

**FLM - FPT EDUCAITON LEARNING MATERIALS**

**Software Design Specification**

– Hanoi, August 2022 –

**Record of changeS**

| **Date** | **A\* M, D** | **In charge** | **Change Description** |
| --- | --- | --- | --- |
| 28/05 | A | quannthe160206 | Thêm mô tả Overview (I) |
| 28/05 | A | quannthe160206 | Thêm code design User Profile (II.1-1.5) |
| 28/05 | A | quannthe160206 | Thêm code design Change Password (II.1-1.6) |
| 28/05 | A | trungnvhs160609 | Thêm code design User Login(II.1-1.3) |
| 28/05 | A | trungnvhs160609 | Thêm code design Reset Password(II.1-1.4) |
| 28/05 | A | tienlvhe161878 | Thêm code design View Setting List (II.2-2.3) |
| 28/05 | A | tienlvhe161878 | Thêm code design Edit Setting (II.2-2.4) |
| 28/05 | A | khoanthe161852 | Thêm code design [View list account](#_3rdcrjn)(II.2-2.1) |
| 28/05 | A | khoanthe161852 | Thêm code design [View account detail](#_3rdcrjn) (II.2-2.2) |
| 28/05 | A | quantnhe161199 | Thêm code design [Home page](#_3dy6vkm)(II.1-1.1) |
| 28/05 | A | quantnhe161199 | Thêm code design Register account(II.1-1.2) |
| 04/06 | A | trungnvhs160609 | Thêm code design Curriculum List(II.3-3.1) |
| 04/06 | A | trungnvhs160609 | Thêm code design New Curriculum(II.3-3.2) |
| 04/06 | A | trungnvhs160609 | Thêm code design View Curriculum(II.3-3.3) |
| 11/06 | A | quannthe160206 | Thêm code design Subject List(II.4-4.1) |
| 11/06 | A | quannthe160206 | Thêm code design New Subject(II.4-4.2) |
| 11/06 | A | quannthe160206 | Thêm code design Subject Import (II.4-4.3) |
| 11/06 | A | quannthe160206 | Thêm code design Subject Detail (II.4-4.4) |
| 11/06 | A | khoanthe161852 | Thêm code design Subject Predecessors(II.4-4.6) |
| 11/06 | A | khoanthe161852 | Thêm code design Subject Successors(II.4-4.7) |
| 11/06 | A | tienlvhe161878 | Thêm code design PLO List (II.5-5.1) |
| 11/06 | A | quantnhe161199 | Thêm code design Curriculum Edit (II.3-3.4) |
| 11/06 | A | quantnhe161199 | Thêm code design Curriculum Subject Detail (II.6-6.1) |
| 11/06 | A | quantnhe161199 | Thêm code design New Curriculum Subject (II.6-6.2) |
| 25/6 | A | khoanthe161852 | Thêm code design PO List (II.7-7.1) |
| 25/6 | A | khoanthe161852 | Thêm code design PO import (II.7-7.2) |
| 25/6 | A | khoanthe161852 | Thêm code design PO Detail(II.7-7.3) |
| 25/6 | A | khoanthe161852 | Thêm code design Subject Successors(II.4-4.6) |
| 25/6 | A | quannthe160206 | Thêm code design PLO import (II.5-5.3) |
| 25/6 | A | quannthe160206 | Thêm code design View Content Group List (II.8-8.3) |
| 25/6 | A | quannthe160206 | Thêm code design View Content Group Details (II.8-8.4) |
| 25/6 | A | trungnvhs160609 | Thêm mô tả chức năng List Decision |
| 25/6 | A | trungnvhs160609 | Thêm mô tả chức năng Add Decision |
| 25/6 | A | trungnvhs160609 | Thêm mô tả chức năng Detail Decision |
| 25/6 | A | trungnvhs160609 | Thêm mô tả chức năng Overview Decision |
| 25/6 | A | quantnhe161199 | Thêm mô tả chức năng View Combo List |
| 25/6 | A | quantnhe161199 | Thêm mô tả chức năng View Combo Detail |
| 25/6 | A | quantnhe161199 | Thêm mô tả chức năng View Elective List |
| 25/6 | A | quantnhe161199 | Thêm mô tả chức năng View Elective Details |
| 20/7 | A | quantnhe161199 | Thêm mô tả chức năng View Session List |
| 20/7 | A | quantnhe161199 | Thêm mô tả chức năng View Session Details |
| 20/7 | A | quantnhe161199 | Thêm mô tả chức năng Session Import |
| 20/7 | A | trungnvhs160609 | Thêm mô tả chức năng View Question List |
| 20/7 | A | trungnvhs160609 | Thêm mô tả chức năng View Question List |
| 20/7 | A | trungnvhs160609 | Thêm mô tả chức năng Question List |
| 20/7 | A | khoanthe161852 | Thêm mô tả chức năng View CLO List |
| 20/7 | A | khoanthe161852 | Thêm mô tả chức năng View CLO Detail |
| 20/7 | A | khoanthe161852 | Thêm mô tả chức năng CLO Import |
| 20/7 | A | khoanthe161852 | Thêm mô tả chức năng CLO-PLO Mapping |
| 20/7 | A | quannthe160206 | Thêm mô tả chức năng View Submit List |
| 20/7 | A | quannthe160206 | Thêm mô tả chức năng View Submit Details |
| 20/7 | A | quannthe160206 | Thêm mô tả chức năng Syllabus Import |
| 21/7 | A | tienlvhe161878 | Thêm mô tả chức năng Syllabus List |
| 21/7 | A | tienlvhe161878 | Thêm mô tả chức năng Syllabus Overview |
| 21/7 | A | tienlvhe161878 | Thêm mô tả chức năng Syllabus Edit |

\*A - Added M - Modified D - Deleted

**Table of Contents**

[I. Overview 8](#_gjdgxs)

[1. Code Packages 8](#_30j0zll)

[2. Database Design 9](#_1fob9te)

[a. Database Schema 9](#_3znysh7)

[b. Table Description 9](#_2et92p0)

[II. Code Designs 10](#_tyjcwt)

[1. User Access 10](#_hlx83r9iwp6)

[1.1 View home page 10](#_xu11kmqx8pkj)

[1.2 Register new account 12](#_lq1qrkowvnx1)

[1.3 Login to the system 13](#_gpw9o3jqr3m6)

[1.4 Reset password 14](#_r5wozmuo1czi)

[1.5 Update user profile 15](#_avo6id4cts11)

[1.6 Change password 17](#_cqx120206km3)

[2. System Admin 19](#_r0jz33gzzo3)

[2.1 View list account 19](#_xq0hfwczvkkr)

[2.2 View account details 21](#_gvk9eafvhtnl)

[2.3 View setting list 22](#_ujixea9pi7dq)

[2.4 Edit setting 23](#_fdtmsyypyvhh)

[3. Curriculum Management 24](#_udk79larb5kl)

[3.1 View Curriculum List 24](#_aea2lulfuqf1)

[3.2 Add New Curriculum 25](#_g6ayu9vmnt5h)

[3.3 View Curriculum Overview 26](#_e559rrteard9)

[3.4 Edit Curriculum 27](#_tgokknmmnbb2)

[4. Subject Management 28](#_qb06tzh49zfq)

[4.1 View Subject List 28](#_dm8kntstcaw)

[4.2 Add New Subject 30](#_krb5z8weu313)

[4.3 Import Subject 31](#_amlbhx3ycn2e)

[4.4 View Subject Detail 33](#_d3aq171e6v3)

[4.5 Mapping Subject-PLO 34](#_6t7ynezhnjqr)

[4.6 View Subject Predecessors 34](#_aug6h1m6gjme)

[4.7 View Subject Successors 36](#_tt2jpe6eypst)

[5. PLO Management 38](#_wlvgak19ewwe)

[5.1 View PLO List 38](#_uzzp71dfuqmb)

[5.2 View PLO Details 40](#_gwzosn7gejaj)

[5.3 Import PLO 40](#_okzbugslj4ik)

[5.4 Mapping PLO-PO 41](#_63kljly0wloi)

[6. Curriculum Subject 43](#_xzno6r1sol3h)

[6.1 View Curriculum Subject Details 43](#_msmneqiqrji4)

[6.2 Add New Curriculum Subject 45](#_s208mk9dj0ge)

[7. PO Management 47](#_wcxax1lw12ut)

[7.1 View PO List 47](#_aj0z7c53rtx9)

[7.2 Import PO 48](#_z5ccev8xr7bf)

[7.3 View PO Detail 50](#_bz3pv2e52v4x)

[8. Subject Groups 51](#_ug642ckthv2z)

[8.1 View Combo List 51](#_oele3i7nbc3b)

[8.2 View Combo Detail 53](#_s6ogly66gkq2)

[8.3 View Content Group List 54](#_mkwfeo42olnv)

[8.4 View Content Group Details 55](#_tc0f8p4dz8x5)

[8.5 View Elective List 56](#_fc76jrlynk60)

[8.6 View Elective Details 58](#_pipy4isabi6z)

[9. Syllabus Management 60](#_q8zpv3sac8j6)

[9.1 View Syllabus List 60](#_9wo8zfeg93yp)

[9.2 View Syllabus Overview 60](#_kb543cwbf6lc)

[9.3 Add New Syllabus 60](#_v6nu8vndrbu6)

[9.4 Edit Syllabus 60](#_qq5t59jltfc9)

[9.5 Import Syllabus 60](#_egh6xs3q4xdf)

[10. Decision Management 60](#_3qx9qod6jh5h)

[10.1 View Decision List 60](#_n5lzivuy5fqk)

[10.2 View Decision Details 62](#_15bsqcreazot)

[10.3 Add New Decision 63](#_8n6aieq3gkfv)

[10.4 View Decision Overview 65](#_wcmtdamih9l7)

[11. Syllabus Submit 66](#_jz7x1y93jvbv)

[11.1 View Submit List 66](#_y7j9e8ebp6wl)

[11.2 View Submit Detail 66](#_71jjujytlvik)

[12. Constructive Question 66](#_ftadtnt9h8ta)

[12.1 View Questions List 66](#_nlfrlq1gleky)

[12.2 View Questions Details 66](#_3e7fhdb006jq)

[12.3 Import Questions 66](#_85w1t93hj77s)

[13. Session Management 66](#_tk15rzun2qey)

[13.1 View Session List 66](#_vc7oubp265h4)

[13.2 View Session Details 66](#_82gnriz9yvg)

[13.3 Import Session 66](#_yhy2jb3v9ken)

[14. CLO Management 66](#_23w0tg9rppfi)

[14.1 View CLO List 66](#_3jsn0wnb27eg)

[14.2 View CLO Details 67](#_fqvgl12zdtya)

[14.3 Import CLO 67](#_iakv7497rej)

[14.4 Mapping CLO-PLO 67](#_bz5phmh1kufe)

[15. Assessment Management 67](#_nkoakbdzm9ax)

[15.1 View Assessment List 67](#_68nct2ivtmqd)

[15.2 View Assessment Details 67](#_ruiqpatufvcl)

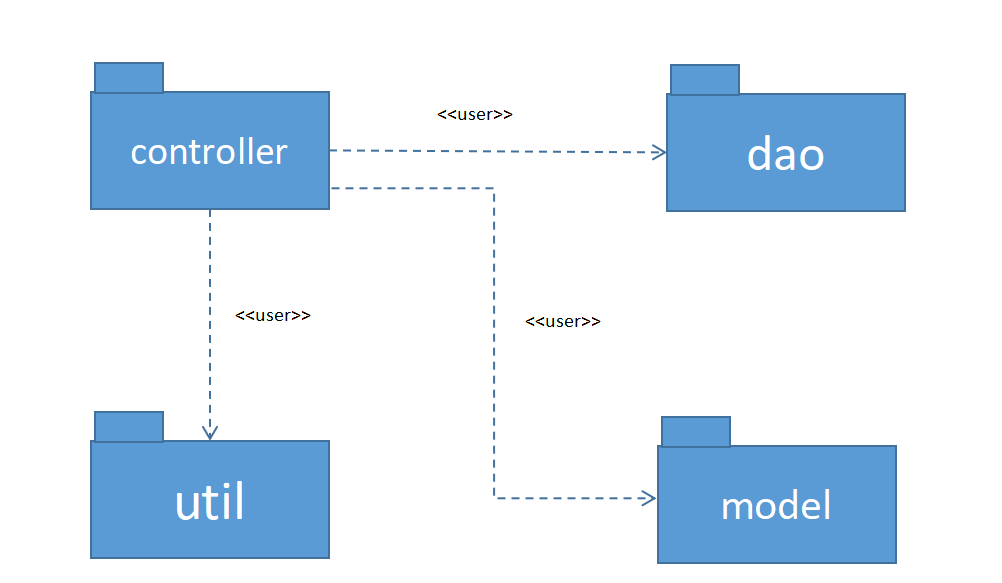
[16. Reference Material 67](#_yvue1f6ddonh)

[16.1 View Material List 67](#_xlzsk3bpeozn)

[16.2 View Material Detail 67](#_l2jp07atjg91)

# I. Overview

## 1. Code Packages

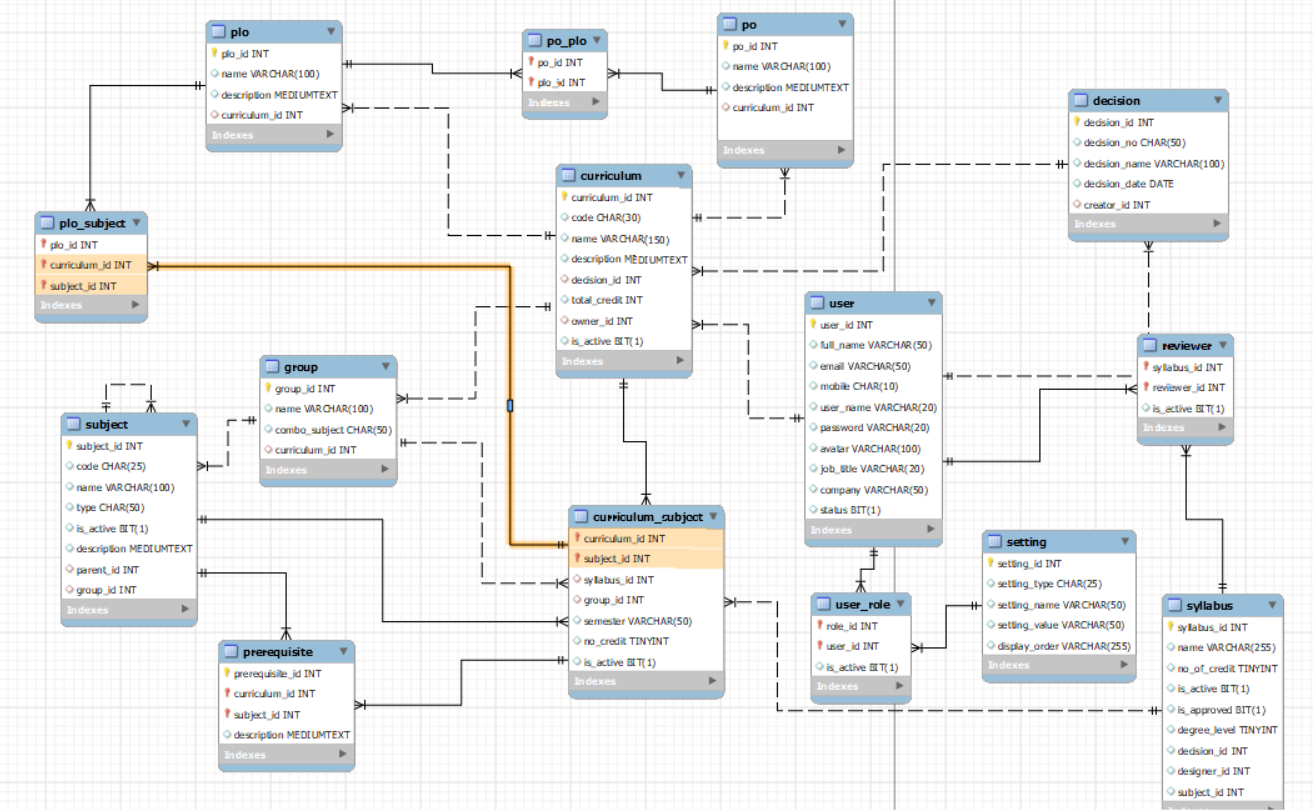


***Package descriptions***

| **No** | **Package** | **Description** |
| --- | --- | --- |
| *01* | *Controller* | *The controller package and its subpackage*  *controller.authentication house the classes responsible for*  *handling user interactions, processing requests, and*  *managing authentication-related functionality. These*  *classes act as the bridge between the UI and the underlying*  *application logic, ensuring smooth communication and*  *facilitating the execution of user-specific actions and*  *operations.* |
| *02* | *dao* | *The DAO package serves as a crucial component in the*  *data layer of the application. It abstracts the interaction*  *with the database, providing methods and interfaces for*  *accessing, manipulating, and persisting data entities. The*  *DAO classes ensure the separation of concerns and promote*  *code reusability by centralizing database-related operations*  *in a dedicated package.* |
| *03* | *model* | *The model package serves as a container for the*  *application's data models or entities, defining their*  *attributes, relationships, and behaviors. These classes*  *encapsulate the essential data structures and functionality*  *necessary to represent and manipulate core entities within the application. Separating the data models into a dedicated*  *package promotes organization, modularity, and*  *maintainability of the application's data layer.* |
| *04* | *util* | *The util package serves as a repository for reusable utility*  *classes and constants, simplifying development and*  *promoting consistency across the codebase. It enhances*  *code organization, encapsulation, and maintainability by*  *separating commonly used functionalities and values from*  *the rest of the application's logic.* |

## 2. Database Design

### a. Database Schema



### b. Table Description

| **No** | **Table** | **Description** |
| --- | --- | --- |
| *01* | *User* | The user table contains all information about the account's full name, user name, mail, mobile, password, job, and company. |
| *02* | *Setting* | The ***settings*** table is to store master data:  • user roles  • system pages  • training types  • assessment types  • assessment categories |
| *03* | User\_Role | Contains user id and role id to store each person's role. A person can have many roles but only 1 role can be active at a time |
| *04* | Subject |  |
| *05* | Curriculum |  |
| *06* | Curriculum\_subject |  |
| *07* | Group |  |
| *08* | plo |  |
| *09* | plo\_subject |  |
| *10* | prerequisite |  |
| *11* | po |  |
| *12* | po\_plo |  |
| *13* | Decision |  |
| *14* | Reviewer |  |
| *15* | Syllabus |  |

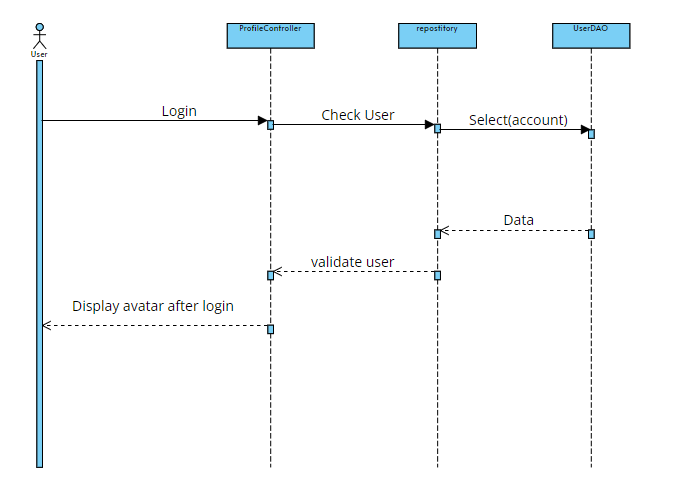
# II. Code Designs

## User Access

### 1.1 View home page

#### a. Class Diagram

#### b. Sequence Diagram(s)



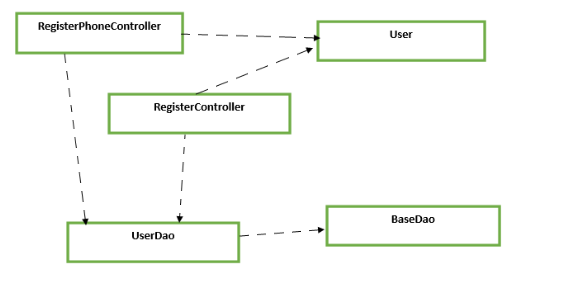
#### c. Database Queries

1.select \* from user where email =?

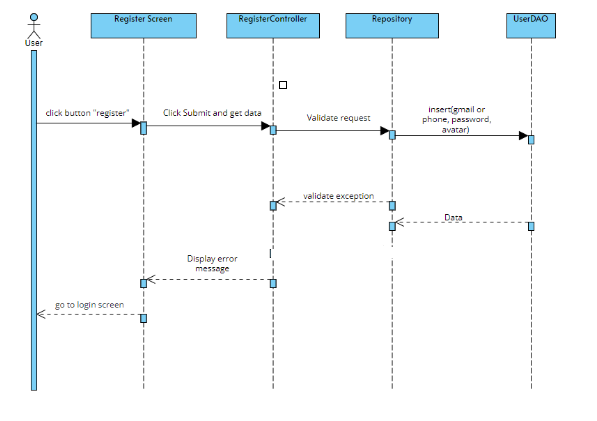
2.select \* from user where mobile =?

### 1.2 Register new account

#### a. Class Diagram



#### b. Sequence Diagram(s)



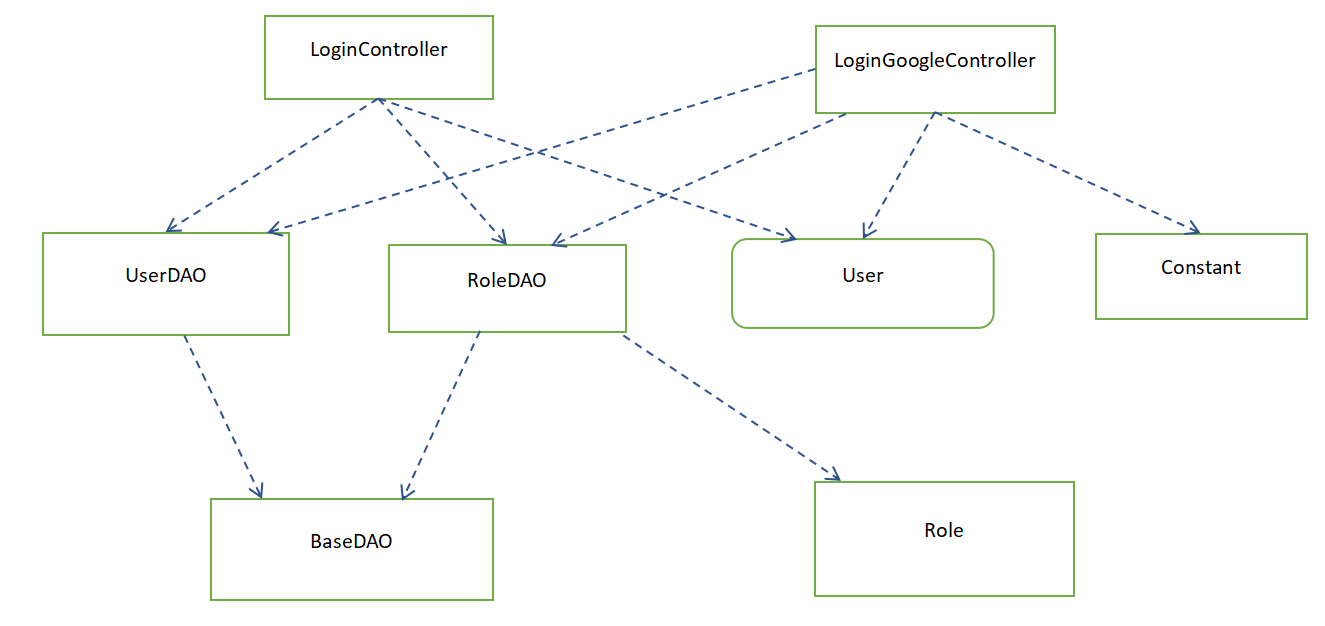
#### c. Database Queries

1:INSERT INTO `swp391\_g3\_flm`.`user` ( `mobile`, `password`, `avatar`) VALUES (?, ?,?)

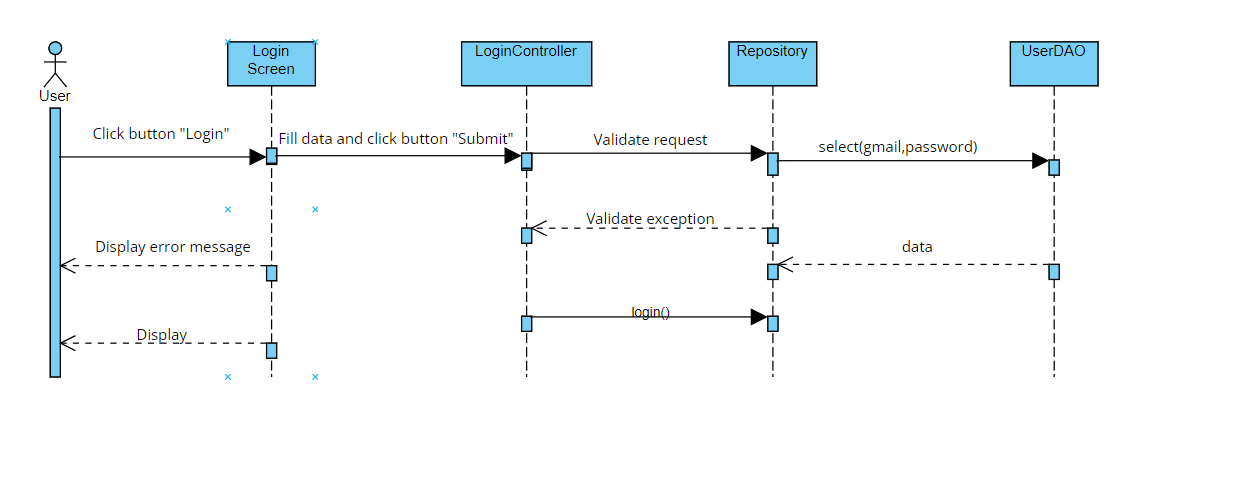
2:INSERT INTO `swp391\_g3\_flm`.`user` ( `email`, `password`, `avatar`) VALUES (?, ?,?)

