

Capstone Project Final Report

**CAPSTONE PROJECT MANAGEMENT**

|  |  |
| --- | --- |
|  | **FUCPM Team** |
|  |  |
|  | Do Dinh Duc – SE04936 |
|  | Bui Trung Kien – SE05182 |
| **Group Members** | Nguyen Ngoc Khanh - SE05045 |
|  |
|  | Nguyen Duc Long – SE05906 |
|  | Nguyen Minh Khang – SE05904 |
| **Supervisor** | Lecturer: Nghiem Thi Lan Phuong |
|  |  |
| **Capstone Project code** | CPM |
|  |  |

**Hanoi, Sep 1st, 2020**

**Table of contents**

[**ACKNOWLEDGEMENTS**](#page4)[**4**](#page4)

[**DEFINITIONS AND ACRONYMS**](#page5)[**5**](#page5)

[**CHAPTER 1 : INTRODUCTION**](#page7)[**7**](#page7)

[1.1](#page7) [PURPOSE](#page7) [7](#page7)

[1.2](#page7) [PROJECT INFORMATION](#page7) [7](#page7)

[1.3](#page7) [THE PEOPLE](#page7) [7](#page7)



[1.4](#page7) [BACKGROUND](#page7) [7](#page7)

[1.5](#page8) [THE EXISTING SYSTEM](#page8) [8](#page8)

[1.6](#page9) [THE PROPOSAL OF SYSTEM](#page9) [9](#page9)



[**CHAPTER 2 : PROJECT PLAN**](#page12)[**12**](#page12)

[2.1](#page12) [PURPOSE](#page12) [12](#page12)



[2.2](#page14) [PROJECT MANAGEMENT PLAN](#page14) [14](#page14)



[**CHAPTER 3 : SOFTWARE REQUIREMENT SPECIFICATION**](#page18)[**18**](#page18)

[3.1](#page18) [PURPOSE](#page18) [18](#page18)

[3.2](#page18) [FUNCTIONAL REQUIREMENTS](#page18) [18](#page18)



[3.3](#page65) [NON-FUNCTIONAL REQUIREMENT](#page65) [65](#page65)



[3.4](#page67) [DATABASE DOCUMENT DIAGRAM](#page67) [67](#page67)

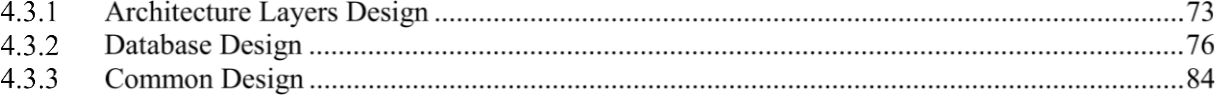
[**CHAPTER 4 : SOFTWARE DESIGN**](#page70)[**70**](#page70)

[4.1](#page70) [PURPOSE](#page70) [70](#page70)

[4.2](#page70) [ARCHITECTURE OVERVIEW](#page70) [70](#page70)



[4.3](#page73) [DESIGN OF ASC SYSTEM](#page73) [73](#page73)





[**CHAPTER 5 : SOFTWARE TESTING DOCUMENTATION**](#page192)[**192**](#page192)

[5.1](#page192) [PURPOSE](#page192) [192](#page192)



[5.2](#page193) [TEST PLAN](#page193) [193](#page193)



[5.3](#page200) [TEST CASE](#page200) [200](#page200)



[5.4](#page206) [TEST REPORT](#page206) [206](#page206)



[**CHAPTER 6 : USER MANUAL**](#page220)[**220**](#page220)

[6.1](#page220) [DEPLOYMENT GUIDELINES](#page220) [220](#page220)



[6.2](#page223) [USER GUIDELINES](#page223) [223](#page223)



**Acknowledgements**

*We wish to express our deepest gratitude to our supervisor Mrs. Nghiem Thi Lan Phuong for his continuously sharing and motivating throughout the project. Under the instructions of Mrs. Phuong, the project team has always felt comfortable and confident. Apart from all the technologies and methodologies, Mrs. Phuong shared with us the philosophy of keeping the right attitude towards problems. We believe that was the most important factor that led to the success of this project.*

*We would also like to thank the instructors at FPT University for all the classes. We hope you will find this project as a reflection of the knowledge and experiences you have given us during this period of four years.*

*Finally, we truly appreciate the random fellows on the internet who has discussed with us.*

*Without them, it would have likely been a real struggle to solve every tiny problem.*

|  |  |
| --- | --- |
|  | **Definitions and Acronyms** |
| **Terminologies** |  |
|  |  |
| **Terminology** | **Definition** |
| **Chat bot (also Bots)** | Software that is used to interact between a computer and a human in |
|  | natural language. |
| **Broadcast** | A message proactively to users. |
|  |  |
| **Nodes** | A node is where different points of a dialog or workflow intersect |
|  |  |
| **Utterance** | Anything someone using chat bot says to it. |
|  |  |
| **Intent** | The few essential words that describe what the user wants the chat bot to |
|  | do. |
| **Entity** | Entities are the fields, data, or words the developer designates are |
|  | necessary for a chat bot to complete the user’s request. An entity could be |
|  | a date, a time, a location, a description or any number of things. |
| **Chat logs** | Histories of all recorded human-to-bot interactions. |
|  |  |
| **Acronyms** |  |
|  |  |
| **Acronym** | **Definition** |
| **API** | Application Programming Interface |
|  |  |
| **BDD** | Driven Development |
|  |  |
| **CPU** | Central Processing Unit |
|  |  |
| **NLP** | Natural Language Processing |
|  |  |
| **NLU** | Natural Language Understanding |
|  |  |
| **NER** | Named Entity Recognition |
|  |  |
| **BOW** | Bag-of-Words |
|  |  |
| **SVM** | Support Vector Machine |
|  |  |
| **TF-IDF** | Term Frequency- Inverse Document Frequency |
|  |  |
| **DBMS** | Database Management System |
|  |  |
| **DOM** | Document Object Model |
|  |  |
| **GUI** | Graphical User Interface |
|  |  |
| **HDD** | Hard Disk Drive |
|  |  |
| **HTTP** | Hypertext Transfer Protocol |
|  |  |
| **HTTPS** | Hypertext Transfer Protocol Secure |
|  |  |
| **IDE** | Integrated Development Environment |
|  |  |
| **UML** | Unified Modelling Language |
|  |  |
| **N/A** | Not Available |
|  |  |
| **OS** | Operating System |
|  |  |
| **PM** | Project manager |
|  |  |
| **REST** | Representational State Transfer |
|  |  |
| **SRS** | Software Requirement Specification |
|  |  |
| **TDD** | Test-Driven Development |
|  |  |

|  |  |
| --- | --- |
| **UC** | Use Case |
|  |  |
| **UI** | User Interface |
|  |  |
| **UX** | User Experience |
|  |  |
| **URL** | Uniform Resource Locator |
|  |  |
| **FAQ** | Frequently Asked Questions |
|  |  |
| **PSID** | Page-Scoped ID |
|  |  |
| **Fb** | Facebook |
|  |  |
| **FU** | FPT University |
|  |  |
| **JWT** | JSON Web Token |
|  |  |
|  |  |

**Chapter 1: INTRODUCTION**

**1.1 Purpose**

This chapter provides an overview of the project including background information, a literature review of the existing system and raising a proposal for ideas of improvement.

**1.2 Project Information**

* Project name: Capstone Project Management
* Project code: **CPM**
* Project group name: **FUCPM**
* Product type: **Web Application**
* Timeline: **Sep 1st** **2020 -**

**1.3** **The people**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Roles** | **E-mail** | **Phone** |
| 1 | Nghiem Thi Lan Phuong | Supervisor | [phuongntl@fpt.edu.vn](mailto:phuongntl@fpt.edu.vn) | 0989376681 |
| 2 | Do Dinh Duc | Leader | [ducdse04936@fpt.edu.vn](mailto:ducdse04936@fpt.edu.vn) | 0968480997 |
| 3 | Bui Trung Kien | Member | [kienbtse05182@fpt.edu.vn](mailto:kienbtse05182@fpt.edu.vn) | 0983268048 |
| 4 | Nguyen Ngoc Khanh | Member | khanhnnse05045@fpt.edu | 0914092466 |
| 5 | Nguyen Duc Long | Member | [longndse05906@fpt.edu.vn](mailto:longndse05906@fpt.edu.vn) | 0978299648 |
| 6 | Nguyen Minh Khang | Member | khangnmse05904@fpt.edu | 0971091547 |

Table1.1 – Team member’s information



**1.4** **Background**

Each student studying at FPT University has a 4-year path to complete subjects according to the school's schedule. Each student must complete a graduation thesis to complete the final test of student time. This makes us think of a project to create a software that performs functions for students to register, discuss, complete deadlines, exchange with teachers ... It would be useful to reduce costs and time.

Ảnh có chứa ký hiệu, ngoài trời, cỏ, tòa nhà

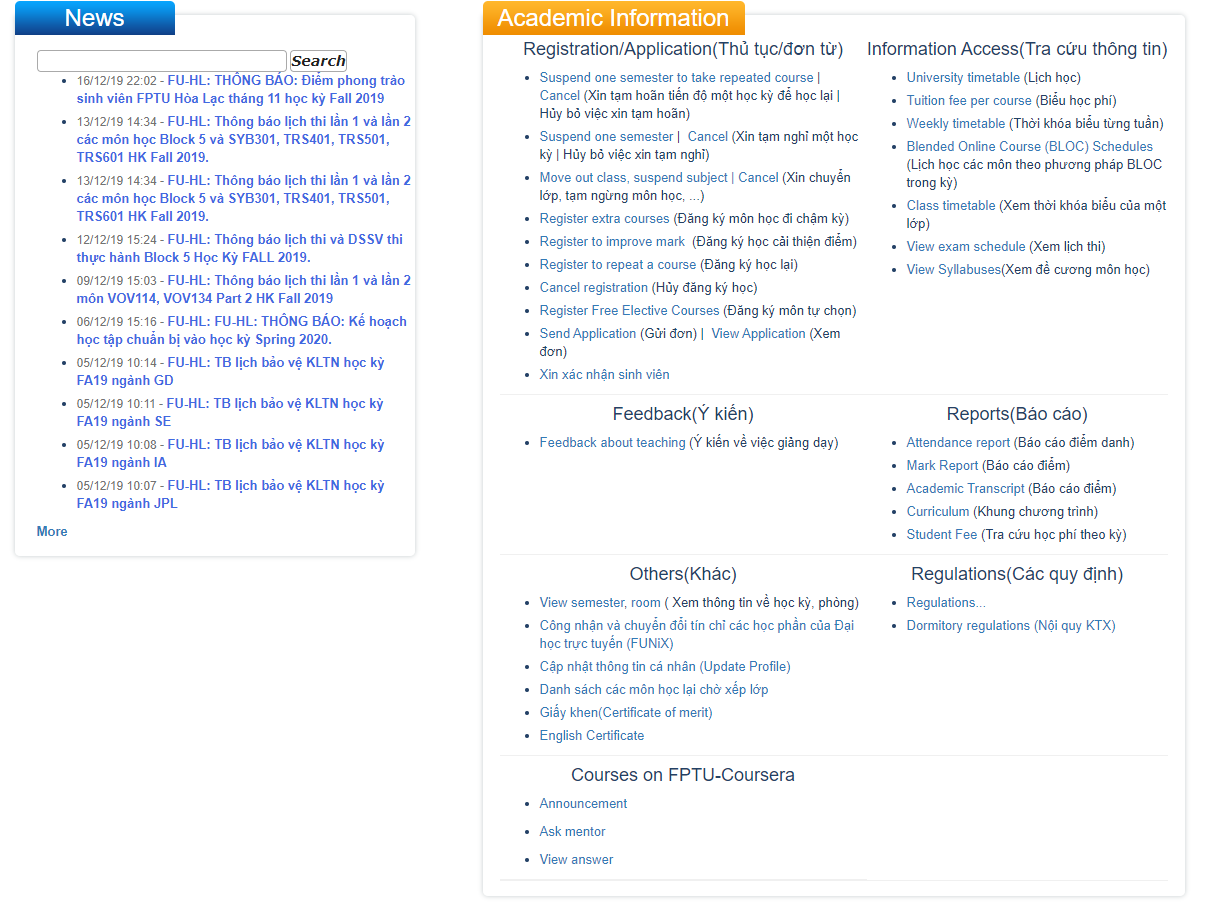
Mô tả được tạo tự động

*Figure 1-1: FPT University*

**1.5** **The Existing System**

As far as we know, major universities in Vietnam still manage their internships manually.

Currently, FPT University has a number of websites for students to conveniently track their progress, such as http://fap.fpt.edu.vn/ helps students of the school to track all subject information, exam schedule, score reports, ...



|  |  |  |
| --- | --- | --- |
| **Features** | **Ana1** | **Harafunel2** |
| Support Vietnamese chatting. |  |  |
|  |  |  |
| Serve in specific admission domain. |  |  |
|  |  |  |
| Throw random greeting responses. |  |  |
|  |  |  |
| The accuracy of a domain specific bot can further improve. |  |  |
|  |  |  |
| Having the flow of conversations and context. |  |  |
|  |  |  |
| Allow the end user to leave personal information. |  |  |
|  |  |  |
| Allow the end user to rate and review. |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| Having dashboard administration. |  |  |
|  |  |  |
| Allow the admin to view statistics. |  |  |
|  |  |  |
| Allow the admin to manage chat logs. |  |  |
|  |  |  |
| Allow the admin to send messages to all end users at time. |  |  |
|  |  |  |
| Allow admin to categorize end users into different groups. |  |  |
|  |  |  |
| *Table 1-3: The existing system information* | |  |

**1.6** **The Proposal of system**

**System functions**

****

* Allow the student.
* Allow the student to project registration.
* Allow the student to login by FPT email account.
* Allow the student to view information of mentor supply.
* Allow the student to rename the title project.
* Allow the student to report the process of working in group.
* Allow the student to statistics process of working in group.
* Allow the student to view report and feedback.
* Allow the mentor.
* Allow the mentor to view the list of register.
* Allow the mentor to pick team project to guide.
* Allow the mentor to accept / reject the project.
* Allow the mentor to statistics process of working in group.
* Allow the mentor to view report and feedback.
* Allow the mentor to assessment the process of project.
* Allow the Academic Affairs Department.
* Allow the academic affairs department to mentor view the project to approve or not.
* Allow the academic affairs department to announce to student to change project.
* Allow the academic affairs department to announce the date of project protection.
* Allow the examination council.
* Allow the examination council to Assessment result

**1.6.2 Out of scope functions**

Because of the time limitation, we will not implement these following functions in the CPM initial version 1.0.0 release of the project but not permanently excluded, and will be developed in future version 2.0.0, although we are aware that they so also important:

* CPM adds function for affairs department.
* CPM adds function for examination council.
* CPM complete function assessment the process of project

Finally, we use some new approaches to solve the stated problems. Having awareness of the importance of new technologies, we aim at applying them to our project.

**Special approaches**

****

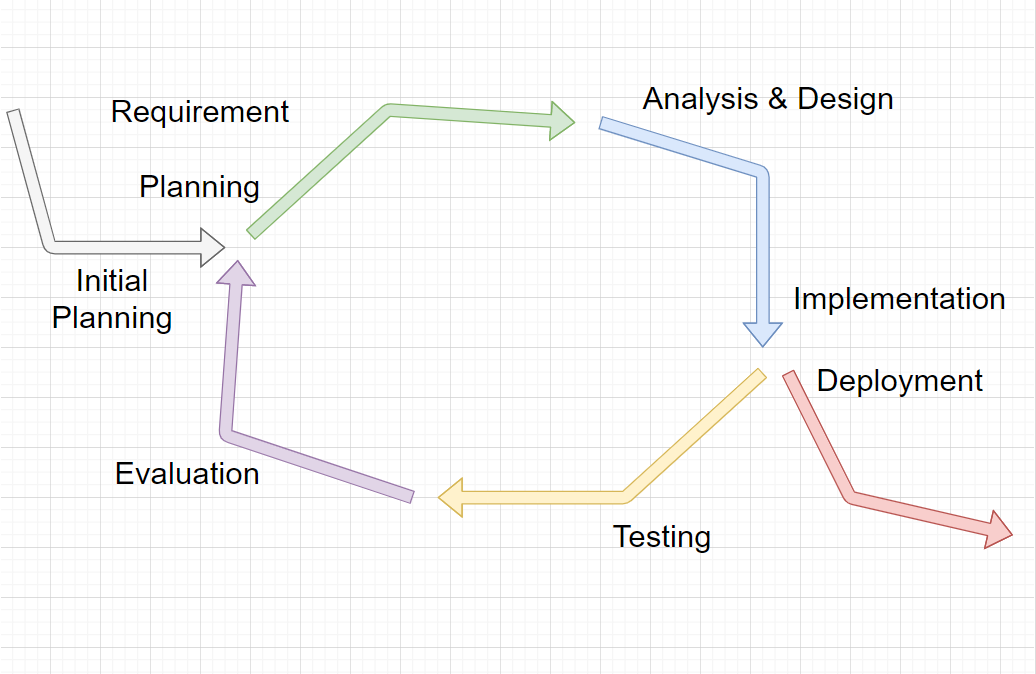
1. Using Java Spring boot for business development for application.
2. Using Spring data JPA.
3. Using MySQL Workbench 8.0 CE to store data.
   1. Using HTML, CSS, Bootstrap create frontend.

**CHAPTER 2: SOFTWARE PROJECT MANAGEMENT PLAN (SPMP)**

## **2.1 Purpose**

* Help readers understand the organization and activities of the project and help readers easily visualize how the project is going.
* Set out the principles that the team members must adhere to and strictly follow the rules throughout the project.
* Each team member must be conscious of the team project, so all team members must use this document to clarify their responsibilities.

**2.1.1 Software process model**



*Figure: Iterative development model*

Project uses the Iterative Process Model

The Iterative Process Model is mostly used when the scope of the project is big, the major requirements are defined clearly, some more details will be added later in software development. By using this software process model, we break down the developing system task into series of smaller tasks which will be completed separately, allowing us to take advantage of what was learned during development of earlier parts of the system. In addition, the iterative model is easier than other models when the issues are discovered. They are fed back to the team, and solutions will be found while the project is still in development.

### **2.1.2 Roles and Responsibilities**

#### **2.1.2.1 Organization Structure**

|  |  |
| --- | --- |
| **Role** | **Responsibility** |
| Project Manager | Planning, divide work among team members, developing schedules, support team members, coordinating communication with team members to feel comfortable and easy to work , always keep the team focused on working for the main goal |
| Technical Leader | Choosing and deciding what technologies should be used, instruct and support team members in using a technology they may not be familiar with, overseeing the work being done by other developers |
| Test Leader | Test execution, including test set-up and test run, evaluation of test run and error recovery, defect logging and test results recording. |
| Developer | Code the product, discuss, exchange and review code of other developers. |
| Designer | Design product’s user interface |
| Tester | Test the product |

*Table 2-1: Project Structure*

#### 

#### **2.1.2.2 Project Team Member**

|  |  |
| --- | --- |
| **Team Member** | **Role** |
| DucDDSE04936 | Project Manager, Technical Leader, Developer, Designer |
| KienBTSE05182 | Test Leader, Developer, Designer, Tester |
| KhanhNNSE05045 | Developer, Designer, Tester |
| LongNDSE05906 | Developer, Designer, Tester |
| KhangNMSE05904 | Developer, Designer, Tester |

*Table: Project Team Member*

### **2.1.2.3 Tool and Techniques**

|  |  |  |
| --- | --- | --- |
| **Software Tool** | **Version** | **Description** |
| Microsoft project | 2013 | Management tools |
| Project | Online | Management tools |
| GitHub | Online | Source code control |
| Source Tree | 3.1.3 | Source code control |
| Eclipse | 2020 | IDE for coding |
| Microsoft Words | 2019 | Document tools |
| Microsoft Excel | 2019 | Document tools |
| MySQL Workbench 8.0 CE | 2019 | My SQL |
| Facebook | Online | Communication tool |
| Skype | Online | Communication tool |
| Process Model |  | Iterative Model |
| Draw.io | Online | Diagram, UML tool, chart |
| Spring Boot | 2.3.3 | IDE for coding |
| Spring Data JPA | 2.0.3 | IDE for coding |
| Hibernate |  | IDE for coding |
| Visual Studio Code | 1.34.0 | IDE for coding |
| HTML |  | IDE for coding |
| CSS |  | IDE for coding |
| Bootstrap |  | IDE for coding |

**2.2 PROJECT MANAGEMENT PLAN**

**2.2.1 Estimate Tasks**

**2.2.2 Meeting minutes**

All meeting minutes will be written following this template:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Meeting/Project  Name: | | FPTUCPM | | | |
| Date of meeting: | | 25/8/2020 | *Time: (Type)* | | *1 hours (Face-to-face)* |
| Meeting Called by: | | DucDD | ***Location:*** | | *FPT University – P205* |
| Note Taker: | | KhangNM | ***Time Keeper:*** | | DucDD |
| 1. Meeting Objective  1. Assign new task for member  2. Do report No.2 | | | | | |
| 2. Attendance | | | | | |
| ***Name*** | ***Roles*** | ***E-mail*** | ***Phone*** | |
| Do Dinh Duc | Project manager | [Ducddse04936@fpt.edu.vn](mailto:Ducddse04936@fpt.edu.vn) | 0968480997 | |
| Bui Trung Kien | Developer ,Tester Leader, Doc | [Kienbtse05182@fpt.edu.vn](mailto:Kienbtse05182@fpt.edu.vn) | 0986268048 | |
| Nguyen Ngoc Khanh | Developer ,Tester, Doc | [Khanhnnse05045@fpt.edu.vn](mailto:Khanhnnse05045@fpt.edu.vn) | 0914092466 | |
| Nguyen Minh Khang | Developer ,Tester, Doc | [Khangnmse05904@fpt.edu.vn](mailto:Khangnmse05904@fpt.edu.vn) | 0971091547 | |
| Nguyen Duc Long | Developer ,Tester, Doc | [Longndse05906@fpt.edu.vn](mailto:Longndse05906@fpt.edu.vn) | 09782996448 | |
| 3. Done task  1. Build project structure  2. Configure git repository with security  3. Do report No.1 | | | | |
| 4. New task  1. Backend    2. Frontend | | | | |
| 5. Risk & Difficulty | | | | |

**2.2.3 Coding Conventions**

**Sonar for eclipse**

**2.2.4 Risk Management Plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Description** | **Avoidance plan** | **Contingency plan** | **Status** |
| R1 | Data loss | Use GitHub for version control.  - Teach members how to use Git  and resolve conflicts.  - Always have important backups. | - Restore backed up  data from GitHub. | Closed |
| R2 | Illness or  absence of  team  members | - Provide schedules in advance.  - For long periods of absence,  members should notify the group  in advance. | - Assign the tasks of  absent member to  other members.  - Work overtime if  necessary. | Closed |
| R3 | Misunderstanding of  Requirements | - Discuss requirements carefully  with the customer.  - Always ask for clarification if  requirement is unclear.  - Comment need to meet | -Make sure idea’s business logic is carefully analysis. | Closed |
| R4 | Requirement  changed | - Every new update of requirement  has to be reviewed by all team  members and supervisor.  - Team member has to analysis  requirement carefully before raise  up to team. | - If requirement has  new update, all  members have to  join the meeting to  aware and make  decision. | Closed |
| R5 | Conflict  between team  members | - Everything must be documented.  - Every team member has to express  clearly and carefully. | - Make sure any  miscommunication  has to be resolved. | Closed |
| R6 | Failure to  meet deadline | - Plan and develop schedule  carefully  - Assign tasks carefully  - Define punishment for team  members who neglect work | - Find the root cause  of the problem  - Reassign tasks  - Focus on important  functions first | Closed |

*Table 2-5: Risk Management*

**2.2.5 Communication Plan**

*Weekly meeting schedule*: We use Iterative and Incremental Process Model, then we divide the system into two sub-systems, each sub-system is divided into a series of small tasks. The amount of work by the whole team, after that the task will be assigned to team members by the Team Leader and depending on difficulty the Technical Leader will assign deadlines for each task. We will have a meeting inform to all team about what each member finished last week, the status (fast, on time or slow), the issues met and how to solve them. If any member raises any issue, the whole team will help to find out a solution together. After that, the team will define detailed stories for next week tasks.

