Software Requirements Specification

Student Management

Version 1.0 approved

Trinh Ba Binh - He130864

Group 4 – Se1324

4/10/2019

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 2

1.1 Purpose 2

1.2 Document Conventions 2

1.3 Project Scope 2

1.4 References 2

2. Overall Description 2

2.1 Product Perspective 2

2.2 User Classes and Characteristics 2

2.3 Operating Environment 2

2.4 Design and Implementation Constraints 2

2.5 Assumptions and Dependencies 2

3. System Features 2

3.1 System Feature 1 2

3.2 System Feature 2 (and so on) 2

4. Data Requirements 2

4.1 Logical Data Model 2

4.2 Data Dictionary 2

4.3 Reports 2

4.4 Data Acquisition, Integrity, Retention, and Disposal 2

5. External Interface Requirements 2

5.1 User Interfaces 2

5.2 Software Interfaces 2

5.3 Hardware Interfaces 2

5.4 Communications Interfaces 2

6. Quality Attributes 2

6.1 Usability 2

6.2 Performance 2

6.3 Security 2

6.4 Safety 2

6.5 [Others as relevant] 2

7. Internationalization and Localization Requirements 2

8. Other Requirements 2

Appendix A: Glossary 2

Appendix B: Analysis Models 2

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

The student management system is a student records management system that includes personal and academic information. The student management system is designed to make managing student lists easier. Extensive information is available at your fingertips through this System. See student data, semester, class, course management, test schedule, score management and related issues made simple and easy. Customizable search capabilities to assist in the search for student information and working on student profiles. This can make the system easy to navigate and use to maximize the efficiency of time, money, effort and other resources

* Planned approach towards working
* Feasibility
* No redundancy
* Accuracy
* Immediate retrieval of information
* Immediate storage of information
* Easy to operate

## Document Conventions

## Project Scope

The scope of the project includes the following:

* Any university, high school or organization can use this system.
* All work related to admission and testing for students can be done by this system.
* Providing electronic workplaces.
* Maintenance of applications after deployment into production.
* The admin module can be reused for projects that also have many users with different rights. Hence it is reusable

## References

* [https://www.academia.edu](https://www.academia.edu/)
* [https://www.slideshare.net](https://www.slideshare.net/)
* [https://fr.scribd.com](https://fr.scribd.com/)
* [https://sourceforge.net](https://sourceforge.net/)

# Overall Description

## Product Perspective

Student management software is a closed system. It does not depend on any other software. The front end is designed by JAVA, HTML, CSS, JAVASCRIPT… so simple, beautiful, convenient for users. The interfaces will involve check boxes, combo boxes, text boxes, and radio buttons. The combo boxes and the radio buttons will be used to prevent users from entering wrong type of information. They will also enable fast data entry. Text boxes will be controlled for avoiding invalid and inconsistent data. The back end uses MY SQL to create the database. Data will be included for storage, and retrieved for use as needed. With the modify function, users can directly change the data as required.

## User Classes and Characteristics

Administrator – The administrator will hold full access to view as well control anywhere in the software and the information.

User- The other user like staff or faculty can view only their individual report their own performance based on that.

The user profiles identified to have interaction with Student Management System that anyone can register and login into system and us the required resources. The students can easily fill up the registration form online and submit it. And the admin will check the details that is the student is eligible as per the admission criteria. After the student will successfully registered he can use college/school system environments as per their limits decided by admin.

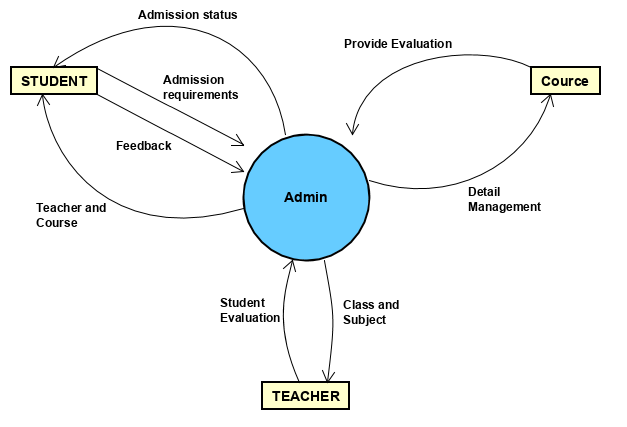
## Operating Environment

Student management software is a closed system. It does not depend on any other software.. The software will be handed over to the registrant after fulfilling the contract. Users have all rights to use depending on the purpose