### 1.3 Login to the system

#### a. Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

1.select \* from user where email =? and password =?

2.select \* from user where email=?

3.select \* from user\_role where user\_id = ?

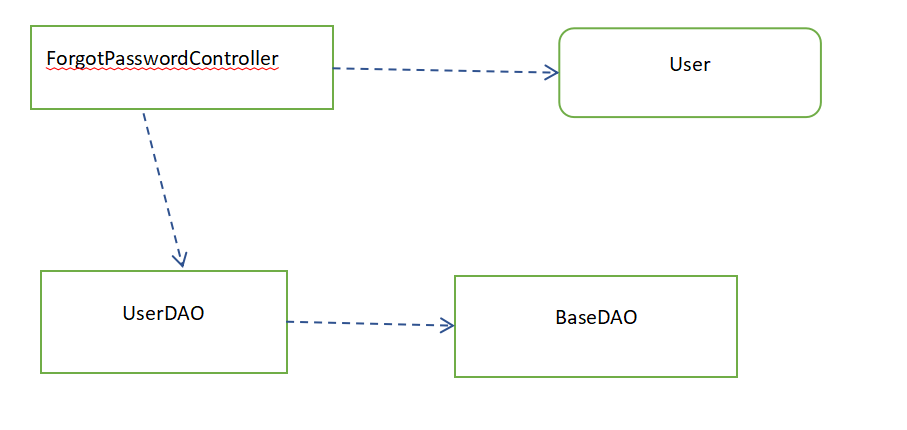
4.select \* from setting where setting\_id = ?

5.select \* from user where full\_name=?

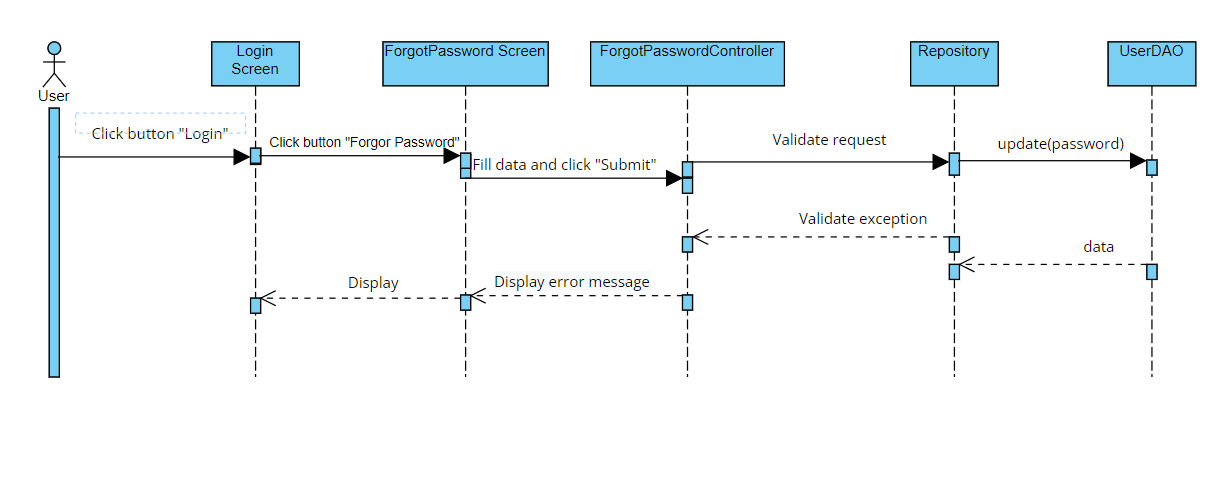
6.select \* from user where mobile=?

### 1.4 Reset password

#### a. Class Diagram

****

#### b. Sequence Diagram(s)



#### c. Database Queries

1.select \* from user where email =?

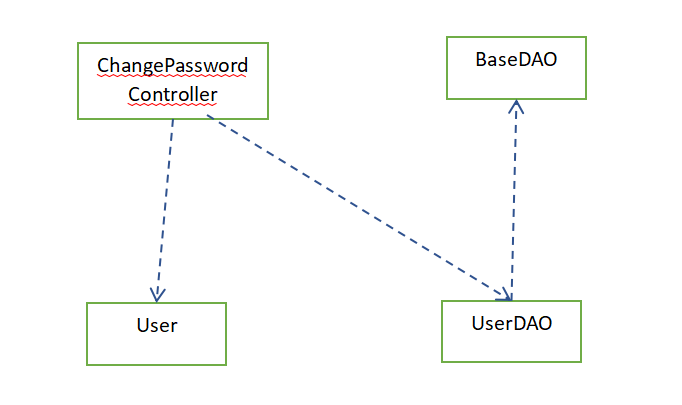
2.select \* from user where mobile =?

3.update user SET password =? WHERE user\_id = ?

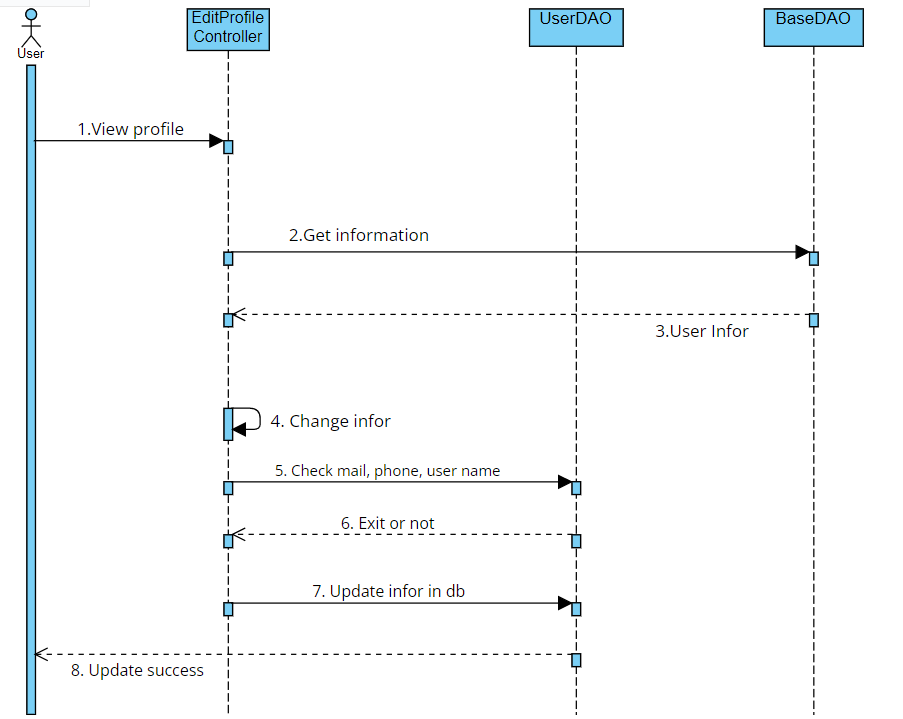
4.insert user\_role(role\_id,user\_id) values ('1',?)

### 1.5 Update user profile

#### a. Class Diagram



#### b. Sequence Diagram(s)

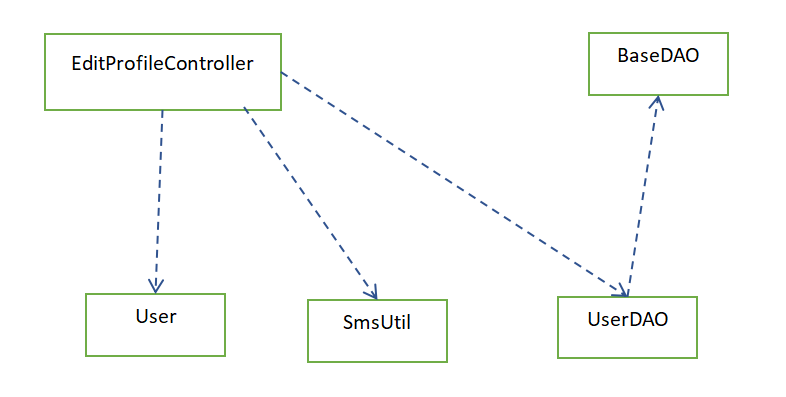


#### c. Database Queries

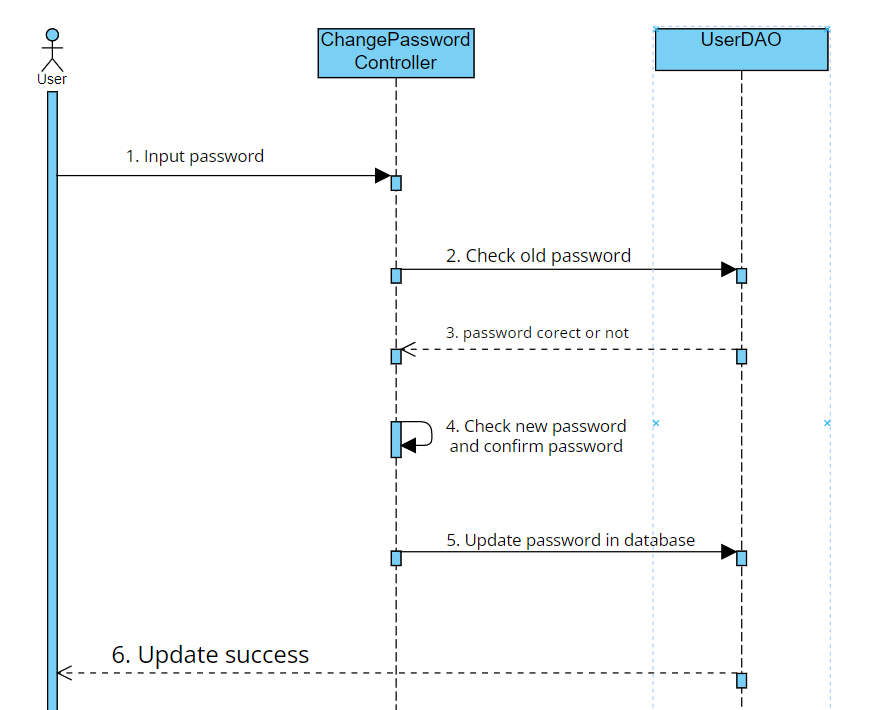
1. UPDATE user SET password = ? WHERE user\_id=?;

### 1.6 Change password

#### a. Class Diagram



#### b. Sequence Diagram(s)



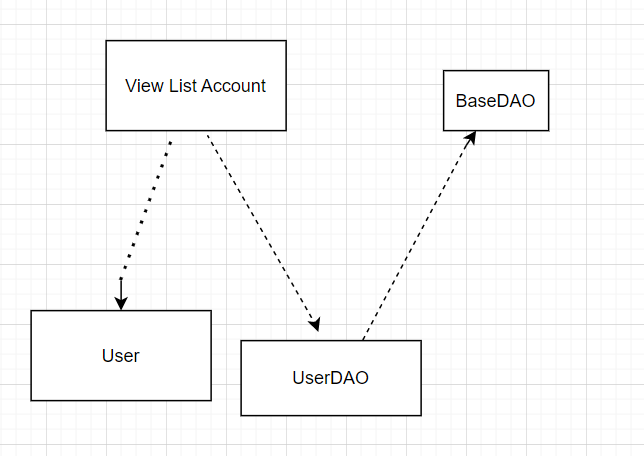
#### c. Database Queries

1. select \* from user where mobile =?
2. select \* from user where email =?
3. UPDATE user SET full\_name = ?, email = ?, mobile = ?, user\_name = ? job\_title = ?, company = ? WHERE (user\_id = ?);
4. UPDATE user SET avatar = ? WHERE (user\_id = ?);

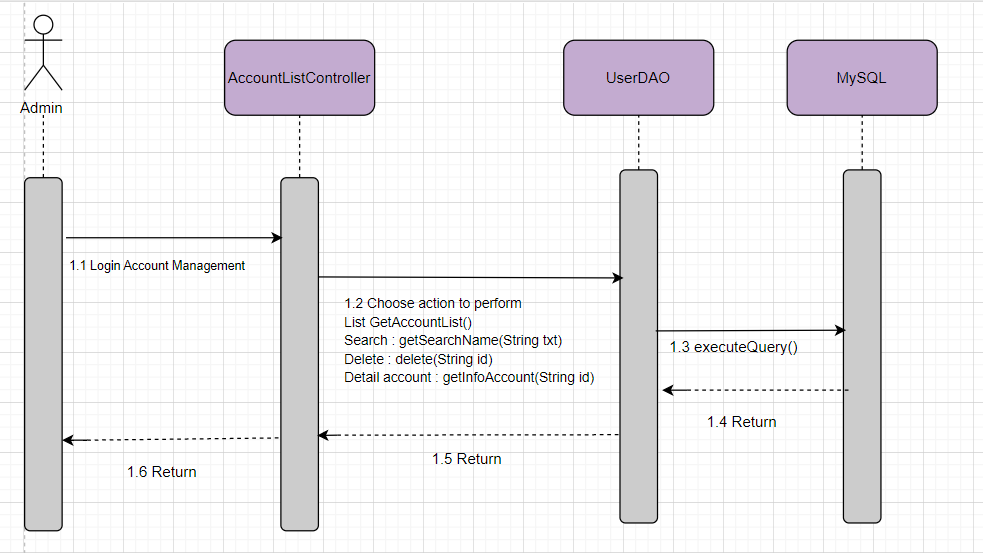
## System Admin

### 2.1 View list account

#### a. Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

1. select \* from user
2. "select \* from user"

+ " where full\_name like '" + "%" + txt + "%" + "'"

+ " OR email like '" + "%" + txt + "%" + "'"

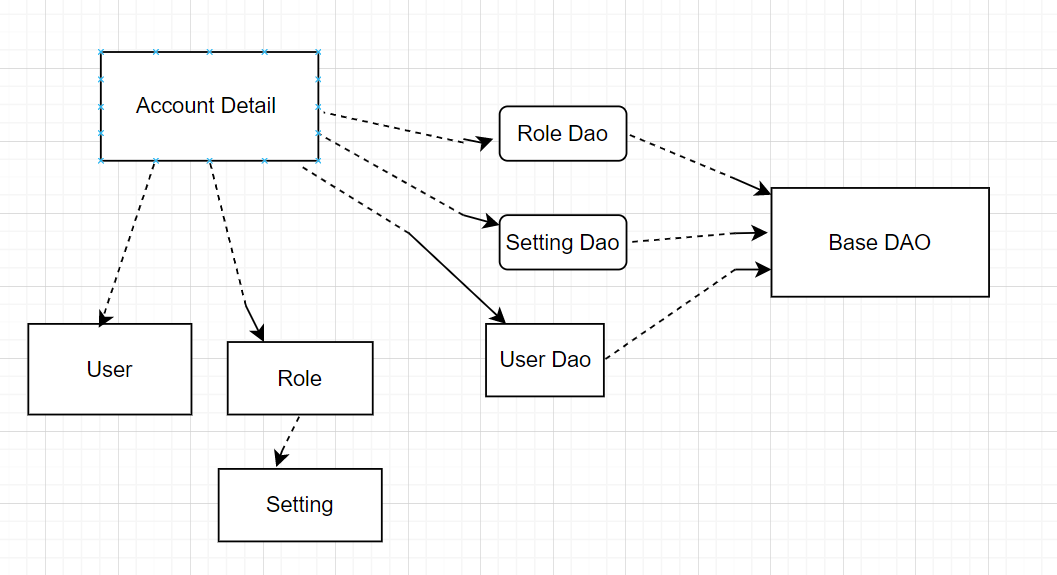
+ " OR mobile like '" + "%" + txt + "%" + "'"

+ " OR job\_title like '" + "%" + txt + "%" + "'";

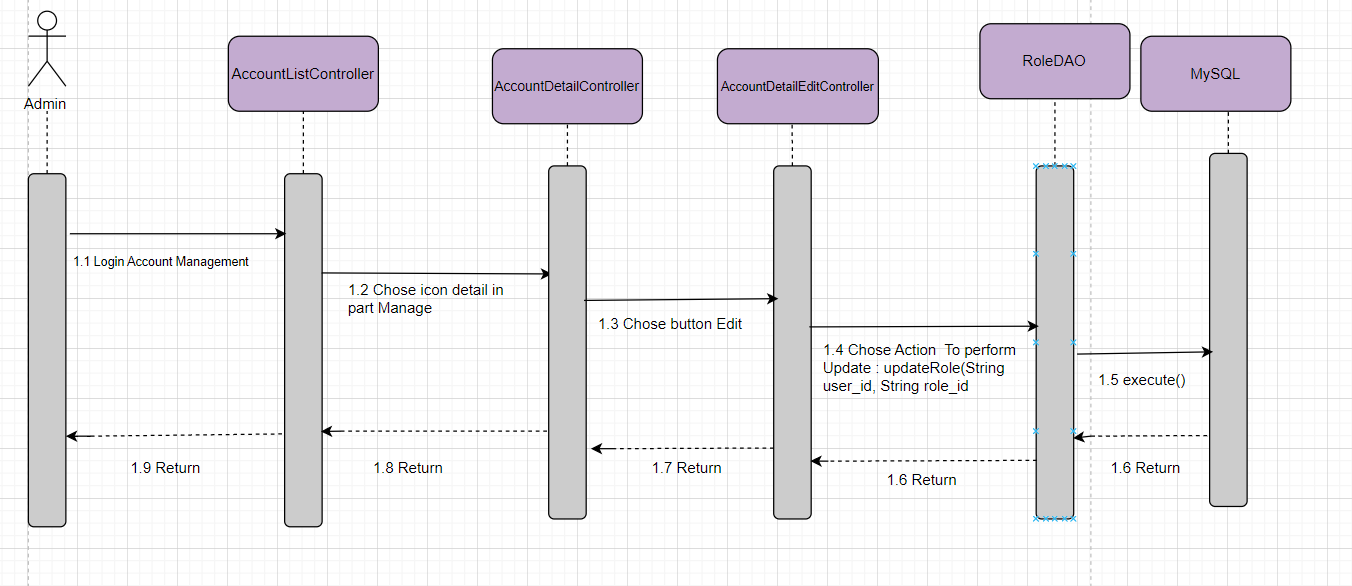
1. delete from user where user\_id=?
2. select \* from user in order by user\_id asc
3. select \* from user in order by user\_id desc
4. select \* from user in order by full\_name asc
5. select \* from user in order by full\_name desc
6. select \* from user in order by email asc
7. select \* from user in order by email desc
8. select \* from user in order by mobile asc
9. select \* from user in order by mobile desc
10. select \* from user in where status = 0
11. select \* from user in where status = 1

### 2.2 View account details

#### a. Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

1. select \* from setting
2. UPDATE user\_role "

+ "SET role\_id = ? "

+ "WHERE (user\_id = ?);"

1. select user.user\_id , full\_name, email, mobile, user\_name, avatar, job\_title, company,setting\_name\n"

+ "from user join user\_role\n"

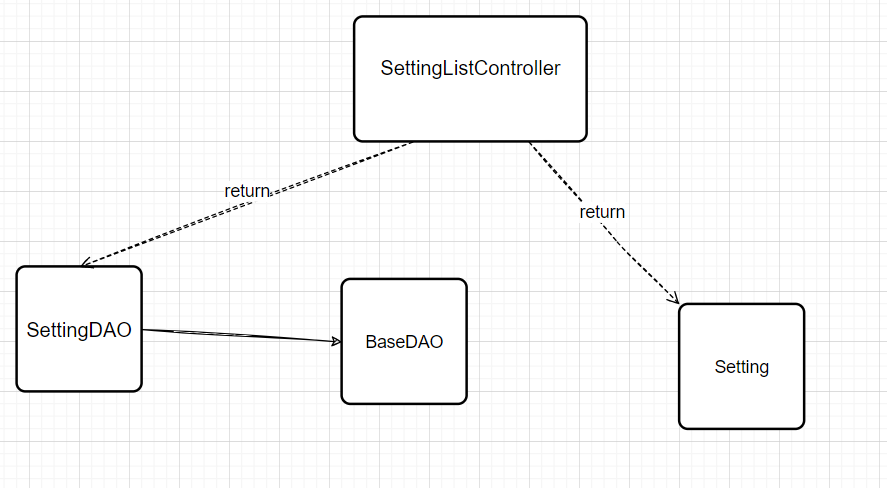
+ "on user\_role.user\_id = user.user\_id \n"

+ "join setting on user\_role.role\_id = setting.setting\_id "

+ " where user.user\_id ='" + id + "' ";

### 2.3 View setting list

#### a. Class Diagram



#### b. Sequence Diagram(s)

#### c. Database Queries

1 SELECT \* FROM setting;

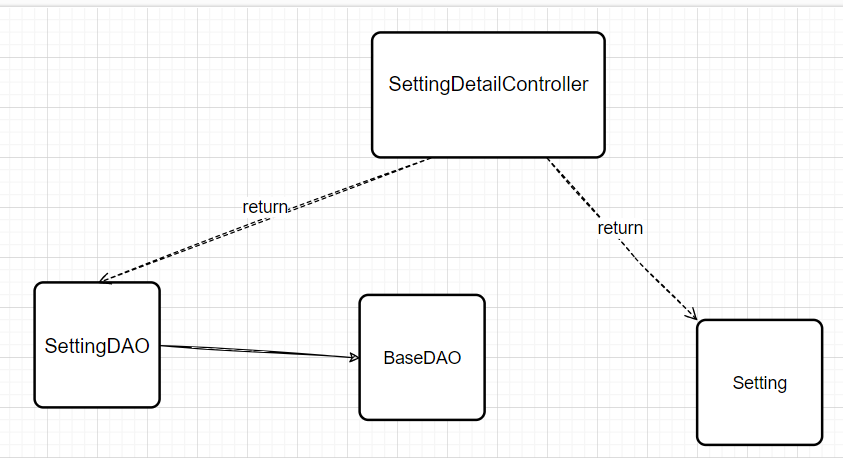
2 SELECT \* FROM setting where setting\_name like ?;

3 SELECT DISTINCT setting\_type FROM setting;

4 SELECT \* FROM setting where setting\_type like ?;

### 2.4 Edit setting

#### a. Class Diagram



#### b. Sequence Diagram(s)

#### c. Database Queries

1 SELECT \* FROM setting where setting\_id like ?;

2 UPDATE swp391\_g3\_flm.setting "

SET setting\_name = ?,

setting\_value = ?,

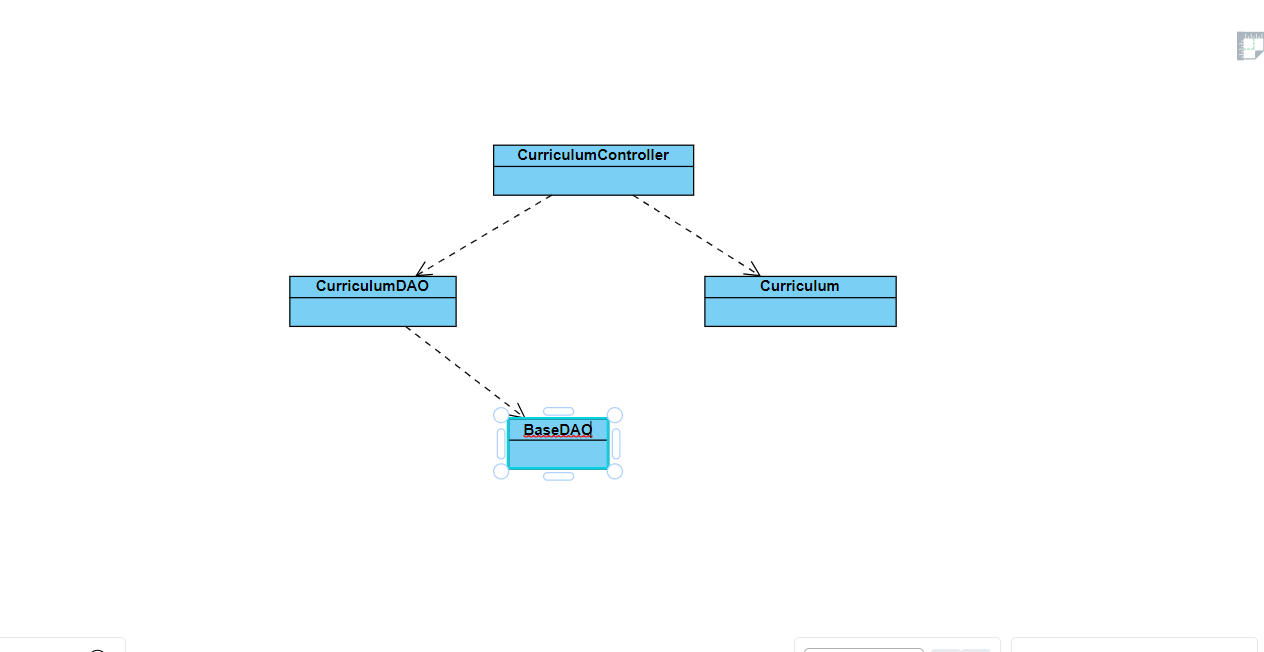
display\_order = ?