*Daily meeting schedule*: Each sub-system has one development team with different schedule. When starting work-day, each team will have a stand-up meeting to inform to others: “What did I do yesterday?”, “What will I do today?” and “Is there any difficulty?”. By focusing on what each person accomplished yesterday and will accomplish today, the team gains an excellent understanding of what has been done and what remains.

*Unscheduled meeting*: If someone has an important problem that he wants to solve immediately, we will have a meeting for discussion, usually via some online channel: Skype, Facebook or Phone.

*Communication channel*: Our main communication channel is Skype. On the other hand, we use face-to-face meeting, Facebook group and comment issues. However, we sometimes

make a phone call or instant message if someone has a problem.

# **CHAPTER 3: SOFTWARE REQUIREMENTS SPECIFICATION (SRS)**

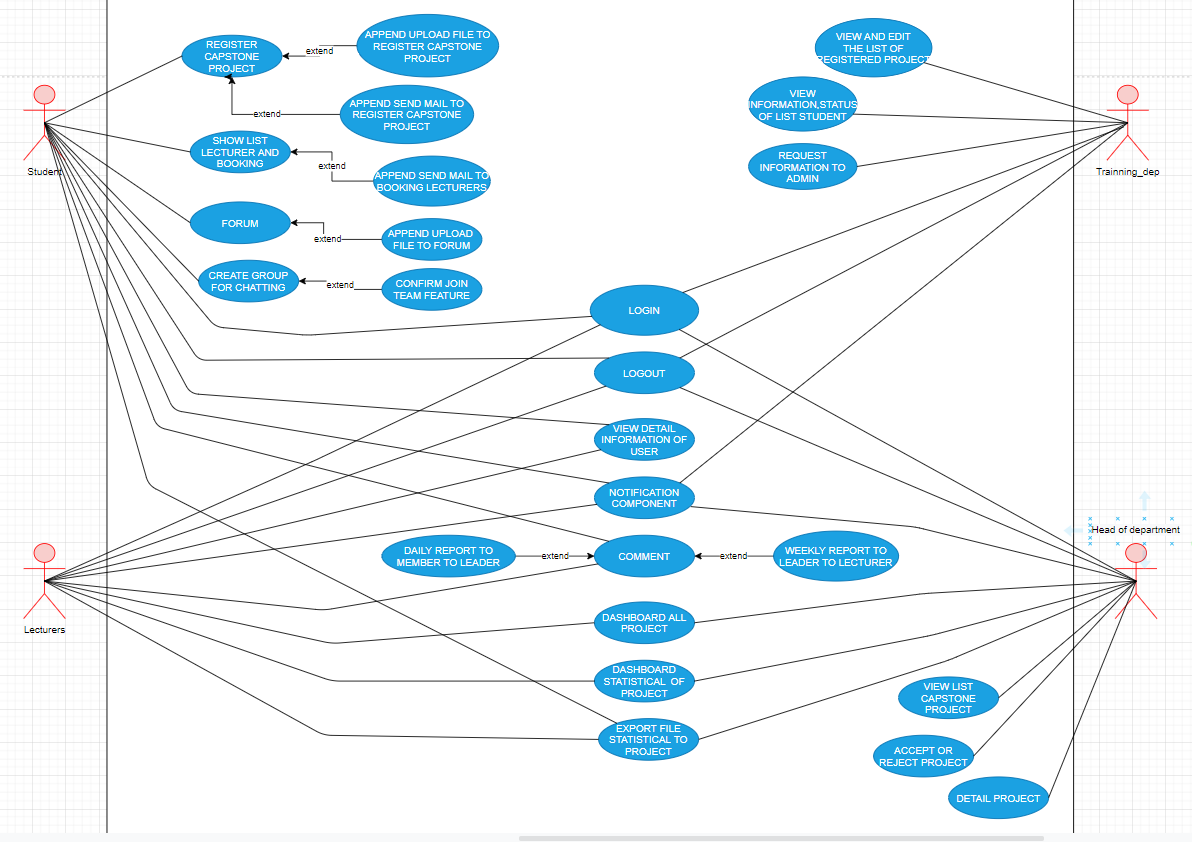
## **3.1 Purpose**

## Software Requirements Specifications chapter provides general, as well as specific information of the system, it outlines functional and non-functional requirements of our website. Following these requirements, FPT Project management system performs its intended functionality used in designing developing and testing. All members work based on the information that provided in this document.

## **3.2 Functional Requirements**

### **3.2.1 Use Case Diagram**

The figure below is display our use case that be had in this project, and it is display on use case diagram.



### **3.2.2 Business Rules**

|  |  |
| --- | --- |
| No | Description |
| B1 | Username must not be empty. |
| B2 | Username must be unique. |
| B3 | User’s full name must not be empty. |
| B4 | User’s address must not be empty. |
| B5 | The email must not be empty. |
| B6 | Email address must be valid. |
| B7 | Each email address may be used for only one account. |
| B8 | Re-enter password must match with password. |
| B9 | Password must not be empty. |
| B11 | The password length must not be less than 6. |
| B12 | Phone number must be valid. |
| B13 | The administrator is able to: active/ deactivate account, change account role, change request state. |
| B14 | Every user accessing dashboard has a role: student, lecturer or training department or head of department |
| B15 | Phone number must be numeric string only. |
| B16 | The administrator is able to: active/ deactivate account, change account role, change request state of student, lecturer. |
| B17 | Exported files must be Microsoft Excel files with valid format. |
| B18 | Answer editing request status only can be changed from “PENDING” to “ACCEPTED” or “REJECTED” |
| B19 | Only “PENDING” editing answer can be accepted by the admin, after that editing answer’s state is “ACCEPTED” |
| B20 | Only “PENDING” editing answer can be rejected by the admin, after that editing answer’s state is “REJECTED” |

*Table: Business rules*

### **3.2.3 Use Cases**

|  |  |
| --- | --- |
| Actor | Description |
| Student | FPT students who have qualified for the project, are preparing to do a project can easily find their project group and register for the project in the fastest and most convenient way. Students will know the information of instructors to choose the right lecturer for the group. And in the process of making projects, students can also exchange and report on weekly work with lecturers. |
| Lecturers | The lecturer looks at the project teams of his choice and makes a decision to accept or reject the instruction. During the project, the lecturer can see the daily and weekly reports sent by the leader of team, and the lecturer will decide whether the project team is protected or not. |
| Training department | The training department will view the project registration list and view information and status of the project group. The training department will decide what the group will do and the lecturer will guide the group. |
| Head of department | The Head of Department will see all of: the information, the project team list and the details of the team's project making process. The Head of Department will decide the success or failure of the project team. |

*Table: Actor description*

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Actor** | **Name** | **Description** |
| **UC-001** | Student, Lecturers, Training department, Head of department | Login | Login to system in FUCPM account. |
| **UC-002** | Student, Lecturers, Training department, Head of department | Logout | Logout to system in FUCPM account |
| **UC-003** | Student | Register Capstone Project | Student register capstone project to system |
| **UC-004** | Student | Upload file to register capstone project | Student upload file : information of team to system |
| **UC-005** | Student | Send mail to register capstone project | Student send mail to register capstone project to system |
| **UC-006** | Student | Show list lectures and booking | Student choose instructors of their group |
| **UC-007** | Student | Send mail to booking lecturers | After the instructor book is finished, the system will send email to the users |
| **UC-008** | Student | Upload file to forum | Student upload file register capstone project to forum |
| **UC-009** | Student | Create group for chatting | Student create group for chatting and invite other student who do not have project group to talk, exchange. |
| **UC-010** | Student | Confirm join team feature | After talking through a group chat, Students will agree to form a group to register project |
| **UC-011** | Student, Lecturers | View detail information of user | Student and lecturers can view detail information of each student and lecturer |
| **UC-012** | Student, Lecturers, Training department, Head of department | Notification component | When there is any information, there will be a notice to the student, lecturers, training department, head of department |
| **UC-013** | Student, Lecturers | Comment | Student and lecturers can comment, exchange information with each other |
| **UC-014** | Student, Lecturers | Daily report to member to leader | Every day, members who complete tasks must report to leader |
| **UC-015** | Student, Lecturers | Weekly report to leader to lecturer | Every week, the leader must report to lecturer during the past week, team has performed |
| **UC-016** | Training department | View and edit the list of registered project | Training department can view 0the list of students who have registered project and can add or change team members |
| **UC-017** | Training department | View information, status of list student | Training department can view information of list student and view status of students who are eligible to do projects or students don’t have groups to add to another group |
| **UC-018** | Training department | Request information to admin | Training department will send information of students to admin |
| **UC-019** | Head of department | View list capstone project | Allow students, lecturer, training department to see the list capstone project |
| **UC-020** | Head of department | View detail project | Head of department can view detail information of list capstone project |
| **UC-021** | Head of department | Accept or reject project | Head of department can accept or reject project of students |
| **UC-022** | Lecturers, Head of department | Dashboard all project | Lecturers and head of department will dashboard all project of registered groups |
| **UC-023** | Lecturers, Head of department | Dashboard statistical of project | Lecturers and head of department will dashboard statistical all information project of registered groups |
| **UC-024** | Student, Lecturer, Head of department | Export file statistical to project | Student, lecturers and head of department will export file statistical to project |

*Table: Use Case List*

### **3.2.4 Use Cases Detail**

***UC-001. Login***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Log in** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student, Lecturers, Training department, Head of department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Allow student, lecturers, training department, head of department who has FUCPM account to login. | | |
| Preconditions: | Student, lecturers, training department, head of department account has been set up and username and password has been generated. | | |
| Post conditions: | 1. Student, lecturers, training department, head of department are logged into FUCPM successfully. 2. System displays Dashboard Home Page. | | |
| Normal Flow: | **Login:** 1. Student visits the website.  2. Student fills valid username and password then clicks “Login”.  3. System displays Dashboard Home page | | |
| Alternative Flows: | Student violates some business rule in step 2  1. System displays help message popup.  2. Student enters username and password complying the business rules then clicks “Login”.  3. Go to step 3 in normal flow. | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-002. Logout***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Logout** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student, Lecturers, Training department, Head of department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | To prevent other users from stealing information | | |
| Preconditions: | User has logged into FUCPM system | | |
| Post conditions: | 1. Student, lecturers, training department, head of department are logged into FUCPM successfully. 2. System displays Dashboard Home Page. | | |
| Normal Flow: | **Logout:** - User is logged out of the system  - Browser displays Login screen  - Access token will be cleared on the browser | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-003. Register Capstone Project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Register Capstone Project** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student register capstone project to system | | |
| Preconditions: | The student has formed his project team | | |
| Post conditions: | 1.The student has successfully formed his project team  2.Students login to their account successfully  3.The system displays the registration page | | |
| Normal Flow: | **Register:**  **-**Students login to their account successfully-Students access the registration page  -Students click the register button  -The system will notify students successfully or failed registration | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-004. Upload file to register capstone project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Upload file to register capstone project** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student upload file : information of team to system to Training department | | |
| Preconditions: | Leader gathered enough information about the group members to complete the word file  Leader complete the word file to upload | | |
| Post conditions: | 1.Students login to their account successfully  2.Leader gathered enough information about the group members to complete the word file to upload  3.The system displays the registration page to upload file | | |
| Normal Flow: | **Upload file:**  **-**Students login to their account successfully-Students access the registration page to upload file  -Students click the upload button  - If the file is large, the system will display a message that cannot upload file | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-005. Send mail to register capstone project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Send mail to register capstone project** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student send mail to register capstone project to system | | |
| Preconditions: | Group has enough to member | | |
| Post conditions: | 1.The group has all the information of its members  2.The group had the project's topic | | |
| Normal Flow: | **Send mail to register project:**  **-**Students login to their account successfully-Students access the email page  -Students click the compose button  -Student upload file word to send to training department | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-006. Show list lectures and booking***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Show list lectures and booking** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student choose instructors of their group | | |
| Preconditions: | Students have registered for the project | | |
| Post conditions: | Students found the details of the lecturers | | |
| Normal Flow: | **Booking lecturer :**  **-**Students login to their account successfully-Students access the list lecturers page  -Student click the name of lecturer to view detail lecturer  - Students return to the list lecturers page  -Students click the choose button  -The system will send a notification you selected successfully | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-007. Send mail to booking lecturers***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Send mail to booking lecturers** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | After the instructor book is finished, the system will send email to student, lecturers, training department, head of department | | |
| Preconditions: | See lecturer details. Choose the instructor you want | | |
| Post conditions: | Book of successful instructors | | |
| Normal Flow: | **Send email:**  **-**Students login to their account successfully-Students access the list lecturers page to choose instructor  -Students click the choose button  -The system will send a notification you selected successfully  - The system will send mail to student, lecturers, training department, head of department | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-008. Upload file to forum***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Upload file to forum** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student upload file register capstone project to forum | | |
| Preconditions: | Student complete the word file to upload | | |
| Post conditions: | 1.Students login to their account successfully  2.Students complete the word file to upload  3.The system displays the forum page to upload file | | |
| Normal Flow: | **Upload file:**  **-**Students login to their account successfully-Students access the forum page to upload file  -Students click the upload button  - If the file is large, the system will display a message that cannot upload file | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-009. Create group for chatting***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Create group for chatting** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student create group for chatting and invite other student who do not have project group to talk, exchange | | |
| Preconditions: | Students have not found their own project group | | |
| Post conditions: | Project group not enough members | | |
| Normal Flow: | **Create group for chatting:**  **-**Students login to their account successfully-Students access the forum page to post  -Students create group for chatting  - If the student is interested, press the join button, it will automatically add to the chat group | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-010. Confirm join team feature***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Confirm join team feature** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | After talking through a group chat, Students will agree to form a group to register project | | |
| Preconditions: | Students have not found their own project group | | |
| Post conditions: | Project group not enough members | | |
| Normal Flow: | **Confirm join team feature:**  **-**Students login to their account successfully-Students access the forum page to post  -Students create group for chatting  - If the student is interested, press the join button, it will automatically add to the chat group  - After the group chat members feel ok, the students in that group will form a group to register the project group. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-011. View detail information of user***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- *View detail information of user*** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student, Lecturer | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student and lecturers can view detail information of each student and lecturer | | |
| Preconditions: | User has logged into FUCPM website. | | |
| Post conditions: | “Homepage” screen will be shown | | |
| Normal Flow: | **View detail:**  **-**Users login to their account successfully-System display “List Lectures” screen  - The user clicks on the name of the lecturer to display detail of lecturer | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

**UC-012. Notification component**

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Notification component** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student, Lecturer, Training department, Head of department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | When there is any information, there will be a notice to the student, lecturers, training department, head of department | | |
| Preconditions: | 1. Users can access the system.  2. Users logins successfully | | |
| Post conditions: | N/A | | |
| Normal Flow: | **Notification component:**  1. Users clicks on “Notification” icon on the left side of the screen.  2. System displays unread notifications.  3. Users clicks on “View All” button to see all notification.  4. Users can click on “More” button to manage notification functions | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

**UC-013. Comment**

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Comment** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student, Lecturer | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student and lecturers can comment, exchange information with each other | | |
| Preconditions: | 1. Student, lecturer can access the system.  2. Student, lecturer logins successfully | | |
| Post conditions: | N/A | | |
| Normal Flow: | **Comment:**  1.Student, lecturer access the forum page to post  2. After posting, students and instructors will comment below the article | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-014. Daily report to member to leader***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Daily report to member to leader** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student, Lecturer | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Every day, members who complete tasks must report to leader | | |
| Preconditions: | Members complete their task | | |
| Post conditions: | N/A | | |
| Normal Flow: | **Report:**  1.Members make their own report on email  2.Send the report to the leader | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

**UC-015. Weekly report to leader to lecturer**

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Weekly report to leader to lecturer** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student, Lecturer | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Every week, the leader must report to lecturer during the past week, team has performed | | |
| Preconditions: | Members complete their task | | |
| Post conditions: | N/A | | |
| Normal Flow: | **Report:**  1.Members make their own report on mail  2.Send the report to the leader  3.Leader will send the report of the group to the instructor | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-016. View and edit the list of registered project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- View and edit the list of registered project** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Training department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Training department can view the list of students who have registered project and can add or change team members | | |
| Preconditions: | Training department has logged into FUCPM website | | |
| Post conditions: | “Registered project page” screen will be shown | | |
| Normal Flow: | **View and edit:**  **1.**Training department login to their account successfully2.System display “List of registered project” screen  3. The dropdown section containing the lecturers and student information will be displayed on the screen  4.Training department will select a group in the dropdown to see list of registered project  5. Training department can change the instructors for the project registration group | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

**UC-017. View information, status of list student**

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- View information, status of list student** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Training department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Training department can view information of list student and view status of students who are eligible to do projects or students don’t have groups to add to another group | | |
| Preconditions: | Training department has logged into FUCPM website | | |
| Post conditions: | “Registered project page” screen will be shown | | |
| Normal Flow: | **View:**  **1.**Training department login to their account successfully2.System display “List of registered project” screen  3. The dropdown section containing the lecturers and student information will be displayed on the screen  4. Training department will select a group in the dropdown to see information, status of list student  5.Training department can switch status from registered \_ capstone -> approve\_ capstone \_ lecturers ->  approve\_ capstone\_ head ->doing\_ capstone | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-018. Request information to admin***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Request information to admin** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Training department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Training department will send information of students to admin | | |
| Preconditions: | Training department has logged into FUCPM website | | |
| Post conditions: | N/A | | |
| Normal Flow: | **Request:**  **1.**Training department login to their account successfully2.System display “List of registered project” screen  3.Training department clicks on request button to send information list of registered project to admin | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-019. View list capstone project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- View list capstone project** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Head of department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Allow students, lecturer, training department to see the list capstone project | | |
| Preconditions: | Students, lecturer, training department have logged into FUCPM website | | |
| Post conditions: | “List capstone project” screen will be shown | | |
| Normal Flow: | **View:**  1. Students, lecturer, training department access in FUCPM website.  2.System display “List capstone project” screen  3. The dropdown section containing the lecturers and student information will be displayed on the screen  4.Training department will select a group in the dropdown to see list of capstone project  5. Students, lecturer, training department click to “See More” button to view more list capstone project | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-020. View detail project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- View detail project** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Head of department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Head of department can view detail information of list capstone project | | |
| Preconditions: | Students have logged into FUCPM website | | |
| Post conditions: | “List capstone project” screen will be shown | | |
| Normal Flow: | **View:**  1. Students, lecturer, training department access in FUCPM website.  2.System display “List capstone project” screen  3. The dropdown section containing the lecturers and student information will be displayed on the screen  4.Training department will select a group in the dropdown to see list of capstone project and view detail project | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-021. Accept or Reject project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- *Accept or Reject project*** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Head of department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Head of department can accept or reject project of students | | |
| Preconditions: | Student, lecturers, training department has logged into FUCPM website | | |
| Post conditions: | Student, lecturers, training department has logged successfully into FUCPM website | | |
| Normal Flow: | **Accept or Reject project:**  1.Training department access in FUCPM website.  2.System display “List capstone project” screen  3. The action button has 2 buttons approve, reject, and pending (press the button with pop-up confirm if you want to approve or reject:  + In the capstone information, when you press the approve reject button, there will be approve reject for the capstone  + In each line of the dropdown there is each approve, reject button of each member | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-022. Dashboard all project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- *Dashboard all project*** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Lecturers, Training department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Lecturers and head of department will dashboard all project of registered groups | | |
| Preconditions: | 1. Student, lecturers, training department, head of department can access the system.  2. Student, lecturers, training department, head of department login successfully | | |
| Post conditions: | N/A | | |
| Normal Flow: | **Dashboard:**  1. Students, lecturer, training department, head of department access in FUCPM website.  2.System display “Home page” screen  3. Users clicks on “Dashboard all project” icon on the top side of the screen.  4. System displays dashboard all project | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-023. Dashboard statistical of project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- *Dashboard statistical of project*** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Lecturer, Training department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Lecturers and head of department will dashboard statistical all information project of registered groups | | |
| Preconditions: | 1. Student, lecturers, training department, head of department can access the system.  2. Student, lecturers, training department, head of department login successfully | | |
| Post conditions: | N/A | | |
| Normal Flow: | **Dashboard statistical:**  1. Students, lecturer, training department, head of department access in FUCPM website.  2.System display “Home page” screen  3. Users clicks on “Dashboard statistical” icon on the top side of the screen.  4. System displays dashboard statistical of project | | |
| Alternative Flows: | N/A | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

***UC-024. Export file statistical to project***

*Use Case Specification*

|  |  |  |  |
| --- | --- | --- | --- |
| **UC ID and Name:** | **UC- Export file statistical to project** | | |
| Create by: |  | Date Created: | 10/08/2020 |
| Primary Actor: | Student, lecturer, head of department | Secondary Actor: |  |
| Trigger: | N/A | | |
| Description: | Student, lecturers and head of department will export file statistical to project | | |
| Preconditions: | Student, lecturers and head of department have logged into FUCPM website | | |
| Post conditions: | Student, lecturers and head of department can export all data of website | | |
| Normal Flow: | 1.Student, lecturers and head of department click on kind of data the employee wants to export  2.The page displays the basic list of data  3.Student, lecturers and head of department click in to “Export Data” button at the bottom of the page  4.The system allows student, lecturers and head of department to download data to computers using excel files | | |
| Alternative Flows: | 1.At the step 1, instead of clicking data type button, Student, lecturers and head of department click to “Back to list” button.  2. System redirects admin to the “View all categories” screen without changing anything at database.  3. System displays error message(s) on the create form | | |
| Exceptions: | Cannot communicate with server 1. FUCPM displays error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

**3.3 Non-functional Requirement**

**3.3.1 Security**

* The system must use token-based authentication
* The authentication server returns an access token user logged in with email.
* All passwords are encrypted
* The security matrix is as the following table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function** | **Student** | **Lecturer** | **Training department** | **Head of department** |
| Login | X | X | X | X |
| Logout | X | X | X | X |
| Register Capstone Project | X |  |  |  |
| Upload file to register capstone project | X |  |  |  |
| Send mail to register capstone project | X |  |  |  |
| Show list lecturers and booking | X |  |  |  |
| Send mail to booking lecturers | X |  |  |  |
| Upload file to forum | X |  |  |  |
| Create group for chatting | X |  |  |  |
| Confirm join team feature | X |  |  |  |
| View detail information of user | X | X |  |  |
| Notification component | X | X | X | X |
| Comment | X | X |  |  |
| Daily report to member to leader | X | X |  |  |
| Weekly report to leader to lecturer | X | X |  |  |
| View and edit the list of registered project |  |  | X |  |
| View information, status of list student |  |  | X |  |
| Request information to admin |  |  | X |  |
| View list capstone project |  |  |  | X |
| View detail project |  |  |  | X |
| Accept or reject project |  |  |  | X |
| Dashboard all project |  | X |  | X |
| Dashboard statistical of project |  | X |  | X |
| Export file statistical to project | X | X |  | X |

**3.3.2 Supportability**

* The application should be easy to use by the user
* The application must have all necessary features to support the user
* The application can be easily extended

## **3.3.3 Accuracy**

* The overall accuracy is calculated by diving total number of correct answers by the number of questions asked

## **3.3.4 Maintainability and Extensibility**

* Source code must follow the coding convention in order to help developers to understand, improve readability of source code and make the website more maintainable or modify the code.

