

**MENTOR FINDER**

**Software Design Document**

– Hanoi, July 2021 –

**Table of Contents**

[I. Overview](#_heading=h.gjdgxs) **7**

[1. Code Packages/Namespaces](#_heading=h.30j0zll) 7

[2. Database Schema](#_heading=h.1fob9te) 8

[II. Code Designs (Not done)](#_heading=h.2et92p0) **9**

[1. Login](#_heading=h.tyjcwt) 9

[a. Class Diagram](#_heading=h.3dy6vkm) 9

[b. Class Specifications](#_heading=h.1t3h5sf) 10

[WebSecurityConfig](#_heading=h.dlp95u8shtt5) 10

[c. Sequence Diagram(s)](#_heading=h.4d34og8) 10

[d. Database queries](#_heading=h.2s8eyo1) 11

[2. Register](#_heading=h.17dp8vu) 11

[Class Diagram](#_heading=h.ls7p5dv3r0wa) 11

[Class Specification](#_heading=h.bu2e11fj6cg1) 11

[Sequence Diagram(s)](#_heading=h.h1m771hge9mb) 11

[Database queries](#_heading=h.ovl63ddu89o8) 11

[3. Request](#_heading=h.4kvoaypw7ddt) 12

[Class Diagram](#_heading=h.lwqn7s2gbfkx) 12

[Class Specifications](#_heading=h.xuchwb7v4ysu) 12

[RequestService](#_heading=h.zh67v7xuxj4h) 12

[Request Controller](#_heading=h.n32j03h4wh02) 12

[Sequence Diagram(s)](#_heading=h.9kscklz9xrty) 12

[Database queries](#_heading=h.g19czx9n82f0) 13

[4. Enrolled](#_heading=h.peqrwj4o3n9f) 13

[Class Diagram](#_heading=h.2ha4qersktcz) 13

[Class Specifications](#_heading=h.e5a51tjaypow) 13

[EnrolledService](#_heading=h.3ifn6x8d59e8) 13

[EnrolledController](#_heading=h.1tnc1nfrqroy) 14

[Sequence Diagram(s)](#_heading=h.puct89asdot2) 15

[Sequence Diagram for Mentee View Enrolled List of a RequestSequence Diagram for Mentee View Enrolled List of a Request](#_heading=h.eeq7sl2vcndt) 15

[Database queries](#_heading=h.jj1y3ati96er) 16

[5. Notification](#_heading=h.nsm1qndgs48w) 17

[Class Diagram](#_heading=h.bwc75ei2lyo3) 17

[Class Specifications](#_heading=h.mp1ntl8lan3d) 17

[Notify: THIS CLASS STORE INFORMATION OF A NOTIFY](#_heading=h.4ij62385wlnz) 17

[NotifyService](#_heading=h.sakjg1iwa6fy) 17

[NotifyController](#_heading=h.t6xfpu4818y9) 18

[Sequence Diagram(s)](#_heading=h.txzpic6z0tyl) 18

[Database queries](#_heading=h.6rcfj5635vwg) 18

[6. Home](#_heading=h.ean2uqdzx109) 18

[Class Diagram](#_heading=h.7ojec3nw0e7g) 19

[Class Specification](#_heading=h.8wixkml7k34o) 19

[Sequence Diagram(s)](#_heading=h.dwoj69kv5ive) 19

[Database queries](#_heading=h.b7jfnz3ca5by) 20

[7. Comment](#_heading=h.ab3t3q4rbf6c) 20

[Class Diagram](#_heading=h.mip9ku6u1gaw) 20

[Class Specification](#_heading=h.t4igofbyft1i) 20

[Sequence Diagram(s)](#_heading=h.54ldlaa2ugqy) 20

[Database queries](#_heading=h.s978ju7f8ds5) 21

[8. Admin view all list request](#_heading=h.ogc2cfbfkkka) 21

[Class Diagram](#_heading=h.8konqzd0itup) 21

[Class Specification](#_heading=h.q4v9ac5sbvfr) 21

[Sequence Diagram(s)](#_heading=h.o0geafdq4iiz) 23

[Database queries](#_heading=h.964p94u7tano) 24

[9. Admin manage requests](#_heading=h.cyicyhm0hhju) 24

[Class Diagram](#_heading=h.saxss1gut0mk) 24

[Class Specification](#_heading=h.g1oiicvhe079) 24

[Sequence Diagram(s)](#_heading=h.cle0ftrxawwu) 28

[Database queries](#_heading=h.36rq47vmjlf) 29

[10. Admin filter requests](#_heading=h.sddf9f6fl49t) 30

[Class Diagram](#_heading=h.itnh3to9f95m) 30

[Class Specification](#_heading=h.r09k7u3h6wh0) 30

[Sequence Diagram(s)](#_heading=h.z3sv2vh5asy2) 33

[Database queries](#_heading=h.rmvbcg6x3ord) 33

[11. Admin notify User](#_heading=h.c602iwers9jc) 34

[Class Diagram](#_heading=h.v3tnt9je33fl) 34

[Class Specification](#_heading=h.p68oojqtuuoe) 34

[Sequence Diagram(s)](#_heading=h.rhi9xej39b89) 38

[Database queries](#_heading=h.hxudoemdpvel) 38

[12. Admin request details](#_heading=h.imcq84cutkvl) 39

[Class Diagram](#_heading=h.t0sh89gqn3n) 39

[Class Specification](#_heading=h.9d9kbsdb7au) 39

[Sequence Diagram(s)](#_heading=h.t5gjy686su64) 42

[Database queries](#_heading=h.4i3mhwo1n4rr) 42

[13. Admin Dashboard](#_heading=h.kdq4sech5uec) 42

[Class Diagram](#_heading=h.emzbjjv63ry3) 43

[Class Specification](#_heading=h.9k0rdtjmpj8p) 43

[Sequence Diagram(s)](#_heading=h.jxfiui9o36br) 47

[Database queries](#_heading=h.jnz4ffruktj3) 47

[14. Change Password](#_heading=h.4lczw4mhs8mc) 48

[Class Diagram](#_heading=h.zhzckn1jcwha) 48

[Class Specification](#_heading=h.skocjo683v) 48

[Sequence Diagrams](#_heading=h.a00g4gpda09j) 49

[Database Queries](#_heading=h.gyz6qogo1fj1) 49

[15. Admin daily report](#_heading=h.2mlwwbndhi9b) 50

[Class Diagram](#_heading=h.d48naakxlkrf) 50

[Class Specification](#_heading=h.kinqk7j1drwi) 50

[Sequence Diagrams](#_heading=h.v488irs2z1ch) 54

[Database Queries](#_heading=h.t0zl3r11qbcb) 54

[16. Admin view mentee list](#_heading=h.rek6dkjmz2w4) 54

[Class Diagram](#_heading=h.9yje572v2pf9) 54

[Class Specification](#_heading=h.ggng7qtp32bc) 55

[Sequence Diagrams](#_heading=h.czl9e95kgfai) 56

[Database Queries](#_heading=h.lxqvoh3mnyh3) 57

[17. Admin update mentee info](#_heading=h.baxgo8lv9mhv) 57

[Class Diagram](#_heading=h.y5cxn5mth4) 57

[Class Specification](#_heading=h.4h38hp4bgctg) 57

[Sequence Diagrams](#_heading=h.bkzlzbdlnoc3) 57

[Database Queries](#_heading=h.il8bdsxapgr) 57

[18. Admin deactivate/activate mentee account](#_heading=h.3od17afewucp) 57

[Class Diagram](#_heading=h.acxk5wu7u5dw) 57

[Class Specification](#_heading=h.nkeo9dot4c8u) 57

[Sequence Diagrams](#_heading=h.sjxhvh73a54k) 57

[Database Queries](#_heading=h.9az2dyry4m13) 57

[19. Admin view mentor list](#_heading=h.ir46awb61tlh) 57

[Class Diagram](#_heading=h.auoeeq4ybg72) 57

[Class Specification](#_heading=h.tgmuaalhrcoe) 57

[Sequence Diagrams](#_heading=h.hjom6zn7ggqs) 57

[Database Queries](#_heading=h.pdgvlw1evonw) 57

[20. Admin update mentor info](#_heading=h.q2u06kdwj592) 57

[Class Diagram](#_heading=h.fm1138rh021f) 57

[Class Specification](#_heading=h.9jr4ihgo60zf) 57

[Sequence Diagrams](#_heading=h.hncxm06mdrjd) 57

[Database Queries](#_heading=h.eoyyj6pfxfg2) 57

[21. Admin deactivate/activate mentor account](#_heading=h.ach5of1h2ph4) 57

[Class Diagram](#_heading=h.ur35kmia60np) 57

[Class Specification](#_heading=h.anf8u1roewzt) 57

[Sequence Diagrams](#_heading=h.djq9hdopyf11) 57

[Database Queries](#_heading=h.ldsi76bu8iyt) 57

[22. Forgot Password](#_heading=h.trow2k1it9vv) 57

[Class Diagram](#_heading=h.d9f56084btri) 57

[Class Specification](#_heading=h.9xk6kptmyjzf) 57

[Sequence Diagrams](#_heading=h.z8tbevpwyc2c) 57

[Database Queries](#_heading=h.8ma13lxz4n01) 57

[III. Database Tables](#_heading=h.3rdcrjn) **57**

[1. LoginInfo](#_heading=h.26in1rg) 57

[2. Authorization](#_heading=h.lnxbz9) 58

[3. LoginInfo\_Authorization](#_heading=h.bvjxbpry4wur) 58

[4. User](#_heading=h.um4mh9y074ae) 58

[5. SkillnExperience](#_heading=h.oeer8jlwxaw7) 58

[6. Subject](#_heading=h.ks98o2m3xlgt) 59

[7. Level](#_heading=h.gomrg9y3msb5) 59

[8. Mentor \_Subject](#_heading=h.sz43nt9omaar) 59

[9. Request](#_heading=h.trvilv9uoc37) 59

[10. Enrolled](#_heading=h.oomrw7k8woei) 60

[11. ChatEndponts](#_heading=h.mf0nfzguvxjh) 60

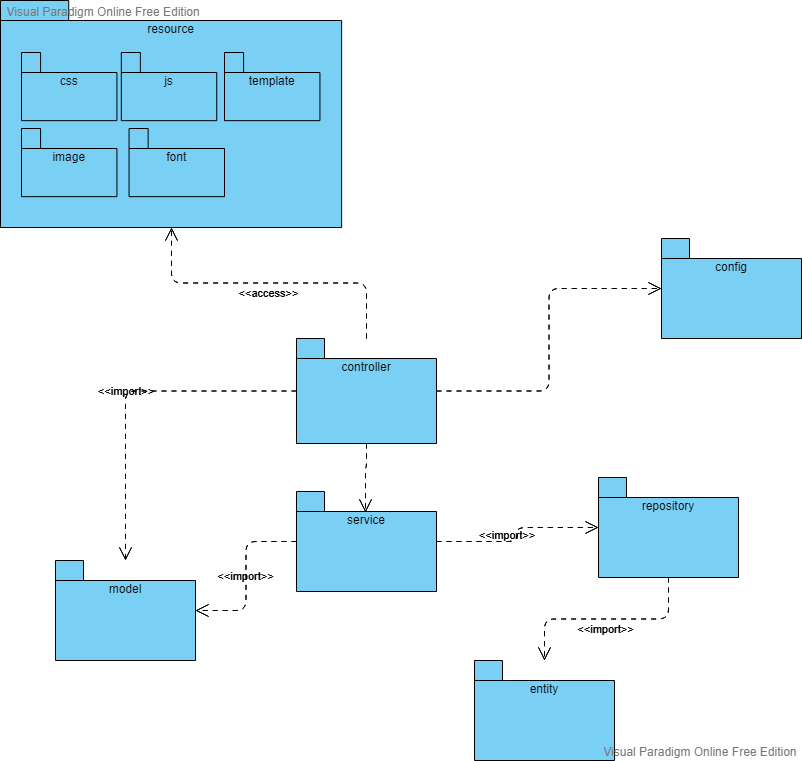
[12. Chat](#_heading=h.lq58p4f80bhj) 60

[13. Notify](#_heading=h.jl3c9nq0ru0g) 60

[14. Comment](#_heading=h.84uufgfr2zkg) 60

# I. Overview

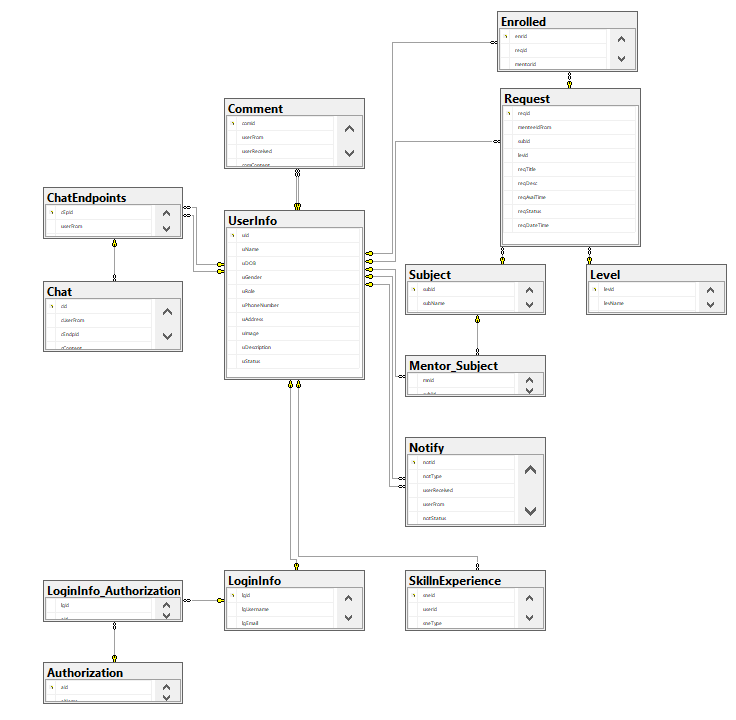
## 1. Code Packages/Namespaces



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

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | resource | Contains static and dynamic view files:   * HTML * JS * CSS |
| 02 | config | Initial security and flow configuration |
| 03 | controller | Flow management |
| 04 | utils | Extra implementation to support web service |
| 05 | service | Implementation of logic and database serving |
| 06 | entity | Entity classes implement directly from the database, serve as a distinct business object |
| 07 | model | Objects that exist merely in the web service, nor the database |
| 08 | repository | Data extraction and I/O from the database |

## 2. Database Schema



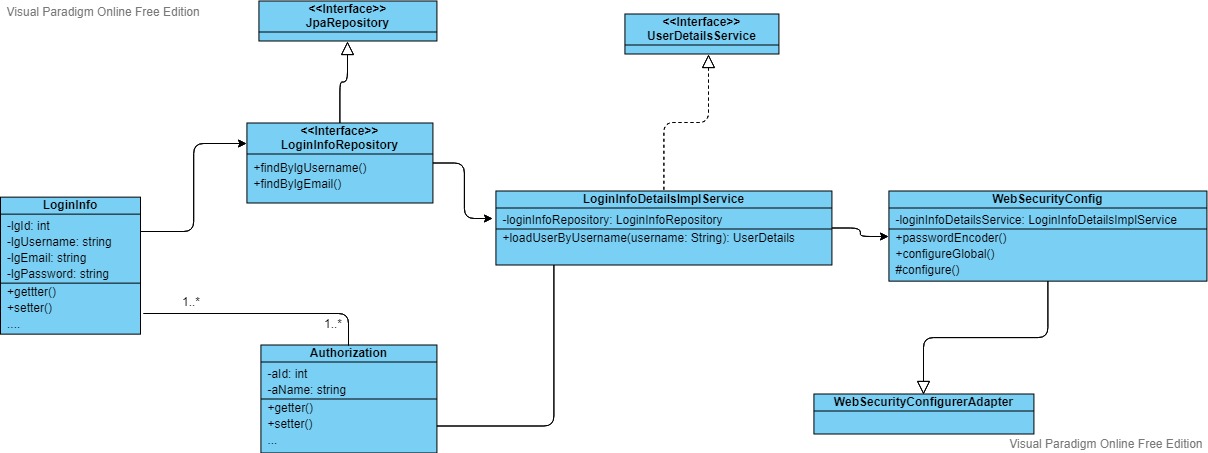
***Table descriptions & package class naming conventions are as below***

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 01 | LoginInfo | Basic login information  - Primary keys: lgId |
| 02 | Authorization | System role: Admin and User  - Primary keys: rId |
| 03 | LoginInfo\_Authorization | Many to many connection of Login\_Info and Authorization  - Foreign keys: lgId  - Foreign keys: rId |
| 04 | User | Object of real User, with information  - Primary keys: uId |
| 05 | SkillnExperience | List of User’s skills and experience  - Primary keys: sneId  - Foreign keys: uId |
| 06 | Subject | The subjects Mentee want to study  - Primary keys: subId |
| 07 | Level | The level user want to study  - Primary keys: levId |
| 08 | Mentor\_Subject | Many to many connection of User(Mentor) and Subject  - Foreign keys: mnId  - Foreign keys: subId |
| 09 | Request | Object of real Request that Users post  - Primary keys: reqId  - Foreign keys: menteeId  - Foreign keys: subId  - Foreign keys: levId |
| 10 | Enrolled | Accepted request is transferred to Enrolled  - Primary keys: enrId  - Foreign keys: reqId  - Foreign keys: mentorId |
| 11 | Notify | List of notification sent from one user to another  - Primary keys: notId  - Foreign keys: userReceived  - Foreign keys: userFrom |
| 12 | Comment | List of comment sent from one user to another  - Primary keys: comId  - Foreign keys: userReceived  - Foreign keys: userFrom |
| 13 | ChatEndpoints | List of chat rooms created between users:  - Primary keys: cEpId  - Foreign keys: userTo  - Foreign keys: userFrom |
| 14 | Chat | A chat message  - Primary keys: cId  - Foreign keys: cEndpId |

# II. Code Designs

## 1. Login

### a. Class Diagram



### b. Class Specifications

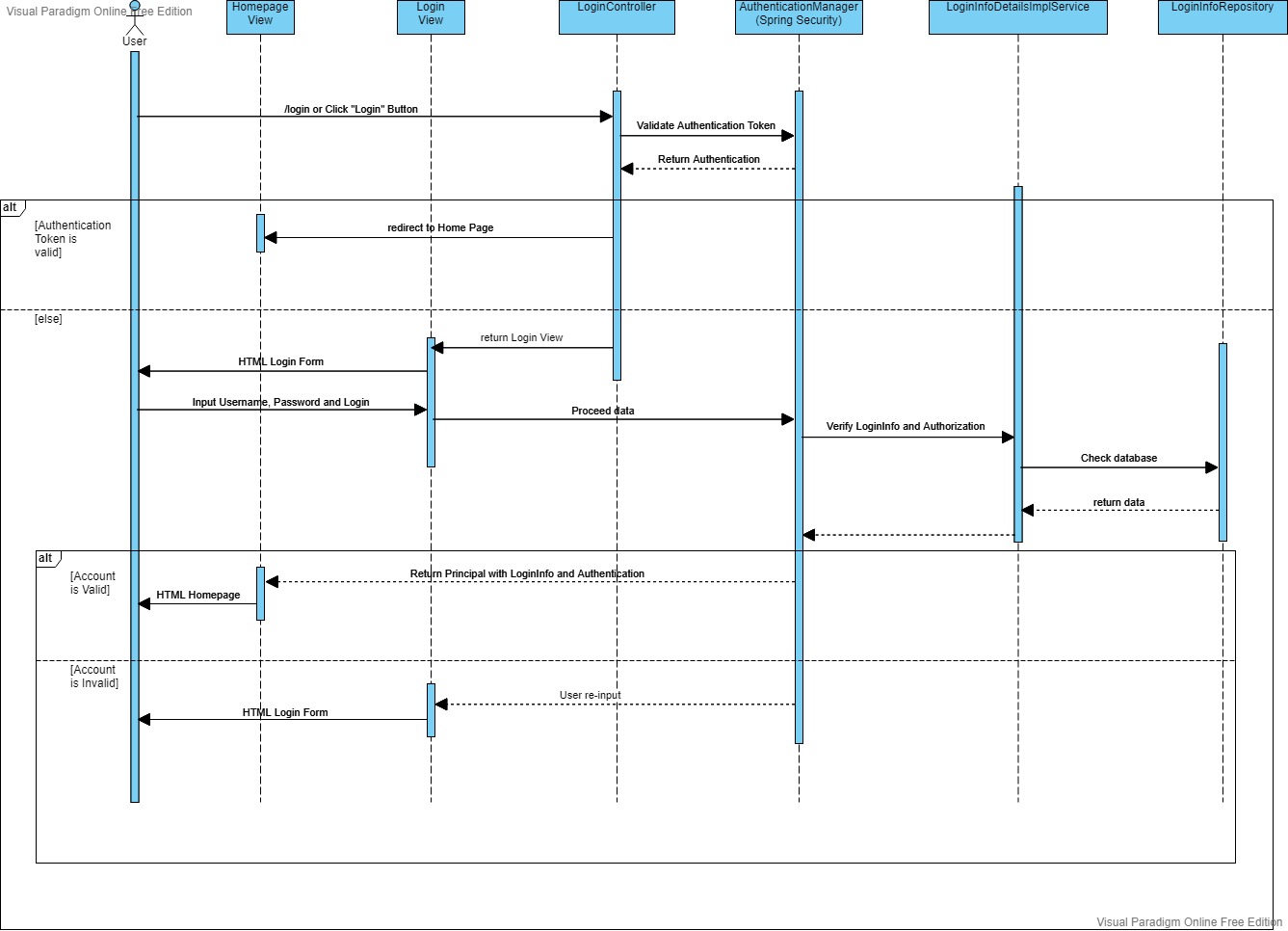
#### WebSecurityConfig: configure Spring Security for authentication and authorization

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | passwordEncoder | Encode Password into Bcrypt |
| 02 | configureGlobal | Configure Authentication for the class, using passwordEncoder |
| 03 | configure | Provide Authentication and Authorization |

***LoginInfoDetailsImplService:*** *combining with SpringSecurity to check data from database*

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | loadUserByUsername | Get User from the provided Info and get their Role |

### c. Sequence Diagram(s)



### d. Database queries

SQL Server equivalent of JPA function:

* loginInfoRepository.findByUsername(username):