+ WHERE setting\_id = ?;

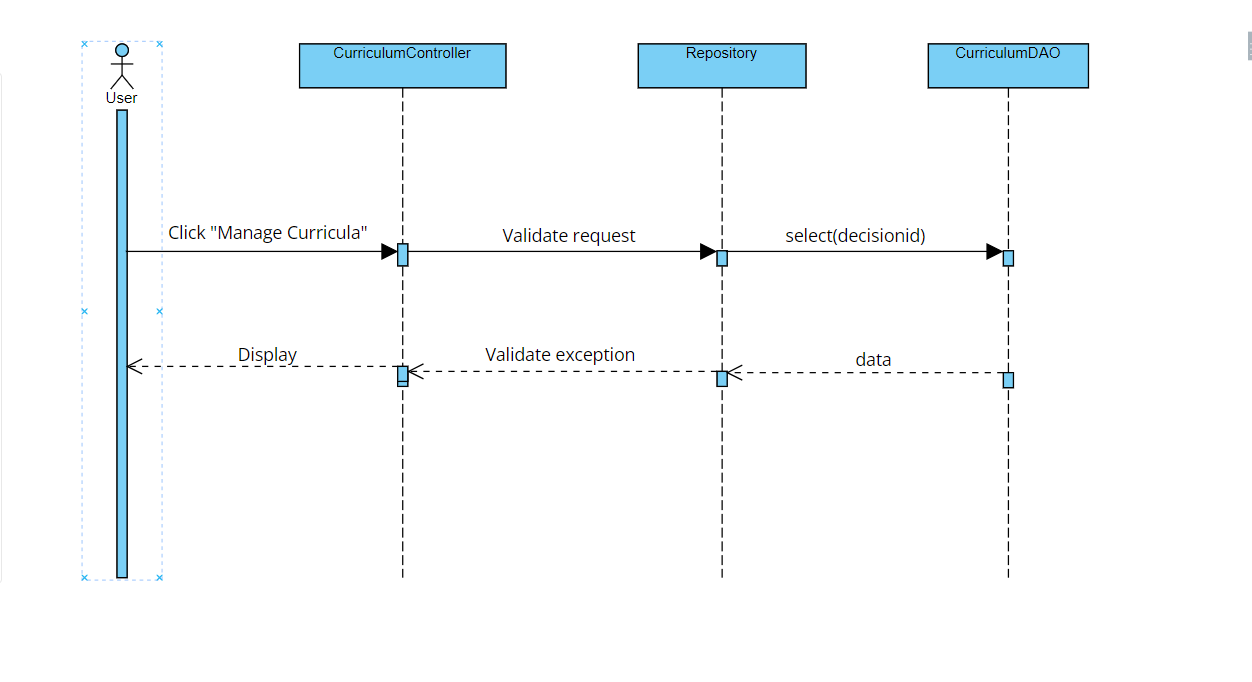
## Curriculum Management

### 3.1 View Curriculum List

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

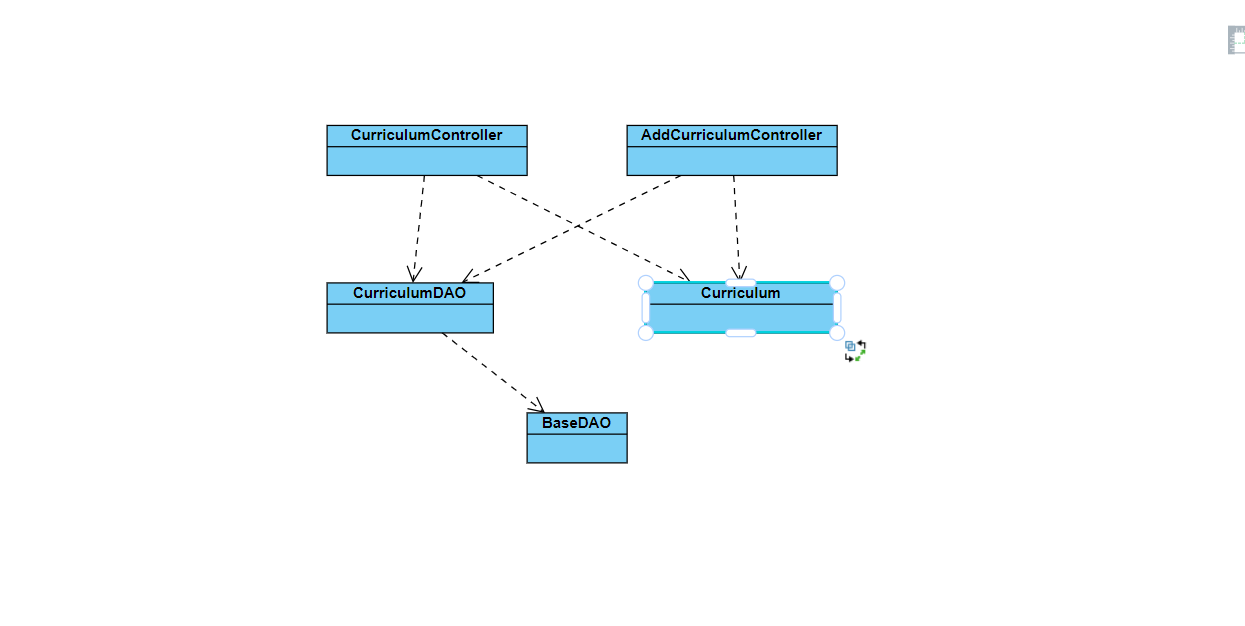
******

#### c. Database Queries

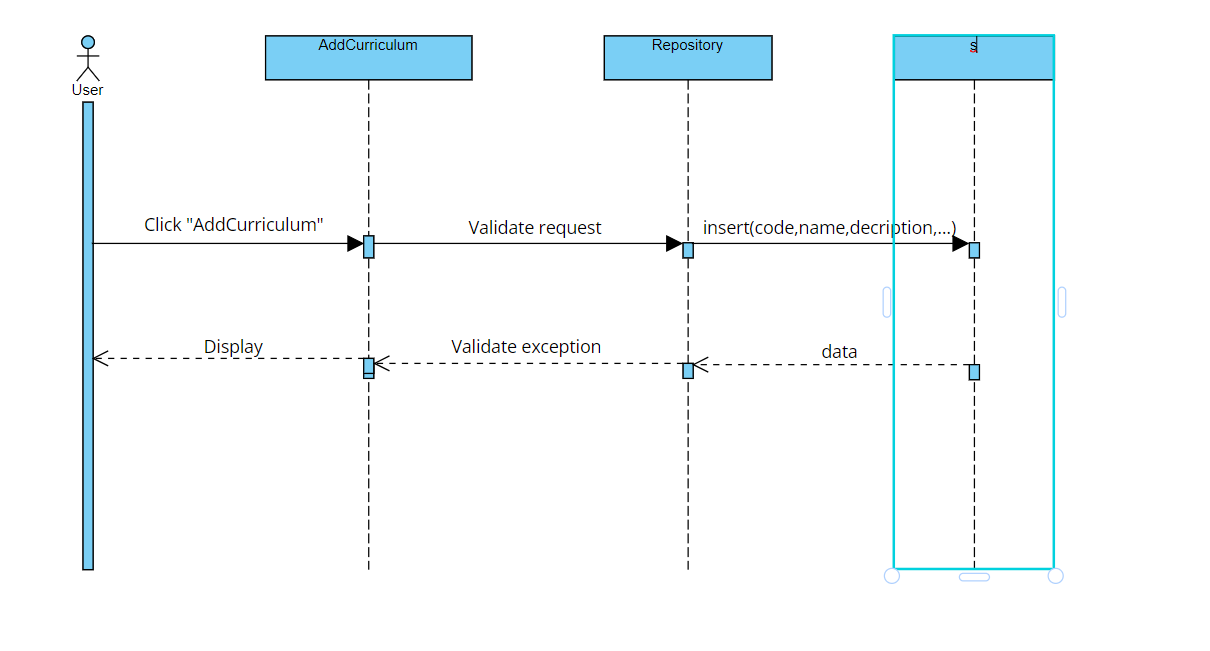
SELECT \* FROM curriculum where code like ?;

### 3.2 Add New Curriculum

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

#### c. Database Queries

select \* from curriculum where code =?

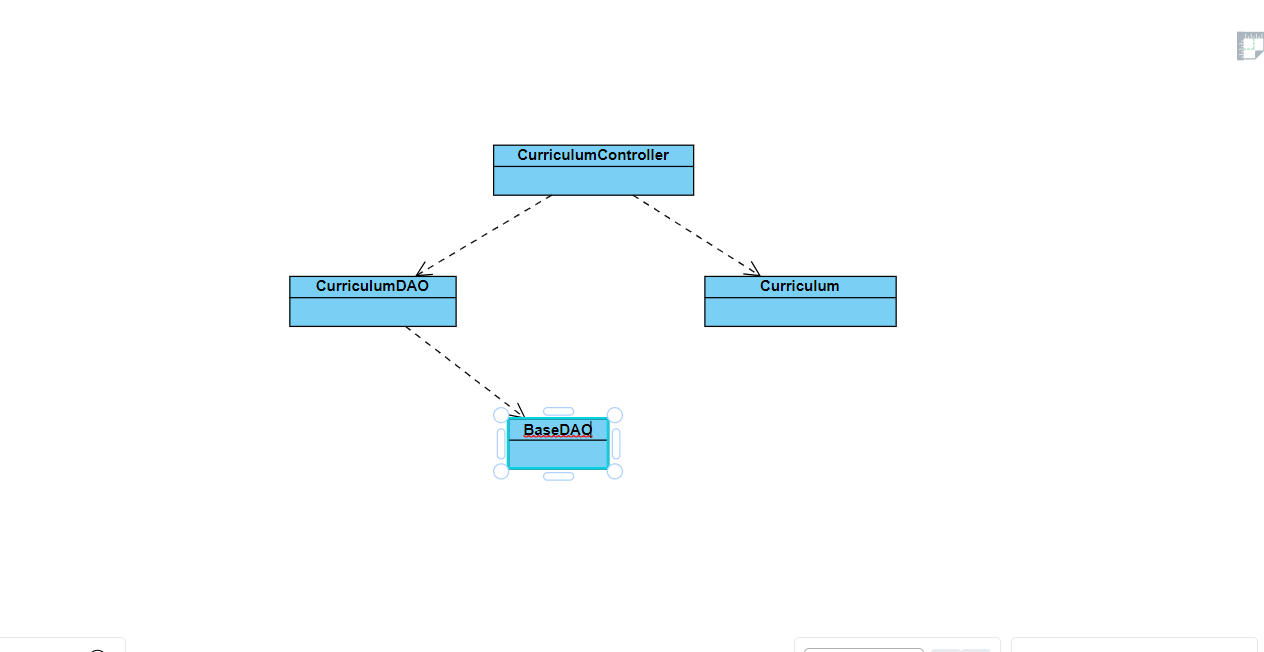
INSERT INTO `swp391\_g3\_flm`.`curriculum` (`code`, `name`, `description`, `decision\_id`, `total\_credit`, `owner\_id`)"

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

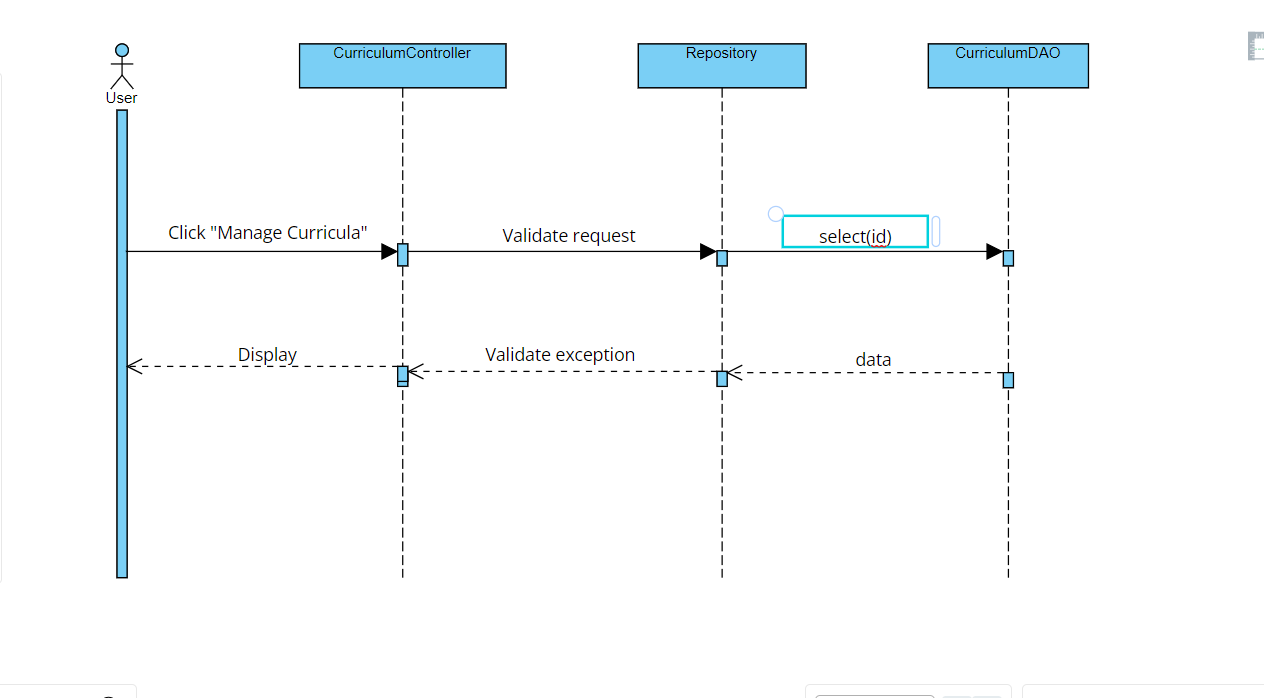
select \* from curriculum

### 3.3 View Curriculum Overview

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

#### c. Database Queries

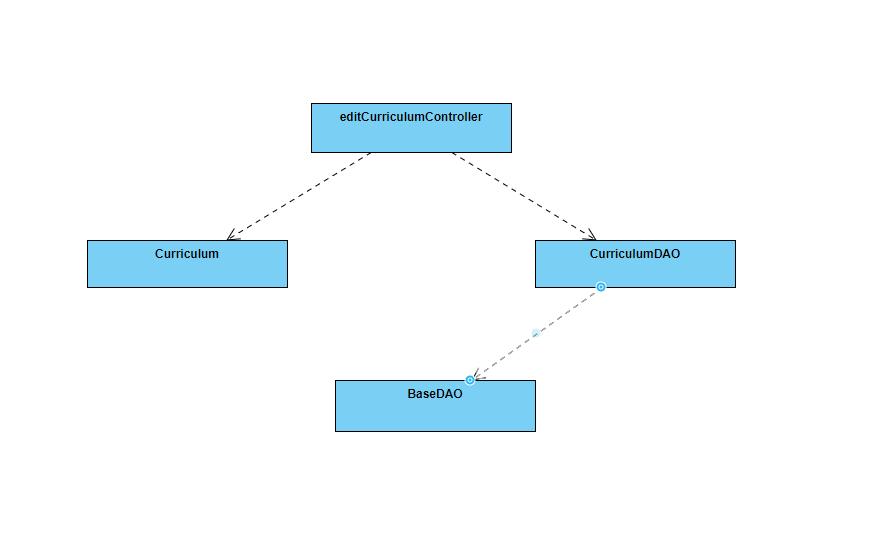
SELECT \* FROM curriculum where curriculum\_id = ?;

select \* from decision where decision\_id = ?

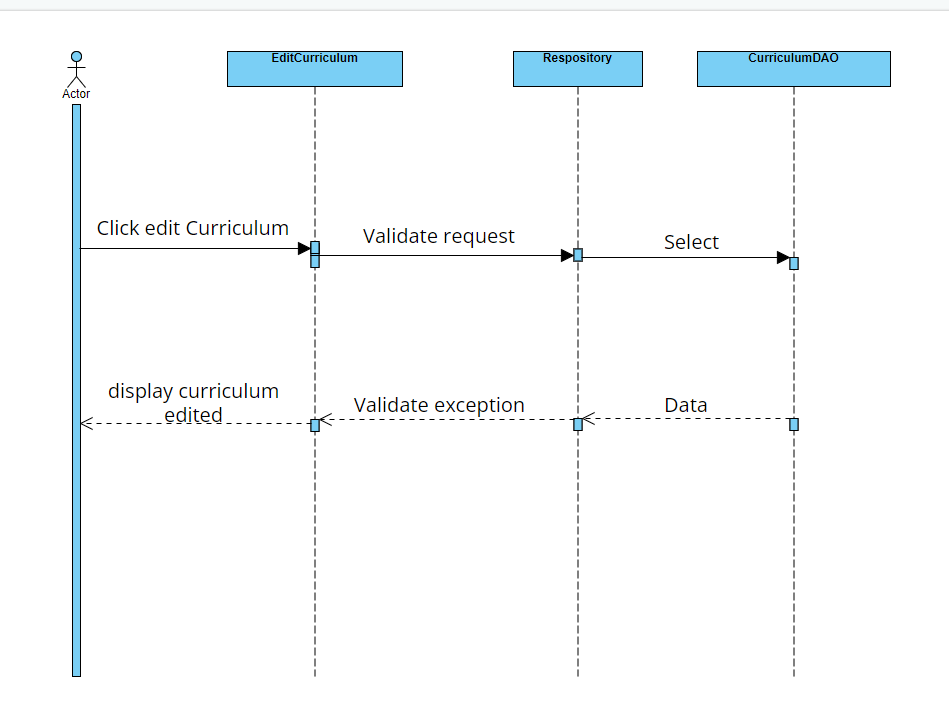
return decision\_id;

### 3.4 Edit Curriculum

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

#### c. Database Queries

UPDATE `swp391\_g3\_flm`.`curriculum`"

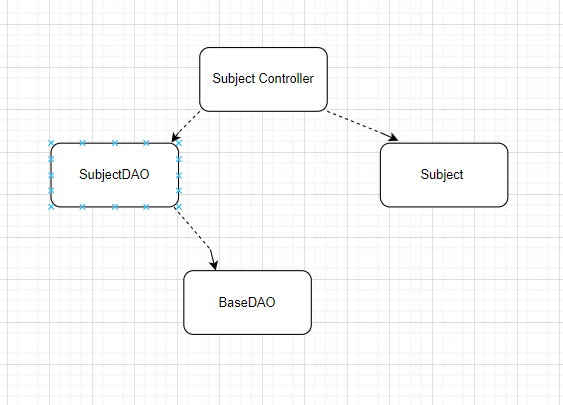
+”SET code = ? ,name = ? , description = ? , decision\_id = ? , total\_credit = ? , owner\_id = ?"

+ "Where curriculum\_id = ?"

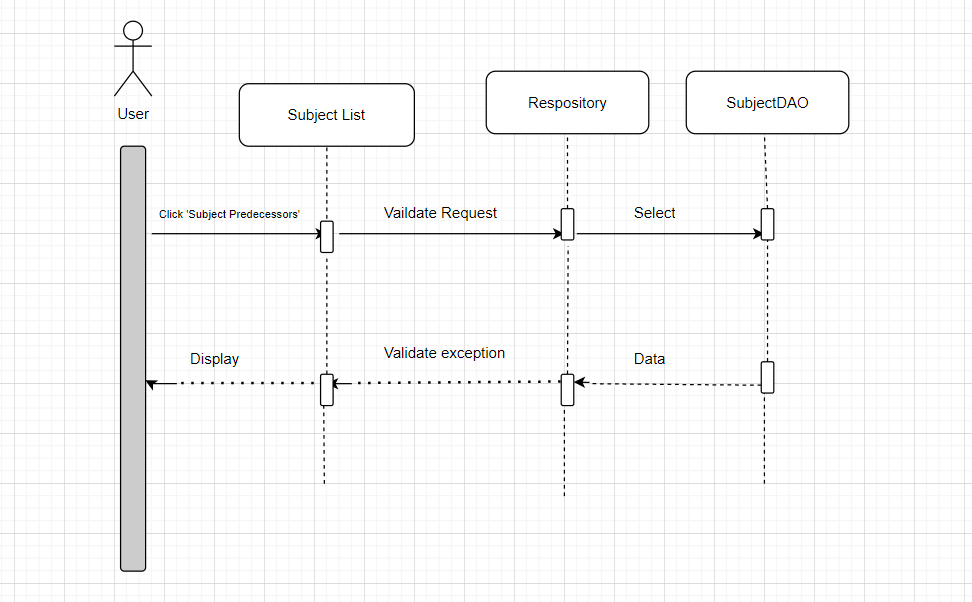
## Subject Management

### 4.1 View Subject List

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

#### c. Database Queries

String strSelect = "SELECT \* FROM swp391\_g3\_flm.subject\n" +

"order by subject\_id\n" +

"limit ?, 10;";

String strSelect = "select \* from subject where code = ?;";

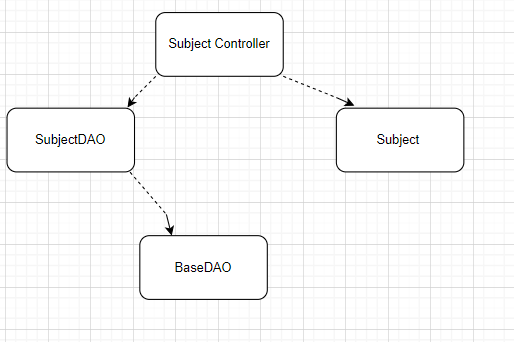
String strSelect = "select \* from subject where code like ?;";

String strSelect = "select \* from subject where name like ?;";

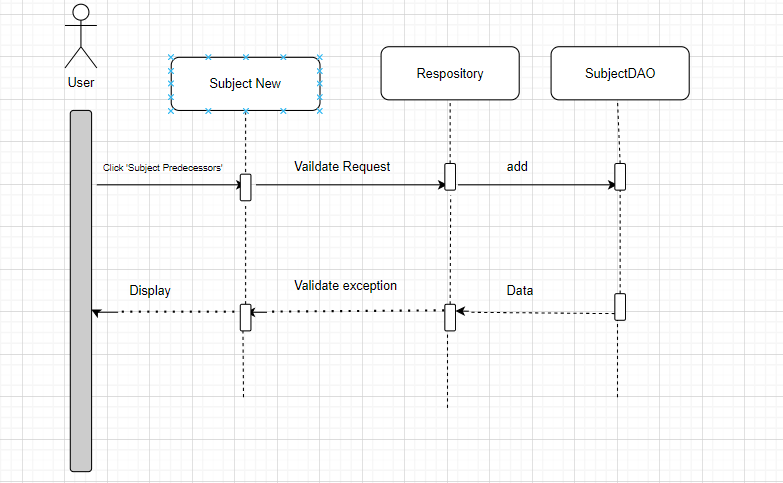
"select count(\*) from subject";

### 4.2 Add New Subject

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

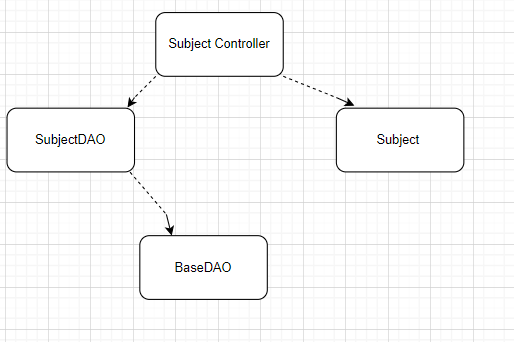
#### c. Database Queries

String strAdd = "INSERT INTO `swp391\_g3\_flm`.`subject` (`code`, `name`,`type`, `is\_active`, `description`, `parent\_id`)"

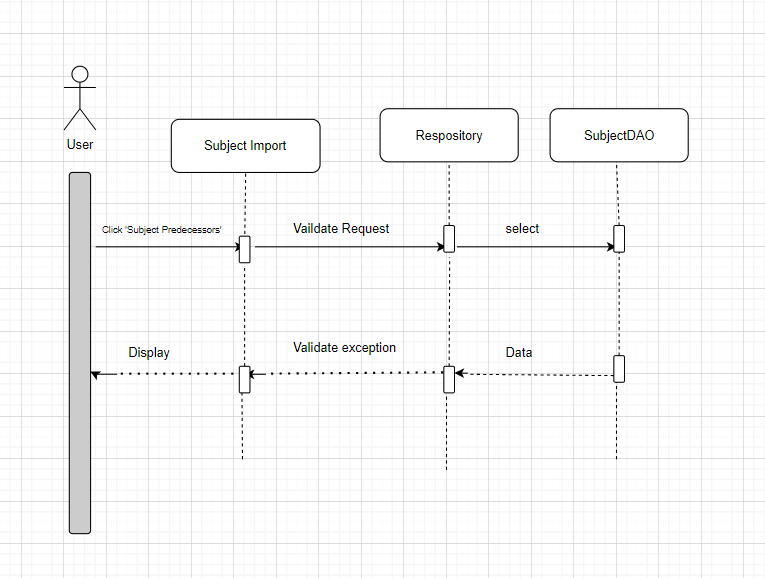
+ " VALUES (?, ?, ?, b?, ?, ?);";