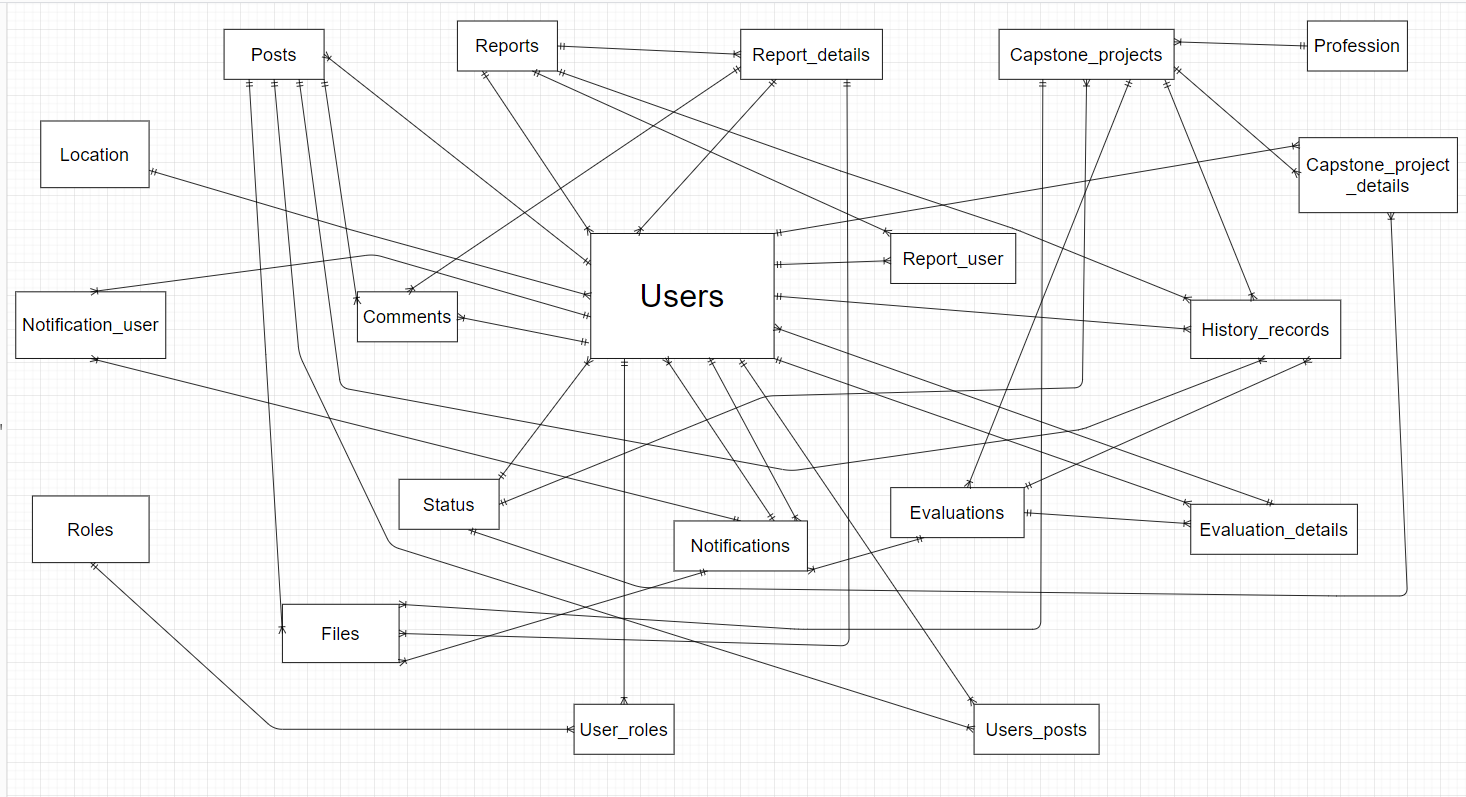
## **3.3.5 Performance**

* Web frontend uses bootstrap to make the website design process simpler and faster, it is also a free toolkit for open source code to help website templates become more complete and eye-catching.
* This software requires a powerful server machine with high band internet access.
* The app must work on all platforms
* The system must be usable when many people use it at the same time or at any time
* Web uses spring boot to help us programmers to simplify the process of programming an application with Spring, focusing only on developing business for the application.

## **3.3.6 Usability**

* The interface is beautiful, good-looking for users, simple and easy to use, it does not break at various zoom levels
* Links and buttons are easily clickable
* Must help users to use the application in the most satisfied way
* The screen layout is clearly streamlined
* Screen should be attractive and attractive to the user
* Main functions are organized into tabs for easy and convenient user access
* Having response buttons and clickable menus to help guide end users to their answer when needed
* Buttons containing important commands must be unique

**3.4 Entity Relationship Diagram**



# **CHAPTER 4: SOFTWARE DESGIN**

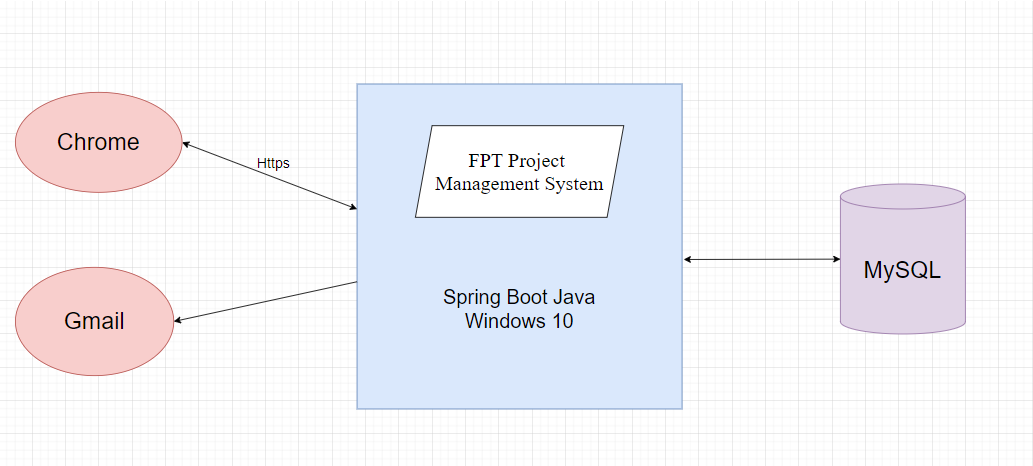
## **4.1 Purpose**

* This chapter is to give the developer team an overview of what the system’s architecture is, and how they should be implemented. This chapter consists of:
* Architecture overview
* Component diagram
* Detailed design
* Detailed description of components
* Database design

## **4.2 Architectural overview**

### **4.2.1** **System Architecture**

#### **4.2.1.1 Diagram**



*Figure: System overview*

### **4.2.2 System Architectural Explanation**

#### **4.2.2.1 MVC Architectural**

##### **4.2.2.1.1 View application – HTML5 & CSS3 & JavaScript & JQuery & Bootstrap.**



* Using HTML5 & CSS3 & JavaScript & JQuery & Bootstrap provides advantage of making the website design process simple, fast, eye-catching, and allows users to freely create ideas, create more and more diverse products. Design also helps to save maximum time designing website interface. Not only that, it also brings many advantages for businesses to develop, web developers can easily use and use in the most convenient way.

#### **4.2.2.2 Controller**

* In Spring Boot, @Controller annotation indicates that the annotated class is a controller. It is a specialization of @Component and is auto detected through class path scanning. It is typically used in combination with annotated handler methods based on the @Request Mapping annotation. @Rest Controller is a sibling convenience annotation for creating Restful controllers.
* In the following application, we demonstrate the usage of @Controller. The application returns current data and time to the client.

#### **4.2.2.3 Model**

##### **4.2.2.3.1 SQL Database – MySQL Workbench 8.0 CE**



*Figure: MySQL Workbench 8.0 CE*

* MySQL Workbench is a visually modelled and designed database access tool for use with

a relational MySQL server database. MySQL Workbench helps to create new physical data models and supports modification of existing MySQL databases with reverse / forward techniques, optional management functions.

#### **4.2.2.4 Service providers**

##### **4.2.2.4.1 Session state**

* The session state service manages session state for every user. They store the state in the memory, in external process or in MySQL database. The membership service stores user names, passwords and other membership data in MySQL

##### **4.2.2.4.2 Mail service**

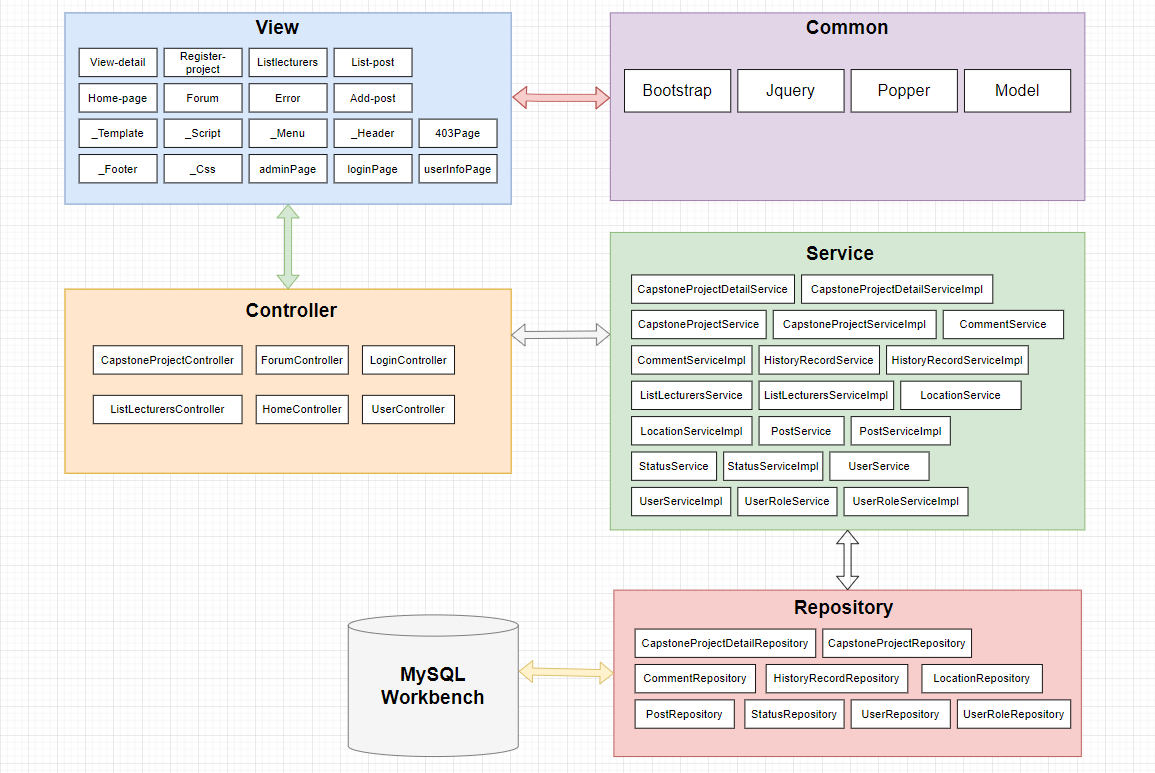


*Figure: Gmail*

## **4.3 Design of FPT Project Management System**

### **4.3.1 Architectural Layers Design**

#### **4.3.1.1 FUCPM – API Layers Design**



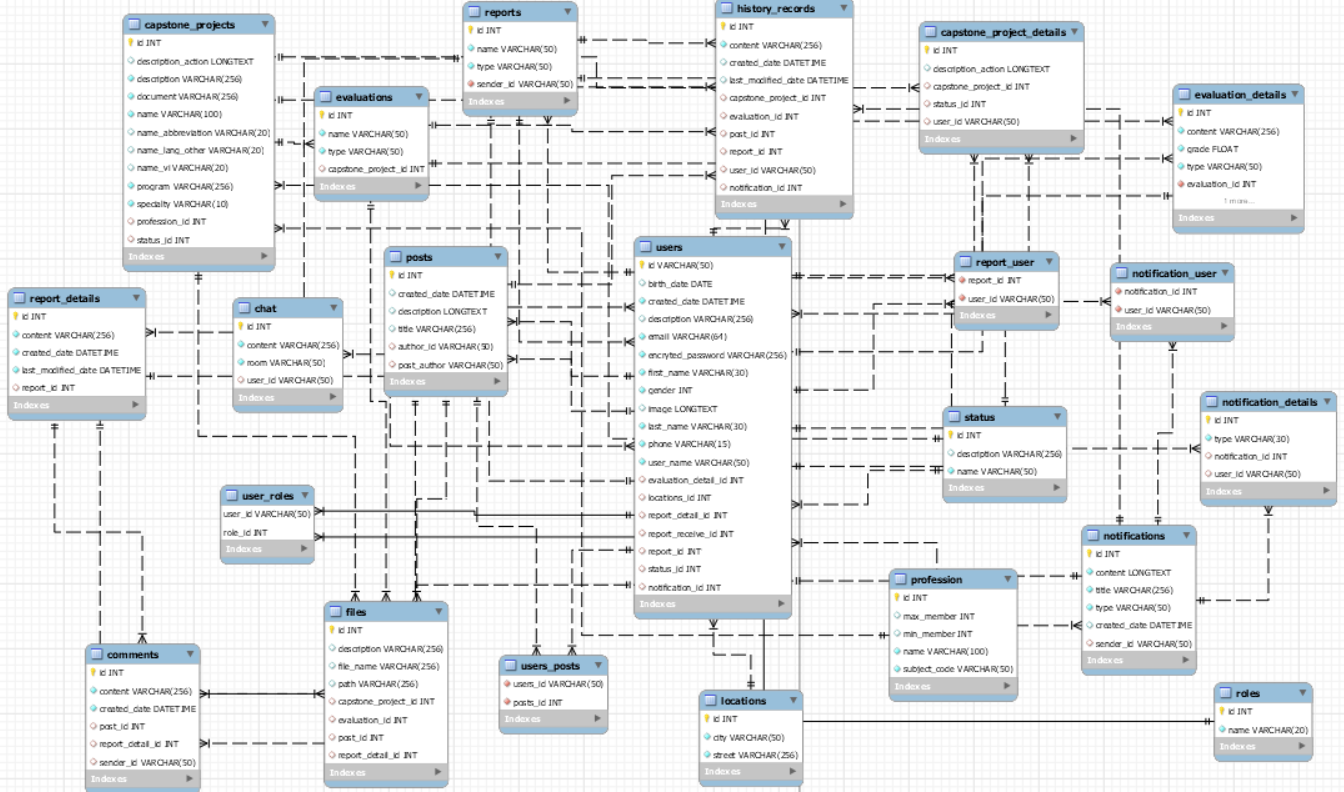
*Figure: FUCPM- API Layers Design*

#### **4.3.1.2 Package design**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Directory** | **Description** | **Convention** |
| 1 | /CONFIG | Where contains the basic configuration of the website | N/A |
| 2 | /Controller | The place we call from the service to handle the URL request | N/A |
| 3 | /DTO | Where DTO to Entity and Entity to DTO Conversion | N/A |
| 4 | /Entity | Where we create entity classes. Each entity will describe a table in DB | N/A |
| 5 | /Repository | The place we write the database query to call from the database | N/A |
| 6 | /Service | The place we call from the repository to process logic and then push back to the controller | N/A |
| 7 | /UTILS | Where we use the web | N/A |
| 8 | /CSS | Where we make the website more beautiful | N/A |
| 9 | /FONTAWESOME | Where we put UI template files that are responsible for rendering output, such as metadata, sprites, … | N/A |
| 10 | /Fonts | Where we choose the right font size | N/A |
| 11 | /Image | Where we put the used images on our website | N/A |
| 12 | /JS | Where we put JavaScript library files and scripts | N/A |
| 13 | /Error | Where we check all the errors | N/A |
| 14 | /Home | Where contains html files to design the homepage interface | N/A |
| 15 | /Login | Where contains html files to design the login interface | N/A |

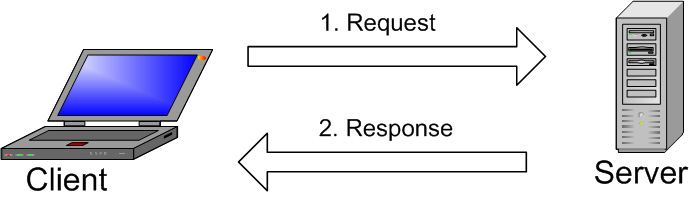
### **4.3.2 Database Design**

#### **4.3.2.1 Database diagram**



*Figure: FUCPM database document diagram*

#### **4.3.2.2 Explanation of database decision**



The image above explains the basic structure of the client-server communication model. A client machine communicates with the server machine in a given network. Each client can send a request from the Graphical user interface (GUI) on the screen, and the server will return the desired result. As long as both understand each other. The same goes for the main operation in a MySQL environment:

* MySQL creates tables to store data, defining the relationships between those tables.
* The client will send the SQL request with a special command on MySQL.
* The application on the server will respond to the information and return the results on the client machine.

So, we chose MySQL because:

* Flexible and easy to use: You can modify the source code to meet your needs without paying any extra costs. The installation process is also very simple and usually takes less than 30 minutes.
* High Performance: Many server clusters use MySQL. Whether you store big data from e-commerce sites or do heavy IT businesses, MySQL can handle it with smooth, high speeds.
* Industry standard: The data and technology industry has been using MySQL for many years, so it is a basic skill as a programmer. MySQL users can also deploy projects quickly and hire data experts for a fee if they need to.
* Safety: Data security is always the most important issue when choosing RDBMS software. With decentralized access and account management, MySQL has a very high standard of security. Host-based authentication and credential encryption are available.

#### **4.3.2.3 Data Dictionary**

##### **4.3.2.3.1 Class Diagram, Table description & Table design**

* **Class Diagram**

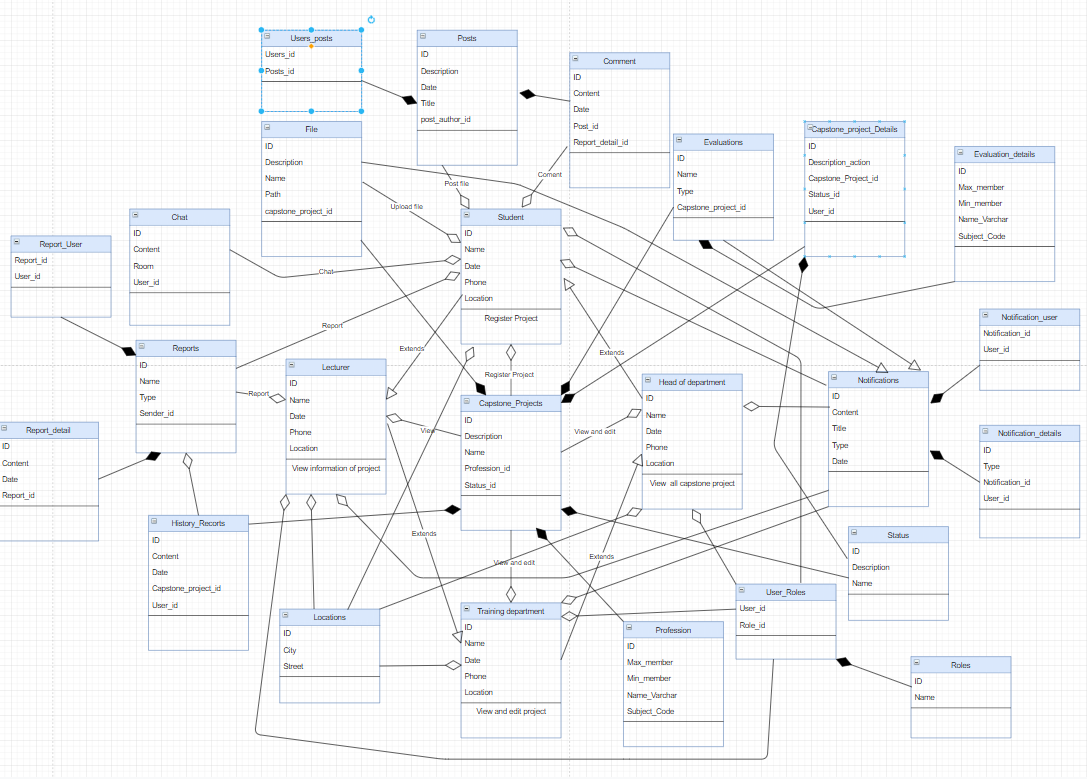


Figure: Class diagram

* **Table description**

|  |  |  |
| --- | --- | --- |
| **No** | **Table Name** | **Description** |
| 1 | Users | Save information users |
| 2 | Location | Save address information of users |
| 3 | Notification-user | When there are new events will notify the users |
| 4 | Roles | 4 role: Student, Lecturers, Training department, Head of department |
| 5 | Post | Save post of student |
| 6 | Comments | Save comment of student |
| 7 | Status | Show and save the status of students who are eligible to do graduate projects or not |
| 8 | Files | Save uploaded student file |
| 9 | Reports | Send evaluation information about training department to students |
| 10 | User-roles | To save which user corresponds to which role |
| 11 | Notifications | Save notifications if there are new events or information changes |
| 12 | Report-details | Send reports detail from member to leader and from leader to instructor |
| 13 | Report-user | Send reports from member to leader and from leader to instructor |
| 14 | Evaluations | Send evaluation information from instructor to students |
| 15 | Users-posts | To save which user corresponds to which post |
| 16 | Capstone-projects | Save information of capstone project |
| 17 | History-records | Save history with transactions such as adding students, delete students ... |
| 18 | Evaluation-details | Send evaluation detail information from instructor to students |
| 19 | Profession | Save profession information of student |
| 20 | Capstone-projects-details | Save detail information of capstone project |
| 21 | Chat | Save student chats |
| 22 | Notification-details | Save detail notification the training department to students, or instructor to students |

* **Table design**

## **4.3.2.3.1.1: Users**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | VARCHAR (50) |  |  |
| BIRTH\_DATE | DATE | NULL |  |
| CREATED\_dATE | DATETIME |  |  |
| DESCRIPTION | VARCHAR (256) | NULL |  |
| EMAIL  ENCRYTED\_PASSWORD  FIRST\_NAME  GENDER  IMAGE  LAST\_NAME  PHONE  USER\_NAME  EVALUATION\_DETAIL\_ID  LOCATIONS\_ID  REPORT\_DETAIL\_ID  REPORT\_RECEIVE\_ID  REPORT\_ID  STATUS\_ID  NOTIFICATION\_ID | VARCHAR (64)  VARCHAR (256)  VARCHAR (30)  INT  LONGTEXT  VARCHAR (30)  VARCHAR (15)  VARCHAR (50)  INT  INT  INT  INT  INT  INT  INT | NULL  NULL  NULL  NULL  NULL  NULL  NULL  NULL |  |