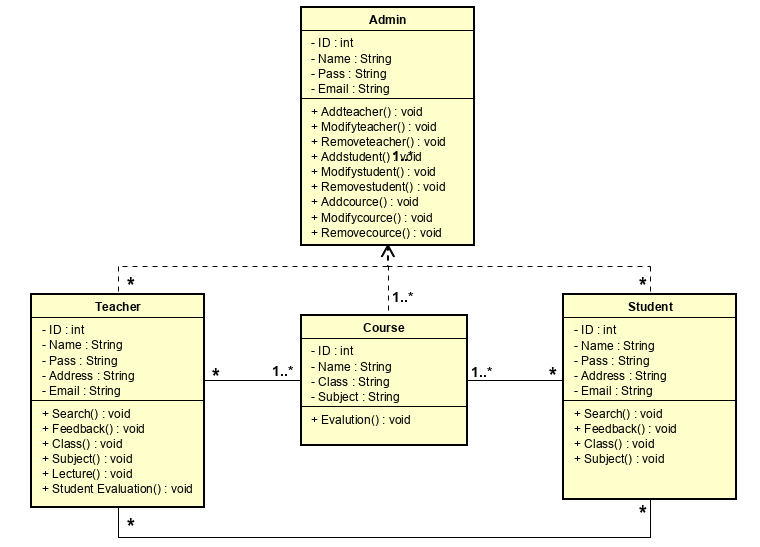
## Assumptions and Dependencies

# System Features

*Figure 0 : Use case diagram*

**

*Figure1: Context diagram*

*Figure 2: Class diagram*

## System Feature Manage Account

### Description

Each user will have an account. For the manager there will be an admin account. Administrators will retain full access to view and manage anywhere in the software and information. Left students and teachers will be provided with a user account. Other users like employees or trainers can only view their personal reports and analyze their performance based on them.

### Stimulus/Response Sequences

After the user create new account, depending on the type of user, the system will provide different functions