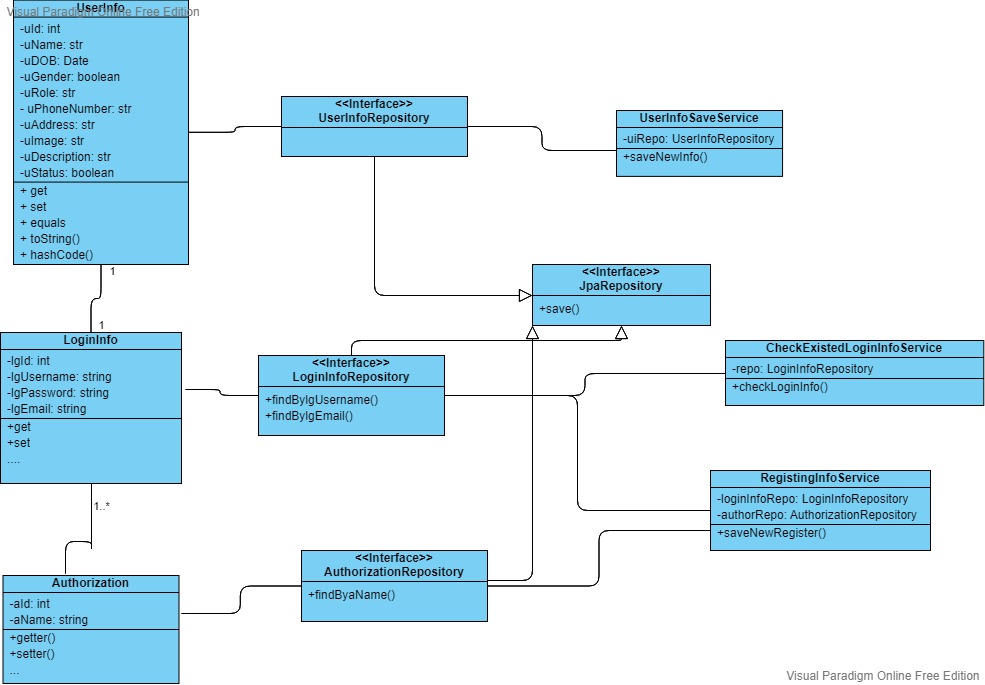
**SELECT \* FROM LoginInfo WHERE igUsername = username**

* Get all the Role of the User. Parameter: lgId

**SELECT rName FROM LoginInfo\_Role, Role WHERE lgId = LoginInfo\_Role.lgId and LoginInfo\_Role.rId = Role.rId**

## 2. Register

### Class Diagram



### Class Specification

***UserInfoSaveService:***

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | saveNewInfo | Save new UserInfo into database |

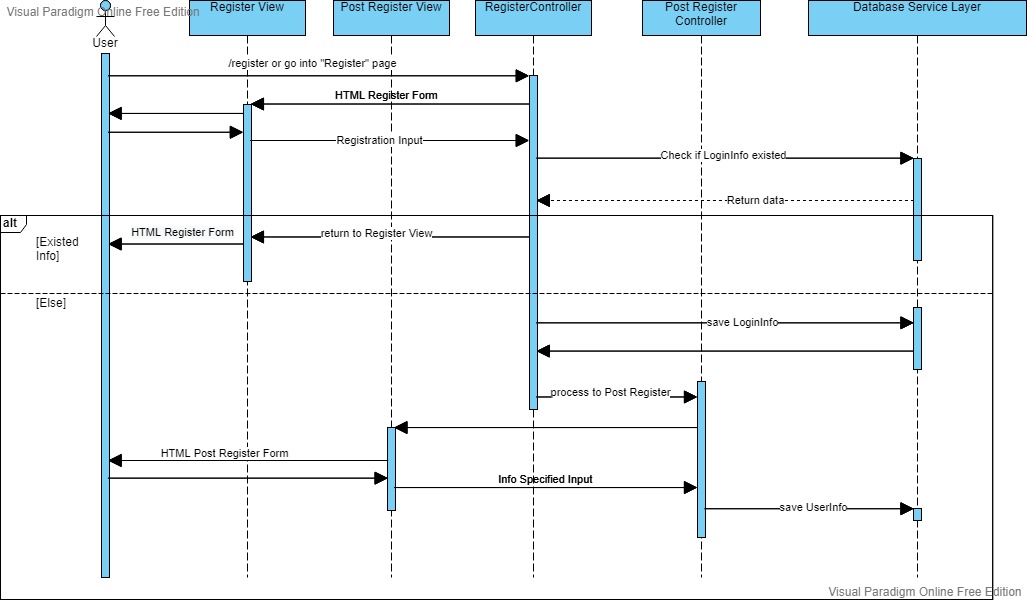
***CheckExistedLoginInfoService***

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | checkLoginInfo | Check if current LoginInfo is already taken |

***RegistingInfoService***

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | saveNewRegister | Save new LoginInfo and Authorization into database |

### Sequence Diagram(s)



### Database queries

* findByaName(name)

SELECT \* FROM Authorization WHERE aName = ?

* findBylgUsername(name)

SELECT \* FROM LoginInfo WHERE lgUsername = ?

* findBylgEmail(email)

SELECT \* FROM LoginInfo WHERE lgEmail =?

* save(UserInfo)

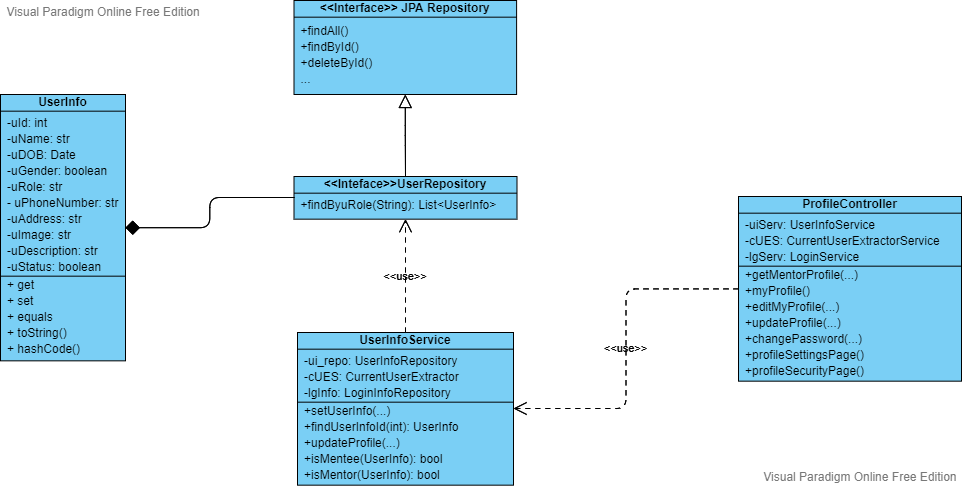
INSERT INTO UserInfo(?,?,?,?,?) VALUES (?,?,?,?,?)

* save(LoginInfo)

INSERT INTO LoginInfo(?,?,?,?) VALUES (?,?,?,?,?

## 3. Profile

### Class Diagram



### Class Specification

#### UserInfoService

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | setUserInfo | update user info of current logged in user |
| 02 | updateProfile | update user info of a specific user |
| 03 | findUserInfoId | get an user by id |
| 04 | isMentee | check if user is mentee |
| 05 | isMentor | check if user is mentor |

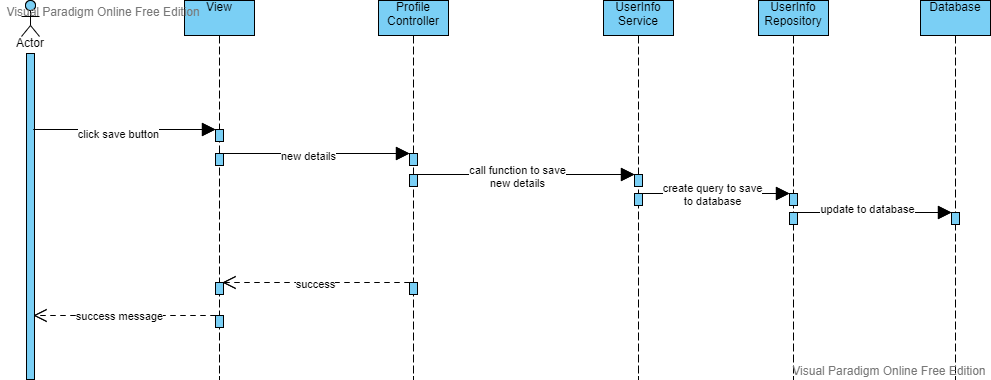
#### 

#### UserInfoController

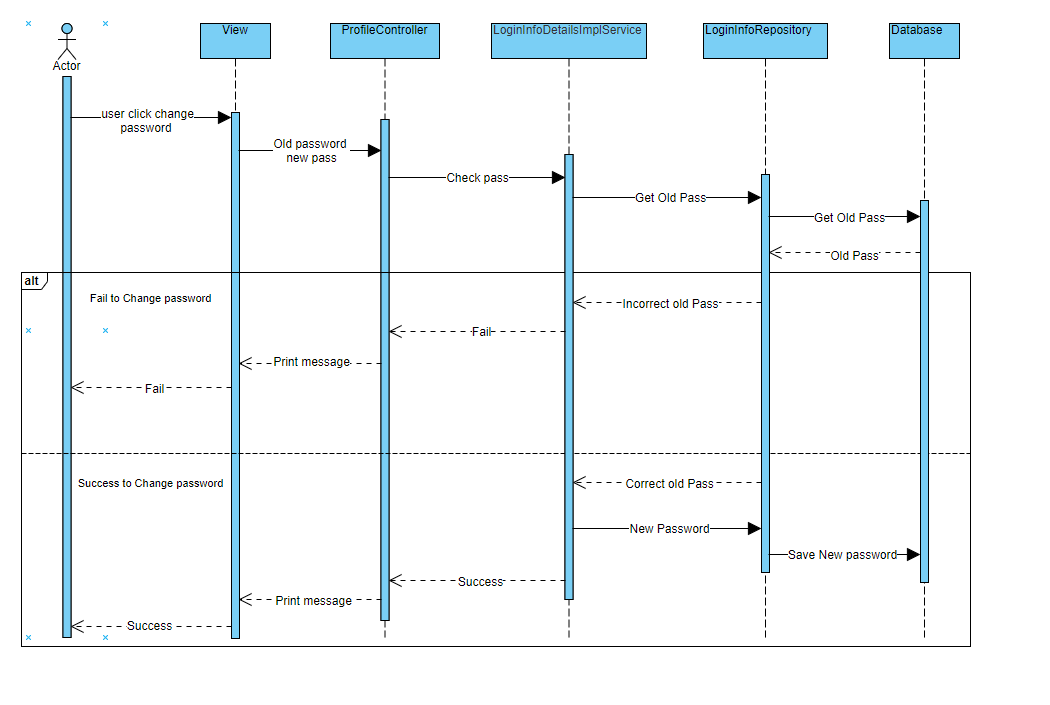
| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getMentorProfile | call function to get profile of a mentor |
| 02 | myProfile | call function to get profile of current logged in user |
| 03 | updateProfile | call function to update profile of current logged in user |
| 04 | changePassword | call function to update profile of current logged in user |
| 05 | profileSettingsPage | load edit profile template |
| 06 | profileSecurityPage | load change password template |

### Sequence Diagram(s)

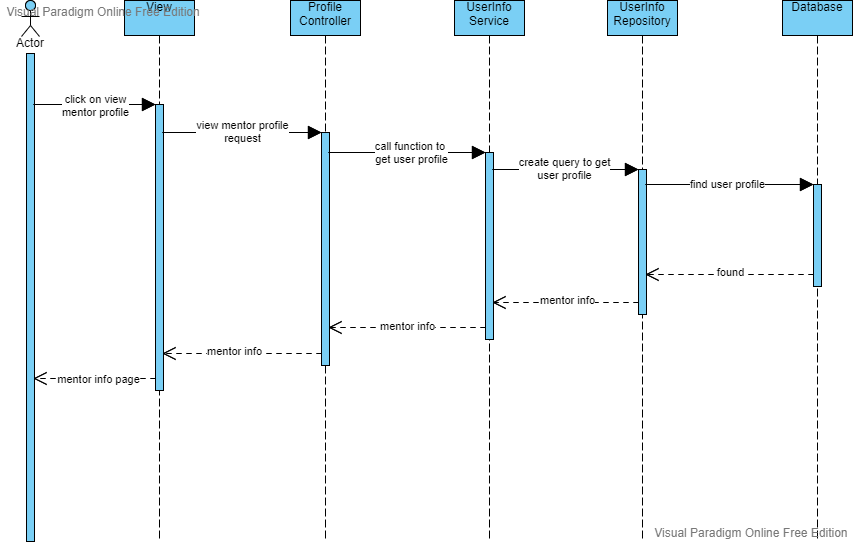
**Sequence Diagram for Update Profile**



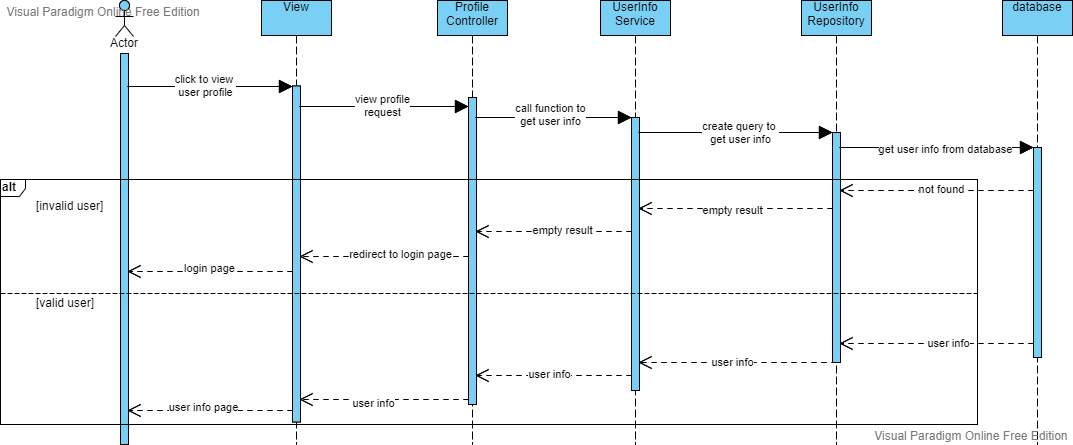
**Sequence Diagram for Update Password**

****

**Sequence diagram for View Mentor Profile**

****

**Sequence diagram for View User Profile**

****

### Database queries

* findById(id)

SELECT \* FROM UserInfo WHERE uId=@id

* save(userInfo)

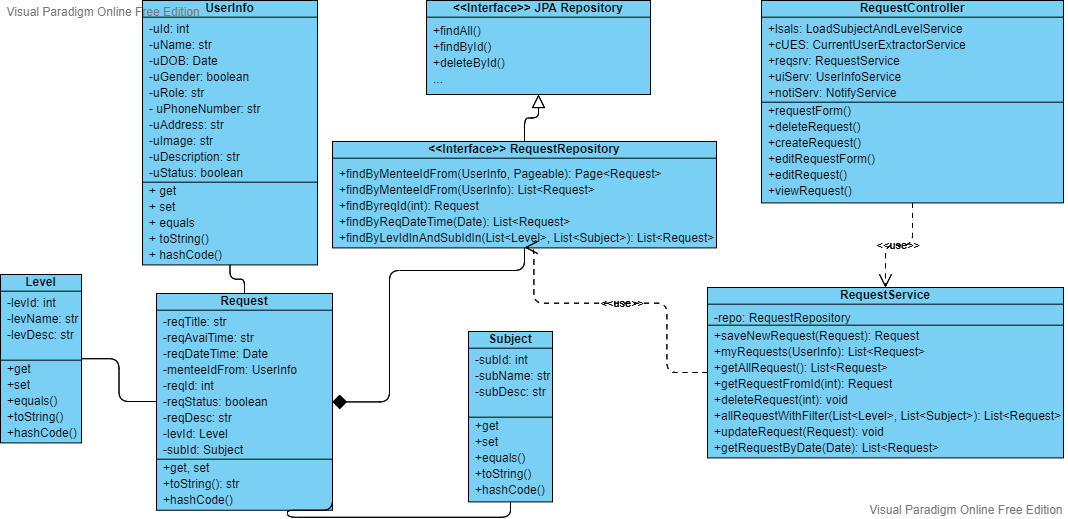
UPDATE UserInfo

SET uName=@uName, uDOB=@uDOB, uGender=@uGender, uRole=@uRole, uPhoneNumber=@uPhoneNumber, uAddress=@uAddress, uDescription=@uDescription, uStatus=@uStatus

WHERE uId=@uId

## 4. Request

### Class Diagram



### Class Specifications

#### RequestService

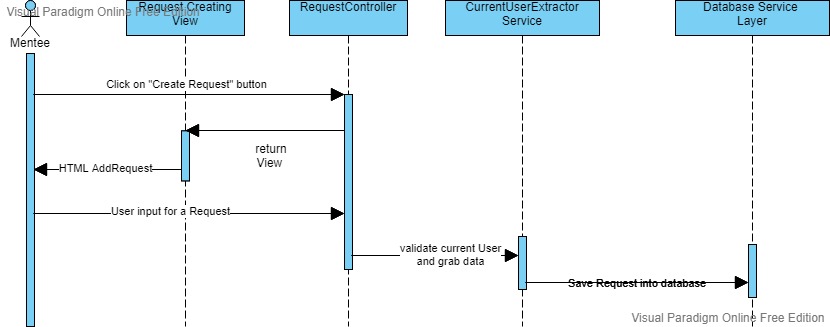
| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | saveNewRequest | save new request into database |
| 02 | myRequest | get request list of an user |
| 03 | getAllRequest | get all request in the database |
| 04 | getRequestFromId | get a request with provided id |
| 05 | deleteRequest | delete a request with provided id |
| 06 | allRequestWithFilter | get request list with level and subject filter |
| 07 | updateRequest | update an existed request |
| 08 | getRequestByDate | get a request created/updated in a provided date |

#### Request Controller

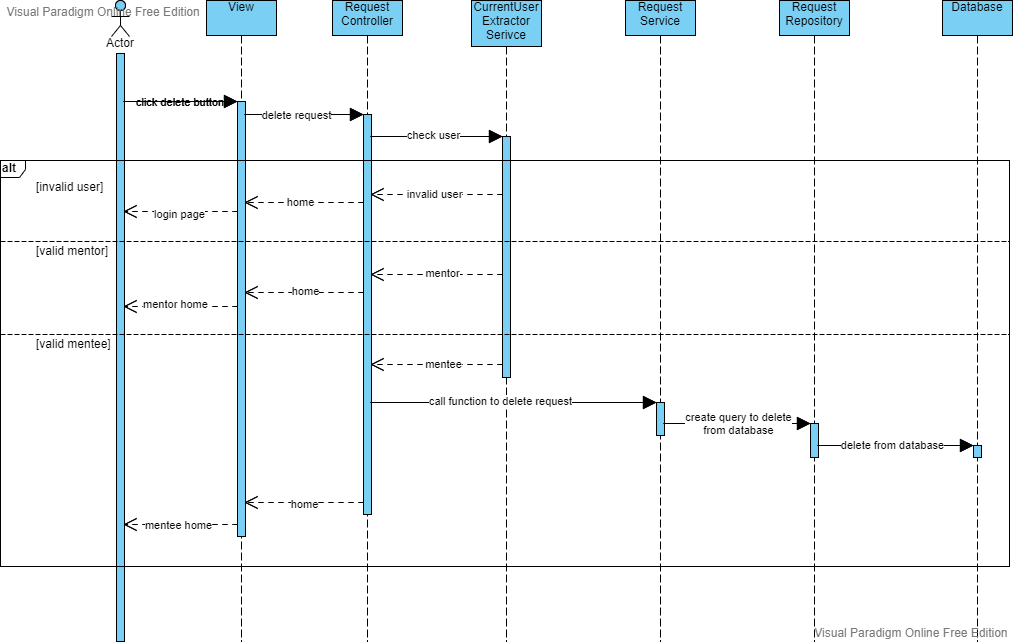
| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | requestForm | load create request template |
| 02 | deleteRequest | call function to delete a request |
| 03 | createRequest | call function to create a request |
| 04 | editRequestForm | load edit request template |
| 05 | editRequest | call function to update a request |
| 06 | viewRequest | call function to get a request detail |

### Sequence Diagram(s)

**Sequence Diagram for Create a Request**

****

**Sequence Diagram for Delete a Request**



### Database queries

* findByMenteeIdFrom(menteeId)

SELECT \* FROM Request WHERE menteeIdFrom=@menteeId

* findByreqId(reqId)

SELECT \* FROM Request WHERE reqId=@reqId

* findByReqDateTime(reqDateTime)

SELECT \* FROM Request WHERE reqDateTime=@reqDateTime

* findByLevIdInAndSubIdIn(levId, subId)

SELECT \* FROM Request WHERE levId In @levId And subId In @subId

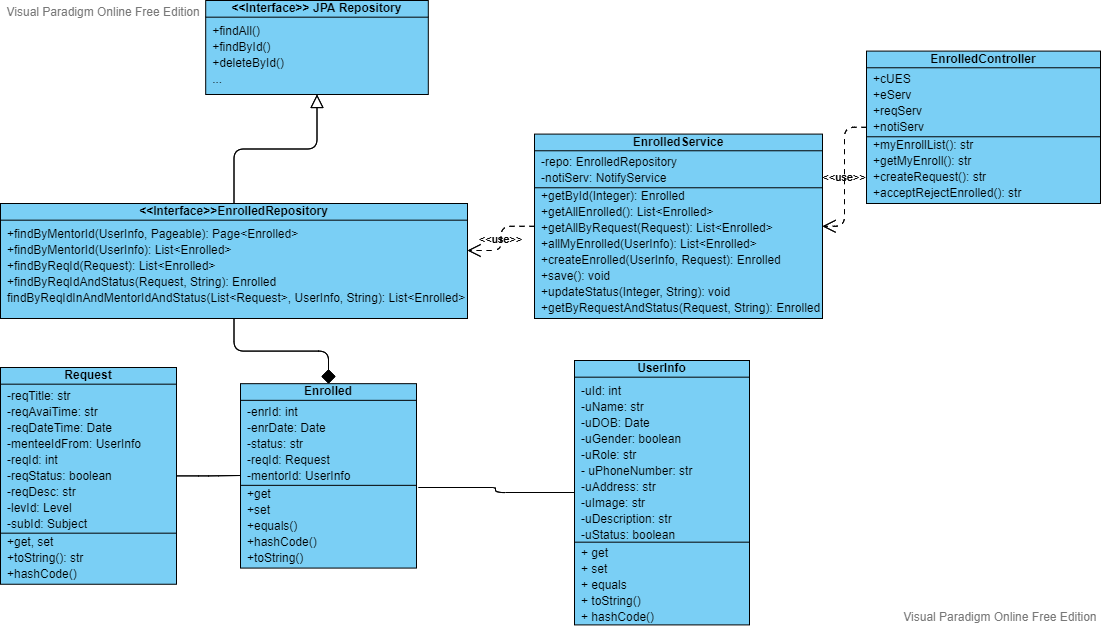
* save(request)