## **4.3.2.3.1.2: Location**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| city  street | VARCHAR (50)  VARCHAR (256) |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## **4.3.2.3.1.3: Notification-user**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| NOTIFICATION\_ID | INT |  |  |
| User\_id | VARCHAR (50) |  |  |

## **4.3.2.3.1.4: Roles**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| NAME | VARCHAR (20) |  |  |

## **4.3.2.3.1.5: Post**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| created\_date  description  title  author\_id  post\_author | DATETIME  LONGTEXT  VARCHAR (256)  VARCHAR (50)  VARCHAR (50) | NULL  NULL  NULL  NULL  NULL |  |

## **4.3.2.3.1.6: Comments**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| content  created\_Date  post\_id  report\_detail\_id  sender\_id | VARCHAR (256)  DATETIME  INT  INT  VARCHAR (50) | NULL  NULL  NULL |  |

## **4.3.2.3.1.7: Status**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| DESCRIPTION | VARCHAR (256) | NULL |  |
| NAME | VARCHAR (50) |  |  |

## **4.3.2.3.1.8: Files**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| DESCRIPTION | INT |  |  |
| FILE\_NAME  PATH  CAPSTONE\_PROJECT\_ID  EVALUATION\_ID  POST\_ID  REPORT\_DETAIL\_ID | VARCHAR (256)  VARCHAR (256)  VARCHAR (256)  INT  INT  INT | NULL  NULL  NULL  NULL  NULL  NULL |  |

## **4.3.2.3.1.9: Reports**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| NAME  TYPE  SENDER\_ID | VARCHAR (50)  VARCHAR (50)  VARCHAR (50) |  |  |

## **4.3.2.3.1.10: User-roles**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| USER\_ID | VARCHAR (50) |  |  |
| ROLE\_ID | INT |  |  |

## **4.3.2.3.1.11: Notifications**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| content  title  type  created\_date  sender\_id | LONGTEXT  VARCHAR (256)  VARCHAR (50)  DATETIME  VARCHAR (50) | NULL  NULL |  |

## **4.3.2.3.1.12: Report-details**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| LAST\_MODIFIED\_DATE | DATETIME |  |  |
| REPORT\_ID | INT | NULL |  |

## **4.3.2.3.1.13: Report-user**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| REPORT\_ID | INT |  |  |
| USER\_ID | VARCHAR (50) |  |  |

## **4.3.2.3.1.14: Evaluations**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| name  type  capstone\_project\_id | VARCHAR (50)  VARCHAR (50)  INT | NULL |  |

## **4.3.2.3.1.15: Users-post**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| USERS\_ID | VARCHAR (50) |  |  |
| POSTS\_ID | INT |  |  |

## **4.3.2.3.1.16: Capstone-projects**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| DESCRIPTION\_ACTION  DESCRIPTION  DOCUMENT  NAME  NAME\_ABBREVIATION  NAME\_LANG\_OTHER  NAME\_VI  PROGRAM  SPECIALTY  PROFESSION\_ID  STATUS\_ID | LONGTEXT  VARCHAR (256)  VARCHAR (256)  VARCHAR (100)  VARCHAR (20)  VARCHAR (20)  VARCHAR (20)  VARCHAR (256)  VARCHAR (10)  INT  INT | NULL  NULL  NULL  NULL  NULL  NULL |  |

## **4.3.2.3.1.17: History-records**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| content  created\_date  last\_modified\_date  capstone\_projecT\_id  evaluation\_id  post\_id  report\_id  user\_id  notification\_id | VARCHAR (256)  DATETIME  DATETIME  INT  INT  INT  INT  VARCHAR (50)  INT | NULL  NULL  NULL  NULL  NULL  NULL  NULL  NULL |  |

## **4.3.2.3.1.18: Evaluation-details**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| content  grade  type  evaluation\_id  assessor\_id | VARCHAR (256)  FLOAT  VARCHAR (50)  INT  VARCHAR (50) |  |  |

## **4.3.2.3.1.19: Profession**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| max\_member  min\_member  name  subject\_code | INT  INT  VARCHAR (100)  VARCHAR (50) | NULL  NULL |  |

## **4.3.2.3.1.20 : Capstone-projects-details**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ID | INT |  |  |
| DESCRIPTION\_ACTION  CAPSTONE\_PROJECT\_ID  STATUS\_ID  USER\_ID | VARCHAR (50)  VARCHAR (256) | NULL  NULL  NULL  NULL |  |

## **4.3.2.3.1.21: Chat**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| ROOM | VARCHAR(50) |  |  |
| USER\_ID | VARCHAR (50) | NULL |  |

## **4.3.2.3.1.2: Notification-details**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Allow Nulls** | **Description** |
| NOTIFICATION\_ID | INT | NULL |  |
| USER\_ID | VARCHAR (50) | NULL |  |

## **4.4 Detail Design**

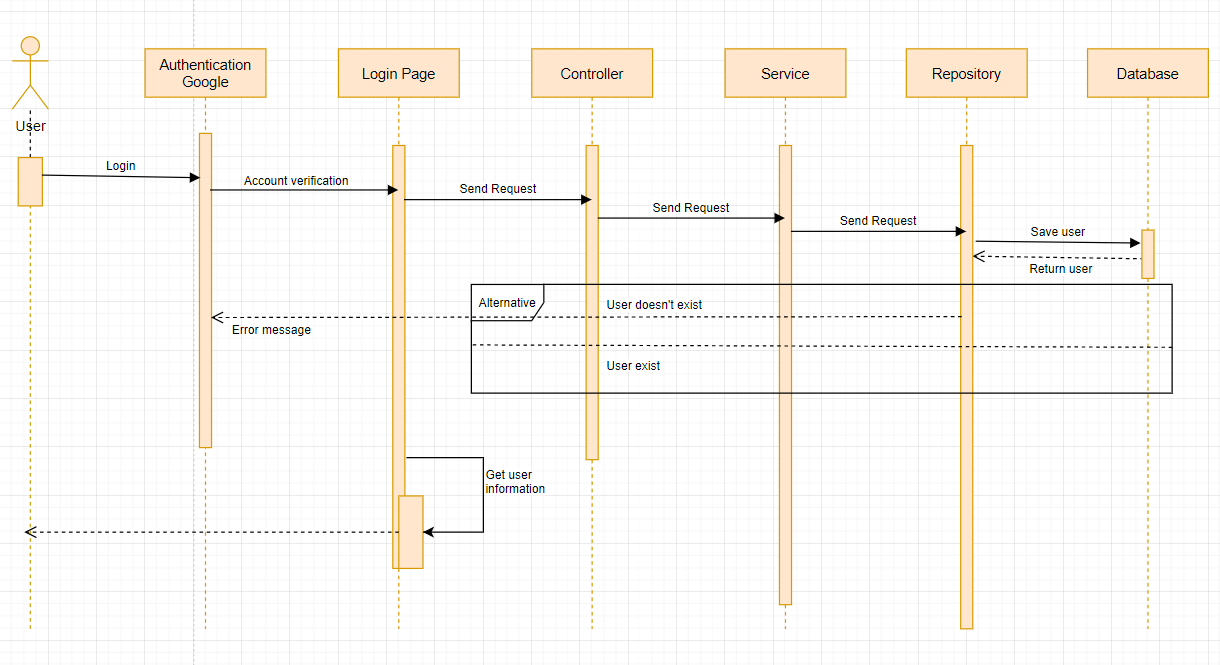
**4.4.1 Sequence Diagram**

**4.4.1.1: Login**

**Screen Design:**

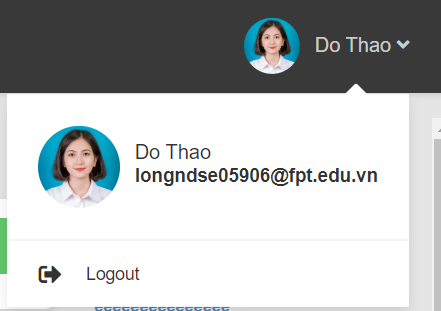
****

**Sequence Diagram:**

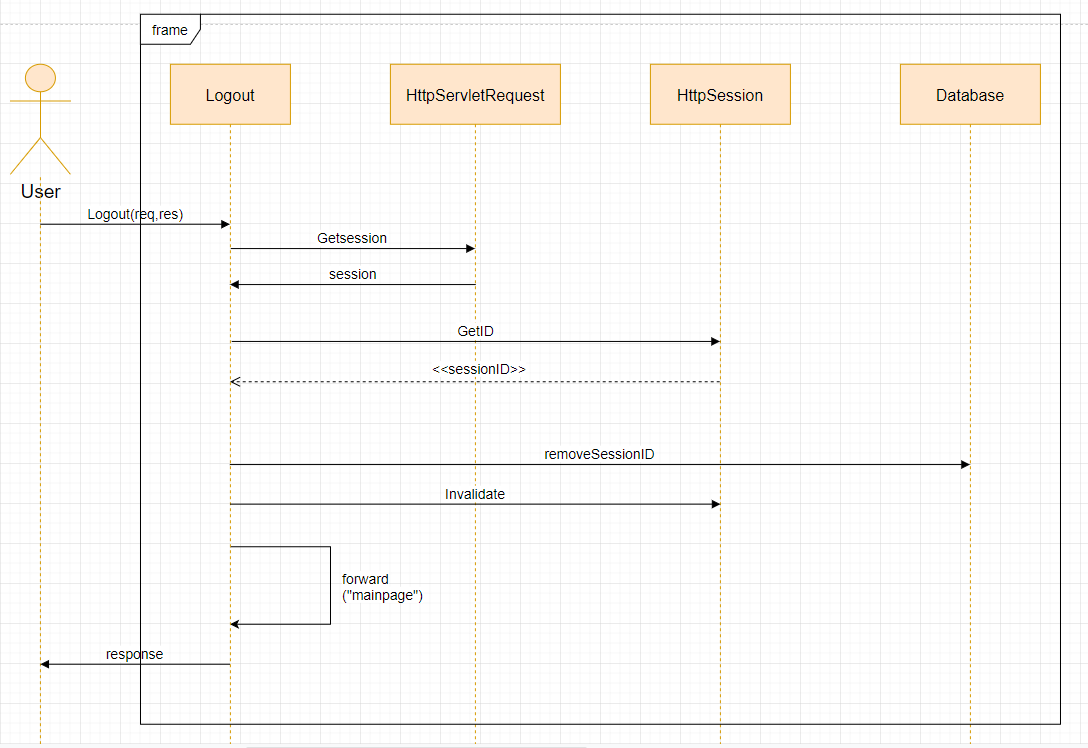


**4.4.1.2: Logout**

**Screen Design:**

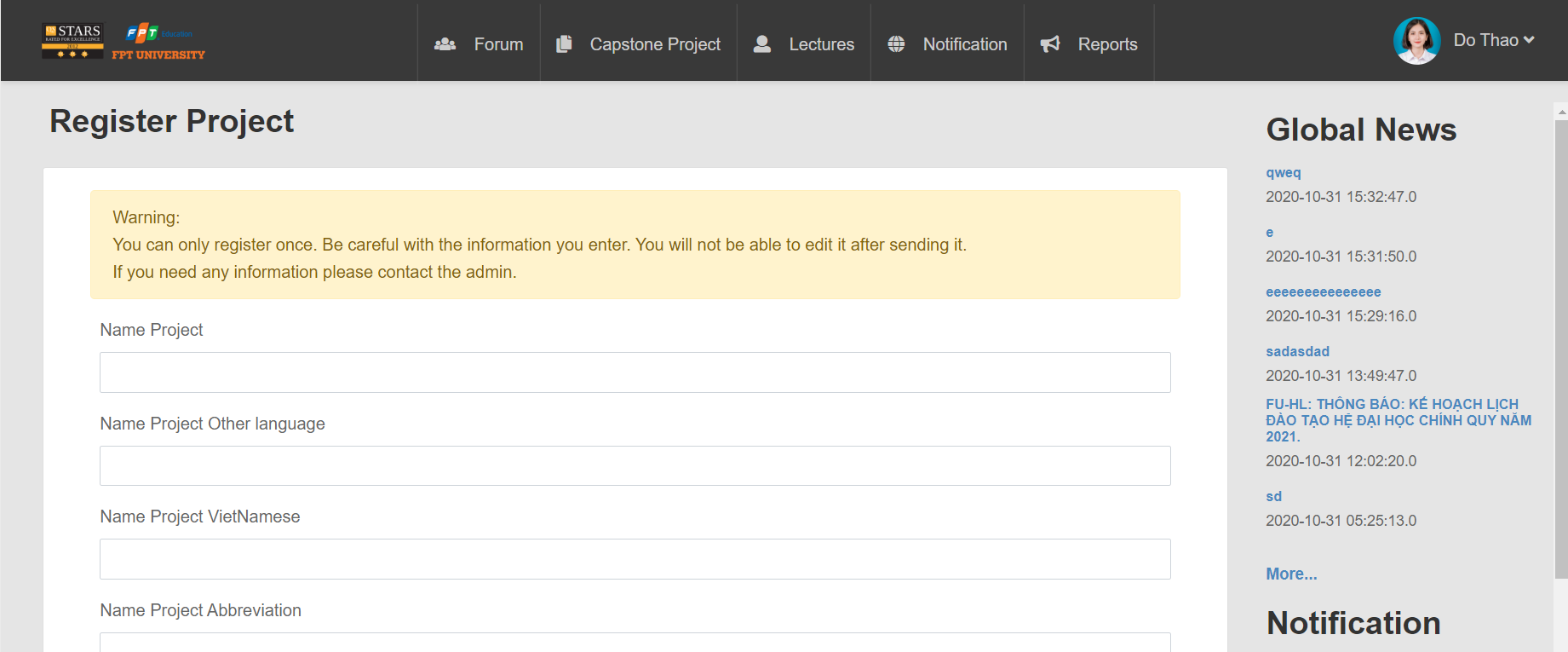
****

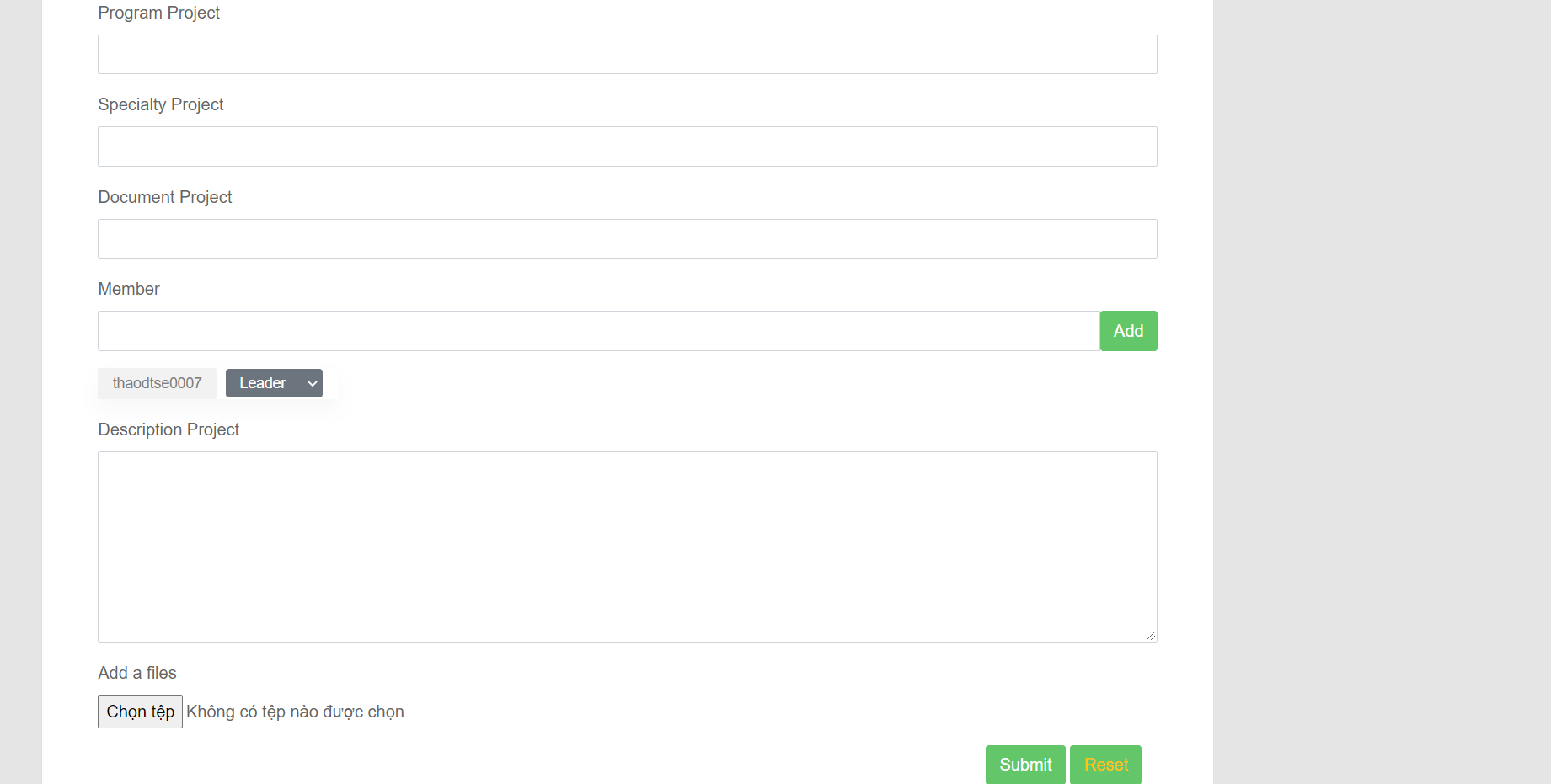
**Sequence Diagram:**

****

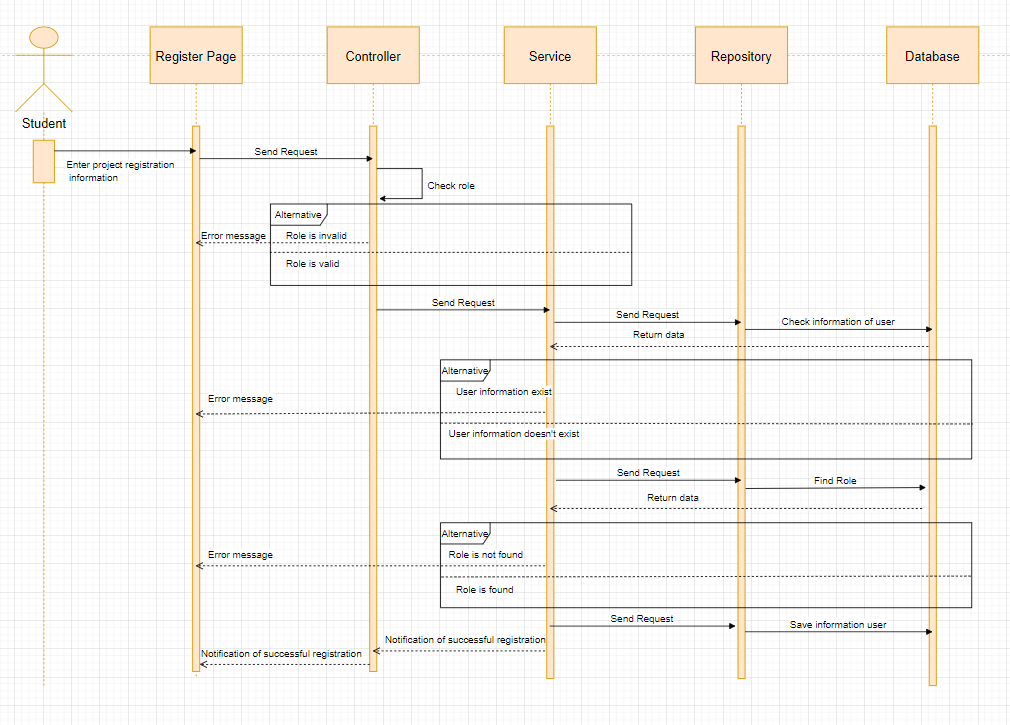
**4.4.1.3: Register Capstone Project**

**Screen Design:**

****

****

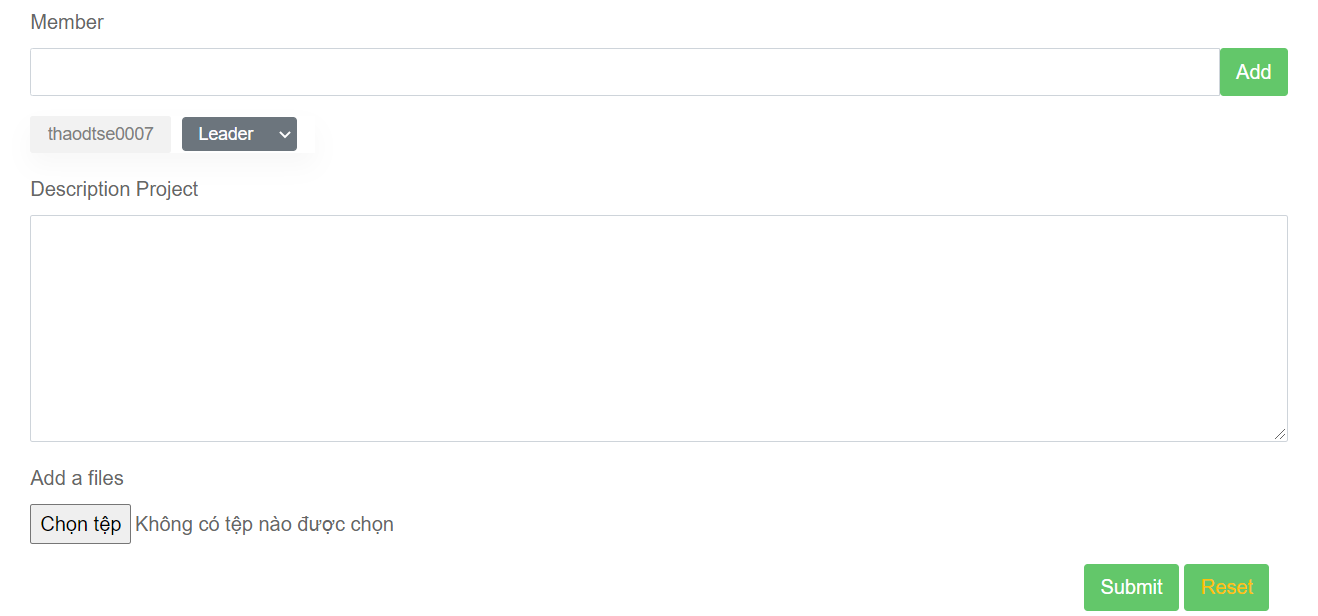
**Sequence Diagram:**

****

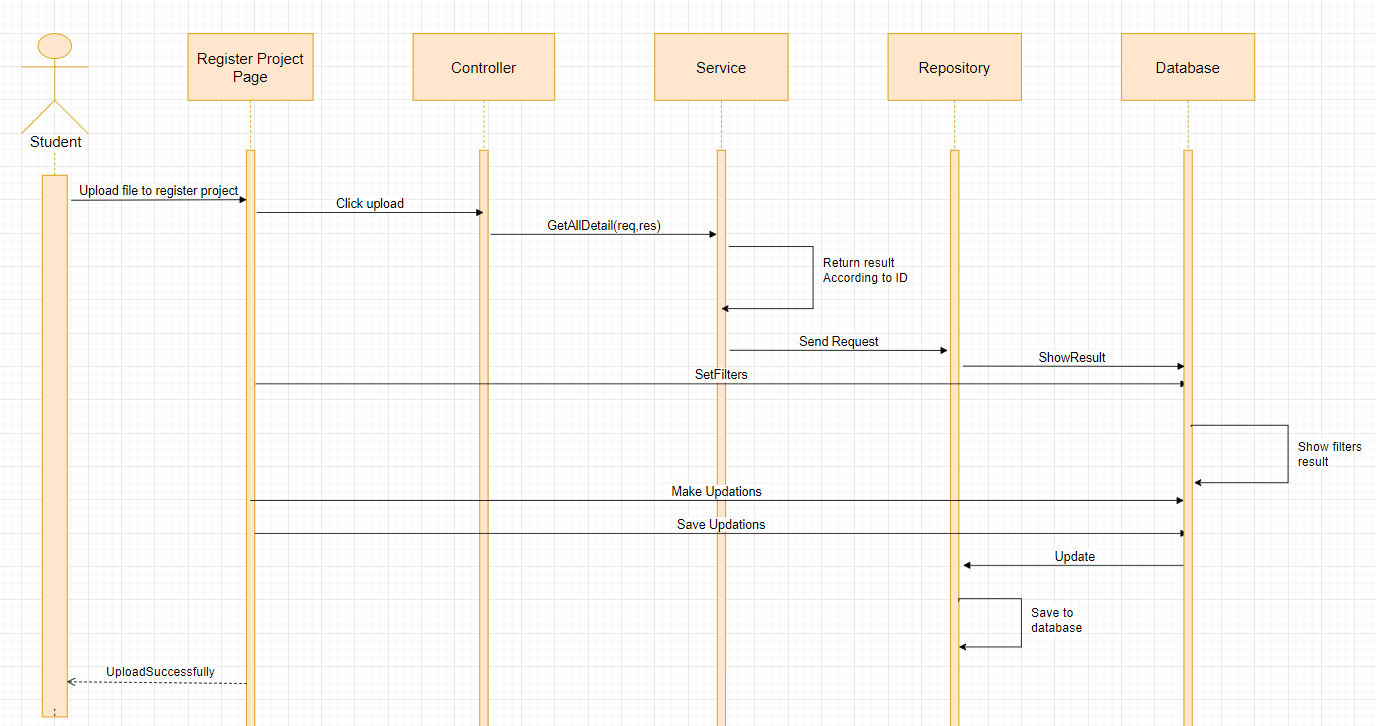
**4.4.1.4: Upload file to register capstone project**