### Functional Requirements Login

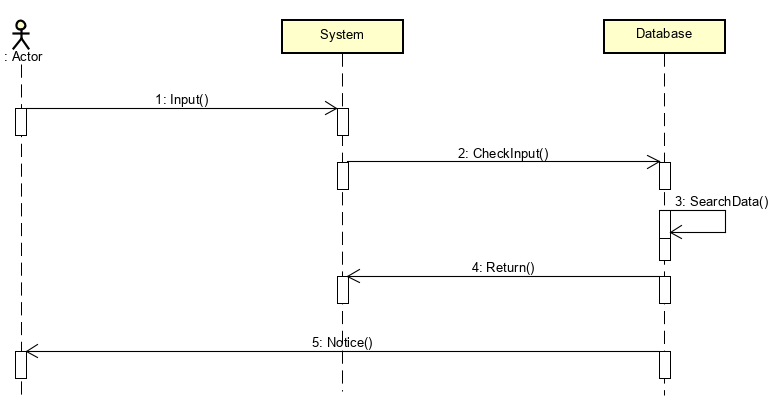
### 

### *Figure 3: Login interface*

### 

### *Figure 4: Login activity diagram*

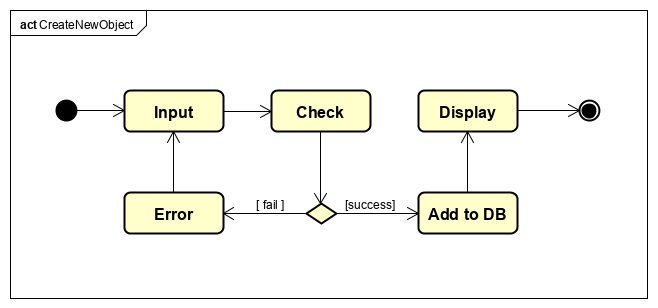
|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | Login | | |
| Created By: | Binhtbhe130864 | Date Created: | 4/10/2019 |
| Primary Actor: | Admin | Secondary Actors: | User |
| Trigger: | None | | |
| Description: | Show the users how to login the system | | |
| Preconditions: | The user must have an account | | |
| Postconditions: | None | | |
| Normal Flow: | 1. The system requires the user to enter username and password 2. The User will fill in and click Sing In 3. The system check data input and allows user to access the system. | | |
| Alternative Flows: | None | | |
| Exceptions: | An error message will appear if username or password you entered is invalid | | |
| Priority: | None | | |
| Frequency of Use: | Unlimited. whenever users access the system | | |
| Business Rules: | Only used internally by the organization | | |
| Other Information: | None | | |
| Assumptions: | None | | |

*Figure 5: Login table detail*

*Figure 6: Login sequence diagram*

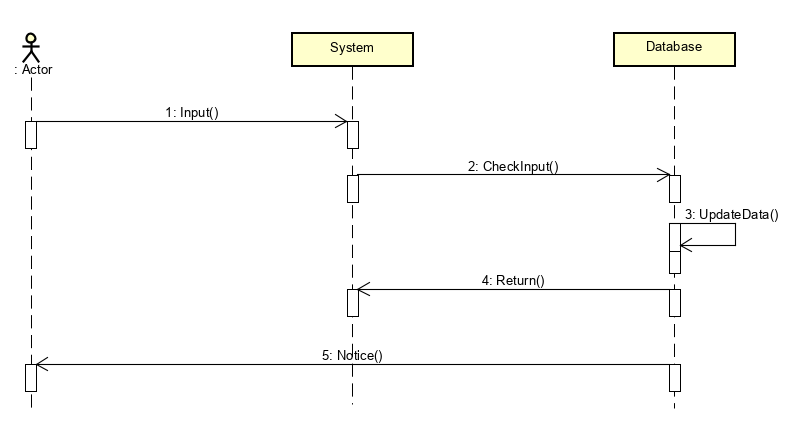
### 3.1.4 Functional Requirements Create new account

*Figure 7: Create new account interface*

**

*Figure 8: Create new account activity diagram*

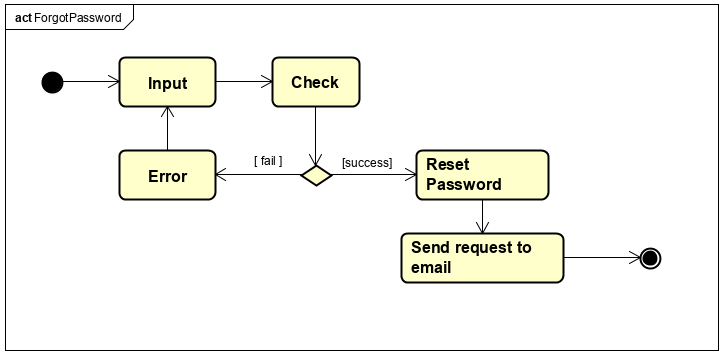
|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | Create new Account | | |
| Created By: | Binhtbhe130864 | Date Created: | 4/10/2019 |
| Primary Actor: | Admin | Secondary Actors: | User |
| Trigger: | None | | |
| Description: | Show the users how to create a new account | | |
| Preconditions: | The user must be member of the organization and create new account by organization's email | | |
| Postconditions: | After create new account, this account will be updated and insert into DB  And user can access the system | | |
| Normal Flow: | 1. The system requires the user to enter information 2. The User will enter name , password and address and click Sign up 3. The system check data input and allows user to access the system. | | |
| Alternative Flows: | None | | |
| Exceptions: | An error message will appear if data you entered is invalid | | |
| Priority: | None | | |
| Frequency of Use: | Unlimited . whenever users access the system | | |
| Business Rules: | Only used internally by the organization | | |
| Other Information: | None | | |
| Assumptions: | None | | |

