

**<<PROJECT NAME>>**

**Software Design Document**

– Hanoi, August 2019 –

**Table of Contents**

[I. Overview 3](#_heading=h.gjdgxs)

[1. Code Packages 3](#_heading=h.30j0zll)

[2. Database Schema 3](#_heading=h.1fob9te)

[II. Code Designs 4](#_heading=h.3znysh7)

[1. <Feature/Function Name1> 4](#_heading=h.tyjcwt)

[a. Class Diagram 4](#_heading=h.3dy6vkm)

[b. Class Specifications 4](#_heading=h.1t3h5sf)

[c. Sequence Diagram(s) 4](#_heading=h.4d34og8)

[d. Database queries 5](#_heading=h.2s8eyo1)

[2. <Feature/Function Name2> 5](#_heading=h.17dp8vu)

[III. Database Tables 5](#_heading=h.3rdcrjn)

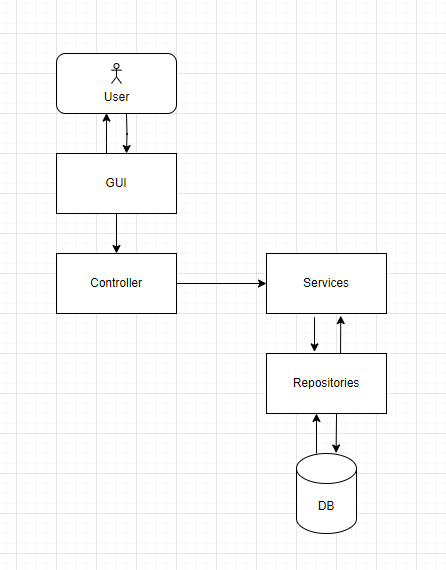
[1. <Table name 1> 5](#_heading=h.26in1rg)

[2. <Table name 2…> 5](#_heading=h.lnxbz9)

# I. Overview

## 1. Code Packages/Namespaces

*[Provide the package diagram for each sub-system. The content of this section including the overall package diagram, the explanation, package and class naming conventions in each package. Please see the sample and description table format below – following Java project naming convention]*

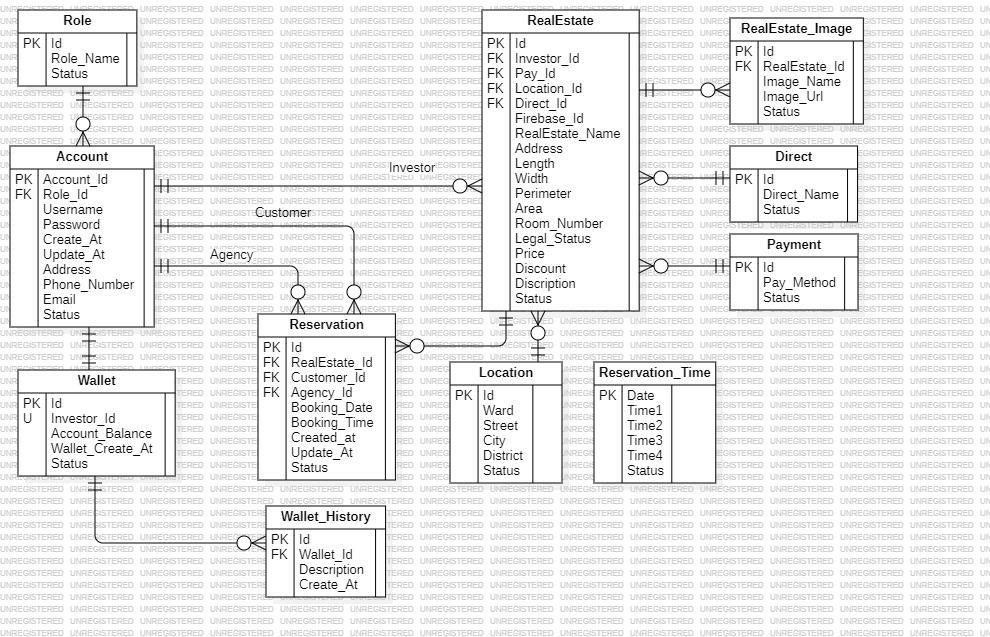


***Package descriptions & package class naming conventions***

| **No** | **Package** | **Description** |
| --- | --- | --- |
| *01* | *Controller* | *- <Description of the package>*  *The controller typically handles requests coming from clients, such as web browsers or mobile apps, and decides what to do with them. Then redirect the request to a service that can handle it*  *- <Class naming convention> NamefuntionController* |
| *02* | *Services* | *- <Description of the package>*  *Services encapsulate the application's business logic. They perform operations requested by the controllers, such as data processing, manipulation, validation, and interaction with repositories.*  *- <Class naming convention> NamefuntionService* |
| *03* | *Repositories* | *- <Description of the package>*  *Repositories handle the interactions with the database. They abstract away the details of the database operations and provide a clean interface for the services to interact with the data.*  *- <Class naming convention> NamefuntionRepositories* |

