

**Supportive Learning**

**--)o0o(--**

Royal Educational Academy

**Faculty :**  **Kieu Trong Khanh**  
**Batch No. : F31008T1**

|  |  |  |
| --- | --- | --- |
| **Students** | **Full Name** | **Roll No.** |
| 1 | Nguyen Manh Hung | Student505656 |
| 2 | Nguyen Cong Thuong | Student508426 |
| 3 | Vo Van Cuong | Student505501 |

**Group No : 03**

Table of contents:

[Review 1 2](#_Toc320635837)

[1. Introduction: 3](#_Toc320635838)

[2. Existing Scenario: 3](#_Toc320635839)

[3. Customer Requirement Specification: 3](#_Toc320635840)

[4. Functional Requirement Specification: 4](#_Toc320635841)

[5. System Requirements: 10](#_Toc320635842)

[6. Development Software: 10](#_Toc320635843)

[7. Technology: 10](#_Toc320635844)

[Review 2 12](#_Toc320635845)

[8. Use Case and Sequence Diagram: 13](#_Toc320635846)

[9. Entity Relationship Diagram (ERD): 69](#_Toc320635847)

[Review 3 84](#_Toc320635848)

[10. Database Relationship Diagram: 85](#_Toc320635849)

[11. Table Definition: 85](#_Toc320635850)

[12. Other constrains: 90](#_Toc320635851)

[13. Site map: 90](#_Toc320635852)

[14. GUI Design: 90](#_Toc320635853)

[15. Report: 90](#_Toc320635854)

[Task Sheet 91](#_Toc320635855)

[16. CRS document: 92](#_Toc320635856)

[17. Use Case and Sequence Diagrams: 92](#_Toc320635857)

[18. ERD – Entity relationship diagram: 92](#_Toc320635858)

[19. GUI -Graphical user interface: 92](#_Toc320635859)

[20. Report: 92](#_Toc320635860)

[21. Permission Security: 92](#_Toc320635861)

[22. Task sheet: 92](#_Toc320635862)

Review 1

1. Introduction:

Our client is Royal Educational Academy, which generally runs the courses for

B.Tech, B.Pharmacy, M.B.A and M.C.A. For the convenience of the students they had even started part time courses for the above mentioned courses. Every year as the classes begin, they schedule the classes as per the time slot and the syllabus assigned to these particular courses.

They generally give assignments to the students, which they need to work upon and submit the same by the assigned date. There by ensuring the effective participation from the students and thus enhancing the learning process. They now want to incorporate one of the most technically advanced systems, by incorporating the web portal. And so they approached us for helping them by developing this portal.

1. Existing Scenario:

Royal Educational Academy is known for prestige and quality though courses such as B.Tech, B.Pharmacy, MBA and MCA… has attracted thousands of students every year. With a large number of students and knowledge to create a huge inconvenience in scheduling study, to work upon and submit assignment’s as well as exchange of knowledge.

The need to expand the scale and create a better learning environment, the academy has decided to develop a portal.

1. Customer Requirement Specification:

A site needs to be designed where user will login as a administrator, staff or student.

If administrator login, the site mentioning the features of the site as the rights for creating, editing and deleting the usernames and passwords, etc.

If staff login, the site mentioning the features of the site as the rights for creating, editing and deleting the assignments then marks.

If student login, the site mentioning the features of the site as the rights for download assignments, feedback.

Addition, the user logged can be view report

The site also includes features FAQs

1. Functional Requirement Specification:
   1. Guest can view information:
      1. Description:

* Guest can view information of website.
  + 1. Functional Requirements:

1. System provides page information.
   1. Guest can login as admin, staff or student:
      1. Description:

* Guest can login through login page by the system provide. The system must distinguish account type and choose appropriate page.
  + 1. Functional Requirements:

1. System provide login page for guest.
2. System auto distinguish account type.
3. System auto choose to next page to show.
   1. Manager Admin:
      1. Management of administrator
         1. Description

* System must provide for administrator a management page.
  + - 1. Functional Requirements

1. Administrator can modify his profile
2. Search student or staff
3. Admin can view creating, edit, lock Account
4. Admin can view creating, edit, lock Course
5. Admin can view creating, edit, lock Batch
6. Admin can view creating, edit, lock Subject
7. Admin can view creating, edit, lock FAQs
8. Admin can view report
   * 1. Administrator can creating, edit, lock administrator or staff account
        1. Description

* The system must provide a page to allow administrator create accounts with the following information:

•Username

•Password

•Full Name

•Address

•Phone

•Select to be lock or not lock

* + - 1. Functional Requirements

1. Administrator can creating, edit, lock administrator or staff account
   * 1. Administrator can creating, editing, lock Student accounts
        1. Description

* The system must provide a page to allow administrator create accounts student with the following information:

•Username

•Password

•Full Name

•Year

•Bath

* + - 1. Functional Requirements

1. Administrator can creating, editing, lock Student accounts
   * 1. Administrator can creating, editing, lock batch
        1. Description

* The system must provide a page to allow administrator creating, editing, lock batch with the following information:

•Batch Name

•Date start

•Date end

•Select lock or not lock

* + - 1. Functional Requirements

1. Administrator can creating, editing, lock bat
   * 1. Administrator can creating, editing, lock Semester
        1. Description

* The system must provide a page to allow administrator creating, editing, lock batch with the following information:

•Semester Name

•Date start

•Date end

•Select lock or not lock

* + - 1. Functional Requirements

1. Administrator can creating, editing, lock Semester
   * 1. Administrator can creating, editing, lock subject
        1. Description

* The system must provide a page to allow administrator creating, editing, lock subject with the following information:

•Subject Name

•Batch

•Date start

•Date end

•Key enroll

•Select lock or not lock

* + - 1. Functional Requirements

1. Administrator can creating, editing, lock subject
   * 1. Administrators can Create, edit or lock FAQs
        1. Description:

* System must provide to Administrator a page which displays frequently questions and answers. Administrators can add, modify or delete these.
  + - 1. Functional Requirements:

1. Administrator can view FAQs page.
2. Administrators can Create, edit or lock FAQs.
   * 1. Administrators can view report:
        1. Description:

* System must provide to Administrator a page which displays report
  + - 1. Functional Requirements:

1. Administrator can view FAQs page.
   1. Manager Staff:
      1. Manager of Staff:
         1. Description:

* System must provide for Staff a manager page.
  + - 1. Functional Requirements:

1. Staff can modify their profile.
2. Staff can create/modify and upload /re-upload assignments.
3. Staff can view assignments of student uploaded and marks.
4. Staff can view complain query of student about marks.
5. Staff can view top 10 assignments more upload.
6. Staff can view FAQs and query to administrator.
   * 1. Staff can modify profile:
        1. Description:

* System must provide for Staff a edit page their profile.
  + - 1. Functional Requirements:

1. Staff can modify their profile.
   * 1. Manager assignments:
        1. Description:

* System must provide for staff a page for create, modify and upload, re-upload assignments. And include following information:

•Title assignments

•Upload file

* Staff could provide title for assignments.
* Staff must choose file to upload.
  + - 1. Functional Requirements:

1. Staff submits to create/modify, upload/re-upload assignments.
   * 1. Staff can download assignments of student have uploaded:
        1. Description:

* Staff can view assignments have uploaded and download them. But before staff must provide for system some information about: course, batch, subject, assignments. And system must show view assignments of student have uploaded
  + - 1. Functional Requirements:

1. Staff can download assignments.
   * 1. Staff can marks and remarks assignments:
        1. Description:

* After download file assignments of student, staff must marks assignments and re-marks if when student complain. Staff just to re-marks one time.
  + - 1. Functional Requirements:

1. Staff submits marks to assignments.
   * 1. Staff can answer complains of student:
        1. Description:

* Staff must provide content of answer. Just can answer one time.
  + - 1. Functional Requirements:

1. Staff submits to answer to complain.
   * 1. Staff can view all student in subject:
        1. Description:

* System must provide a page for staff views all students.
  + - 1. Functional Requirements:

1. Staff view list student.
   * 1. Staff can view top 10 assignments more upload:
        1. Description:

* Staff can view list top 10 assignments have most upload.
  + - 1. Functional Requirements:

1. Staff views top assignment.
   * 1. Staff can view FAQs and feedback to administrator.
        1. Description:

* System must display the FAQs to FAQs page for Staff choice. The Staff can query to administrator on query page.
  + - 1. Function Requirement:

1. Staff cans view the FAQs.
2. Staff cans feedback to Administrator.
   1. Manager Student:
      1. Management of Student.
         1. Description.

System must provide display to student with following function: modify his/her profile, view report top 10 assignment, view subject information, view FAQs and allow student feedback to Administrator.

* + - 1. Functional Requirements.

1. Student view report top 10 assignment.
2. Student view information profile.
3. Student view subject
4. Student view FAQs.
5. Student feedback to administrator.
   * 1. Student can modify his/her profile.
        1. Description.

Only student can to modify his/her profile with following information:

* Change the avatar of student.
* Edit contact.
* Change password.
  + - 1. Function Requirements.

1. Student submit to complete modify his/her profile.
2. Student change password.
   * 1. Student can report about top mark of assignment.
        1. Description.

System must provide for student to report page. Report page provide for student view all top 10 mark of assignment on subject.

* + - 1. Function Requirement.

1. Student view report assignment
   * 1. Student can view subject information.
        1. Description.

System must show all subject sort by semester of course in the body of management page of student. Student can select a subject have learned or learning to view.

After view subject:

* Student can select unit to views about information of unit:

+ Download assignment uploaded by the staff or student.

+ Upload assignment by student.

* Students can query to staff on tree days about mark of student.
  + - 1. Function Requirement.

1. Student uploads assignment.
2. Student downloads assignment.
3. Student query mark to staff.
   * 1. Student can view FAQs.
        1. Description.

System must display the FAQs to FAQs page for student views.

* + - 1. Function Requirement.

1. Student view the FAQs.
   * 1. Students can Feedback to Administrator.
        1. Description.

System must redirect to Feedback page for student can enter his/her question to Administrator.

* + - 1. Function Requirement.

1. Student feedback to administrator.
   * 1. Students can query mark.
        1. Description.

System must redirect to Query Mark page for student to enter his/her question about mark has been marked by staff. Link query marks to exist on tree days.

* + - 1. Function Requirement

1. Student query about mark to staff.
   1. Part Policies:
      1. System should generate a StudentID sequentially.

* System should generate a StudentID (username of student) number sequentially. This should be unique and cannot be changed later on. Example: stu0001…
  + 1. Time for staff modify assignment:
* Before “end-date” of assignment, staff can modify assignment.
  + 1. Policy for time marks assignments of staff:
* After “end-date” of assignment, staff can mark assignment of student.
* In 3 day after staff mark assignment, staff can re-mark assignment.
  + 1. Policy for time cans student upload assignments:
* Time for student upload assignment is must before “end-date” of assignment.
  + 1. Policy for student cans to send complains (query) about mark of assignments:
* After have marks of assignment, student can send complain mark but just only once.
  + 1. Date of the creation of the Query should default as today.
* Date of the creation of the query should default as today – This cannot be changed late on.

