

**QUIZLET PROJECT**

**Software Design Specification**

– Hanoi, February 2023 –

**Record of changeS**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | A\* M, D | In charge | Change Description |
| 13/Feb | A | PhuongNV | * Create UI of folder page * CRUD folder * CRUD study sets in folders * Modified database |
| 13/Feb | A | AnhTHV | * Create UI screen home page before login page * Create UI of class screen * CRUD class |
| 13/Feb | A | GiangNH | * Create UI of footer * Search by study set name * Answers of professors * Create UI of answers of professors |
| 13/Feb | M, A | VuBL | * Create login, register * CRUD study sets * Modified animation, UI * Share study sets |
| 13/Feb | M, A | HungBM | * Modified a verify, login and register * Modified filter * Create your library * Forgot password |
| 24/Feb | M,A | PhuongNV | * Send Email * Change Password * Share links join folder |
| 24/Feb | M,A | HungBm | * Filter * Add study set into folder, class * Modified class in library |
| 24/Feb | M,A | AnhTHV | * CRUD folder in classes * Share link join class * Invite member * Send email join class |
| 19/Mar | A,D,M | VuBL | * Learn study set * Test study set * Show status learning of card * Modified sql |

|  |  |  |  |
| --- | --- | --- | --- |
| Date | A\* M, D | In charge | Change Description |
| 13/Feb | A | PhuongNV | * Create UI of folder page * CRUD folder * CRUD study sets in folders * Modified database |
| 13/Feb | A | AnhTHV | * Create UI screen home page before login page * Create UI of class screen * CRUD class |
| 13/Feb | A | GiangNH | * Create UI of footer * Search by study set name * Answers of professors * Create UI of answers of professors |
| 13/Feb | M, A | VuBL | * Create login, register * CRUD study sets * Modified animation, UI * Share study sets |
| 13/Feb | M, A | HungBM | * Modified a verify, login and register * Modified filter * Create your library * Forgot password |

\*A - Added M - Modified D - Deleted

**Table of Contents**

[I. Overview 12](#_Toc130256730)

[1. Code Packages 12](#_Toc130256731)

[2. Database Design 13](#_Toc130256732)

[a. Database Schema 13](#_Toc130256733)

[b. Table Description 13](#_Toc130256734)

[II. Code Designs 14](#_Toc130256735)

[1. Register 14](#_Toc130256736)

[a. Class Diagram 14](#_Toc130256737)

[b. Class Specifications 14](#_Toc130256738)

[User Class 14](#_Toc130256739)

[Ultilities Class 15](#_Toc130256740)

[DAO Class 15](#_Toc130256741)

[Register Controller Class 15](#_Toc130256742)

[DBContext Class 15](#_Toc130256743)

[c. Sequence Diagram(s) 15](#_Toc130256744)

[d. Database Queries 16](#_Toc130256745)

[2. Login 16](#_Toc130256746)

[a. Class Diagram 16](#_Toc130256747)

[b. Class Specifications 17](#_Toc130256748)

[User Class 17](#_Toc130256749)

[DBContext Class 17](#_Toc130256750)

[Register Controller Class 18](#_Toc130256751)

[DAO Class 18](#_Toc130256752)

[c. Sequence Diagram(s) 18](#_Toc130256753)

[d. Database Queries 19](#_Toc130256754)

[*select \* from [User]* 19](#_Toc130256755)

[3. Logout 19](#_Toc130256756)

[a. Class Diagram 19](#_Toc130256757)

[b. Class Specifications 19](#_Toc130256758)

[User Class 20](#_Toc130256759)

[Register Controller Class 20](#_Toc130256760)

[c. Sequence Diagram(s) 20](#_Toc130256761)

[d. Database Queries 21](#_Toc130256762)

[4. Search study sets by name 21](#_Toc130256763)

[a. Class Diagram 21](#_Toc130256764)

[b. Class Specifications 22](#_Toc130256765)

[Study sets Class 22](#_Toc130256766)

[Search controller Class 23](#_Toc130256767)

[DAO Class 23](#_Toc130256768)

[DBContext Class 23](#_Toc130256769)

[c. Sequence Diagram(s) 23](#_Toc130256770)

[d. Database Queries 23](#_Toc130256771)

[*Select \* from StudySet* 23](#_Toc130256772)

[5. Verify code 24](#_Toc130256773)

[a. Class Diagram 24](#_Toc130256774)

[b. Class Specifications 24](#_Toc130256775)

[User Class 24](#_Toc130256776)

[DBContext Class 25](#_Toc130256777)

[Verify controller Class 25](#_Toc130256778)

[DAO Class 25](#_Toc130256779)

[c. Sequence Diagram(s) 25](#_Toc130256780)

[d. Database Queries 26](#_Toc130256781)

[6. FlashCard 26](#_Toc130256782)

[a. Class Diagram 26](#_Toc130256783)

[b. Class Specifications 27](#_Toc130256784)

[User Class 27](#_Toc130256785)

[DBContext Class 27](#_Toc130256786)

[FlashCrard controller Class 27](#_Toc130256787)

[DAO Class 27](#_Toc130256788)

[StudySet Class 27](#_Toc130256789)

[Card Class 28](#_Toc130256790)

[c. Sequence Diagram(s) 28](#_Toc130256791)

[d. Database Queries 28](#_Toc130256792)

[7.Create StudySet 29](#_Toc130256793)

[a. Class Diagram 29](#_Toc130256794)

[b. Class Specifications 30](#_Toc130256795)

[User Class 30](#_Toc130256796)

[DBContext Class 30](#_Toc130256797)

[CreateSet controller Class 31](#_Toc130256798)

[DAO Class 31](#_Toc130256799)

[StudySet Class 31](#_Toc130256800)

[Card Class 31](#_Toc130256801)

[c. Sequence Diagram(s) 31](#_Toc130256802)

[d. Database Queries 32](#_Toc130256803)

[8.Delete StudySet 33](#_Toc130256804)

[a. Class Diagram 33](#_Toc130256805)

[b. Class Specifications 33](#_Toc130256806)

[User Class 33](#_Toc130256807)

[DBContext Class 34](#_Toc130256808)

[Delete controller Class 34](#_Toc130256809)

[DAO Class 34](#_Toc130256810)

[StudySet Class 34](#_Toc130256811)

[Card Class 34](#_Toc130256812)

[c. Sequence Diagram(s) 34](#_Toc130256813)

[d. Database Queries 35](#_Toc130256814)

[9.Add Studied Cards 35](#_Toc130256815)

[a. Class Diagram 35](#_Toc130256816)

[b. Class Specifications 36](#_Toc130256817)

[User Class 36](#_Toc130256818)

[DBContext Class 36](#_Toc130256819)

[AddStudiedCard controller Class 36](#_Toc130256820)

[DAO Class 37](#_Toc130256821)

[StudySet Class 37](#_Toc130256822)

[Card Class 37](#_Toc130256823)

[c. Sequence Diagram(s) 37](#_Toc130256824)

[d. Database Queries 38](#_Toc130256825)

[10.Forget password 38](#_Toc130256826)

[a. Class Diagram 38](#_Toc130256827)

[b. Class Specifications 39](#_Toc130256828)

[User Class 39](#_Toc130256829)

[DBContext Class 40](#_Toc130256830)

[FogetPass controller Class 40](#_Toc130256831)

[DAO Class 40](#_Toc130256832)

[SendEmail Class 40](#_Toc130256833)

[c. Sequence Diagram(s) 40](#_Toc130256834)

[d. Database Queries 40](#_Toc130256835)

[11.Change password 41](#_Toc130256836)

[a. Class Diagram 41](#_Toc130256837)

[b. Class Specifications 41](#_Toc130256838)

[User Class 41](#_Toc130256839)

[DBContext Class 42](#_Toc130256840)

[ChangePass controller Class 42](#_Toc130256841)

[DAO Class 42](#_Toc130256842)

[c. Sequence Diagram(s) 42](#_Toc130256843)

[d. Database Queries 42](#_Toc130256844)

[12.Update studyset 43](#_Toc130256845)

[a. Class Diagram 43](#_Toc130256846)

[b. Class Specifications 43](#_Toc130256847)

[User Class 43](#_Toc130256848)

[DBContext Class 44](#_Toc130256849)

[Update controller Class 44](#_Toc130256850)

[DAO Class 44](#_Toc130256851)

[StudySet Class 44](#_Toc130256852)

[Card Class 44](#_Toc130256853)

[c. Sequence Diagram(s) 45](#_Toc130256854)

[d. Database Queries 45](#_Toc130256855)

[13.Home 45](#_Toc130256856)

[a. Class Diagram 45](#_Toc130256857)

[b. Class Specifications 46](#_Toc130256858)

[User Class 46](#_Toc130256859)

[DBContext Class 46](#_Toc130256860)

[Home controller Class 47](#_Toc130256861)

[DAO Class 47](#_Toc130256862)

[StudySet Class 47](#_Toc130256863)

[Folder Class 47](#_Toc130256864)

[Class Class 47](#_Toc130256865)

[c. Sequence Diagram(s) 48](#_Toc130256866)

[d. Database Queries 48](#_Toc130256867)

[14.Class set 49](#_Toc130256868)

[a. Class Diagram 49](#_Toc130256869)

[b. Class Specifications 49](#_Toc130256870)

[User Class 49](#_Toc130256871)

[DBContext Class 50](#_Toc130256872)

[ClassSet controller Class 50](#_Toc130256873)

[DAO Class 50](#_Toc130256874)

[StudySet Class 50](#_Toc130256875)

[Class Class 51](#_Toc130256876)

[c. Sequence Diagram(s) 51](#_Toc130256877)

[d. Database Queries 51](#_Toc130256878)

[15.Folder set 52](#_Toc130256879)

[a. Class Diagram 52](#_Toc130256880)

[b. Class Specifications 52](#_Toc130256881)

[User Class 52](#_Toc130256882)

[DBContext Class 53](#_Toc130256883)

[FolderSet controller Class 53](#_Toc130256884)

[DAO Class 53](#_Toc130256885)

[StudySet Class 53](#_Toc130256886)

[Folder Class 53](#_Toc130256887)

[c. Sequence Diagram(s) 54](#_Toc130256888)

[d. Database Queries 54](#_Toc130256889)

[16.Study set list (wait) 54](#_Toc130256890)

[a. Class Diagram 54](#_Toc130256891)

[b. Class Specifications 55](#_Toc130256892)

[User Class 55](#_Toc130256893)

[DBContext Class 55](#_Toc130256894)

[Verify controller Class 56](#_Toc130256895)

[DAO Class 56](#_Toc130256896)

[StudySet Class 56](#_Toc130256897)

[Folder Class 56](#_Toc130256898)

[Class Class 56](#_Toc130256899)

[Card Class 57](#_Toc130256900)

[c. Sequence Diagram(s) 57](#_Toc130256901)

[d. Database Queries 58](#_Toc130256902)

[17.Controller folder set 58](#_Toc130256903)

[a. Class Diagram 58](#_Toc130256904)

[b. Class Specifications 59](#_Toc130256905)

[User Class 59](#_Toc130256906)

[DBContext Class 59](#_Toc130256907)

[ControllerFolderSet controller Class 60](#_Toc130256908)

[DAO Class 60](#_Toc130256909)

[StudySet Class 60](#_Toc130256910)

[Folder Class 60](#_Toc130256911)

[c. Sequence Diagram(s) 61](#_Toc130256912)

[d. Database Queries 61](#_Toc130256913)

[18.Controller class set 62](#_Toc130256914)

[a. Class Diagram 62](#_Toc130256915)

[b. Class Specifications 62](#_Toc130256916)

[User Class 62](#_Toc130256917)

[DBContext Class 63](#_Toc130256918)

[Controller Class Set Class 63](#_Toc130256919)

[StudySet Class 63](#_Toc130256920)

[Class Class 63](#_Toc130256921)

[c. Sequence Diagram(s) 64](#_Toc130256922)

[d. Database Queries 64](#_Toc130256923)

[19. Create class 65](#_Toc130256924)

[a. Class Diagram 65](#_Toc130256925)

[b. Class Specifications 66](#_Toc130256926)

[User Class 66](#_Toc130256927)

[DBContext Class 66](#_Toc130256928)

[CreateClass controller Class 66](#_Toc130256929)

[DAO Class 66](#_Toc130256930)

[Class Class 66](#_Toc130256931)

[c. Sequence Diagram(s) 67](#_Toc130256932)

[d. Database Queries 67](#_Toc130256933)

[20. Create folder 68](#_Toc130256934)

[a. Class Diagram 68](#_Toc130256935)

[b. Class Specifications 69](#_Toc130256936)

[User Class 69](#_Toc130256937)

[DBContext Class 70](#_Toc130256938)

[CreateFolder controller Class 70](#_Toc130256939)

[DAO Class 70](#_Toc130256940)

[Folder Class 70](#_Toc130256941)

[c. Sequence Diagram(s) 70](#_Toc130256942)

[d. Database Queries 71](#_Toc130256943)

[21. Reset studied Cards 71](#_Toc130256944)

[a. Class Diagram 71](#_Toc130256945)

[b. Class Specifications 72](#_Toc130256946)

[User Class 72](#_Toc130256947)

[DBContext Class 72](#_Toc130256948)

[ResetStudiedCard controller Class 73](#_Toc130256949)

[DAO Class 73](#_Toc130256950)

[StudySet Class 73](#_Toc130256951)

[Card Class 73](#_Toc130256952)

[c. Sequence Diagram(s) 73](#_Toc130256953)

[d. Database Queries 74](#_Toc130256954)

[22. Send email 74](#_Toc130256955)

[a. Class Diagram 74](#_Toc130256956)

[b. Class Specifications 75](#_Toc130256957)

[User Class 75](#_Toc130256958)

[DBContext Class 76](#_Toc130256959)

[SendEmail Class 76](#_Toc130256960)

[DAO Class 76](#_Toc130256961)

[c. Sequence Diagram(s) 76](#_Toc130256962)

[d. Database Queries 77](#_Toc130256963)

[23. Study 77](#_Toc130256964)

[a. Class Diagram 77](#_Toc130256965)

[b. Class Specifications 78](#_Toc130256966)

[User Class 78](#_Toc130256967)

[DBContext Class 78](#_Toc130256968)

[Study controller Class 79](#_Toc130256969)

[DAO Class 79](#_Toc130256970)

[StudySet Class 79](#_Toc130256971)