*Figure 9: Create new account table detail*

*Figure 10: Create new account sequence diagram*

### Functional Requirements Forgot password

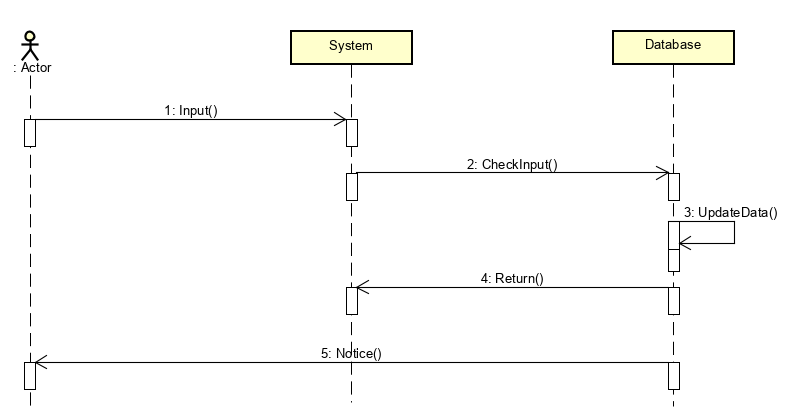
*Figure 11: Forgot password interface*



*Figure 12: Forgot password activity diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | Forgot password | | |
| Created By: | Binhtbhe130864 | Date Created: | 4/10/2019 |
| Primary Actor: | Admin | Secondary Actors: | User |
| Trigger: | None | | |
| Description: | Show the user how to create new password when forgot | | |
| Preconditions: | The user must already have the valid account and email | | |
| Postconditions: | None | | |
| Normal Flow: | 1. The user click “Forgot the password” in login interface 2. The system will require you enter username and email 3. You fill in and click “send request” 4. The system will send to you an email to create new password | | |
| Alternative Flows: | None | | |
| Exceptions: | If your email is not valid, show error message | | |
| Priority: | None | | |
| Frequency of Use: | Unlimited. whenever users access the system | | |
| Business Rules: | Only used internally by the organization | | |
| Other Information: | None | | |
| Assumptions: | None | | |

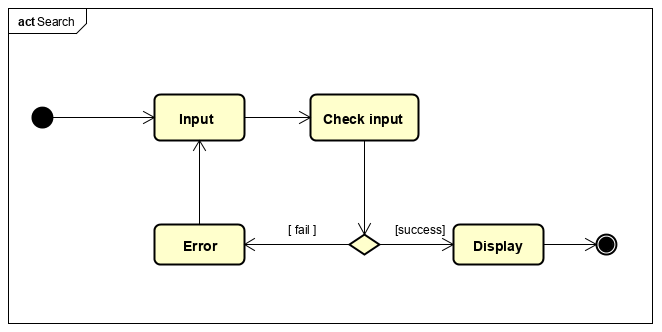
*Figure 13: Forgot password table detail*