### 4.3 Import Subject

#### a.Class Diagram

****

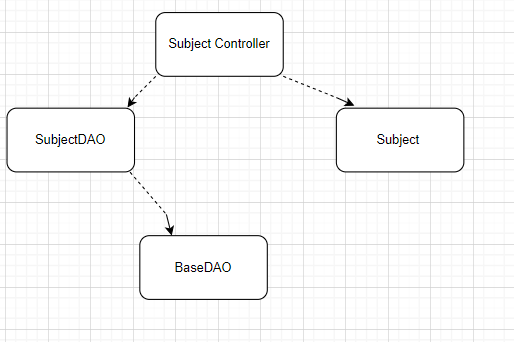
#### b. Sequence Diagram(s)

******

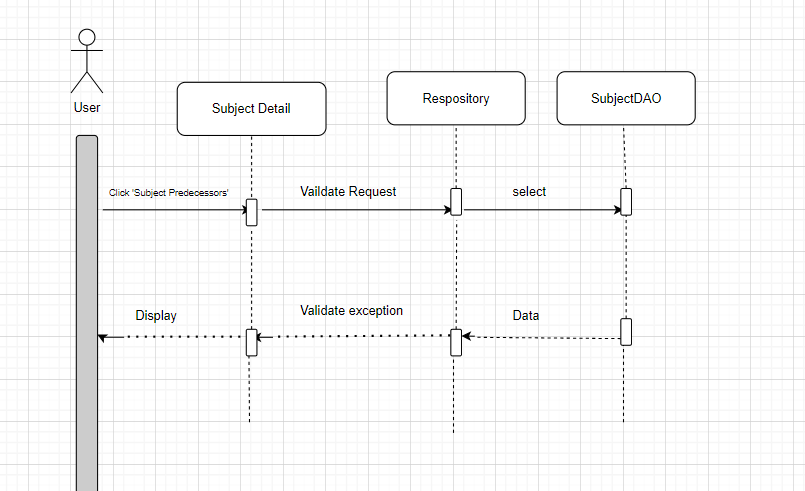
#### c. Database Queries

### 4.4 View Subject Detail

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

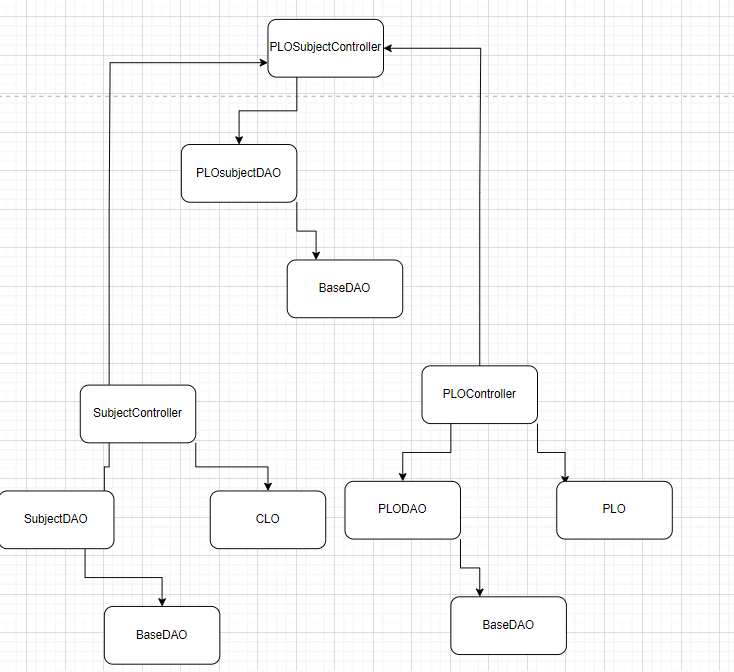
******

#### c. Database Queries

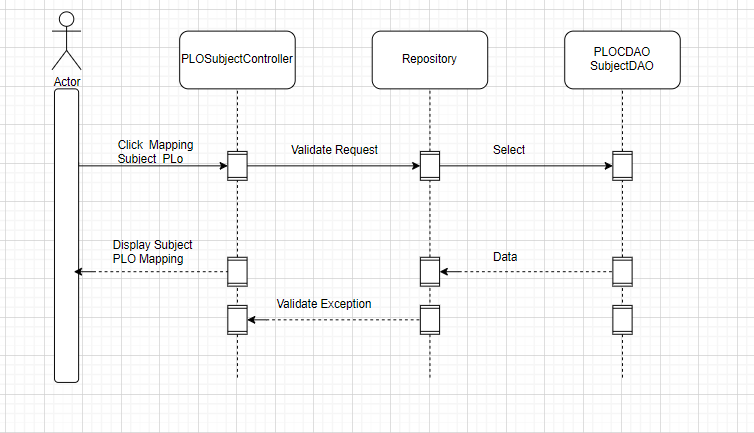
String strSelect = "select \* from subject where code = ?;";

### 4.5 Mapping Subject-PLO

#### a.Class Diagram



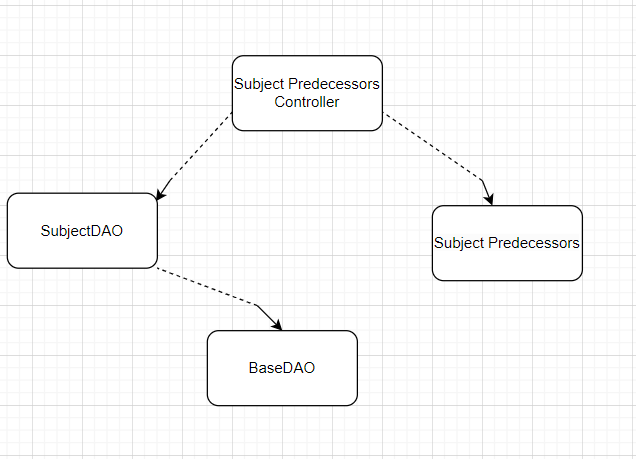
#### b. Sequence Diagram(s)



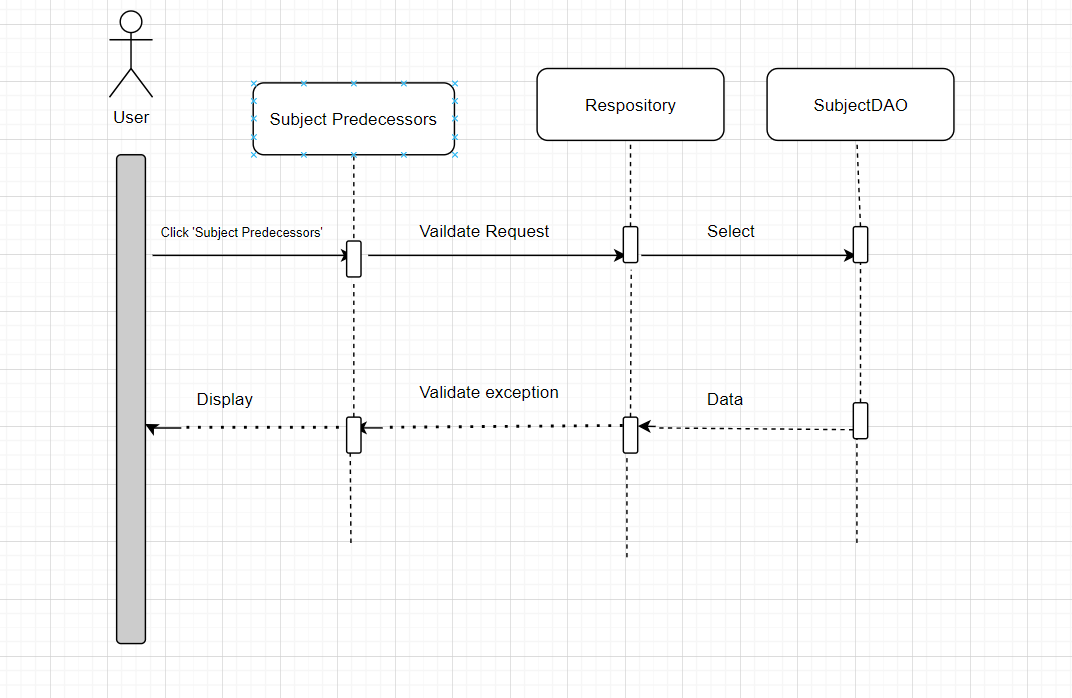
#### c. Database Queries

### 4.6 View Subject Predecessors

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

#### c. Database Queries

select d.subject\_id, d.code, d.name, d.type, d.is\_active, d.description, d.parent\_id, d.group\_id

from subject s join subject d

on s.subject\_id = d.parent\_id

order by d.subject\_id

Limit ?,10

select DISTINCT d.subject\_id, d.code, d.name, d.type, d.is\_active, d.description, d.parent\_id, d.group\_id

from subject s join subject d

on s.subject\_id = d.parent\_id

WHERE d.name like ?

select DISTINCT d.subject\_id, d.code, d.name, d.type, d.is\_active, d.description, d.parent\_id, d.group\_id

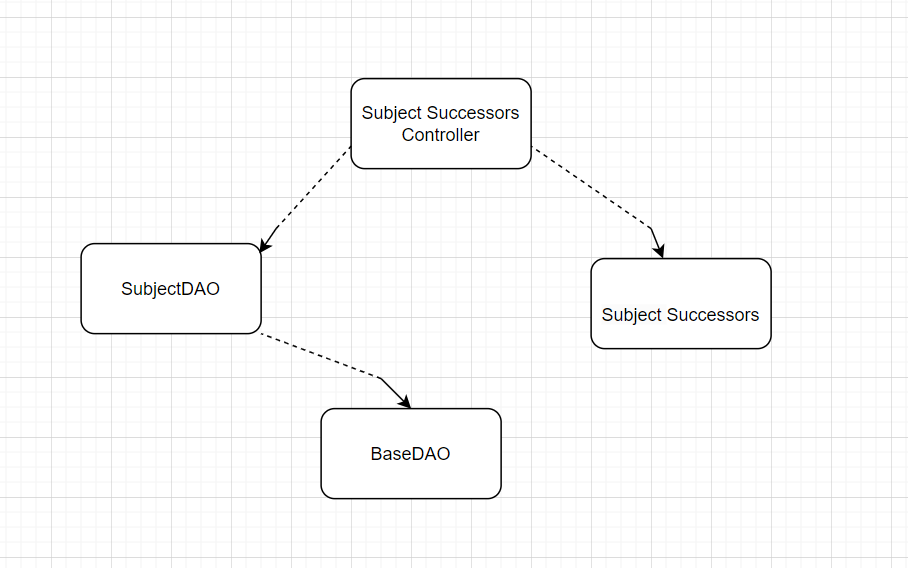
from subject s join subject d

on s.subject\_id = d.parent\_id

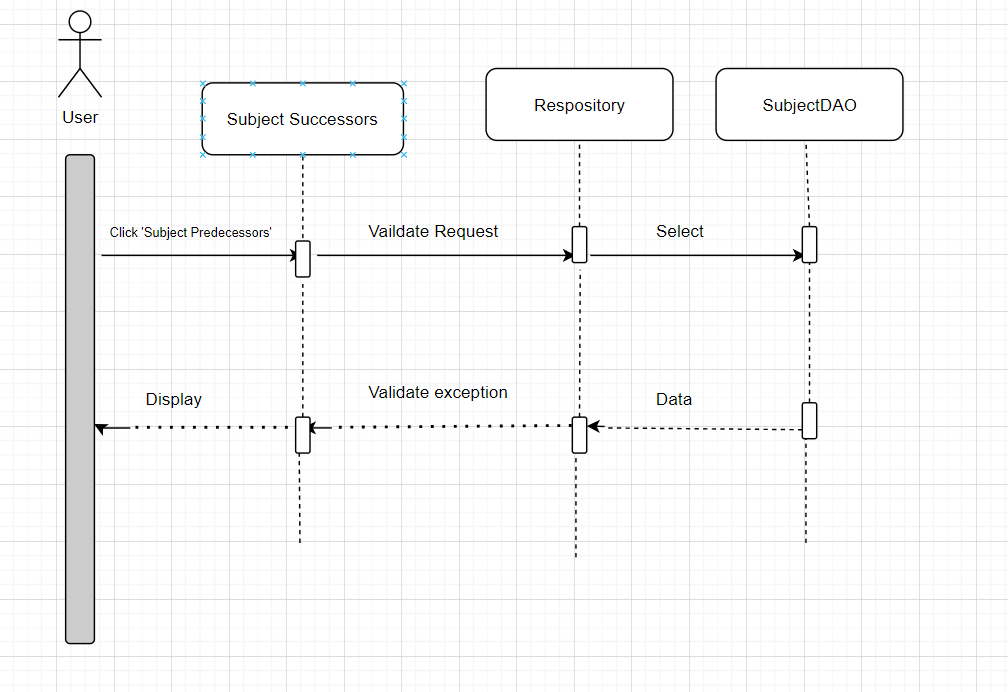
WHERE d.code like ?

### 4.7 View Subject Successors

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

#### c. Database Queries

select DISTINCT s.subject\_id, s.code, s.name, s.type, s.is\_active, s.description, s.parent\_id, s.group\_id

from subject s join subject d

on s.subject\_id = d.parent\_id

WHERE s.parent\_id is null

order by s.subject\_id

Limit ?,10;

select DISTINCT s.subject\_id, s.code, s.name, s.type, s.is\_active, s.description, s.parent\_id, s.group\_id

from subject s join subject d

on s.subject\_id = d.parent\_id

WHERE s.parent\_id is null and s.name like ?

select DISTINCT s.subject\_id, s.code, s.name, s.type, s.is\_active, s.description, s.parent\_id, s.group\_id

from subject s join subject d

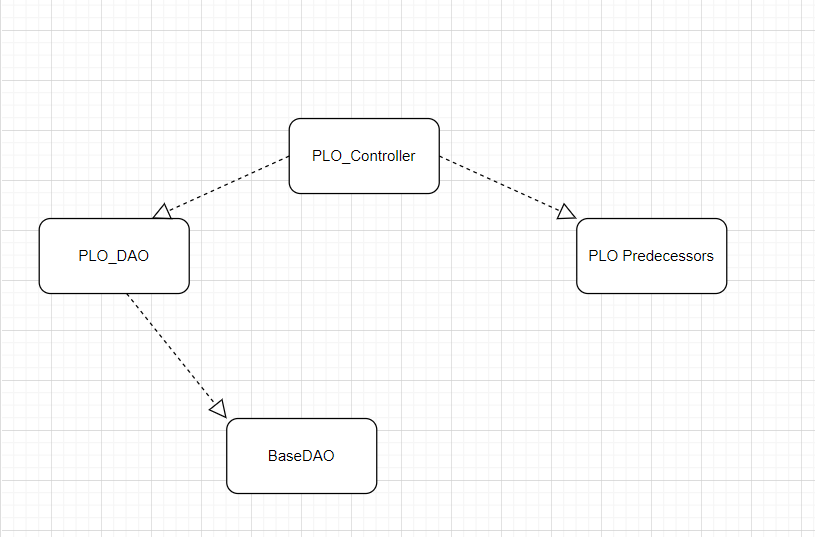
on s.subject\_id = d.parent\_id

WHERE s.parent\_id is null and s.code like ?

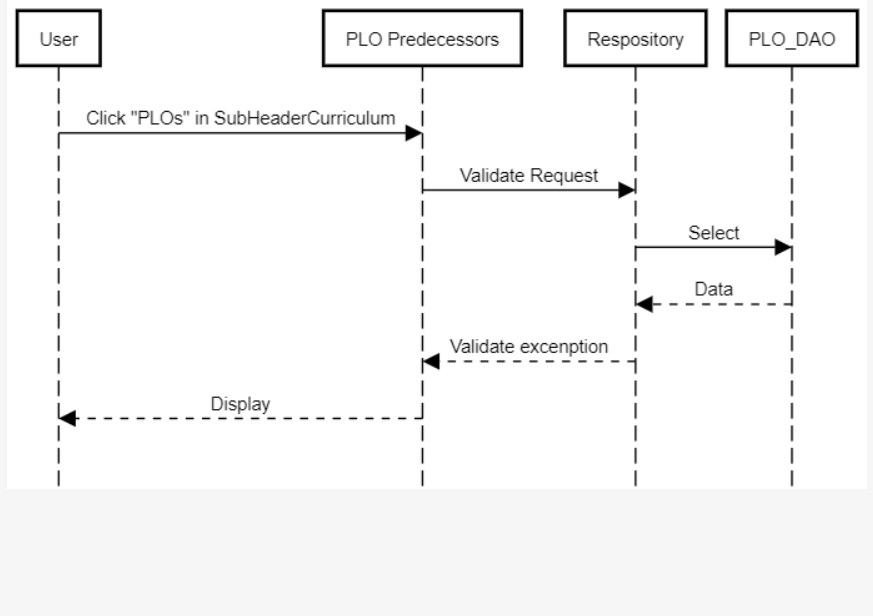
## PLO Management

### 5.1 View PLO List

#### a.Class Diagram

****

#### b. Sequence Diagram(s)



#### c. Database Queries

select count(\*) from plo where curriculum\_id = ?;

SELECT \* FROM swp391\_g3\_flm.plo where curriculum\_id = ?

order by plo\_id

limit ?, 10;

### 5.2 View PLO Details

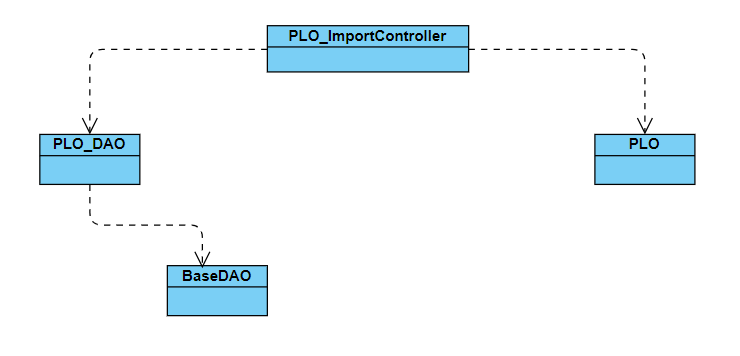
#### a.Class Diagram

#### b. Sequence Diagram(s)

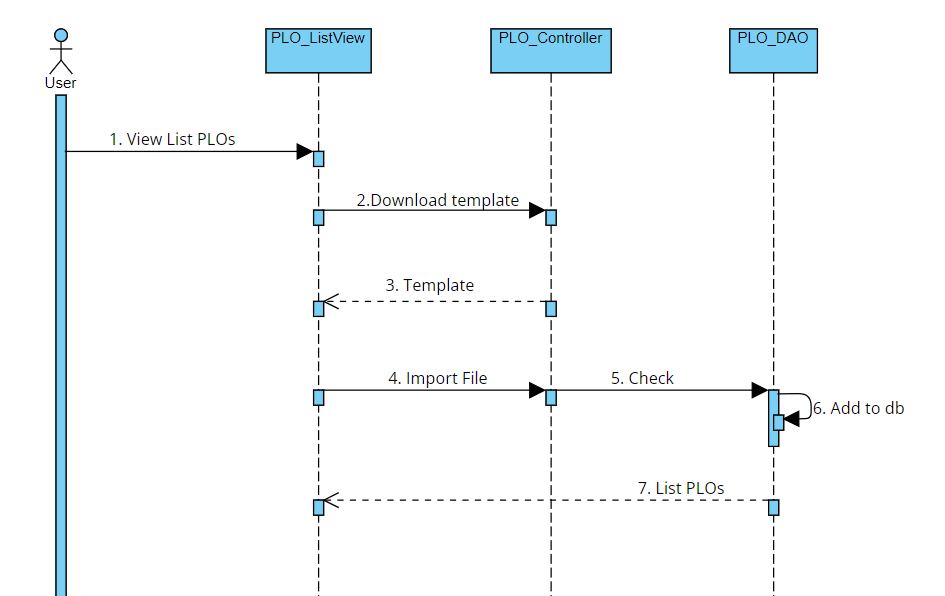
#### c. Database Queries

### 5.3 Import PLO

#### a.Class Diagram



#### b. Sequence Diagram(s)

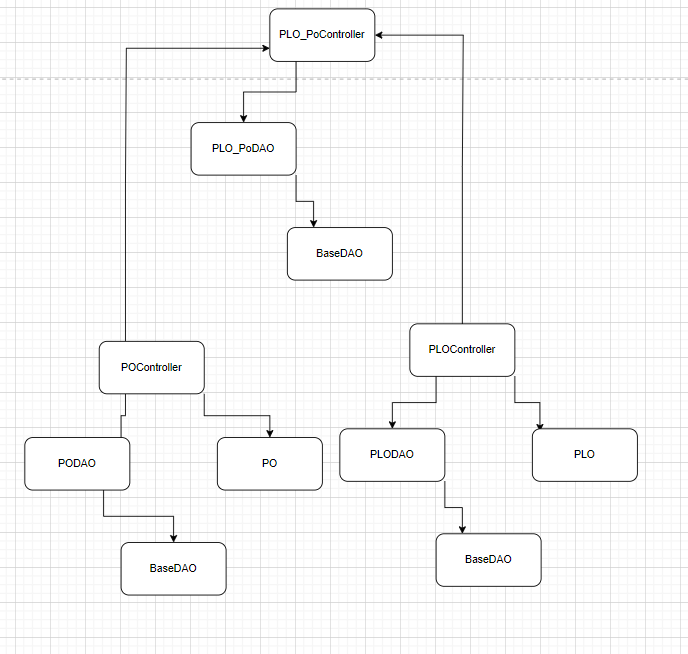


#### c. Database Queries

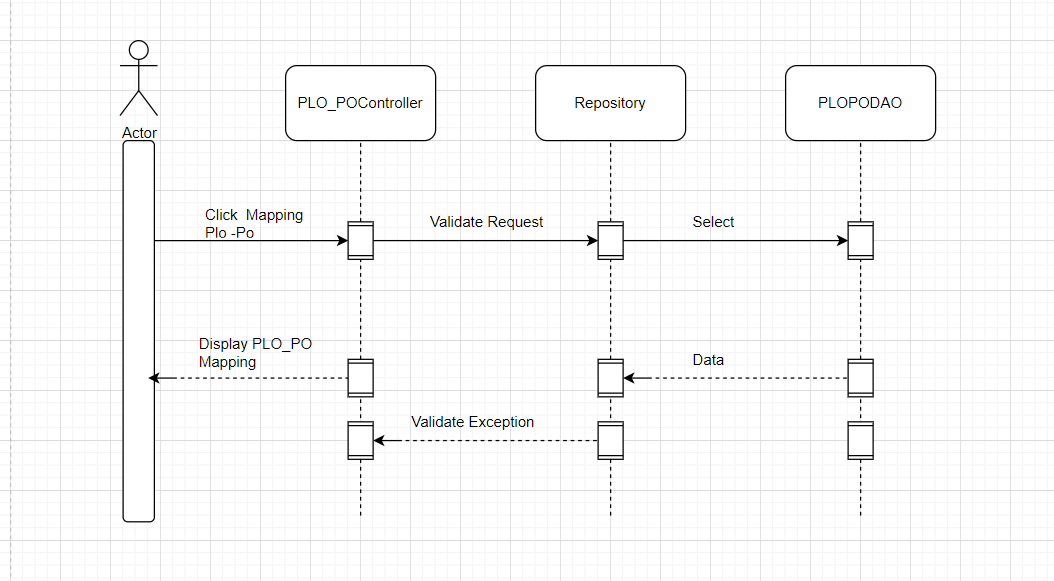
1. select count(\*) from plo where curriculum\_id = ?;
2. SELECT \* FROM swp391\_g3\_flm.plo where curriculum\_id = ? order by plo\_id limit ?, 10;
3. INSERT INTO `swp391\_g3\_flm`.`plo` (`name`, `description`, `curriculum\_id`) VALUES (?, ?, ?);

### 5.4 Mapping PLO-PO

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

"select \* FROM po\_plo pp\n"

+ "where pp.po\_id = ? and pp.plo\_id = ?;"

""

+ "DELETE FROM `po\_plo`\n"