INSERT INTO Request(reqTitle, reqAvaiTime, reqDateTime, menteeIdFrom, reqId, reqStatus, reqDesc, levId, subId)

VALUES (@reqTitle, @reqAvaiTime, @reqDateTime, @menteeIdFrom, @reqId, @reqStatus, @reqDesc, @levId, @subId)

## 5. Enrolled

### Class Diagram



### Class Specifications

#### EnrolledService

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getById | get an Enrolled by its ID |
| 02 | getAllEnrolled | get all Enrolled |
| 03 | getAllByRequest | get all Enrolled of a Request |
| 04 | allMyEnrolled | get all Enrolled of an User |
| 05 | createEnrolled | create an Enrolled |
| 06 | save | save an Enrolled |
| 07 | updateStatus | update status of an Enrolled |
| 08 | getByRequestAndStatus | get an Enrolled by its Request and Status |

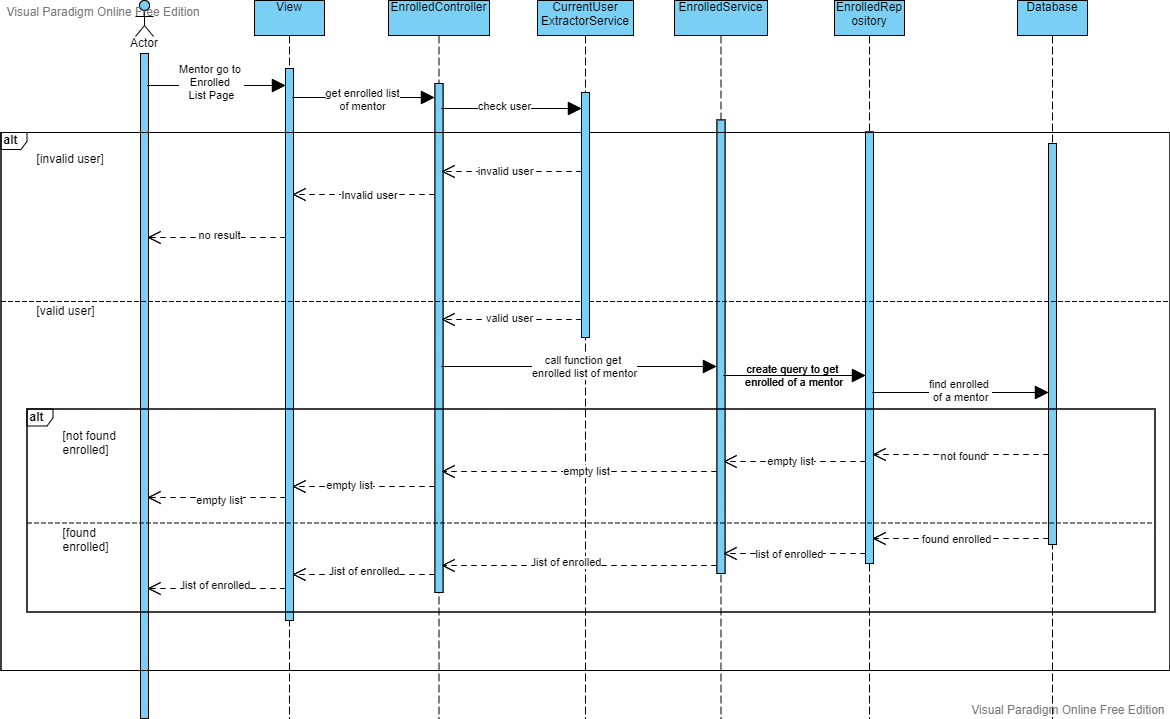
#### 

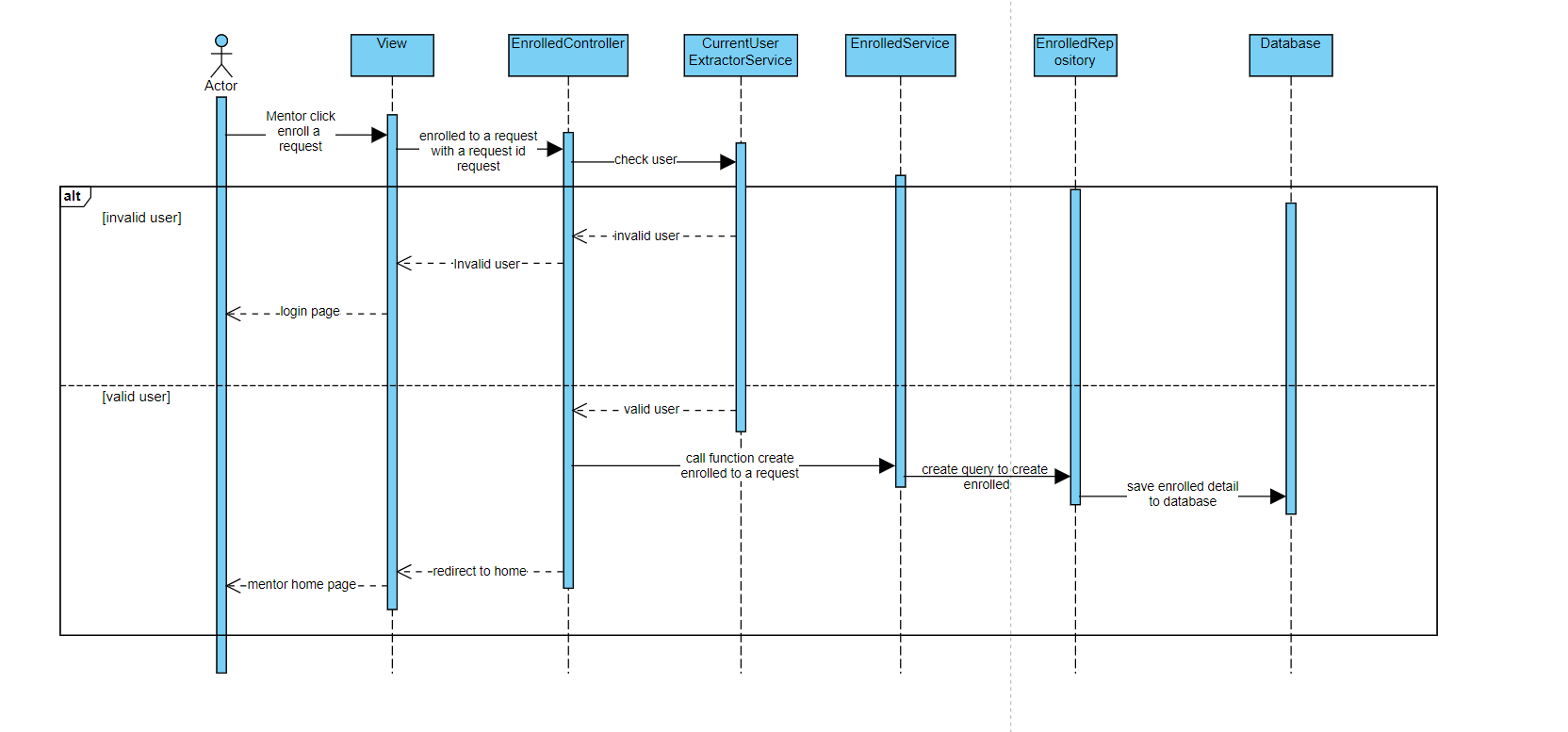
#### EnrolledController

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | myEnrollList | get all Enrolled of a Mentor |
| 02 | getMyEnroll | get all Enrolled to a Request |
| 03 | createRequest | enroll to a Request |
| 04 | acceptRejectEnrolled | set status accept or reject to an Enrolled |

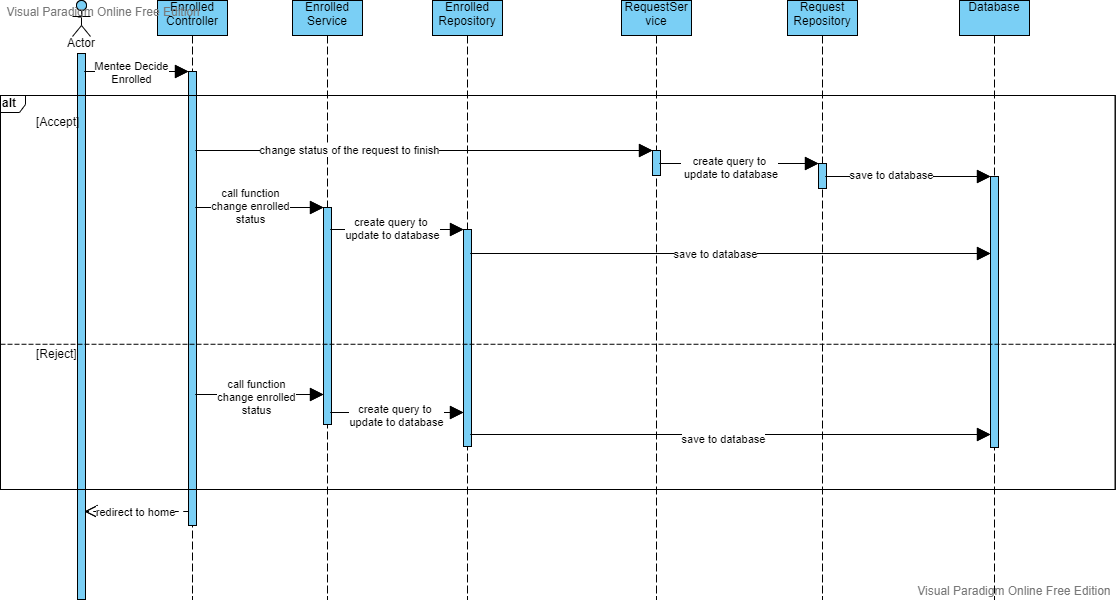
### Sequence Diagram(s)

### Sequence Diagram for Mentee View Enrolled List of a RequestSequence Diagram for Mentee View Enrolled List of a Request

**Sequence Diagram for Mentor Enrolled a Request**

****

**Sequence Diagram for Decide a Request**

****

### Database queries

* findAll:

SELECT \* FROM Enrolled

* findById(id):

SELECT \* FROM Enrolled WHERE enrId=@id

* findByMentorId(mentorId):

SELECT \* FROM Enrolled WHERE mentorId=@mentorId

* findByReqId(reqId):

SELECT \* FROM Enrolled WHERE reqId=@reqId

* findByReqIdAndStatus(reqId, status):

SELECT \* FROM Enrolled WHERE reqId=@reqId and status=@status

* findByReqIdAndMentorIdAndStatus(reqId, mentorId, status):

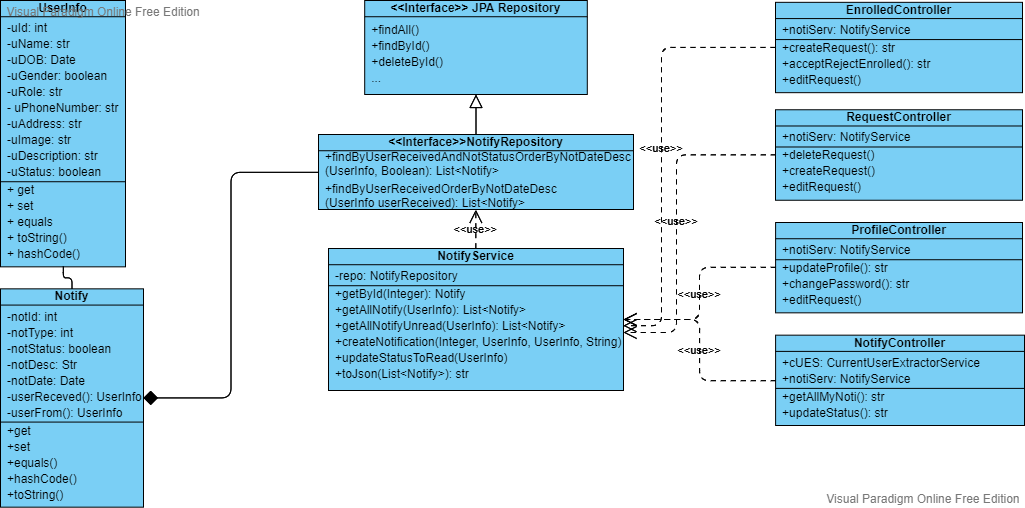
SELECT \* FROM Enrolled WHERE reqId=@reqId and mentorId=@mentorId and status=@status

* save(enroll)

INSERT INTO Enrolled(enrDate, status, reqId, mentorId) VALUES(@enrDate, @status, @reqId, @mentorId)

## 6. Notification

### Class Diagram



### Class Specifications

#### Notify: THIS CLASS STORE INFORMATION OF A NOTIFY

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getById | get a specific notification |
| 02 | getAllNotify | get all notification of an User |
| 03 | createNotification | create a Notification |
| 04 | updateStatusToRead | update all notification of an user to status read |
| 05 | toJson | transform a List of Notification into JSON string |

#### 

#### NotifyService

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getById | get a specific notification |
| 02 | getAllNotify | get all notification of an User |
| 03 | createNotification | create a Notification |
| 04 | updateStatusToRead | update all notification of an user to status read |
| 05 | toJson | transform a List of Notification into JSON string |

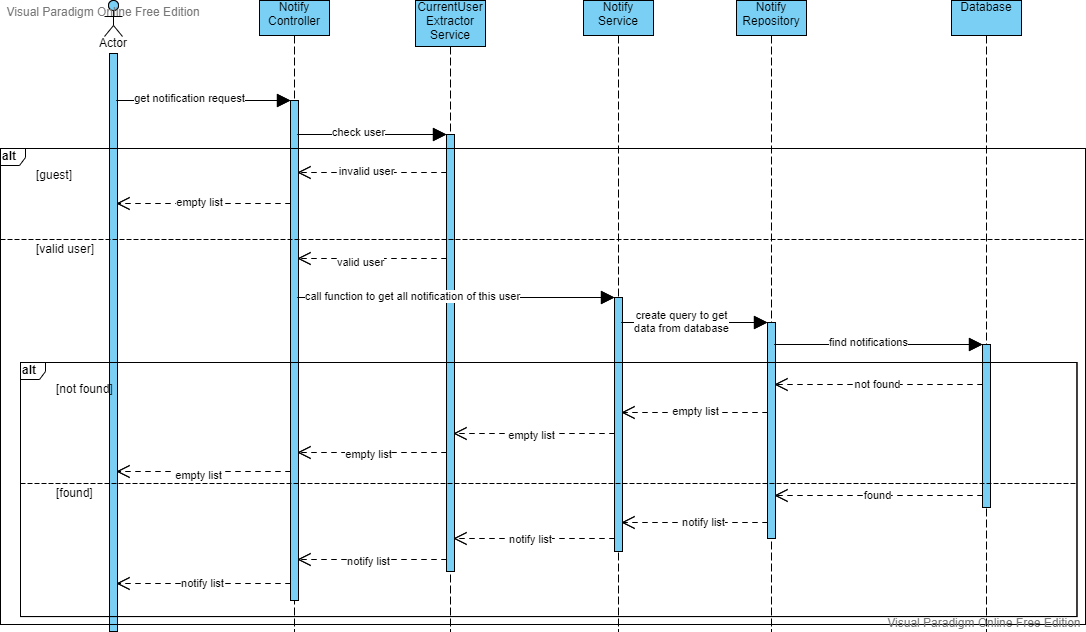
#### 

#### NotifyController

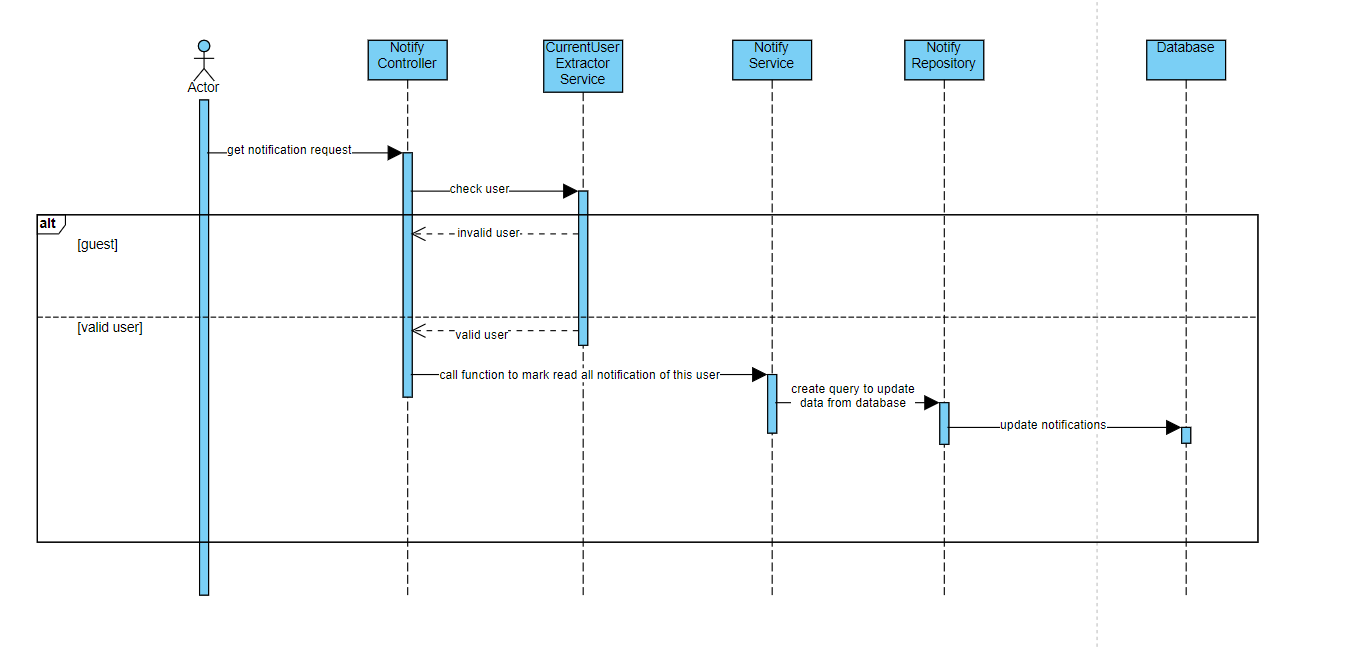
| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getAllMyNoti | get all notification of an user |
| 02 | updateStatus | update all notification of an user to status read |

### Sequence Diagram(s)

**Sequence Diagram for get All User Notification**



**Sequence Diagram for update Notification to Read**



### Database queries

* findByUserReceivedAndNotStatusOrderByNotDateDesc(userReceived, notStatus)

SELECT \* FROM Notify WHERE userReceived=@userReceived and notStatus=@notStatus ORDER BY notDate DESC

* findByUserReceivedOrderByNotDateDesc(userReceived)

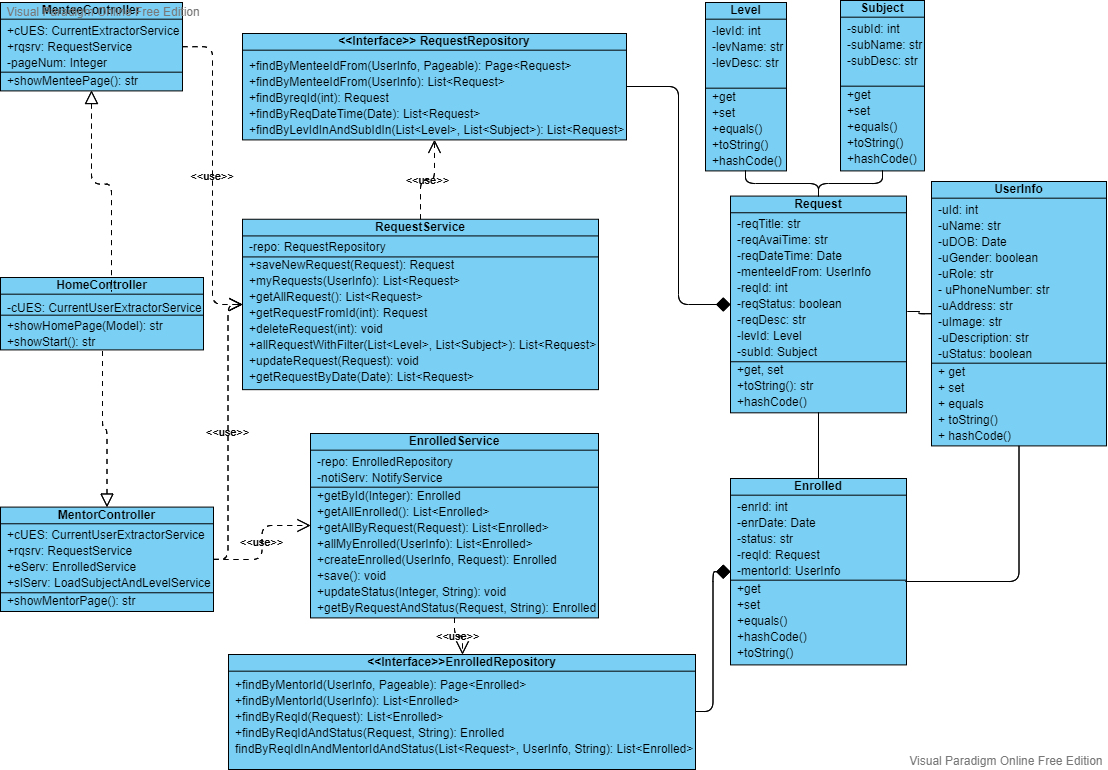
SELECT \* FROM Notify WHERE userReceived=@userReceived ORDER BY notDate DESC

save(notify)

INSERT INTO Notify(notType, notStatus, notDate, notDesc, userReceived, userFrom) VALUES (@notType, @notStatus, @notDate, @notDesc, @userReceived, @userFrom)

## 7. Home

### Class Diagram



### Class Specification

***HomeController***

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | showHomePage | redirect user to homepage |
| 02 | showStart | redirect user to landing page |

**MenteeController**

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | showMenteePage | redirect user to mentee homepage |