**Screen design:**

**Sequence Diagram:**

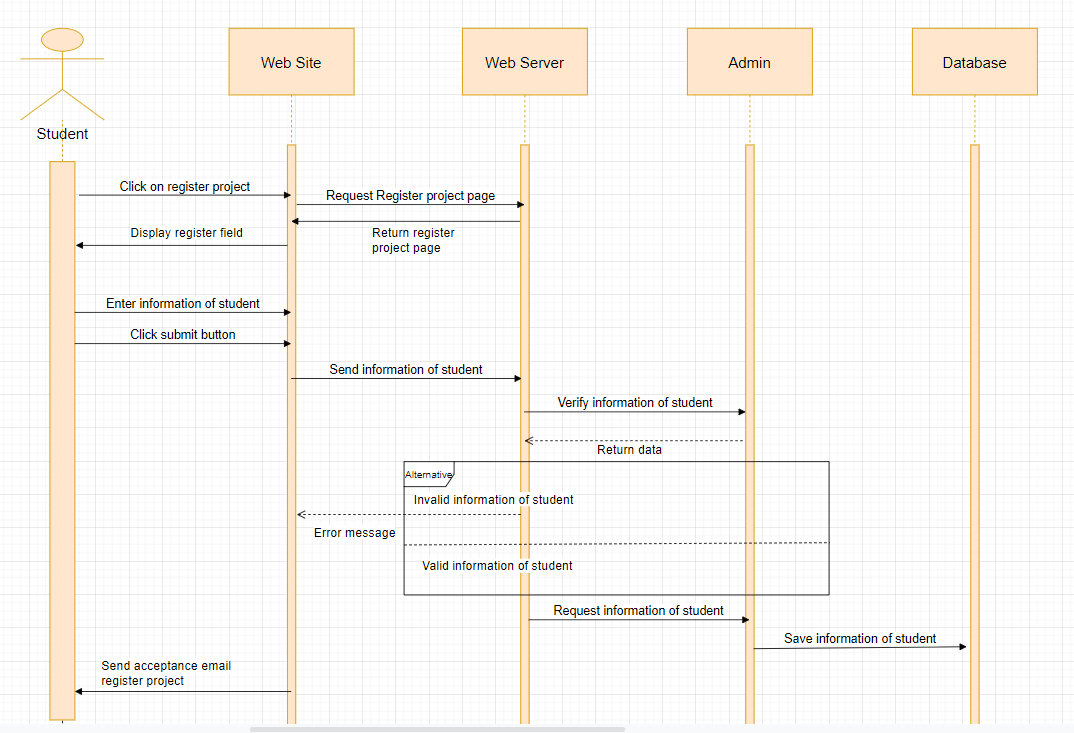
****

**Sequence Diagram:**

****

**4.4.1.5: Send mail to register capstone project**

**Sequence Diagram:**

****

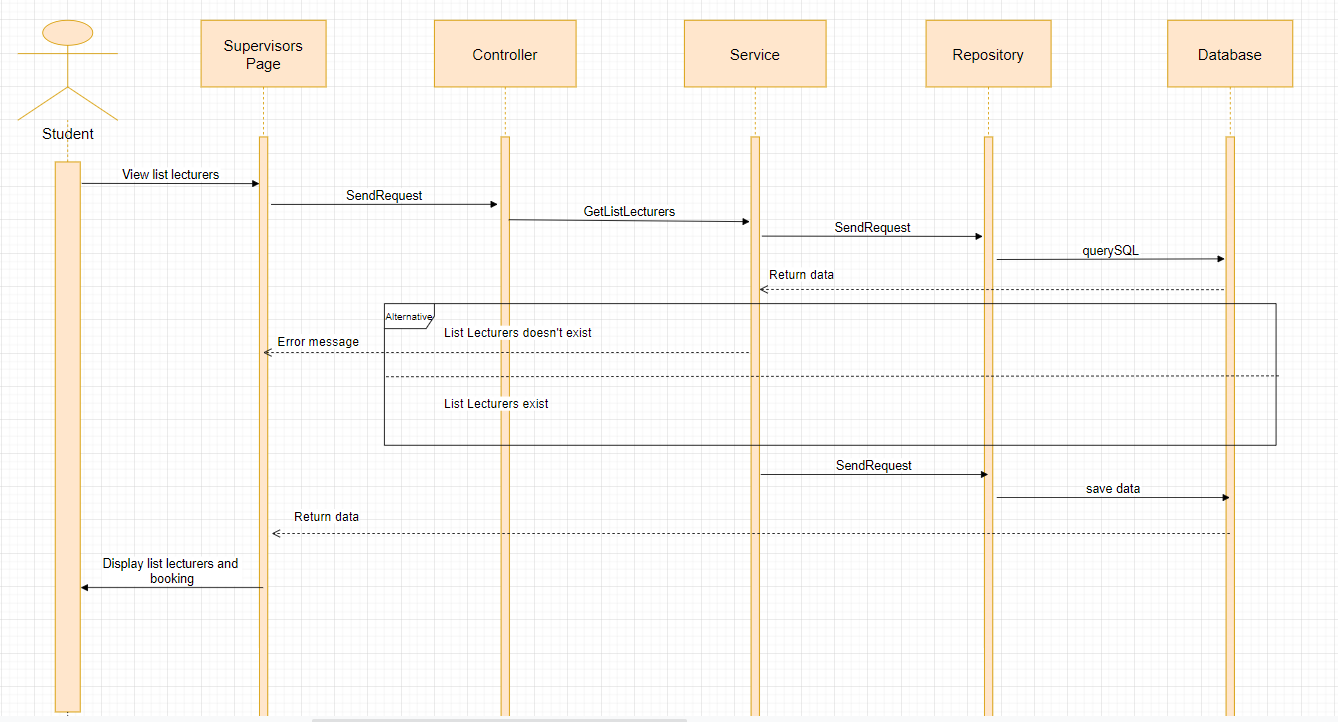
**4.4.1.6: Show list lecturers and booking**

**Screen Design:**

****

****

**Sequence Diagram:**

****

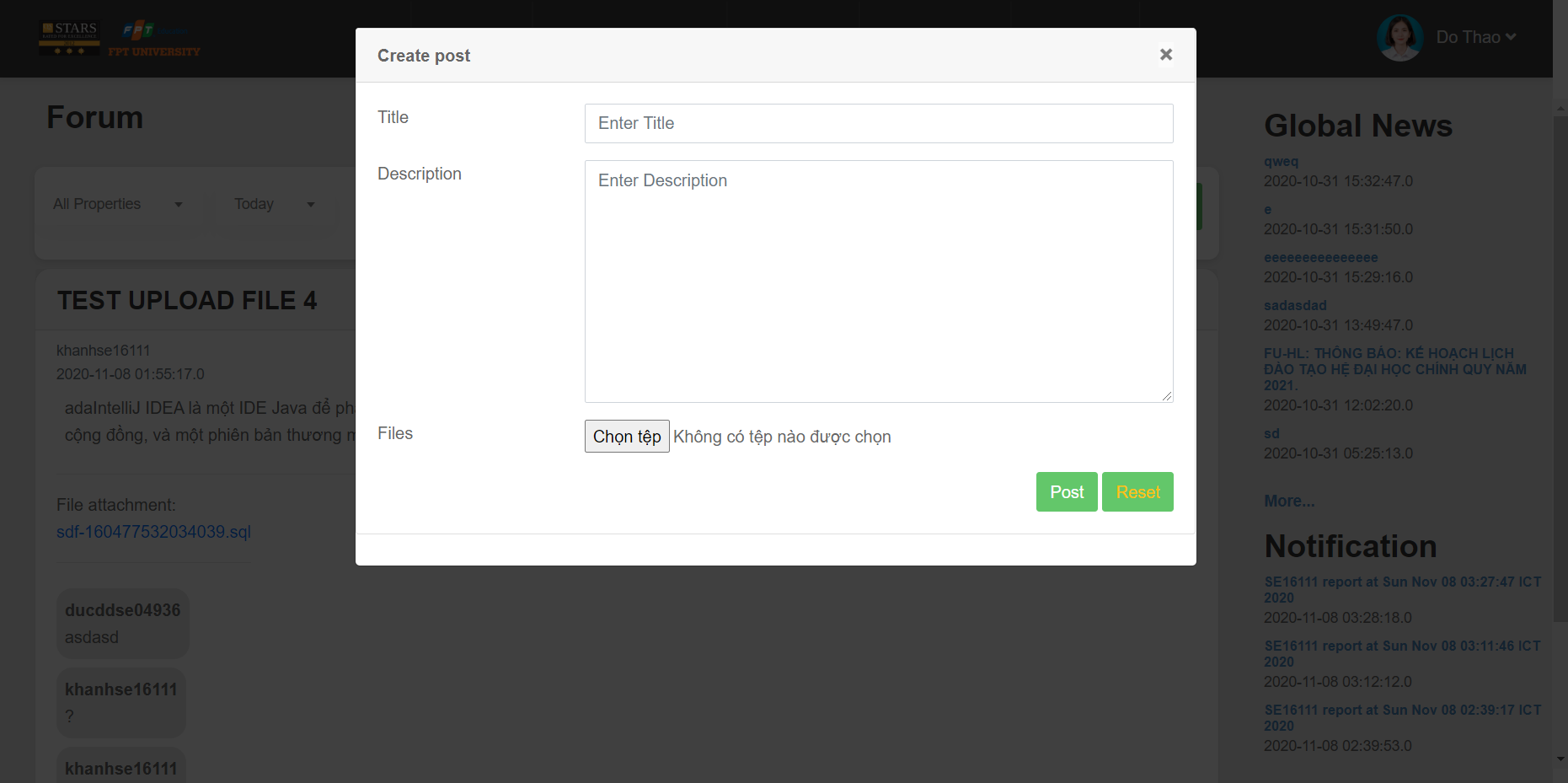
**4.4.1.7: Send mail to booking lecturer**

