**MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Q & A Platform for Educators**

|  |  |
| --- | --- |
| **Group 11** | |
| **Group members** | Kha Hoàng Minh – Team Leader – SE60979  Trương Như Khang – Team Member – SE60992  Trần Đìng Thi – Team Member – SE61137 (Dropped out)  Tạ Đức Tùng – Team Member – 60416 (Dropped out) |
| **Supervisor** | Nguyễn Huy Hùng |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | QAPE |

-Ho Chi Minh City, May 11 2015-

*This page is intentionally left blank*

# Table of Contents

[Table of Contents 3](#_Toc424719131)

[List of Tables 4](#_Toc424719132)

[List of Figures 4](#_Toc424719133)

[Definitions, Acronyms, and Abbreviations 4](#_Toc424719134)

[A. Report No. 1 Introduction 5](#_Toc424719135)

[1. Project Information 5](#_Toc424719136)

[2. Introduction 5](#_Toc424719137)

[3. Current Situation 5](#_Toc424719138)

[4. Problem Definition 5](#_Toc424719139)

[5. Proposed Solution 6](#_Toc424719140)

[5.1 Feature functions 6](#_Toc424719141)

[5.2 Advantages and disadvantages 6](#_Toc424719142)

[6. Functional Requirements 6](#_Toc424719143)

[7. Role and Responsibility 7](#_Toc424719144)

[B. Report No.2 Software Project Management Plan 8](#_Toc424719145)

[1. Problem Definition 8](#_Toc424719146)

[1.1 Name of this Capstone Project 8](#_Toc424719148)

[1.2 Problem Abstract 8](#_Toc424719149)

[1.3 Project Overview 8](#_Toc424719150)

[2. Project organization 10](#_Toc424719151)

[2.1 Software Process Model 10](#_Toc424719152)

[2.2 Roles and responsibilities 11](#_Toc424719153)

[2.3 Tools and Techniques 12](#_Toc424719154)

[3. Project Management Plan 13](#_Toc424719155)

[3.1 Product Backlog 13](#_Toc424719156)

[3.2 Sprint Backlog 13](#_Toc424719157)

[3.3 All Meeting Minutes 14](#_Toc424719158)

[4. Coding Convention 14](#_Toc424719159)

[C. Report No. 3 Software Requirement Specification 15](#_Toc424719160)

[1. User Requirement Specification 15](#_Toc424719161)

[1.1 Guest Requirement 15](#_Toc424719162)

[1.2 Student Requirement 15](#_Toc424719163)

# List of Tables

[Table 1: Definitions 4](#_Toc419743945)

[Table 2: Roles and Responsibilities 7](#_Toc419743946)

[Table 3 Current Situation 8](#_Toc419743947)

[Table 4: Hardware Requirement for Server 10](#_Toc419743948)

[Table 5: Roles and Responsibilities Details 11](#_Toc419743949)

[Table 6: Product Backlog Detail 12](#_Toc419743950)

[Table 7: Sprint 1 13](#_Toc419743951)

# List of Figures

[Figure 1: Scrum Development Model\* 12](#_Toc424836817)

[Figure 2: System Overview Use Case 18](#_Toc424836818)

[Figure 3: <Guest> Overview Use Case 18](#_Toc424836819)

[Figure 4: <Guest> Login Use Case Diagram 19](#_Toc424836820)

[Figure 5: <Guest> Register Use Case Diagram 20](#_Toc424836821)

[Figure 6: <Teacher> Overview Use Case 22](#_Toc424836822)

[Figure 7: <Teacher> Create class Use Case Diagram 23](#_Toc424836823)

[Figure 7: <Teacher> Invite student Use Case Diagram 25](#_Toc424836824)

[Figure 9: <Teacher> Edit class Use Case Diagram 27](#_Toc424836825)

[Figure 10: <Teacher> Close class Use Case Diagram 29](#_Toc424836826)

[Figure 11: <Teacher> Reopen class Use Case Diagram 30](#_Toc424836827)

[Figure 12: <Teacher> Accept request of student Use Case Diagram 32](#_Toc424836828)

[Figure 13: <Teacher> Remove student Use Case Diagram 33](#_Toc424836829)

[Figure 14: <Teacher> Create post Use Case Diagram 35](#_Toc424836830)

[Figure 15: <Teacher> Edit post Use Case Diagram 37](#_Toc424836831)

[Figure 16: <Teacher> Close post Use Case Diagram 38](#_Toc424836832)

[Figure 17: <Teacher> Open post Use Case Diagram 40](#_Toc424836833)

[Figure 18: <Teacher> Delete post Use Case Diagram 41](#_Toc424836834)

[Figure 19: <Teacher> Upload material Use Case Diagram 43](#_Toc424836835)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Name** | **Definition** |
| QAPE | Q & A Platform for Educators |

Table 1: Definitions

# Report No. 1 Introduction

## Project Information

* Project name: **Q&A Platform for Educators**
* Project Code: **QAPT11**
* Product Type: **Web Application**
* Start Date: **May 11th, 2015**
* End Date: **August 11th, 2015**

## Introduction

Nowadays, the development of internet and information technology led to the dramatic increase in trend of studying online. More and more information together with limited school time make having a website that supports studying is needed.