[Card Class 79](#_Toc130256972)

[c. Sequence Diagram(s) 80](#_Toc130256973)

[d. Database Queries 80](#_Toc130256974)

[24. Test 81](#_Toc130256975)

[a. Class Diagram 81](#_Toc130256976)

[b. Class Specifications 81](#_Toc130256977)

[User Class 81](#_Toc130256978)

[DBContext Class 82](#_Toc130256979)

[Test controller Class 82](#_Toc130256980)

[DAO Class 82](#_Toc130256981)

[StudySet Class 82](#_Toc130256982)

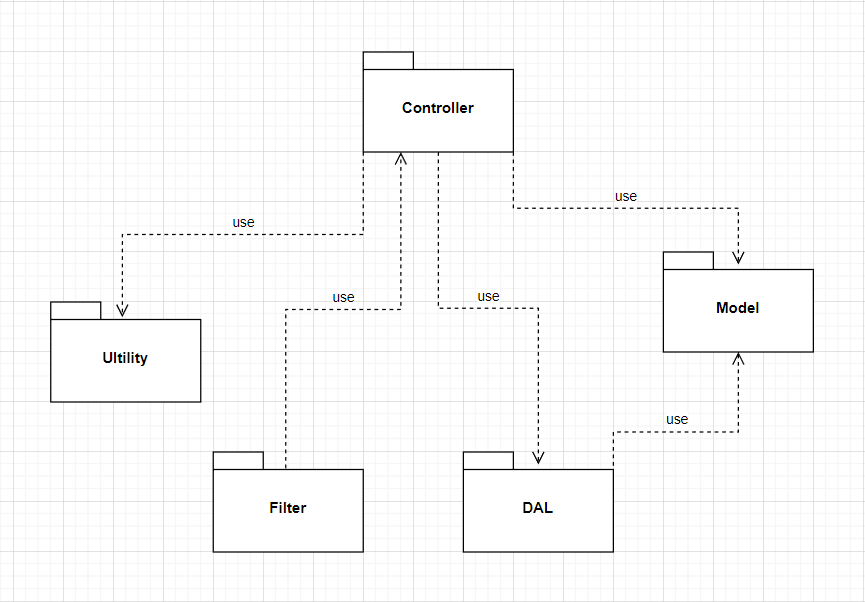
[Card Class 82](#_Toc130256983)

[c. Sequence Diagram(s) 83](#_Toc130256984)

[d. Database Queries 83](#_Toc130256985)

# I. Overview

## 1. Code Packages

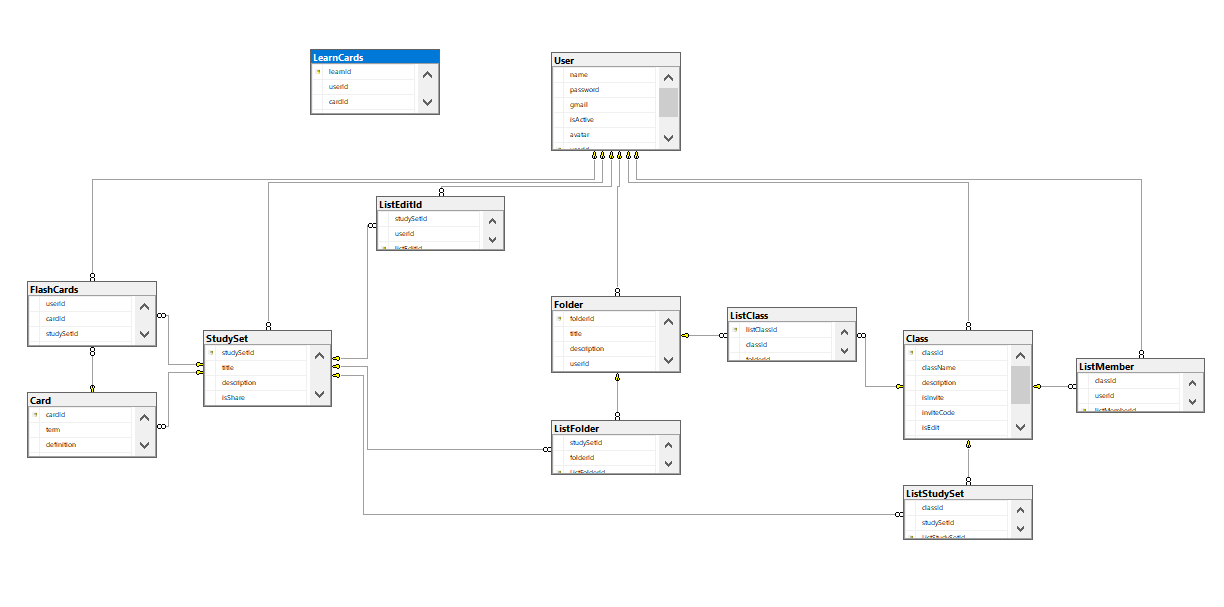


***Package descriptions***

|  |  |  |
| --- | --- | --- |
| **No** | **Package** | **Description** |
| *01* | *Controller* | *Controller package included Servlet file* |
| *02* | *Model* | *Model package support represent entities in database* |
| *03* | *DAL* | *DAL package included DBcontext and DAO* |
| *04* | *Ultility* | *Ultility package included some regex and verified file* |
| *05* | *Filter* | *Filter package included filter file to decentralize user and guest* |

## 2. Database Design

### a. Database Schema



### b. Table Description

|  |  |  |
| --- | --- | --- |
| **#** | **Entity** | **Description** |
| 1 | User | All information of users |
| 2 | Card | All information of list cards |
| 3 | Study Set | All information of list study sets |
| 4 | Class | All information of list class |
| 5 | Folder | All information of list folder |
| 6 | List Folder | A folder contain study set and a study set can be in many different folder |
| 7 | List study set | A class contain study sets and a study set can be in many different class |
| 8 | List member | A class contain user and an user can join many class |
| 9 | List edit id | A user has many study sets |
| 10 | FlashCards | All information of cards in study set, which user were studied |
| 11 | ListClass | A class contain many folders and a folder can be in many different class |
| 12 | LearnCards | Cards were studied |

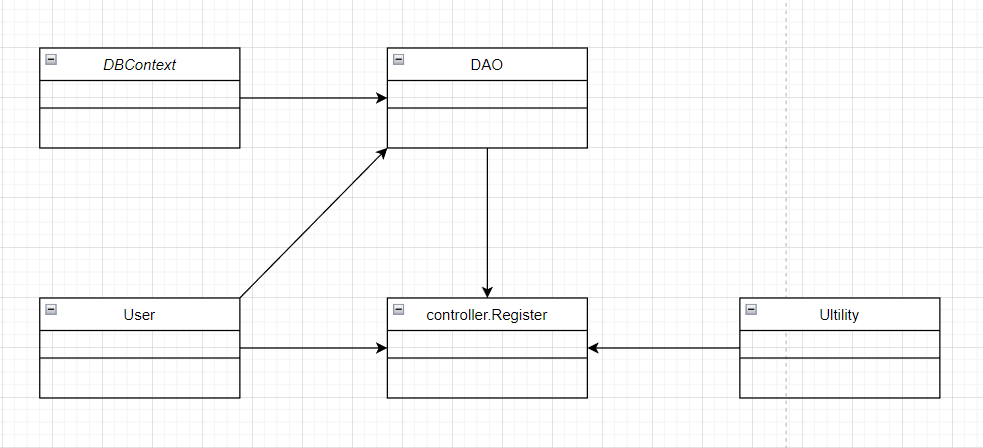
# 

# 

# II. Code Designs

## 1. Register

### a. Class Diagram

**

### b. Class Specifications

#### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### Ultilities Class

***Class Methods***

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *isBlank(string u){}* | *Check string is blank* |
| *02* | *checkExistEmail(string email){}* | *Check existed email* |
| *03* | *ValidInputRequire(string e, string p, string u){}* | *Check valid input* |
| *04* | *validEmail(string e){}* | *Check valid email* |
| *05* | *validUsername(string u){}* | *Check valid username* |
| *06* | *validPassword(string p){}* | *Check valid password* |

#### DAO Class

***Class Methods***

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *register(user u){}* | *Insert new user to database* |
|  |  |  |

#### Register Controller Class

***Class Methods***

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

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Doget()* | *Sendredirect to register.jsp* |
| *02* | *doPost()* | *Check input and register* |

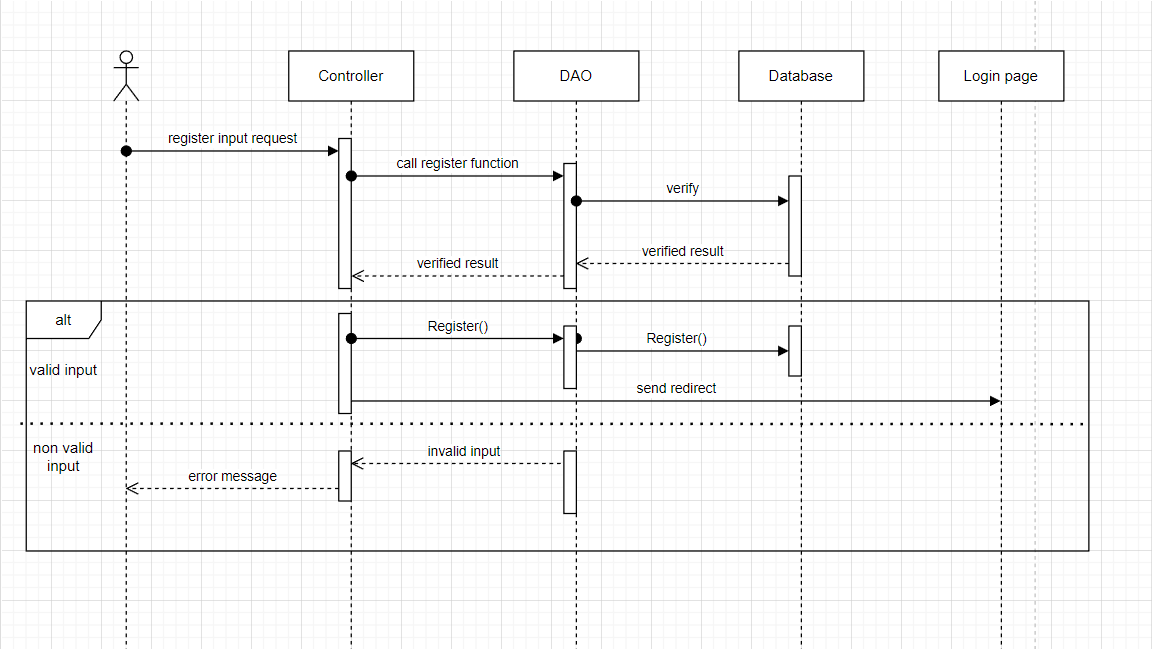
#### DBContext Class

***Class Methods***

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

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "INSERT INTO [dbo].[User]\n"*

*+ " ([name]\n"*

*+ " ,[password]\n"*

*+ " ,[gmail]\n"*

*+ " ,[isActive]\n"*

*+ " ,[avatar]\n"*

*+ " ,[language])\n"*

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

## 

## 

## 

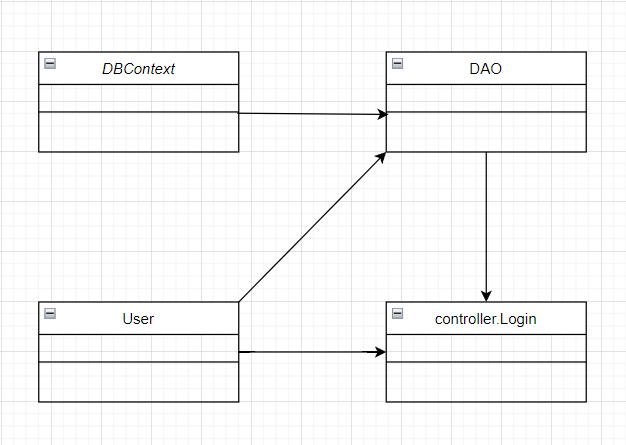
## 

## 

## 

## 2. Login

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

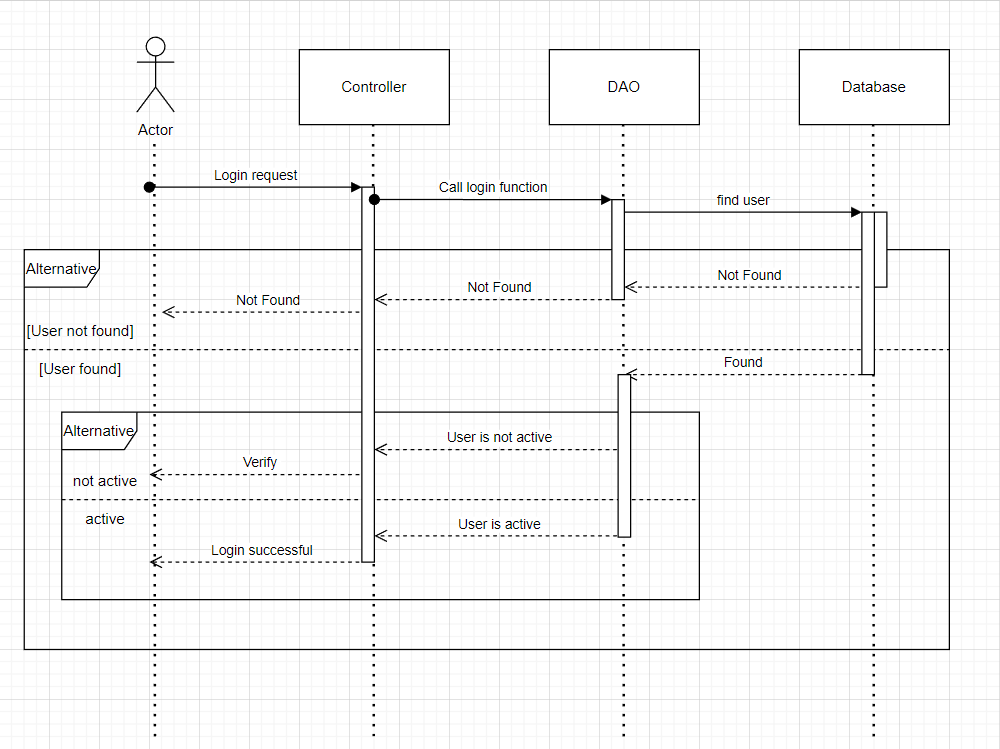
#### Register Controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Doget()* | *Get session and Send redirect to login.jsp* |
| *02* | *doPost()* | *Check input and login* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *checkUser(username, password)* | *Check user have username, email, pass to login* |
| *02* | *GetAllUser()* | *Get all user in database* |