1. System Requirements:
   1. Hardware:
      1. Web Server

|  |  |
| --- | --- |
| Processor | Core 2 Dual or higher. |
| Memory | 1GB or greater. |
| Modem/ADSL | Connecting to internet 24/24 – 10Mbps |

* + 1. Client

|  |  |
| --- | --- |
| Processor | Core 2 Dual or higher. |
| Memory | 1GB |
| Monitor | Super VGA (800x600) or higher resolution with 256 colors. |
| Modem/ADSL | Connecting to internet - 1Mbps |

* 1. Software:
     1. Web Server

|  |  |
| --- | --- |
| Operation System | Windows XP |
| Browser | Internet Explorer 7 |
| Database | Microsoft SQL Server 2005 RTM(9.0.1399) |
| Software | Java JDK1.6.0\_22 |
|  | Jboss 5.1.0.GA |

* + 1. Client

|  |  |
| --- | --- |
| Operation System | Windows XP or later. |
| Browser | Internet Explorer 7 |

1. Development Software:

* Netbean 6.9.1
* Java JDK1.6.0\_22
* Microsoft SQL Server 2005 RTM(9.0.1399)
* Jboss 5.1.0.GA
* jQuery
* Ajax

1. Technology:

* J2EE 1.4/JAVAEE 5

|  |  |  |
| --- | --- | --- |
|  | **Prepared by (Student)** | **Approved by (Faculty)** |
| **Signature** | Team leader  **Nguyen Manh Hung** | **Kieu Trong Khanh** |
| **Date** |  |  |

Review 2

1. Use Case and Sequence Diagram:
   1. Part general:



* + 1. Login:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung | | |
| **Use Case Name** | Login | | |
| **Actors** | Guest | | |
| **Description** | Guest wants to login the system. | | |
| **Requirements** | N/A | | |
| **Pre-conditions** | N/A | | |
| **Post-conditions** | Success: | Redirect to correlative page with each role different. | |
| Fail: | Inform list of errors to Actor. | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Guest click link “Login”.  3. Guest input username and password. Then click button “Login”.  [exception 1]  [Alternative 1]  [Alternative 2] | | 2.System display login form with the following control:  “Username” text field. (required)  “Password” password field. (required)  “Login” button.  “Cancel” button  4. System checks validate of information.  [exception 2]  5. System transfer to correlative page of role. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| [Alternative 1]  1. Actor clicks link “Home”.  [Alternative 2]  3. Actor click “Cancel” button. | | 2. System transfer to home page.  4. Close form login. |
| **Exception** | Actor Actions: | | System Responses: |
| [exception 1]  1. Username or password is bank.  [exception 2]  3. Guest is not exist username and password. | | 2. Show message to notice: “Username or password not bank…!”  4. System transfer to login page. And notice for actor is invalid username and password. |

* Sequence Diagram:



* + 1. Log-out:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung | | |
| **Use Case Name** | Log-out | | |
| **Actors** | Admin/Staff/Student | | |
| **Description** | Actor wants to log-out of the System. | | |
| **Requirements** | N/A | | |
| **Pre-conditions** | Actor has logged in website. | | |
| **Post-conditions** | Success: | Return home page. | |
| Fail: | N/A | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Actor clicks ‘Log out’ button on the Actor’s Home menu. | | 2. System redirect to Logout page, removes session, and then redirect to Login page. |
| **Alternative flow** | N/A | | N/A |
| **Exception** | N/A | | N/A |

* Sequence diagram:



* + 1. Manage Profile:



* + - 1. Admin, Staff, Student modify Profile:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Vo Van Cuong | | |
| **Use Case Name** | Modify Profile | | |
| **Actors** | Admin, student, student | | |
| **Description** | Actor wants to modify profile. | | |
| **Requirements** | N/A | | |
| **Pre-conditions** | Actor has logged in website as admin/staff/student | | |
| **Post-conditions** | Success: | - Profile information of Actor has been update to database.  - Refresh page and display information after updated. | |
| Fail: | Report message to Actor in page. | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Actor clicks ‘Modify profile’ link in Actor’s Home page menu.  3. Actor updates information and click ‘Save’ button.  [Alternative]. | | 2. System redirects to Modify Profile page and display Modify Profile form with the following control:   * ‘Username’ text field (only read).   - ‘Name’ text field (is required).   * ‘Birthday’ text field (is required). * ‘Gender’ radio button group. * ‘Address’ text field * ‘Phone’ text field * ‘Email’ text field (is required). * ‘Avatar’ image. * ‘Browser’ file browser. * ‘Save’ button.   4. System checks the information [Exception 1]  5. System updates the account into database and then redirect to Actor home page. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| [Alternative]  1. Actor clicks to ‘Browser’ button on Modify Profile page.  3. Actor select path of picture after click ‘Open’ to complete selected. | | 2. System opens a dialog to select a picture to make the avatar. |
| **Exception** | Actor Actions: | | System Responses: |
| [Exception 1]  1. Actor lefts ‘Name’ text field blank.  2. Actor lefts ‘Birthday’ text field blank.  3. Actor enter invalid ‘Birthday’ text field.  4. Actor lefts ‘Address’ text field blank.  5. Actor lefts ‘Phone’ text field blank.  6. Actor enter invalid ‘Phone’ text field.  7. Actor lefts ‘Email’ text field blank.  8. Actor enter invalid ‘Email’ text field. | | System displays message to Actor:  - ‘Name is can’t be blank. Please enter your name!’  - ‘Birthday can’t be blank. Please enter your birthday!’  - ‘Birthday is invalid. Your age must from 18 ages!’  - ‘Address can’t be blank. Please enter your address!’  - ‘Phone can’t be blank. Please enter your phone!’  - ‘Phone is invalid. Phone must have 12 numbers’.  - ‘Email can’t be blank. Please enter your email’.  - ‘Email invalid. Email format is abc@example.com’ |

* Sequence Diagram:



* + - 1. Admin/Staff/Student change Password:

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Van Cuong | |
| **Use Case Name** | Change Password form | |
| **Actors** | Admin, student, student | |
| **Description** | Actor wants to change password of her/his. | |
| **Requirements** | Actor must have old password. | |
| **Pre-conditions** | Actor has logged in website as admin/staff/student | |
| **Post-conditions** | Success: Password of Actor has been updated into database.  Fail: Report message to Actor in form. | |
| **Basic flow** | Actor Actions: | System Responses: |
| 1. Actor clicks ‘Profile’ link in Actor’s Home page menu.  3. Actor click ‘Change’ button. | 2. System redirects to Modify Profile page and display Change Password form.  - Change Password form with following control:  + ‘Old Password’ password field (is required).  + ‘New Password’ password field (is required).  + ‘Confirm New Password’ password field (is required).  + ‘Change’ button.  4. System checks the information.  [Exception]  5. System process updates password information to database. |
| **Alternative low** | N/A | N/A |
| **Exception** | Actor Actions: | System Responses: |
| [Exception 1]  1. Actor lefts ‘Old Password’ text field blank.  2. Actor enter ‘Old Password’ text field wrong.  3. Actor lefts ‘New Password’ text field blank.  4. Actor lefts ‘New Password’ text field less than 6 and more than 30 characters.  5. Actor lefts ‘Confirm New Password’ text field blank.  6. Actor lefts ‘Confirm New Password’ text field less than 6 and more than 30 characters.  7. New Password and New Password. | System display message to Actor:  - ‘Old Password can’t be blank. Please try again!’  - ‘Old Password is invalid. Please try again!’  - ‘New Password can’t be blank. Please try again!’  - ‘New Password is invalid. Please try again!’  - ‘Confirm Password can’t be blank. Please try again!’  - ‘Confirm Password is invalid. Please try again!’  - ‘Confirm Password not match’. |

* Sequence Diagram:



* + 1. FAQs

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Vo Van Cuong | | |
| **Use Case Name** | FAQs. | | |
| **Actors** | Student or Staff. | | |
| **Description** | Allow Actors to views FAQs entry. | | |
| **Requirements** | - FAQs question and answer is existing in database system. | | |
| **Pre-conditions** | Actor has logged in as Staff/Student. | | |
| **Post-conditions** | Success: | Display FAQs information to Actors. | |
| Fail: | Return message to Actors in page. | |
| **Basic flow** | Actor Actions:  1. Actors click link ‘FAQs’ in Actor’s home page.  4. Actors click title of any FAQs. | | System Responses:  2. System redirect to FAQs page and checking FAQs information in database.  [Exception]  3. System display FAQs to Actor in FAQs page with FAQs list.  5. System show content of these FAQs below title FAQs. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| N/A | | N/A |
| **Exception** | Actor Actions: | | System Responses: |
| [Exception]  1. Actors click link ‘FAQs’ in Actor’s home page when FAQs does not exist in database system. | | System display message to Actor.  - ‘FAQs not found or updating by system. Sorry for this inconvenient!’ |

* Sequence Diagram:



* + 1. Feedback:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Vo Van Cuong | | |
| **Use Case Name** | Feedback | | |
| **Actors** | Staff, Student | | |
| **Description** | Actor want to feedback to Administrator | | |
| **Requirements** | N/A | | |
| **Pre-conditions** | Actor has logged in as Staff/Student. | | |
| **Post-conditions** | Success: | Feedback insert to database. | |
| Fail: | Report message to Actor in page. | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Actor clicks link “Feedback”.  3. Actor click ‘Feedback button.  [Alternative] | | 2. System redirect to Feedback page with following control:  -“Title” text field.  - “Content” text field  (Is required).  - ‘Feedback’ button.  - ‘Back’ button.  4. System checks information  [Exception]  5. System process inserts “Feedback” into database and redirect to actor home page. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| [Alternative]  1. Actor click ‘Back’ button. | | - System redirect to Student home page. |
| **Exception** | Actor Actions: | | System Responses: |
| [Exception]  1. Actors enter left ‘Title’ text field blank.  2. Actors enter left ‘Title’ text field less than 5 and more than 50 characters.  3. Actors enter left ‘Content’ text field blank.  4. Actors enter left ‘Content’ text field less than 6 and more than 256 characters. | | System display message to actor:  - “Title can’t be blank. Please try again!”  - “Title must more than 5 and less than 50 characters”  - “Content can’t be blank. Please try again!”  - “Content must more than 6 and less than 256 characters” |

* Sequence Diagram:



* 1. Part Admin:  
       
       
     1. Administrator Manager Batch



* + - 1. Administrator Search Batch:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | N/A | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Search Batch | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Search Batch in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager Batch’ link in Admin Home page  3. Actor must input require Search Name and click ‘Search’ button. [Alternative 1] | System Responses:  2. System redirects to Search Staff page and displays Search Staff form with the following control :  - ‘Batch Name’ text field  - ‘Search’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System search in database and display table with columns and data necessary follow:  - Batch Name  - Date Create  - Edit  - Delete  - Create Batch button  [Alternative 2] |
| **Alternative flow** | Actor Actions:  [Alternative 1]   1. Actor click ‘Back to Home’ button.   [Alternative 1]   1. Actor click ‘Create New Batch’ button. | System Responses:   * System returns to Admin Home page. * System returns to Admin Create Batch page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Search Name’ text field blank. | System Responses:  System displays message to Actor :  - ‘Search Name cannot be blank’. |



* + - 1. Administrator create Batch:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Batch | |
| **Actors** | Administrator | |
| **Description** | Actor wants to create Batch | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor creates a Batch in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Batch’ button in Manager Batch page  3. Actor must input all require Batch’s information and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects and displays Create Batch form with the following control :  - ‘Batch Name’ text field (require)  - ‘Day Start’ Drop-down list field (require)  - ‘Month Start’ Drop-down list field (require)  - ‘Year Start’ Drop-down list field (require)  - ‘Day End’ Drop-down list field (require)  - ‘Month End’ Drop-down list field (require)  - ‘Year End’ Drop-down list field (require)- ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System inserts the Batch into database, show message: ‘Batch has been created!’ and then redirects to Manager Batch page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Batch Name’ text field blank. 2. Actor lefts ‘Batch Name’ less than 6 or greater than 50. 3. Batch Name has already exists. 4. Actor select date Start less than current date. 5. Actor select date End less date Start. | System Responses:  System displays message to Actor :  - ‘Batch Name cannot be blank’.  - ‘Batch Name greater than 6 and less than 50.  - ‘This Batch Name has already exist!’.  - ‘Date Start required greater than current date’.  - ‘Date End required greater than Date Start’. |

* Sequence diagram:



* + - 1. Administrator edit Batch:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Batch | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Edit Batch | |
| **Requirements** | Batch must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Edit a Batch in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit Batch’ button in Manager Batch page  3. Actor must input all require Batch’s information and click ‘Update’ button. [Alternative 1] | System Responses:  2. System redirects and displays Create Batch form with the following control :  - ‘Batch Name’ text field (require)  - ‘Day Start’ Drop-down list field (require)  - ‘Month Start’ Drop-down list field (require)  - ‘Year Start’ Drop-down list field (require)  - ‘Day End’ Drop-down list field (require)  - ‘Month End’ Drop-down list field (require)  - ‘Year End’ Drop-down list field (require)- ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System update the Batch into database, show message: ‘Batch has been update!’ and then redirects to Manager Batch page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Batch Name’ text field blank. 2. Actor lefts ‘Batch Name’ less than 6 or greater than 50. 3. Batch Name has already exists. 4. Actor select date Start less than current date. 5. Actor select date End less date Start. | System Responses:  System displays message to Actor :  - ‘Batch Name cannot be blank’.  - ‘Batch Name greater than 6 and less than 50.  - ‘This Batch Name has already exist!’.  - ‘Date Start required greater than current date’.  - ‘Date End required greater than Date Start’. |

* + - 1. Administrator Delete Batch:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Batch | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete Batch | |
| **Requirements** | Batch must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a Batch in this website  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete Batch’ button in Manager Batch page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + 1. Administrator Manager Year:  
         
         
       1. Administrator View Year:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Year | |
| **Actors** | Administrator | |
| **Description** | Actor wants to View Year | |
| **Requirements** | Year of must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor View Year  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager Year’ link in Admin Home page  3. [Alternative 1] | System Responses:  2. System redirects to Manager Year page and displays table with columns and data necessary follow:  - ‘Year Name’  - ‘Date Create’  - ‘Edit’  - ‘Delete’  - Create Year button |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Create Year’ button. | System Responses:  2. System returns to Admin Create Year page. |
| **Exception** | N/A | N/A |



* + - 1. Administrator create Year:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Year | |
| **Actors** | Administrator | |
| **Description** | Actor wants to create Year | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor creates a Year in this Batch  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Year’s button in Manager Year page  3. Actor must input all require ‘Year’ information and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to Create Year (year of this batch) page and displays Create Year form with the following control :  - ‘Year Name’ text field (require)  - ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System inserts the Year into database, show message: ‘Year has been create!’ and then redirects to Admin home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Year Name’ text field blank. 2. Year has already exits. 3. Actor lefts ‘Year name’ less than 3 or greater than 20. 4. Actor select date Star less current date. | System Responses:  System displays message to Actor :  - ‘Year Name cannot be blank’.  - ‘This Year has already existed!’.  - ‘Semester name than 3 and less than 20’.  - ‘Date Start greater date current’. |

* Sequence diagram:



* + - 1. Administrator edit Year:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Year | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Edit Year | |
| **Requirements** | Year must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Update a Year in this Batch  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit Year’s button in Manager Year page  3. Actor must input all require ‘Year’ information and click ‘Update’ button. [Alternative 1] | System Responses:  2. System redirects to Edit Year page and displays Edit Year form with the following control :  - ‘Year Name’ text field (require)  - ‘Update’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System update the Year into database, show message: ‘Year has been update!’ and then redirects to Admin home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Year name’ text field blank. 2. Year has already exits. 3. Actor lefts ‘Year name’ less than 3 or greater than 20. 4. Actor select date Star less current date. | System Responses:  System displays message to Actor :  - ‘Year Name cannot be blank’.  - ‘This Year has already existed!’.  - ‘Year Name than 3 and less than 20’.  - ‘Date Start greater date current’. |

* Sequence diagram:



* + - 1. Administrator Delete Year:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Year | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete Year | |
| **Requirements** | Year must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a Year in this Batch  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete Year’ button in Manager Year page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + 1. Administrator Manager Semester:  
         
       
       1. Administrator View Semester:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Semester | |
| **Actors** | Administrator | |
| **Description** | Actor wants to View Semester | |
| **Requirements** | Semester must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor View Semester  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager Semester’ link in Admin Home page  3. [Alternative 1] | System Responses:  2. System redirects to Manager Semester page and displays table with columns and data necessary follow:  - ‘Year Name’  - ‘Date Create’  - ‘Edit’  - ‘Delete’  - Create Semester button |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Create Semester’ button. | System Responses:  2. System returns to Create Semester page. |
| **Exception** | N/A | N/A |



* + - 1. Administrator create Semester:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Semester | |
| **Actors** | Administrator | |
| **Description** | Actor wants to create Semester | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor creates a Semester in this Year  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Semester’s button in Manager Semester page  3. Actor must input all require ‘Semester’ information and click ‘Create’ button. [Alternative 1] | System Responses:  2.System redirects to Create Semester page and displays Create Semester form with the following control :  - ‘Semester name’ text field (require)  - ‘Day Start’ Drop-down list field (require)  - ‘Month Start’ Drop-down list field (require)  - ‘Year Start’ Drop-down list field (require)  - ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  6. System inserts the Semester into database, show message: ‘Semester has been create!’ and then redirects to Admin home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Semester name’ text field blank. 2. Semester has already exits. 3. Actor lefts ‘Semester name’ less than 6 or greater than 50. 4. Actor select date Star less current date. | System Responses:  System displays message to Actor :  - ‘Semester name cannot be blank’.  - ‘This Semester has already existed!’.  - ‘This Semester has already existed!’.  - ‘Semester name than 6 and less than 50’.  - ‘Date Start greater date current’. |

* Sequence diagram:



* + - 1. Administrator edit Semester:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Semester | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Edit Semester | |
| **Requirements** | Semester must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Update a Semester in this Year  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit Semester’s button in Manager Semester page  3. Actor must input all require ‘Semester’ information and click ‘Update’ button. [Alternative 1] | System Responses:  2. System redirects to Edit Semester page and displays edit Semester form with the following control :  - ‘Semester Name’ text field (require)  - ‘Day Start’ Drop-down list field (require)  - ‘Month Start’ Drop-down list field (require)  - ‘Year Start’ Drop-down list field (require)  - ‘Update’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System update the Semester into database, show message: ‘Semester has been update!’ and then redirects to Admin home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Semester name’ text field blank. 2. Semester has already exits. 3. Actor lefts ‘Semester name’ less than 6 or greater than 50. 4. Actor select date Star less current date. | System Responses:  System displays message to Actor :  - ‘Semester name cannot be blank’.  - ‘This Semester has already existed!’.  - ‘Semester name than 6 and less than 50’.  - ‘Date Start greater date current’. |

* Sequence diagram:



* + - 1. Administrator Delete Semester:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Semester | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete Semester | |
| **Requirements** | Semester must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a Semester in this Year  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete Semester’ button in Manager Semester page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + 1. Administrator Manager Subject:  
         
         
       1. Administrator View Subject:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Subject | |
| **Actors** | Administrator | |
| **Description** | Actor wants to View Subject | |
| **Requirements** | Subject must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor View Subject  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager Subject’ link in Admin Home page  3. [Alternative 1] | System Responses:  2. System redirects to Manager Subject page and displays table with columns and data necessary follow:  - ‘Year Name’  - ‘Date Create’  - ‘Edit’  - ‘Delete’  - Create Subject button |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Create Subject’ button. | System Responses:  2. System returns to Create Subject page. |
| **Exception** | N/A | N/A |



* + - 1. Administrator create Subject:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Subject | |
| **Actors** | Administrator | |
| **Description** | Actor wants to create Subject | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor creates a Subject in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Subject’s button in Manager Subject page  3. Actor must input all require ‘Subject’ information and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to Create Subject page and displays create Subject form with the following control :  - ‘Subject name’ text field (require)  - ‘Key Enroll’ text field (require)  - ‘Confirm Key Enroll’ text field (required)  - ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System inserts the Subject into database, show message: ‘Subject has been create!’ and then redirects to Admin home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Semester name’ text field blank. 2. Subject has already exist in Batch. 3. Actor select date Start less than current date. 4. Actor select date End less date Start. 5. Actor lefts ‘Key Enroll’ text field blank. 6. Actor lefts ‘Key Enroll’ less than 6 or greater than 50. 7. Actor lefts ‘Confirm Password’ text field blank. 8. Actor lefts ‘Confirm Key Enroll’ less than 6 or greater than 50. 9. Actor left ‘Password’ and ‘Confirm Password’ don’t match. | System Responses:  System displays message to Actor :  - ‘Semester name cannot be blank’.  - ‘This Subject has already existed in Batch!’.  - ‘Date Start required greater than current date’.  - ‘Date End required greater than Date Start’. |

* Sequence diagram:



* + - 1. Administrator edit Subject:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Subject | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Edit Subject | |
| **Requirements** | Subject must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Update a Subject in this Semester  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit Subject’s link in Manager Subject page  3. Actor must input all require ‘Subject’ information and click ‘Update’ button. [Alternative 1] | System Responses:  2. System redirects to Edit Subject page and displays edit Subject form with the following control :  - ‘Subject name’ text field (require)  - ‘Day Start’ Drop-down list field (require)  - ‘Month Start’ Drop-down list field (require)  - ‘Year Start’ Drop-down list field (require)  - ‘Day End’ Drop-down list field (require)  - ‘Month End’ Drop-down list field (require)  - ‘Year End’ Drop-down list field (require)  - ‘Key Enroll’ text field (require)  - ‘Confirm Key Enroll’ text field (required)  - ‘Update’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System update the Subject into database, show message: ‘Subject has been update!’ and then redirects to Admin Home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Semester name’ text field blank. 2. Subject has already exits. 3. Actor select date Start less than current date. 4. Actor select date End less date Start. 5. Actor lefts ‘Key Enroll’ text field blank. 6. Actor lefts ‘Key Enroll’ less than 6 or greater than 50. 7. Actor lefts ‘Confirm Password’ text field blank. 8. Actor lefts ‘Confirm Key Enroll’ less than 6 or greater than 50. 9. Actor left ‘Password’ and ‘Confirm Password’ don’t match. | System Responses:  System displays message to Actor :  - ‘Semester name cannot be blank’.  - ‘This Subject has already existed!’.  - ‘Date Start required greater than current date’.  - ‘Date End required greater than Date Start’.  - ‘Key Enroll cannot be blank’.  - ‘Key Enroll greater than 6 and less than 50’.  - ‘Confirm Key Enroll cannot be blank’.  - ‘Confirm Key Enroll greater than 6 and less than 50’.  - ‘Confirm Key Enroll cannot match Key Enroll’. |

* Sequence diagram:



* + - 1. Administrator Delete Subject:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Subject | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete Subject | |
| **Requirements** | Subject must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a Subject in this Semester  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete Subject’ button in Manager Subject page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + 1. Administrator Manager Subject in Batch:
       1. Administrator View Subject in Batch:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Subject Batch | |
| **Actors** | Administrator | |
| **Description** | Actor wants to View Subject in Batch | |
| **Requirements** | Subject must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor View Subject in Batch  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager Subject in Semester’ link in Admin Home page | System Responses:  2. System redirects to Manager Subject in Batch page and displays table with columns and data necessary follow:  - ‘Year Name’  - ‘Semester Name’  - ‘Subject Name’  - ‘Date Start’  - ‘Date End’  - ‘Edit’  - ‘Delete’  - ‘Year’ Drop-down list  - ‘Semester’ Drop-down list  - ‘Subject’ Drop-down list  - ‘Date Start’ datepicker  - ‘Date End’ datepicker |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + - 1. Administrator Add Subject in Batch:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Subject Batch | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Add Subject to the Batch | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Add a Subject to the Batch  Fail: Refill information | |
| **Basic flow** | In View Subject in Batch page:  Actor Actions:  1. Actor must select all required field information and click ‘Add’ button. | System Responses:  3. System checks the information  [Exception 1]  4. System inserts the Subject into database, show message: ‘Subject has been create!’ and fresh page. |
| **Alternative flow** | N/A | N/A |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor select Subject exist in Batch 2. Actor select date Start less than current date. 3. Actor select date End less than current date. 4. Actor select date End less date Start. | System Responses:  System displays message to Actor :  - ‘This Subject has already existed in Batch!’.  - ‘Date Start required greater than current date’.  - ‘Date End required greater than current date’.  - ‘Date End required greater than Date Start’. |