This project is a Q&A Web application where students and teachers can exchange information and other educational materials. Students can learn more things about the subject from posts of teachers or find the answers from other people.

## Current Situation

There is a lot of popular Q&A websites like Quora, Yahoo! Answers, Stack Overflow… People can post threads or answer the questions pretty easy using its account or connect to another account. These sites also have rating system for questions and responses in order to rate users as well.

Quora is rated as the best Q&A websites where you can find the answer for your question from real doctors, lawyers, chefs, engineers and more.

Yahoo! Answer is such as a longstanding Q&A website that everyone can join a thread and answer their question. The response will be awarded 10 points if the rate is highly recommended or 2 points if the rate is lower.

Stack Overflow is a Q&A websites for IT people with a lot of the professional and enthusiasm users. You can post a thread and wait responses from a lot of experienced programmers.

## Problem Definition

Most of Q&A problems are about the rewards of their good contributions. Somebody complained they should deserve better.

Quora and Yahoo! Answer are known as the Q&A famous website but their objects are about a lot of different issues in life. These websites doesn’t only focus on education so that students and teachers will be difficult for sharing their knowledge. Spamming questions are also problems of them. On the other hand, Stack Overflow only focuses on programing rather than diversifying subjects. All of them do not yet support the true studying environment.

## Proposed Solution

We suggest developing a Q&A website that focuses on studying. This website is such as a university in reality that includes many classes, teachers and students. It supports teachers share their additional lectures because of lacking of time. It also helps students ask self-confidently because some of them feel shy when asking in school.

### **Feature functions**

* Teachers have their classes with lectures and students. They can share additional lectures and studying materials, invites students join their classes. They also can invite other teachers together discuss about lectures or question.
* Students can ask about not yet understanding issues. They also join in classes to learn additional subject, follow their favorite teachers and manage studying material that uploaded by teachers.

### Advantages and disadvantages

The advantages and disadvantages of the proposed solution:

* Advantage:
* Teachers can upload their studying material (.doc, .pdf…) and students can manage it online.
* By supporting real time notification and discussion, students can imagine this website as the university with many classes, teachers and friends. They will feel free to ask anything.
* Users will have their right answers faster than they do on their own. That leads to learn better in real classes.
* System has a variety of categories that is better than Stack Overflow but it is not a mess such as Quora or Yahoo! Answers.
* Disadvantage:
* System do not support the users suggest teacher to join threads to answer the questions.
* Cost time waiting to approve to join a class.

## Functional Requirements

Function requirements of the system are listed as below:

**For Teachers:**

* Create classes.
* Create a post. Post types include: articles, questions, answer, and upload material.
* Invite other teachers to answer questions in the classes.
* Invite students to join the discussion in a class.
* Track students’ questions to see which topics a student is less known about.
* Support real time notification.
* Support real time in discussion.

**For Students:**

* Create a post: articles, questions, answer.
* Manage learning materials: allowing students to create folders to store different kinds of studying documents.
* Support real time notification.
* Support real time in discussion.
* Follow a teacher.
* Send a request to join in a teacher’s class.
* Support an easy way to add mathematic equation, code highlighting besides other normal text formatting for editor.
* Suggest related posts.

## Role and Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Full Name** | **Role** | **Position** | **Contact** |
| **1** | Nguyễn Huy Hùng | Project Manager | Instructor | HungNH@fpt.edu.vn |
| **2** | Kha Hoàng Minh | Developer | Leader | MinhKHSE60979@fpt.edu.vn |
| **3** | Trương Như Khang | Developer | Member | KhangTNSE60992@fpt.edu.vn |
| **4** | Trần Đình Thi | Developer | Member | ThiTDSE61137@fpt.edu.vn |
| **5** | Tạ Đức Tùng | Developer | Member | TùngTD60416@fpt.edu.vn |

Table 2: Roles and Responsibilities

# Report No.2 Software Project Management Plan

## Problem Definition



### Name of this Capstone Project

* Q&A Platform for Educators (QAPE)

### Problem Abstract

Nowadays, in almost schools and university, the level of exchange knowledge between students and teachers are not much. A result of learning and teaching has not reached the highest efficiency with the following reasons:

Teaching hours of pretty little, much knowledge so that teachers cannot answer all questions of students during class, students fell shy to ask about their problems are not understood.