### c. Sequence Diagram(s)

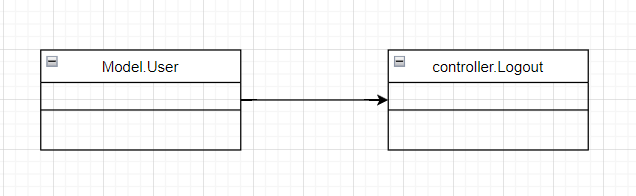


### d. Database Queries

## *select \* from [User]*

## 3. Logout

### a. Class Diagram

**

### b. Class Specifications

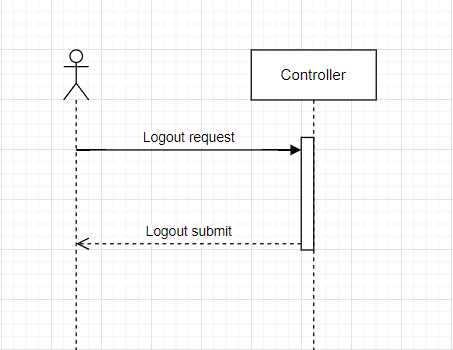
### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### Register Controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Doget()* | *Remove session to logout* |

### c. Sequence Diagram(s)

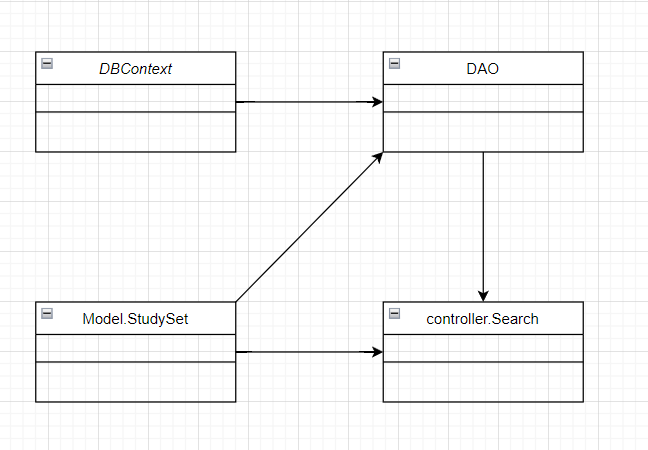


### d. Database Queries

*Logout function do not use database queries.*

## 4. Search study sets by name

### a. Class Diagram

**

### b. Class Specifications

#### Study sets Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet(){}* | *Default constructor* |
| *02* | *StudySet(….){}* | *Parameterized constructor* |
| *03* | *getId(){}* | *Getter of id* |
| *04* | *setId(int id){}* | *Setter of id* |
| *05* | *getTitle(){}* | *Getter of title* |
| *06* | *setTitle(string title){]* | *Setter of title* |
| *07* | *getDescription(){}* | *Getter of description* |
| *08* | *setDescription(string des){}* | *Setter of description* |
| *09* | *isIsShare(){}* | *Getter of isShare* |
| *10* | *setIsShare(Boolean isShare){]* | *Setter of isShare* |
| *11* | *getuserId(){}* | *Getter of user id* |
| *12* | *setUserid(int id){]* | *Setter of user id* |
| *13* | *getnumberCard(){}* | *Count number of cards in a study set* |
| *14* | *getAuthor(){}* | *Get name of user create study sets* |

#### Search controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Search by name in list study sets* |
|  |  |  |

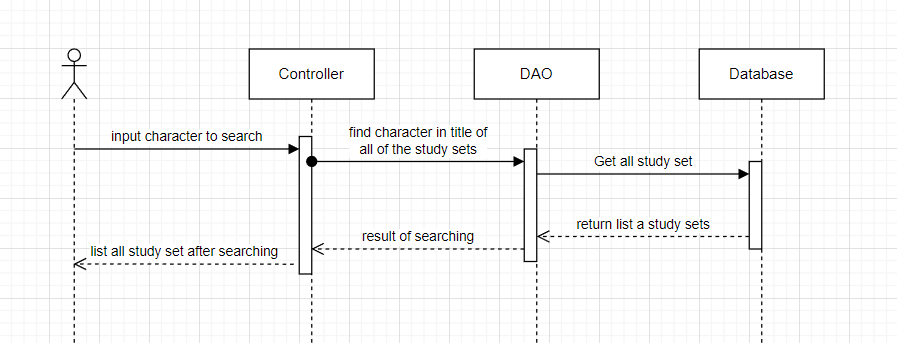
#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getAllStudySet(){}* | *Get all study sets in database* |
|  |  |  |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

### c. Sequence Diagram(s)

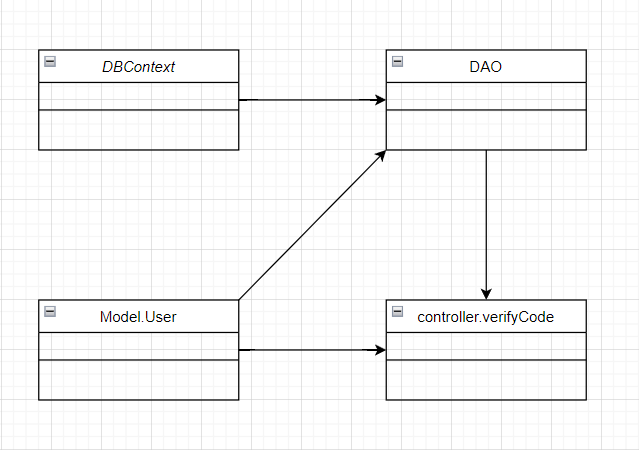


### d. Database Queries

# *Select \* from StudySet*

## 5. Verify code

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

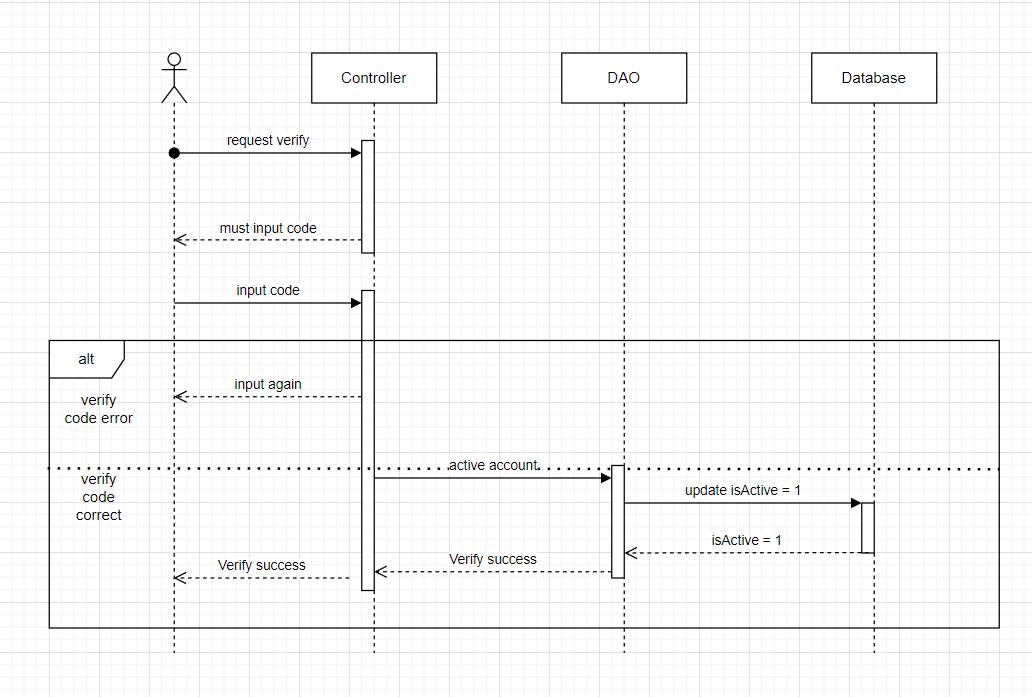
#### Verify controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Send redirect to verify.jsp* |
| *02* | *doPost(){}* | *Active account by verify code* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Active account(string email){}* | *Update is active in database* |

### c. Sequence Diagram(s)

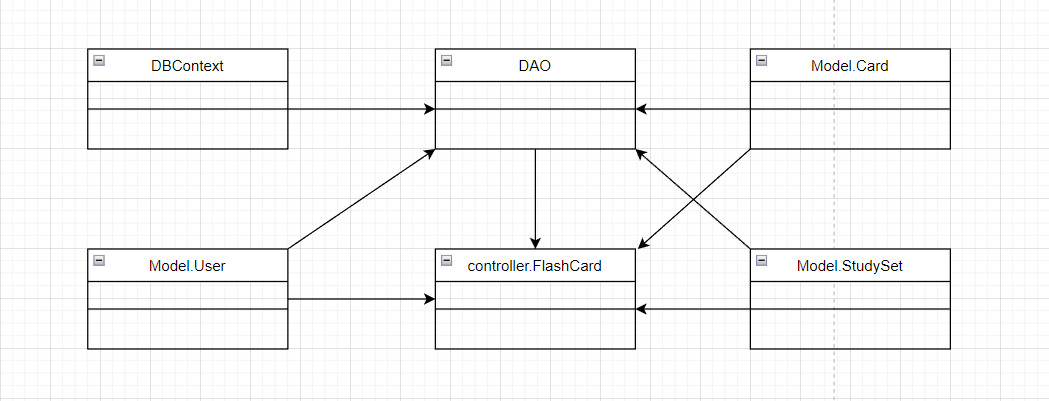


### d. Database Queries

*String sql = "update [User] set isActive = 1 where gmail like ?";*

## 6. FlashCard

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### FlashCrard controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Call function get all card in study sets* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getStudySetById(int id){}* | *Get study sets by its id* |
| *02* | *getUserByUserId(int id){}* | *Get user by id* |
| *03* | *getAllCardInSet(int id){}* | *Get all card in study set* |
| *04* | *getStudiedCard(int setId, int uId)* | *Get all studied cards* |

#### StudySet Class

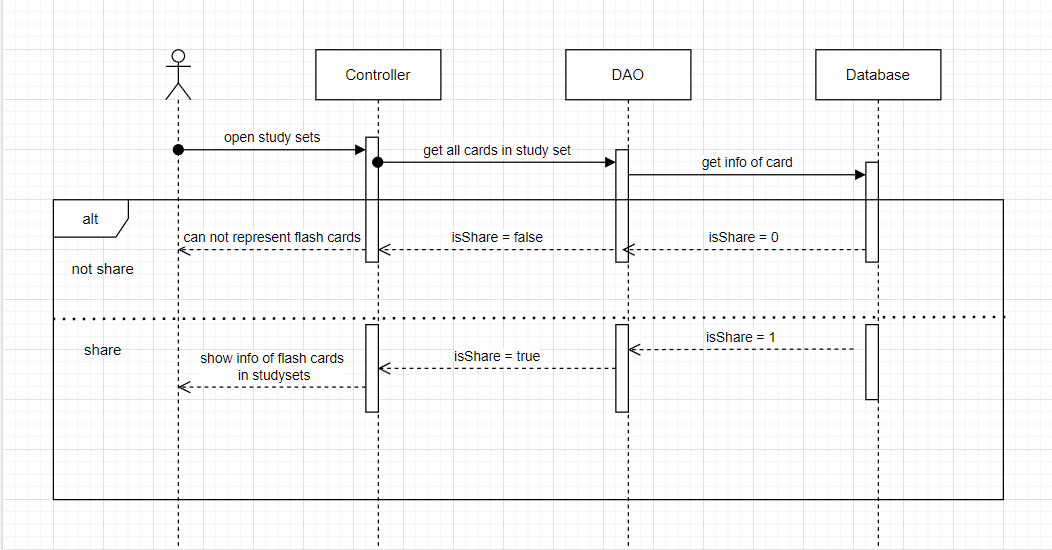
|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

### Card Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "SELECT [studySetId]\n"*

*+ " ,[title]\n"*

*+ " ,[description]\n"*

*+ " ,[isShare]\n"*

*+ " ,[userId]\n"*

*+ " FROM [dbo].[StudySet]\n"*

*+ " WHERE studySetId=?";*

*String sql = "SELECT [cardId]\n"*

*+ " ,[term]\n"*

*+ " ,[definition]\n"*

*+ " ,[studySetId]\n"*

*+ " FROM [dbo].[Card]\n"*

*+ " WHERE studySetID=?";*

*String sql = "SELECT [userId]\n"*

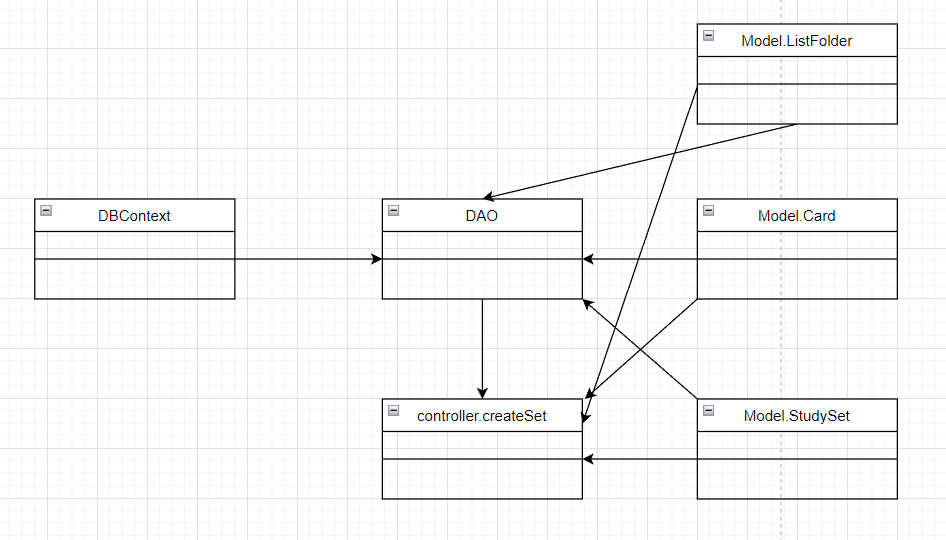
*+ " ,[cardId]\n"*

*+ " FROM [dbo].[FlashCards]"*

*+ " WHERE studySetId = ? and userId = ?";*

## 7.Create StudySet

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### CreateSet controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Fill info to card* |
| *02* | *doPost(){}* | *Add study set and card in set* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *addStuySet(StudySet set){}* | *Add new study set* |
| *02* | *addCard(Card c){]* | *Add card to study set* |
| *03* | *createListFolder(listF){}* | *Add study set to a folder* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