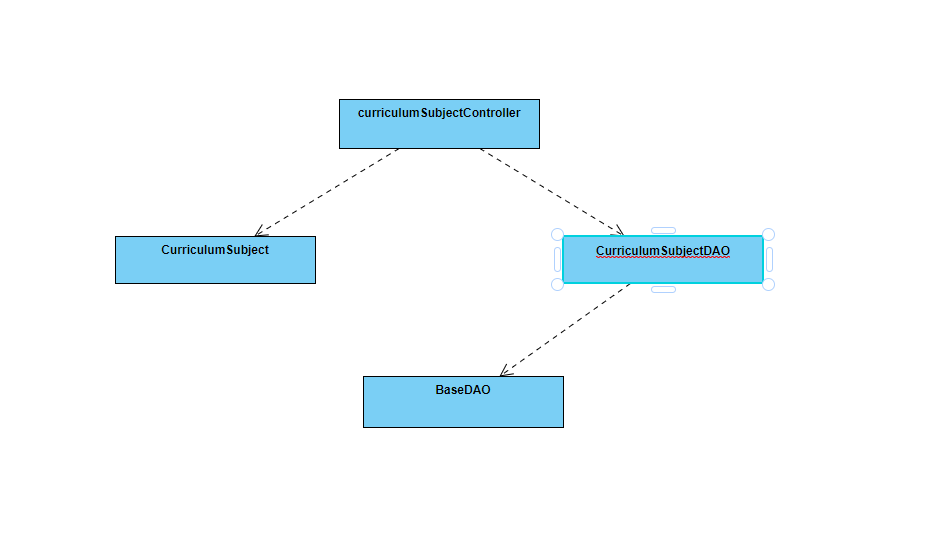
+ "WHERE po\_id = ? AND plo\_id = ?";

+ "INSERT INTO `po\_plo` (`po\_id`, `plo\_id`) VALUES (?, ?);";

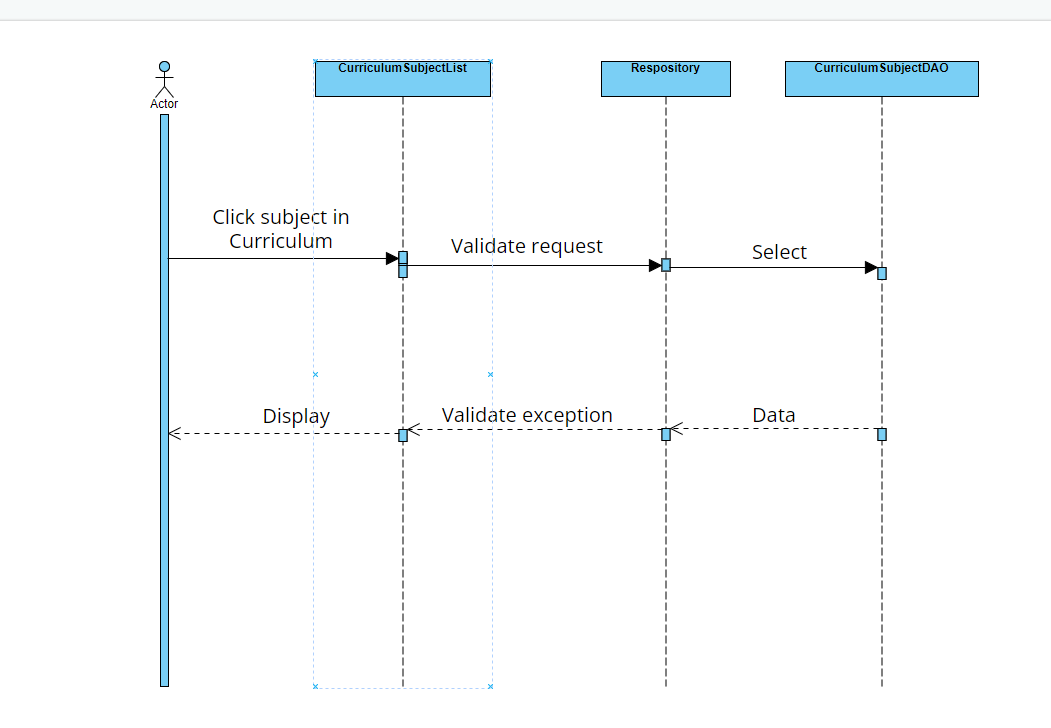
## Curriculum Subject

### 6.1 View Curriculum Subject Details

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

#### c. Database Queries

"select curriculum\_subject.subject\_id,code,name,semester,no\_credit "

+ "from curriculum\_subject join subject on curriculum\_subject.subject\_id = subject.subject\_id "

+ "where curriculum\_id = ? and type <> 'subcombo'"

+ "order by curriculum\_subject.subject\_id\n"

+ "limit ?,3; "

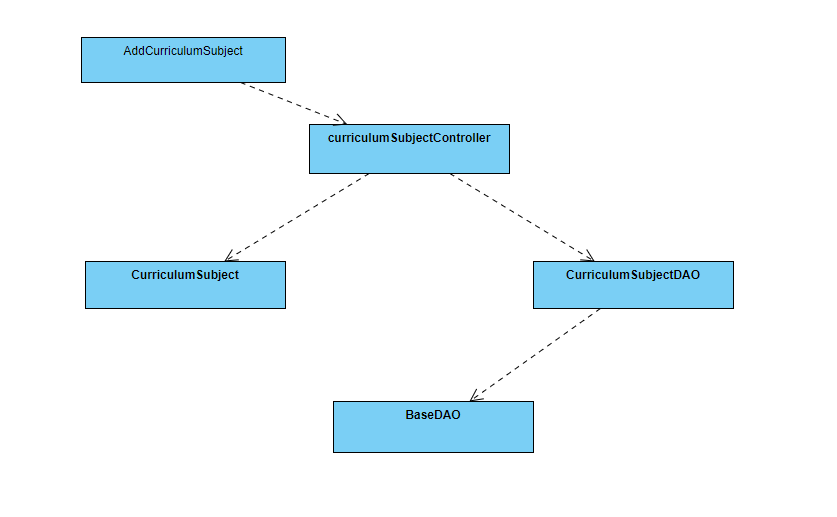
"select count(\*) "

+ "from curriculum\_subject join subject on curriculum\_subject.subject\_id = subject.subject\_id "

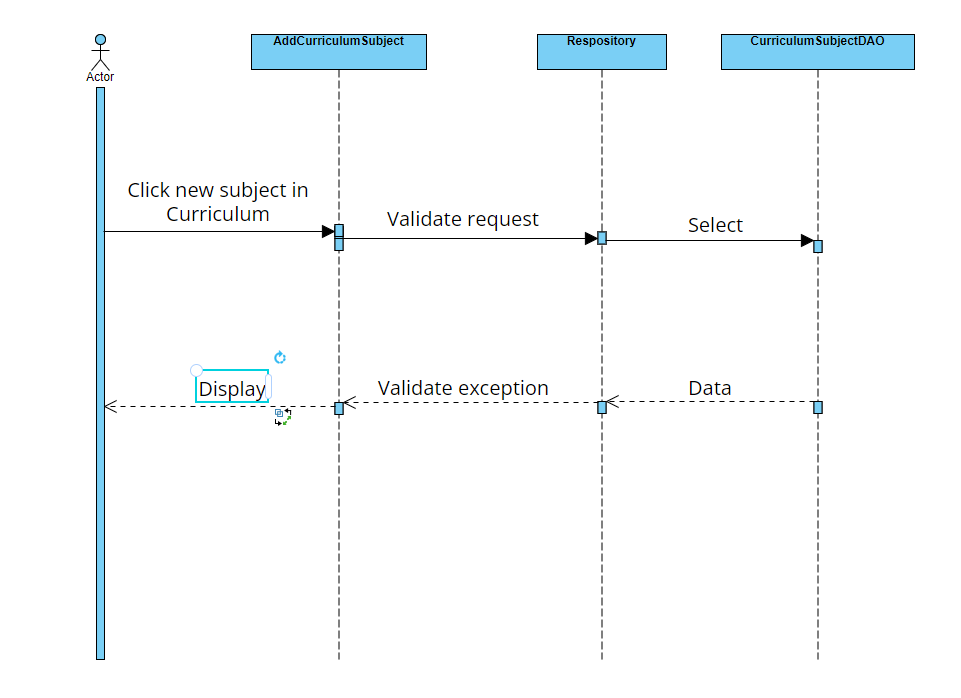
+ "where curriculum\_id = ? and type <> 'subcombo'"

### 6.2 Add New Curriculum Subject

#### a.Class Diagram

****

#### b. Sequence Diagram(s)

******

#### c. Database Queries

"select \* from curriculum\_subject where subject\_id =? and curriculum\_id= ?"

"select \* from `group` where group\_id =? "

"select \* from subject where subject\_id =? "

"select \* from syllabus where syllabus\_id =? "

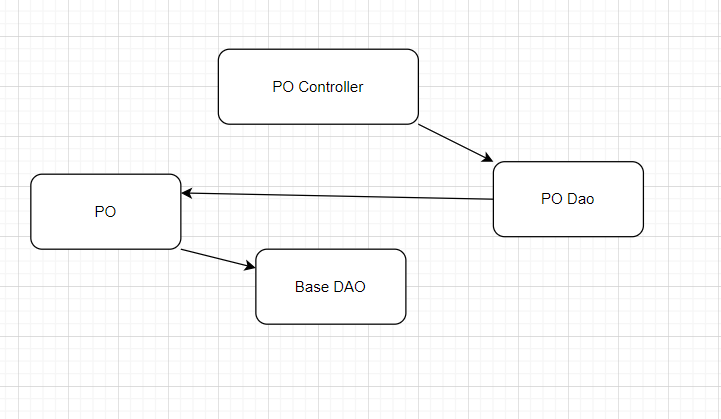
"INSERT INTO `swp391\_g3\_flm`.`curriculum\_subject` (`curriculum\_id`, `subject\_id`,`syllabus\_id`, `group\_id`, `semester`, `no\_credit`, `is\_active`)"

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

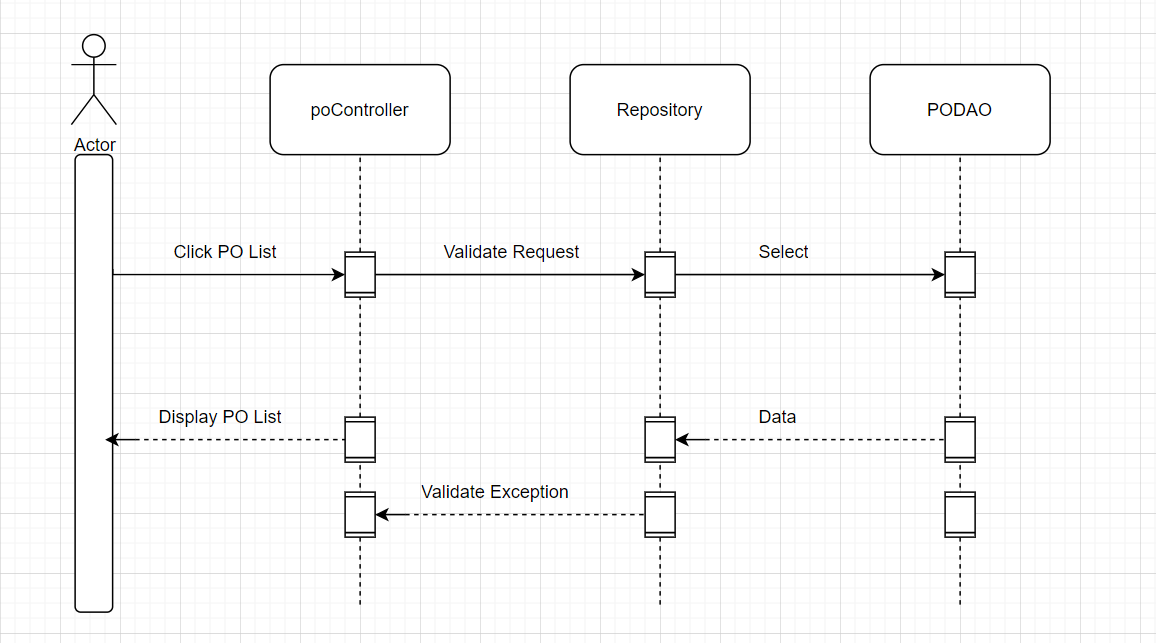
## PO Management

### 7.1 View PO List

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

SELECT po\_id, p.name, p.description, p.curriculum\_id, p.is\_active, c.code

FROM po p left join curriculum c on p.curriculum\_id = c.curriculum\_id

order by po\_id

limit 0, 10;

select count(\*) from po

"SELECT po\_id, p.name, p.description, p.curriculum\_id, p.is\_active, c.code\n"

+ "FROM po p left join curriculum c on p.curriculum\_id = c.curriculum\_id "

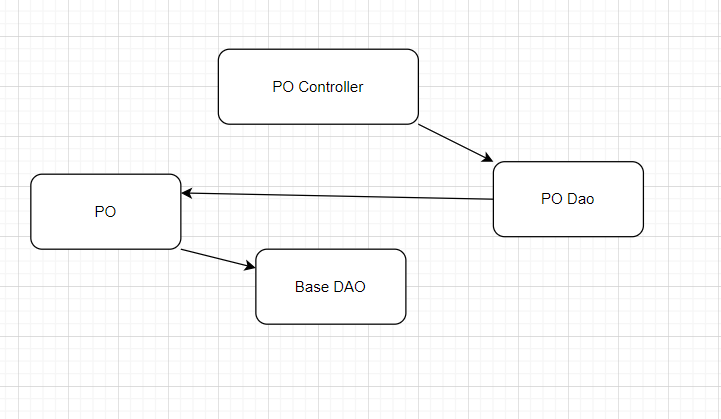
+ " where p.name like '" + "%" + txt + "%" + "'"

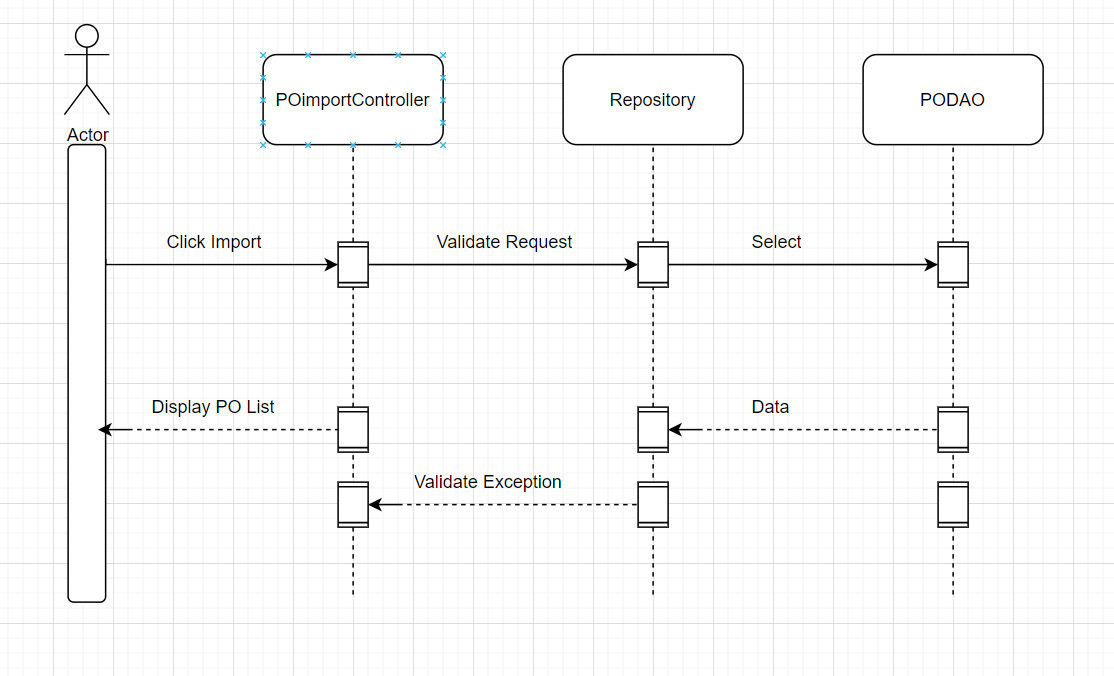
+ " OR po\_id like '" + "%" + txt + "%" + "'"

+ " OR c.code like '" + "%" + txt + "%" + "'";

### 7.2 Import PO

#### a.Class Diagram

b. Sequence Diagram(s)



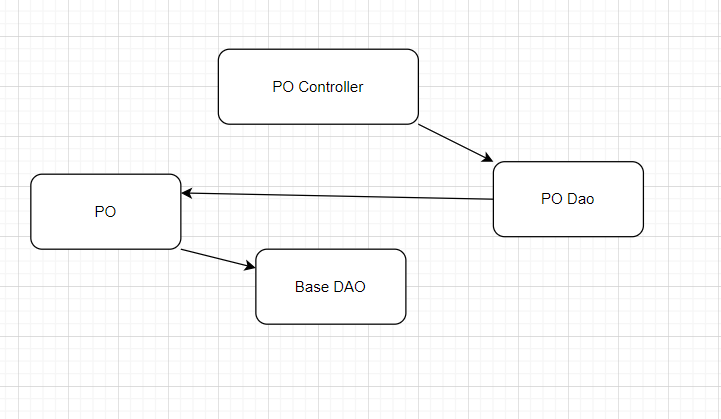
#### c. Database Queries

"INSERT INTO `swp391\_g3\_flm`.`po` (`name`, `description`, `is\_active`)"

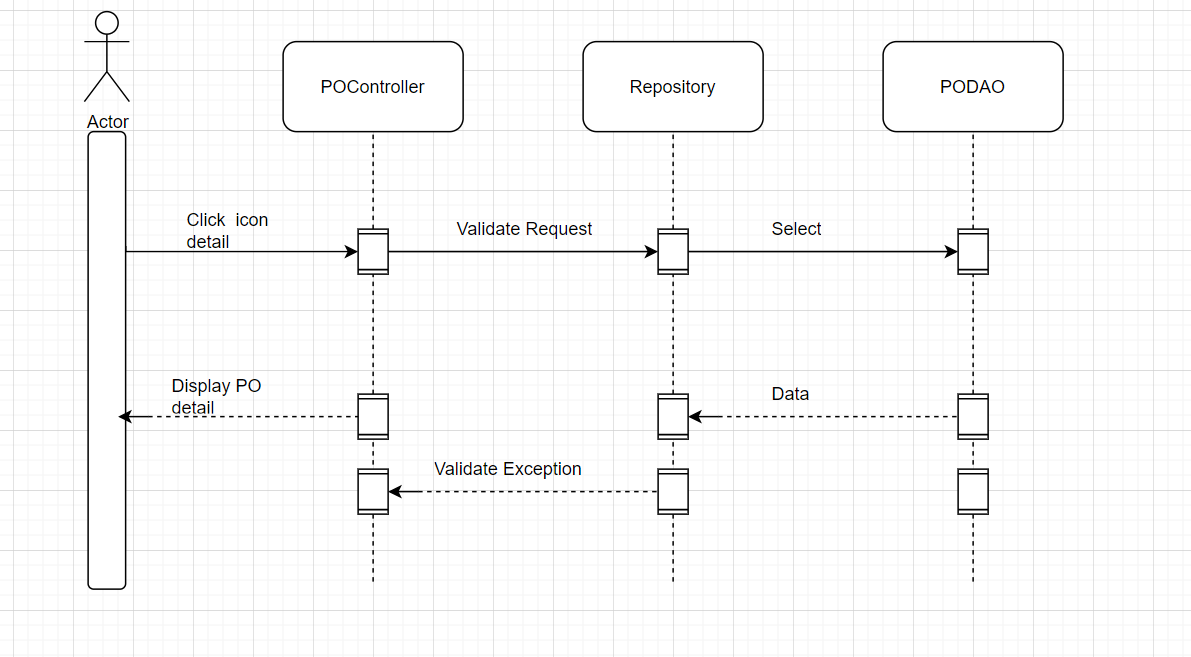
+ " VALUES (?, ?, b?);";

### 7.3 View PO Detail

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

"SELECT po\_id, p.name, p.description, p.curriculum\_id, p.is\_active, c.code\n"

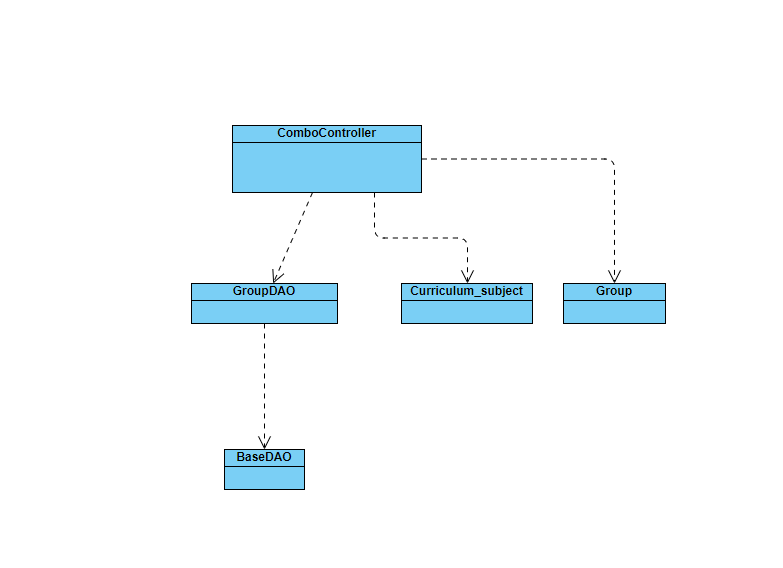
+ "FROM po p left join curriculum c on p.curriculum\_id = c.curriculum\_id \n"

+ "where po\_id = ?;";

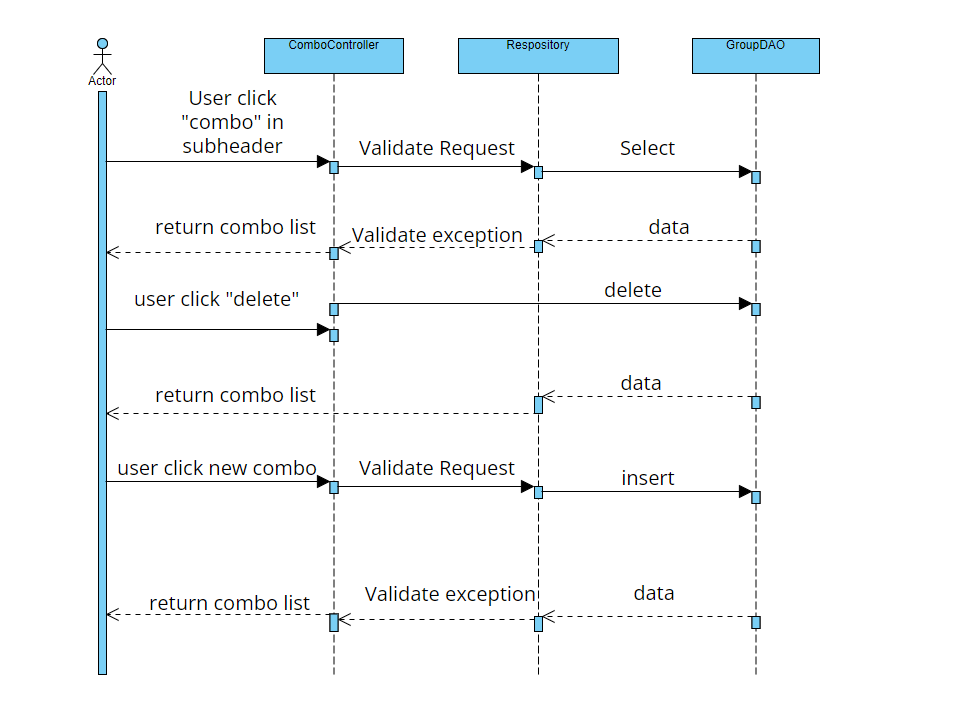
## Subject Groups

### 8.1 View Combo List

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

SELECT `group`.group\_id,`group`.name,subject.code,subject.name "

+ "FROM swp391\_g3\_flm.curriculum\_subject join subject "

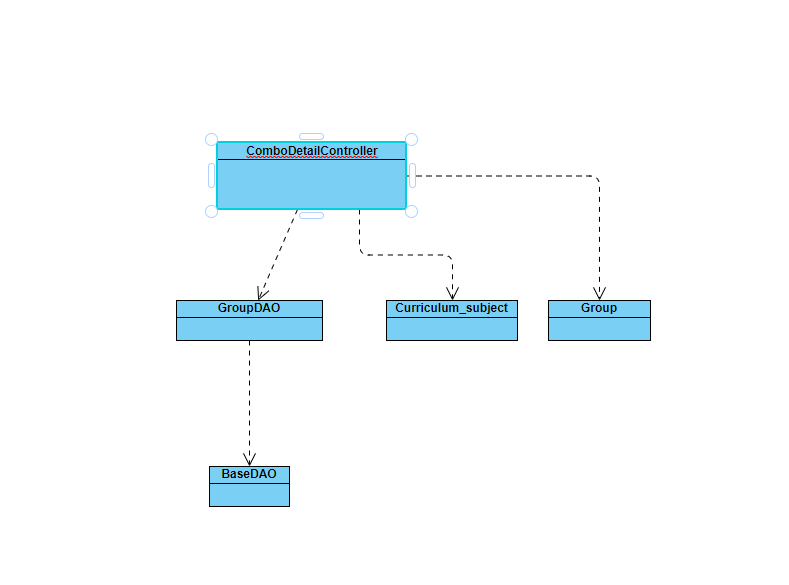
+ "on curriculum\_subject.subject\_id = subject.subject\_id "

+ "join `group` on `group`.group\_id = subject.group\_id "

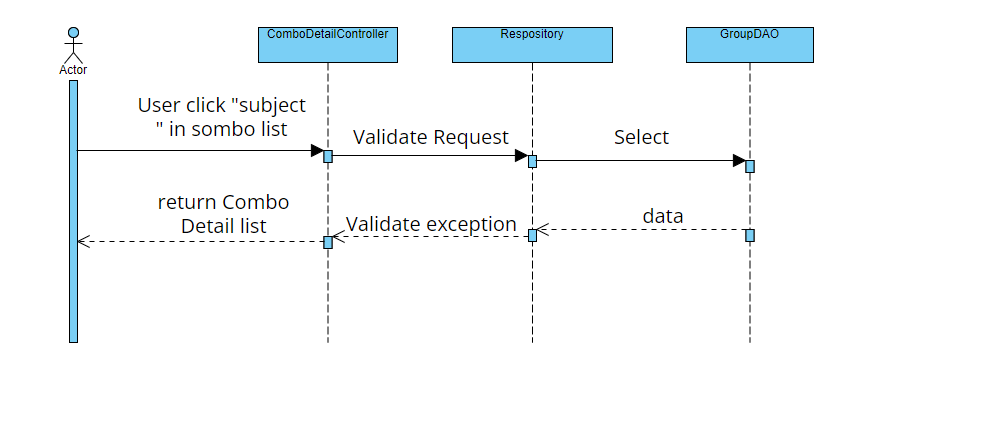
+ "where curriculum\_subject.curriculum\_id = ? and type = 'Combo'

### 8.2 View Combo Detail

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

SELECT subject.subject\_id,subject.name,subject.code,curriculum\_subject.semester "

+ "FROM swp391\_g3\_flm.curriculum\_subject join subject "

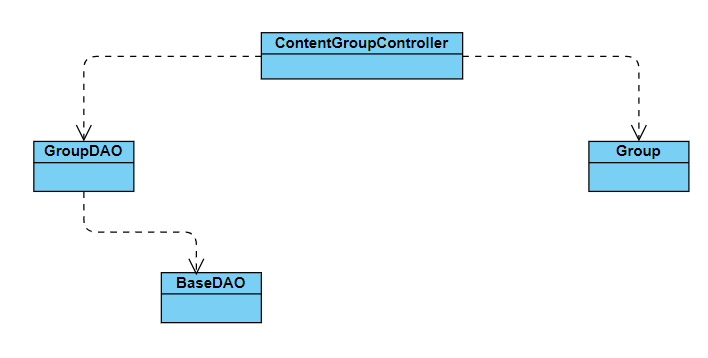
+ "on curriculum\_subject.subject\_id = subject.subject\_id "

+ "join `group` on `group`.group\_id = subject.group\_id "

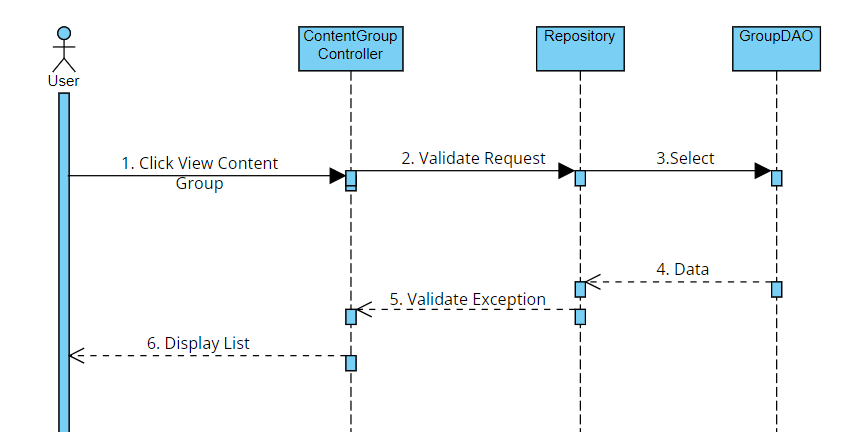
+ "where curriculum\_subject.curriculum\_id = ? and type = 'Default' and `group`.group\_id = ?