#### 

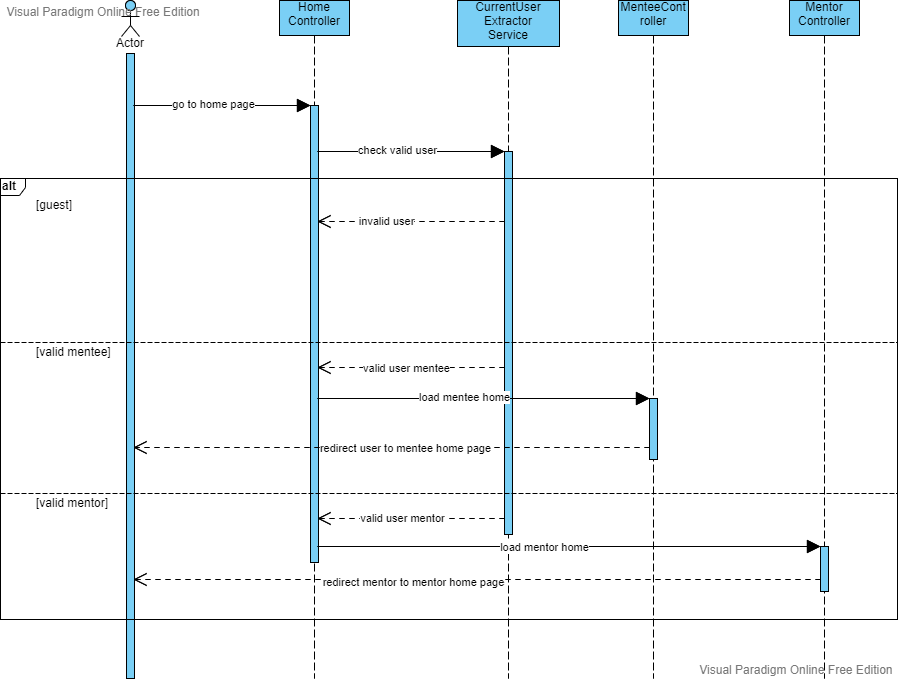
**MentorController**

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | showMentorPage | redirect user to mentor homepage |

#### 

#### 

### Sequence Diagram(s)

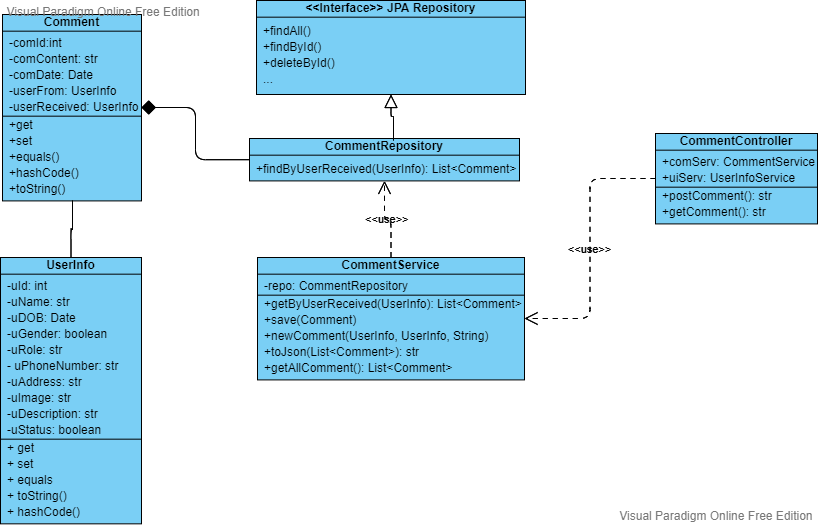


### Database queries

N/A

## 8. Comment

### Class Diagram



### Class Specification

***CommentService***

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getByUserReceived | get all Comments about an User |
| 02 | save | save a comment |
| 03 | newComment | create a comment |
| 04 | getAllComment | get all comments |
| 05 | toJson | transform List of comments into JSON string |

#### 

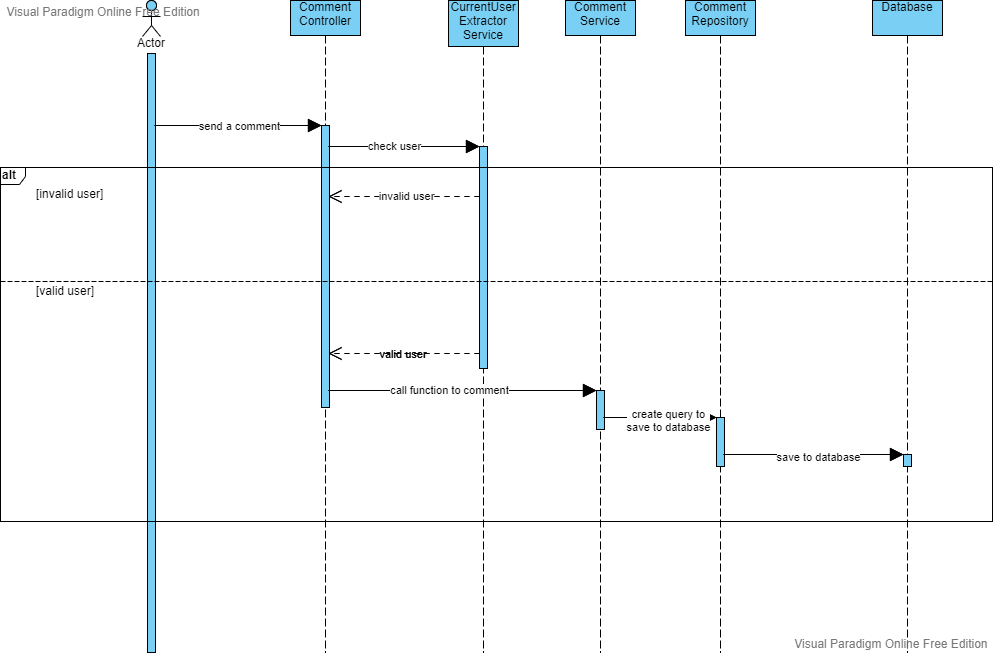
**CommentController**

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | postComment | comment about an User |
| 02 | getComment | get all Comments about an User |

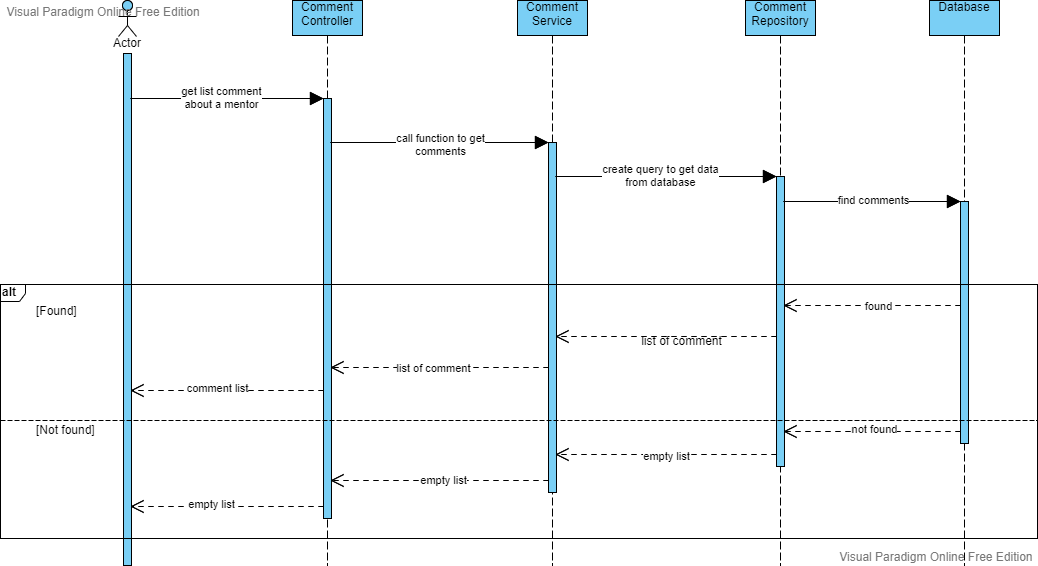
#### 

### Sequence Diagram(s)

**Sequence Diagram for Post a Comment**



**Sequence Diagram for get all comment about an User**



### Database queries

* findByUserReceived(userReceived):

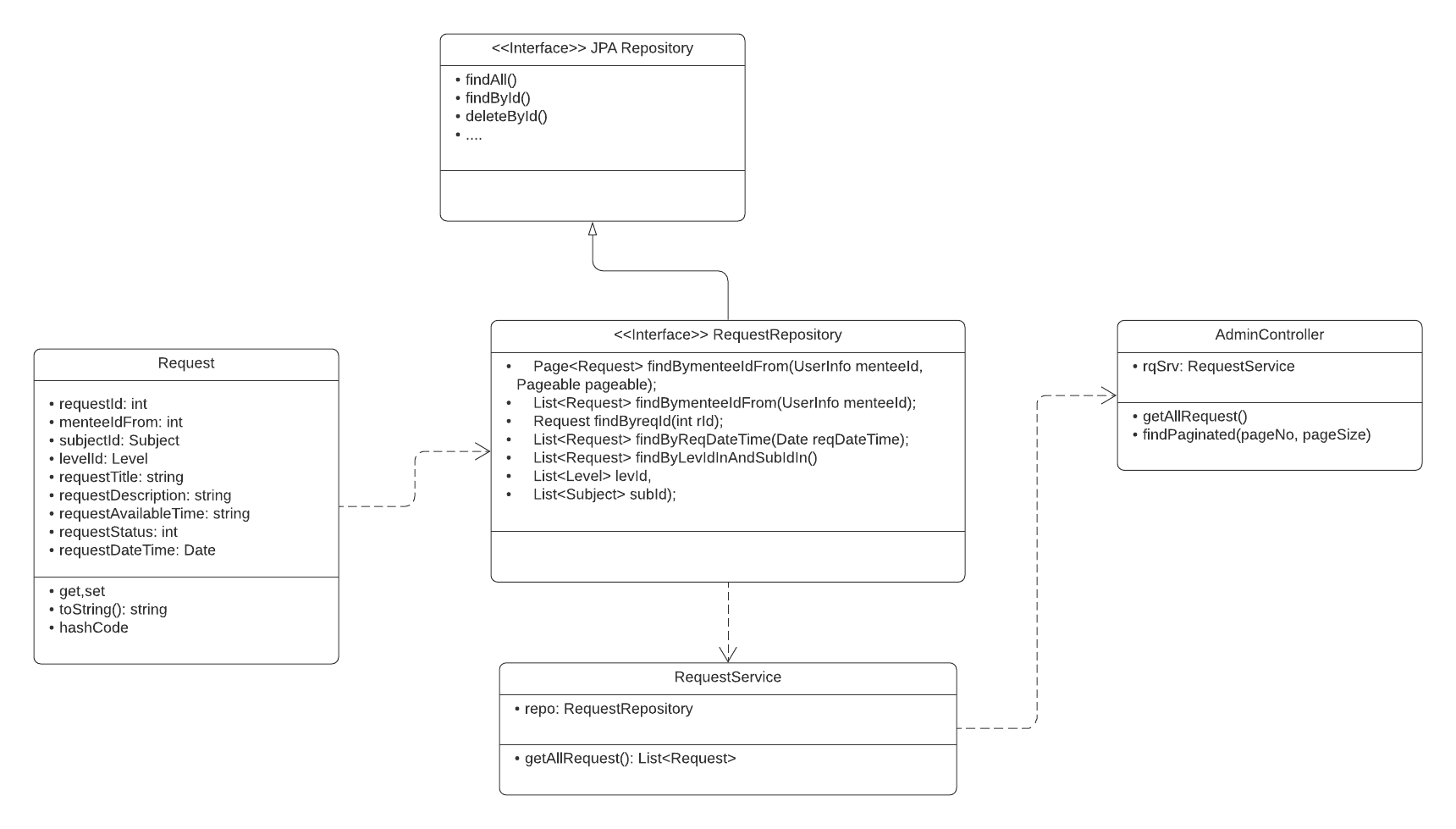
SELECT \* FROM Comment WHERE userReceived=@userReceived

* save(comment)

INSERT INTO Comment(comContent, comDate, userFrom, userReceived) VALUES (@comContent, @comDate, @userFrom, @userReceived)

## 9. Admin view all list request

### Class Diagram

****

### Class Specification

Request Class: This class store information about request

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getReqId() | get request ID from a request object |
| 02 | setReqId | set request ID for a request object |
| 03 | getReqStatus() | get request status from a request object |
| 04 | setReqStatus | set request status for a request object |
| 05 | getReqDesc() | get request description from a request object |
| 06 | setReqDesc() | set request description for a request object |
| 07 | getLevId() | get level from a request object |
| 08 | setLevId() | set level for a request object |
| 09 | getSubId() | get subject from a request object |
| 10 | setSubId() | set subject for a request object |
| 11 | getReqTitle() | get request title from a request object |
| 12 | setReqTitle() | set request title for a request object |
| 13 | getReqAvaiTime() | get request available time from a request object |
| 14 | setReqAvaiTime() | set request available time for a request object |
| 15 | getReqDateTime() | get request date time |
| 16 | setReqDateTime() | set request date time for a request object |
| 17 | getMenteeIdFrom() | get userinfo from a request object |
| 18 | setMenteeIdFrom() | set userinfo for a request object |
| 19 | toString() | get to string from a request object |

RequestRepository: this is an interface that helps connect with JPA

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | findBymenteeIdFrom() | using hibernate to connect to the database and get request by mentee user id |
| 02 | findByreqId() | using hibernate to connect to the database and get request by request id |
| 03 | findByReqDateTime() | using hibernate to connect to the database and get request by request date time |
| 04 | findByLevIdInAndSubIdIn() | using hibernate to connect to the database and get request by level id |

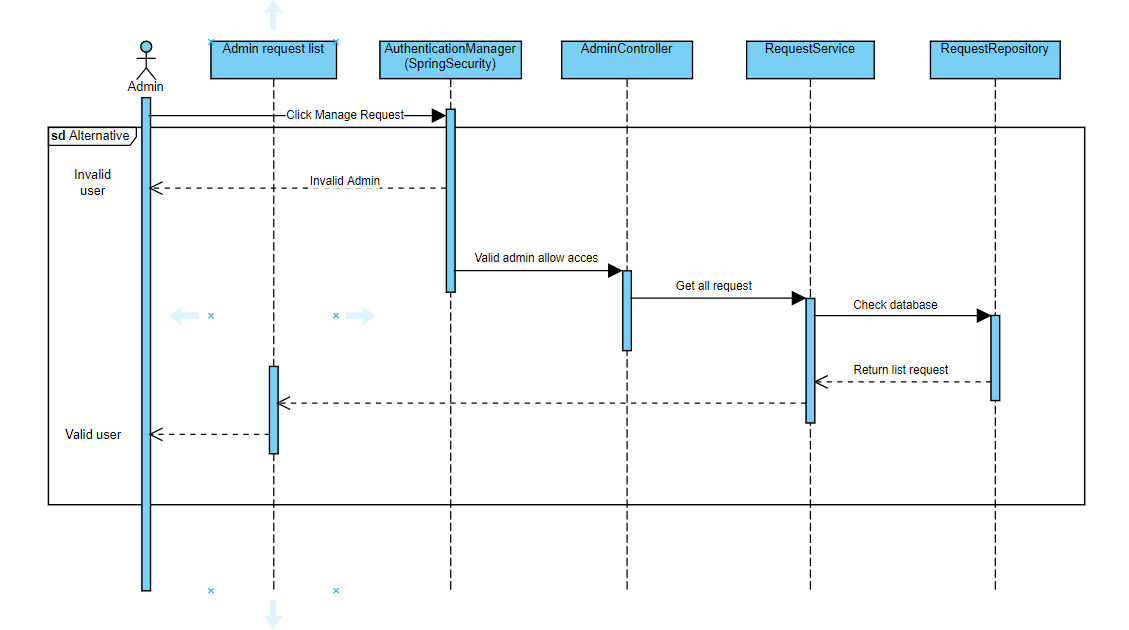
RequestService: implement the request repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | listAllMyByPage() | get the paginated list of request |
| 02 | listAllByPage() | get all the request list after paging |

AdminController

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | findPaginated() | get request ID |

### Sequence Diagram(s)



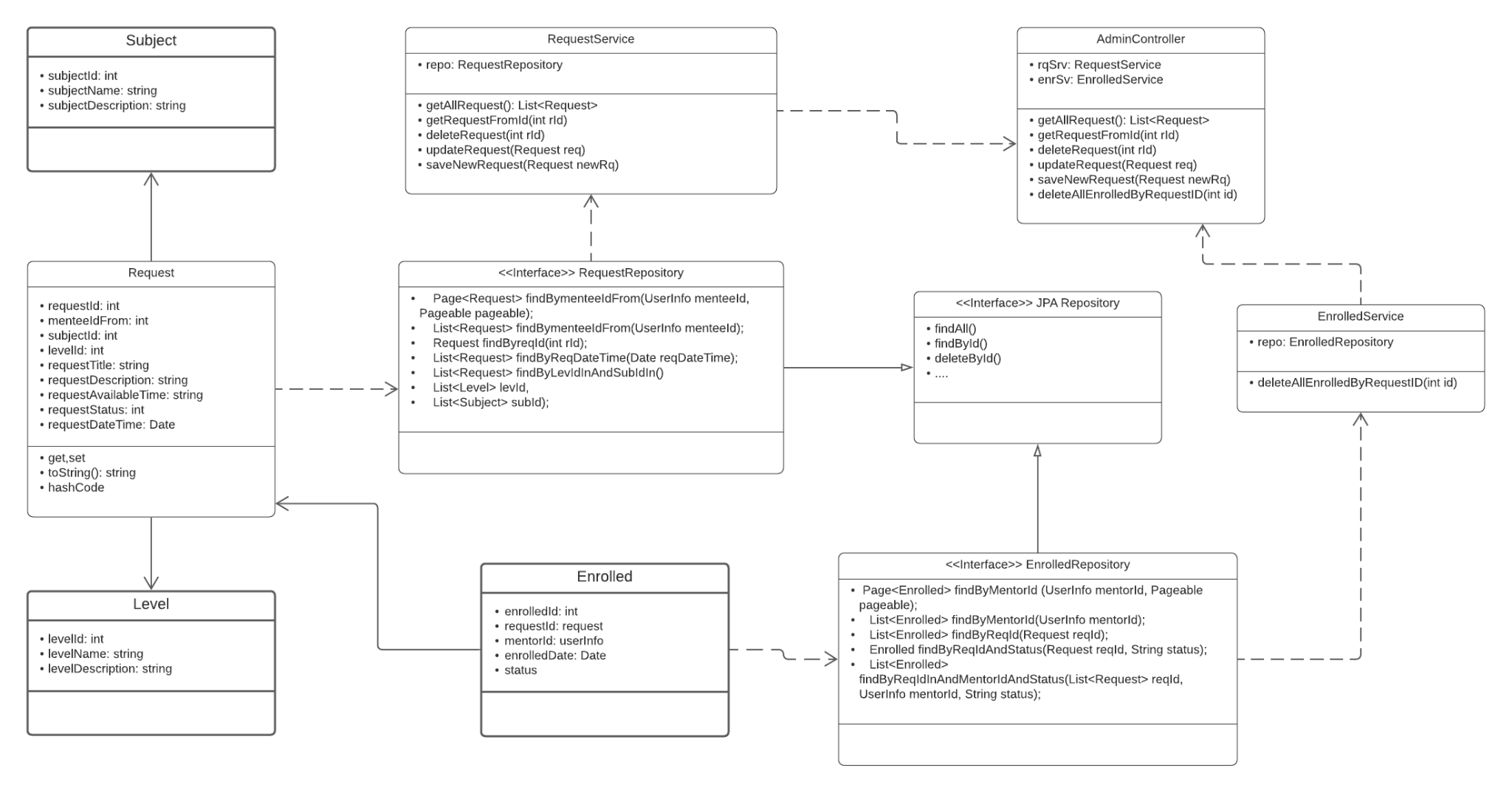
### Database queries

- repo.findAll() : select \* from Request

- repo.findPaginated(): select top 15 from Request

## 10. Admin manage requests

### Class Diagram



### Class Specification

Request Class: This class store information about request

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getReqId() | get request ID from a request object |
| 02 | setReqId | set request ID for a request object |
| 03 | getReqStatus() | get request status from a request object |
| 04 | setReqStatus | set request status for a request object |
| 05 | getReqDesc() | get request description from a request object |
| 06 | setReqDesc() | set request description for a request object |
| 07 | getLevId() | get level from a request object |
| 08 | setLevId() | set level for a request object |
| 09 | getSubId() | get subject from a request object |
| 10 | setSubId() | set subject for a request object |
| 11 | getReqTitle() | get request title from a request object |
| 12 | setReqTitle() | set request title for a request object |
| 13 | getReqAvaiTime() | get request available time from a request object |
| 14 | setReqAvaiTime() | set request available time for a request object |
| 15 | getReqDateTime() | get request date time |
| 16 | setReqDateTime() | set request date time for a request object |
| 17 | getMenteeIdFrom() | get userinfo from a request object |
| 18 | setMenteeIdFrom() | set userinfo for a request object |
| 19 | toString() | get to string from a request object |

Level Class: This class store the information about object level

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getLevId() | get level ID from a level object |
| 02 | setLevId() | set level ID for a level object |
| 03 | getLevName() | get level name from a level object |
| 04 | setLevName() | set level name for a level object |
| 05 | getLevDesc() | get level description from a level object |
| 06 | setLevDesc() | set level description for a level object |
| 07 | toString() | get to string |

Subject Class: This class store the information about object subject

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getSubId() | get level ID |
| 02 | setSubId() | set level ID |
| 03 | getSubName()() | get level name |
| 04 | setSubName() | set level name |
| 05 | getSubDesc() | get level description |
| 06 | setSubDesc() | set level description |
| 07 | toString() | get to string |

Enrolled Class: This class store the information about object enrolled

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getEnrId() | get request ID |
| 02 | setEnrId() | set request ID |
| 03 | getEnrDate() | get request status |
| 04 | setEnrDate() | set request status |
| 05 | getReqId() | get request description |
| 06 | setReqId() | set request description |
| 07 | getMentorId() | get level |
| 08 | setMentorId() | set level |
| 09 | getStatus() | get subject |
| 10 | setStatus() | set subject |
| 11 | toString() | get request title |

RequestService: implement the request repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | listAllMyByPage() | get the paginated list of request |
| 02 | listAllByPage() | get all the request list after paging |
| 03 | saveNewRequest() | save a new created request |
| 04 | deleteRequest() | delete a selected request |
| 05 | updateRequest() | update information in the selected request |

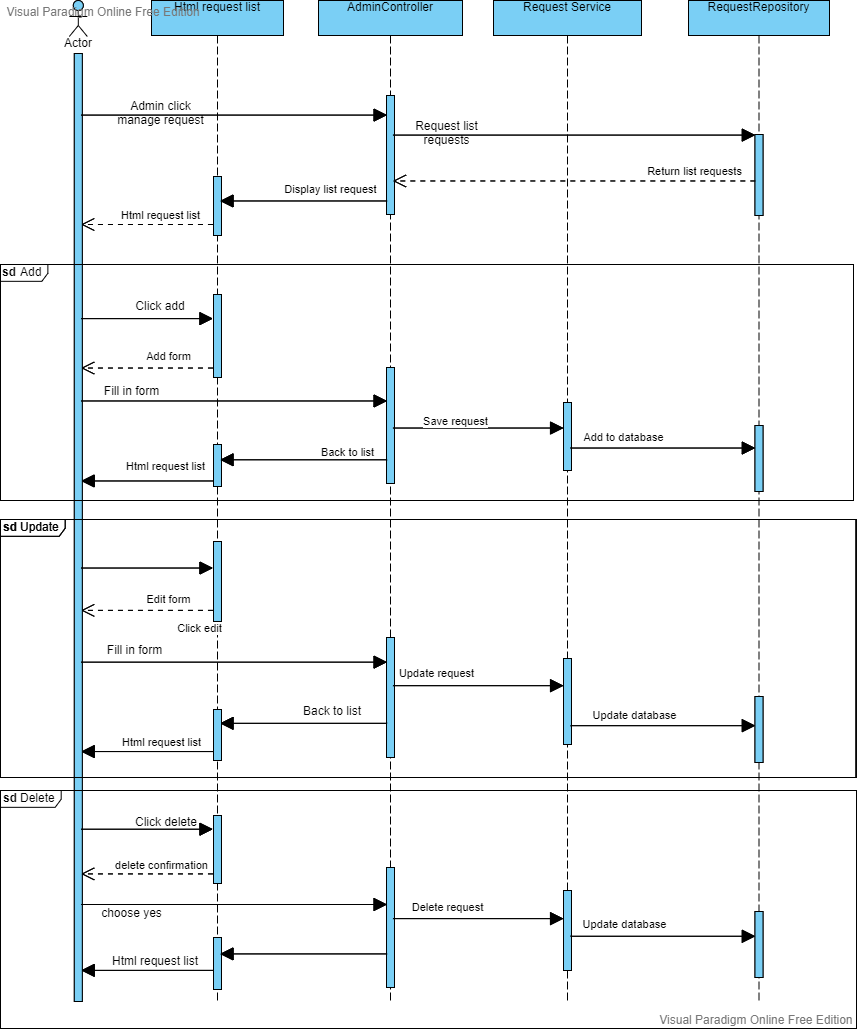
SubjectService: implement the subject repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getAllSubject() | get all the subjects contain in the database |
| 02 | findSubjectbyId(int id) | get the subject by subject id |
| 03 | getAllLevel() | get all the levels contain in the database |
| 04 | findLevelbyId(int id) | get the level by level id |

Enrolled Service:

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | deleteAllEnrolledByRequestID | delete all enrolled object in the database by request id |
| 02 | getAllEnrolledByRequestID() | get all enrolled object in the database by request id |

### Sequence Diagram(s)



### Database queries

rqSrv.saveNewRequest(request): Insert into request(menteeIdFrom,subId,levId,reqTitle,reqDesc) values(request.getMenteeIdfrom,request.getsubId(),..)