So building online Q&A system is urgently needed employment in the current educational context. This system will help teachers and students interact with each other through an application Website. Teachers can create their own classes are teaching on the Website and allow their student can join and post all question in that. Teacher can answer of student or invites another teacher can join and answer that question. They also can upload material and document for students that help them understand clearly the problem they do not understand.

.

### Project Overview

#### Current Situation

|  |  |  |
| --- | --- | --- |
| Website | Advantages | Disadvantages |
| Quora.com | -Can get answer by a prof.  -There are more Category  -User can edit question and edit answer from other user. | -Too much Category  -Everyone can answer question.  -Hard to find old answer.  -There is not real time discussion. Answer only. |
| Yahoo! Answer | -Can best answer by rating point  -Can ask everything: travel, book, films, social…. | -Everyone can answer question.  -No good interface design (too much word) |
| Stack Overflow | -Always pop Google search.  -So much data and load fast. | -No good design interface.  -There is not Category.  -There are much Basic question. |

Table 3 Current Situation

#### The Proposed System

From above disadvantages, this project is going to build a website that help teacher and student can communicate with each other easier and can get the answer from who what they want.

#### Website

**For Teachers:**

Create classes

Create a post. Post types include: articles, questions, answer, and upload material

Invite other teachers to answer questions in the classes

Invite students to join the discussion in a class

Track students’ questions to see which topics a student is less known about

Real time notification

Real time in discussion

**For Students:**

Create a post: articles, questions, answer

Manage learning materials: allowing students to create folders to store different kinds of studying documents.

Real time notification

Real time in discussion

Follow a teacher

Send a request to join in a teacher’s class.

Note: the editor must support an easy way to add mathematic equation, code highlighting besides other normal text formatting.

Suggest related posts

#### Boundaries of the System

This system is a Web application that helps student and teacher communicate with each other by Question and Answer.

There are three type of account: Student, Teacher, and Admin. They have different features.

#### Development Environment

##### Hardware requirements

**For server**

|  |  |  |
| --- | --- | --- |
| Windows | Minimum Requirements | Recommended |
| Internet Connection | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (8 Mbps) |
| Operating System | Window Server 2008 | Window Server 2008 |
| Computer Processor | Intel® Xeon ® 1.4GHz | Intel® Xeon ® Quad Core  (12M Cache, 2.50 GHz) |
| Computer Memory | 1GB RAM | 2GB or more |

Table 4: Hardware Requirement for Server

##### Software requirements

* Handle Products and sprints backlog: MS Excel 2013.
* Implement Website and service: IntelliJ v13.1.6.
* Create and manage database for system: MySQL v5.6.24 and MySQL Workbench v6.6.3.
* Source control: Github.
* Create models and diagrams: Software Ideas Modeler.
* Communicate and meeting: Skype 7.0 and Google mail.
* System and platform development: Window 7.

## Project organization

### Software Process Model



Figure 1: Scrum Development Model\*

Because of the project characteristics, we decide to make progress in series of sprints which are time boxed iterations in one week. At the beginning of one sprint, through sprint planning meeting, all team members will discuss together to define sprint backlog which is suitable to be completed within a week. Daily meeting and online conference are used to manage all activities and issues troubleshoot. At the end of sprint, product owner and developer team will review completed product to figuring out the necessary changes for products.

\*Reference from http://www.hanoiscrum.net

### Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Nguyễn Huy Hùng | Scrum Master/Product Owner | * Specify user requirement * Control the development process * Give out technique and business analysis support |
| **2** | Kha Hoàng Minh | Team Leader, BA, DEV, Tester | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **3** | Trương Như Khang | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **4** | Trần Đình Thi | DEV | * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **5** | Tạ Đức Tùng | DEV | * Prepare documents * GUI Design * Create test plan * Coding * Testing |

Table 5: Roles and Responsibilities Details

### Tools and Techniques

- Front-end technologies: HTML5, CSS3, JavaScript, jQuery, AJAX.

- Back-end technologies:

+Website: Java

+Framework: Spring.

- Web Server: Installed Tomcat 7.0.61.

- Database Management System: MySQL 5.6.