**Sequence Diagram:**

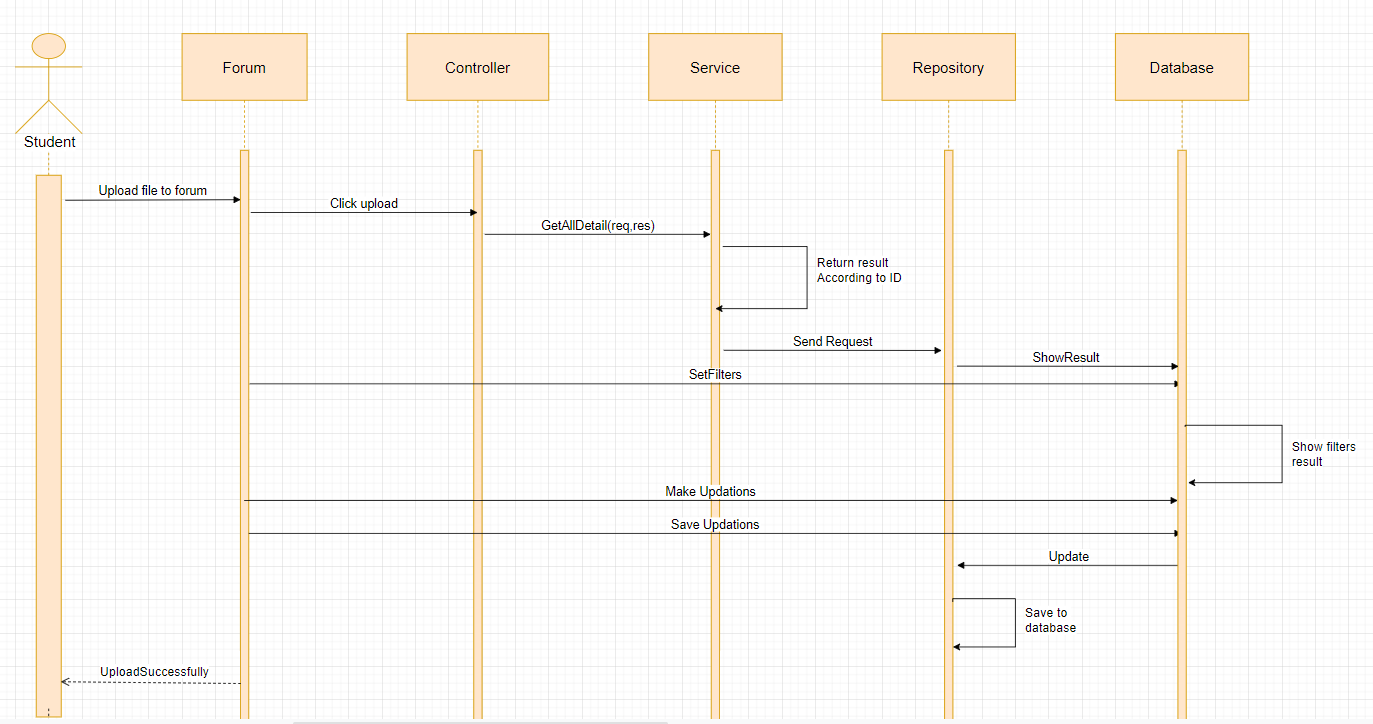
****

**4.4.1.8: Upload file to forum**

**Screen design:**

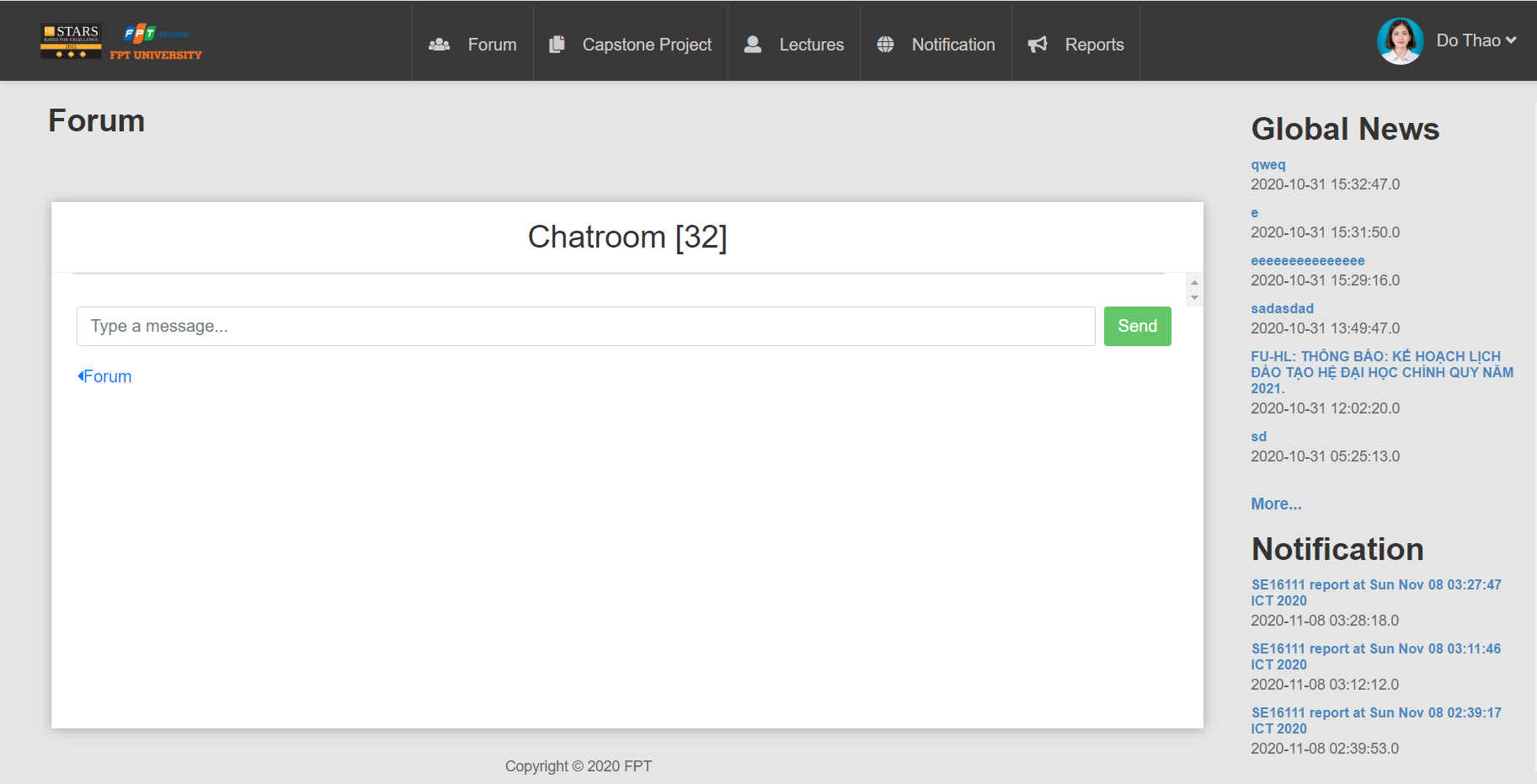
****

**Sequence Diagram:**

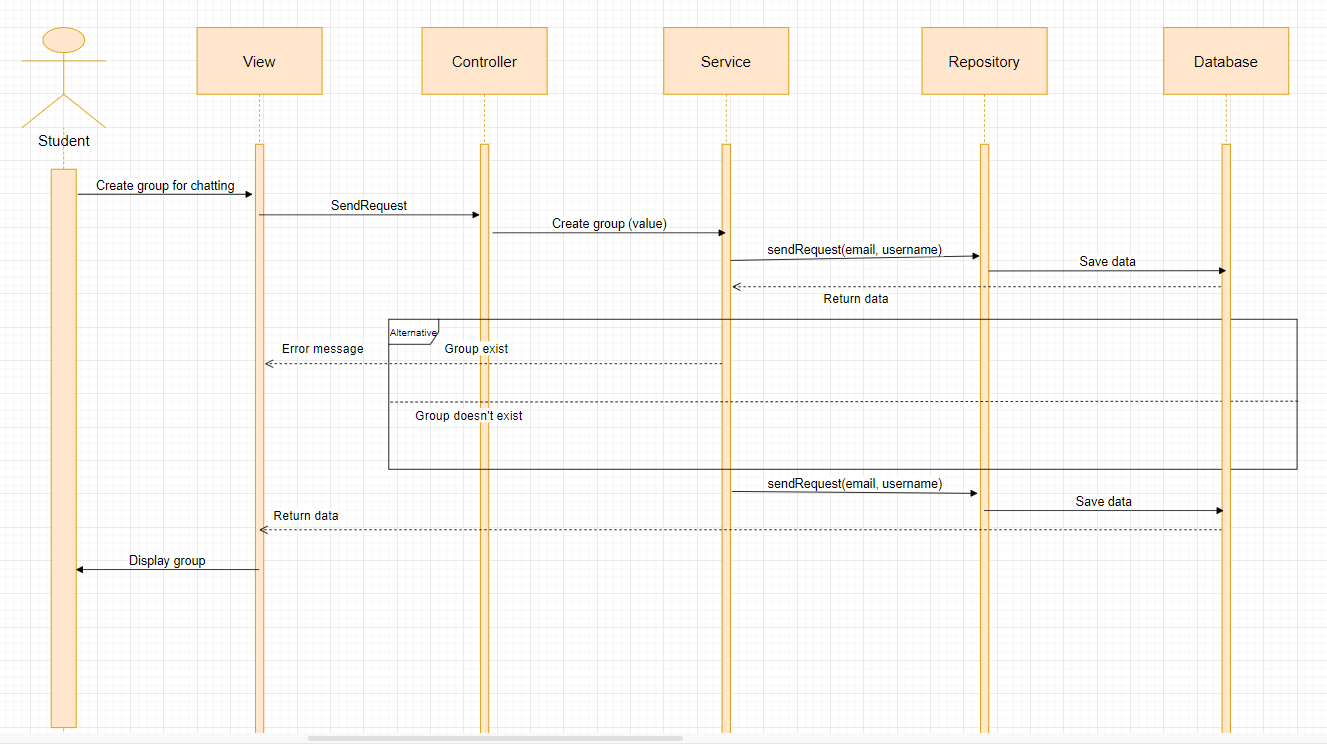
****

**4.4.1.9: Create group for chatting**

**Screen design:**

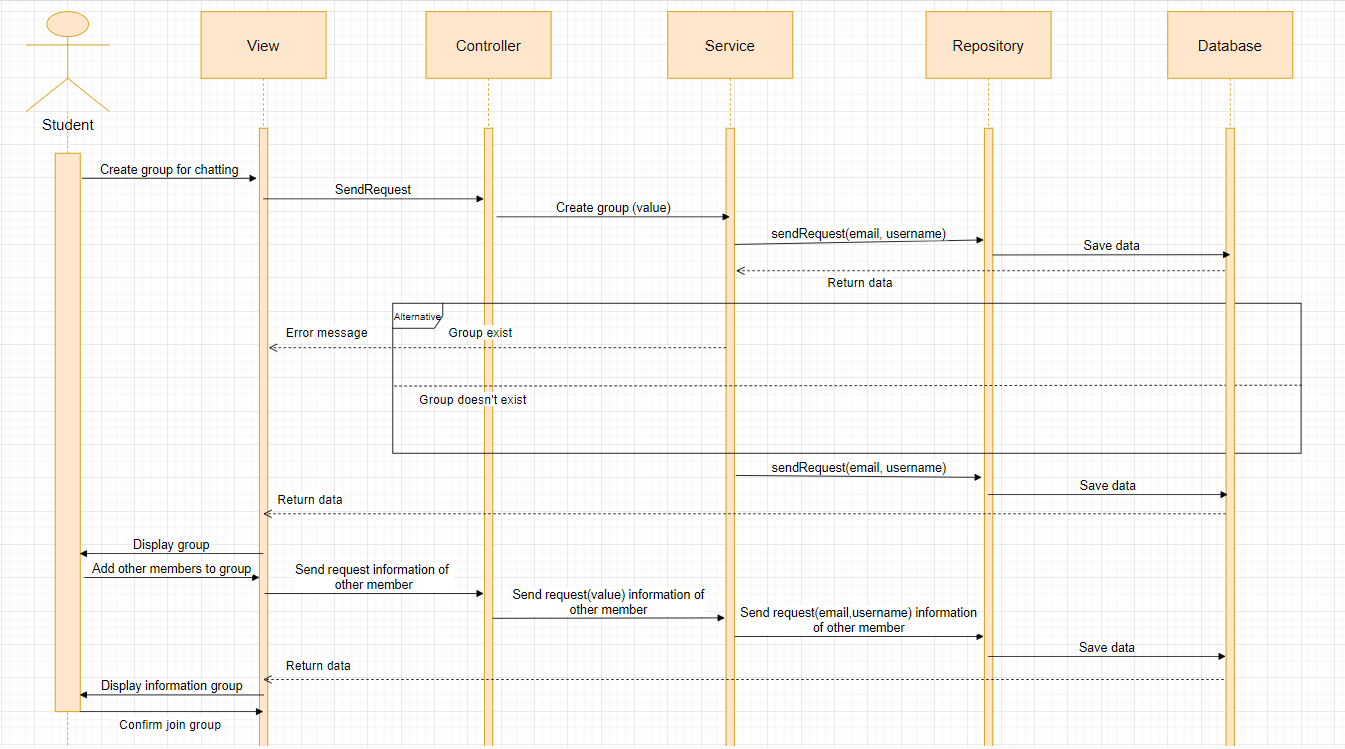
****

**Sequence Diagram:**

****

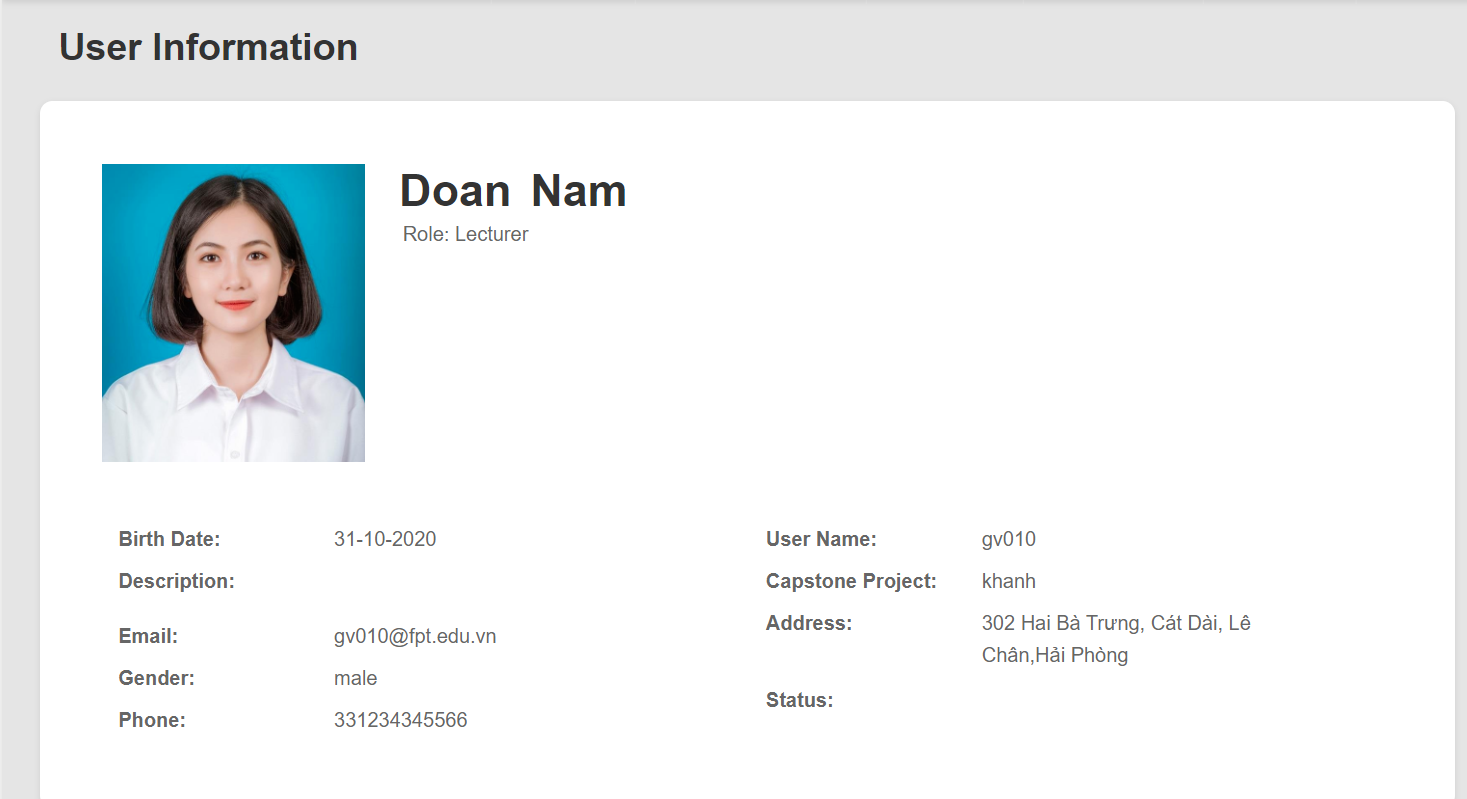
**4.4.1.10: Confirm join team feature**

**Sequence Diagram:**

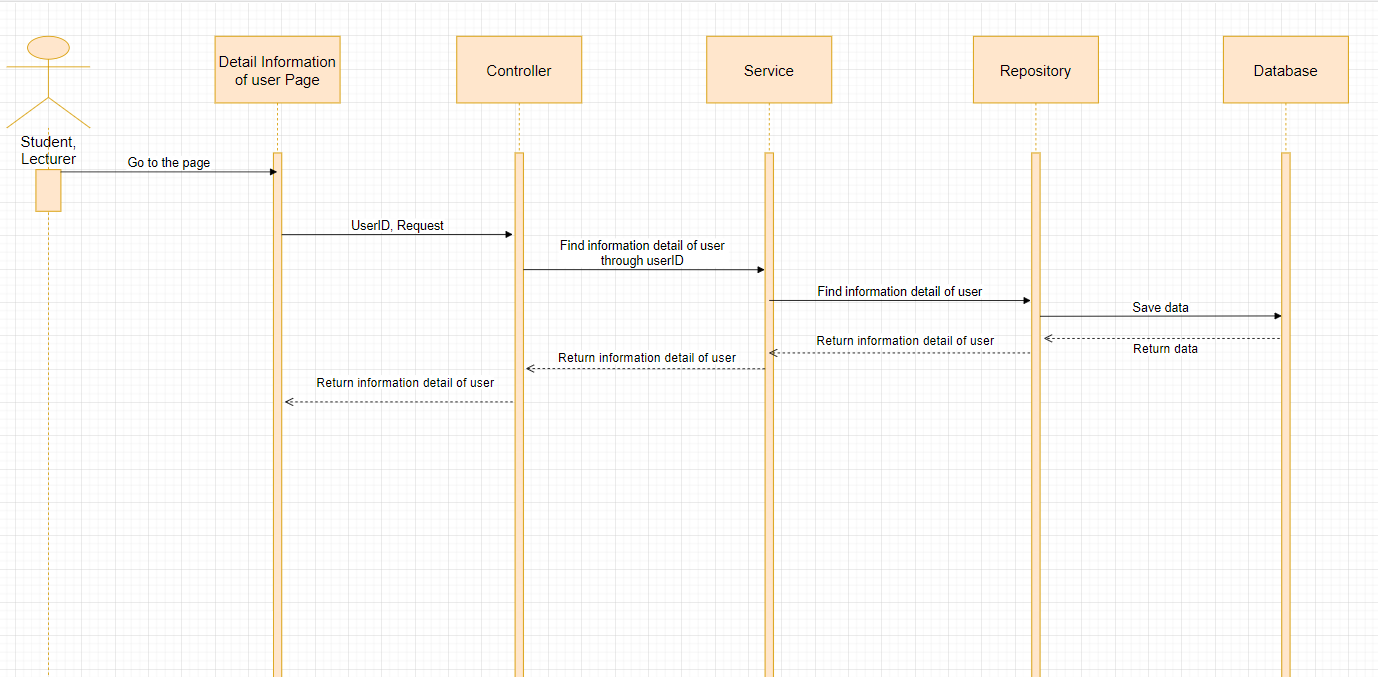
****

**4.4.1.11: View detail information of user**

**Screen design:**

****

**Sequence Diagram:**

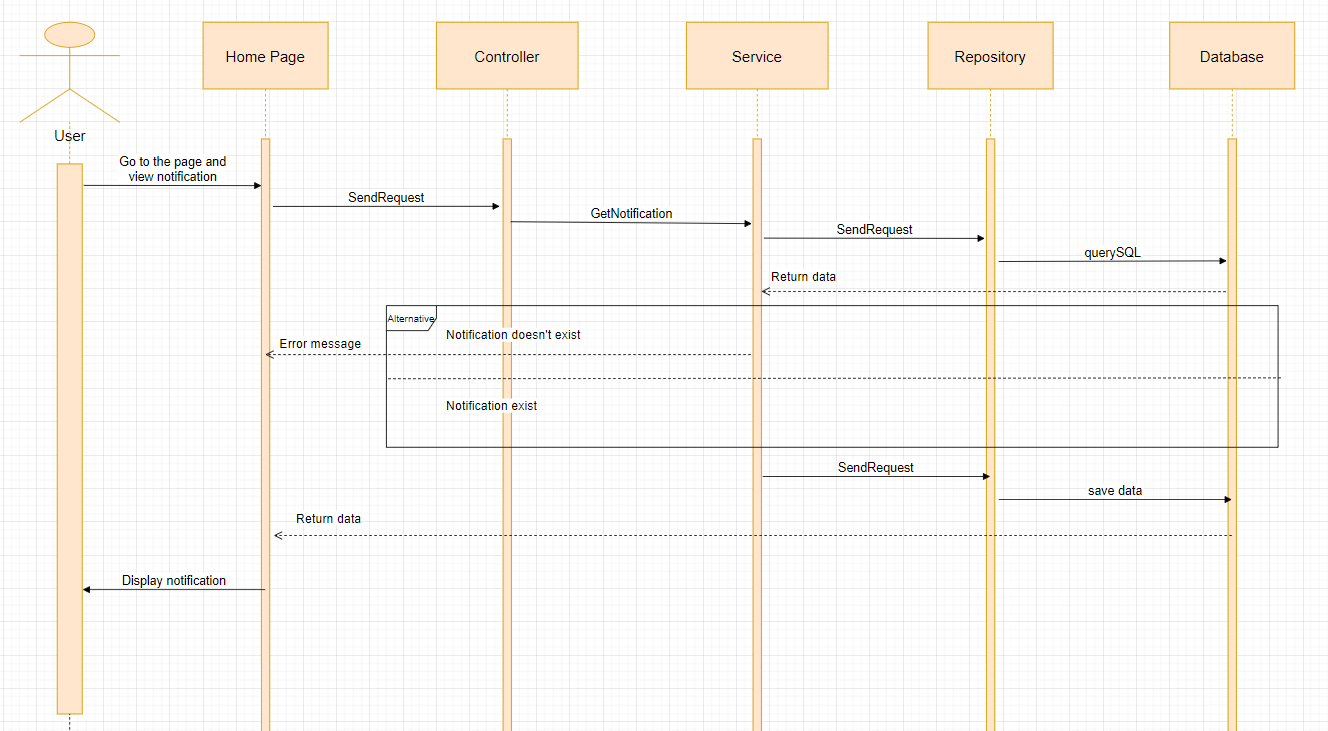
****

**4.4.1.12: Notification component**

**Screen Design:**

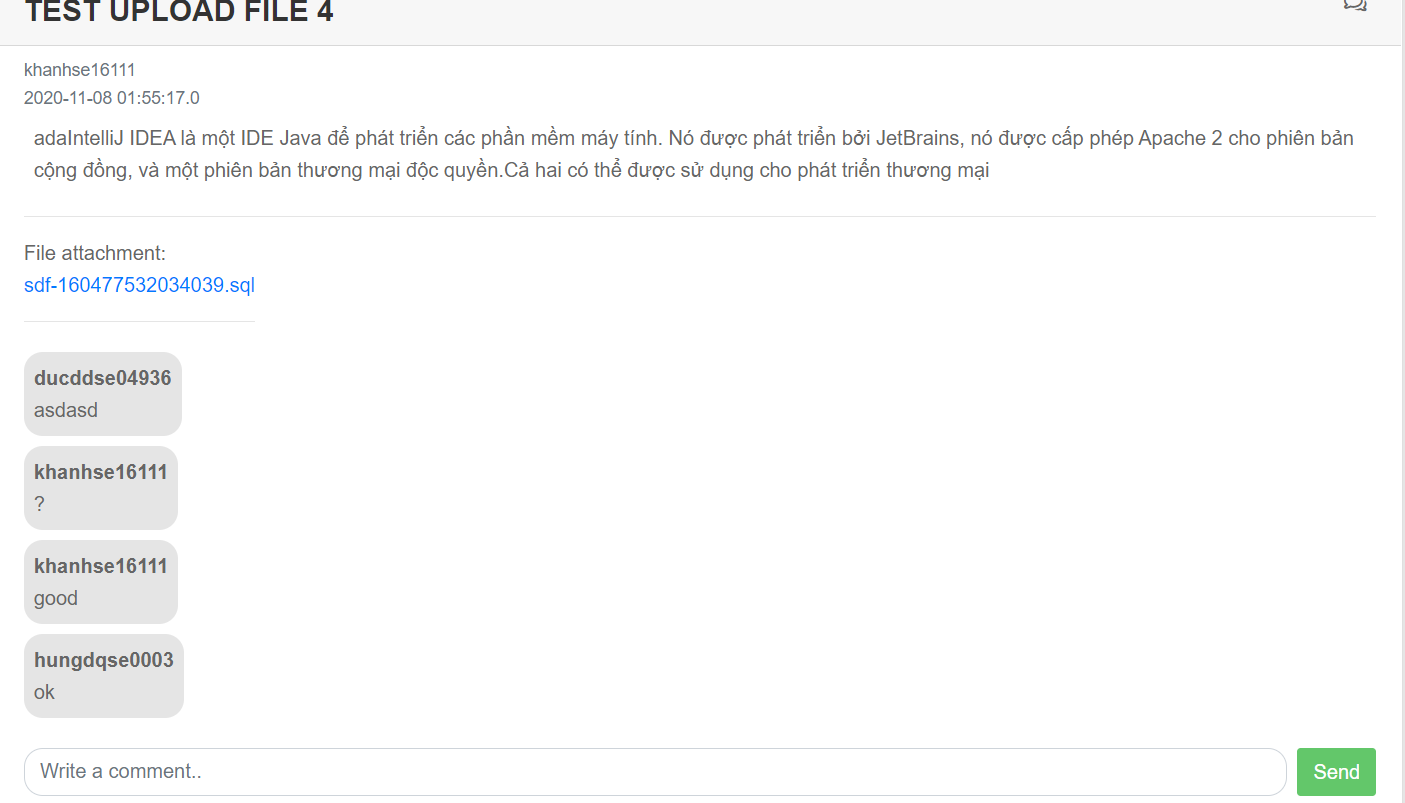
****

**Sequence Diagram:**

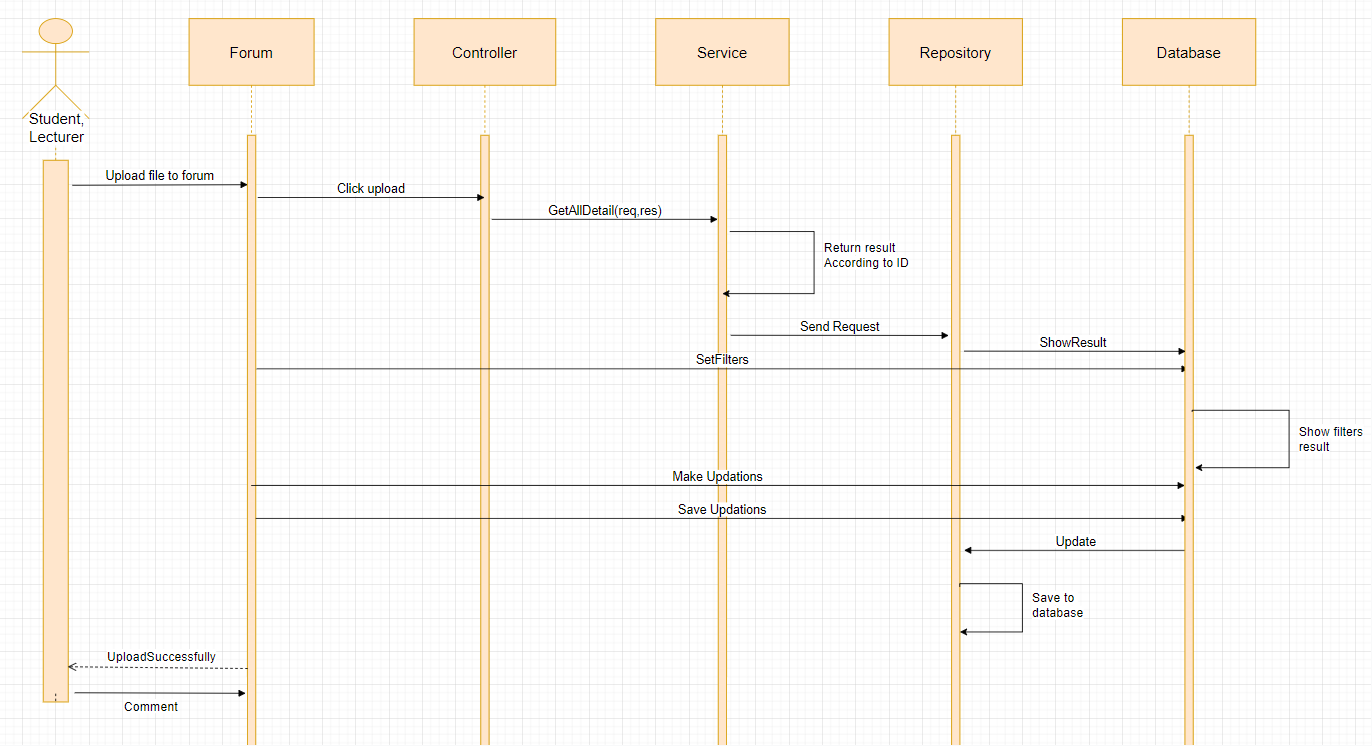