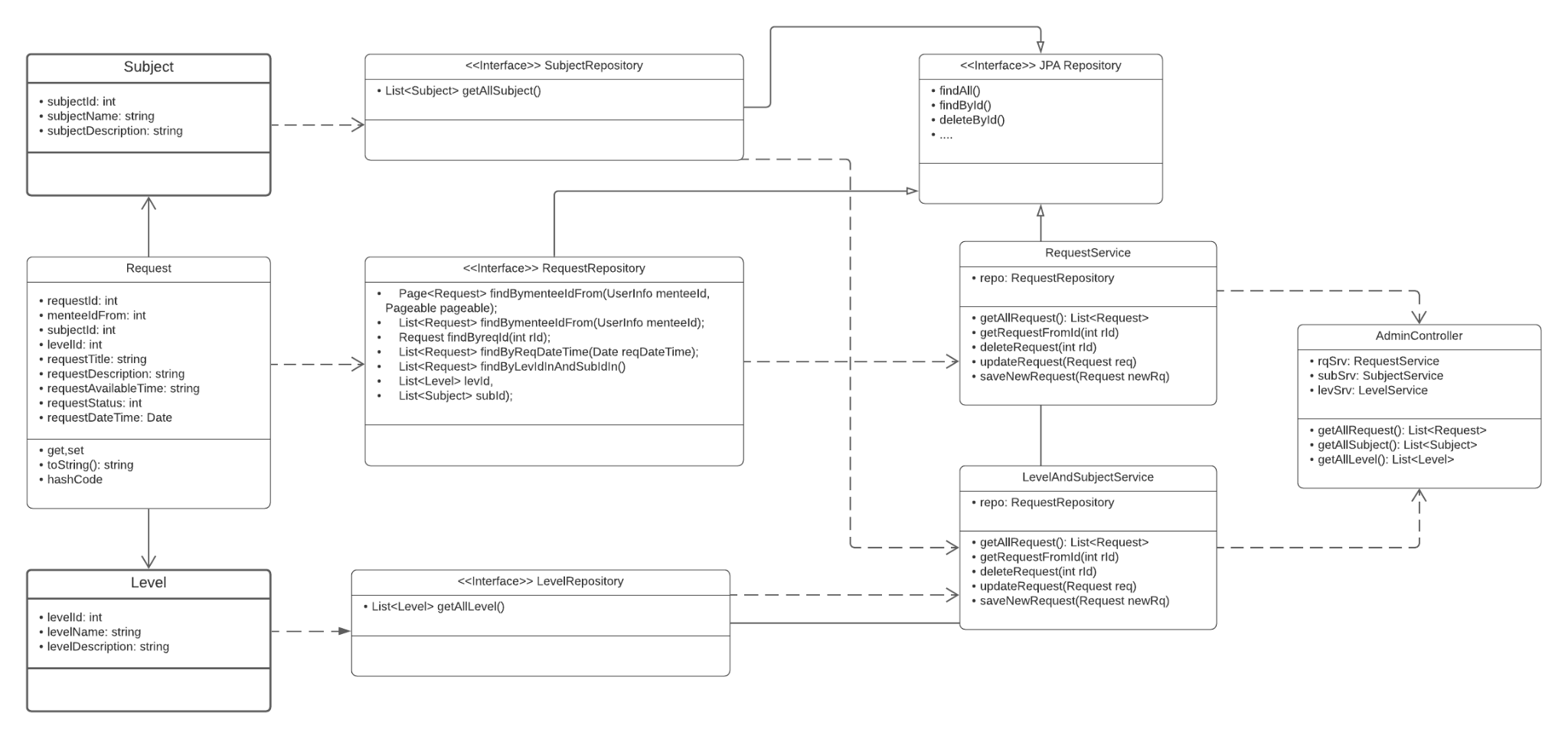
rqSrv.deleteRequest(request) :Delete from request where reqId = request.requestId

rqSrv.updateRequest(request):Update request set menteeIdFrom = request.getMenteeIdfrom, ...

enrSrv.

## 11. Admin filter requests

### Class Diagram



### Class Specification

Request Class: This class store information about request

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getReqId() | get request ID from a request object |
| 02 | setReqId | set request ID for a request object |
| 03 | getReqStatus() | get request status from a request object |
| 04 | setReqStatus | set request status for a request object |
| 05 | getReqDesc() | get request description from a request object |
| 06 | setReqDesc() | set request description for a request object |
| 07 | getLevId() | get level from a request object |
| 08 | setLevId() | set level for a request object |
| 09 | getSubId() | get subject from a request object |
| 10 | setSubId() | set subject for a request object |
| 11 | getReqTitle() | get request title from a request object |
| 12 | setReqTitle() | set request title for a request object |
| 13 | getReqAvaiTime() | get request available time from a request object |
| 14 | setReqAvaiTime() | set request available time for a request object |
| 15 | getReqDateTime() | get request date time |
| 16 | setReqDateTime() | set request date time for a request object |
| 17 | getMenteeIdFrom() | get userinfo from a request object |
| 18 | setMenteeIdFrom() | set userinfo for a request object |
| 19 | toString() | get to string from a request object |

Level Class: This class store the information about object level

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getLevId() | get level ID from a level object |
| 02 | setLevId() | set level ID for a level object |
| 03 | getLevName() | get level name from a level object |
| 04 | setLevName() | set level name for a level object |
| 05 | getLevDesc() | get level description from a level object |
| 06 | setLevDesc() | set level description for a level object |
| 07 | toString() | get to string |

Subject Class: This class store the information about object subject

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getSubId() | get level ID |
| 02 | setSubId() | set level ID |
| 03 | getSubName() | get level name |
| 04 | setSubName() | set level name |
| 05 | getSubDesc() | get level description |
| 06 | setSubDesc() | set level description |
| 07 | toString() | get to string |

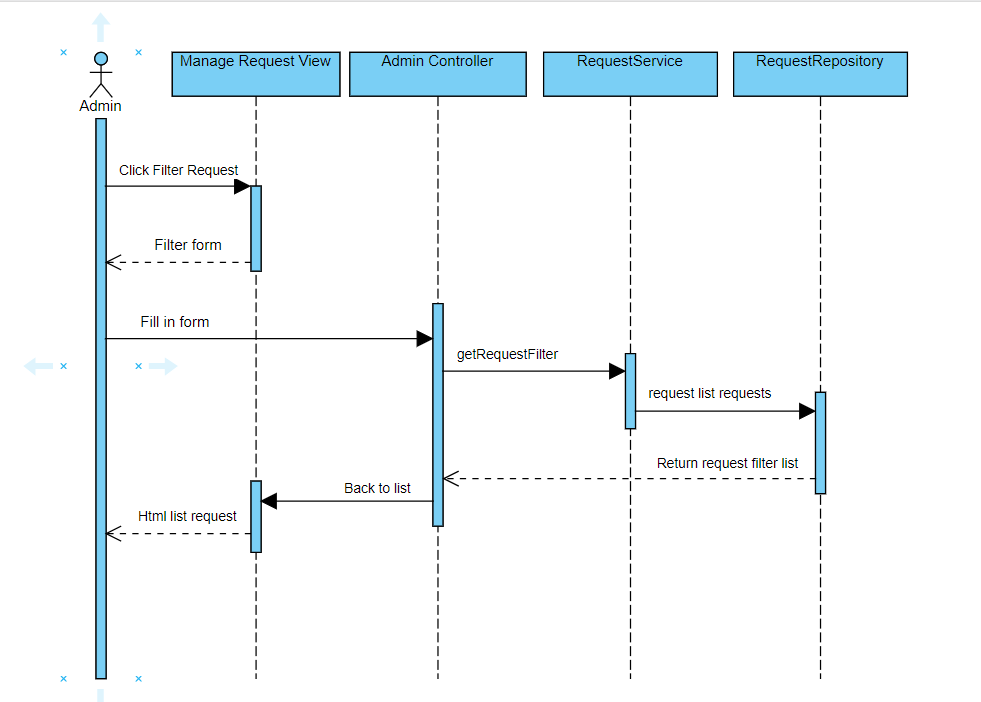
RequestService: implement the request repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | listAllMyByPage() | get the paginated list of request |
| 02 | listAllByPage() | get all the request list after paging |
| 03 | saveNewRequest() | save a new created request |
| 04 | deleteRequest() | delete a selected request |
| 05 | updateRequest() | update information in the selected request |

SubjectService: implement the subject repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getAllSubject() | get all the subjects contain in the database |
| 02 | findSubjectbyId(int id) | get the subject by subject id |
| 03 | getAllLevel() | get all the levels contain in the database |
| 04 | findLevelbyId(int id) | get the level by level id |

### Sequence Diagram(s)



### Database queries

select \* from request

select \* from subject

select \* from level

select \* from request where subjectId = ?

select \* from request where levelId = ?

select \* from request where levelId = ? and subjectId = ?

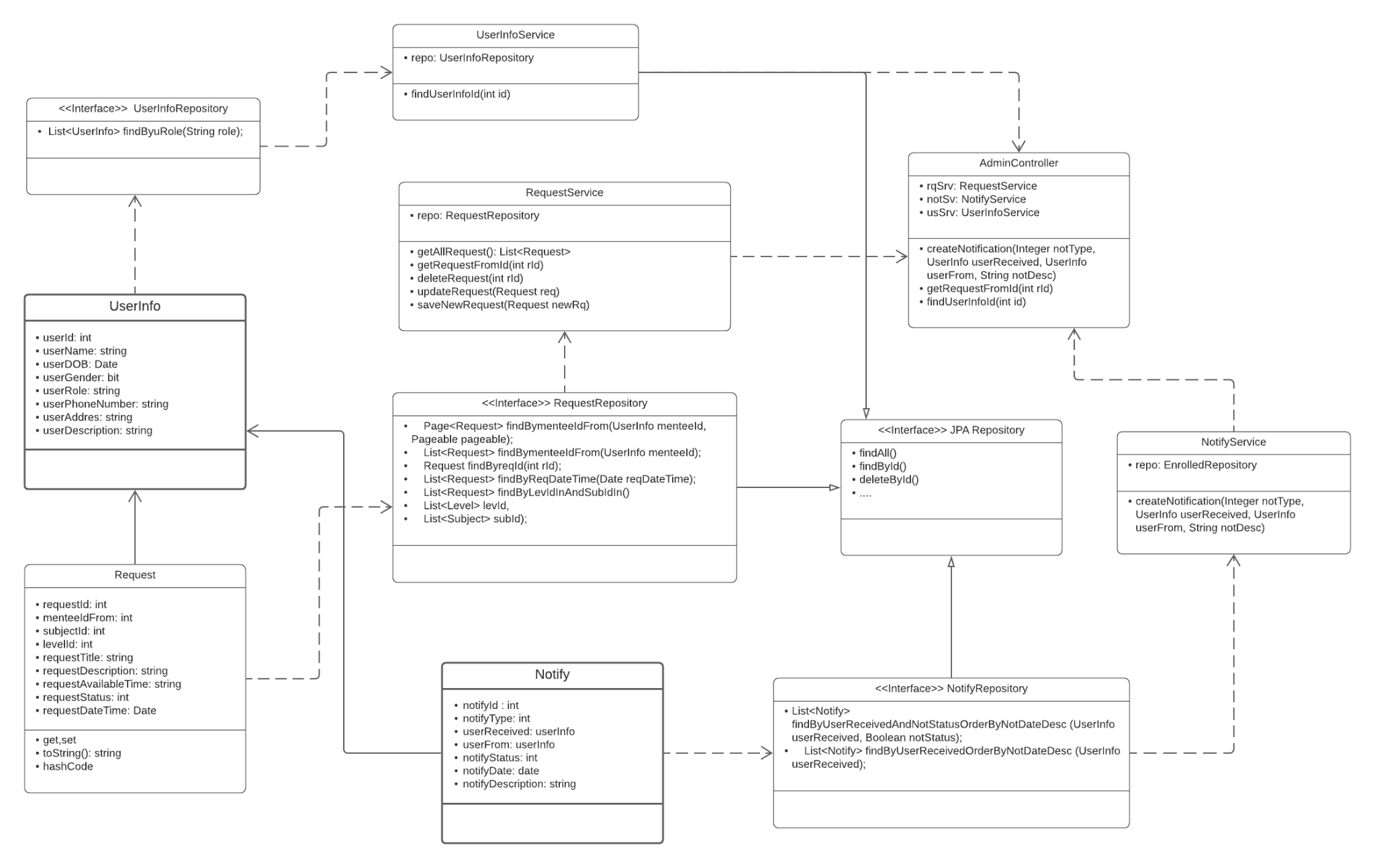
select \* from request where subjectId = ? and requestDateTime between ? and ?

select \* from request where levelId = ? and requestDateTime between ? and ?

select \* from request where levelId = ? and subjectId = ? and requestDateTime between ? and ?

## 12. Admin notify User

### Class Diagram



### Class Specification

Request Class: This class store information about request

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getReqId() | get request ID from a request object |
| 02 | setReqId | set request ID for a request object |
| 03 | getReqStatus() | get request status from a request object |
| 04 | setReqStatus | set request status for a request object |
| 05 | getReqDesc() | get request description from a request object |
| 06 | setReqDesc() | set request description for a request object |
| 07 | getLevId() | get level from a request object |
| 08 | setLevId() | set level for a request object |
| 09 | getSubId() | get subject from a request object |
| 10 | setSubId() | set subject for a request object |
| 11 | getReqTitle() | get request title from a request object |
| 12 | setReqTitle() | set request title for a request object |
| 13 | getReqAvaiTime() | get request available time from a request object |
| 14 | setReqAvaiTime() | set request available time for a request object |
| 15 | getReqDateTime() | get request date time |
| 16 | setReqDateTime() | set request date time for a request object |
| 17 | getMenteeIdFrom() | get userinfo from a request object |
| 18 | setMenteeIdFrom() | set userinfo for a request object |
| 19 | toString() | get to string from a request object |

UserInfo Class: This class store information about user

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getUId() | get user ID from a userInfo object |
| 02 | setUId() | set user ID for a request object |
| 03 | getUName() | get user name from a userInfo object |
| 04 | setUName() | set user name from a userInfo object |
| 05 | getUDOB() | get user date of birth from a userInfo object |
| 06 | setUDOB() | set user date of birth from a userInfo object |
| 07 | getUGender() | get user gender from a userInfo object |
| 08 | setUGender() | set user gender from a userInfo object |
| 09 | getURole() | get user role from a userInfo object |
| 10 | setURole() | set user role from a userInfo object |
| 11 | getUPhoneNumber() | get user phone number from a userInfo object |
| 12 | getUAddress() | get user address from a userInfo object |
| 13 | setUAddress() | set user address from a userInfo object |
| 14 | getUImage() | get user image from a userInfo object |
| 15 | setUImage() | set user image from a userInfo object |
| 16 | getUDescription() | get user description from a userInfo object |
| 17 | setUDescription() | set user description from a userInfo object |
| 18 | getUStatus() | get user status from a userInfo object |
| 19 | setUStatus() | set user status from a userInfo object |
| 20 | getLoginInfo() | get login info from a userInfo object |
| 21 | setLoginInfo() | set login info from a userInfo object |
| 22 | toString() | to string method |

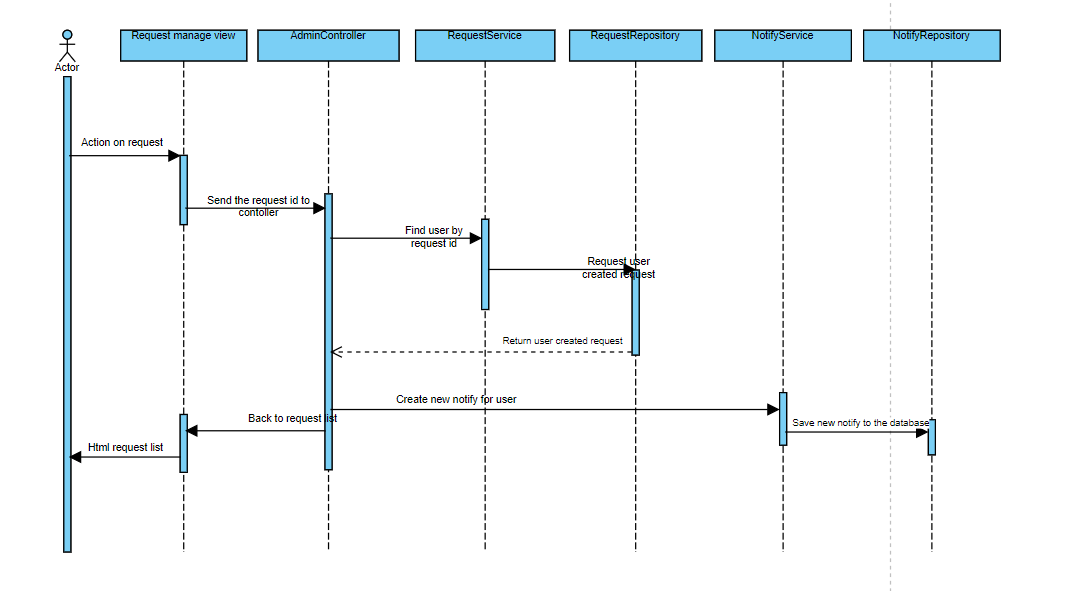
Notify Class: This class store information about user

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getNotId() | get request ID from a request object |
| 02 | setNotId() | set request ID for a request object |
| 03 | getNotType() | get request status from a request object |
| 04 | setNotType() | set request status for a request object |
| 05 | getNotStatus() | get request description from a request object |
| 06 | setNotStatus() | set request description for a request object |
| 07 | getNotDate() | get level from a request object |
| 08 | setNotDate() | set level for a request object |
| 09 | getNotDesc() | get subject from a request object |
| 10 | setNotDesc() | set subject for a request object |
| 11 | getUserReceived() | get request title from a request object |
| 12 | setUserReceived() | set request title for a request object |
| 13 | getUserFrom() | get request available time from a request object |
| 14 | setUserFrom() | set request available time for a request object |
| 19 | toString() | get to string from a request object |

Notify Service

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getAllNotify | get request ID from a request object |

### Sequence Diagram(s)

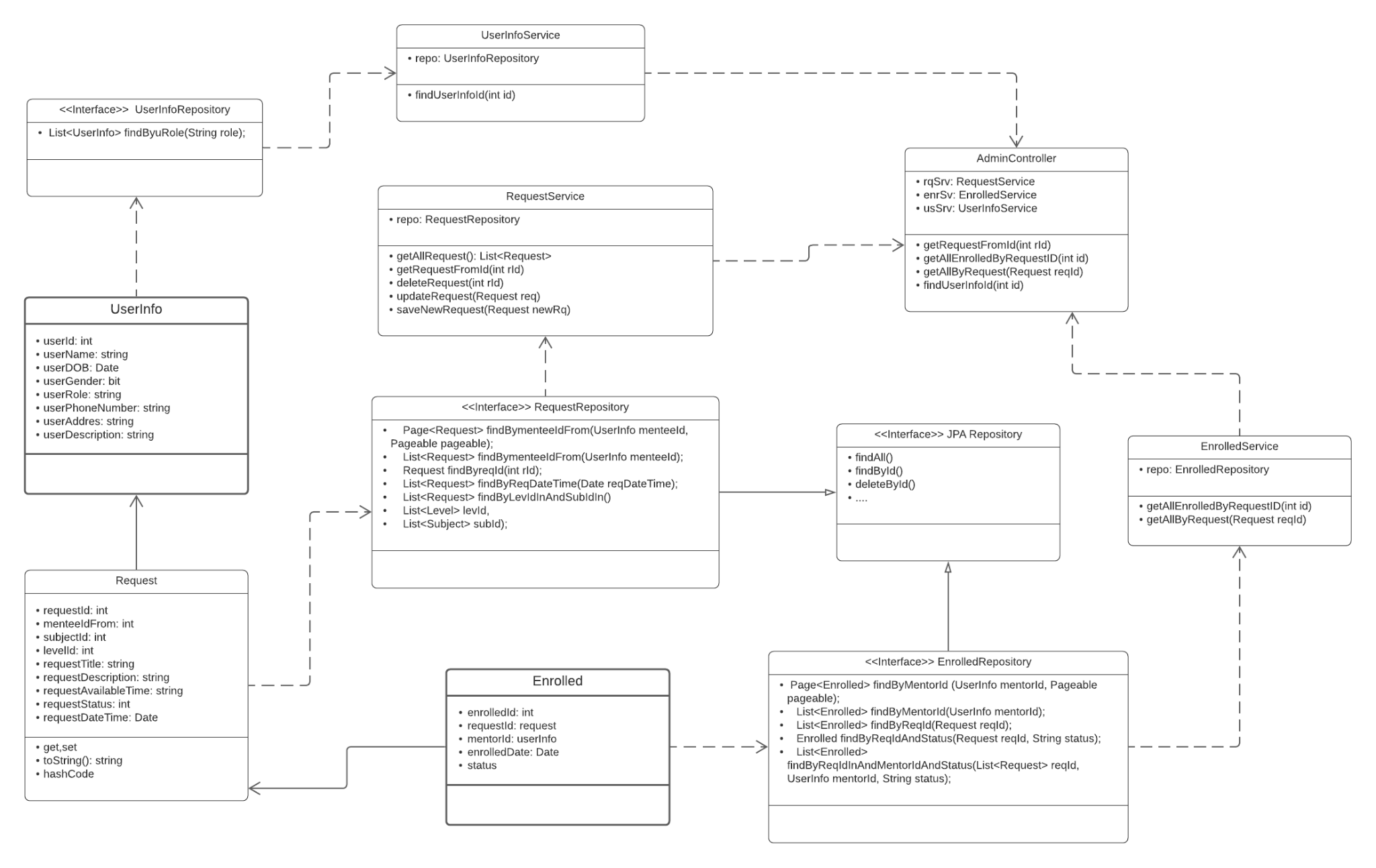


### Database queries

Insert into Notify (notType, userReceived, userFrom, notStatus, notDate, notDesc) values (?,?,?,?,?,?)