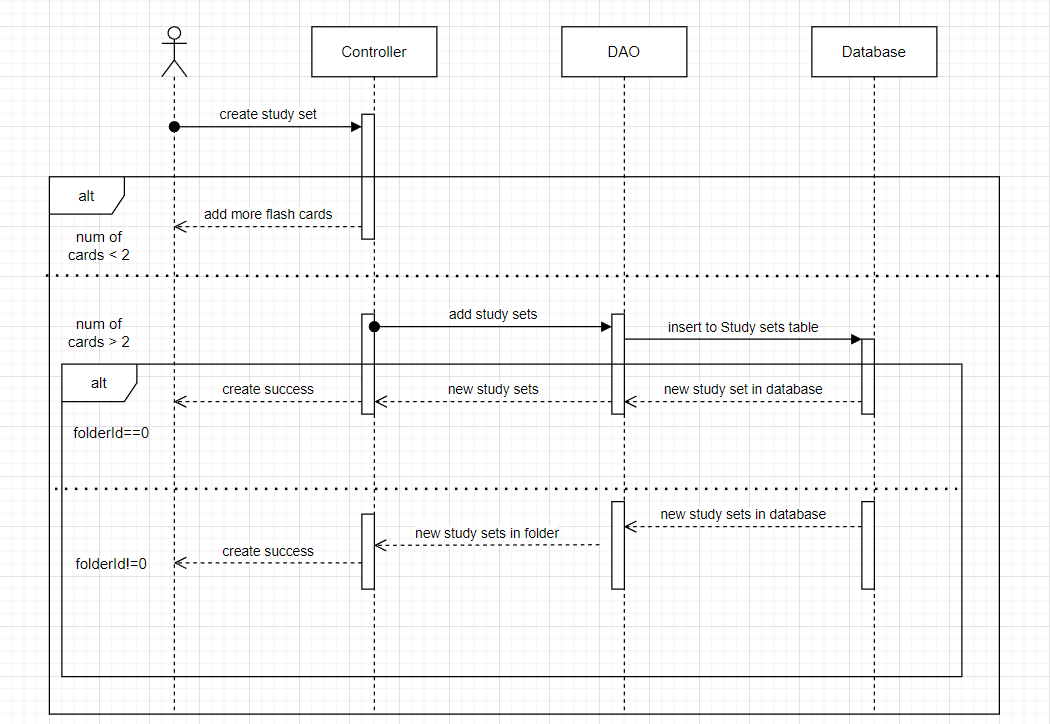
### 

### Card Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "INSERT INTO [dbo].[StudySet]\n"*

*+ " ([title]\n"*

*+ " ,[description]\n"*

*+ " ,[isShare]\n"*

*+ " ,[userId])\n"*

*+ " VALUES\n"*

*+ " (?\n"*

*+ " ,?\n"*

*+ " ,?\n"*

*+ " ,?)";*

*String sql = "INSERT INTO [dbo].[ListFolder]\n"*

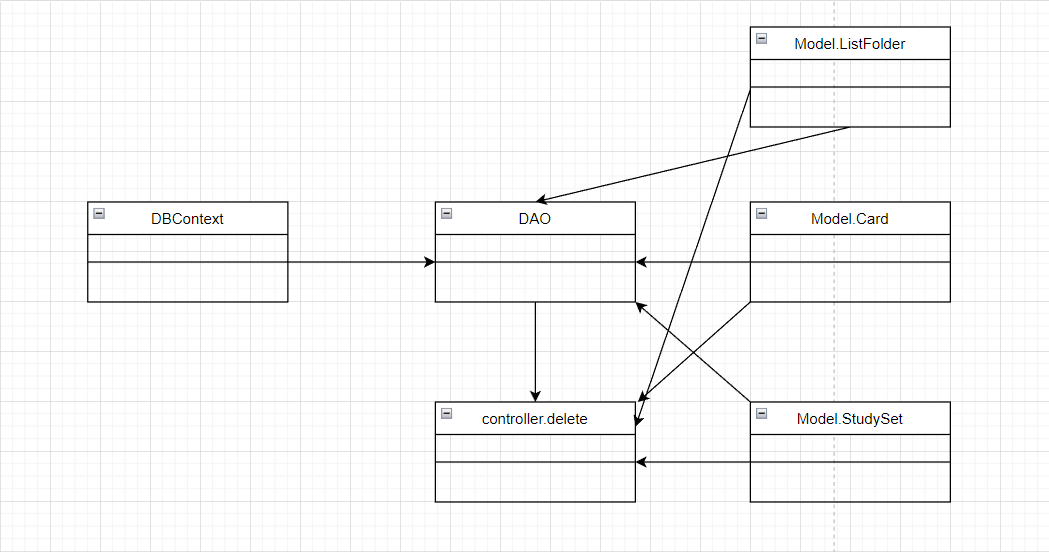
*+ " ([studySetId]\n"*

*+ " ,[folderId])\n"*

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

## 8.Delete StudySet

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### Delete controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Delete card in study set, study set, and delete study set in folder* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *deleteAllCards(int id){}* | *Delete all cards in study sets* |
| *02* | *deleteStudySet(int id){}* | *Delete study set* |
| *03* | *deleteInListFolder(int id){]* | *Delete study set in folders* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

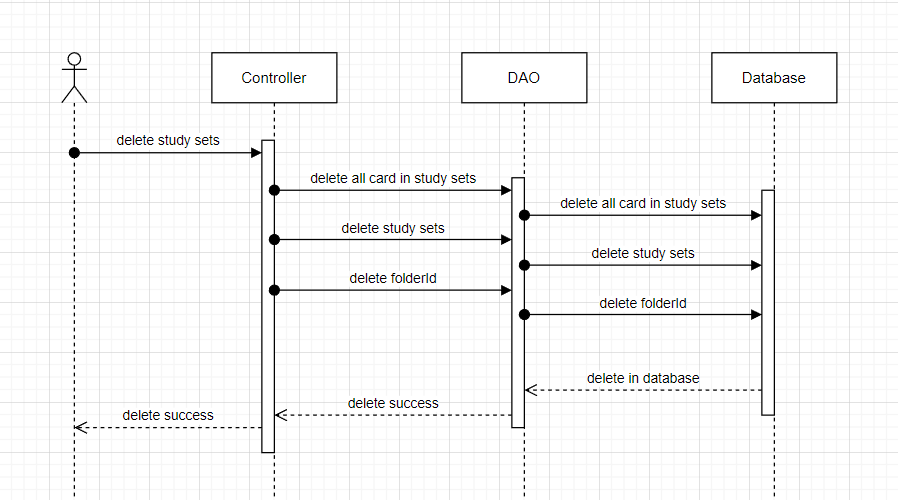
### 

### Card Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "DELETE FROM [dbo].[ListFolder]\n"*

*+ " WHERE studySetId = ?";*

*String sql = "DELETE FROM [dbo].[Card]\n"*

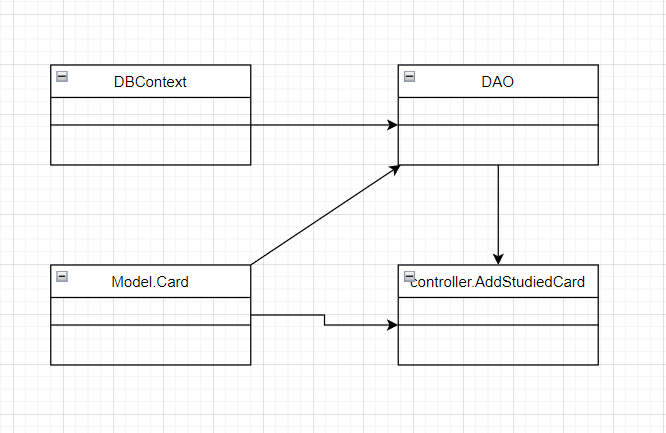
*+ " WHERE studySetId=?";*

*String sql = "DELETE FROM [dbo].[StudySet]\n"*

*+ " WHERE studySetId=?";*

## 9.Add Studied Cards

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### AddStudiedCard controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Add all studied card to a list* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getAllCardinSet(int id){}* | *Get all card in study set* |
| *02* | *getListStudiedCardId(int sid, int uid){}* | *Get list all studied card id* |
| *03* | *addStudiedCard(int uid, int id, studySetId){}* | *Add all studied card to a list* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

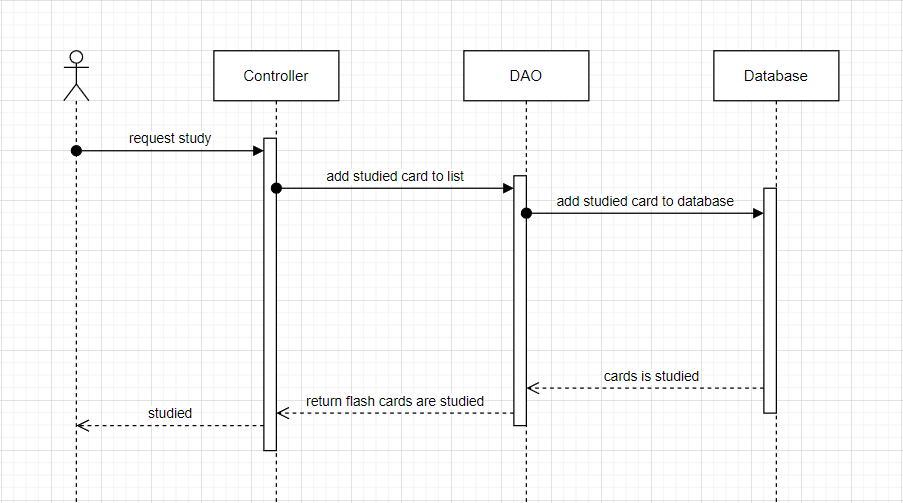
### 

### Card Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "SELECT [cardId]\n"*

*+ " ,[term]\n"*

*+ " ,[definition]\n"*

*+ " ,[studySetId]\n"*

*+ " FROM [dbo].[Card]\n"*

*+ " WHERE studySetID=?";*

*String sql = "SELECT [userId]\n"*

*+ " ,[cardId]\n"*

*+ " FROM [dbo].[FlashCards]"*

*+ " WHERE studySetId = ? and userId = ?";*

*String sql = "INSERT INTO [dbo].[FlashCards]\n"*

*+ " ([userId]\n"*

*+ " ,[cardId]\n"*

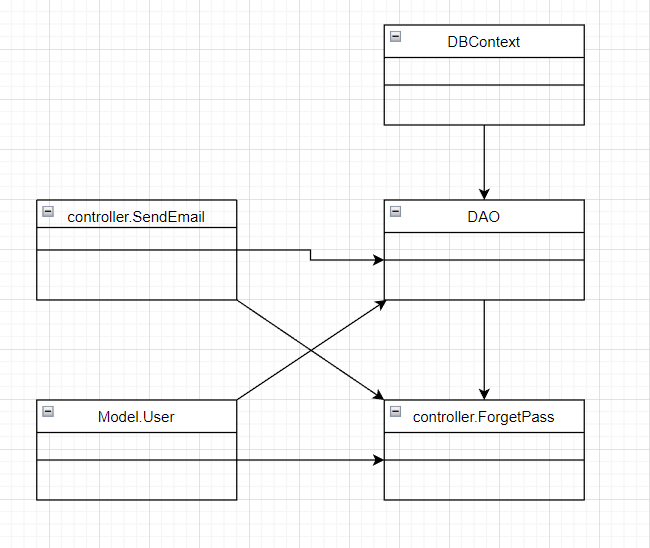
*+ " ,[studySetId])\n"*

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

## 

## 10.Forget password

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### FogetPass controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doPost(){}* | *Send link change password to email of user* |

#### DAO Class

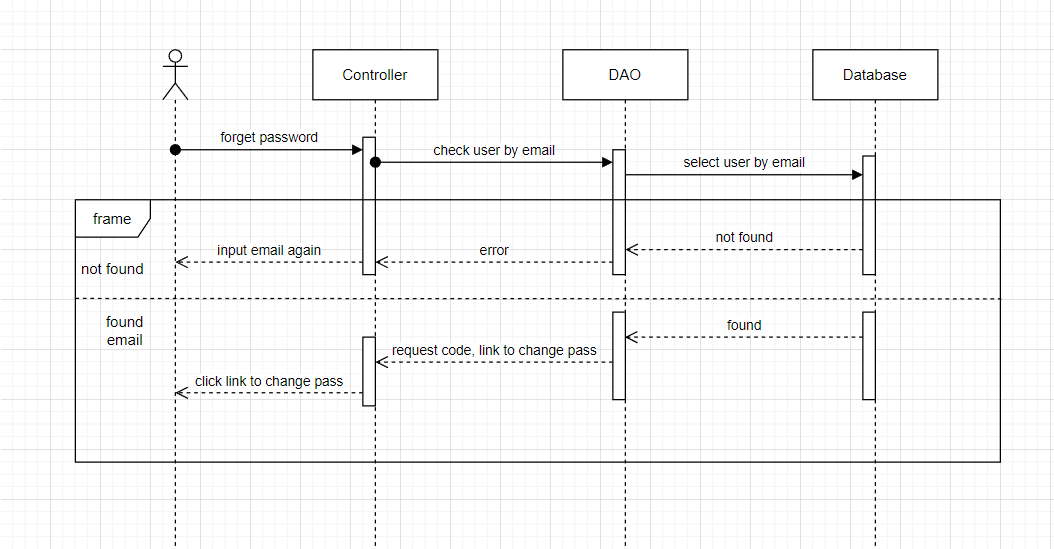
|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *checkUser(string email){}* | *Check email in database* |

#### SendEmail Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *SendEmailPass(){}* | *Send link to change password to user* |