**

*Figure 14: Forgot password sequence diagram*

## System Feature Search information

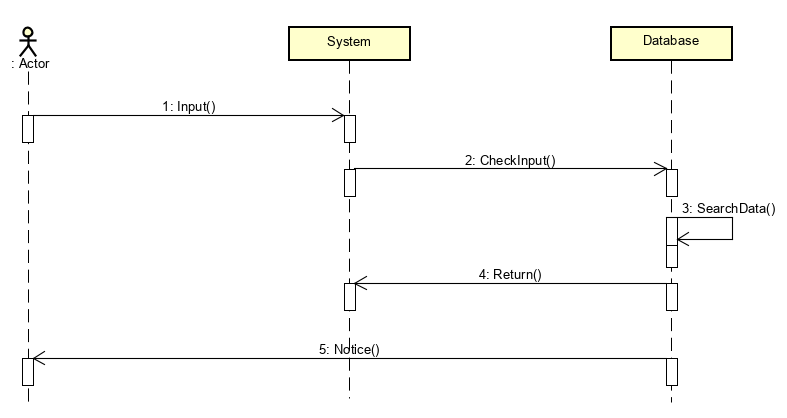
*Figure 15: Search interface*



*Figure 16: Search activity diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | Search information | | |
| Created By: | Binhtbhe130864 | Date Created: | 4/10/2019 |
| Primary Actor: | Admin | Secondary Actors: | User |
| Trigger: | None | | |
| Description: | Show the user how to search any information about student, teacher course | | |
| Preconditions: | The user must already have the valid account | | |
| Postconditions: | None | | |
| Normal Flow: | Enter information user want to search  1. The system will check and return result if it exists or return null if not found | | |
| Alternative Flows: | None | | |
| Exceptions: | If information user search not found, the system will return nothing | | |
| Priority: | None | | |
| Frequency of Use: | Unlimited . whenever users access the system | | |
| Business Rules: | Only used internally by the organization | | |
| Other Information: | None | | |
| Assumptions: | None | | |

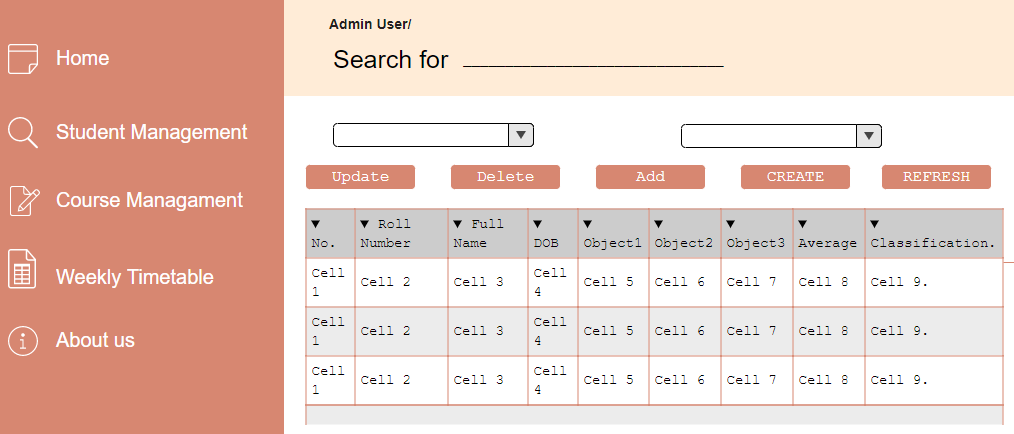
*Figure 17: Search table detail*

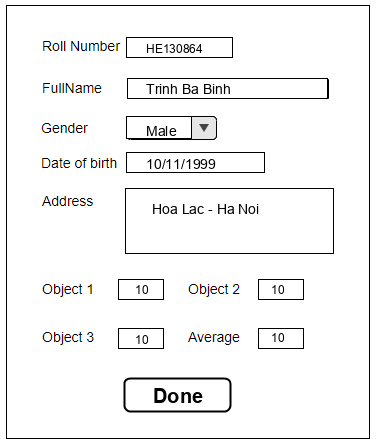
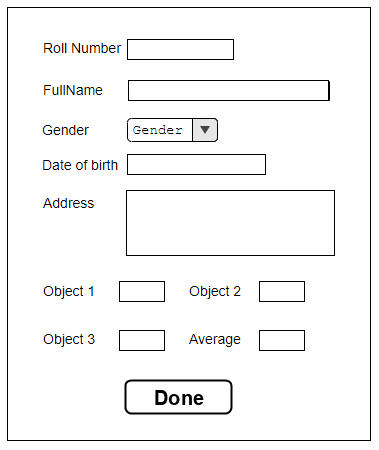
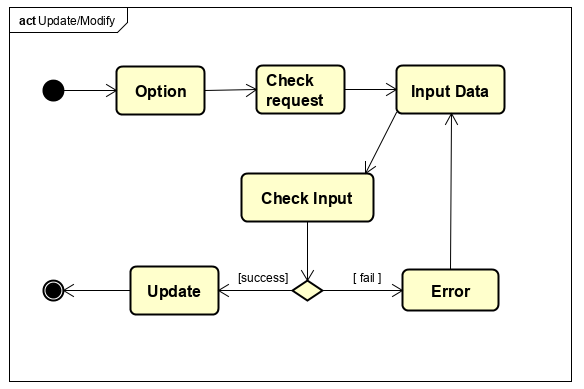
**