## 13. Admin request details

### Class Diagram



### Class Specification

Request Class: This class store information about request

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getReqId() | get request ID from a request object |
| 02 | setReqId | set request ID for a request object |
| 03 | getReqStatus() | get request status from a request object |
| 04 | setReqStatus | set request status for a request object |
| 05 | getReqDesc() | get request description from a request object |
| 06 | setReqDesc() | set request description for a request object |
| 07 | getLevId() | get level from a request object |
| 08 | setLevId() | set level for a request object |
| 09 | getSubId() | get subject from a request object |
| 10 | setSubId() | set subject for a request object |
| 11 | getReqTitle() | get request title from a request object |
| 12 | setReqTitle() | set request title for a request object |
| 13 | getReqAvaiTime() | get request available time from a request object |
| 14 | setReqAvaiTime() | set request available time for a request object |
| 15 | getReqDateTime() | get request date time |
| 16 | setReqDateTime() | set request date time for a request object |
| 17 | getMenteeIdFrom() | get userinfo from a request object |
| 18 | setMenteeIdFrom() | set userinfo for a request object |
| 19 | toString() | get to string from a request object |

UserInfo Class: This class store information about user

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getUId() | get user ID from a userInfo object |
| 02 | setUId() | set user ID for a request object |
| 03 | getUName() | get user name from a userInfo object |
| 04 | setUName() | set user name from a userInfo object |
| 05 | getUDOB() | get user date of birth from a userInfo object |
| 06 | setUDOB() | set user date of birth from a userInfo object |
| 07 | getUGender() | get user gender from a userInfo object |
| 08 | setUGender() | set user gender from a userInfo object |
| 09 | getURole() | get user role from a userInfo object |
| 10 | setURole() | set user role from a userInfo object |
| 11 | getUPhoneNumber() | get user phone number from a userInfo object |
| 12 | getUAddress() | get user address from a userInfo object |
| 13 | setUAddress() | set user address from a userInfo object |
| 14 | getUImage() | get user image from a userInfo object |
| 15 | setUImage() | set user image from a userInfo object |
| 16 | getUDescription() | get user description from a userInfo object |
| 17 | setUDescription() | set user description from a userInfo object |
| 18 | getUStatus() | get user status from a userInfo object |
| 19 | setUStatus() | set user status from a userInfo object |
| 20 | getLoginInfo() | get login info from a userInfo object |
| 21 | setLoginInfo() | set login info from a userInfo object |
| 22 | toString() | to string method |

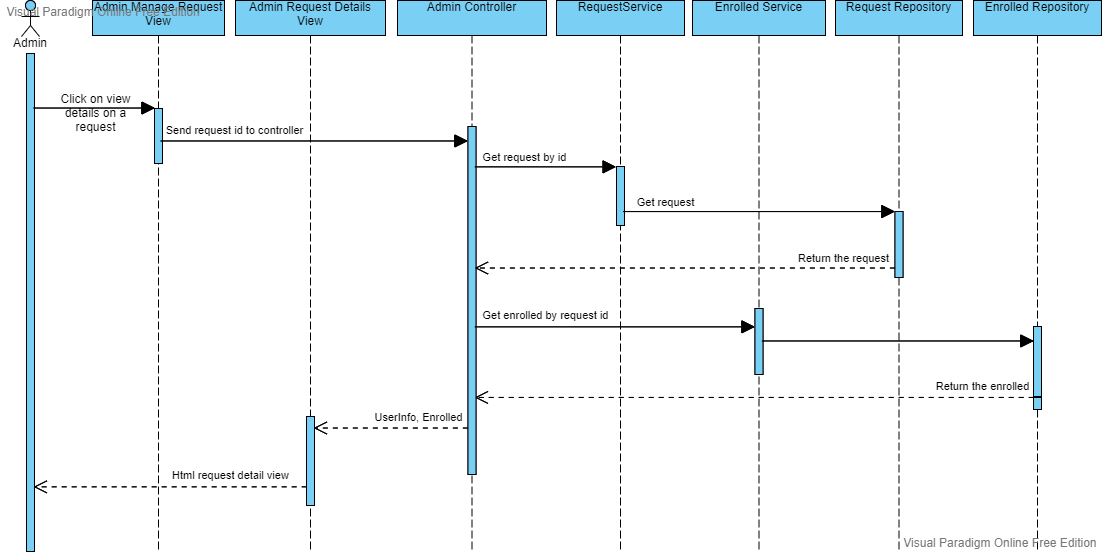
RequestService: implement the request repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | listAllMyByPage() | get the paginated list of request |
| 02 | listAllByPage() | get all the request list after paging |
| 03 | saveNewRequest() | save a new created request |
| 04 | deleteRequest() | delete a selected request |
| 05 | updateRequest() | update information in the selected request |

Enrolled Service:

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | deleteAllEnrolledByRequestID | delete all enrolled object in the database by request id |
| 02 | getAllEnrolledByRequestID() | get all enrolled object in the database by request id |

### Sequence Diagram(s)



### Database queries

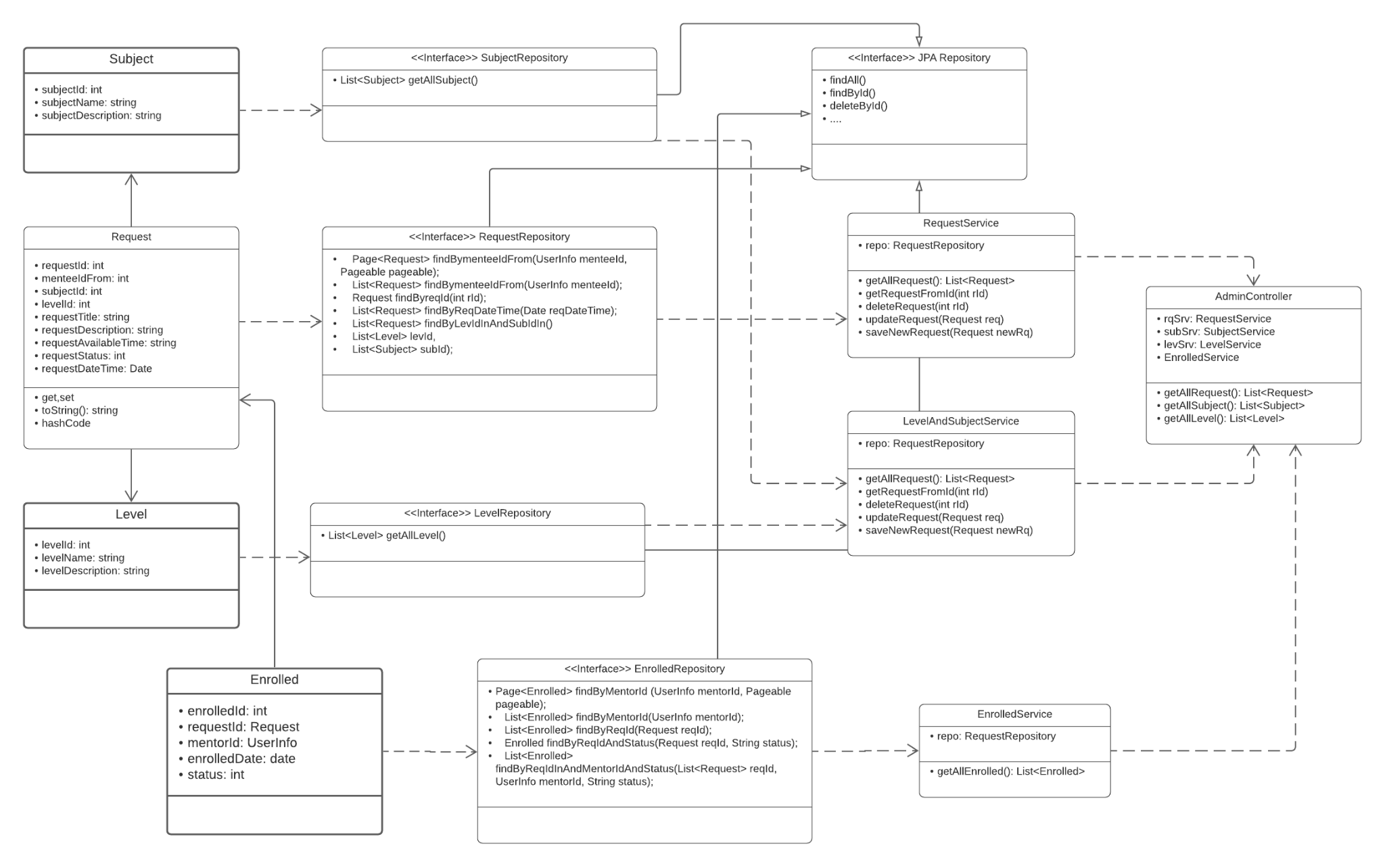
select \* from Request where requestId = ?

select \* from Enrolled where requestId = ?

select \* from Enrolled where status = “ACCEPT”

## 14. Admin Dashboard

### Class Diagram



### Class Specification

Request Class: This class store information about request

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getReqId() | get request ID from a request object |
| 02 | setReqId | set request ID for a request object |
| 03 | getReqStatus() | get request status from a request object |
| 04 | setReqStatus | set request status for a request object |
| 05 | getReqDesc() | get request description from a request object |
| 06 | setReqDesc() | set request description for a request object |
| 07 | getLevId() | get level from a request object |
| 08 | setLevId() | set level for a request object |
| 09 | getSubId() | get subject from a request object |
| 10 | setSubId() | set subject for a request object |
| 11 | getReqTitle() | get request title from a request object |
| 12 | setReqTitle() | set request title for a request object |
| 13 | getReqAvaiTime() | get request available time from a request object |
| 14 | setReqAvaiTime() | set request available time for a request object |
| 15 | getReqDateTime() | get request date time |
| 16 | setReqDateTime() | set request date time for a request object |
| 17 | getMenteeIdFrom() | get userinfo from a request object |
| 18 | setMenteeIdFrom() | set userinfo for a request object |
| 19 | toString() | get to string from a request object |

Level Class: This class store the information about object level

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getLevId() | get level ID from a level object |
| 02 | setLevId() | set level ID for a level object |
| 03 | getLevName() | get level name from a level object |
| 04 | setLevName() | set level name for a level object |
| 05 | getLevDesc() | get level description from a level object |
| 06 | setLevDesc() | set level description for a level object |
| 07 | toString() | get to string |

Subject Class: This class store the information about object subject

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getSubId() | get level ID |
| 02 | setSubId() | set level ID |
| 03 | getSubName()() | get level name |
| 04 | setSubName() | set level name |
| 05 | getSubDesc() | get level description |
| 06 | setSubDesc() | set level description |
| 07 | toString() | get to string |

Enrolled Class: This class store the information about object enrolled

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getEnrId() | get request ID |
| 02 | setEnrId() | set request ID |
| 03 | getEnrDate() | get request status |
| 04 | setEnrDate() | set request status |
| 05 | getReqId() | get request description |
| 06 | setReqId() | set request description |
| 07 | getMentorId() | get level |
| 08 | setMentorId() | set level |
| 09 | getStatus() | get subject |
| 10 | setStatus() | set subject |
| 11 | toString() | get request title |

RequestService: implement the request repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | listAllMyByPage() | get the paginated list of request |
| 02 | listAllByPage() | get all the request list after paging |
| 03 | saveNewRequest() | save a new created request |
| 04 | deleteRequest() | delete a selected request |
| 05 | updateRequest() | update information in the selected request |

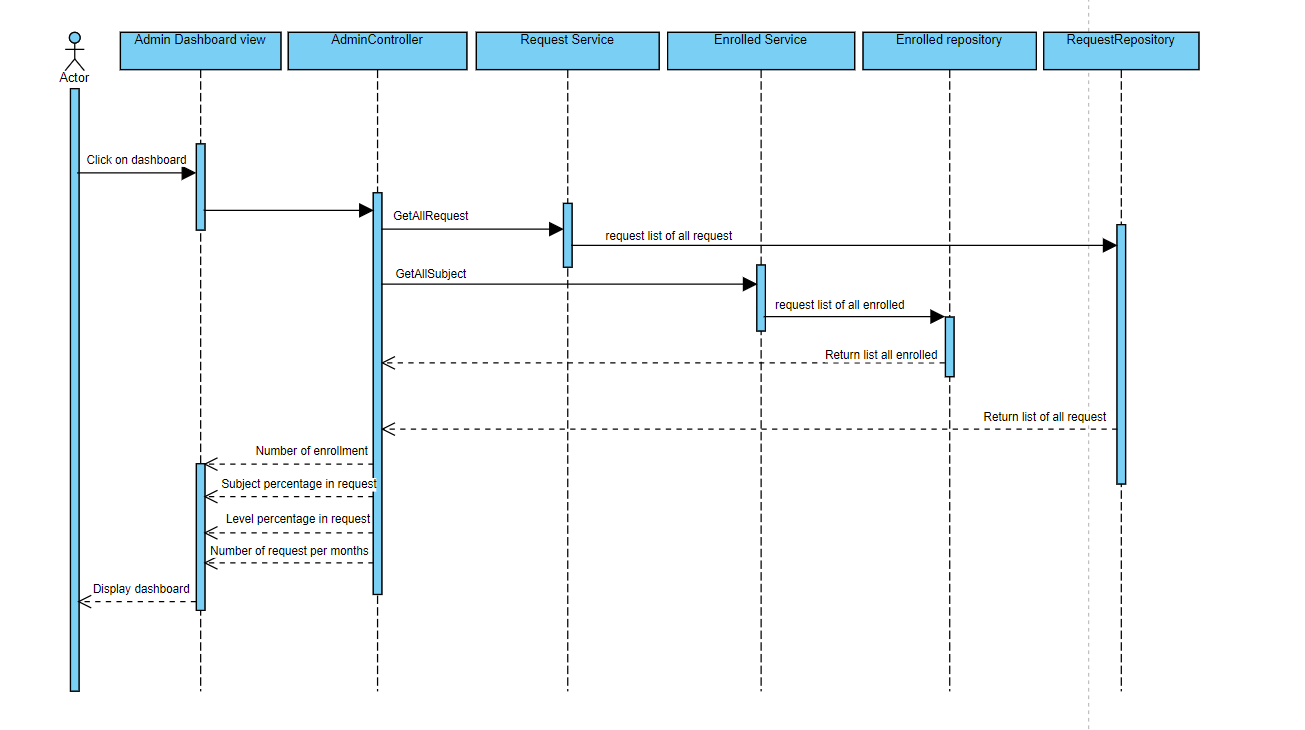
SubjectService: implement the subject repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getAllSubject() | get all the subjects contain in the database |
| 02 | findSubjectbyId(int id) | get the subject by subject id |
| 03 | getAllLevel() | get all the levels contain in the database |
| 04 | findLevelbyId(int id) | get the level by level id |

Enrolled Service:

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | deleteAllEnrolledByRequestID | delete all enrolled object in the database by request id |
| 02 | getAllEnrolledByRequestID() | get all enrolled object in the database by request id |

### Sequence Diagram(s)



### Database queries

select \* from request

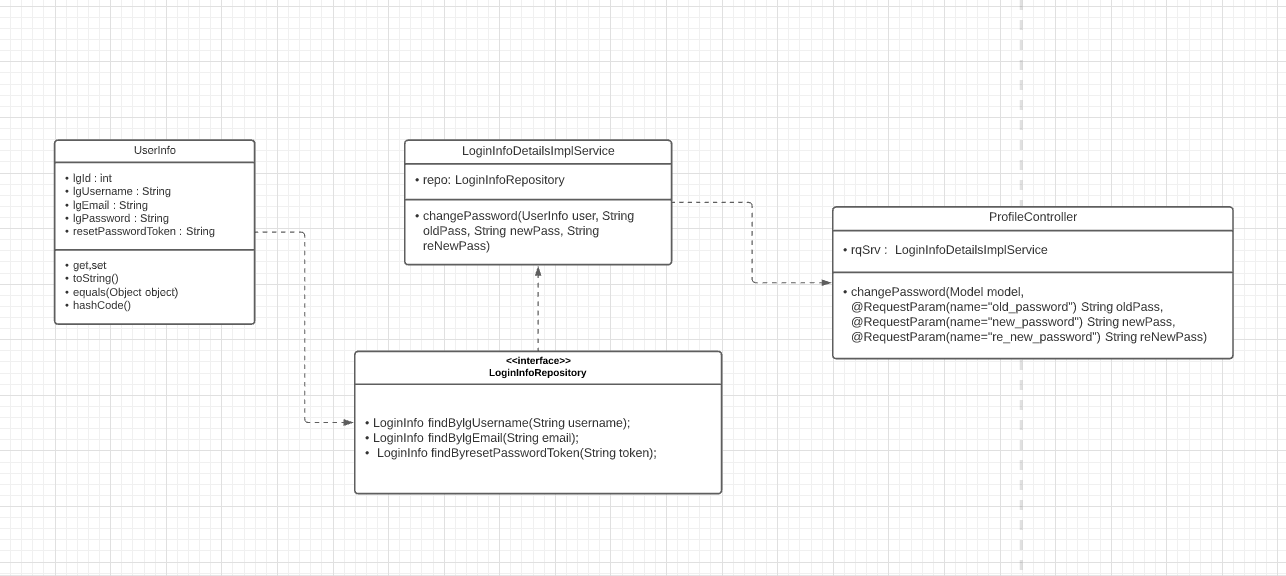
select \* from subject

select \* from level

count \* from request

## 15. Change Password

### Class Diagram



### Class Specification

UserInfo Class: This class store information about user

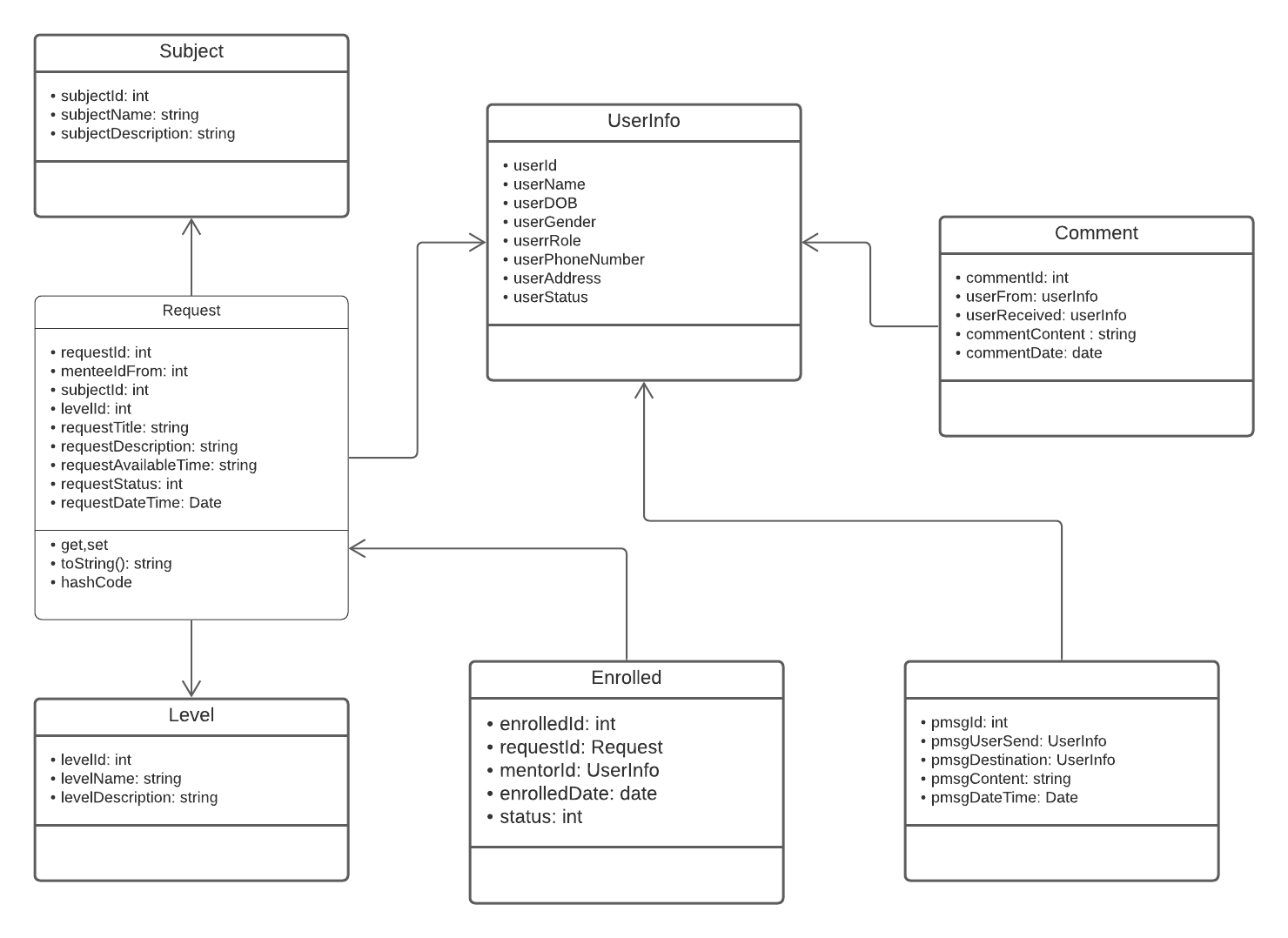
| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getUId() | get user ID from a userInfo object |
| 02 | setUId() | set user ID for a request object |
| 03 | getUName() | get user name from a userInfo object |
| 04 | setUName() | set user name from a userInfo object |
| 05 | getUDOB() | get user date of birth from a userInfo object |
| 06 | setUDOB() | set user date of birth from a userInfo object |
| 07 | getUGender() | get user gender from a userInfo object |
| 08 | setUGender() | set user gender from a userInfo object |
| 09 | getURole() | get user role from a userInfo object |
| 10 | setURole() | set user role from a userInfo object |
| 11 | getUPhoneNumber() | get user phone number from a userInfo object |
| 12 | getUAddress() | get user address from a userInfo object |
| 13 | setUAddress() | set user address from a userInfo object |
| 14 | getUImage() | get user image from a userInfo object |
| 15 | setUImage() | set user image from a userInfo object |
| 16 | getUDescription() | get user description from a userInfo object |
| 17 | setUDescription() | set user description from a userInfo object |
| 18 | getUStatus() | get user status from a userInfo object |
| 19 | setUStatus() | set user status from a userInfo object |
| 20 | getLoginInfo() | get login info from a userInfo object |
| 21 | setLoginInfo() | set login info from a userInfo object |
| 22 | toString() | to string method |