### 

### c. Sequence Diagram(s)

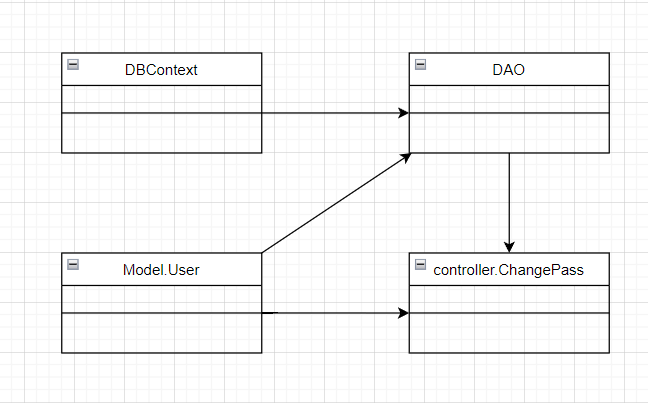


### d. Database Queries

*String sql = "select \* from [User] where gmail =? ";*

## 11.Change password

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### ChangePass controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Check code verify when send email* |
| *02* | *doPost(){}* | *Change password* |

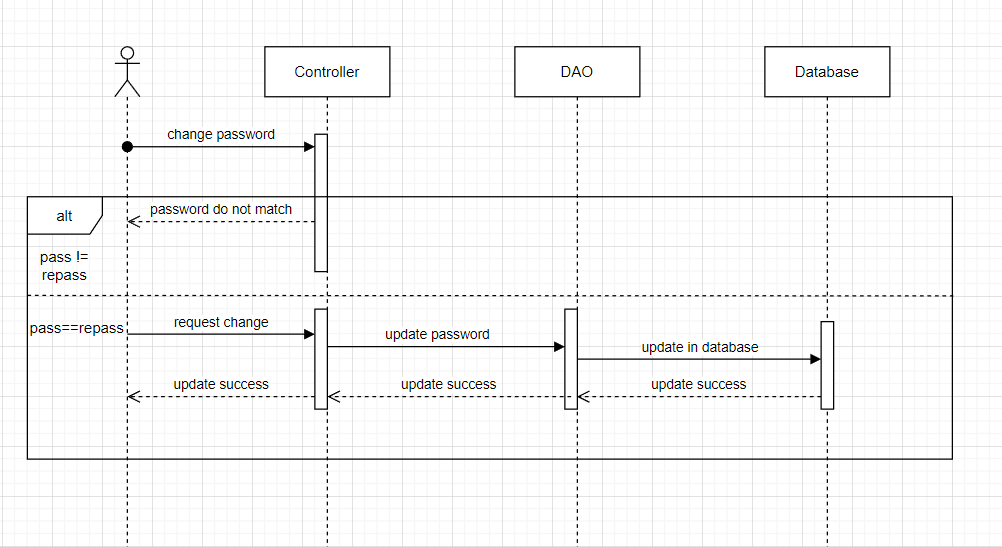
#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getIdByEmail(string email){}* | *Get id of user by email* |
| *02* | *updatePass(string pass, int uId){}* | *Update new password* |

### 

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "UPDATE [User]\n"*

*+ "SET\n"*

*+ "\n"*

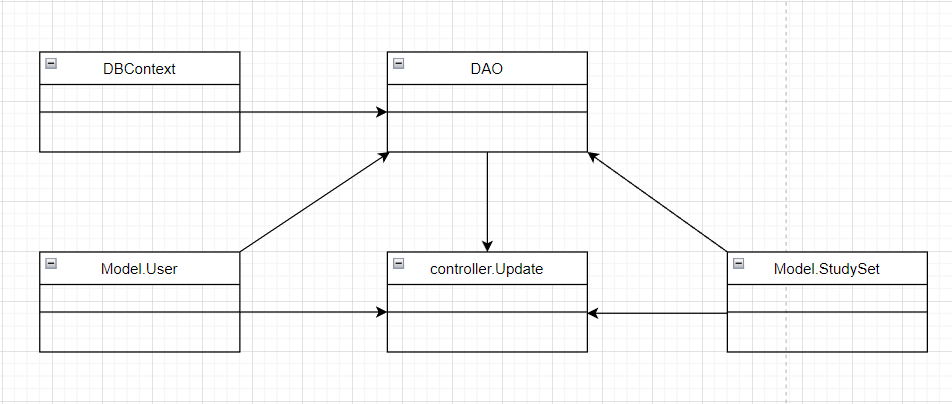
*+ "password = ?\n"*

*+ "\n"*

*+ "WHERE userId = ?";*

## 12.Update studyset

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### Update controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Get information of all card in study set to update* |
| *02* | *doPost(){}* | *Update or add new cards in study set* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getStudySetByID(int id){}* | *Get study set by study set id* |
| *02* | *getAllCardInSet(int id){}* | *Get all cards in study set* |
| *03* | *updateStudySet(studyset s){}* | *Update study set* |
| *04* | *updateCard(Card c){}* | *Update cards in study set* |
| *05* | *addCard(Card c){}* | *Add more cards in study sets* |
|  |  |  |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

### 

### 

### Card Class

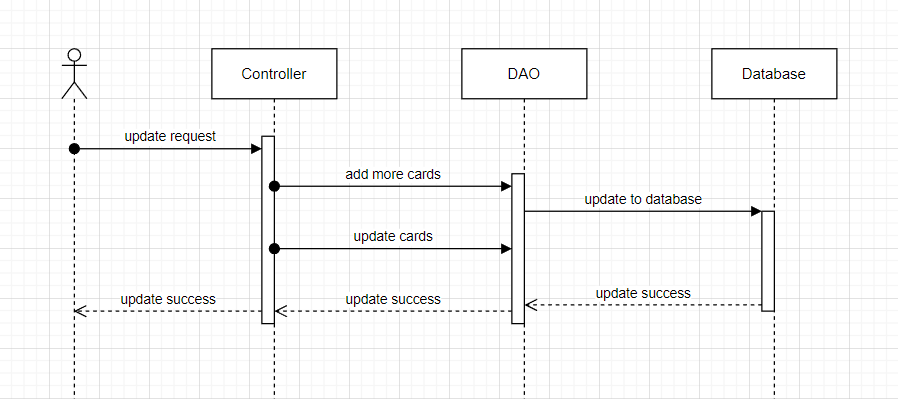
|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### 

### 

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "UPDATE [dbo].[StudySet]\n"*

*+ " SET [title] = ?\n"*

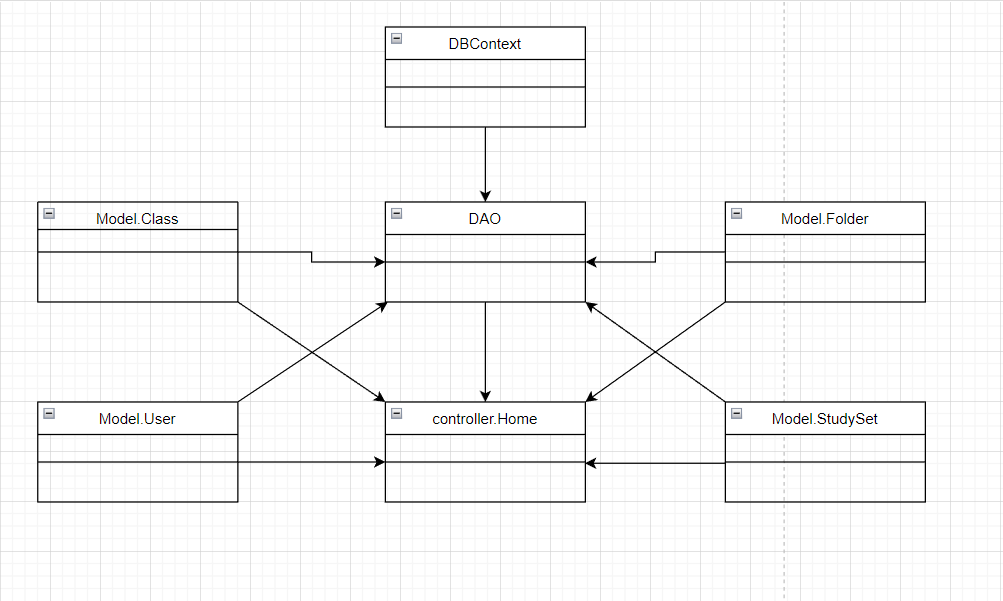
*+ " ,[description] = ?\n"*

*+ " ,[isShare] = ?\n"*

*+ " WHERE studySetId=?";*

## 13.Home

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### Home controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *List all studySet and your library* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getAllStudySet(string email){}* | *Get all study set* |
| *02* | *getFiveStudySet(int id){}* | *Get top 5 study set order by id* |
| *03* | *getUserByUserID(int id){}* | *Get user by user id* |
| *04* | *getTopFiveFolder(int id){}* | *Get top 5 folder order by id* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

### 

### 

### Folder Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Folder() {)* | *Default Constructor* |
| *02* | *Folder(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string term){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Definition){}* | *setter of Description* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |

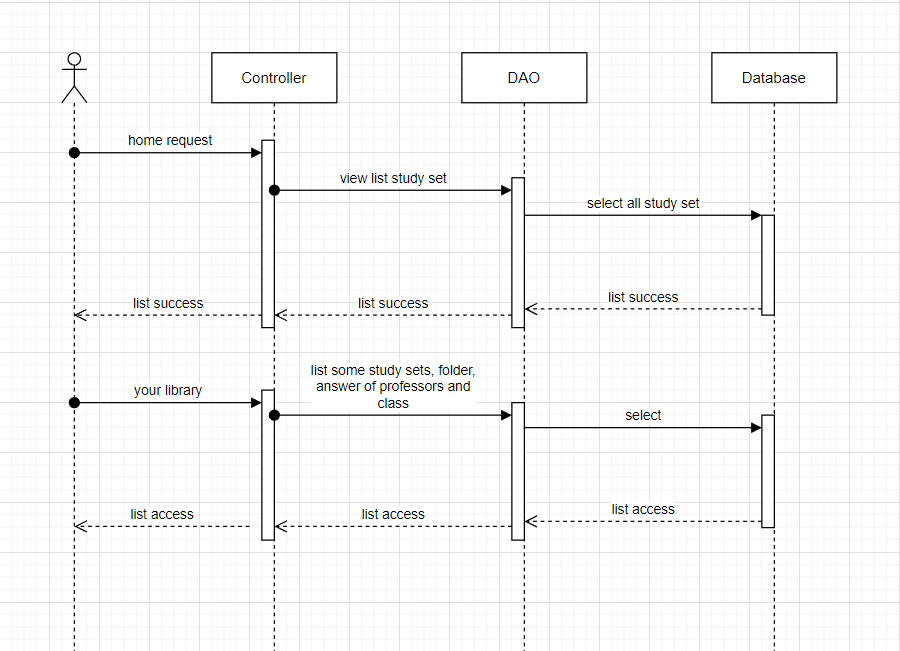
### 

### Class Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Class() {)* | *Default Constructor* |
| *02* | *Class(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of Class Id* |
| *04* | *setId(int id){}* | *setter of Class Id* |
| *05* | *getName(){}* | *getter of Class Name* |
| *06* | *setName(string name){}* | *setter of Class Name* |
| *07* | *getDesc(){}* | *getter of Desc* |
| *08* | *setDesc(string Desc){}* | *setter of Desc* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |
| *11* | *isIsInvite() {}* | *getter of isInvite* |
| *12* | *setIsInvite(boolean isInvite) { }* | *setter of isInvite* |
| *13* | *isIsEdit() {}* | *getter of isEdit* |
| *14* | *setIsEdit(boolean isEdit) { }* | *setter of isEdit* |
| *15* | *getInviteCode() {}* | *getter of inviteCode* |
| *16* | *setInviteCode(string inviteCode) { }* | *setter of inviteCode* |
| *17* | *getSchoolName() {}* | *getter of schoolName* |
| *18* | *setSchoolName(string inviteCode) { }* | *setter of schoolName* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

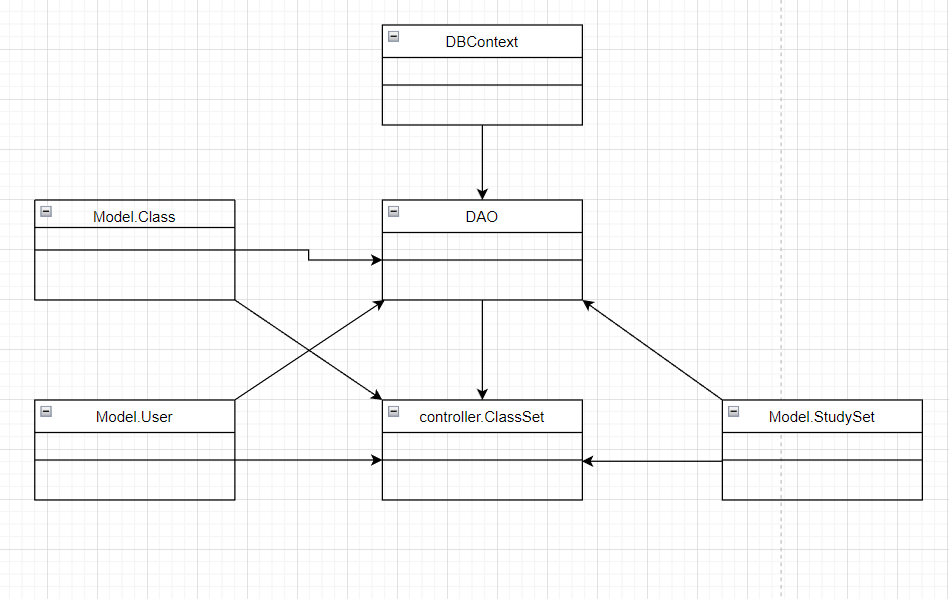
*String sql = "Select \* from StudySet";*

*String sql = "select top 5 \* from StudySet where userId = ? order by studySetId desc";*

*String sql = "select top 5 \* from Folder where userId = ? order by folderId desc ";*

## 14.Class set

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### ClassSet controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *List information of class and information of study sets in class* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getListStudySetbyClassId(int id){}* | *Get all study set in a class* |
| *02* | *getClassByClassID(int id){}* | *Get class by class id* |
| *03* | *getAllStudySet(){}* | *Get all study set* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