* Sequence diagram:



* + - 1. Administrator Edit Subject in Batch:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Subject Batch | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Edit Subject to the Batch | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Edit a Subject to the Batch  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit’s link in Manager Subject in Batch page  3. Actor must input all require ‘Subject’ information and click ‘Update’ button. [Alternative 1] | System Responses:  2. System redirects to Edit Subject in Batch page and displays table with columns and data necessary follow:  - ‘Year’ Drop-down list  - ‘Semester’ Drop-down list  - ‘Subject’ Drop-down list  - ‘Date Start’ datepicker  - ‘Date End’ datepicker  4. System checks the information  [Exception 1]  5. System update the Subject into database, show message: ‘Subject has been update!’ and redirects Manager Subject in Batch page. |
| **Alternative flow** | N/A | N/A |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor select Subject exist in Batch 2. Actor select date Start less than current date. 3. Actor select date End less than current date. 4. Actor select date End less date Start. | System Responses:  System displays message to Actor :  - ‘This Subject has already existed in Batch!’.  - ‘Date Start required greater than current date’.  - ‘Date End required greater than current date’.  - ‘Date End required greater than Date Start’. |

* Sequence diagram:



* + - 1. Administrator Delete Subject:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Subject Batch | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete Subject in Batch | |
| **Requirements** | Subject must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a Subject in this Batch  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete’ button in Manager Subject in Batch page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + 1. Administrator Manager Administrator Account  
         
         
       1. Administrator View Admin Account:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Admin Account | |
| **Actors** | Administrator | |
| **Description** | Actor wants to View Admin Account | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor View Admin Account in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager Admin Account’ link in Admin Home page  3. [Alternative 1] | System Responses:  2. System redirects to Manager Admin Account page and displays table with columns and data necessary follow:  - ‘username’  - ‘Date Create’  - ‘Edit’  - ‘Delete’  - Create Admin button |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Create Admin’ button. | System Responses:  2. System returns to Create Admin page. |
| **Exception** | N/A | N/A |



* + - 1. Administrator create Administrator account:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Admin Account | |
| **Actors** | Administrator | |
| **Description** | Actor wants to create Administrator account | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor creates a Administrator account in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Administrator’ button in Manager Administrator page  3. Actor must input all require Administrator’s information and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to Create Administrator page and displays Create Administrator account form with the following control :  - ‘Username’ text field (require)  - ‘Password’ password field (require)  - ‘Confirm Password’ password field (require)  - ‘Full Name’ text field (require)  - ‘Address’ text field (require)  - ‘Phone’ text field (require)  - ‘Reset’ button  - ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System inserts the Administrator account into database, show message: ‘Administrator account has been created!’ and then redirects to Administrator home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Username’ text field blank. 2. Actor lefts ‘Username’ less than 6 or greater than 50. 3. Username has already exists. 4. Actor lefts ‘Password’ password field blank. 5. Actor lefts ‘Password’ less than 6 or greater than 50. 6. Actor lefts ‘Confirm Password’ password field blank. 7. Actor lefts ‘Confirm Password’ less than 6 or greater than 50. 8. Actor left ‘Password’ and ‘confirm password’ don’t match. 9. Actor lefts ‘Full Name’ text field blank. 10. Actor lefts ‘Full Name’ less than 6 or greater than 50. 11. Actor lefts ‘Address’ text field blank. 12. Actor lefts ‘Address’ less than 6 or greater than 50. 13. Actor lefts ‘Phone’ wrong format. | System Responses:  System displays message to Actor :  - ‘Username cannot be blank’.  - ‘Username greater than 6 and less than 50’.  - ‘This Username has already existed!’.  - ‘Password cannot be blank’.  - ‘Password greater than 6 and less than 50’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm Password greater than 6 and less than 50’.  - ‘Confirm password not match password.’  - ‘Full Name cannot be blank’.  - ‘Full Name greater than 6 and less than 50’.  - ‘Address cannot be blank’.  - ‘Address greater than 6 and less than 50’.  - ‘Phone is invalid’. |

* Sequence diagram:



* + - 1. Administrator Delete Admin Account:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Admin Account | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete Admin Account | |
| **Requirements** | Admin Account must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a Admin Account in this Subject  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete Admin Account’ button in Manager Year page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |

* + 1. Administrator Manager Staff  
         
         
       1. Administrator Search Staff:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | N/A | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Search Staff | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Search Staff in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager Staff’ link in Admin Home page  3. Actor must input require Search Key, select Search by and click ‘Search’ button. [Alternative 1] | System Responses:  2. System redirects to Search Staff page and displays Search Staff form with the following control :  - ‘Search by’ Drop-down List field  - ‘Search Key’ text field (require)  - ‘Search’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System search in database and display table with columns and data necessary follow:  - username  - Full Name  - Edit  - Delete |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Search Key’ text field blank. | System Responses:  System displays message to Actor :  - ‘Search Key cannot be blank’. |



* + - 1. Administrator create staff account:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager staff | |
| **Actors** | Administrator | |
| **Description** | Actor wants to create staff account | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor creates a Staff account in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create staff’ button in Manager Staff page  3. Actor must input all require staff’s information and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to Create Staff page and displays Create Staff account form with the following control :  - ‘Username’ text field (require)  - ‘Password’ password field (require)  - ‘Confirm Password’ password field (require)  - ‘Full Name’ text field (require)  - ‘Address’ text field (require)  - ‘Phone’ text field  - ‘Reset’ button  - ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System inserts the Staff account into database, show message: ‘Staff account has been created!’ and then redirects to Administrator home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Username’ text field blank. 2. Actor lefts ‘Username’ less than 6 or greater than 50. 3. Username has already exists. 4. Actor lefts ‘Password’ password field blank. 5. Actor lefts ‘Password’ less than 6 or greater than 50. 6. Actor lefts ‘Confirm Password’ password field blank. 7. Actor lefts ‘Confirm Password’ less than 6 or greater than 50. 8. Actor left ‘Password’ and ‘confirm password’ don’t match. 9. Actor lefts ‘Full Name’ text field blank. 10. Actor lefts ‘Full Name’ less than 6 or greater than 50. 11. Actor lefts ‘Address’ text field blank. 12. Actor lefts ‘Address’ less than 6 or greater than 50. 13. Actor lefts ‘Phone’ wrong format. | System Responses:  System displays message to Actor :  - ‘Username cannot be blank’.  - ‘Username greater than 6 and less than 50’.  - ‘This Username has already existed!’.  - ‘Password cannot be blank’.  - ‘Password greater than 6 and less than 50’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm Password greater than 6 and less than 50’.  - ‘Confirm password not match password.’  - ‘Full Name cannot be blank’.  - ‘Full Name greater than 6 and less than 50’.  - ‘Address cannot be blank’.  - ‘Address greater than 6 and less than 50’.  - ‘Phone is invalid’. |

* Sequence diagram:



* + - 1. Administrator Delete Staff:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Staff | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete Staff | |
| **Requirements** | Staff must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a Staff in this Subject  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete Staff’ button in Manager Staff page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + 1. Administrator Manager Student:  
         
         
       1. Administrator Search Student:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | N/A | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Search Student | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Search Student in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager Student’ link in Admin Home page  3. Actor must input require Search Key, select Search by and click ‘Search’ button. [Alternative 1] | System Responses:  2. System redirects to Search Student page and displays Search Student form with the following control :  - ‘Search by’ Drop-down List field  - ‘Search Key’ text field (require)  - ‘Search’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System search in database and display table with columns and data necessary follow:  - username  - Full Name  - Batch Name  - Edit |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Search Key’ text field blank. | System Responses:  System displays message to Actor :  - ‘Search Key cannot be blank’. |



* + - 1. Administrator create student account:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Student | |
| **Actors** | Administrator | |
| **Description** | Actor wants to create student account | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor creates a Student account in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Student’ button in Manager Student page  3. Actor must input all require student’s information, select Batch Drop-down List and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to Create Student page and displays Create Student account form with the following control :  - ‘Username’ text field (require)  - ‘Password’ password field (require)  - ‘Confirm Password’ password field (require)  - ‘Full Name’ text field (require)  - ‘Address’ text field (require)  - ‘Phone’ text field  - ‘Batch’ Drop-down List field  - ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System inserts the Student account into database, show message: ‘Student account has been created!’ and then redirects to Administrator home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Username’ text field blank. 2. Actor lefts ‘Username’ less than 6 or greater than 50. 3. Username has already exists. 4. Actor lefts ‘Password’ password field blank. 5. Actor lefts ‘Password’ less than 6 or greater than 50. 6. Actor lefts ‘Confirm Password’ password field blank. 7. Actor lefts ‘Confirm Password’ less than 6 or greater than 50. 8. Actor left ‘Password’ and ‘confirm password’ don’t match. 9. Actor lefts ‘Full Name’ text field blank. 10. Actor lefts ‘Full Name’ less than 6 or greater than 50. 11. Actor lefts ‘Address’ text field blank. 12. Actor lefts ‘Address’ less than 6 or greater than 50. 13. Actor lefts ‘Phone’ wrong format. | System Responses:  System displays message to Actor :  - ‘Username cannot be blank’.  - ‘Username greater than 6 and less than 50’.  - ‘This Username has already existed!’.  - ‘Password cannot be blank’.  - ‘Password greater than 6 and less than 50’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm Password greater than 6 and less than 50’.  - ‘Confirm password not match password.’  - ‘Full Name cannot be blank’.  - ‘Full Name greater than 6 and less than 50’.  - ‘Address cannot be blank’.  - ‘Address greater than 6 and less than 50’.  - ‘Phone is invalid’. |

* Sequence diagram:



* + - 1. Administrator Delete Student:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager Student | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete Student | |
| **Requirements** | Staff must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a Student in this website  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete Student’ button in Manager Student page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + 1. Administrator Manager FAQs:  
         
       
       1. Administrator View FAQs:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager FAQs | |
| **Actors** | Administrator | |
| **Description** | Actor wants to View FAQs | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor View FAQs in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manager FAQs’ link in Admin Home page  3. [Alternative 1] | System Responses:  2. System redirects to Manager FAQs page and displays table with columns and data necessary follow:  - ‘Question’  - ‘Answer’  - ‘Edit’  - ‘Delete’  - Create Admin button |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Create FAQs’ button. | System Responses:  2. System returns to Create FAQs page. |
| **Exception** | N/A | N/A |



* + - 1. Administrator create FAQs:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager FAQs | |
| **Actors** | Administrator | |
| **Description** | Actor wants to Create FAQs | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Creates a FAQs in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create FAQs’ button in Manager FAQs page  3. Actor must input all require ‘FAQs’ and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to Create FAQs page and displays Create FAQs form with the following control :  - ‘Question’ text field (require)  - ‘Answer text field (require)  - ‘Create’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System inserts the FAQs into database, show message: ‘FAQs has been create!’ and then redirects to Admin home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Question’ text field blank. 2. Actor lefts ‘Question’ less than 3 or greater than 1000. 3. Actor lefts ‘Answer’ text field blank. 4. Actor lefts ‘Answer’ less than 3 or greater than 1000. | System Responses:  System displays message to Actor :  - ‘Question cannot be blank’.  - ‘Question greater than 6 and less than 1000’.  - ‘Answer cannot be blank’.  - ‘Question greater than 3 and less than 1000’. |

* Sequence diagram:



* + - 1. Administrator edit FAQs:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager FAQs | |
| **Actors** | Administrator | |
| **Description** | Actor wants to edit FAQs | |
| **Requirements** | This FAQs must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Edit a FAQs in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘edit FAQs’ button in Manager FAQs page  3. Actor must input all require ‘FAQs’ information and click ‘Update’ button. [Alternative 1] | System Responses:  2. System redirects to edit FAQs page and displays edit FAQs form with the following control :  - ‘Question’ text field (require)  - ‘Answer text field (require)  - ‘Edit’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System update the FAQs into database, show message: ‘FAQs has been update!’ and then redirects to Administrator home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Question’ text field blank. 2. Actor lefts ‘Question’ less than 3 or greater than 1000. 3. Actor lefts ‘Answer’ text field blank. 4. Actor lefts ‘Answer’ less than 3 or greater than 1000. | System Responses:  System displays message to Actor :  - ‘Question cannot be blank’.  - ‘Question greater than 6 and less than 1000’.  - ‘Answer cannot be blank’.  - ‘Question greater than 3 and less than 1000’. |