### Sequence Diagrams

### Database Queries

## 16. Admin daily report

### Class Diagram



### Class Specification

Request Class: This class store information about request

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getReqId() | get request ID from a request object |
| 02 | setReqId | set request ID for a request object |
| 03 | getReqStatus() | get request status from a request object |
| 04 | setReqStatus | set request status for a request object |
| 05 | getReqDesc() | get request description from a request object |
| 06 | setReqDesc() | set request description for a request object |
| 07 | getLevId() | get level from a request object |
| 08 | setLevId() | set level for a request object |
| 09 | getSubId() | get subject from a request object |
| 10 | setSubId() | set subject for a request object |
| 11 | getReqTitle() | get request title from a request object |
| 12 | setReqTitle() | set request title for a request object |
| 13 | getReqAvaiTime() | get request available time from a request object |
| 14 | setReqAvaiTime() | set request available time for a request object |
| 15 | getReqDateTime() | get request date time |
| 16 | setReqDateTime() | set request date time for a request object |
| 17 | getMenteeIdFrom() | get userinfo from a request object |
| 18 | setMenteeIdFrom() | set userinfo for a request object |
| 19 | toString() | get to string from a request object |

Level Class: This class store the information about object level

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getLevId() | get level ID from a level object |
| 02 | setLevId() | set level ID for a level object |
| 03 | getLevName() | get level name from a level object |
| 04 | setLevName() | set level name for a level object |
| 05 | getLevDesc() | get level description from a level object |
| 06 | setLevDesc() | set level description for a level object |
| 07 | toString() | get to string |

Subject Class: This class store the information about object subject

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getSubId() | get level ID |
| 02 | setSubId() | set level ID |
| 03 | getSubName()() | get level name |
| 04 | setSubName() | set level name |
| 05 | getSubDesc() | get level description |
| 06 | setSubDesc() | set level description |
| 07 | toString() | get to string |

Enrolled Class: This class store the information about object enrolled

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getEnrId() | get request ID |
| 02 | setEnrId() | set request ID |
| 03 | getEnrDate() | get request status |
| 04 | setEnrDate() | set request status |
| 05 | getReqId() | get request description |
| 06 | setReqId() | set request description |
| 07 | getMentorId() | get level |
| 08 | setMentorId() | set level |
| 09 | getStatus() | get subject |
| 10 | setStatus() | set subject |
| 11 | toString() | get request title |

RequestService: implement the request repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | listAllMyByPage() | get the paginated list of request |
| 02 | listAllByPage() | get all the request list after paging |
| 03 | saveNewRequest() | save a new created request |
| 04 | deleteRequest() | delete a selected request |
| 05 | updateRequest() | update information in the selected request |

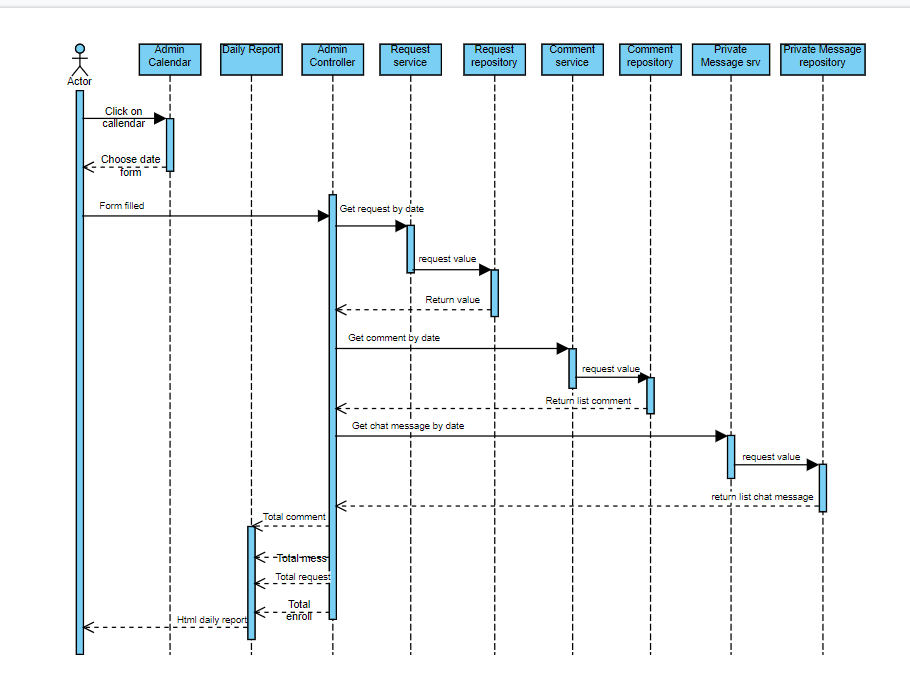
SubjectService: implement the subject repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getAllSubject() | get all the subjects contain in the database |
| 02 | findSubjectbyId(int id) | get the subject by subject id |
| 03 | getAllLevel() | get all the levels contain in the database |
| 04 | findLevelbyId(int id) | get the level by level id |

Enrolled Service:

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | deleteAllEnrolledByRequestID | delete all enrolled object in the database by request id |
| 02 | getAllEnrolledByRequestID() | get all enrolled object in the database by request id |

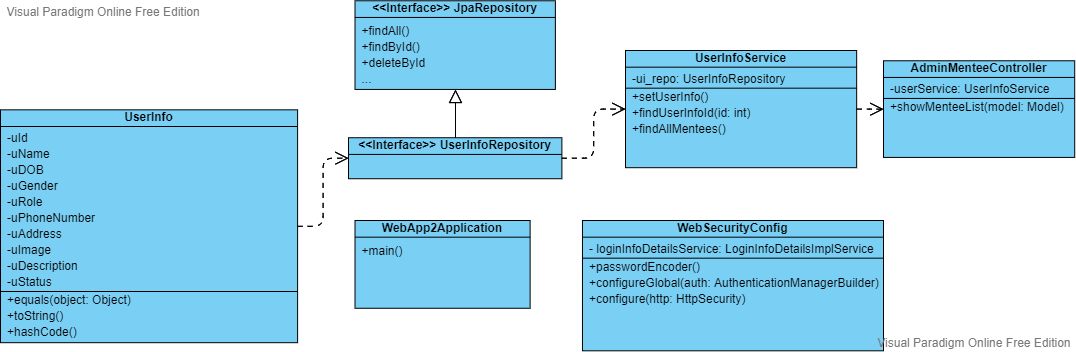
### Sequence Diagrams



### Database Queries

## 17. Admin view mentee list

### Class Diagram



### Class Specification

UserInfo Class: This class store the information about object UserInfo

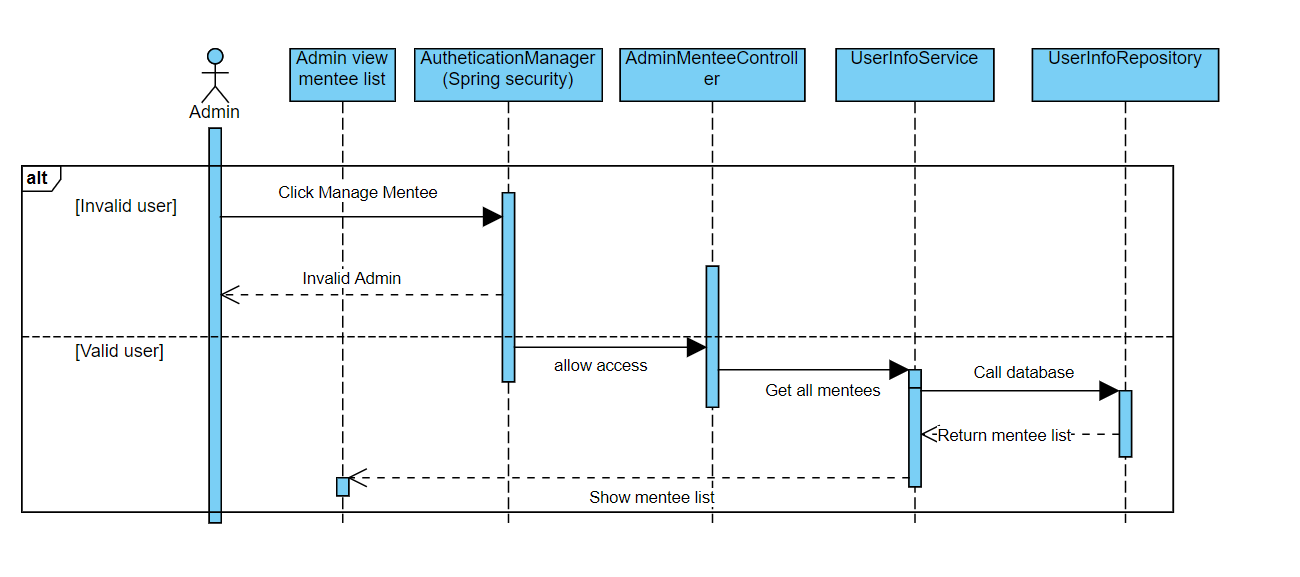
| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getUId() | get UserInfo ID |
| 02 | setUId() | set UserInfo ID |
| 03 | getUName() | get UserInfo Name |
| 04 | setUName() | set UserInfo Name |
| 05 | getUDOB() | get UserInfo DOB |
| 06 | setUDOB() | set UserInfo DOB |
| 07 | getUGender() | get UserInfo Gender |
| 08 | setUGender() | set UserInfo Gender |
| 09 | getURole() | get UserInfo Role |
| 10 | setURole() | set UserInfo Role |
| 11 | getUPhoneNumber | get UserInfo Phone Number |
| 12 | setUPhoneNumber | set UserInfo Phone Number |
| 13 | toString() |  |

UserInfoService: implement the User Info repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | findAllMentees | get the paginated list of request |
| 02 | saveMentee | get all the request list after paging |

AdminMenteeController:

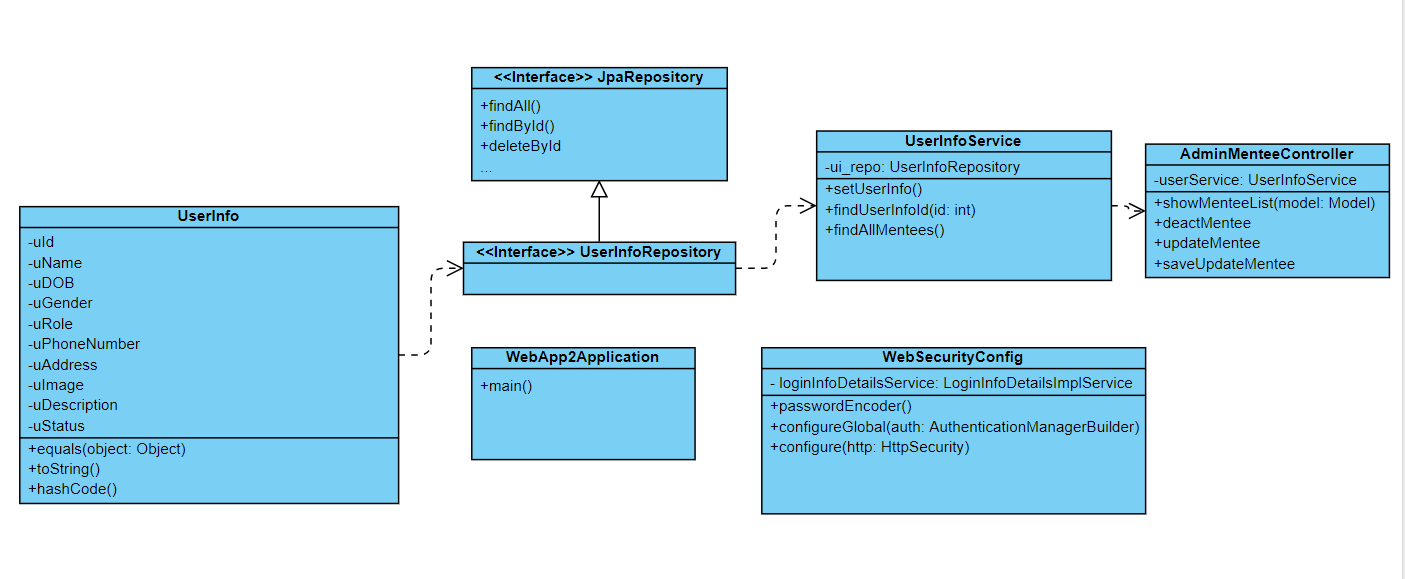
### Sequence Diagrams



### Database Queries

## 18. Admin Manage mentee

### Class Diagram



### Class Specification

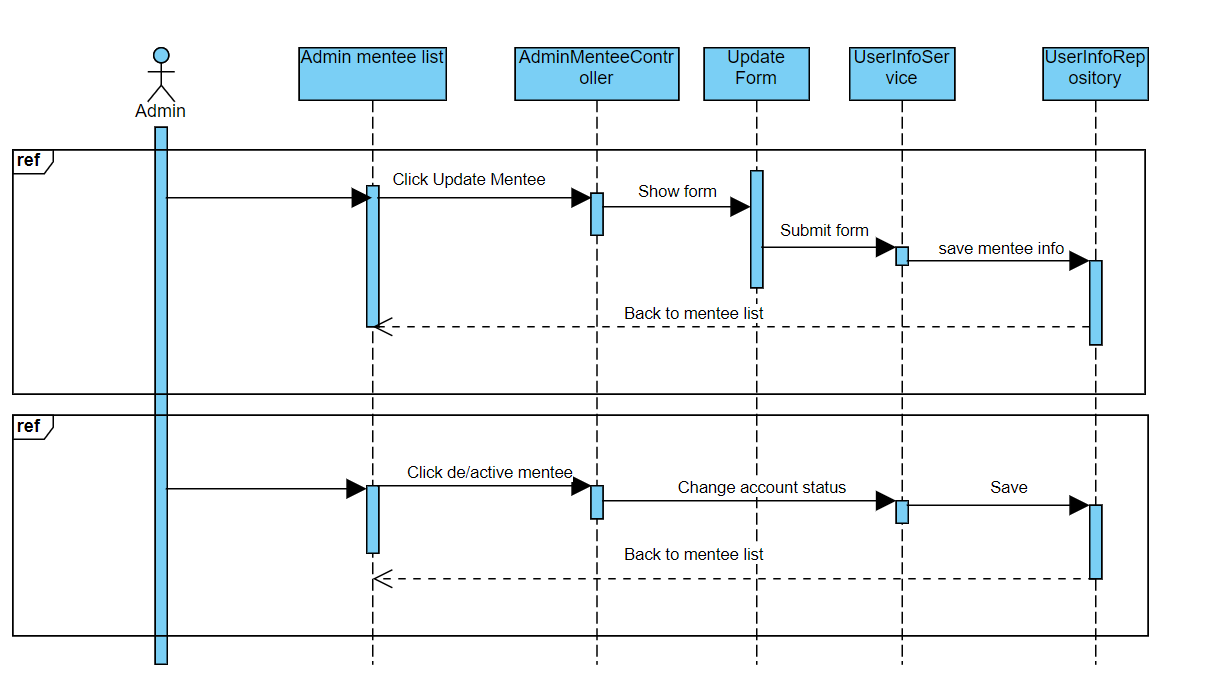
UserInfo Class: This class store the information about object UserInfo

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getUId() | get UserInfo ID |
| 02 | setUId() | set UserInfo ID |
| 03 | getUName() | get UserInfo Name |
| 04 | setUName() | set UserInfo Name |
| 05 | getUDOB() | get UserInfo DOB |
| 06 | setUDOB() | set UserInfo DOB |
| 07 | getUGender() | get UserInfo Gender |
| 08 | setUGender() | set UserInfo Gender |
| 09 | getURole() | get UserInfo Role |
| 10 | setURole() | set UserInfo Role |
| 11 | getUPhoneNumber | get UserInfo Phone Number |
| 12 | setUPhoneNumber | set UserInfo Phone Number |
| 13 | toString() |  |

UserInfoService: implement the User Info repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | findAllMentees | get the paginated list of request |
| 02 | saveMentee | get all the request list after paging |

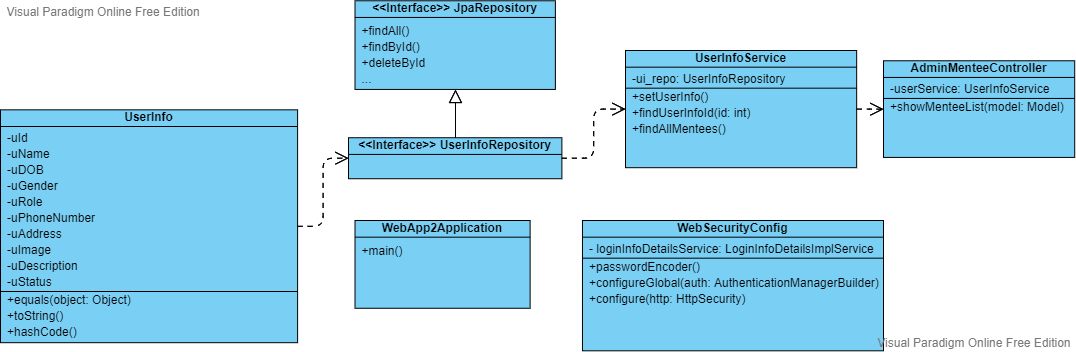
### Sequence Diagrams



### Database Queries

## 19. Admin View mentor list

### Class Diagram



### Class Specification

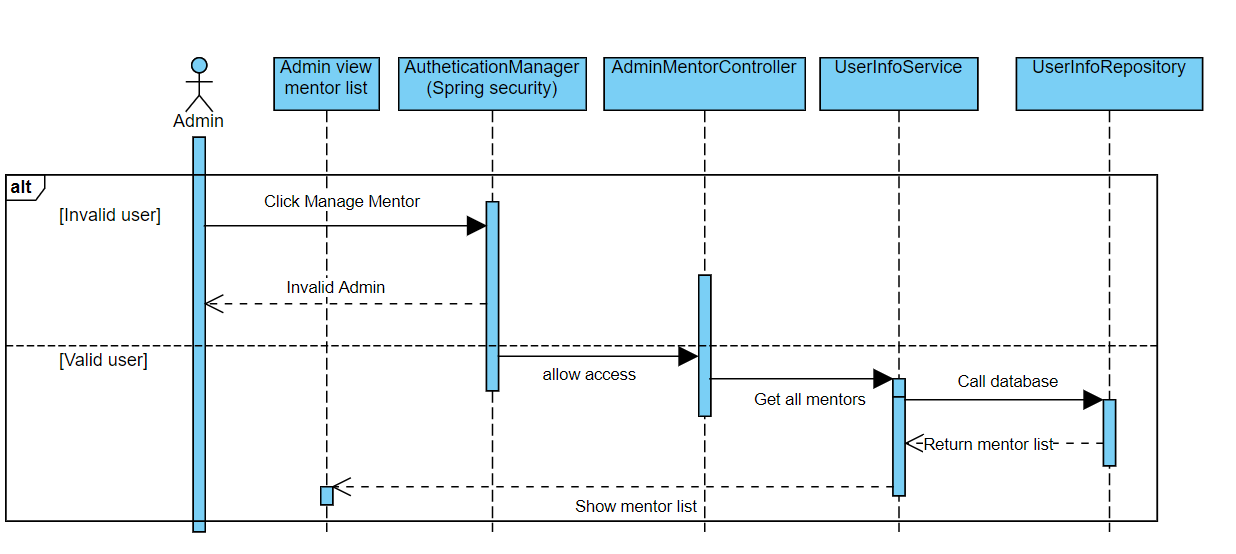
UserInfo Class: This class store the information about object UserInfo

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getUId() | get UserInfo ID |
| 02 | setUId() | set UserInfo ID |
| 03 | getUName() | get UserInfo Name |
| 04 | setUName() | set UserInfo Name |
| 05 | getUDOB() | get UserInfo DOB |
| 06 | setUDOB() | set UserInfo DOB |
| 07 | getUGender() | get UserInfo Gender |
| 08 | setUGender() | set UserInfo Gender |
| 09 | getURole() | get UserInfo Role |
| 10 | setURole() | set UserInfo Role |
| 11 | getUPhoneNumber | get UserInfo Phone Number |
| 12 | setUPhoneNumber | set UserInfo Phone Number |
| 13 | toString() |  |