****

**4.4.1.13: Comment**

**Screen Design:**

****

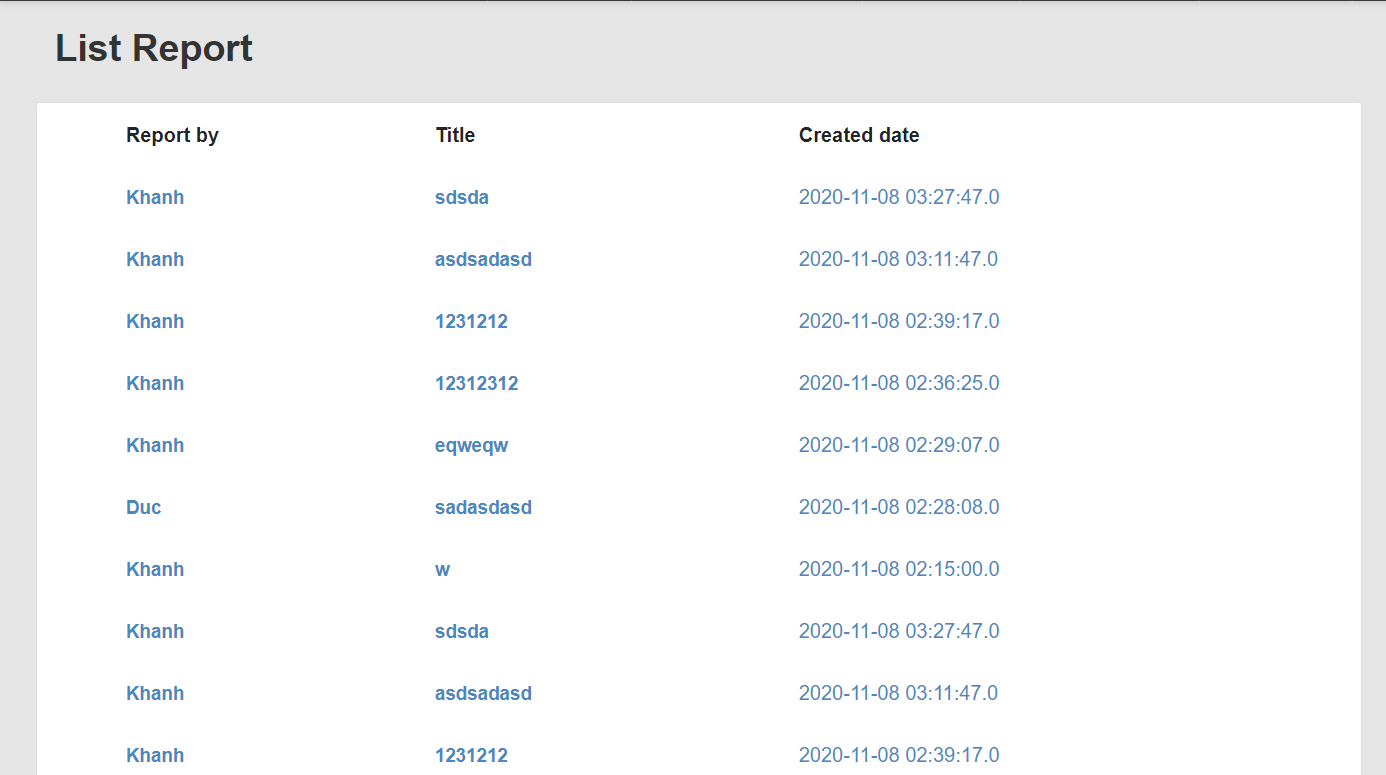
**Sequence Diagram:**

****

**4.4.1.14: Daily report to member to leader**

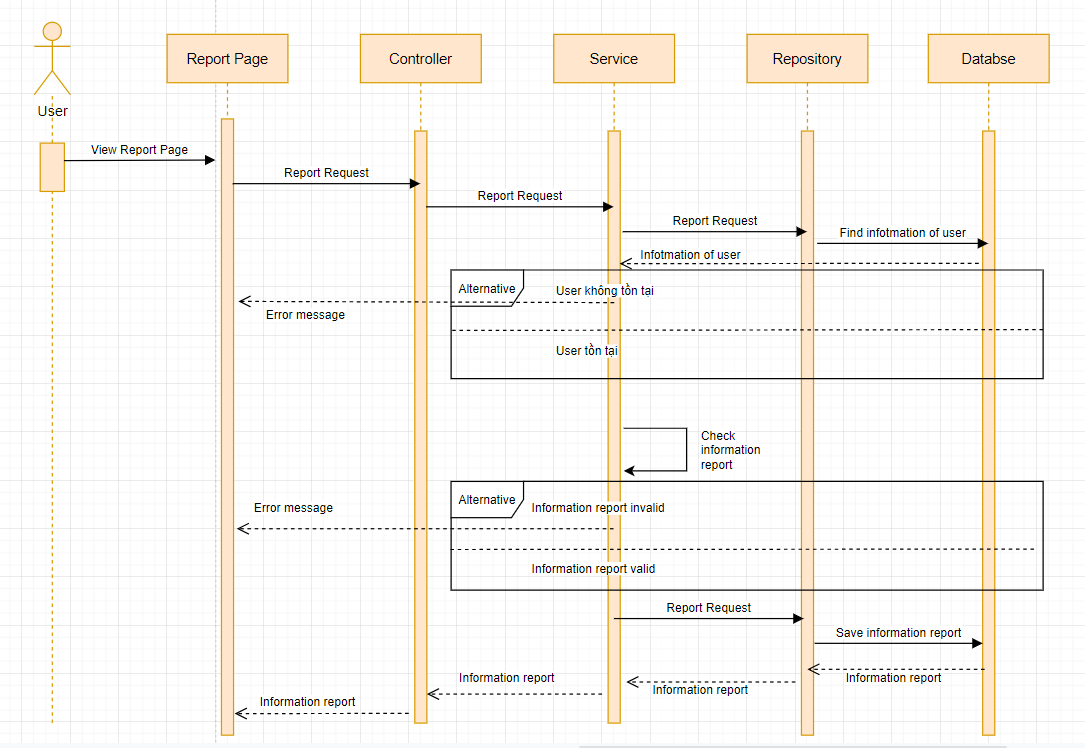
**Screen Design:**

****

****

****

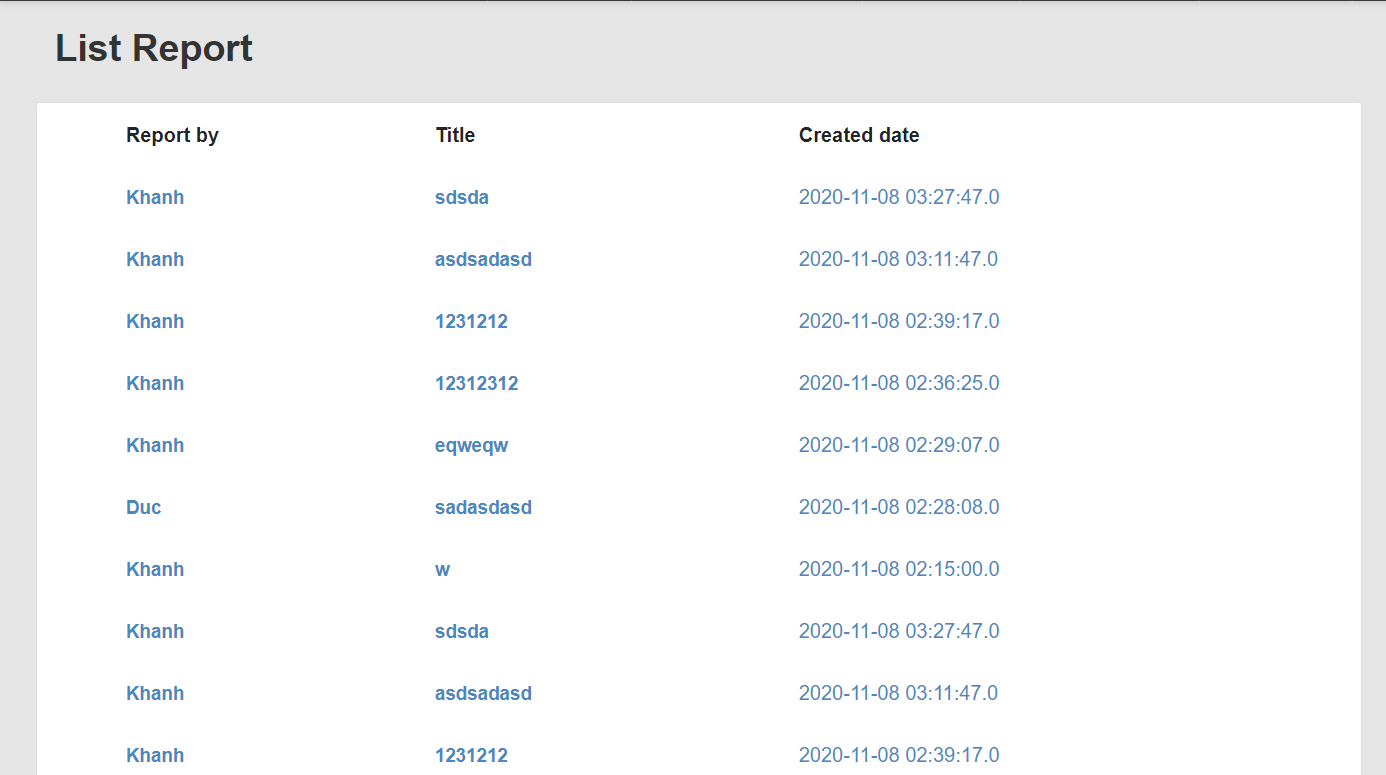
**Sequence Diagram:**

****

**4.4.1.15: Weekly report to leader to lecturer**

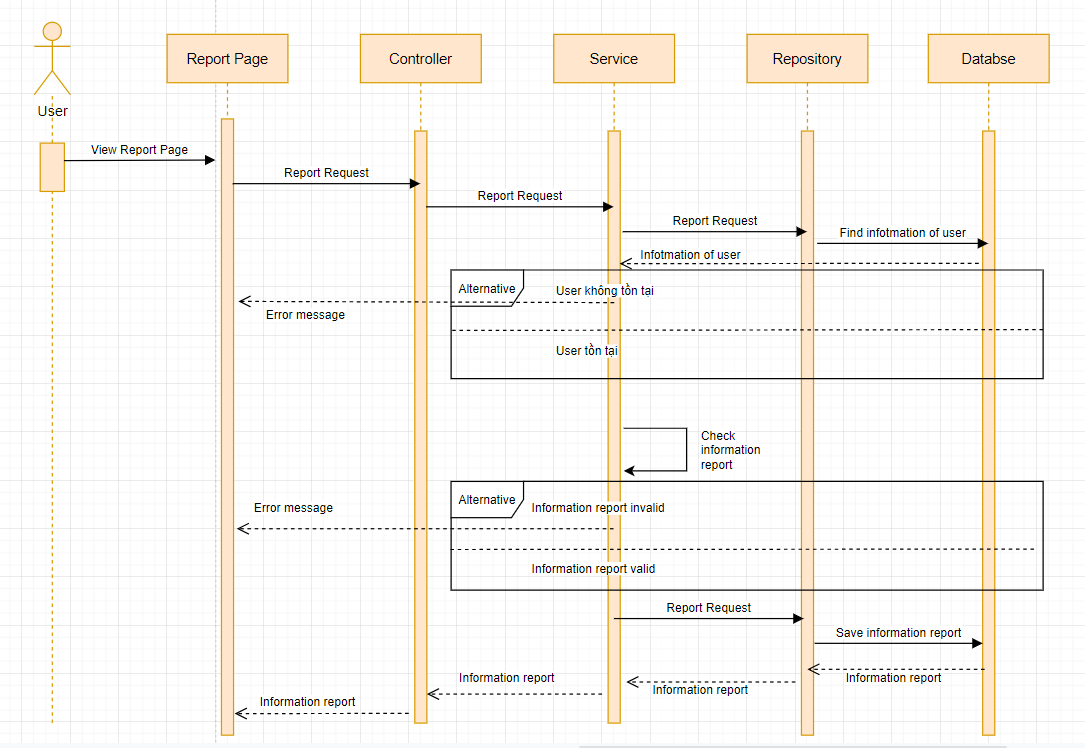
**Screen Design:**

****

****

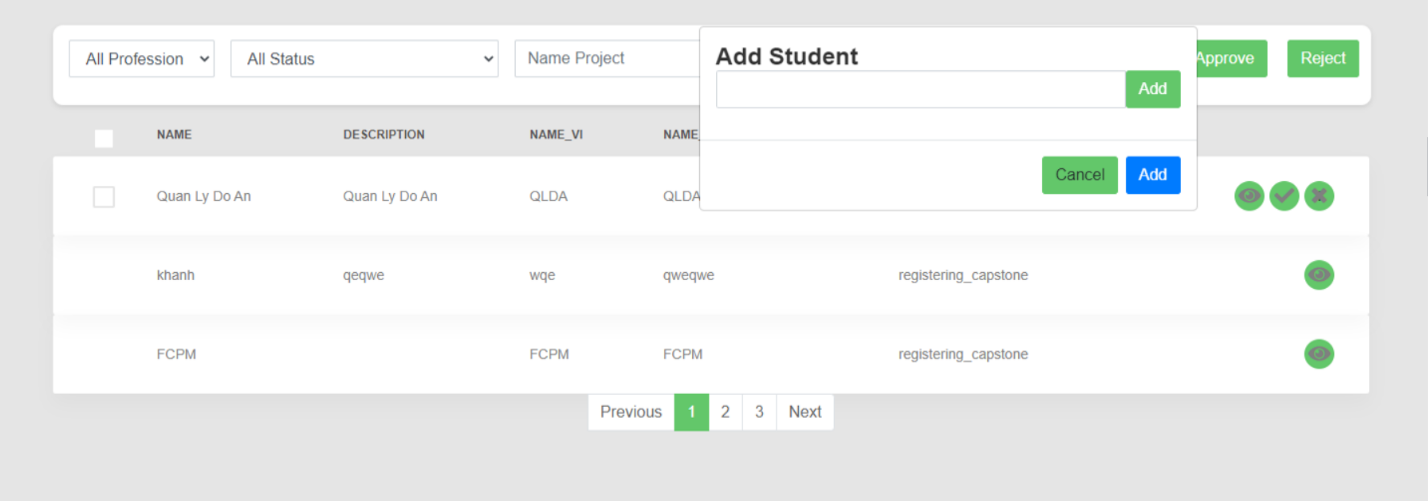
****

**Sequence Diagram:**

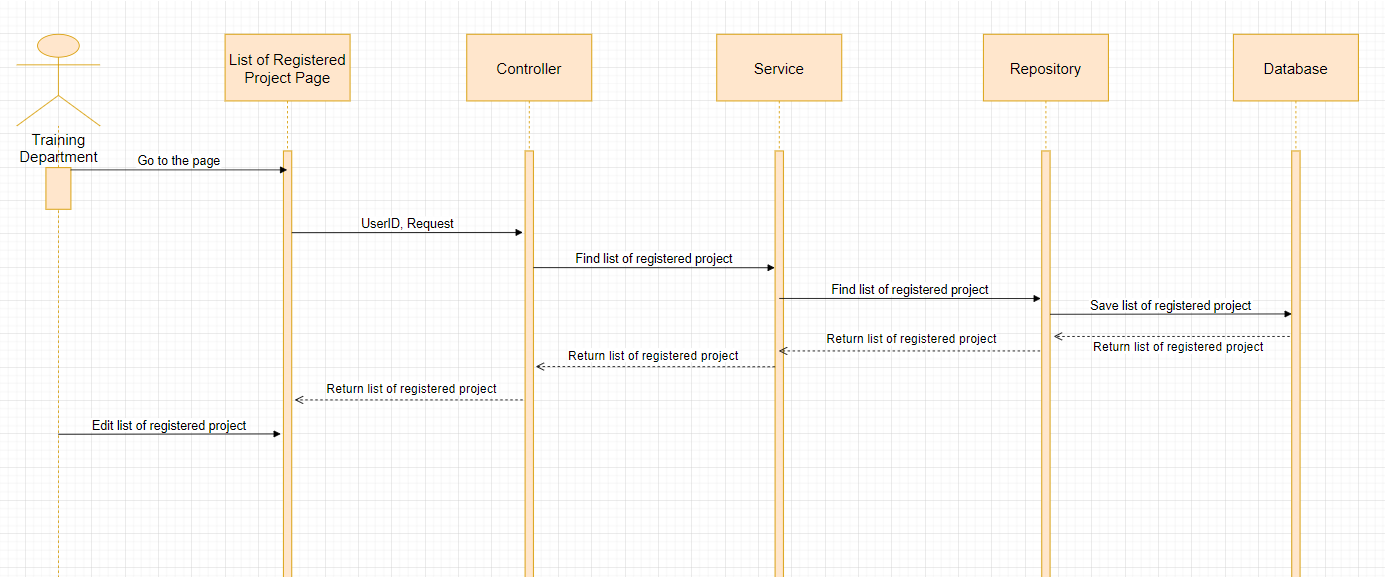
****

**4.4.1.16: View and edit the list of registered project**

**Screen Design:**

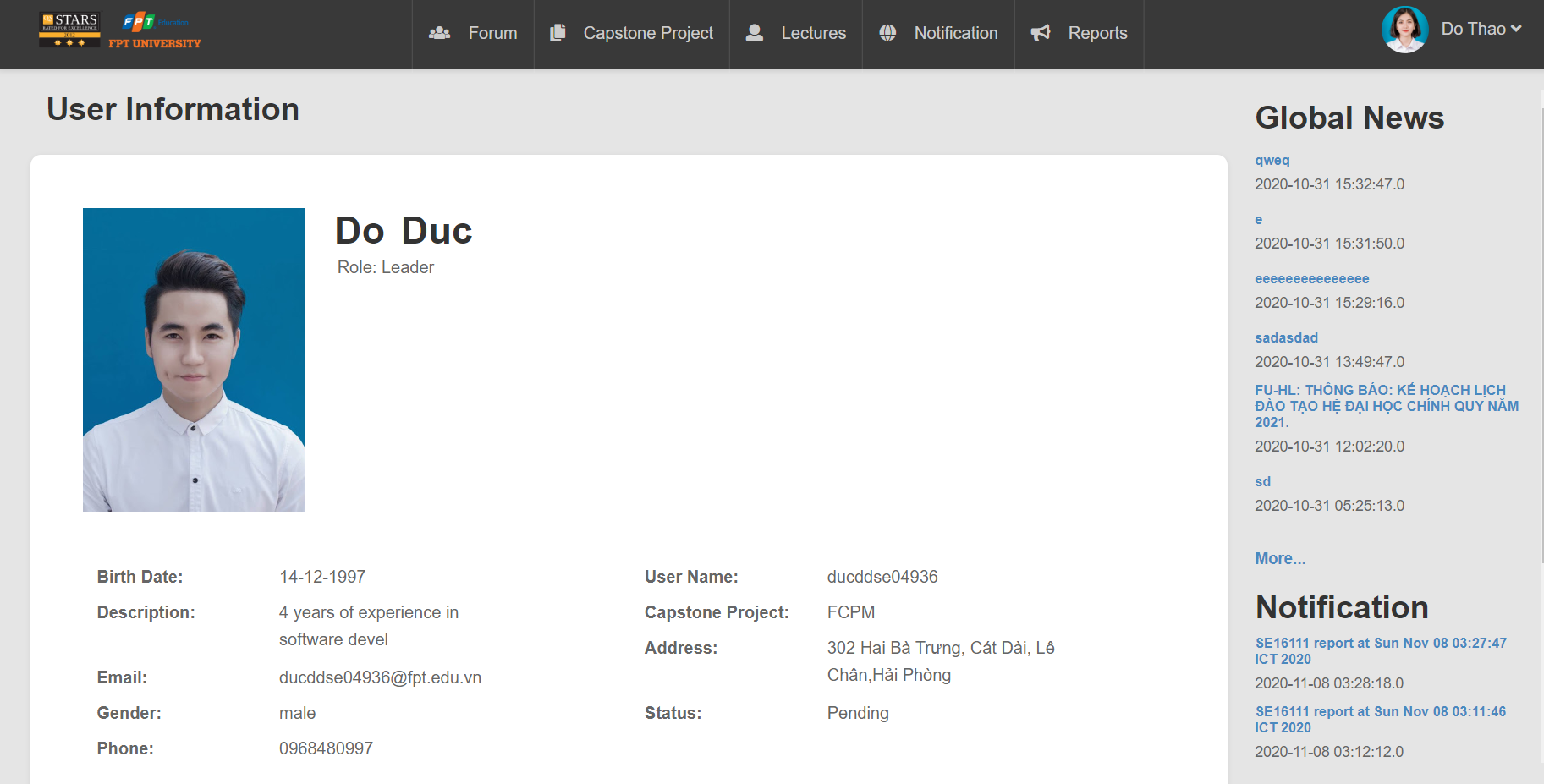
****

**Sequence Diagram:**

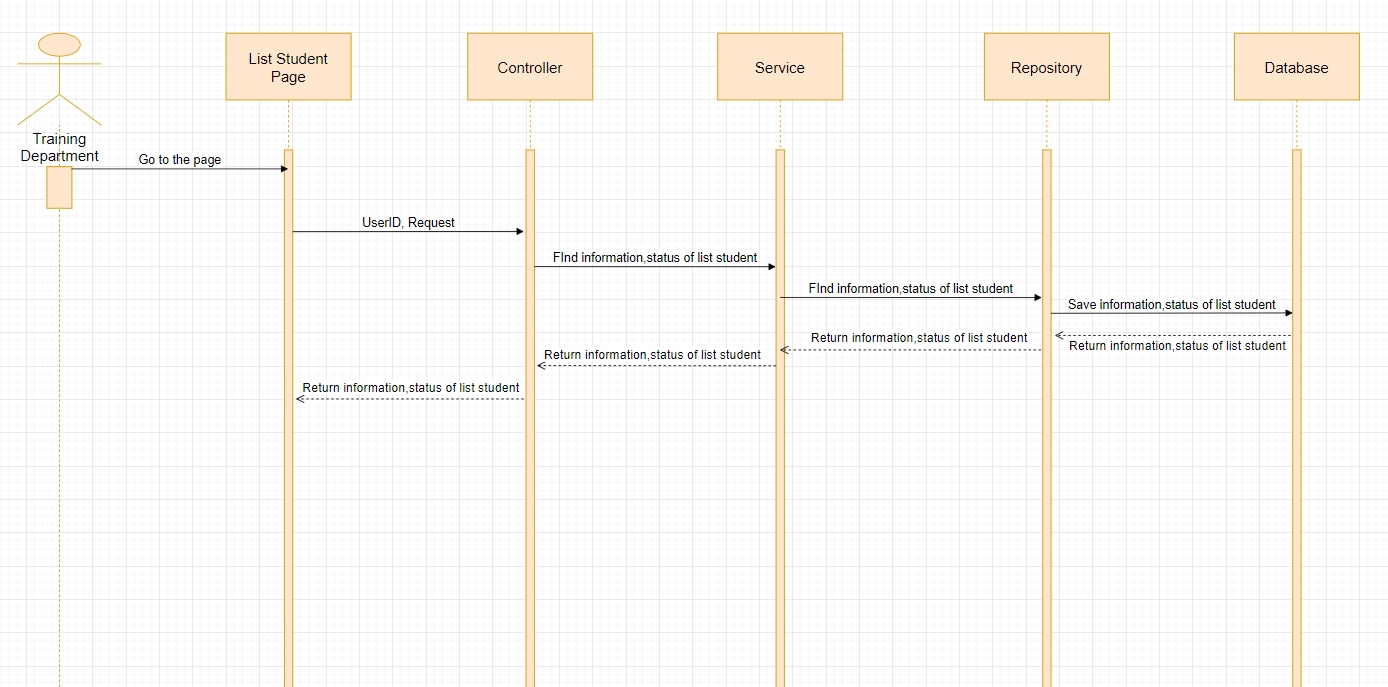
****

**4.4.1.17: View information, status of list student**

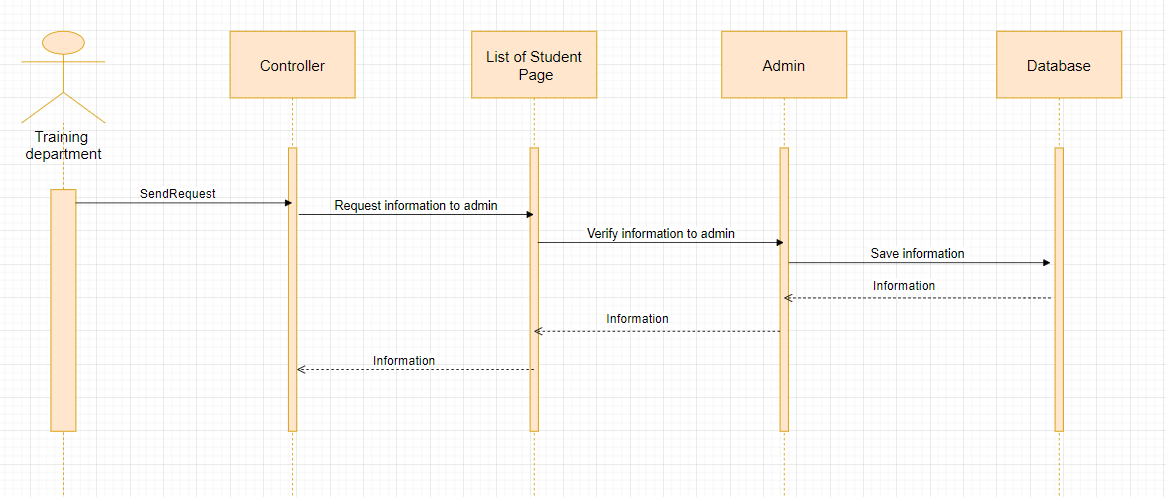