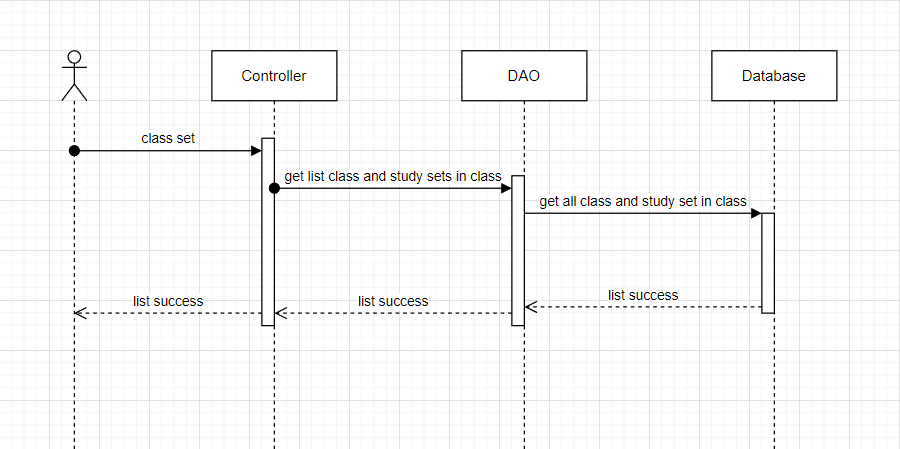
### 

### Class Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Class() {)* | *Default Constructor* |
| *02* | *Class(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of Class Id* |
| *04* | *setId(int id){}* | *setter of Class Id* |
| *05* | *getName(){}* | *getter of Class Name* |
| *06* | *setName(string name){}* | *setter of Class Name* |
| *07* | *getDesc(){}* | *getter of Desc* |
| *08* | *setDesc(string Desc){}* | *setter of Desc* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |
| *11* | *isIsInvite() {}* | *getter of isInvite* |
| *12* | *setIsInvite(boolean isInvite) { }* | *setter of isInvite* |
| *13* | *isIsEdit() {}* | *getter of isEdit* |
| *14* | *setIsEdit(boolean isEdit) { }* | *setter of isEdit* |
| *15* | *getInviteCode() {}* | *getter of inviteCode* |
| *16* | *setInviteCode(string inviteCode) { }* | *setter of inviteCode* |
| *17* | *getSchoolName() {}* | *getter of schoolName* |
| *18* | *setSchoolName(string inviteCode) { }* | *setter of schoolName* |

### 

### c. Sequence Diagram(s)



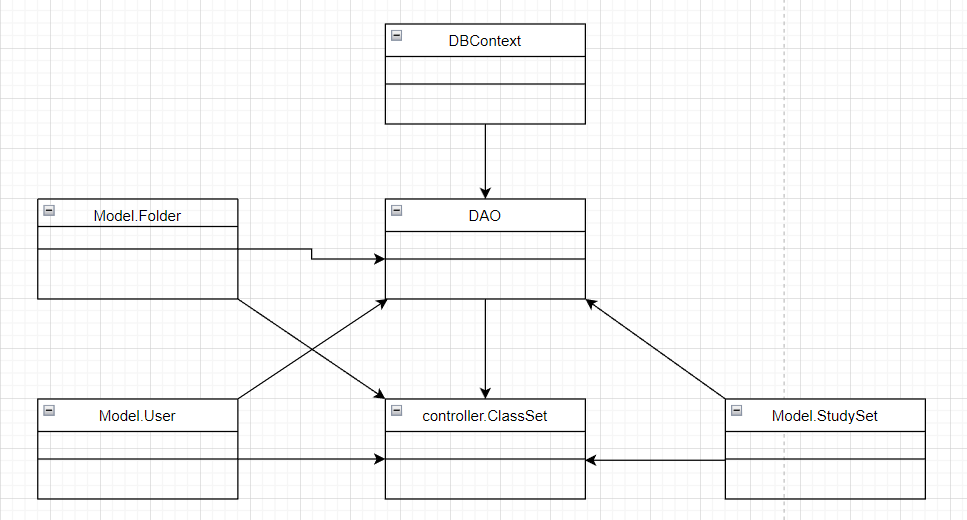
### d. Database Queries

*String sql = "select\*from [Class] where classId=?";*

*String sql = "Select \* from StudySet";*

## 15.Folder set

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### FolderSet controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Get List study set* |
| *02* | *doPost(){}* | *Delete folder and study set in folder* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getListStudySet(int id){}* | *Get list study set and card in study set* |
| *02* | *getFolderByFolderId(int id){}* | *Get folder by folder id* |
| *03* | *getAllStudySet()* | *Get info of study set* |
| *04* | *removeListFolder(int id){}* | *Get remove list folder* |
| *05* | *deleteFolderByFolderId(int id){}* | *Delete folder by folder id* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

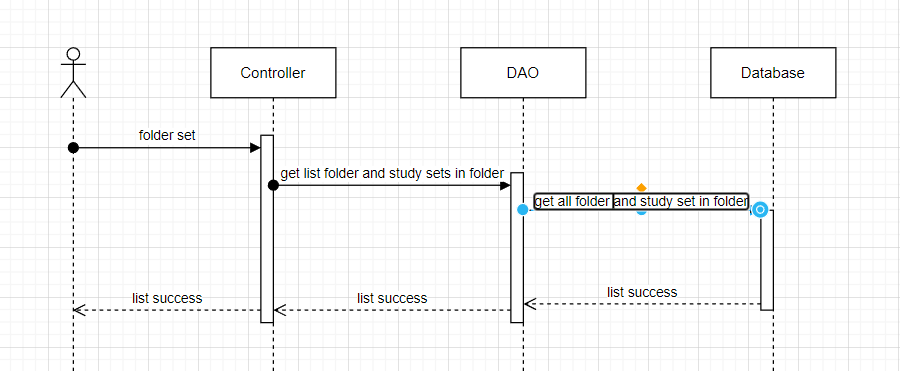
### 

### Folder Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Folder() {)* | *Default Constructor* |
| *02* | *Folder(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string term){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Definition){}* | *setter of Description* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |

### 

### c. Sequence Diagram(s)



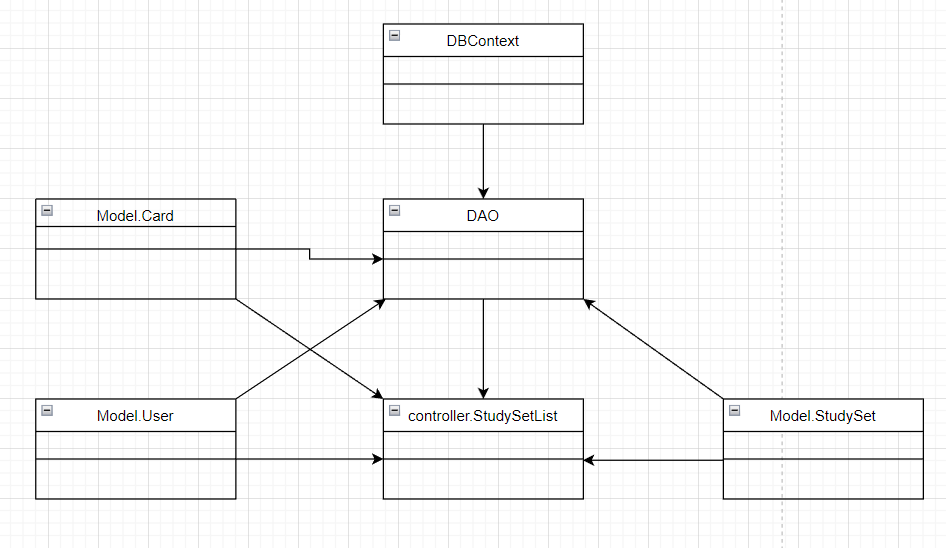
### d. Database Queries

*String sql = "select \* from [Folder] where folderId =? ";*

*String sql = "Select \* from StudySet";*

## 16.Study set list (wait)

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### Verify controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Send redirect to verify.jsp* |
| *02* | *doPost(){}* | *Active account by verify code* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Active account(string email){}* | *Update is active in database* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

### 

### 

### Folder Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Folder() {)* | *Default Constructor* |
| *02* | *Folder(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string term){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Definition){}* | *setter of Description* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |

### 

### Class Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Class() {)* | *Default Constructor* |
| *02* | *Class(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of Class Id* |
| *04* | *setId(int id){}* | *setter of Class Id* |
| *05* | *getName(){}* | *getter of Class Name* |
| *06* | *setName(string name){}* | *setter of Class Name* |
| *07* | *getDesc(){}* | *getter of Desc* |
| *08* | *setDesc(string Desc){}* | *setter of Desc* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |
| *11* | *isIsInvite() {}* | *getter of isInvite* |
| *12* | *setIsInvite(boolean isInvite) { }* | *setter of isInvite* |
| *13* | *isIsEdit() {}* | *getter of isEdit* |
| *14* | *setIsEdit(boolean isEdit) { }* | *setter of isEdit* |
| *15* | *getInviteCode() {}* | *getter of inviteCode* |
| *16* | *setInviteCode(string inviteCode) { }* | *setter of inviteCode* |
| *17* | *getSchoolName() {}* | *getter of schoolName* |
| *18* | *setSchoolName(string inviteCode) { }* | *setter of schoolName* |

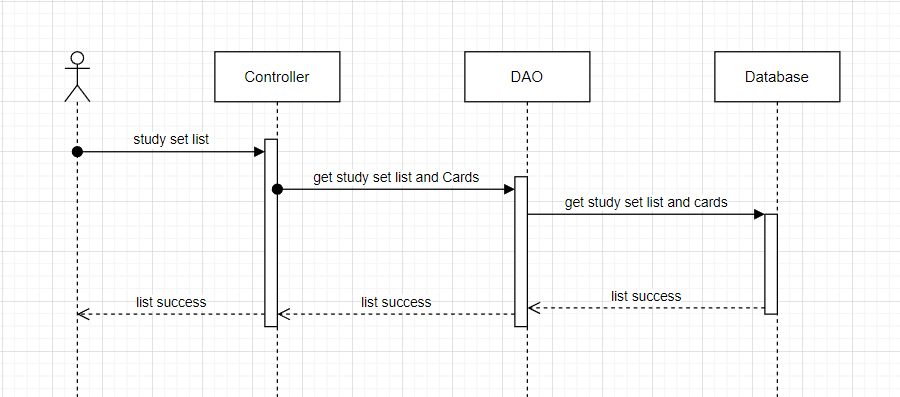
### 

### Card Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "UPDATE [User]\n"*

*+ "SET\n"*

*+ "\n"*

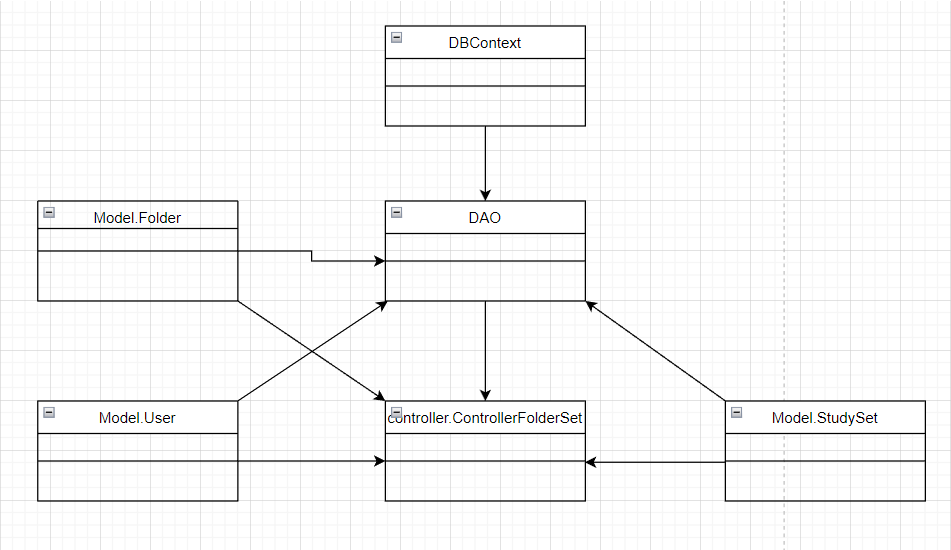
*+ "password = ?\n"*

*+ "\n"*

*+ "WHERE userId = ?";*

## 17.Controller folder set

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### ControllerFolderSet controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Delete or add new study set to folder set* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *deleteStudySetInFolder(int fid, int sid){}* | *Delete study set from folder* |
| *02* | *addStudySetInFolder(int fid, int sid){}* | *Add new study set to folder* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

### 

### 

### Folder Class

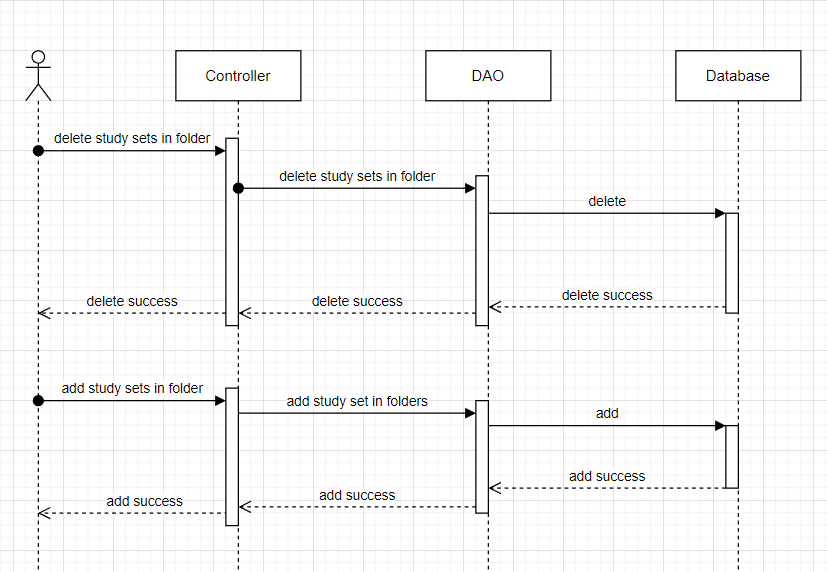
|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Folder() {)* | *Default Constructor* |
| *02* | *Folder(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string term){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Definition){}* | *setter of Description* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |

### 

### 

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "DELETE FROM [dbo].[ListFolder]\n"*