UserInfoService: implement the User Info repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | findAllMentees | get list of all mentees |
| 02 | saveMentee | save mentee with new info |

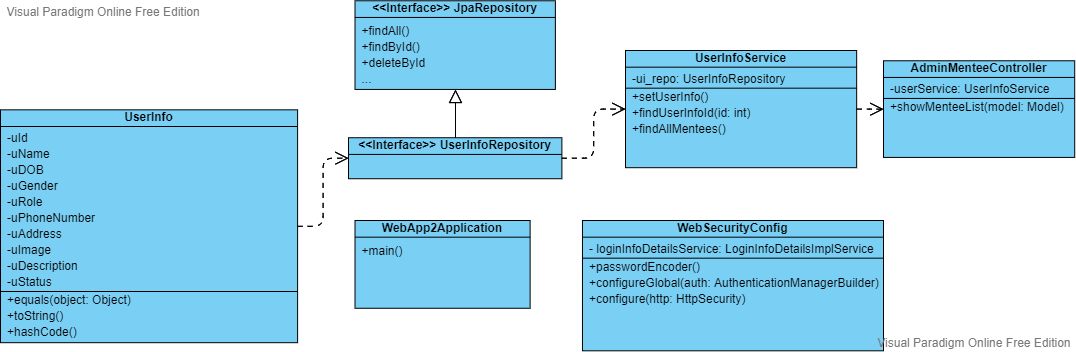
### Sequence Diagrams



### Database Queries

## 20. Admin Manage mentor

### Class Diagram

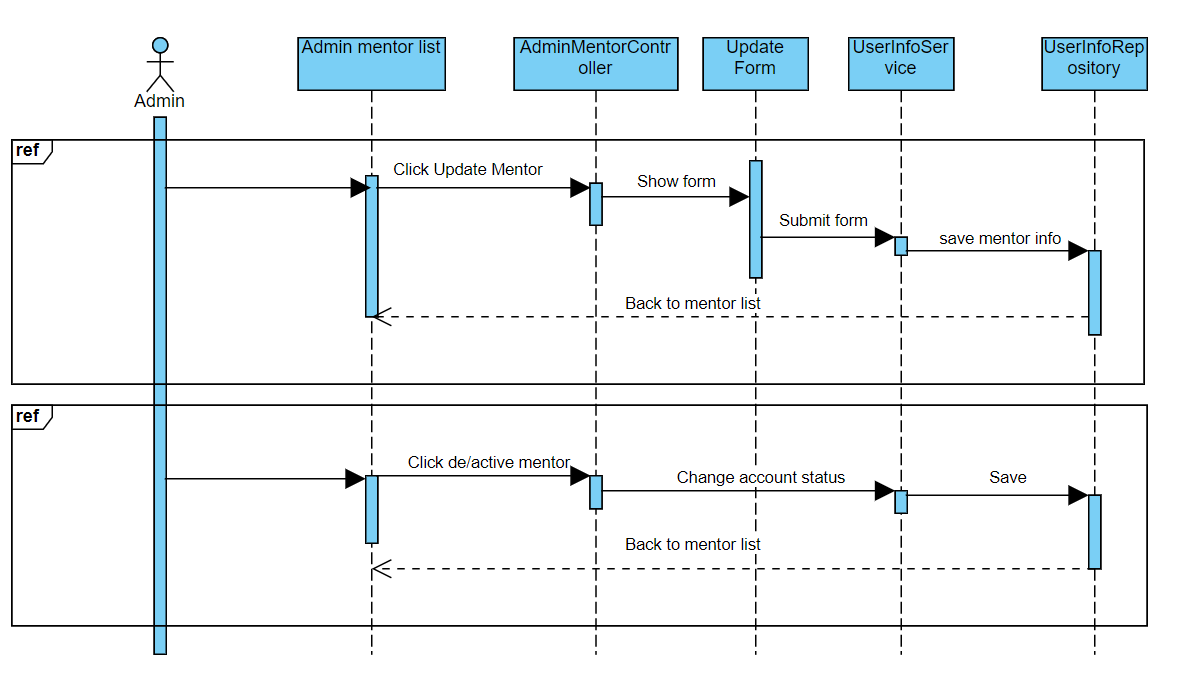


### Class Specification

UserInfo Class: This class store the information about object UserInfo

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getUId() | get UserInfo ID |
| 02 | setUId() | set UserInfo ID |
| 03 | getUName() | get UserInfo Name |
| 04 | setUName() | set UserInfo Name |
| 05 | getUDOB() | get UserInfo DOB |
| 06 | setUDOB() | set UserInfo DOB |
| 07 | getUGender() | get UserInfo Gender |
| 08 | setUGender() | set UserInfo Gender |
| 09 | getURole() | get UserInfo Role |
| 10 | setURole() | set UserInfo Role |
| 11 | getUPhoneNumber | get UserInfo Phone Number |
| 12 | setUPhoneNumber | set UserInfo Phone Number |
| 13 | toString() |  |

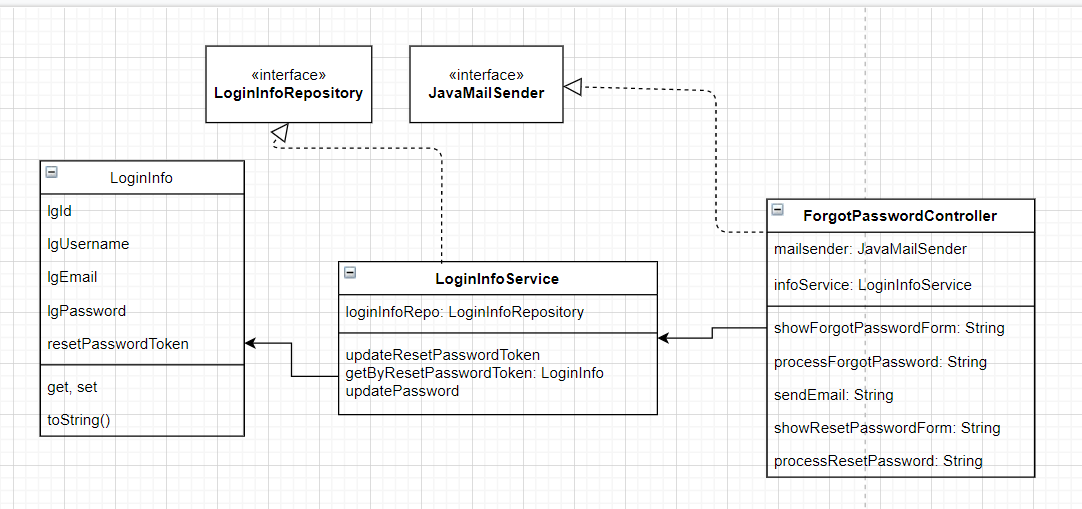
### Sequence Diagrams



### Database Queries

## 21. Forgot Password

### Class Diagram



### Class Specification

LoginInfo Class: This class store the information about object LoginInfo

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | getUId() | get UserInfo ID |
| 02 | setUId() | set UserInfo ID |
| 03 | getUName() | get UserInfo Name |
| 04 | setUName() | set UserInfo Name |
| 05 | getUDOB() | get UserInfo DOB |
| 06 | setUDOB() | set UserInfo DOB |
| 07 | getUGender() | get UserInfo Gender |
| 08 | setUGender() | set UserInfo Gender |
| 09 | getURole() | get UserInfo Role |
| 10 | setURole() | set UserInfo Role |
| 11 | getUPhoneNumber | get UserInfo Phone Number |
| 12 | setUPhoneNumber | set UserInfo Phone Number |
| 13 | toString() |  |

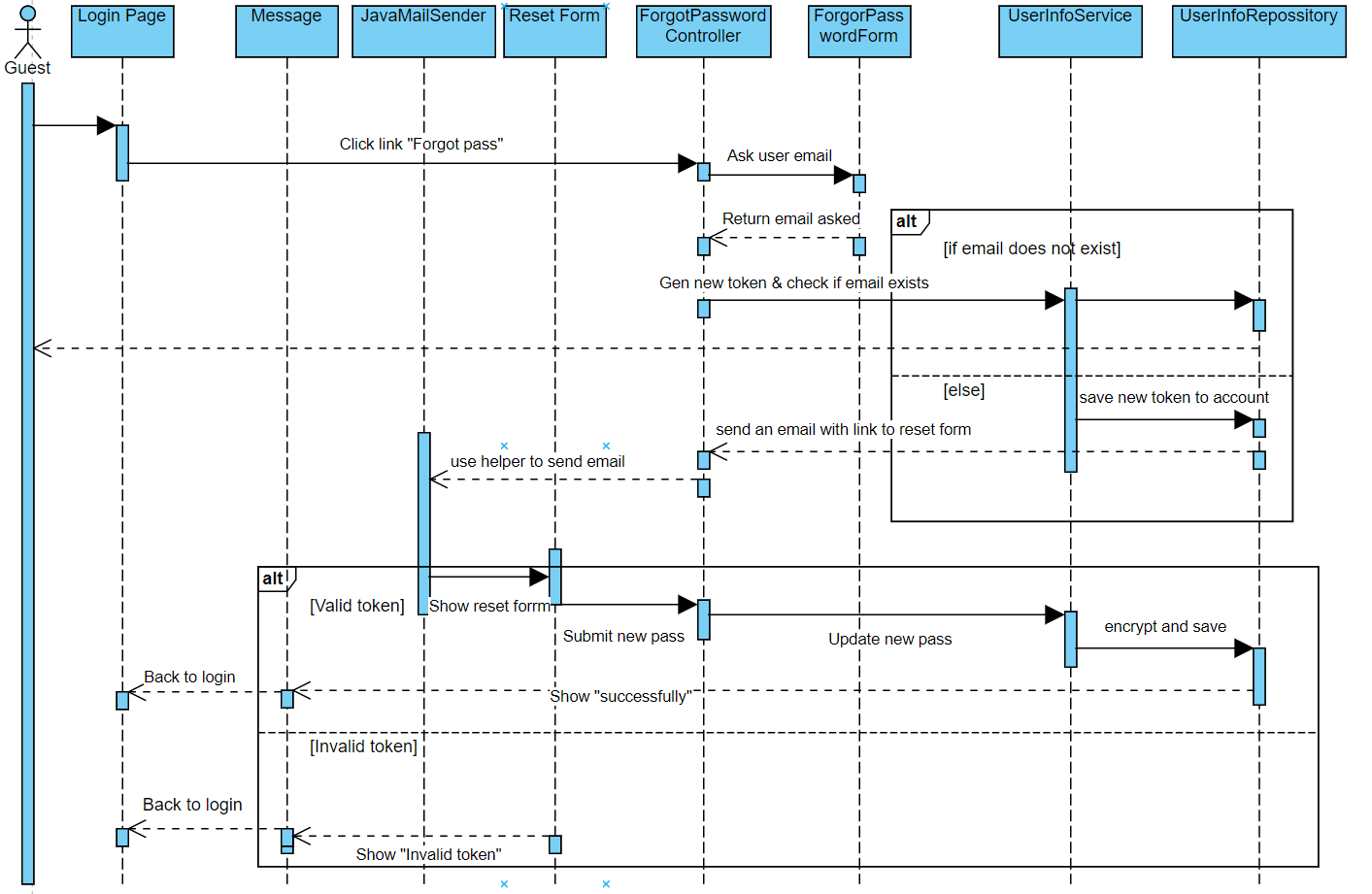
LoginInfo Service: implement the LoginInfo repository

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | updateResetPasswordToken | set reset password token to LoginInfo |
| 02 | getByResetPasswordToken | Find loginInfo by resetPasswordToken |
| 03 | updatePassword | Update new encrypted password to LoginInfo |

ForgotPassword Controller

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | showForgotPasswordForm() | set reset password token to LoginInfo |
| 02 | processForgotPassword | Find loginInfo by resetPasswordToken |
| 03 | sendEmail | Update new encrypted password to LoginInfo |
| 04 | showResetPasswordForm | Show reset password form |
| 05 | processResetPassword | Encrypted & save new password |

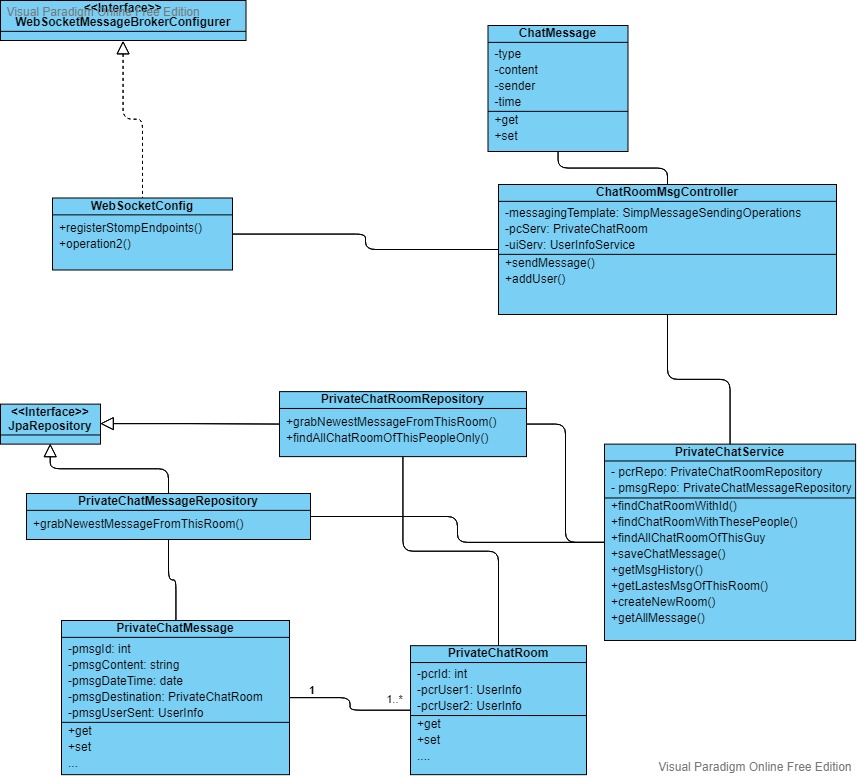
### Sequence Diagrams



### Database Queries

## 22. Chat

### Class Diagram



### Class Specification

***WebSocketConfig***

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | registerStompEndpoints | configure Endpoint to handle WebSocket |

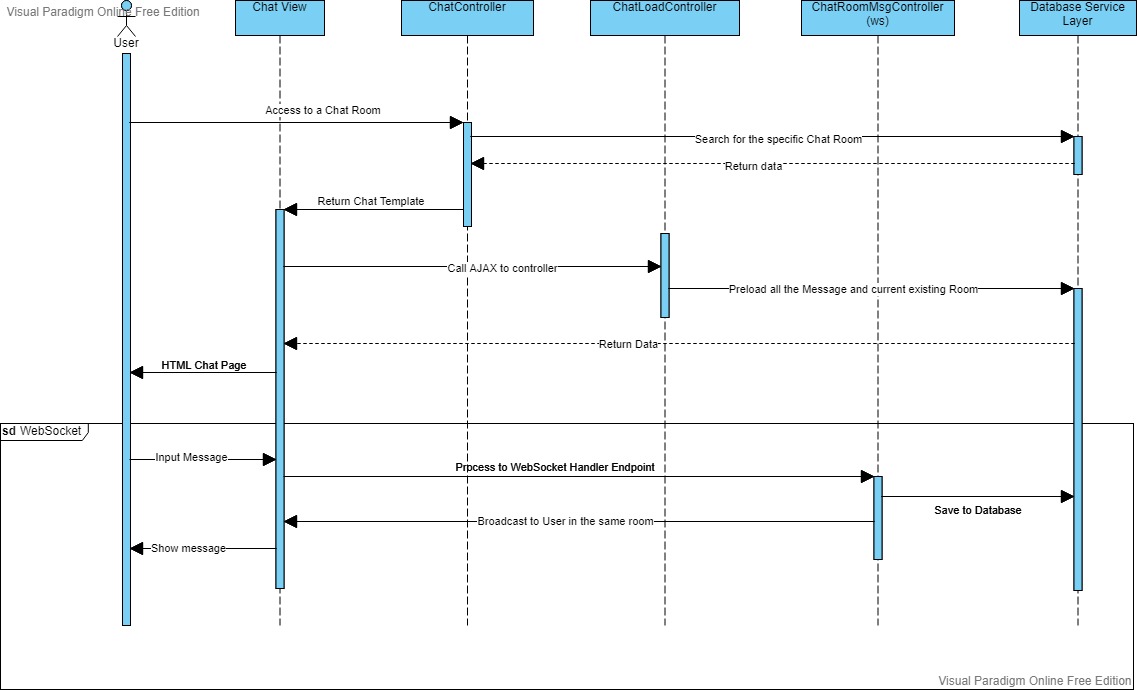
***ChatRoomMsgController***

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | sendMessage | A Endpoint Handler for message sent to this path |
| 02 | addUser | A Endpoint Handler for user added to this path |

***PrivateChatRoomService***

| **No** | **Method** | **Description** |
| --- | --- | --- |
| 01 | findChatRoomWithId | Get Chatroom of Id |
| 02 | findChatRoomWithThesePeople | Get ChatRoom of Specific People in the Room |
| 03 | findAllChatRoomOfThisGuy | Find all Room have this people |
| 04 | getMsgHistory | List all Message of a Room |
| 05 | getLatestMsgOfThisRoom() | List Latest Message of this Room |
| 06 | createNewRoom | Create new Room |
| 07 | getAllMessage | Find all Message |

### Sequence Diagram



1. **Database Queries**

* findChatRoomWithThesePeople()

SELECT c FROM PrivateChatRoom c WHERE (c.pcrUser1.uId = ?1 AND c.pcrUser2.uId = ?2) OR (c.pcrUser2.uId= ?1 AND c.pcrUser1.uId = ?2)

* findAllChatRoomOfThisPeopleOnly()

SELECT c FROM PrivateChatRoom c WHERE (c.pcrUser1.uId= ?1) OR (c.pcrUser2.uId= ?1)

* grabNewestMessageFromThisRoom()

SELECT c FROM PrivateChatMessage c WHERE c.pmsgDateTime IN (SELECT MAX(p.pmsgDateTime) FROM PrivateChatMessage p WHERE p.pmsgDestination.pcrId = ?1) AND c.pmsgDestination = ?1

# III. Database Tables

## 1. LoginInfo

*[Give some lines about the table here>>*

*[Table fields, in the form of table format as below]*

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | lgId | int |  |  |  | PK | auto increment |
| 2 | lgUsername | nvarchar | 15 | true | true |  |  |
| 3 | lgEmail | nvarchar | 255 | true | true |  |  |
| 4 | lgPassword | varchar | 60 |  | true |  |  |

## 2. Authorization

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | aId | int |  |  |  | PK |  |
| 2 | aName | nvarchar | 60 | true | true |  |  |

## 3. LoginInfo\_Authorization

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | lgId | int |  |  | true | FK |  |
| 2 | aId | int |  |  | true | FK |  |

## 4. User

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | uId | int |  |  |  | PK | REFERENCES LoginInfo(lgId) |
| 2 | uName | nvarchar | 256 |  | true |  |  |
| 3 | uDOB | date |  |  | true |  |  |
| 4 | uGender | bit |  |  | true |  |  |
| 5 | uRole | nvarchar | 32 |  | true |  |  |
| 6 | uPhoneNumber | nvarchar | 32 |  |  |  |  |
| 7 | uAddress | nvarchar | 256 |  |  |  |  |
| 8 | uImage | nvarchar | 256 |  |  |  |  |
| 9 | uDescription | nvarchar | 256 |  |  |  |  |
| 10 | uStatus | bit |  |  | true |  | default 1 |

## 5. SkillnExperience

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | sneId | int |  |  |  | PK | auto increment |
| 2 | userId | int |  |  |  | FK |  |
| 3 | sneType | int |  |  |  |  |  |
| 4 | sneDescription | nvarchar | 256 |  |  |  |  |

## 6. Subject

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | subId | int |  |  |  | PK | auto increment |
| 2 | subName | nvarchar | 32 |  | true |  |  |
| 3 | subDesc | nvarchar | 256 |  |  |  |  |

## 

## 7. Level

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | levId | int |  |  |  | PK | auto increment |
| 2 | levName | nvarchar | 32 |  | true | FK |  |
| 3 | levDesc | nvarchar | 256 |  |  |  |  |

## 8. Mentor \_Subject

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | mnId | int |  |  |  | FK |  |
| 2 | subjId | int |  |  |  | FK |  |

## 9. Request

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | reqId | int |  |  |  | PK | auto increment |
| 2 | menteeId | int |  |  |  | FK |  |
| 3 | subId | int |  |  |  | FK |  |
| 4 | levId | int |  |  |  |  |  |
| 5 | reqStatus | bit |  |  | true |  | default 0 |
| 6 | reqDesc | nvarchar | 256 |  |  |  |  |
| 7 | reqDate | date |  |  |  |  |  |
| 8 | reqAvailableDate | date |  |  |  |  |  |

## 10. Enrolled

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | enrId | int |  |  |  | PK | auto increment |
| 2 | reqId | int |  |  |  | FK |  |
| 3 | mentorId | int |  |  |  | FK |  |
| 4 | enrDate | date |  |  |  |  |  |

## 

## 11. ChatEndponts

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | cEdId | int |  |  |  | PK | auto increment |
| 2 | userFrom | int |  |  |  | FK |  |
| 3 | userTo | int |  |  |  | FK |  |

## 12. Chat

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | cId | int |  |  |  | PK | auto increment |
| 2 | cUserFrom | bit |  |  |  |  |  |
| 3 | cEndpTo | int |  |  |  | FK |  |
| 4 | cContent | nvarchar | 256 |  |  |  |  |
| 5 | cData | Date |  |  |  |  |  |

## 13. Notify

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | notId | int |  |  |  | PK | auto increment |
| 2 | notType | int |  |  |  |  |  |
| 3 | userReceived | int |  |  |  | FK |  |
| 4 | userFrom | int |  |  |  | FK |  |
| 5 | notStatus | bit |  |  | true |  | default 0 |
| 6 | notDate | Date |  |  |  |  |  |
| 7 | notDesc | nvarchar | 256 |  |  |  |  |

## 14. Comment

| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | comId | int |  |  |  | PK | auto increment |
| 2 | userFrom | int |  |  |  | FK |  |
| 3 | userReceived | int |  |  |  | FK |  |
| 4 | comContent | nvarchar | 256 |  |  |  |  |
| 5 | comDate | Date |  |  |  |  |  |