* Sequence diagram:



* + - 1. Administrator Delete FAQs:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | Manager FAQs | |
| **Actors** | Administrator | |
| **Description** | Administrator wants to Delete FAQs | |
| **Requirements** | Staff must exist | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor Delete a FAQs in this website  Fail: Send message error | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Delete FAQs’ button in Manager FAQs page | System Responses:  2. System update information into database |
| **Alternative flow** | N/A | N/A |
| **Exception** | N/A | N/A |



* + 1. Administrator view report:



|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | View Report | |
| **Actors** | Administrator | |
| **Description** | Actor wants to View Report | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor View Report in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘View Report’s button in Administrator Home page  3. Actor must input all require Report’s information and click ‘Submit’ button. [Alternative 1] | System Responses:  2. System redirects to View Report page and displays View Report form with the following control :  - ‘From Day’ Drop-down list field (require)  - ‘From Month’ Drop-down list field (require)  - ‘From Year’ Drop-down list field (require)  - ‘To Day’ Drop-down list field (require)  - ‘To Month’ Drop-down list field (require)  - ‘To Year’ Drop-down list field (require)  - ‘Submit’ button  - ‘Back to Admin Home’ button  4. System checks the information  [Exception 1]  5. System select in database and display table with columns and data necessary follow:  - Query  - Answer  - Staff Name |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Admin Home page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor select From Date greater current. 2. Actor select To Date greater current. 3. Actor select To Date less From Date. | System Responses:  System displays message to Actor :  - ‘From Date required less than or equal current Date’.  - ‘To Date required less than or equal current Date’.  - ‘To Date required greater than From Date’. |



* + 1. Administrator view Feedback:



|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Cong Thuong | |
| **Use Case Name** | View Feedback | |
| **Actors** | Administrator | |
| **Description** | Actor wants to View Feedback | |
| **Requirements** | N/A | |
| **Pre-conditions** | Actor has logged in with administrator roll. | |
| **Post-conditions** | Success: Actor View Feedback in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘View Feedback’s link in Home Administrator page  3. [Alternative 1] | System Responses:  2. System redirects to View Feedback page and displays Feedback |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Home page. |
| **Exception** | N/A | N/A |



* 1. Part Staff:



* + 1. Staff choose batch:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung | | |
| **Use Case Name** | Batch | | |
| **Actors** | Staff | | |
| **Description** | Staff wants to display all subject in batch. | | |
| **Requirements** | N/A | | |
| **Pre-conditions** | Actor has logged in as staff. | | |
| **Post-conditions** | Success: | Staff can view all subject in batch. | |
| Fail: | N/A | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Staff click “Batch” link.  3. Staff chooses batch in “Batch” dropdown list. | | 2. System displays “Batch Manager” page and include controls:  “Batch” dropdown list.  4. System display all subjects have in batch. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| N/A | | N/A |
| **Exception** | Actor Actions: | | System Responses: |
| N/A | | N/A |

* Sequence diagram:
  + 1. Staff manager assignments:



|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung | | |
| **Use Case Name** | Manager assignments | | |
| **Actors** | Staff | | |
| **Description** | Staff wants to view all assignments of subject. | | |
| **Requirements** | N/A | | |
| **Pre-conditions** | Actor has logged in as staff. | | |
| **Post-conditions** | Success: | Staff can view all assignments in subject that. | |
| Fail: | N/A | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Staff click “[Name subject]” link. | | 2. System displays “manager assignments” page. Include all assignment of subject. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| N/A | | N/A |
| **Exception** | Actor Actions: | | System Responses: |
| N/A | | N/A |

* Sequence diagram:



* + 1. Staff create assignments:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung | | |
| **Use Case Name** | Create assignments | | |
| **Actors** | Staff | | |
| **Description** | Staff wants to create assignments | | |
| **Requirements** | N/A | | |
| **Pre-conditions** | Actor has logged as Staff and stay at manage assignment page. | | |
| **Post-conditions** | Success: | Assignments insert completed to database. | |
| Fail: | Inform list of errors to Actor. | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Staff clicks to link “Add Assignments”.  4. Actor must input all required information then click “create” button.  [Alternative 1]  [exception 1]  6. Click “Cancel” button. | | 2. System pop-up form “modify assignments” and include controls:  “Title” text field.( previously existing data)  “File” file browser.  “Date start” text field: ( previously existing data)  “End date” text field: ( previously existing data)  “Create” button.  “Cancel” button.  5. System call method creates assignments.  [exception 2]  7. Close pop-up. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| [Alternative 1]  - Staff clicks any link. | | - System redirect into correlative page. |
| **Exception** | Actor Actions: | | System Responses: |
| [exception 1]  1. “Title” is blank or too long.  2. “File” is blank and not exist.  3. Start date than lower date current.  4. End date than lower date current.  [exception 2]  5. System fails during the creation assignments. | | - System show message for staff: “Title does not blank and limit must from 5 – 50 characters.”  - System show message for staff: “Not provided file to upload or file doesn’t exist.”  - System show message for staff: “Start date must than current date.”  - System show message for staff: “End date than start date and must be at least 1 day.”  - System display message to notice to staff: “Have error create assignments. Try again!!” |

* Sequence diagram:



* + 1. Staff modify assignments:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung | | |
| **Use Case Name** | Modify assignments | | |
| **Actors** | Staff | | |
| **Description** | Staff wants to modify assignments. | | |
| **Requirements** | Assignments not dead-time. | | |
| **Pre-conditions** | Actor has logged in as staff. | | |
| **Post-conditions** | Success: | Completed modify assignments. | |
| Fail: | Inform list of errors to Actor | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Staff clicks to link “Modify”.  3. Staff enters new data for update assignments.  4. Click “Update” button.  [Alternative 1]  [exception 1]  6. Click “Cancel” button. | | 2. System pop-up form “modify assignments” and include controls:  “Title” text field.( previously existing data)  “File” file browser.  “Date start” text field: ( previously existing data)  “End date” text field: ( previously existing data)  “Update” button.  “Cancel” button.  5. System call method creates assignments.  [exception 2]  7. Close pop-up. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| [Alternative 1]  1. Staff clicks any link. | | - System redirect into correlative page. |
| **Exception** | Actor Actions: | | System Responses: |
| [exception 1]  1. “Title” is blank or too long/short.  2. “File” is not exist.  3. Start date than lower date current.  4. End date than lower date current.  [exception 2]  5. System fails during the update assignments. | | - System show message for staff: “Title does not blank and limit must from 5 – 50 characters.”  - System show message for staff: “File doesn’t exist.”  - System show message for staff: “Start date must than current date.”  - System show message for staff: “End date than start date and must be at least 1 day.”  - System display message to notice to staff: “Have error update assignments. Try again!!” |

* Sequence diagram:



* + 1. Staff download assignments have uploaded:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung. | | |
| **Use Case Name** | Download assignments have uploaded. | | |
| **Actors** | Staff | | |
| **Description** | Staff wants download assignments of student. | | |
| **Requirements** | Must exist assignments have uploaded. | | |
| **Pre-conditions** | Actor has logged in as staff. | | |
| **Post-conditions** | Success: | N/A | |
| Fail: | Inform list of errors to Actor | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Staff click “Download” link in manager assignment page. | | 2. System check file exist and access download file.  [exception 1] |
| **Alternative flow** | Actor Actions: | | System Responses: |
|  | |  |
| **Exception** | Actor Actions: | | System Responses: |
| [exception 1]  1. System error. | | 2. System display message to notice to staff: “Have some error in system. Please, try again!!” |

* Sequence diagram:



* + 1. Staff mark assignment:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung. | | |
| **Use Case Name** | Download assignments have uploaded. | | |
| **Actors** | Staff | | |
| **Description** | Staff wants download assignments of student. | | |
| **Requirements** | Must exist assignments have uploaded. | | |
| **Pre-conditions** | Actor has logged in as staff. | | |
| **Post-conditions** | Success: | N/A | |
| Fail: | Inform list of errors to Actor | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Click title assignment.  3. Actor enters mark and click “Marks” button. | | 2. System displays all assignment of student. And controls:  -1 text field.  -“Marks” button.  4. System check validates.  [exception1]  5. System insert mart to database. |
| **Alternative flow** | Actor Actions: | | System Responses: |
|  | |  |
| **Exception** | Actor Actions: | | System Responses: |
| [exception1]  - Mark not invalid. | | - System display message to notice to staff: “Mark must is 0-100.” |

* Sequence diagram:
  + 1. Staff answer complain(query marks of student) and re-mark assignment:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung. | | |
| **Use Case Name** | Answer complains, Re-mark assignment. | | |
| **Actors** | Staff. | | |
| **Description** | Staff answer complains about marks assignments. | | |
| **Requirements** | Must have to complain of student. | | |
| **Pre-conditions** | Actor has logged in as staff. | | |
| **Post-conditions** | Success: | Answer complains completed. | |
| Fail: | Inform list of errors to Actor. | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Staff | |  |
| **Alternative flow** | Actor Actions: | | System Responses: |
|  | |  |
| **Exception** | Actor Actions: | | System Responses: |
|  | |  |

* Sequence diagram:



* + 1. Staff view top assignments most uploaded:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Nguyen Manh Hung | | |
| **Use Case Name** | view top assignments most uploaded | | |
| **Actors** | Staff | | |
| **Description** | Staff want view top assignments most uploaded. | | |
| **Requirements** | N/A | | |
| **Pre-conditions** | Actor has logged in as staff. | | |
| **Post-conditions** | Success: | Display list top assignments most upload. | |
| Fail: | N/A | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Staff lick link “View Top Assignments upload” | | 2. System provide a page include top 10 assignments have uploaded. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| N/A | | N/A |
| **Exception** | Actor Actions: | | System Responses: |
| N/A | | N/A |

* Sequence diagram:



* 1. Part Student:



* + 1. Subject:



|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Vo Van Cuong | | |
| **Use Case Name** | Subject | | |
| **Actors** | Student. | | |
| **Description** | - Actors want to views subject information.  - Actors want to view mark.  - Actors want to query for mark of assignment has been marked by Staff. | | |
| **Requirements** | - Subject must beginning or finished.  - Assignment has been marked by staff. | | |
| **Pre-conditions** | Actor has logged in System as Student. | | |
| **Post-conditions** | Success: | Display information about subject. | |
| Fail: | N/A | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Actor clicks to ‘[Subject name]’ link below [Semester name] on body Student page.    3. Actor clicks to ‘[Assignment name]’ link.  [Alternative] | | 2. System redirect to Subject page menu and display Assignment (ASM)of subject with following control:  - ‘Name of assignment’s list’ table.  - ‘[Mark of ASM’s list]’ link on table.  4. System redirect to Assignment page and display with following control:  - ‘Download’ button.  - ‘Browse’ file browser.  - ‘Upload’ button.  - ‘[Assignment name of student]’ link. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| [Alternative]  1. Actor click ‘[Mark of ASM]’ link. | | 2. System redirect to Query page and display with following control:  - ‘Student question’ text field.  - ‘Staff answer’ text field.  - ‘Send Query’ button.  - ‘Back’ button. |
| **Exception** | Actor Actions: | | System Responses: |
| N/A | | N/A |

* Sequence diagram:



* + - 1. [Assignment name]