## 2. Database Schema

*[Provide the tables relationship like example below – following SQL Server database naming convention]*



***Table descriptions & package class naming conventions are as below***

| **No** | **Table** | **Description** |
| --- | --- | --- |
| *01* | *<Table Role>* | *- Primary keys: Id* |
| *02* | *<Table Account>* | *- Primary keys: Id*  *-Foreign keys: Role\_Id* |
| *03* | *<Table Wallet>* | *- Primary keys: Id*  *-Unique keys: Investor\_Id* |
| *04* | *<Table Wallet\_History>* | *- Primary keys: Id*  *- Foreign keys: Wallet\_Id* |
| *05* | *<Table Reservation>* | *- Primary keys: Id*  *- Foreign keys: RealEstate\_Id*  *- Foreign keys: Customer\_Id*  *- Foreign keys: Agency\_Id* |
| *06* | *<Table RealEstate>* | *- Primary keys: Id*  *- Foreign keys: Investor\_Id*  *- Foreign keys: Pay\_Id*  *- Foreign keys: Direct\_Id* |
| *07* | *<Table Location>* | *- Primary keys: Id* |
| *08* | *<Table RealEstate\_Image>* | *- Primary keys: Id*  *- Foreign keys: RealEstate\_Id* |
| *09* | *<Table Direct>* | *- Primary keys: Id* |
| *10* | *<Table Payment>* | *- Primary keys: Id* |
| *11* | *<Table Reservation Time>* | *- Primary keys: Date* |

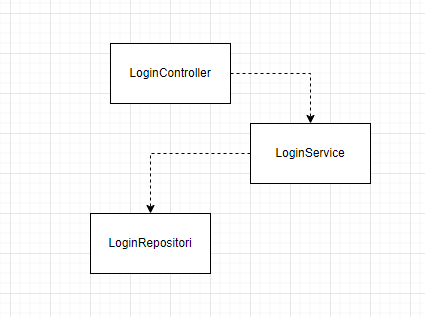
# II. Code Designs

## 1. <Feature/Function Login>

*[Provide the detailed design for the function <Feature/Function Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*



### b. Class Specifications

*[Provide the description for each class and the methods in each class, following the table format as below]*

#### XYZ Class

*[Provide the detailed description for the class methods]*

| **No** | **Method** | **Description** |
| --- | --- | --- |
| *01* | *<method name>* | *<Description of the method, including the inputs, outputs & internal method processing>* |
|  |  |  |

#### ABC Class

***Class Methods***

*[Provide the detailed description for the class methods]*

| **No** | **Method** | **Description** |
| --- | --- | --- |
| *01* | *<method name>* | *<Description of the method, including the inputs, outputs & internal method processing>* |
|  |  |  |

### c. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature, see the sample below]*



### d. Database queries

*[Provide the detailed SQL (select, insert, update...) which are used in implementing the function/screen]*

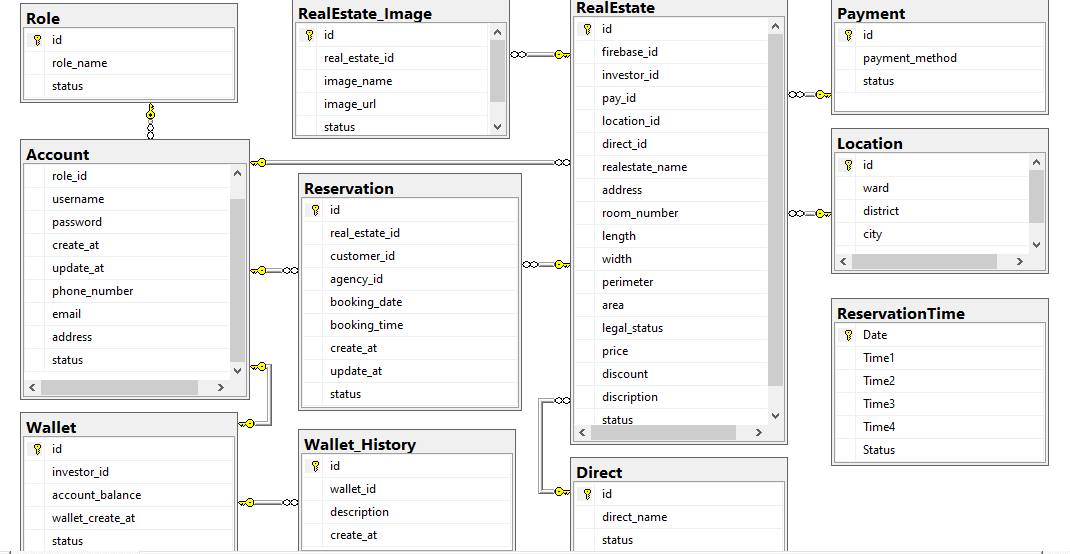
## 2. <Feature/Function Post>

## 2. <Feature/Function Banking>

## 2. <Feature/Function Reservations>

## 2. <Feature/Function Banking>

# III. Database Tables



***SWP\_VillaRealEstate \_Database\_Diagram***

## 

## 1. <Table Role >

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | role\_name | nvarchar | 20 |  | True |  |  |
| 3 | status | bit |  |  | True |  |  |