*Figure 18: Search sequence diagram*

## System Feature Update/Modify information

3**.3.1 Functional Requirements Update/add information**

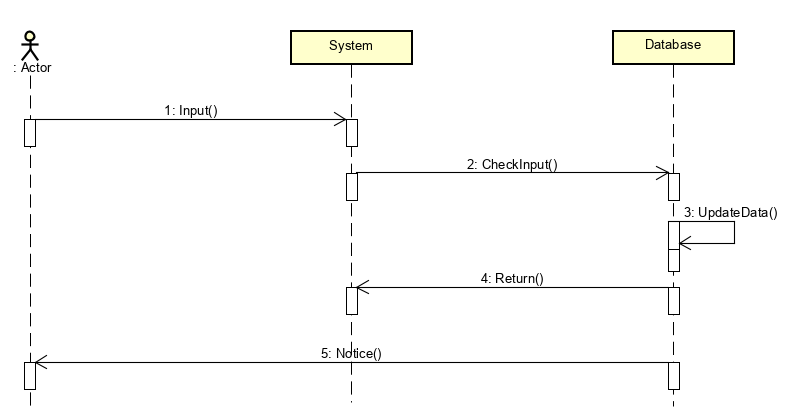


*Figure 19: Update/modify interface*

*Figure 20: Update/modify activity diagram*

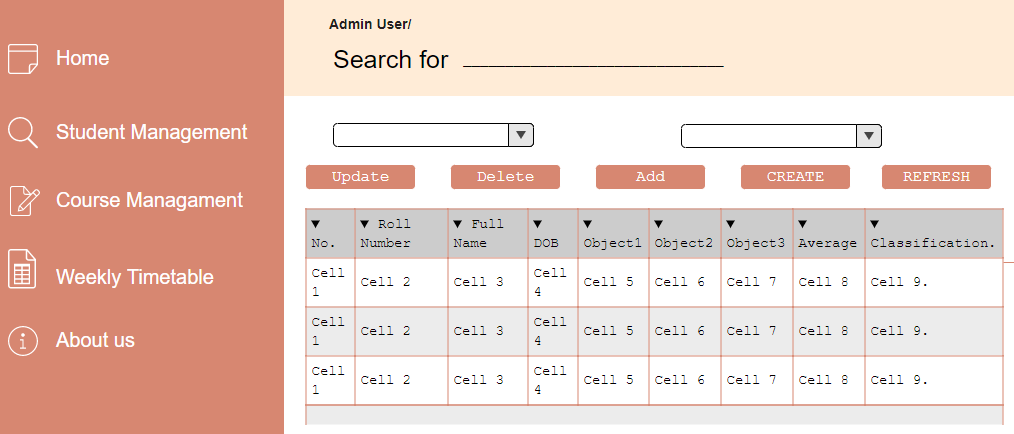
|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | Update/modify | | |
| Created By: | Binhtbhe130864 | Date Created: | 4/10/2019 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Trigger: | None | | |
| Description: | Show the user how to update/add information | | |
| Preconditions: | The user must already have the valid account | | |
| Postconditions: | None | | |
| Normal Flow: | Click button “Update” or “Add” or ”Create” and the system will show you a dialog to input data  1. The user must fill in all information about object to update or modify 2. The user click button “Done” to save information | | |
| Alternative Flows: | None | | |
| Exceptions: | If information user search not found, the system will return nothing | | |
| Priority: | None | | |
| Frequency of Use: | Unlimited . whenever users access the system | | |
| Business Rules: | Only used internally by the organization | | |
| Other Information: | None | | |
| Assumptions: | None | | |