* + - * 1. Download Assignment of Staff:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Vo Van Cuong | | |
| **Use Case Name** | Download Assignment of Staff | | |
| **Actors** | Student. | | |
| **Description** | Actor wants to download assignments of staff uploaded. | | |
| **Requirements** | An assignment (ASM) has been uploaded by staff and ASM is beginning.  Assignment is exist in system. | | |
| **Pre-conditions** | Actor has logged in System as Student. | | |
| **Post-conditions** | Success: | Return assignment file to Actor. | |
| Fail: | Return message to actor. | |
| **Basic flow** | Actor Actions:  1. Actor clicks to ‘Download’ button in line [Assignment name of Staff] on Assignment Page.  4. Actor select where to save file Assignment and click ‘Save’ to save file. | | System Responses:  2. System checks information of assignment file has been uploaded by staff on database.  [Exception]  3. System return assignment file to Actor through web browser with dialog. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| N/A | | N/A |
| **Exception** | Actor Actions: | | System Responses: |
| [Exception]  1. Actors click ‘Download’ button when file not exist in system. | | System display message to Actor:  - ‘File not found or be deleted by user.’ |

* Sequence diagram:



* + - * 1. Download Assignment of Student:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Vo Van Cuong | | |
| **Use Case Name** | Download Assignment of Student | | |
| **Actors** | Student. | | |
| **Description** | Actor wants to download assignments of his/her uploaded. | | |
| **Requirements** | - Assignments (ASM) have been uploaded by student.  - Assignment is existing in system. | | |
| **Pre-conditions** | Actor has logged in System as Student. | | |
| **Post-conditions** | Success: | Return assignment file to Actor. | |
| Fail: | Return message to actor. | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Actor click to ‘[Assignment name]’ link in line Student submitted.  4. Actor select where to save file Assignment and click ‘Save’ to save file. | | 2. System checks information of assignment file has been uploaded by student on database.  [Exception]  3. System return assignment file to Actor through web browser with dialog. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| N/A | | N/A |
| **Exception** | Actor Actions: | | System Responses: |
| [Exception]  1. Actors click ‘[Assignment name]’ button when file not exist in system. | | System display message to Actor:  - ‘File not found or be deleted by user.’ |

* Sequence diagram:



* + - * 1. Upload Assignment:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Vo Van Cuong | | |
| **Use Case Name** | Upload Assignment | | |
| **Actors** | Student. | | |
| **Description** | Actor wants to upload Assignments of his/her. | | |
| **Requirements** | An assignment is beginning and has spare time to upload. | | |
| **Pre-conditions** | Actor has logged in System as Student. | | |
| **Post-conditions** | Success: | Display information assignment on Assignment page. | |
| Fail: | Report message to Actor in page. | |
| **Basic flow** | Actor Actions: | | System Responses: |
| Actor Actions:  1. Actor clicks to ‘Browser’ button on Assignment page.  3. Actors select path file on device after click ‘Open’ to complete select file.  4. Actors click to ‘Upload’ button on the Assignment page. | | System Responses:  2. System opens a dialog for Actor select assignment file.  5. System checks information of assignment file.  [Exception]  6. System process to receive file of Actor uploaded and insert to database. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| N/A | | N/A |
| **Exception** | Actor Actions: | | System Responses: |
| [Exception]  1. Actor clicks to ‘Upload’ button on the Assignment page with path file blank. | | System display message to Actor:  - ‘Can’t find the file to upload! Click to “Browser” button to select file’. |

* Sequence diagram:



* + - 1. [Mark]



* + - * 1. Query Mark:

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | Vo Van Cuong | | |
| **Use Case Name** | Query Mark | | |
| **Actors** | Student. | | |
| **Description** | Student wants to query about mark of assignment. | | |
| **Requirements** | - Assignment has been marked by Staff and link to query has spare time.  - Query for mark of that assignment does not exist before. | | |
| **Pre-conditions** | Actor has logged in System as Student. | | |
| **Post-conditions** | Success: | Query marks insert to database. | |
| Fail: | Report message to Actor in page. | |
| **Basic flow** | Actor Actions: | | System Responses: |
| 1. Actor clicks to ‘[Mark]’ link of Assignment on Subject page.  3. Actor click ‘Send Query’ button.  [Alternative] | | 2. System redirect to Query page with following control:  - ‘Student question’ text field.  - ‘Staff answer’ text field.  - ‘Send Query’ button.  - ‘Back’ button.  4. System checks information and insert question of student to database and redirect to Subject page.  [Exception]. |
| **Alternative flow** | Actor Actions: | | System Responses: |
| [Alternative]  1. Actor enters to ‘Student question’ text field.  3. Actor view ‘Staff answer’ text field.  5. Actor click ‘Back’ button. | | 2. System process  - If this is first question on this unit then allows student enter and submit  - If not first question on this unit then allow student read only his/her question.  4. System display answer of staff about [Mark] of student below question of student.  6. System redirect to Subject page. |
| **Exception** | Actor Actions: | | System Responses: |
| [Exception]  1. Actor lefts ‘Student question’ text field blank. | | System display message to Actor:  - ‘You did not enter your question. Please enter your question!’ |

* Sequence diagram:



* + - 1. Report TOP ASM:

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Van Cuong | |
| **Use Case Name** | Report TOP ASM. | |
| **Actors** | Student. | |
| **Description** | Actor want to views report about top students have top mark’ assignment of subject selected. | |
| **Requirements** | - Subject is beginning.  - Assignments of students have been to marks by Staff. | |
| **Pre-conditions** | Actor has logged in System as Student. | |
| **Post-conditions** | Success: Display report about top students has top mark.  Fail: Report message to Actor in page. | |
| **Basic flow** | Actor Actions:  1. Actor click ‘Report TOP Assignment’ button on Subject page menu.  [Alternative] | System Responses:  2. System process checking information.  [Exception]  3. System redirect to report page and display list student with mark on top 10 with following control.  - ‘[List student name on top 10]’ table.  - ‘Back’ button. |
| **Alternative flow** | Actor Actions: | System Responses: |
| [Alternative]  1. Actor click ‘Back’ button. | 2. System redirect to Subject page. |
| **Exception** | Actor Actions: | System Responses: |
| [Exception]  1. Actor click ‘Report TOP Assignment’ button on Subject page menu when don’t have any assignments have been marked. | System display message to Actor:  - ‘Assignment is not yet rate. Please come back later!’ |

* Sequence diagram:



1. Entity Relationship Diagram (ERD):
   1. Conceptual



* 1. Logical



1. Table:
   1. Batch:



* 1. Semester:



* 1. Subject:



* 1. BatchSubject



* 1. Student



* 1. Staff



* 1. Assignment:



* 1. Mark



* 1. Account



* 1. Roles



* 1. FAQs



* 1. Feedback



Review 3

1. Database Relationship Diagram:



1. Table Definition:
   1. Account

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **username** | nvarchar | 64 | No |  |  | Username identity |
|  | password | nvarchar | 20 | No | aa1122 |  | Password |
|  | lockAccount | bit |  | No | 0 |  | 0: not lock  1: Lock |
|  | birthday | datetime |  | Yes |  |  | Birthday |
|  | fullName | nvarchar | 50 | No |  |  | Full Name |
|  | gender | bit |  | Yes | 0 |  | Gender:  0: Female  1: Male |
|  | avatar | nvarchar | 50 | Yes |  |  | avatar |
|  | address | nvarchar | 50 | Yes |  |  | Address |
|  | phone | nvarchar | 20 | Yes |  |  | Phone |
|  | email | nvarchar | 20 | Yes |  |  | Email |

* + 1. Constraints:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| account | +  (username,password) | - | +  (password, lockAccount) |

* + 1. Exception:
* ‘Username cannot be blank’.
* ‘Username greater than 6 and less than 50’.
* ‘This Username has already existed!’.
* ‘Password cannot be blank’.
* ‘Password greater than 6 and less than 50’.
* ‘Confirm Password can’t be blank’.
* ‘Confirm Password greater than 6 and less than 50’.
* ‘Confirm password not match password.’
* ‘Full Name cannot be blank’.
* ‘Full Name greater than 6 and less than 50’.
* ‘Address cannot be blank’.
* ‘Address greater than 6 and less than 50’.
* ‘Phone is invalid’.
  1. Roles

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **role** | varchar | 50 | No |  |  | Roles |
| PK | **username** | varchar | 64 | No |  |  | User name |

* + 1. Constraints:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Roles | +  (role,username) | - | +  (role,username) |

* 1. Students:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **studentID** | nvarchar | 64 | No |  |  | Batch ID Identity |
|  | batchID | int |  | No |  |  | Batch ID |
|  | lockStudent | bit |  | No | 0 |  | Lock:  0: not lock  1: lock |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
|  |  |  |  |

* + 1. Exception:
  1. Batch

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **batchID** | int |  | No | 1 |  | Batch ID Identity |
|  | batchName | nvarchar | 30 | No |  |  | Batch Name |
|  | dateStart | datetime | 8 | No |  |  | Date Start batch |
|  | lockBatch | bit |  | No | 0 |  | Gender:  0: Not Lock  1: Lock |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Batch | +  (batchName,dateStart) | - | +  (batchName,dateStart,lockBatch) |

* + 1. Exception:
  1. Semester

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **semesterID** | int |  | No | 1 |  | Semester ID Identity |
|  | semesterName | nvarchar | 30 | No |  |  | Semester Name |
|  | lockSemester | bit |  | No | 0 |  | Lock:  0: Not Lock  1: Lock |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Semester | +  (semesterName) | - | +  (semesterName,lockSemester) |

* + 1. Exception:
  1. Subject

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **subjectID** | int |  | No | 1 |  | Subject ID Identity |
|  | subjectName | nvarchar | 30 | No |  |  | Subject Name |
|  | lockSubject | bit |  | No | 0 |  | Lock:  0: not lock  1: lock |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
|  |  |  |  |

* + 1. Exception:
  1. BatchSubject

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **batchSubjectID** | int |  | No |  |  | Batch Subject ID Identity |
|  | batchID | int |  | No |  |  | Batch ID |
|  | semesterID | int |  | No |  |  | Semester ID |
|  | subjectID | int |  | No |  |  | Subject ID |
|  | startDate | datetime |  | No |  |  | Date Start Subject |
|  | dateEnd | datetime |  | No |  |  | Date End Subject |
|  | lockBatchSubject | bit |  | No | 0 |  | Lock:  0: not lock  1: lock |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| BatchSubject | + | - | + |

* + 1. Exception:
  1. Assignment

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | assignmentID | int |  | No |  |  | Assignment Staff ID |
|  | assignmentTitle | nvarchar | 50 | No |  |  | Assignment Name |
|  | pathFile | nvarchar | 30 | No |  |  | Assignment Path |
|  | dateStart | datetime |  | No |  |  | Date Start assignment |
|  | dateEnd | datetime |  | No |  |  | Date End assignment |
|  | subjectID | int |  | No |  |  | Subject ID |
|  | staffID | nvarchar | 64 | No |  |  | Staff ID |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
|  |  |  |  |

* + 1. Exception:
  1. Marks:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | markID | Int |  |  |  |  | Mark ID identity |
|  | assignmentID | Int |  | No |  |  | Assignment ID |
|  | studentID | Varchar | 64 | No |  |  | Assignment Student ID |
|  | pathFile | varchar | 30 | No |  |  | Path File |
|  | mark | Tinyint |  | No |  |  | Mark of assignment |
|  | query | nvarchar | 100 |  |  |  | Query of Student |
|  | answer | nvarchar | 100 |  |  |  | Answer of Staff |
|  | dateQuery | datetime |  |  | © |  | Date query |
|  | dateMark | datetime |  |  |  |  | Date upload assignment |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
|  |  |  |  |

* + 1. Exception:
  1. FAQs

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **faqID** | int |  | No | 1 |  | FAQs ID Identity |
|  | faqTitle | nvarchar | 30 | No |  |  | FAQs title |
|  | fagContent | nvarchar | 200 | No |  |  | FAQs content |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
|  |  |  |  |