### 8.3 View Content Group List

#### a.Class Diagram



#### b. Sequence Diagram(s)



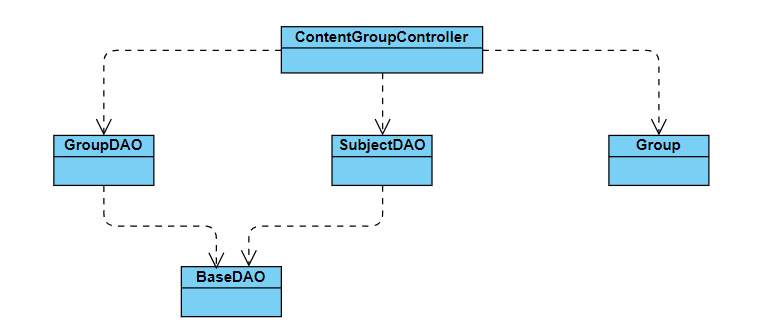
#### c. Database Queries

SELECT \* FROM swp391\_g3\_flm.`group` where curriculum\_id = ? and combo\_subject = ? order by display\_order limit ?, 10;

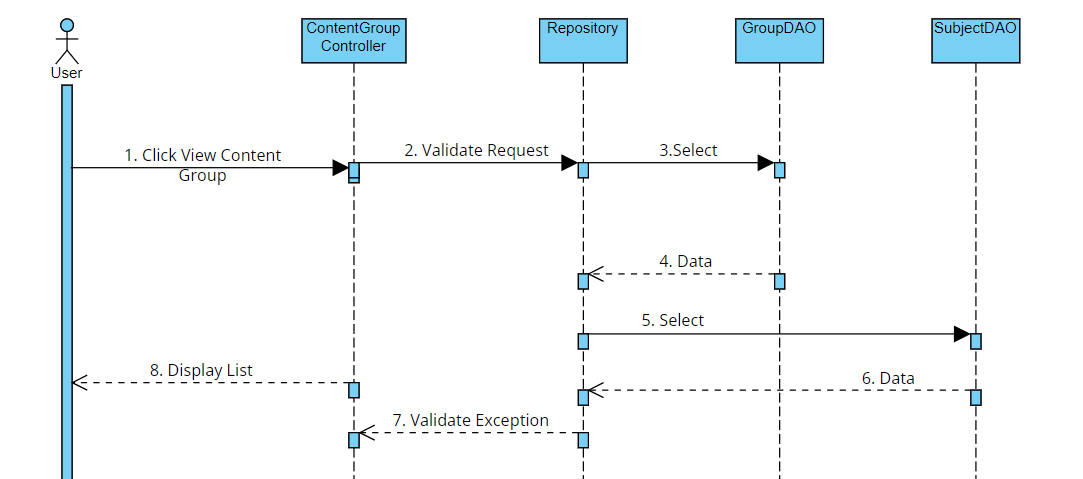
select count(\*) from `group` where curriculum\_id = ?

### 8.4 View Content Group Details

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

SELECT \* FROM swp391\_g3\_flm.`group` where curriculum\_id = ? and combo\_subject = ? order by display\_order limit ?, 10;

select count(\*) from `group` where curriculum\_id = ?

select \* from `group` where group\_id =?

select \* from `group` where name =? and curriculum\_id = ?

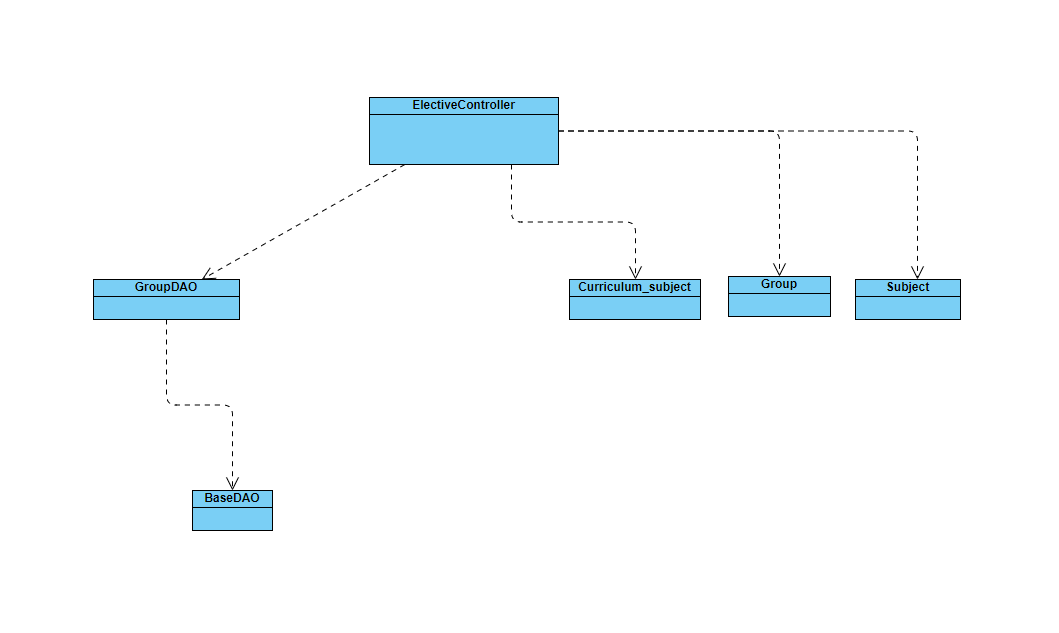
INSERT INTO `swp391\_g3\_flm`.`group` (`name`, `combo\_subject`, `curriculum\_id`, `display\_order`, `is\_active`) VALUES (?, ?, ?, ?, b?);

UPDATE `swp391\_g3\_flm`.`curriculum\_subject` SET `group\_id` = null WHERE (`group\_id` = ?);

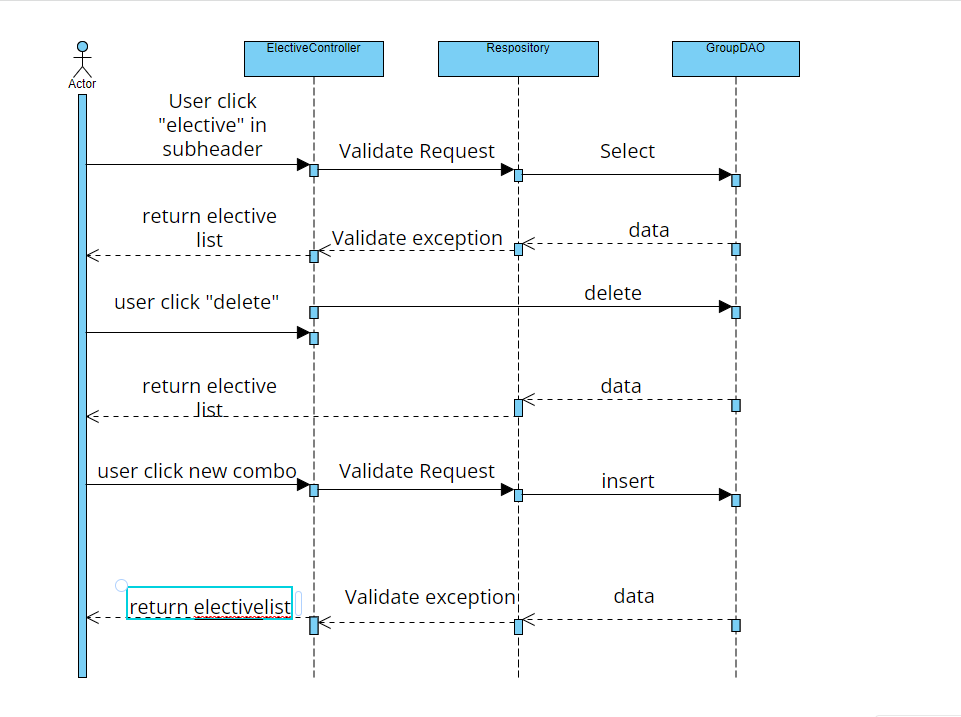
DELETE FROM `swp391\_g3\_flm`.`group` WHERE (`group\_id` = ?);

### 8.5 View Elective List

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

SELECT `group`.group\_id,`group`.name,subject.code,subject.name "

+ "FROM swp391\_g3\_flm.curriculum\_subject join subject "

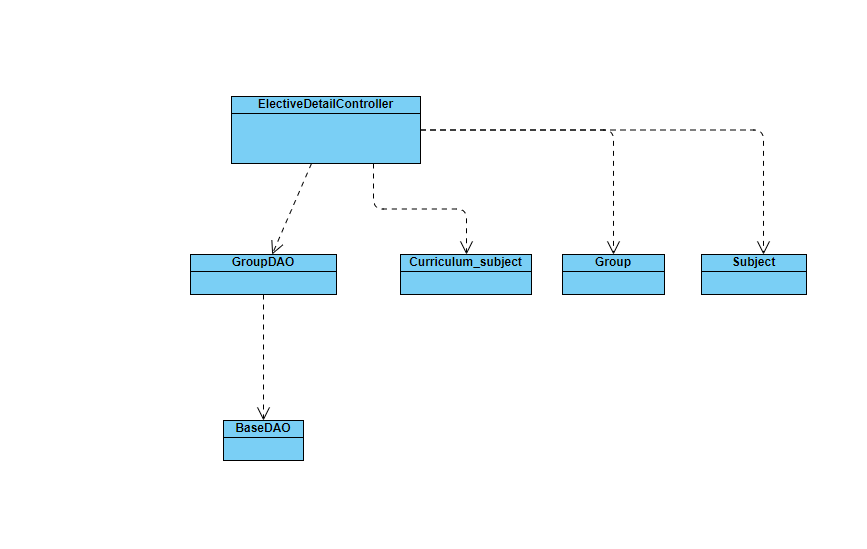
+ "on curriculum\_subject.subject\_id = subject.subject\_id "

+ "join `group` on `group`.group\_id = subject.group\_id "

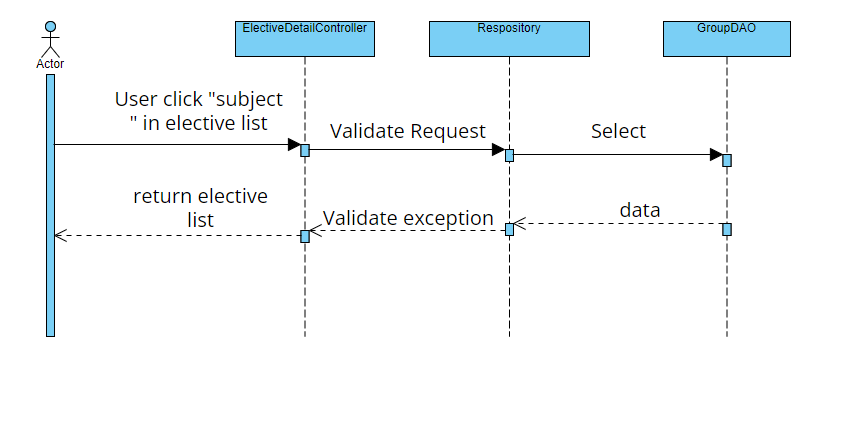
+ "where curriculum\_subject.curriculum\_id = ? and type = 'Elective' and `group`.is\_active = 1

### 8.6 View Elective Details

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

SELECT subject.subject\_id,`group`.name,subject.code,subject.name "

+ "FROM swp391\_g3\_flm.curriculum\_subject join subject "

+ "on curriculum\_subject.subject\_id = subject.subject\_id "

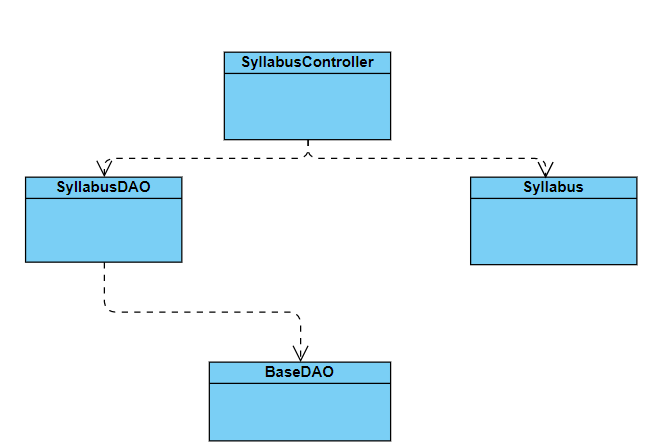
+ "join `group` on `group`.group\_id = subject.group\_id "

+ "where curriculum\_subject.curriculum\_id = ? and type = 'Default' and `group`.group\_id = ?

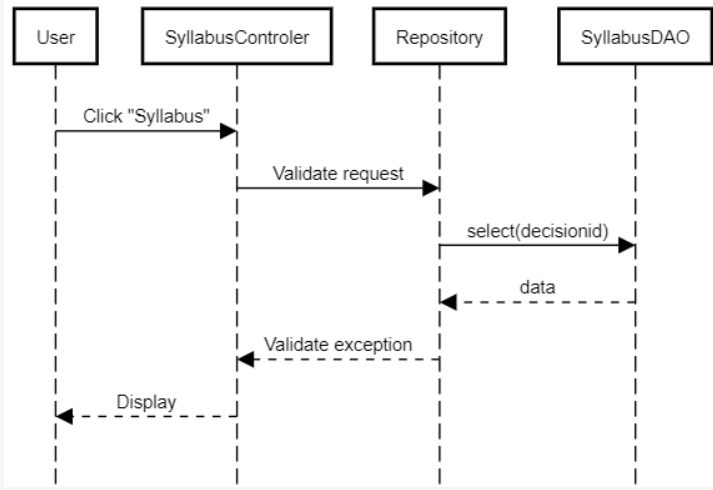
## Syllabus Management

### 9.1 View Syllabus List

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

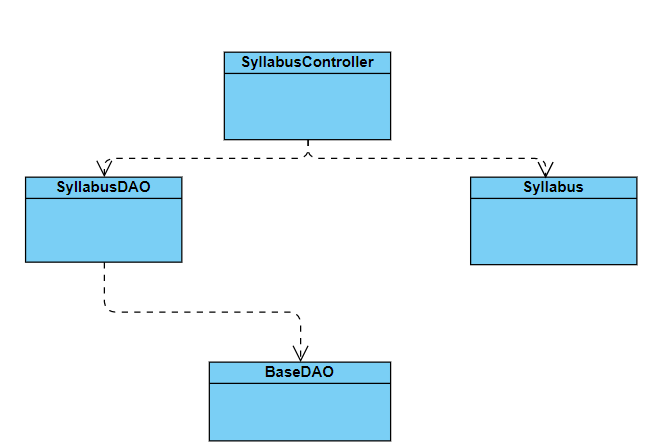
SELECT \* FROM syllabus order by syllabus\_id limit ?, 10

select count(\*) from syllabus

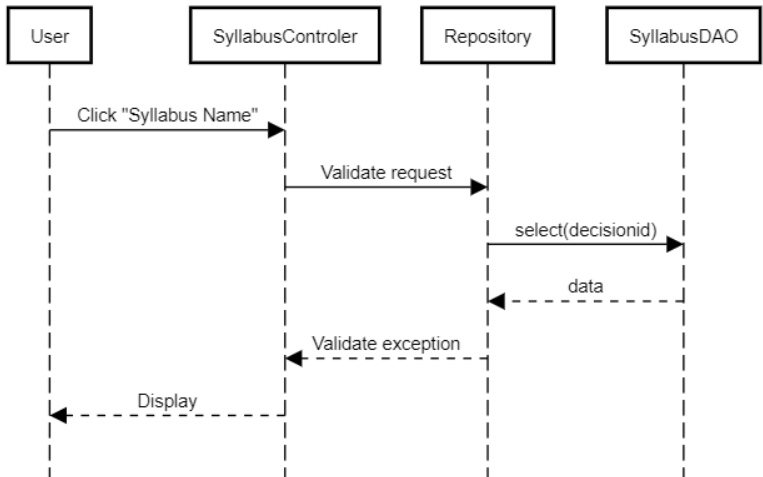
SELECT \* FROM syllabus where name like ? ;

### 9.2 View Syllabus Overview

#### a.Class Diagram



#### b. Sequence Diagram(s)



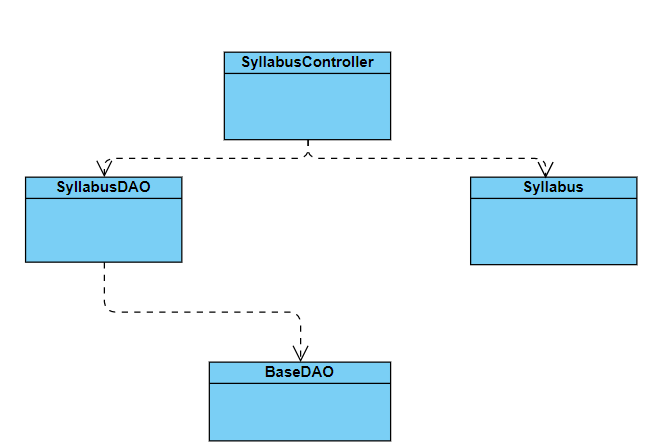
#### c. Database Queries

SELECT \* FROM syllabus where syllabus\_id = ?;

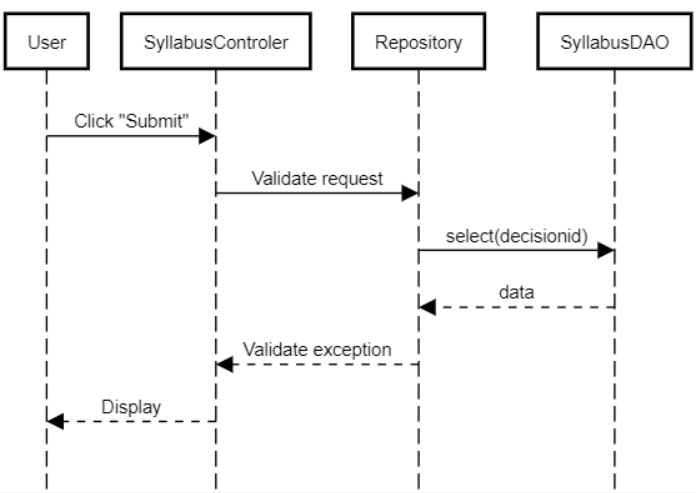
#### 9.3 Add Syllabus

#### a Class Diagram

#### 



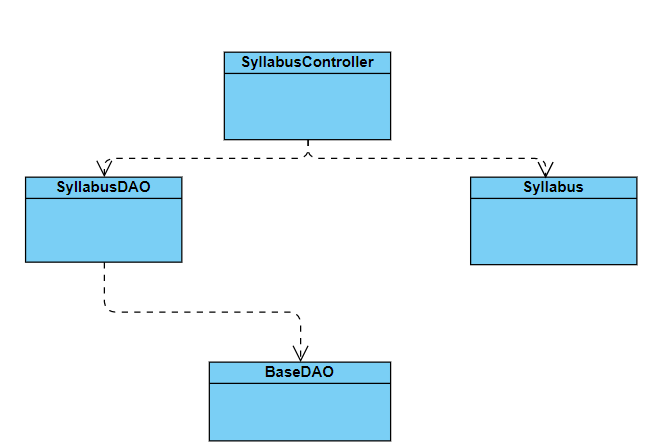
#### b. Sequence Diagram(s)



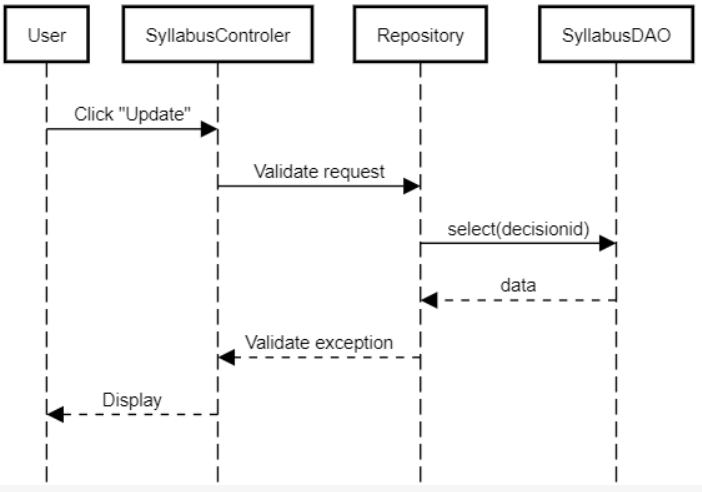
#### c. Database Queries

### 9.4 Edit Syllabus

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

UPDATE syllabus SET name = ?, no\_of\_credit = ?, is\_active = ?, is\_approved = ?, degree\_level = ?