*Figure 21: Update/modify table detail*

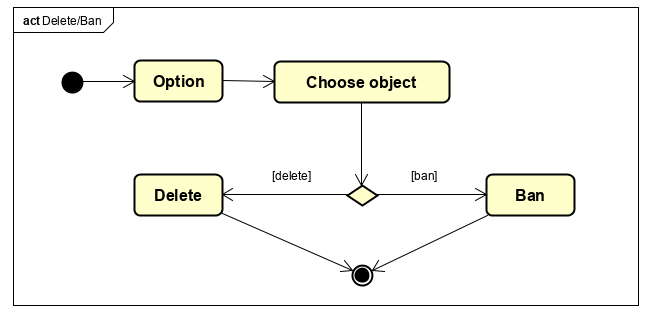
**

*Figure 22: Update/modify sequence diagram*

**3.3.2 Function Requirements Delete/ban information**



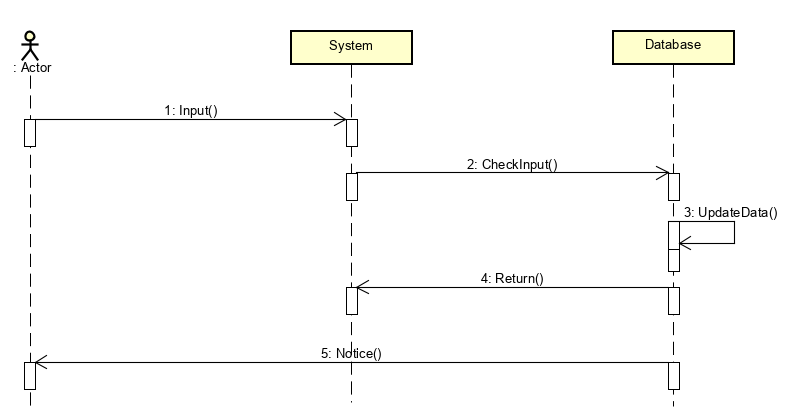
*Figure 23: Delete interface*

**

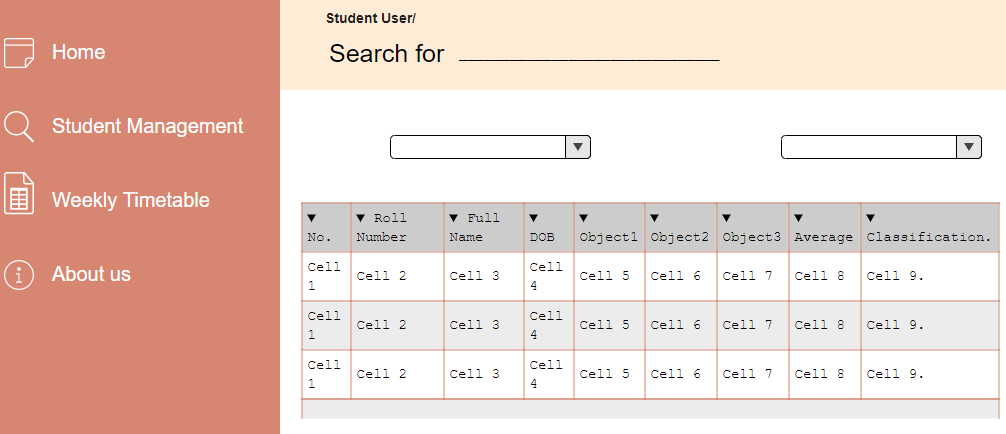
*Figure 24: Delete/ban activity diagram*