*+ " WHERE studySetId=? and folderId=?";*

*String sql = "INSERT INTO [dbo].[ListFolder]\n"*

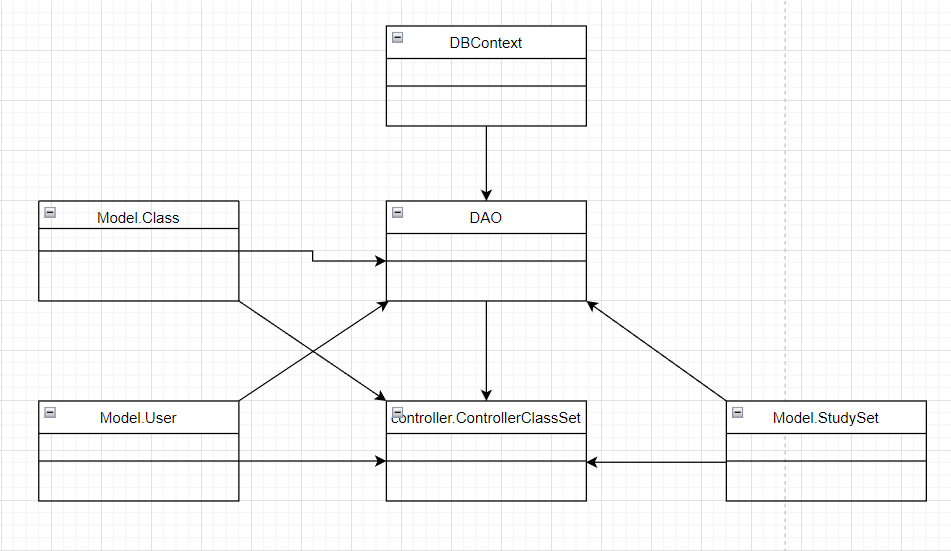
*+ " ([studySetId]\n"*

*+ " ,[folderId])\n"*

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

## 18.Controller class set

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### Controller Class Set Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Delete or add study set in class* |

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *deleteStudySetInClass(int cid, int sid){}* | *Delete study set from class* |
| *02* | *addStudySetInClass(int cid, int sid){}* | *Add new study set to class* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

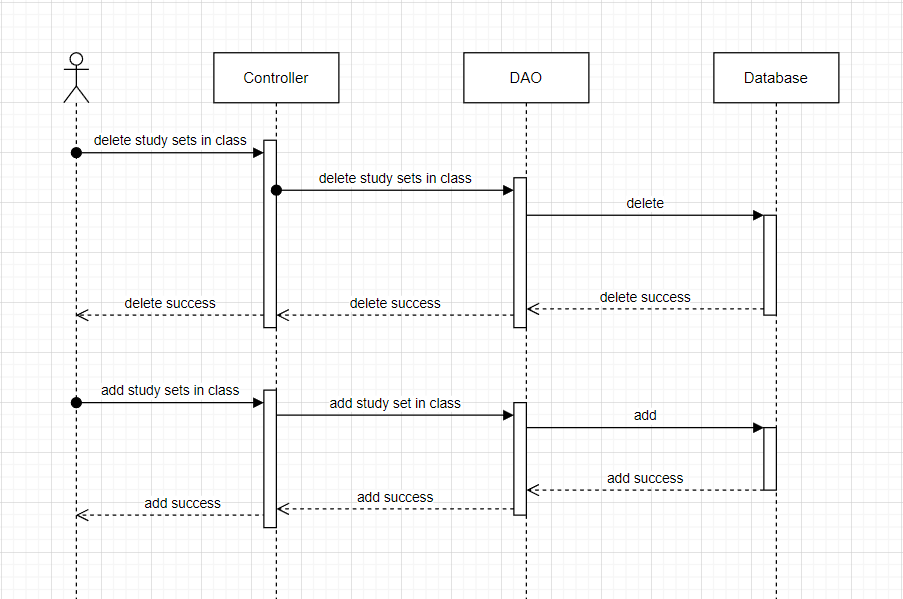
### 

### Class Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Class() {)* | *Default Constructor* |
| *02* | *Class(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of Class Id* |
| *04* | *setId(int id){}* | *setter of Class Id* |
| *05* | *getName(){}* | *getter of Class Name* |
| *06* | *setName(string name){}* | *setter of Class Name* |
| *07* | *getDesc(){}* | *getter of Desc* |
| *08* | *setDesc(string Desc){}* | *setter of Desc* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |
| *11* | *isIsInvite() {}* | *getter of isInvite* |
| *12* | *setIsInvite(boolean isInvite) { }* | *setter of isInvite* |
| *13* | *isIsEdit() {}* | *getter of isEdit* |
| *14* | *setIsEdit(boolean isEdit) { }* | *setter of isEdit* |
| *15* | *getInviteCode() {}* | *getter of inviteCode* |
| *16* | *setInviteCode(string inviteCode) { }* | *setter of inviteCode* |
| *17* | *getSchoolName() {}* | *getter of schoolName* |
| *18* | *setSchoolName(string inviteCode) { }* | *setter of schoolName* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "DELETE FROM [dbo].[ListStudySet]\n"*

*+ " WHERE studySetId=? and classId=?";*

*String sql = "INSERT INTO [dbo].[ListStudySet]\n"*

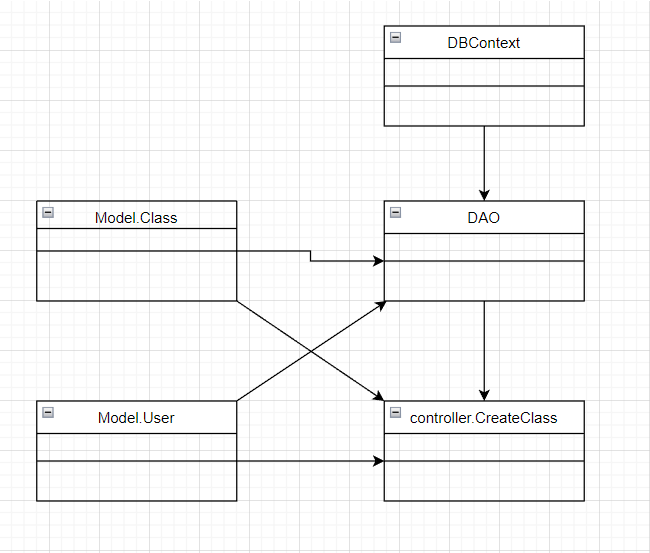
*+ " ([studySetId]\n"*

*+ " ,[classId])\n"*

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

## 19. Create class

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### CreateClass controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *List some class* |
| *02* | *doPost(){}* | *Create new class* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getAllClass(){}* | *Get all class in database* |
| *02* | *createClass(class c)* | *Create a new class* |

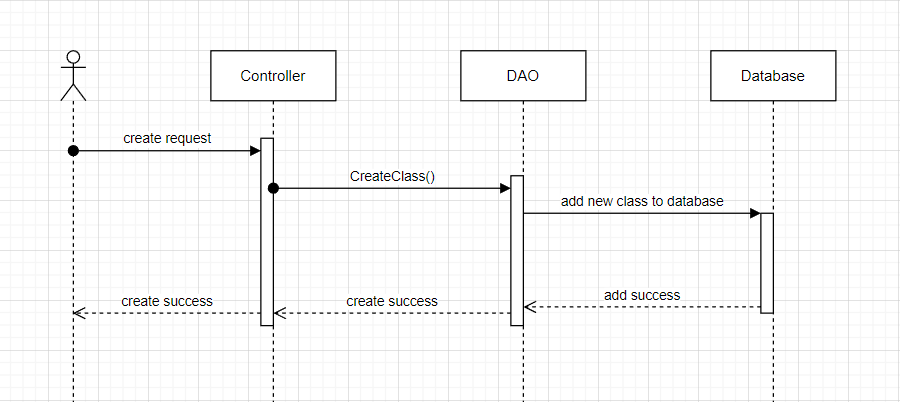
### 

### Class Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Class() {)* | *Default Constructor* |
| *02* | *Class(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of Class Id* |
| *04* | *setId(int id){}* | *setter of Class Id* |
| *05* | *getName(){}* | *getter of Class Name* |
| *06* | *setName(string name){}* | *setter of Class Name* |
| *07* | *getDesc(){}* | *getter of Desc* |
| *08* | *setDesc(string Desc){}* | *setter of Desc* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |
| *11* | *isIsInvite() {}* | *getter of isInvite* |
| *12* | *setIsInvite(boolean isInvite) { }* | *setter of isInvite* |
| *13* | *isIsEdit() {}* | *getter of isEdit* |
| *14* | *setIsEdit(boolean isEdit) { }* | *setter of isEdit* |
| *15* | *getInviteCode() {}* | *getter of inviteCode* |
| *16* | *setInviteCode(string inviteCode) { }* | *setter of inviteCode* |
| *17* | *getSchoolName() {}* | *getter of schoolName* |
| *18* | *setSchoolName(string inviteCode) { }* | *setter of schoolName* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "INSERT INTO [dbo].[Class]\n"*

*+ " ([className]\n"*

*+ " ,[description]\n"*

*+ " ,[isInvite]\n"*

*+ " ,[inviteCode]\n"*

*+ " ,[isEdit]\n"*

*+ " ,[schoolName]\n"*

*+ " ,[userId])\n"*

*+ " VALUES\n"*

*+ " (?\n"*

*+ " ,? \n"*

*+ " ,? \n"*

*+ " ,?\n"*

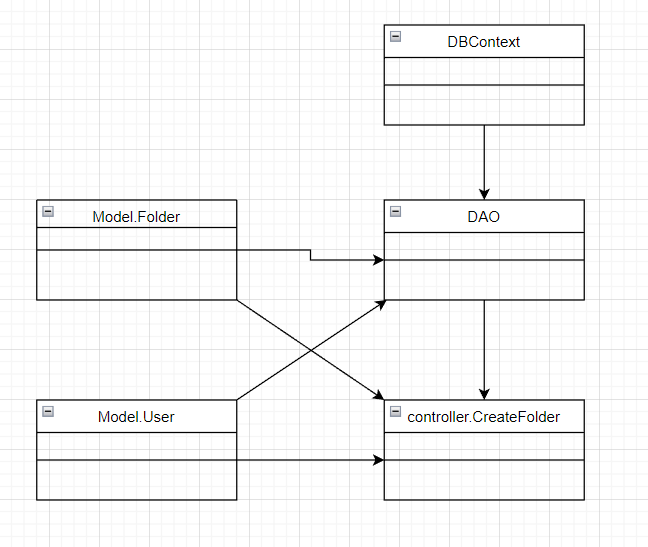
*+ " ,?\n"*

*+ " ,?\n"*

*+ " ,?)";*

## 20. Create folder

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### CreateFolder controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *List all folder by user id* |
| *02* | *doPost(){}* | *Create new folder* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getAllFolderByUserId(int id){}* | *Get all folder by user id after creating* |
| *02* | *CreateFolder(Folder f){}* | *Create a new folder* |

### 

### 

### Folder Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Folder() {)* | *Default Constructor* |
| *02* | *Folder(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string term){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Definition){}* | *setter of Description* |
| *09* | *getUserId() { }* | *getter of UserId* |
| *10* | *setUserId(int UserId) { }* | *setter of UserId* |

### 

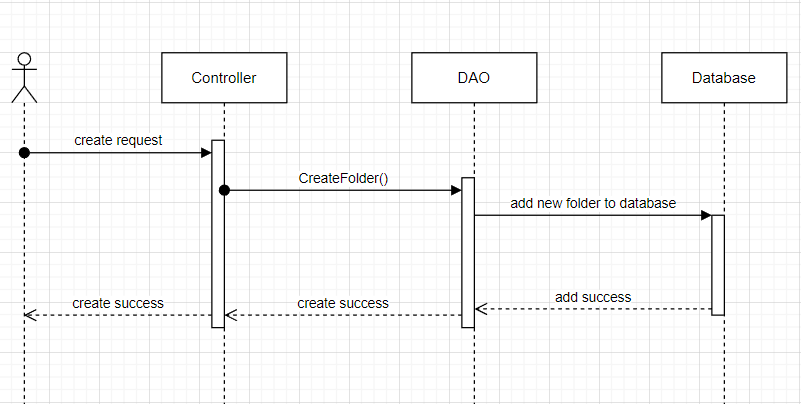
### 

### 

### 

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "INSERT INTO [dbo].[Folder]\n"*

*+ " ([title]\n"*

*+ " ,[description]\n"*

*+ " ,[userId]\n"*

*+ " ,[isShare])\n"*

*+ " VALUES\n"*

*+ " (?\n"*

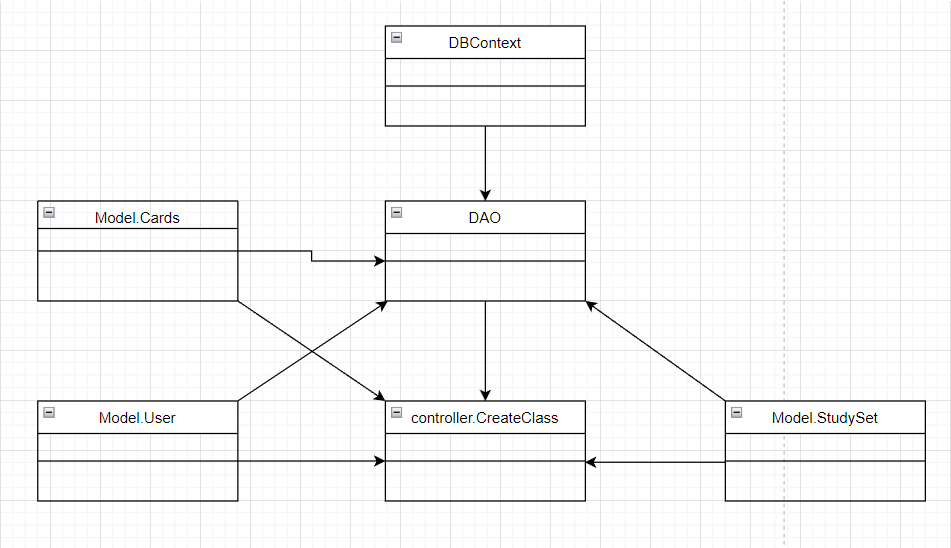
*+ " ,?\n"*

*+ " ,?\n"*

*+ " ,?)";*

## 21. Reset studied Cards

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### ResetStudiedCard controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Reset studied cards in study set* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *resetStudiedCard(int uid, int studysetId){}* | *Reset studied card to study again.* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