**Screen Design:**

****

**Sequence Diagram:**

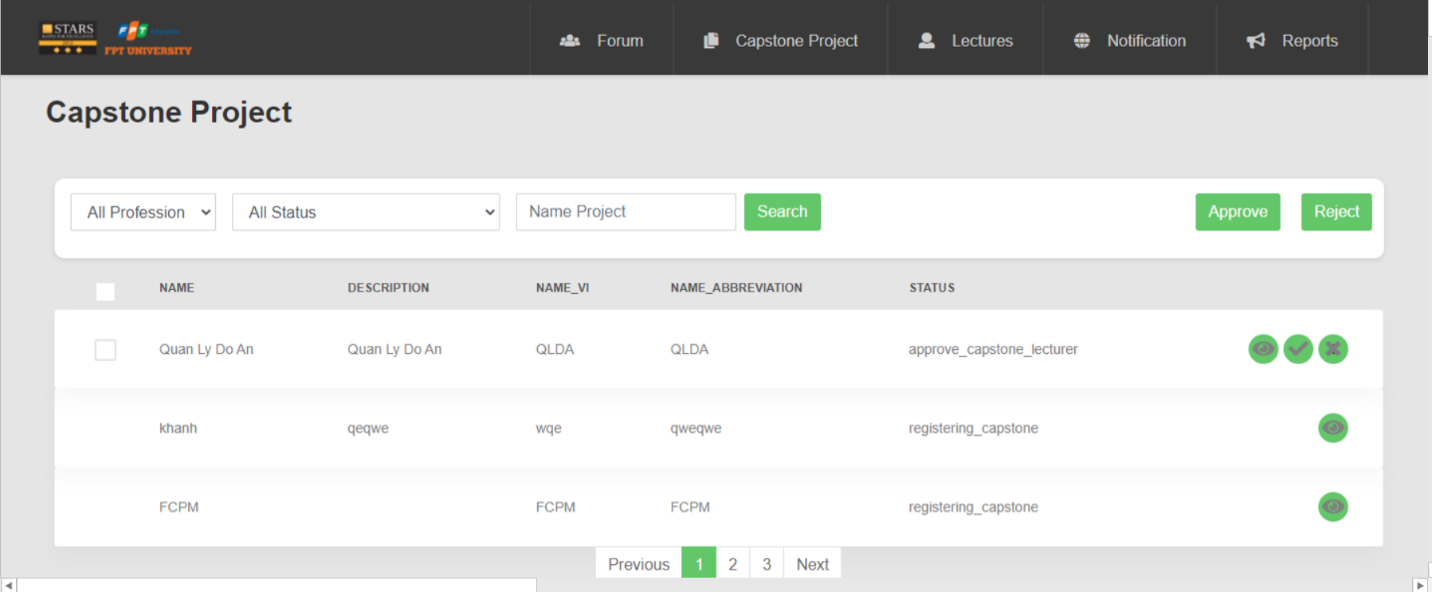
****

**4.4.1.18: Request information to admin**

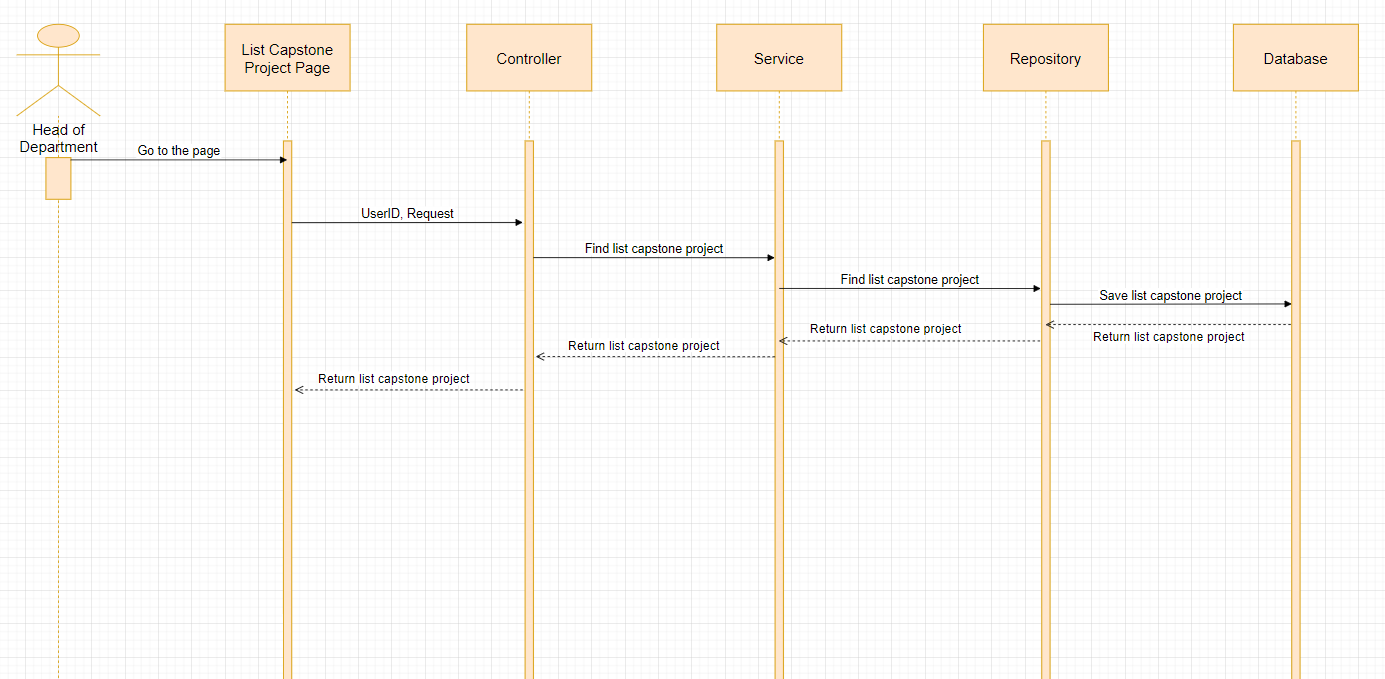
****

**4.4.1.19: View list capstone project**

**Screen Design:**

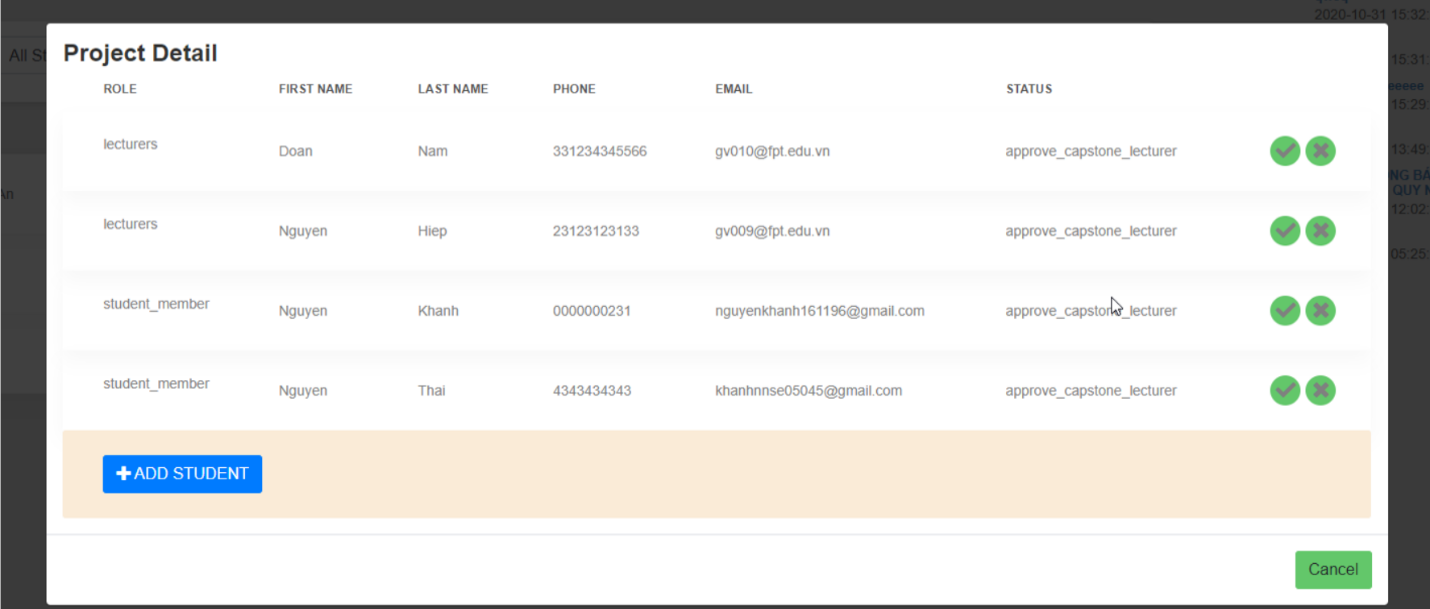
****

**Sequence Diagram:**

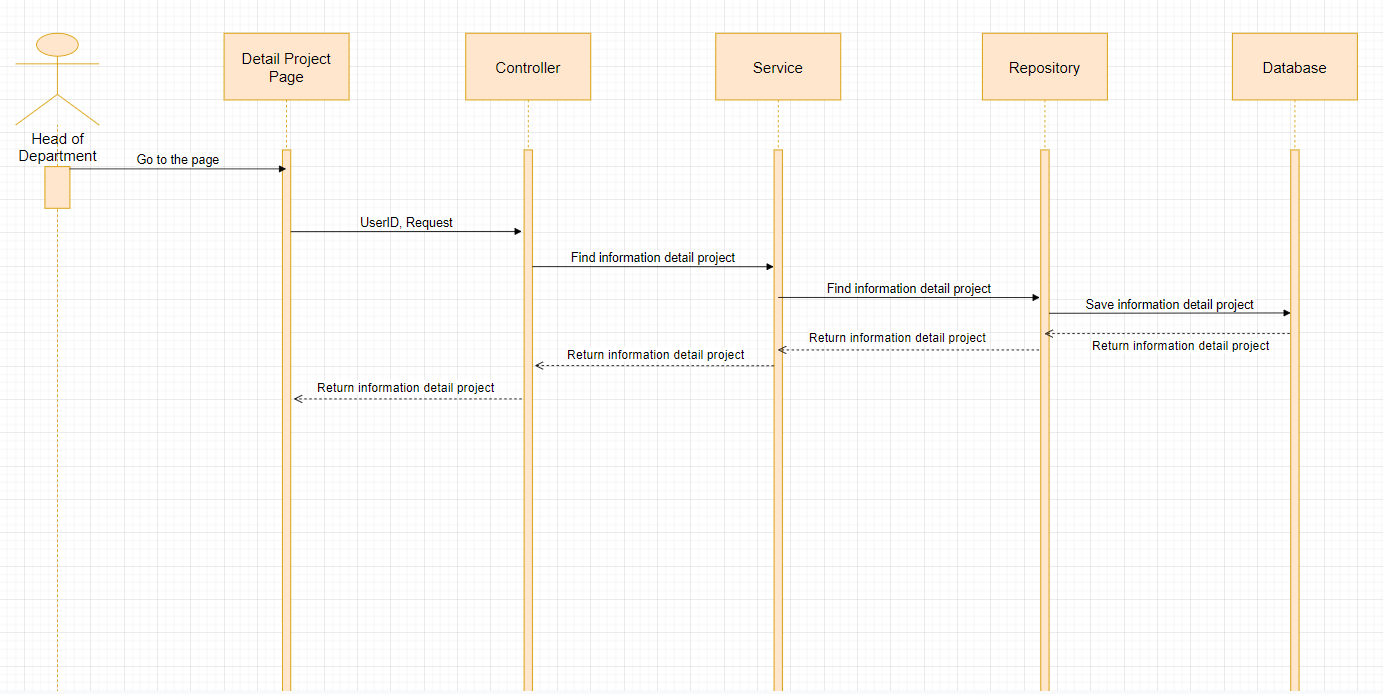
****

**4.4.1.20: View detail project**

**Screen Design:**

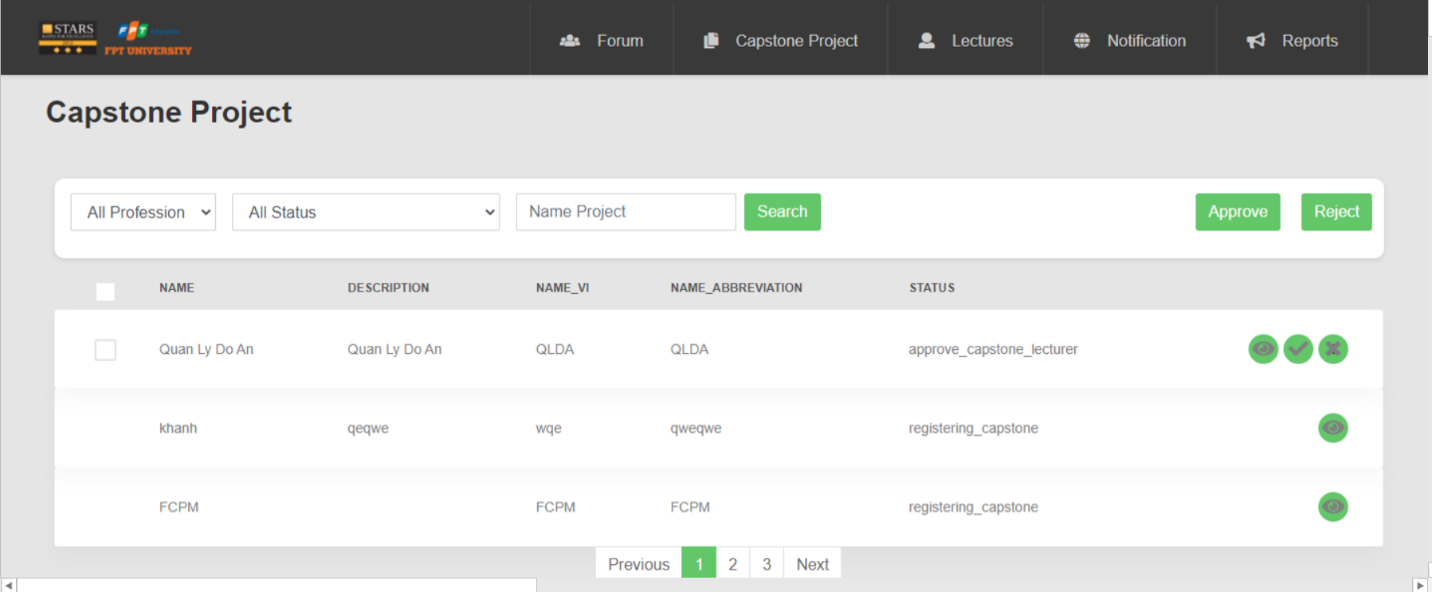
****

**Sequence Diagram:**

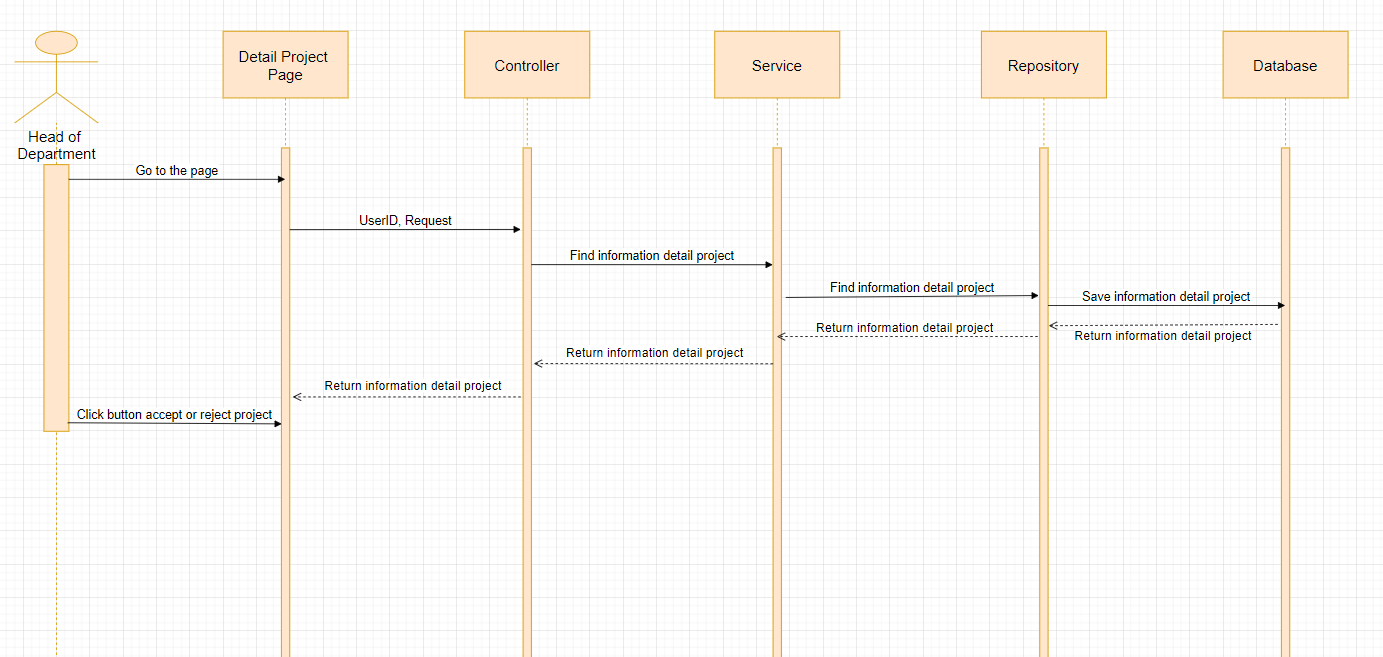
****

**4.4.1.21: Accept or reject project**

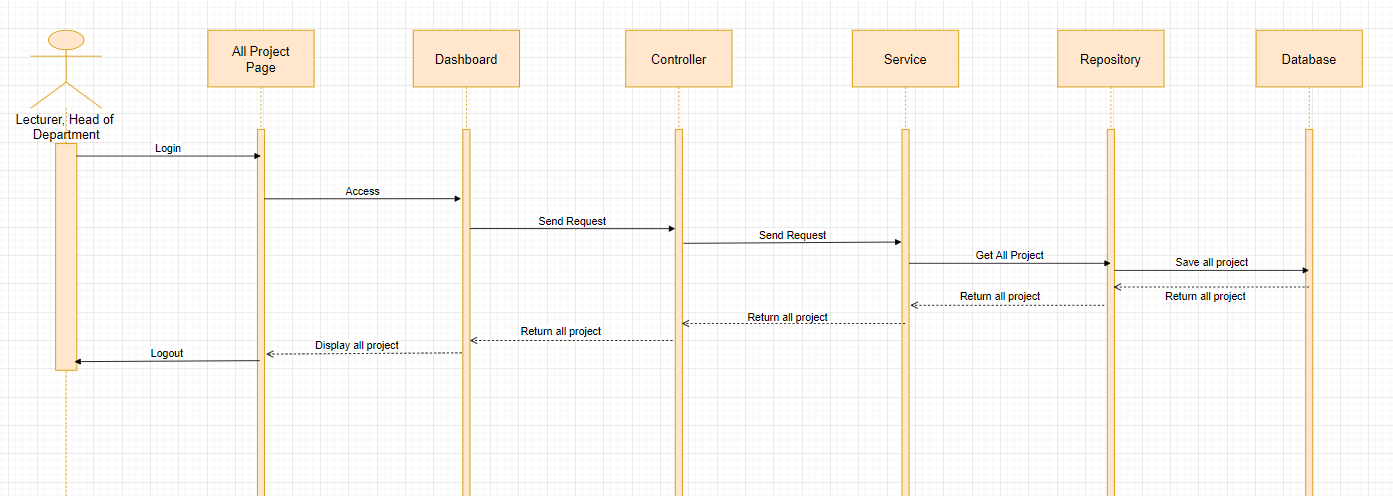
**Screen Design:**

****

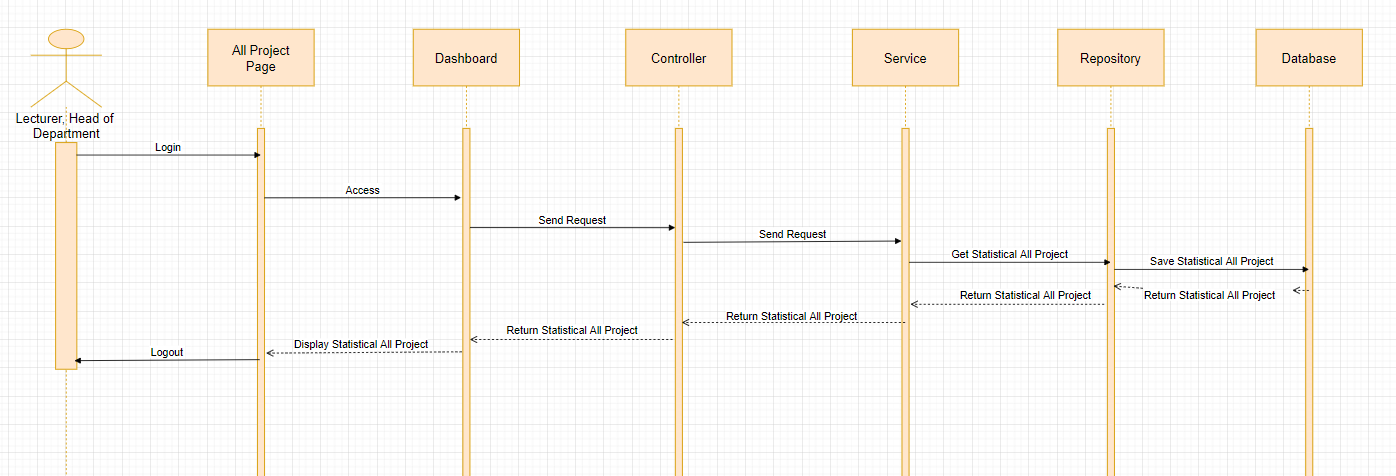
**Sequence Diagram:**

****

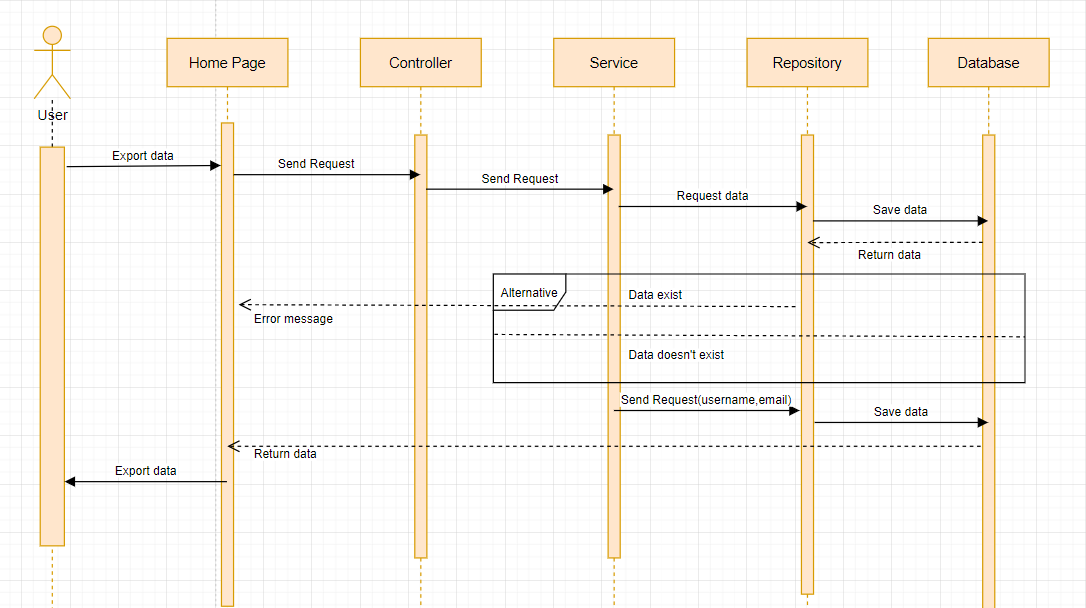
**4.4.1.22: Dashboard all project**

****

**4.4.1.22: Dashboard statistical all project**

****

**4.4.1.24: Export file statistical to project**

****