WHERE syllabus\_id = ?;

### 9.5 Import Syllabus

#### a.Class Diagram

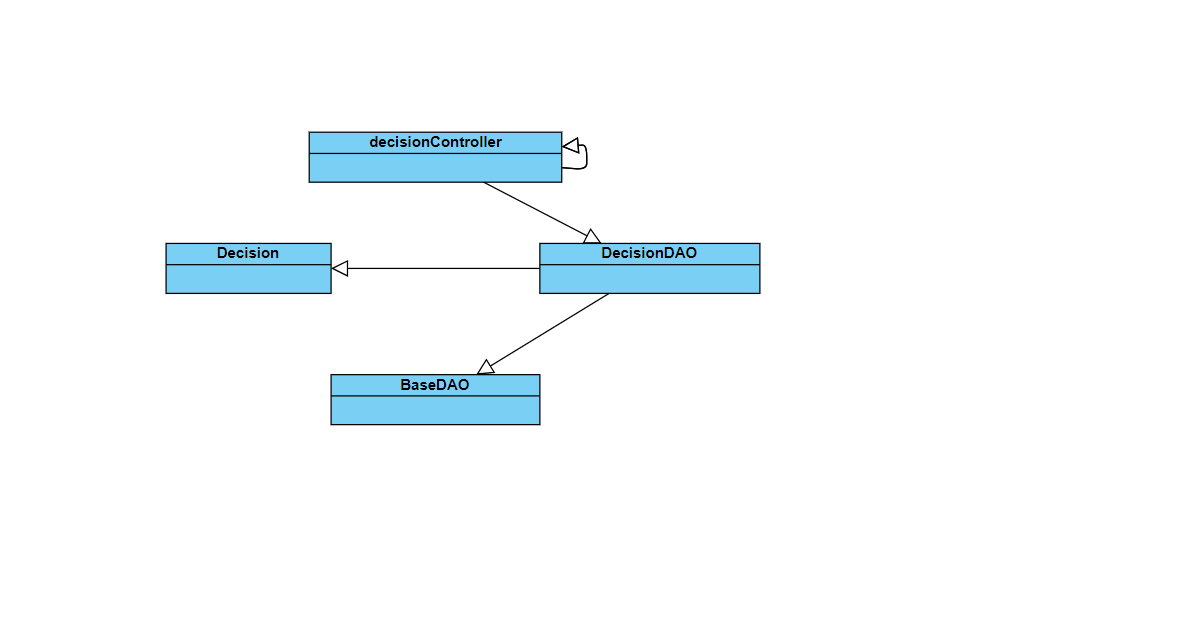
#### b. Sequence Diagram(s)

#### c. Database Queries

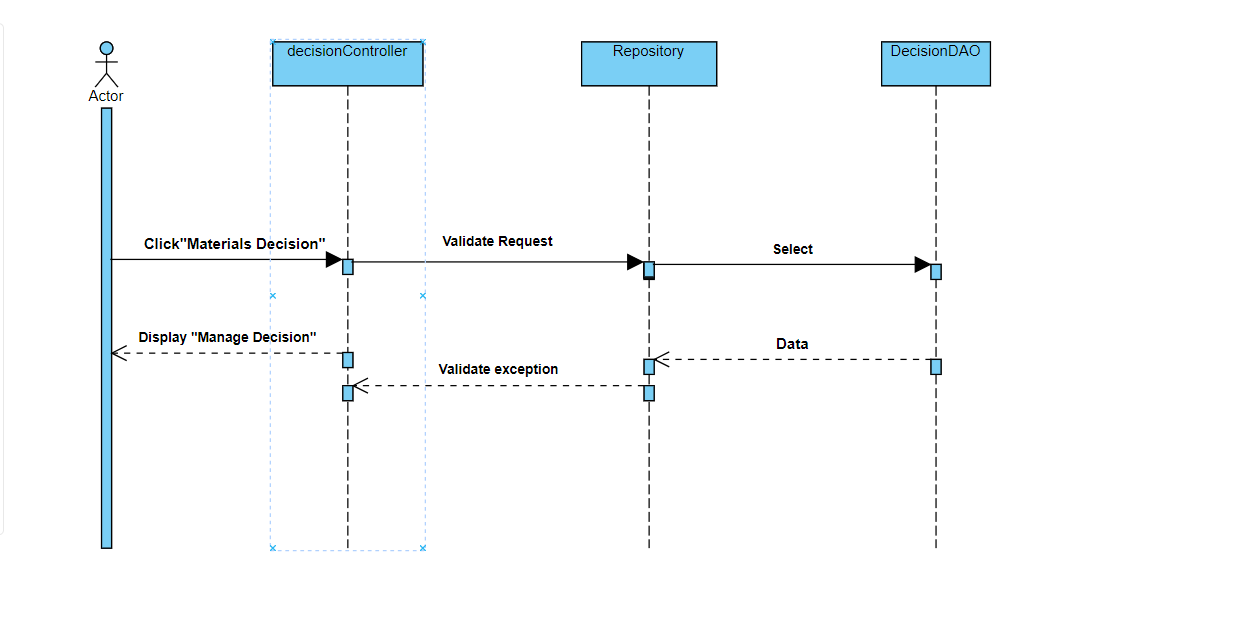
## Decision Management

### 10.1 View Decision List

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

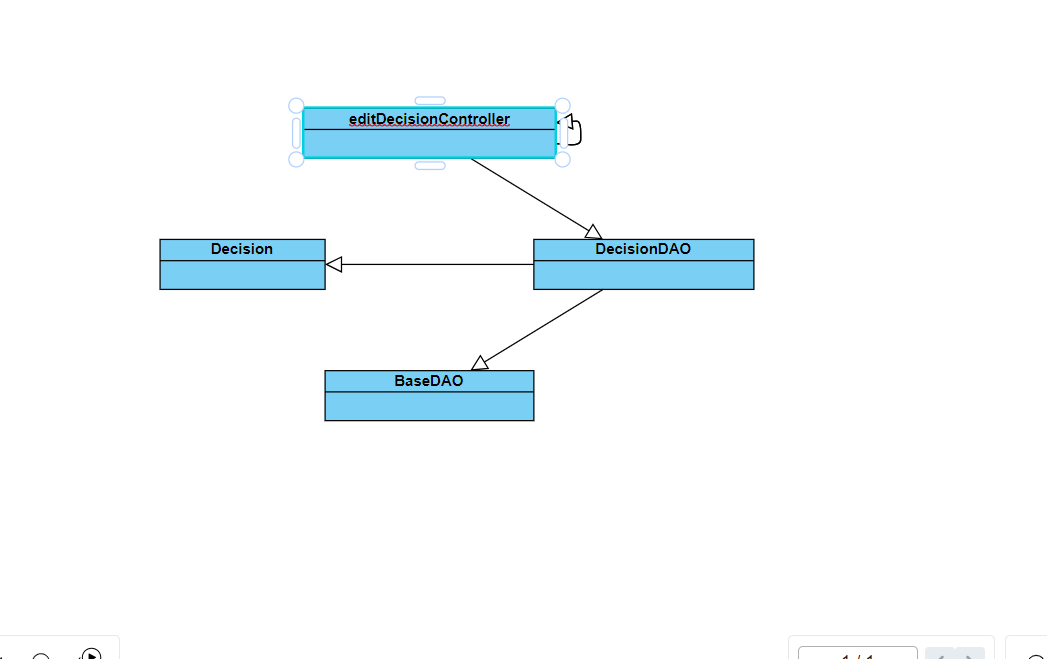
SELECT \* FROM decision where creator\_id = ? order by decision\_id\n"

+ "limit ?, 10;"

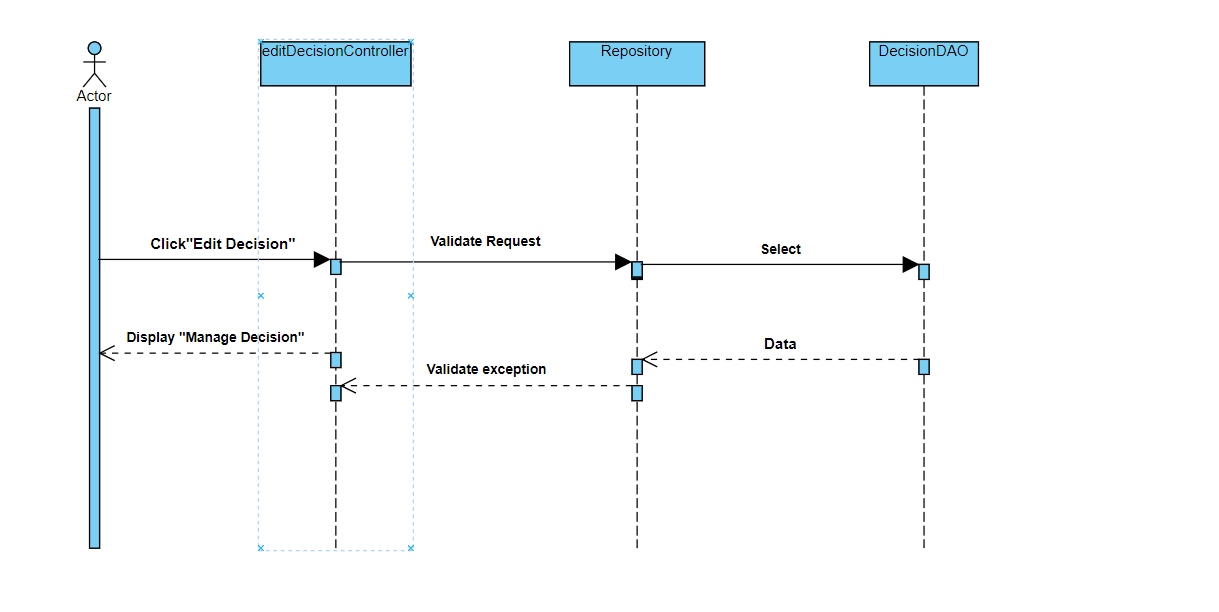
select count(\*) from decision

### 10.2 View Decision Details

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

1.UPDATE `swp391\_g3\_flm`.`decision`"

+ " SET decision\_no = ? ,decision\_name = ? , decision\_ApprovedDate = ? , note = ? , decision\_CreateDate = ? , creator\_id = ?"

+ "Where decision\_id = ?

2.SELECT \* FROM decision where creator\_id = ? order by decision\_id\n"

+ "limit ?, 10;

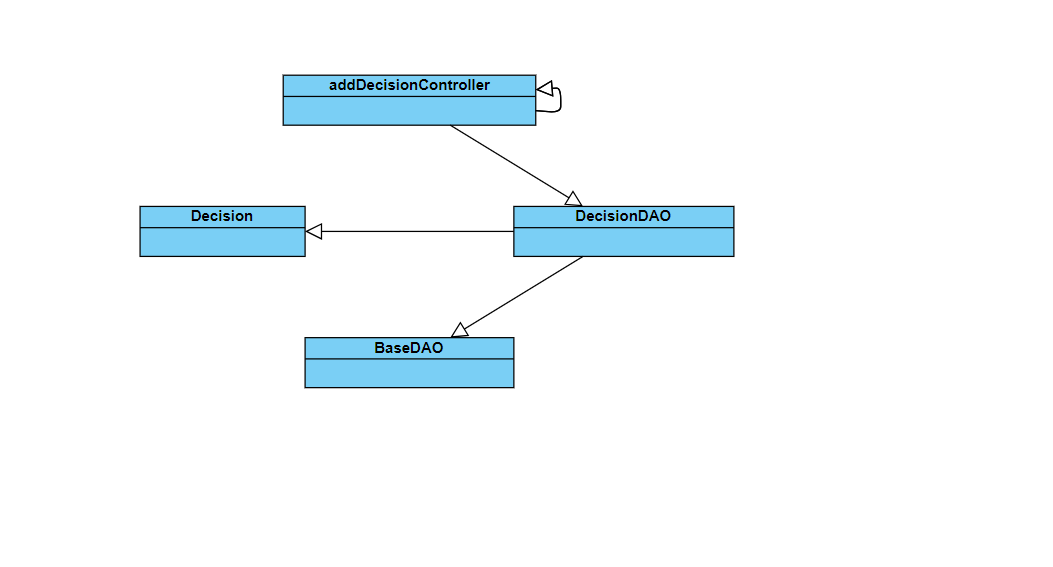
3.UPDATE `swp391\_g3\_flm`.`curriculum` SET decision\_id = ? Where curriculum\_id = ?

4.SELECT \* FROM decision where decision\_id = ?;

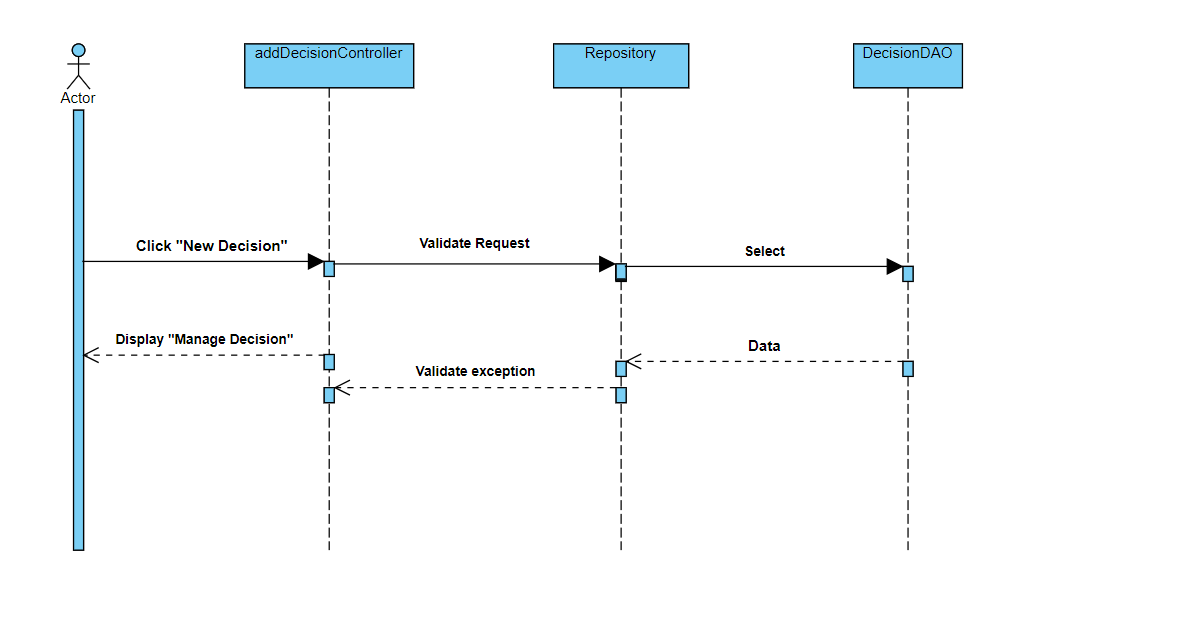
5.SELECT \* FROM curriculum where decision\_id is null;

### 10.3 Add New Decision

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

select \* from decision where decision\_no =?

INSERT INTO `swp391\_g3\_flm`.`decision` (`decision\_no`, `decision\_name`, `decision\_date`, `creator\_id`)"

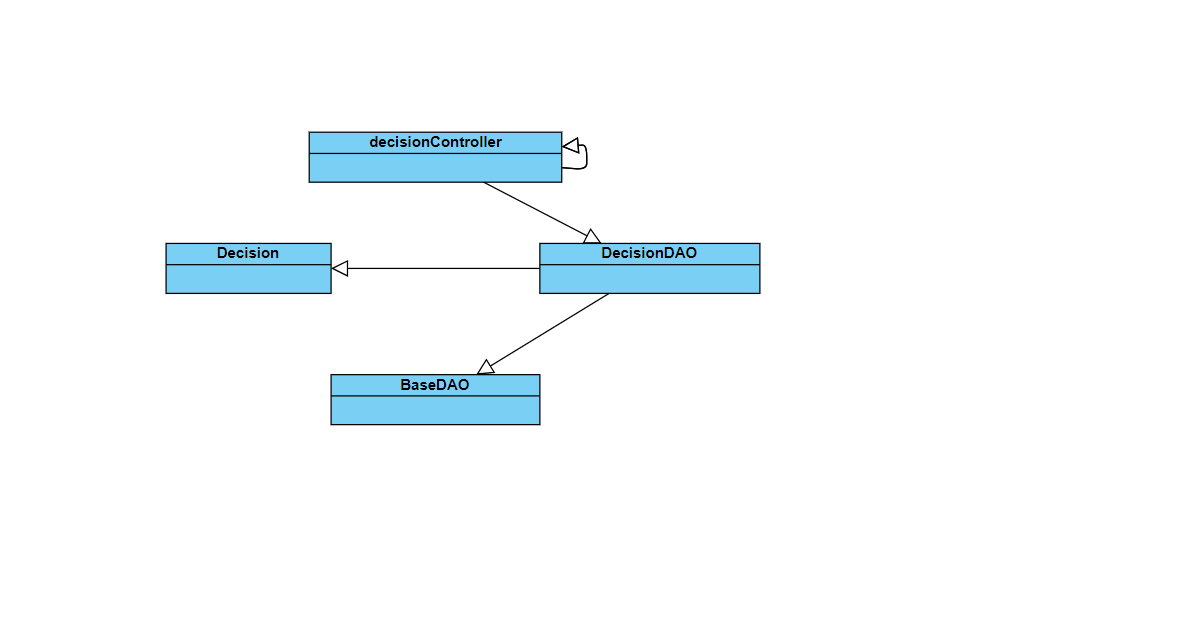
+ " VALUES (?, ?, ?, ?);

SELECT \* FROM decision where creator\_id = ? order by decision\_id\n"

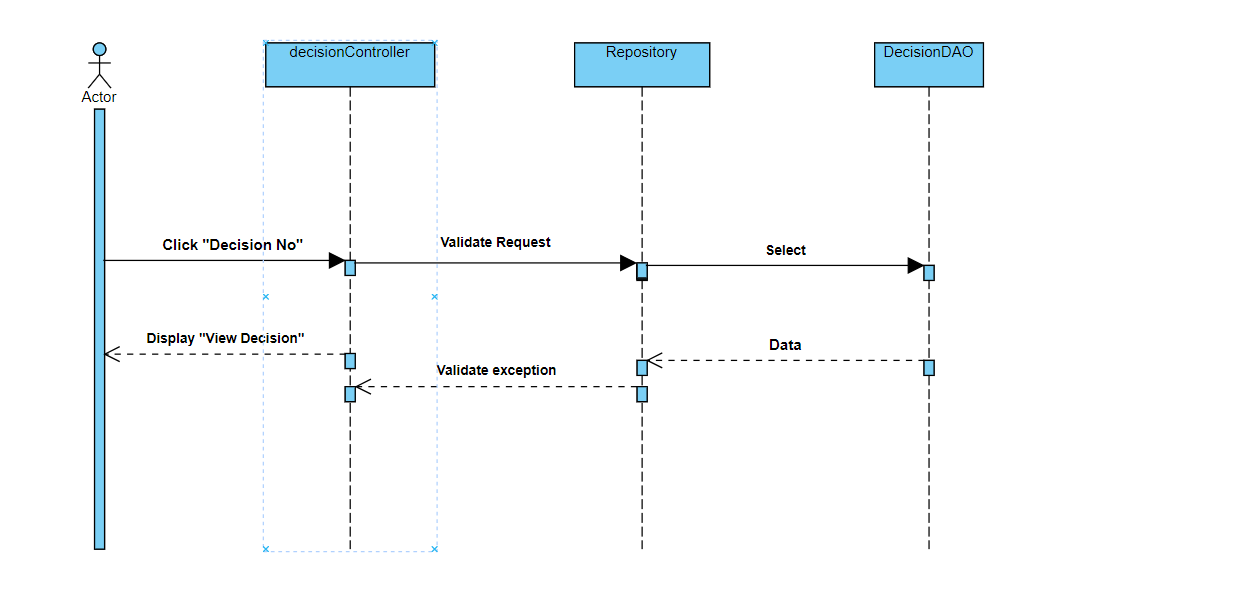
+ "limit ?, 10;

### 10.4 View Decision Overview

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

SELECT \* FROM decision where decision\_id = ?;

## Syllabus Submit

### 11.1 View Submit List

#### a.Class Diagram

#### b. Sequence Diagram(s)

#### c. Database Queries

### 11.2 View Submit Detail

#### a.Class Diagram

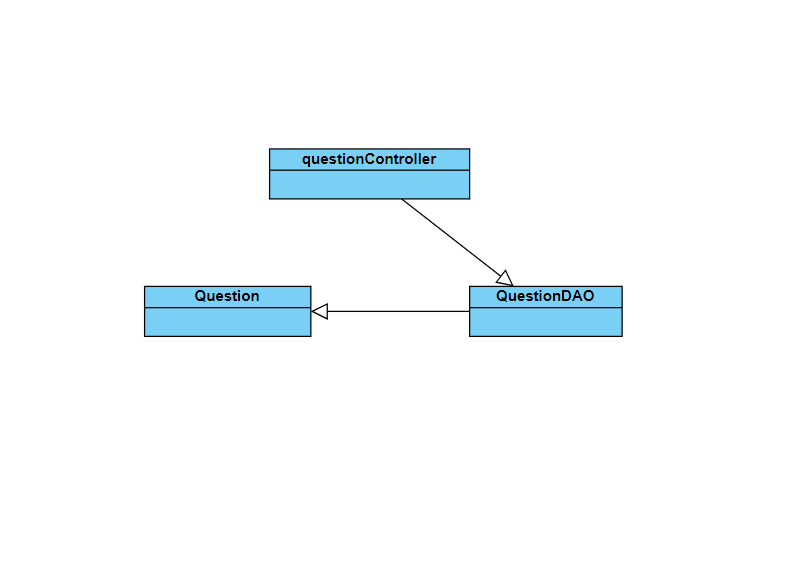
#### b. Sequence Diagram(s)

#### c. Database Queries

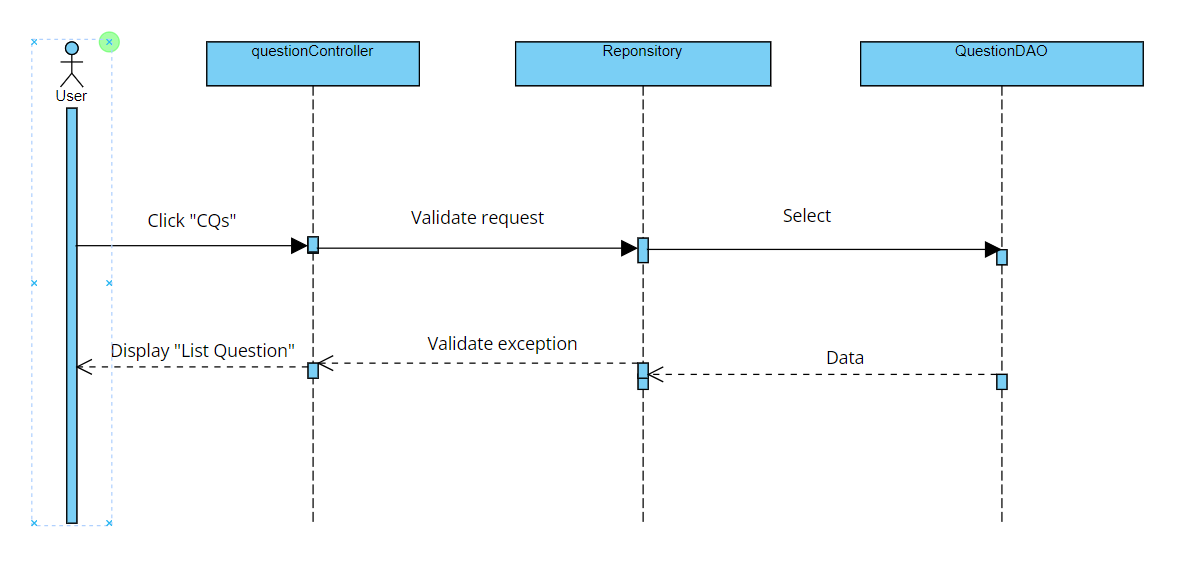
## Constructive Question

### 12.1 View Questions List

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

select \* from question q join session s on q.session\_id = s.session\_id where s.syllabus\_id = ? "

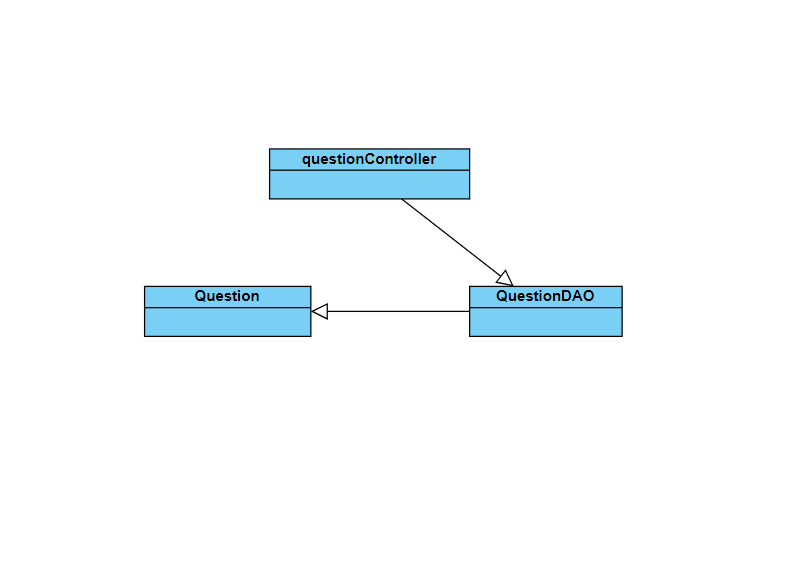
+ "limit ?, 10;

select count(\*) from question q join session s on q.session\_id = s.session\_id where s.syllabus\_id =?;