* + 1. Exception:
  1. Feedback

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | feedbackID | int |  | No | 1 |  | Feedback ID identity |
|  | feedbackTitle | nvarchar | 50 | No |  |  | Feedback title |
|  | feedbackContent | nvarchar | 200 | No |  |  | Feedback Answer |
|  | username | nvarchar | 64 | No |  |  | Username create feedback |

* + 1. Constraint:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
|  |  |  |  |

* + 1. Exception:

1. Other constrains:
   1. Role & account:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| account | - | +  (username not exist in role) | - |
| Role | +  (account.[username] must exist) | - | +  (account.[username] must exist) |

* 1. Account & student:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Account | - | +  (username not exist student) | - |
| student | +  (account.[username] must exist) | - | +  (account.[username] must exist) |

* 1. Account & assignment:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Account | - | +  (username[staffID] not exist assignment) | - |
| assignment | +  (Account.[ username] must exist) | - | +  (Account.[ username] must exist) |

* 1. Account & feedback:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| account | - | +  (username not exist in Feedback) | - |
| Feedback | +  (Account.[ username] must exist) | - | +  (Account.[ username] must exist) |

* 1. Student & marks:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Student | - | +  (student.[studentID] not exist in marks) | - |
| marks | +  (student.[studentID] must exist) | - | +  (student.[studentID] must exist) |

* 1. Assignments & marks:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Assignments | - | +  (Assignments.[ AssignmentsID] not exist in marks) | - |
| marks | +  (Assignments.[ AssignmentsID] must exist) | - | +  (Assignments.[ AssignmentsID] must exist) |

* 1. Subject & batchsubject:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Subject | - | +  (Subject.[subjectID] not exist in batchsubject) | - |
| batchsubject | +  (Subject.[subjectID] must exist) | - | +  (Subject.[subjectID] must exist) |

* 1. semester & batchsubject:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| semester | - | +  (semester.[ semesterID] not exist in batchsubject) | - |
| batchsubject | +  (semester.[ semesterID] must exist) | - | +  (semester.[ semesterID] must exist) |

* 1. Batch & batchsubject:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Batch | - | +  (Batch.[ batchID] not exist in batchsubject) | - |
| batchsubject | +  (Batch.[ batchID] must exist) | - | +  (Batch.[ batchID] must exist) |

1. Site map:



* 1. Part Admin:



* 1. Part Staff:



* 1. Part Student:



1. GUI Design:
   1. Homepage:



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtUsername | Textbox |  |  |  |  |
| 2 | txtPassword | Textbox |  |  |  |  |
| 3 | btnLogin | Button |  | click | Login to the system and redirect to User’s Home page.  Check txtUsername, txtPassword validation |  |
| 4 | SlideShow | Image |  |  |  |  |

* 1. Part Admin:
  2. Part Staff:
     1. Manager batch page:
        1. Before:



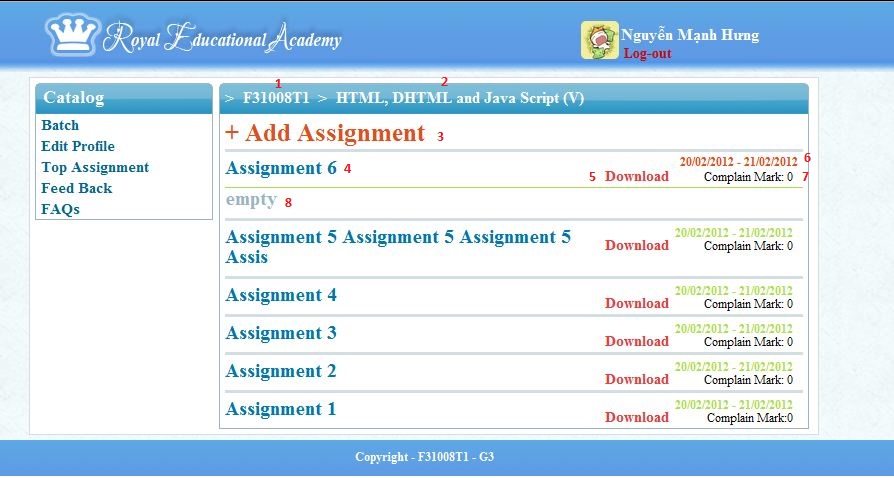
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | avata | imange |  |  |  |  |
| 2 | fullname | Static text |  |  | Display full name of user.  [Username.fullname] |  |
| 3 | LogOut | Link |  | Click | Log-out system and redirect to homepage. |  |
| 4 | batch | Link |  | Click | Redirect to batch manager page. |  |
| 5 | EditProfile | Link |  | Click | Redirect to edit profile page |  |
| 6 | TopAssignment | Link |  | Click | Redirect to top assignment page |  |
| 7 | Feed back | Link |  | Click | Redirect to feedback page |  |
| 8 | FAQs | Link |  | Click | Redirect to FAQs page |  |
| 9 | Batch manager | Static text |  |  | Display information about current page. |  |
| 10 | chooseBatch | Dropdown list |  | select | Display all batch in system. And choose batch will display all subject of batch. |  |

* + - 1. After:



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | semester | Static text |  |  | Display semester of batch  [semester.semesterName] |  |
| 2 | subject | link |  | click | Redirect to manager assignment.  [subject.subjectName] |  |

* + 1. Manager assignment page:





|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Batch name | link |  | click | Redirect to batch manager and display subject of batch.  [batch.batchID] |  |
| 2 | Subject name | Static text |  | click | Display name of subject.  [subject.subjectName] |  |
| 3 | Add assignment | link |  | click | Pop-up form adds assignment. |  |
| 4 | Name assignment | link |  | click | Display student have upload assignment. And null will display “Empty”  [assignment.assignmentName] |  |
| 5 | download | link |  | click | Download assignment.  [assignment.pathFile] |  |
| 6 | Date assignment | link |  | click | Pop-up forms edit/modify assignment.  [assignment.startDate – assignment.endDate] |  |
| 7 | Count Complain | Static text |  |  | Display sum complain re-mark (query) of student. |  |
| 8 | Show Empty | Static text |  |  | Display no assignment. |  |
| 9 | Avatar Student | image |  |  | Display avatar of student |  |
| 10 | Student ID | Static text |  |  | Display student ID  [Student.studentID] |  |
| 11 | Student name | Static text |  |  | Display student name  [account.fullname] |  |
| 12 | studentUpload | Static text |  |  | Display date student upload assignment.  [marks.date] |  |
| 13 | downloadStudent | link |  | click | Download assignment of student upload.  [marks.pathFile] |  |
| 14 | mark | link |  | click | Pop-up form view complains mark or re-marks assignment.  [marks.mark] |  |
| 15 | Mark complain | link |  | click | Pop-up form answer complains mark and re-marks assignment.  [marks.mark]+[!] |  |
| 16 | mark | button |  | click | Insert mark to database.  [marks.mark] |  |
| 17 | txtmark | textbox | Not null, 0 <= value <=100 | enter | Enter mark of assignment. |  |

* + 1. Pop-up add new assignment:



* + 1. Pop-up edit/modify assignment:



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtTitle | Textbox | Not null, 5 – 50 characters |  | Title of assignment. |  |
| 2 | txtStart | Textbox | Not null,  [mm-dd-yyyy] |  | Start date of assignment |  |
| 3 | txtEnd | Textbox | Not null,  [mm-dd-yyyy] |  | End date of assignment |  |
| 4 | fileUpload | File browser | Document file. |  | File assignmet will upload. |  |
| 5 | Ok | Button |  |  | Insert/update assignment to database |  |
| 6 | Cancel | Button |  |  | Close pop-up |  |
| 7 | Calendar | calendar |  |  | Pop-up calendar. |  |
| 8 | close | clink |  |  | Close pop-up |  |

* + 1. Pop-up form complain mark and re-marks:



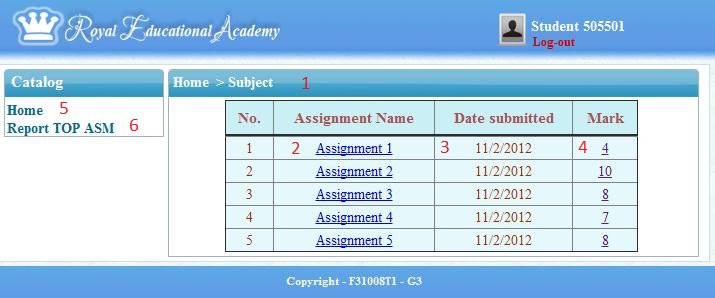
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | ID & name student | Static text |  |  | Display student ID and full name.  [student.studentID], [username.fullname] |  |
| 2 | Content complain | Static text |  |  | Display complain mark (query) of student  [marks.query] |  |
| 3 | linkDown | link |  | lick | Download assignment of student.  [marks.pathFile] |  |
| 4 | txtRe-mark | textbox | 0 <= value <=100 | enter | Enter mark of assignment.  [marks.mark] |  |
| 5 | btnMark | button |  | lick | Update mark for assignment. |  |
| 6 | Content answer | Static text |  |  | Display content answer of staff  [marks.answer] |  |
| 7 | txtAnswer | Text area | 200 <= character |  |  |  |
| 8 | btnSend | Button |  | Click | Insert answer to database. |  |
| 9 | btnCancel | button |  | click | Close pop-up. |  |

* 1. Part Student:



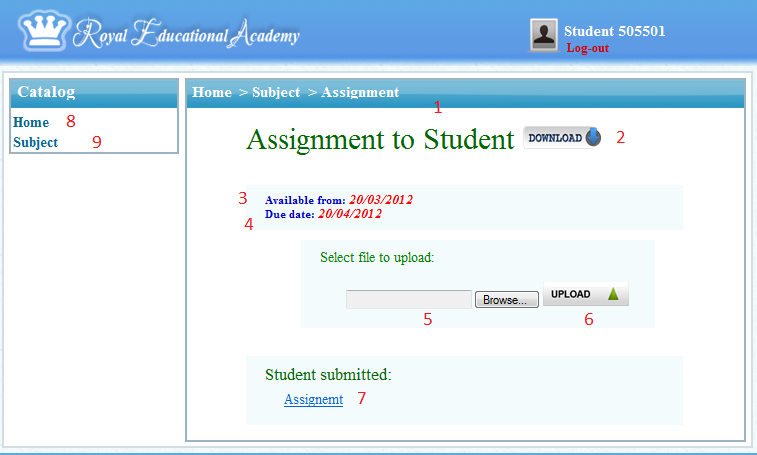
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Avatar | Image |  |  | Display Avatar of student |  |
| 2 | lnkUsername | Hyper Link |  | Click | Redirect to Profile Page |  |
| 3 | lnkLogout | Hyper Link |  | Click | Log out system |  |
| 4 | lnkProfile | Hyper Link |  | Click | Redirect to Profile Page |  |
| 5 | lnkFeedback | Hyper Link |  | Click | Redirect to Feedback Page |  |
| 6 | lnkFAQs | Hyper Link |  | Click | Redirect to FAQs Page |  |
| 7 | lnkSemester | Hyper Link |  | Click | Display subject’s list of lnkSemester. |  |
| 8 | lnkSubject | Hyper Link |  | Click | Redirect to Subject Page |  |

* + 1. Subject page



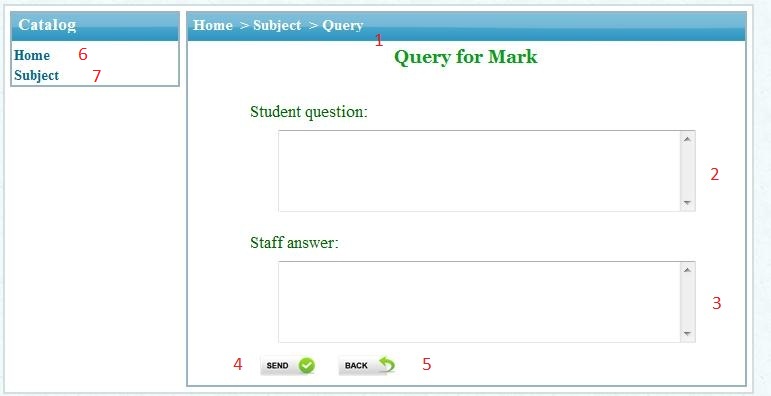
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | frmSubject | Form |  |  | Display report top ASM with control: 2, 3, 4, 5. |  |
| 2 | lnkAssignment | Hyper Link |  | Click | Redirect to Assignment page |  |
| 3 | lblDateSubmit | Label |  |  |  |  |
| 4 | lnkQueryMark | Hyper Link |  | Click | Redirect to Query page |  |
| 5 | lnkHome | Hyper Link |  | Click | Redirect to Student home page |  |
| 6 | lnkReportASM | Hyper Link |  | Click | Redirect to Report Assignment page |  |

