtype Where roomtype\_id = ? AND apartment\_id = ?"*

***- View RoomType***

*"SELECT \* FROM roomtype WHERE roomtype\_id = ? AND apartment\_id = ?"*

***- View Tenant***

***"****with te as(\n"*

*+ "SELECT t.tenant\_id, "*

*+ "t.account\_id, "*

*+ "t.room\_id, "*

*+ "t.tenant\_citizen\_identification, "*

*+ "t.tenant\_name, "*

*+ "t.tenant\_dob, "*

*+ "t.tenant\_phone\_number, "*

*+ "t.tenant\_parent\_phone, "*

*+ "t.tenant\_countryside, "*

*+ "a.apartment\_id,"*

*+ "a.account\_username,"*

*+ "a.account\_password,"*

*+ "a.account\_accessible,"*

*+ "a.role\_id\n"*

*+ "FROM apamandb.tenant t Join apamandb.account a ON t.account\_id = a.account\_id \n"*

*+ "Where a.apartment\_id = ? AND a.account\_accessible = true) \n"*

*+ "\n"*

*+ "select te.\* , r.room\_name\n"*

*+ "FROM te JOIN apamandb.room r on te.room\_id = r.room\_id"*

***- Update Tenant***

*"UPDATE tenant SET\n"*

*+ "tenant\_citizen\_identification = ?, "*

*+ "tenant\_name = ?, "*

*+ "tenant\_dob = ?, "*

*+ "tenant\_phone\_number = ?, "*

*+ "tenant\_parent\_phone = ?, "*

*+ "tenant\_countryside = ?"*

*+ "WHERE tenant\_id = ?"*

***- Add Vehicle***

*"INSERT INTO vehicle(vehicle\_type\_id, vehicle\_license\_plate, vehicle\_description,tenant\_id,room\_id,apartment\_id,vehicle\_img\_path)"*

*+ " VALUES(?, ?, ?, ?, ?, ?, ?)"*

***- Update Vehicle***

*"UPDATE vehicle Set vehicle\_type\_id = ?, "*

*+ "vehicle\_license\_plate = ? , "*

*+ "vehicle\_description = ? ,"*

*+ "tenant\_id = ? ,"*

*+ "room\_id = ? ,"*

*+ "apartment\_id = ?,"*

*+ "vehicle\_img\_path = ?"*

*+ "WHERE vehicle\_id = ?"*

### 

## 6. Feature 6: Manage Payment

### a. Class Diagram

### b. Class Specifications

PaymentController

| No | Method | Description |
| --- | --- | --- |
| 1 | doGet | displayed last month b |
| 2 | doPost | update and delete tenant in room |

GetElectricWaterControllerController

| No | Method | Description |
| --- | --- | --- |
| 1 | doGet | Display last updated payment |
| 2 | doPost |  |