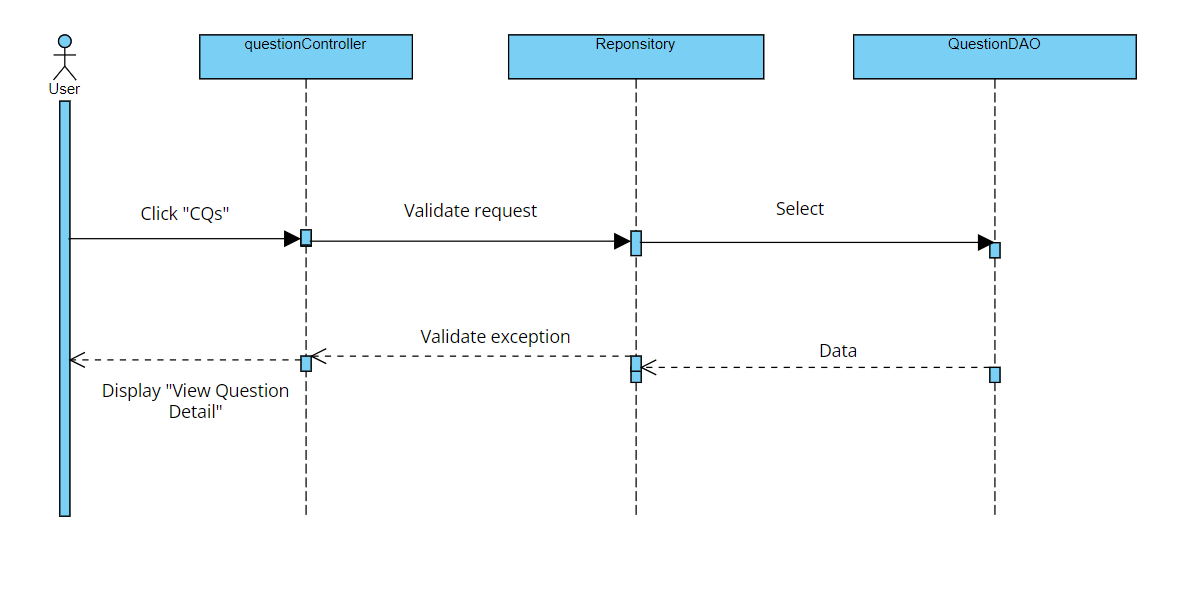
select \* from question q join session s on q.session\_id = s.session\_id where (q.name like ? OR q.details like ?)and s.syllabus\_id = ?;

### 12.2 View Questions Details

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

select \* from question q join session s on q.session\_id = s.session\_id where s.syllabus\_id = ? "

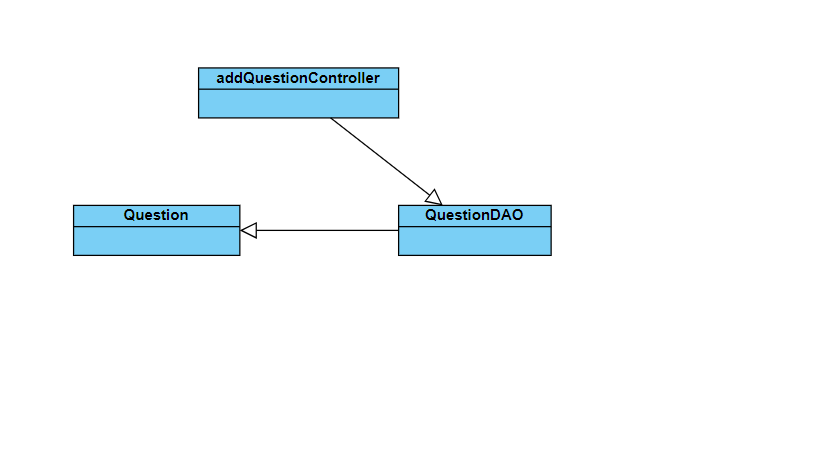
+ "limit ?, 10;

select count(\*) from question q join session s on q.session\_id = s.session\_id where s.syllabus\_id =?;

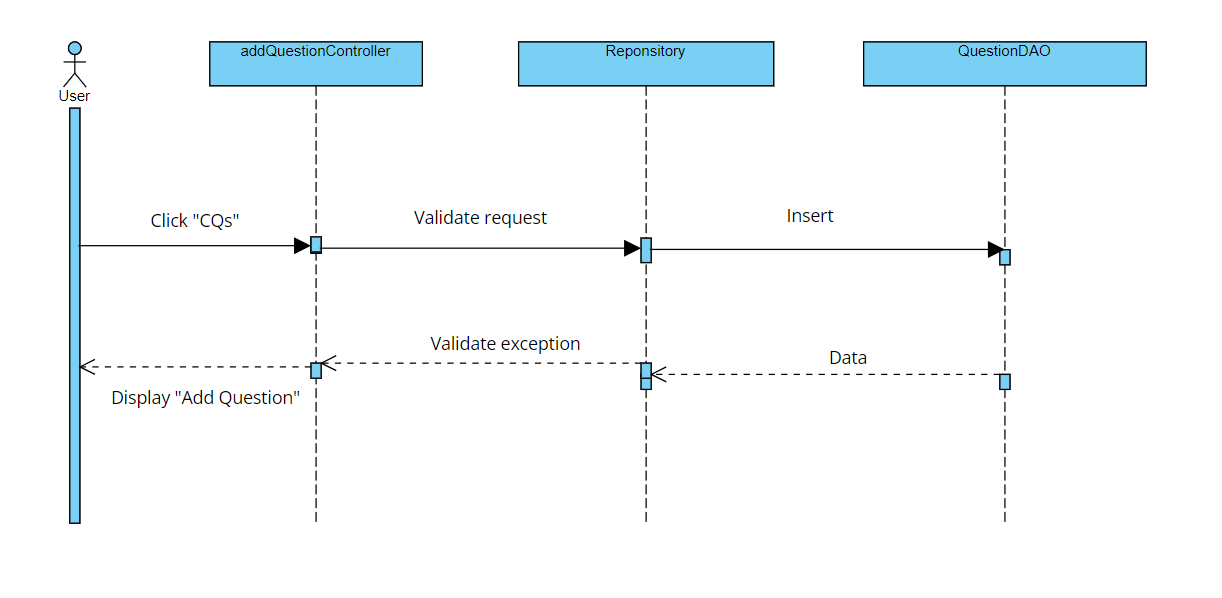
SELECT \* FROM question where question\_id = ?;

### 12.3 Import Questions

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

select count(\*) from question q join session s on q.session\_id = s.session\_id where s.syllabus\_id =?;

select \* from question q join session s on q.session\_id = s.session\_id where s.syllabus\_id = ? "

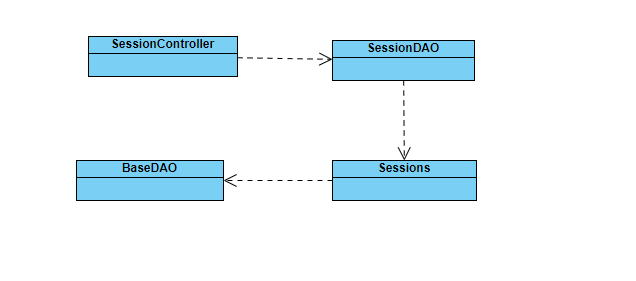
+ "limit ?, 10;

INSERT INTO `swp391\_g3\_flm`.`question` (`session\_id`, `name`, `details`) VALUES (?, ?, ?);

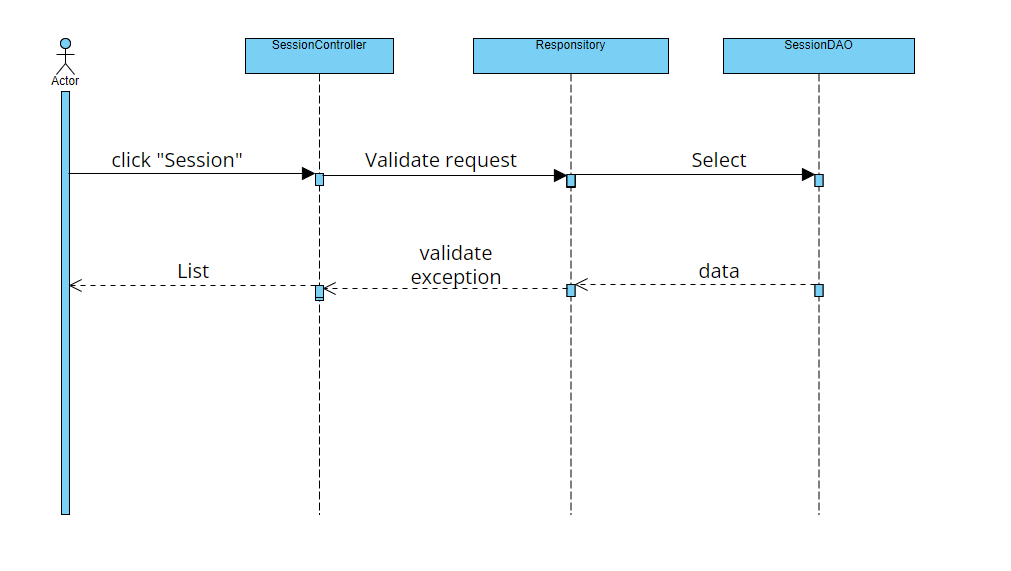
## Session Management

### 13.1 View Session List

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

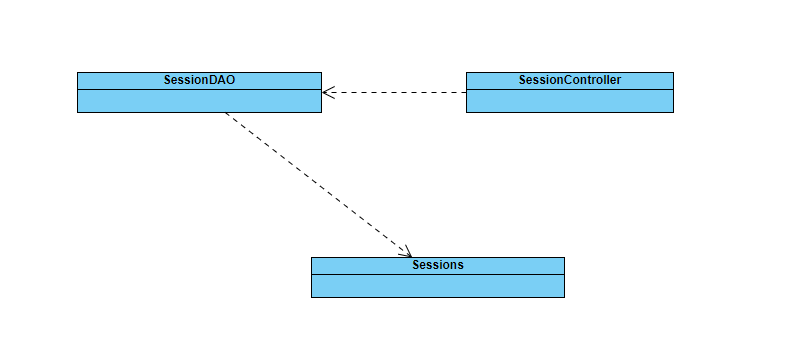
select session\_id,syllabus\_id,name,details,learning\_type"

+ " from session where syllabus\_id = ? "

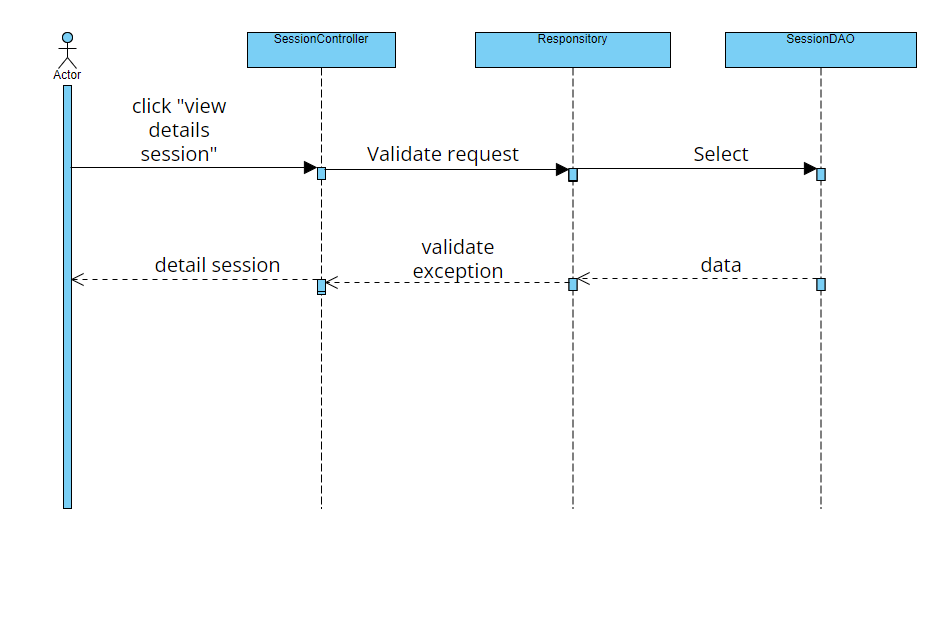
+ "limit ?, 10;

### 13.2 View Session Details

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

SELECT \* FROM session where session\_id = ?;

### 13.3 Import Session

#### a.Class Diagram

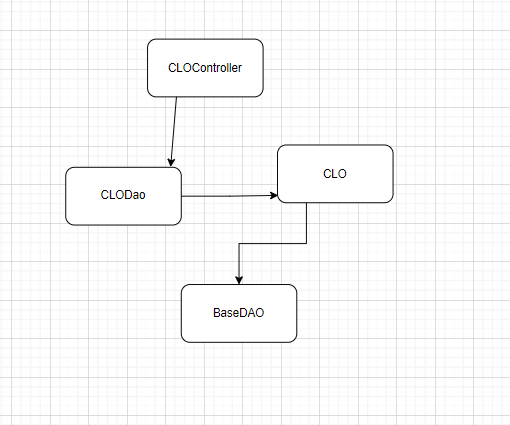
#### b. Sequence Diagram(s)

#### c. Database Queries

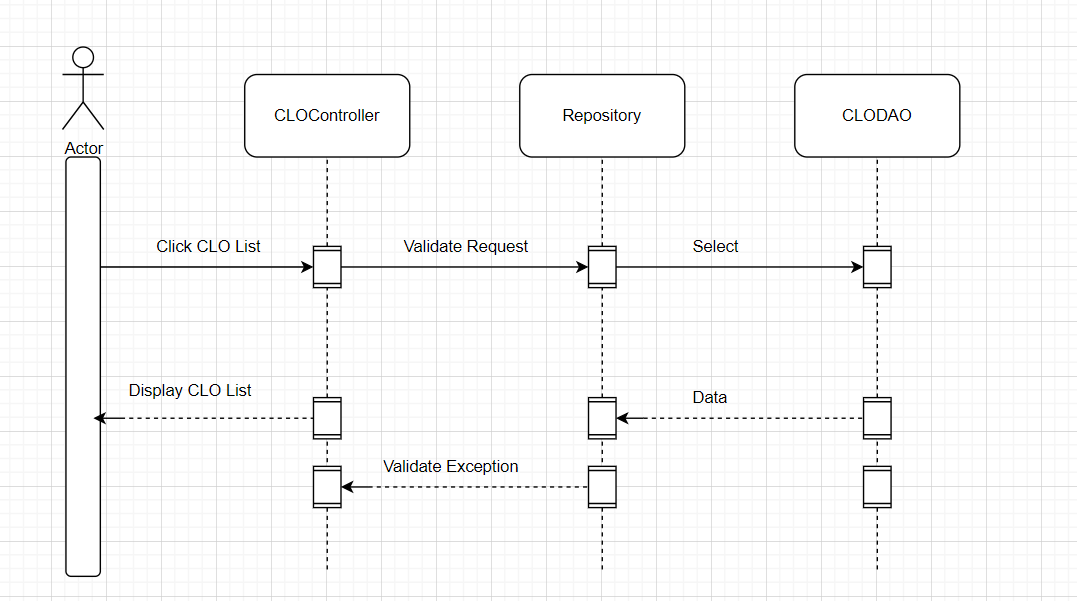
## CLO Management

### 14.1 View CLO List

#### a.Class Diagram



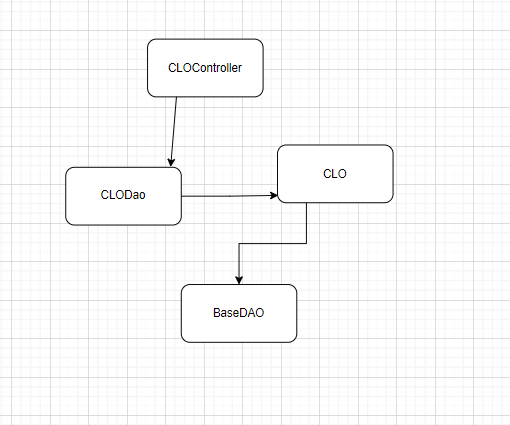
#### b. Sequence Diagram(s)



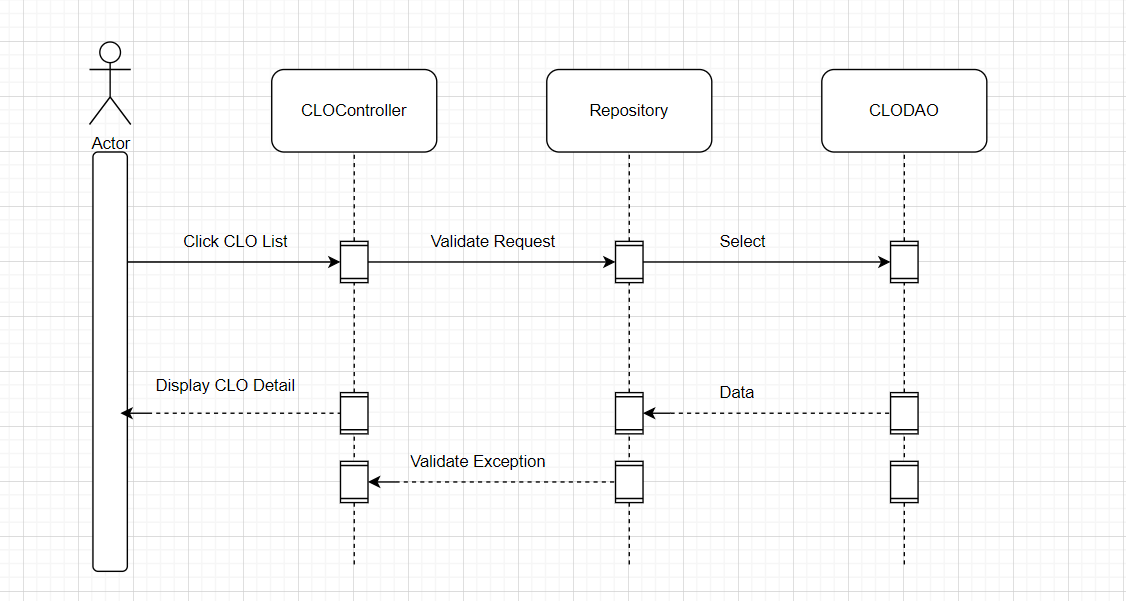
#### c. Database Queries

### 14.2 View CLO Details

#### a.Class Diagram



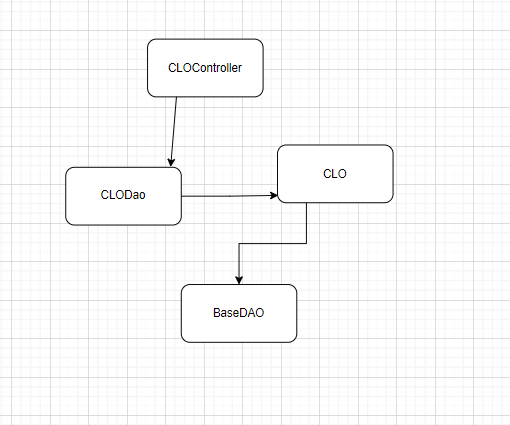
#### b. Sequence Diagram(s)



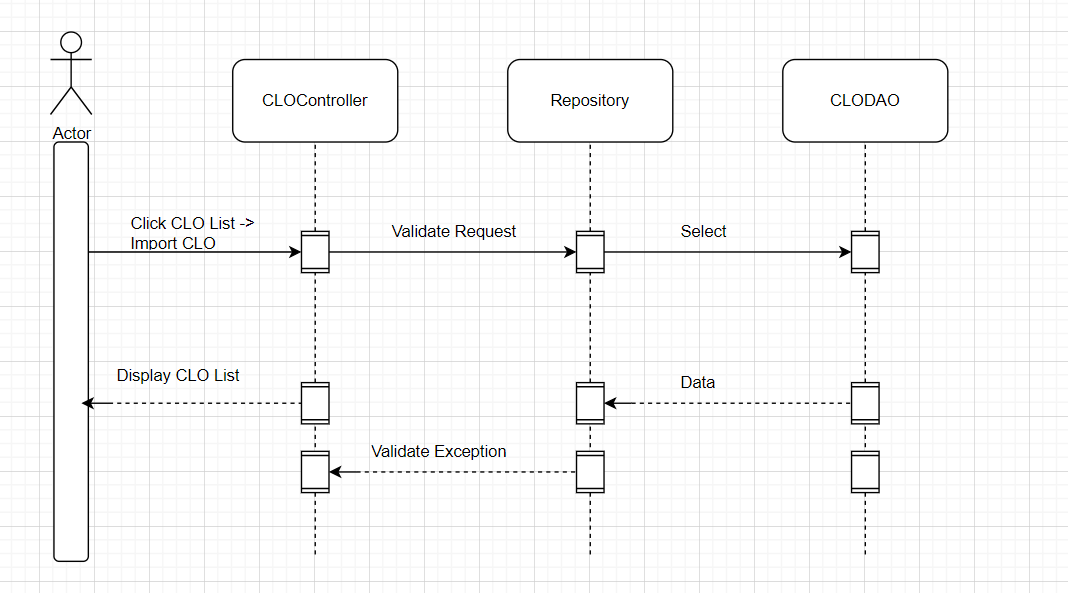
#### c. Database Queries

### 14.3 Import CLO

#### a.Class Diagram



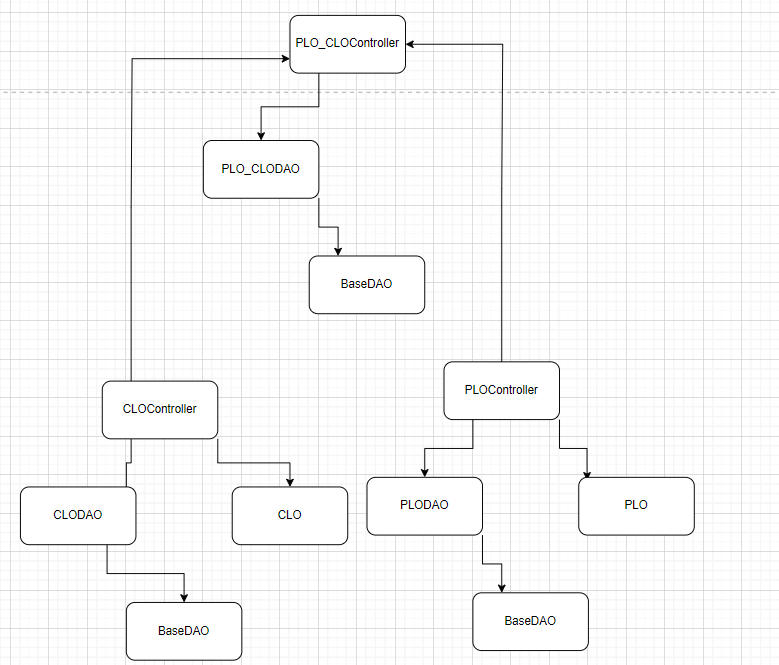
#### b. Sequence Diagram(s)



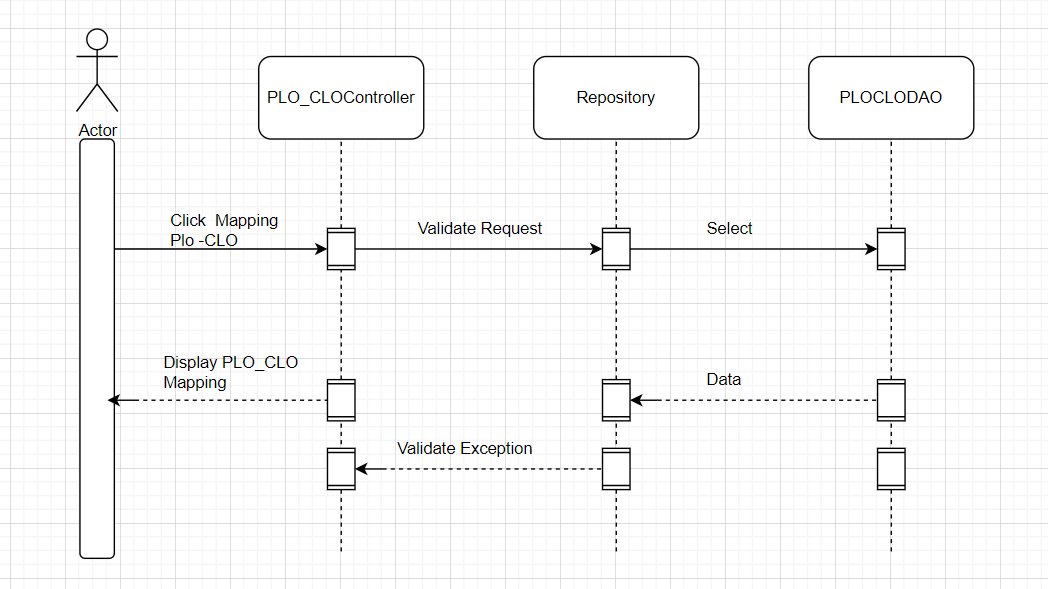
#### c. Database Queries

### 14.4 Mapping CLO-PLO

#### a.Class Diagram



#### b. Sequence Diagram(s)



#### c. Database Queries

## Assessment Management

### 15.1 View Assessment List

#### a.Class Diagram

#### b. Sequence Diagram(s)

#### c. Database Queries

### 15.2 View Assessment Details

#### a.Class Diagram

#### b. Sequence Diagram(s)

#### c. Database Queries

## Reference Material

### 16.1 View Material List

#### a.Class Diagram

#### b. Sequence Diagram(s)

#### c. Database Queries

### 16.2 View Material Detail

#### a.Class Diagram

#### b. Sequence Diagram(s)

#### c. Database Queries