## 

## 2. <Table Account>

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  | True | True | PK |  |
| 2 | role\_id | int |  |  | True | FK |  |
| 3 | username | nvarchar | 50 |  | True |  |  |
| 4 | password | nvarchar | 50 |  | True |  |  |
| 5 | create\_at | date |  |  | True |  |  |
| 6 | update\_at | date |  |  | False |  |  |
| 7 | phone\_number | nvarchar | 50 |  | True |  |  |
| 8 | email | nvarchar | 50 |  | True |  |  |
| 9 | address | nvarchar | 50 |  | True |  |  |
| 10 | status | bit |  |  | True |  |  |

## 3. <Table Reservation>

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | real\_estate\_id | int |  |  | True | FK |  |
| 3 | customer\_id | int |  |  | True | FK |  |
| 4 | agency\_id | int |  |  | True | FK |  |
| 5 | booking\_date | date |  |  | True |  |  |
| 6 | booking\_time | nvarchar | 50 |  | True |  |  |
| 7 | create\_at | nvarchar | 50 |  | True |  |  |
| 8 | update\_at | nvarchar | 50 |  | False |  |  |
| 9 | status | bit |  |  | True |  |  |

## 4. <Table Wallet>

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | investor\_id | int |  | True | True |  |  |
| 3 | account\_ballance | nvarchar | Max |  | False | FK |  |
| 4 | wallet\_create\_at | datetime |  |  | True | FK |  |
| 5 | status | bit |  |  | True |  |  |

## 5. <Table Wallet\_History>

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | wallet\_id | int |  |  | True |  |  |
| 3 | description | nvarchar | Max |  | False |  |  |
| 4 | create\_at | datetime |  |  | True |  |  |

## 6. <Table RealEstate>

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | firebase\_id | nvarchar | Max |  | True |  |  |
| 3 | Investor\_id | int |  |  | True | FK |  |
| 4 | Pay\_id | int |  |  | True | FK |  |
| 5 | Location\_id | Int |  |  | True | FK |  |
| 6 | Direct\_id | int |  |  | True | FK |  |
| 7 | Realestate\_name | nvarchar | 50 |  | True |  |  |
| 8 | address | nvarchar | 100 |  | True |  |  |
| 9 | Room\_number | int |  |  | True |  |  |
| 10 | length | nvarchar | 50 |  | True |  |  |
| 11 | width | nvarchar | 50 |  | True |  |  |
| 12 | perimeter | nvarchar | 50 |  | True |  |  |
| 13 | area | nvarchar | 50 |  | True |  |  |
| 14 | legal\_status | nvarchar | 50 |  | True |  |  |
| 15 | price | nvarchar | 50 |  | True |  |  |
| 16 | discount | nvarchar | 10 |  | True |  |  |
| 17 | description | nvarchar | 500 |  | True |  |  |
| 18 | status | bit |  |  | True |  |  |

## 7. <Table RealEstate\_Image>

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | real\_estate\_id | int |  |  | True | FK |  |
| 3 | image\_name | nvarchar | 50 |  | False |  |  |
| 4 | image\_url | nvarchar | Max |  | True |  |  |
| 5 | status | bit |  |  | True |  |  |

## 8. <Table Direct >

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | direct\_name | nvarchar | 50 |  | True |  |  |
| 3 | status | bit |  |  | True |  |  |

## 9. <Table Payment >

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | payment\_method | nvarchar | 50 |  | True |  |  |
| 3 | status | bit |  |  | True |  |  |

## 10. <Table ReservationTime>

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Date | date |  |  | True | PK |  |
| 2 | Time1 | nvarchar | 50 |  | False |  |  |
| 3 | Time2 | nvarchar | 50 |  | False |  |  |
| 4 | Time3 | nvarchar | 50 |  | False |  |  |
| 5 | Time4 | nvarchar | 50 |  | False |  |  |
| 6 | status | bit |  |  | True |  |  |

## 11. <Table Location >

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | int |  |  | True | PK |  |
| 2 | ward | nvarchar | 50 |  | True |  |  |
| 3 | district | nvarchar | 50 |  | True |  |  |
| 4 | city | nvarchar | 50 |  | True |  |  |
| 5 | status | bit |  |  | True |  |  |