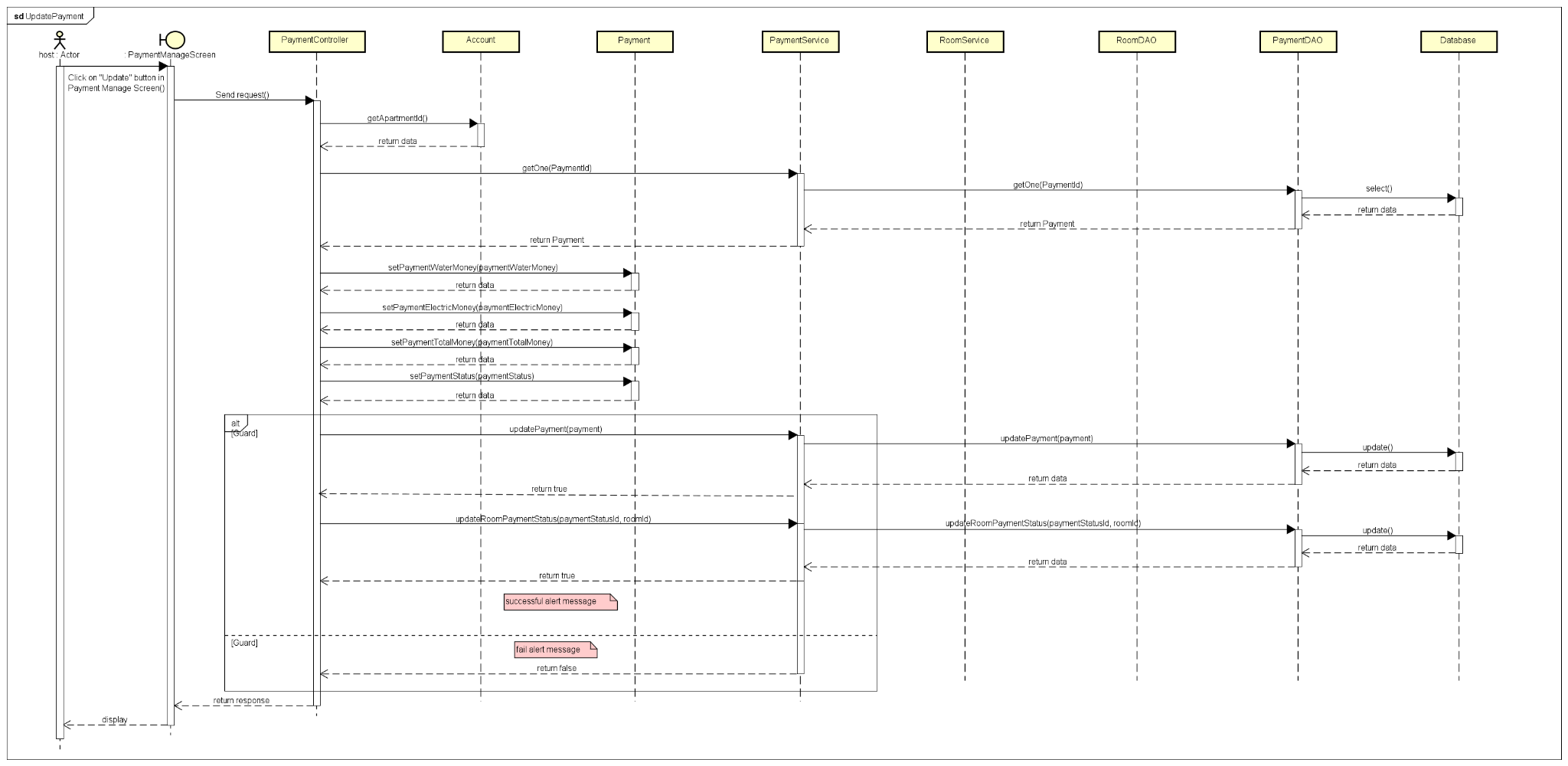
confirmPaymentControllerController

| No | Method | Description |
| --- | --- | --- |
| 1 | doGet | View room type detail by apartmentID and roomtypeID |
| 2 | doPost | Update and Delete room type of apartment |

#### 

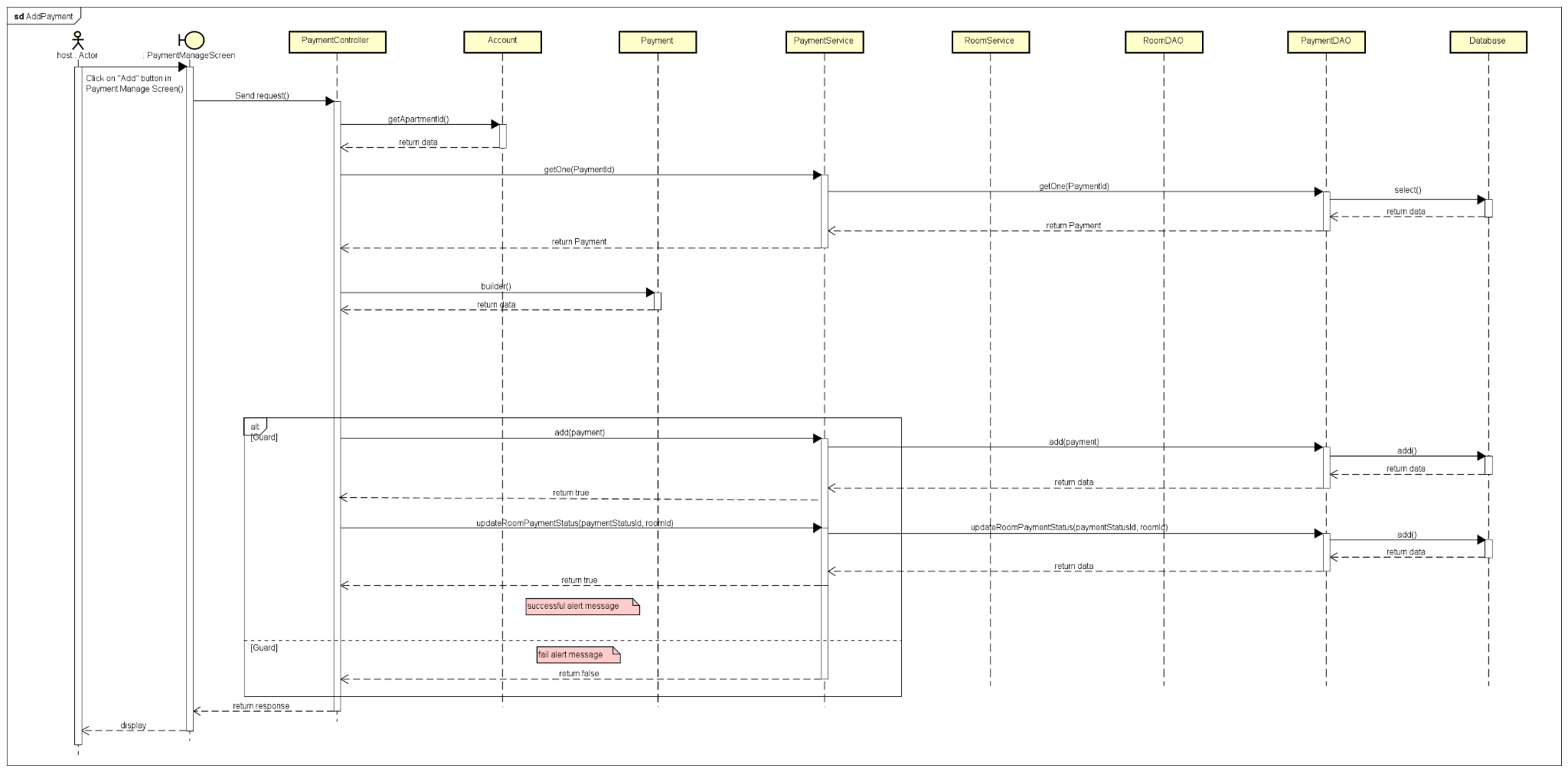
### c. Sequence Diagram(s)