* + 1. Assignment page



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | frmAssignment | Form |  |  |  |  |
| 2 | btnDownloadASta | Button |  | Click | Download assignment (ASM) uploaded by Staff |  |
| 3 | lblDateABegin | Label |  | Click | Display limit time to do ASM from day |  |
| 4 | lblDateAEnd | Label |  | Click | Display limit time to do ASM to day |  |
| 5 | btnSelectFile | Button | Not Null | Click, Select | Select ASM’s file of Student in device of his/her. |  |
| 6 | btnUploadA | Button |  | Click | Upload ASM’s file selected by Student to system database. |  |
| 7 | lnkStdASM | Hyper Link |  | Click | Download assignment (ASM) uploaded by Student |  |
| 8 | lnkHome | Hyper Link |  | Click | Redirect to Student home page |  |
| 9 | lnkSubject | Hyper Link |  | Click | Redirect to Subject page |  |

* + 1. Query page



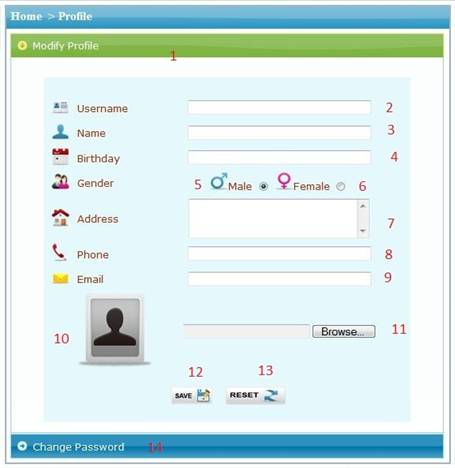
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | frmQuery | Form |  |  | Display report top ASM with control: 2, 3, 4, 5. |  |
| 2 | txtStdQuestion | Text Area | Not Null, 6<length <=200 characters | Enter | Question of Student to query for mark |  |
| 3 | txtStaAnswer | Text Area |  | Enter | Display answer of staff after reply. |  |
| 4 | btnStdSendQ | Button |  | Click | Insert question of student to database |  |
| 5 | btnBack | Button |  | Click | Redirect to Subject page |  |
| 6 | lnkHome | Hyper Link |  | Click | Redirect to Student home page |  |
| 7 | lnkSubject | Hyper Link |  | Click | Redirect to Subject page |  |

* + 1. Report Top ASM page



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | frmReportASM | Form |  |  | Display report for ASM with control: 2, 3 |  |
| 2 | lblStdName | Label |  |  | Name of student |  |
| 3 | lblTotalASM | Label |  |  | Total of Assignment |  |
| 4 | lnkHome | Hyper Link |  | Click | Redirect to Student home page |  |

* 1. Profile Page
     1. Modify Profile form



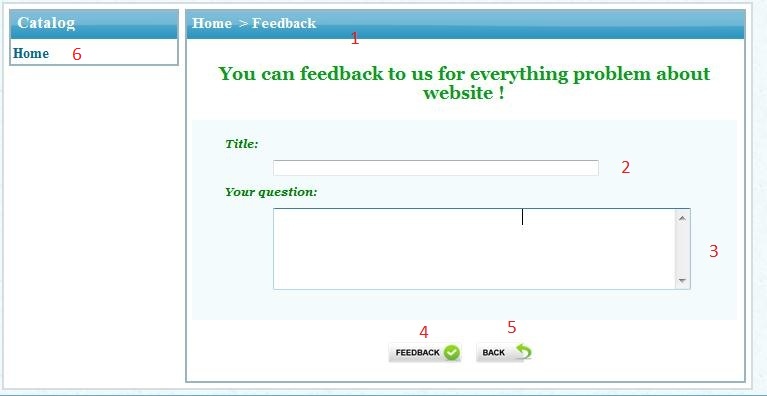
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lnkModifyProfile | Hyper Link |  | Click | Display information and modify profile of Actor with following control:2,3,4,5,6,7,8,9,10,11,12,13 |  |
| 2 | txtUsername | Text field |  |  | Display username of Actor. |  |
| 3 | txtFullName | Text field | Not Null | Enter | Full name’s Actors |  |
| 4 | txtBirthday | Text field | Not Null, More than 18 years old | Enter | Birthday’s Actor |  |
| 5 | rdoMale | Radio button |  | Select | Gender | Selected |
| 6 | rdoFemale | Radio button |  | Select | Gender |  |
| 7 | txtAddress | Text field | Not Null | Enter | Address’s Actor |  |
| 8 | txtPhone | Text field | Not Null, Format is valid | Enter | Phone number’s Actor |  |
| 9 | txtEmail | Text field | Not Null, Format is valid | Enter | Email address’s Actor |  |
| 10 | imgAvatar | Image |  |  | Avatar’s Actor |  |
| 11 | btnSelectAvt | Button |  | Click | Select image to make Avatar. |  |
| 12 | btnSave | Button |  | Click | Update information’ profile to database. |  |
| 13 | btnReset | Button |  | Click | Reset all text field is null. |  |
| 14 | lnkChangePass | Hyper Link |  | Click | Display change password form in below. |  |

* + 1. Change Password form



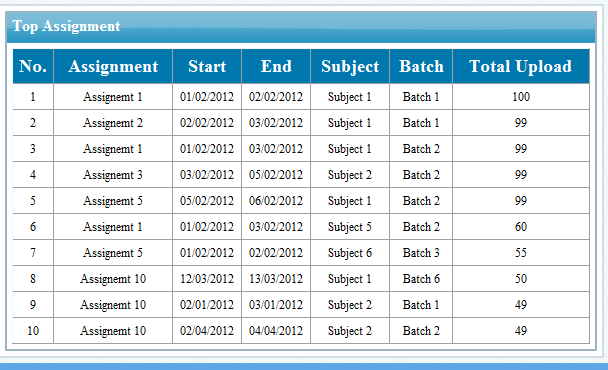
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lnkChangePass | Hyper Link |  | Click | Display change password form with following control: 2, 3, 4, 5, 6. |  |
| 2 | pwdOldPassword | Password field | Not Null, Exist in database | Enter | Old password’s Actor to login. |  |
| 3 | pwdNewPassword | Password field | Not Null, 6<= length <= 30 | Enter | New password’s Actor to login. |  |
| 4 | pwdConfirmPassword | Password field | Same with txtNewPassword | Enter | Confirm new password match with new password. |  |
| 5 | btnChange | Button |  | Click | Update password to database |  |
| 6 | btnReset | Button |  | Click | Reset all password field null |  |
| 7 | lnkModifyProfile | Hyper Link |  | Click | Display information and modify profile form. |  |

* + 1. Feedback Page



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | frmFeedback | Form |  |  | Display feedback form with following control: 2, 3, 4, 5 |  |
| 2 | txtFBTitle | Text field | Not Null,  5 < length <=50 | Enter | Title of question to feedback. |  |
| 3 | txtFBQuestion | Text field | Not Null,  6 < length<=200 | Enter | Question to feedback |  |
| 4 | btnFeedback | Button |  | Click | Insert feedback to database |  |
| 5 | btnBack | Button |  | Click | Redirect to Student home page |  |

1. Report:
   1. Part Admin:
   2. Part Staff:



|  |  |
| --- | --- |
| **Report Resources** | |
| **Table name** | **Field name** |
| Assignment | Title, startDate, endDate, |
| Subject | subjectName |
| Batch | batchName |

* 1. Part Student:

Task Sheet

* CRS document:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No 1** | | **Project Title** | **Activity Plan  Prepared By** | **Date of Preparation of Activity Plan** | | | | |
| **Sr. No** | **Task** | **Start Date (dd/mm/yyyy)** | **End Date (dd/mm/yyyy)** | **Actual Days** | **Member’s name** | **Status** |
| **CRS document** | | | | | | | | |
| 1 | Introduction | **Supportive Learning** | **Nguyen Manh Hung** | 20/03/2012 | 21/03/2012 | 1 | Nguyen Manh Hung | FINISH |
| 2 | Existing Scenario | 20/03/2012 | 21/03/2012 | 1 | Vo Van Cuong | FINISH |
| 3 | Customer Requirement  Specification | 20/03/2012 | 21/03/2012 | 1 | Nguyen Cong Thuong | FINISH |
| 4 | Functional Requirement  Specification | 20/03/2012 | 21/03/2012 | 1 | All team | FINISH |
| 5 | System Requirements | 20/03/2012 | 21/03/2012 | 1 | Vo Van Cuong | FINISH |
| 6 | Development Software | 20/03/2012 | 21/03/2012 | 1 | Nguyen Cong Thuong | FINISH |
| 7 | Technology | 20/03/2012 | 21/03/2012 | 1 | Nguyen Manh Hung | FINISH |

* Use Case and Sequence Diagrams - Entity Relationship Diagram:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No 1** | | **Project Title** | **Activity Plan  Prepared By** | **Date of Preparation of Activity Plan** | | | | |
| **Sr. No** | **Task** | **Start Date (dd/mm/yyyy)** | **End Date (dd/mm/yyyy)** | **Actual Days** | **Member’s name** | **Status** |
| **Use Case and Sequence Diagrams - Entity Relationship Diagram** | | | | | | | | |
| 1 | Part general | **Supportive Learning** | **Nguyen Manh Hung** | 22/03/2012 | 28/03/2012 | 7 | **Nguyen Manh Hung,**  **Vo Van Cuong** | FINISH |
| 2 | Part admin | 22/03/2012 | 28/03/2012 | 7 | **Nguyen Cong Thuong** | FINISH |
| 3 | Part staff | 22/03/2012 | 28/03/2012 | 7 | **Nguyen Manh Hung** | FINISH |
| 4 | Part student | 29/03/2012 | 30/03/2012 | 2 | **Vo Van Cuong** | FINISH |
| 5 | Entity Relationship Diagrams | 29/03/2012 | 30/03/2012 | 2 | **Nguyen Manh Hung** | FINISH |
| 6 | Table | 29/03/2012 | 30/03/2012 | 2 | **Nguyen Cong Thuong Vo Van Cuong** | FINISH |
| 7 | Database Relationship | 31/03/2012 | 02/04/2012 | 2 | **Nguyen Manh Hung** | FINISH |
| 8 | Table Definition | 31/03/2012 | 02/04/2012 | 2 | **Nguyen Cong Thuong Vo Van Cuong** | FINISH |

* Gui Design:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No 1** | | **Project Title** | **Activity Plan  Prepared By** | **Date of Preparation of Activity Plan** | | | | |
| **Sr. No** | **Task** | **Start Date (dd/mm/yyyy)** | **End Date (dd/mm/yyyy)** | **Actual Days** | **Member’s name** | **Status** |
|  | | | | | | | | |
| 1 |  | **Supportive Learning** | **Nguyen Manh Hung** |  |  |  |  | FINISH |
| 2 |  |  |  |  |  | FINISH |
| 3 |  |  |  |  |  | FINISH |
| 4 |  |  |  |  |  | FINISH |
| 5 |  |  |  |  |  | FINISH |
| 6 |  |  |  |  |  | FINISH |

|  |  |  |
| --- | --- | --- |
|  | **Prepared by (Student)** | **Approved by (Faculty)** |
| **Signature** | Team leader  **Le Viet Truc** | **Kieu Trong Khanh** |
| **Date** |  |  |