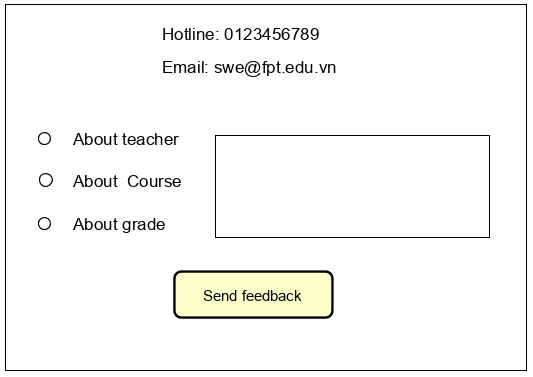
|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | Delete/ban information | | |
| Created By: | Binhtbhe130864 | Date Created: | 4/10/2019 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Trigger: | None | | |
| Description: | Show the user how to delete/ban object or information | | |
| Preconditions: | The user must already have the valid account | | |
| Postconditions: | None | | |
| Normal Flow: | The user choose an object or information and click button “Delete”  1. The system will delete in DB and object not appear on interface | | |
| Alternative Flows: | None | | |
| Exceptions: | If you do not choose object you can do the next step | | |
| Priority: | None | | |
| Frequency of Use: | Unlimited . whenever users access the system | | |
| Business Rules: | Only used internally by the organization | | |
| Other Information: | None | | |
| Assumptions: | None | | |

*Figure 25: Delete/ban table detail*

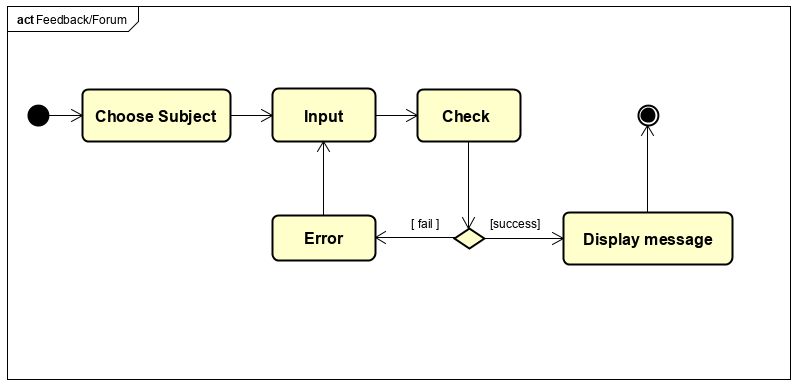
**

*Figure 26: Delete/ban sequence diagram*

**3.4 System Feature Feedback**



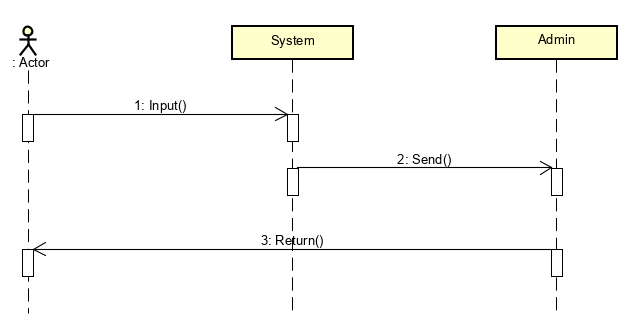
*Figure 27: Feedback interface*



*Figure 28: Feedback activity diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | Feedback | | |
| Created By: | Binhtbhe130864 | Date Created: | 4/10/2019 |
| Primary Actor: | User | Secondary Actors: | None |
| Trigger: | None | | |
| Description: | Show the user how to feedback about problems in organization | | |
| Preconditions: | The user must already have the valid account | | |
| Postconditions: | None | | |
| Normal Flow: | The user choose “About Us” and the system show you a dialog  1. The user choose subject to feedback and enter in text field 2. Click button “Send feedback” to finish | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | None | | |
| Frequency of Use: | Unlimited . whenever users access the system | | |
| Business Rules: | Only used internally by the organization | | |
| Other Information: | None | | |
| Assumptions: | None | | |

*Figure 29: Feedback table detail*

****

*Figure 30: Feedback sequence diagram*

# Data Requirements

## Logical Data Model

## Reports

## Data Acquisition, Integrity, Retention, and Disposal

# External Interface Requirements

## User Interfaces

## Software Interfaces

## Hardware Interfaces

## Communications Interfaces

# Quality Attributes

## Usability

## Performance

## Security

## Safety

## [Others as relevant]

# Internationalization and Localization Requirements

# Other Requirements

Appendix A: Glossary

Appendix B: Analysis Models