### Update Payment

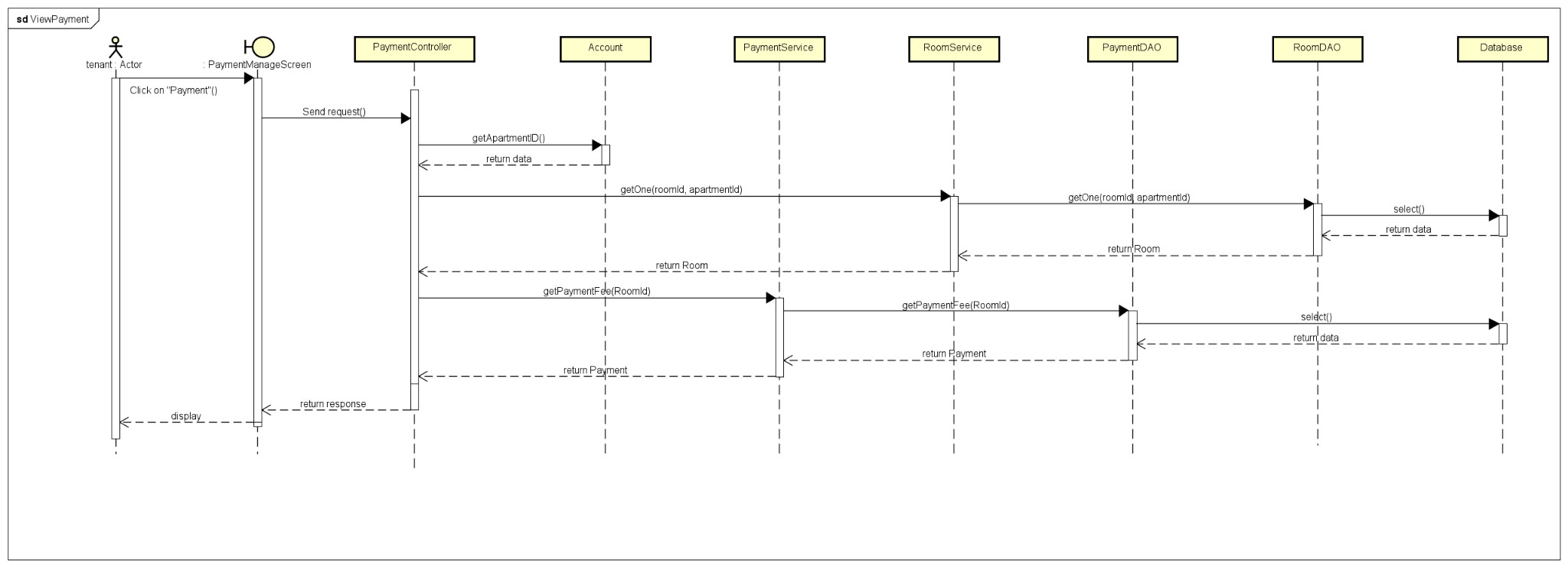


### 

### Add Payment



### View Payment



### d. Database Queries

***- Add Payment function queries:***

*"INSERT INTO payment(room\_id, payment\_room\_unit\_fee, "*

*+ "payment\_water\_index\_pre, payment\_electric\_index\_pre, payment\_day\_update\_pre, "*

*+ "payment\_water\_index\_cur, payment\_electric\_index\_cur, payment\_day\_update\_cur, "*

*+ "payment\_water\_unit\_fee, payment\_water\_money,"*

*+ "payment\_electric\_unit\_fee, payment\_electric\_money,"*

*+ "payment\_car\_quantity, payment\_car\_unit\_fee, payment\_car\_money,"*

*+ "payment\_motor\_quantity, payment\_motor\_unit\_fee, payment\_motor\_money,"*

*+ "payment\_bike\_quantity, payment\_bike\_unit\_fee, payment\_bike\_money,"*

*+ "payment\_total\_money,"*

*+ "payment\_status\_id)"*

*+ " VALUES(?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,?)"*

*"UPDATE room Set room\_payment\_status\_id = ? WHERE room\_id = ?"*

*-* ***Edit Payment function queries:***

*"UPDATE payment Set payment\_water\_money = ?, payment\_electric\_money = ? , payment\_total\_money = ?, payment\_status\_id = ? WHERE payment\_id = ?"*

*"UPDATE room Set room\_payment\_status\_id = ? WHERE room\_id = ?"*

* ***View Payment function queries:***