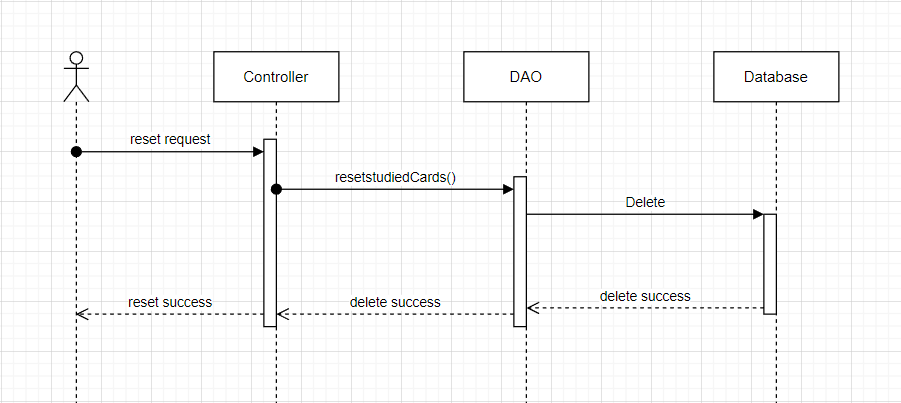
### 

### Card Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### 

### c. Sequence Diagram(s)



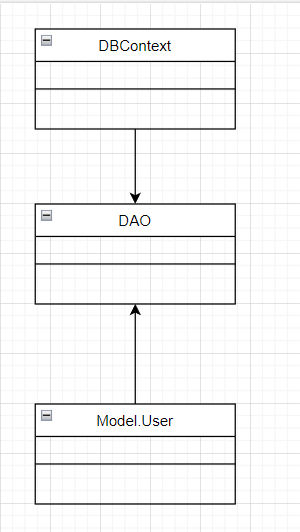
### d. Database Queries

*String sql = "DELETE FROM [dbo].[FlashCards]\n"*

*+ " WHERE userId=? and studySetId=?";*

## 22. Send email

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### SendEmail Class

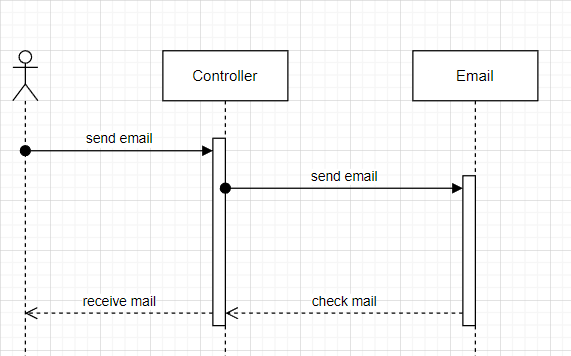
|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getRandom(){}* | *Get a code 6 character reandom* |
| *02* | *sendEmail(user, code){}* | *Send code to email of user* |
| *03* | *sendEmailPass(id, to ,cid){}* | *Send link to change password to email* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Active account(string email){}* | *Update is active in database* |

### 

### c. Sequence Diagram(s)

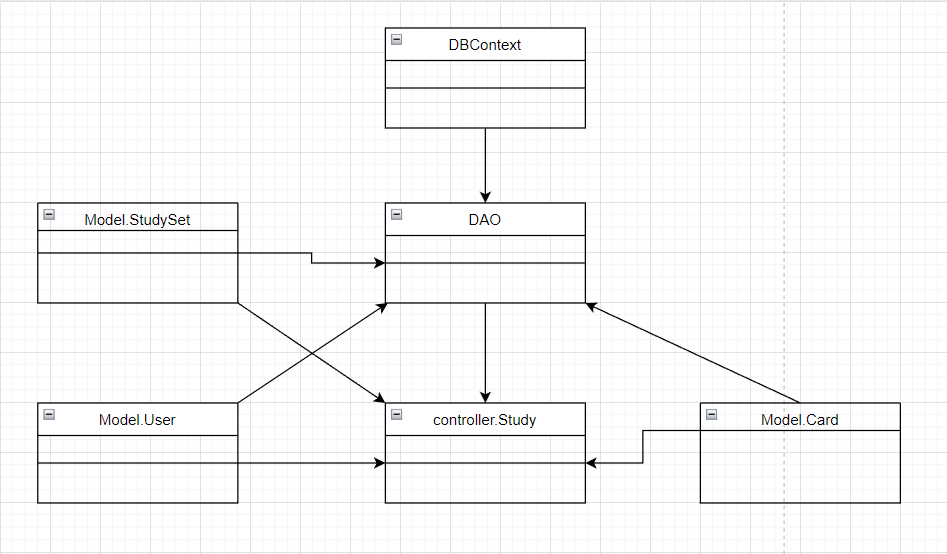


### d. Database Queries

*Send email function do not use database queries.*

## 23. Study

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### Study controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Get all cards to test* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *GetStudySetByID(int id){}* | *Get study set by id* |
| *02* | *ListCTF(int id, nit uid, int num){}* | *List cards true false* |
| *03* | *listATF(int id, int uid, int num){}* | *Get answer of cards* |
| *04* | *listCMC(int id, int uid, listCTF,int num){}* | *List card to multiple choice* |

#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

### 

### 

### 

### 

### 

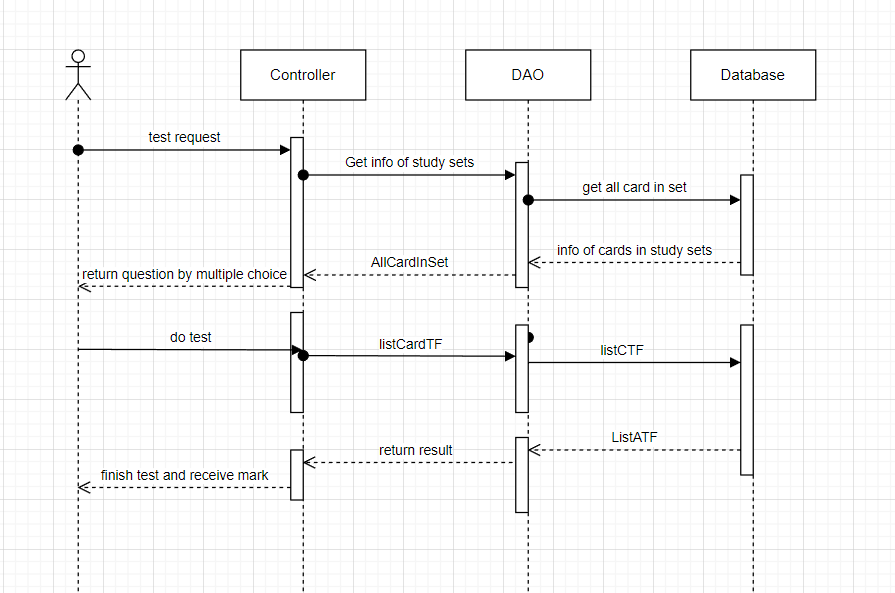
### 

### Card Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### 

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "SELECT [studySetId]\n"*

*+ " ,[title]\n"*

*+ " ,[description]\n"*

*+ " ,[isShare]\n"*

*+ " ,[userId]\n"*

*+ " FROM [dbo].[StudySet]\n"*

*+ " WHERE studySetId=?";*

*String sql = "SELECT [cardId]\n"*

*+ " ,[term]\n"*

*+ " ,[definition]\n"*

*+ " ,[studySetId]\n"*

*+ " FROM [dbo].[Card]\n"*

*+ " WHERE studySetID=?";*

*String sql = "SELECT [userId]\n"*

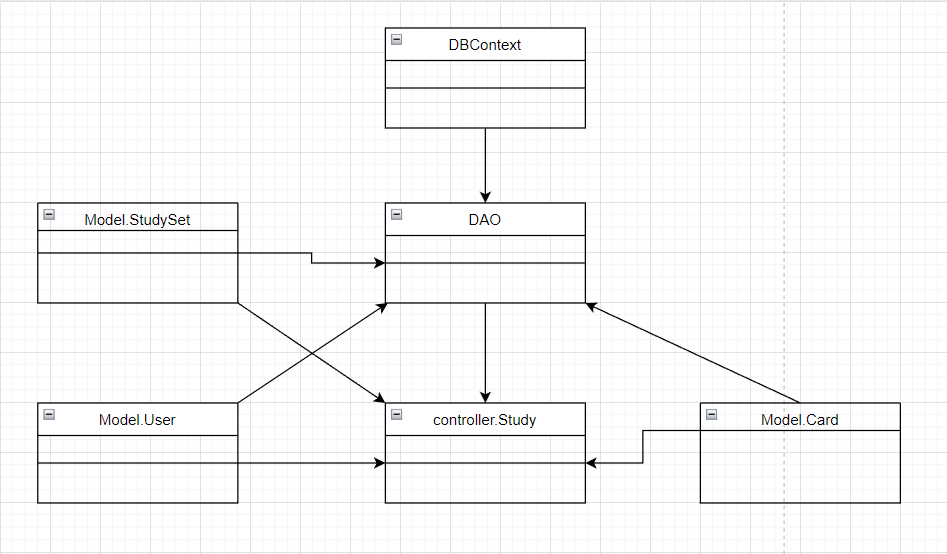
*+ " ,[cardId]\n"*

*+ " FROM [dbo].[FlashCards]"*

*+ " WHERE studySetId = ? and userId = ?";*

## 24. Test

### a. Class Diagram

**

### b. Class Specifications

### User Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *User(){}* | Default constructor |
| *02* | *User(….){}* | *Parameterized constructor* |
| *03* | *getName(){}* | *Getter of Name* |
| *04* | *setName(string name){}* | *Setter of Name* |
| *05* | *getPass(){}* | *Getter of Pass* |
| *06* | *setPass(string pass){}* | *Setter of Pass* |
| *07* | *getEmail(){}* | *Getter email* |
| *08* | *setEmail(string email){}* | *Setter of email* |
| *09* | *isIsActive(){}* | *Getter of isActive* |
| *10* | *setIsActive(Boolean isActive){}* | *Setter of isActive* |
| *11* | *getAvatar(){}* | *Getter of avatar* |
| *12* | *setAvatar(string avatar)* | *Setter of avatar* |
| *13* | *getLanguage(){}* | *Getter of language* |
| *14* | *setLanguage(string language){}* | *Setter of language* |

#### DBContext Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *DBContext(){}* | *Connect to database* |

#### Test controller Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet(){}* | *Get all Cards to study* |

#### DAO Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Active account(string email){}* | *Update is active in database* |

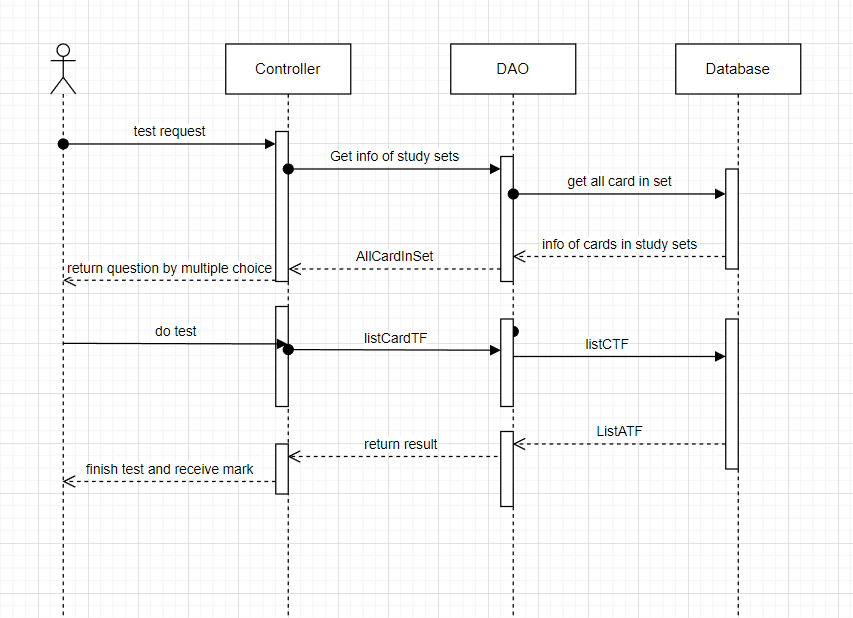
#### StudySet Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *StudySet() {)* | *Default Constructor* |
| *02* | *StudySet(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of StudySetId* |
| *04* | *setId(int id){}* | *setter of StudySetId* |
| *05* | *getTitle(){}* | *getter of Title* |
| *06* | *setTitle(string title){}* | *setter of Title* |
| *07* | *getDescription(){}* | *getter of Description* |
| *08* | *setDescription(string Description){}* | *setter of Description* |
| *09* | *isIsShare() {}* | *getter of isShare* |
| *10* | *setIsShare(boolean isShare) { }* | *setter of isShare* |
| *11* | *getUserId() { }* | *getter of UserId* |
| *12* | *setUserId(int userId) { }* | *setter of UserId* |
| *13* | *getNumberCard() {}* | *get number flashcard in studyset* |
| *14* | *String getAuthor() {}* | *get name of User created studyset* |

### Card Class

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Card() {)* | *Default Constructor* |
| *02* | *Card(){.....}* | *Parameterized constructor* |
| *03* | *getId(){}* | *getter of CardId* |
| *04* | *setId(int id){}* | *setter of CardId* |
| *05* | *getTerm(){}* | *getter of Term* |
| *06* | *setTerm(string term){}* | *setter of Term* |
| *07* | *getDefinition(){}* | *getter of Definition* |
| *08* | *setDefinition(string Definition){}* | *setter of Definition* |
| *09* | *getStudySetId() { }* | *getter of StudySetId* |
| *10* | *setStudySetId(int studySetId) { }* | *setter of StudySetId* |

### c. Sequence Diagram(s)



### d. Database Queries

*String sql = "SELECT [cardId]\n"*

*+ " ,[term]\n"*

*+ " ,[definition]\n"*

*+ " ,[studySetId]\n"*

*+ " FROM [dbo].[Card]\n"*

*+ " WHERE studySetID=?";*

String sql = "SELECT [cardId]\n"

+ " ,[term]\n"

+ " ,[definition]\n"

+ " ,[studySetId]\n"

+ " FROM [dbo].[Card]\n"

+ " WHERE studySetID=?";