## Project Management Plan

### Product Backlog

|  |  |  |  |
| --- | --- | --- | --- |
| **Project: Q&A Platform for Educators** | | | |
|
| **ID** | **Function** | **Description** | **Added Sprint** |
| 1 | Register | Registering to become member of website |  |
| 2 | Login/Logout | Accessing the account to use their features |  |
| 3 | Get forgotten password | Resetting password if forgetting |  |
| 4 | Edit profile | Users edit their profile |  |
| 5 | Create class | Teachers create class for student's joining |  |
| 6 | Create article, question, answer | User post an article, ask a question or answer a question |  |
| 7 | Upload material | Teachers upload their studying documents |  |
| 8 | Invite teacher | Teachers invite other teachers to answer a question on |  |
| 9 | Invite student | Teachers invite students to join the discussion in class |  |
| 10 | Accept/Deny an invitation | Users accept/deny teachers' invitation |  |
| 11 | Accept/Deny a request | Teachers accept/deny a joining request from student |  |
| 12 | Track student | Teachers track students’ questions to see which topics a student is less known about |  |
| 13 | Manage post | Users manage their post (edit, remove post), Admin only can remove post. |  |
| 14 | Manage class | Teachers manage students in their class |  |
| 15 | Real time notify | System support real time notification when followed teachers have new posts or posts have new reply |  |
| 16 | Real time discuss | New answers or replies will show at real time without refreshing page |  |
| 17 | Manage learning materials | Students to create folders to store different kinds of studying documents |  |
| 18 | Follow teacher | Students follow their favorite teachers |  |
| 19 | Send a request | Students send a request to join in a teacher’s class. |  |
| 20 | Suggest related post | System suggest related posts (related articles, related questions) for users |  |
| 21 | Search | Users search posts or other users |  |
| 22 | Manage users | Admin can ban/unban users |  |

Table 6: Product Backlog Detail

### Sprint Backlog

#### Sprint 1

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Task** | **Implementer** | **Estimated hours** |
| 1 | Design homepage | MinhKH | 10 |
| 2 | Design register page | ThiTD | 6 |
| 3 | Design login form | ThiTD | 6 |
| 4 | Design get forget password page | ThiTD | 6 |
| 5 | Design student dashboard | TungTD | 8 |
| 6 | Design teacher dashboard | TungTD | 8 |
| 7 | Design admin dashboard | TungTD | 8 |
| 8 | Report 1 | ThiTD | 4 |
| 9 | Review report 1 | MinhKH | 3 |
| 10 | Setup environment turorial | KhangTN | 8 |
| 11 | Create overall use-case | KhangTN | 15 |
| 12 | Create guest use-cases | ThiTD | 10 |
| 12 | Create ERD | MinhKH | 12 |
| 13 | Create System Architectural Design | KhangTN | 10 |
| 14 | Report 2 | TungTD | 8 |
| 15 | Review report 2 | MinhKH | 3 |

Table 7: Sprint 1

### All Meeting Minutes

Place at Meeting Minute folder in Github.

## Coding Convention

Use Java coding convention to develop website.

Summary:

Naming Convention:

Use camel case for both variable and function name.

Use Pascal case for class name.

Indentation:

Avoid lines longer than 80 characters, since they are not handled well by many terminals and tools.

Declaration:

One declaration per line is recommended since it encourages commenting.

In absolutely no case should variables and functions be declared on the same line.

Do not put different types on the same line.

Code Examples:

Follow “Code Conventions for the Java TM Programming Language, by Sun Microsystems, rev April 20, 1999”.

<http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>

# Report No. 3 Software Requirement Specification

## User Requirement Specification

### Guest Requirement

Guest is a person who doesn’t access to the system. There are some functions guest can use:

* Register
* Login

### Student Requirement

Student is a person who accesses to the system with role “student”. There are functions student can use:

* Manage owned questions/articles/answers (create new, edit and delete)
* Request to join/Leave classrooms
* Manage materials (add material to folder, delete material in folder)
* Manage folders (create new, delete)
* Want answer the questions
* Accept answers
* Like/Unlike articles
* Follow/Un-follow teachers
* Edit profile
* Search
* Logout

### Teacher Requirement

Teacher is a person who accesses to the system with role “teacher”. There are functions teacher can use:

* Mange owned classrooms (create new, edit, close)
* Manage questions/articles/answers in owned classrooms (create new, edit, close)
* Manage owned questions/articles/answers (create new, edit, delete)
* Upload/Delete materials in owned classrooms
* Accept/Remove students in owned classrooms
* Invite students to join owned classrooms
* Invite teachers to answer questions in owned classrooms
* Merge questions in owned classrooms
* Answer multiple questions at once
* Want answer the questions
* Accept answers
* Like/Unlike articles
* Follow/Un-follow teachers
* Edit profile
* Search
* Logout

### Admin Requirement

Admin is a person who has the right of administration. There are functions admin can use:

* Manage articles/questions/answers (delete)
* Manage users (ban, un-ban)
* Manage category (create new, delete)

## System Requirement Specification

### External Interface Requirement

#### User Interface

* General requirement for graphics user interface is the GUI should be simple, clear, intuitive, and reminiscent.
* The interface design is an iterate process includes design, sketching, prototyping, user assessment.

#### Hardware Interface

* N/A

#### Software Interface

* Run with Chrome (Version 43.0.2357.132 m) and Cốc Cốc (Version 43.0.2357.126) browsers and must support JavaScript and html5.
* The screen of browsers must be at least 1024 x 768px to browse complete part of website.

#### Communication Protocol

* Using HTTP/HTTPS protocol

### System Overview Use Case

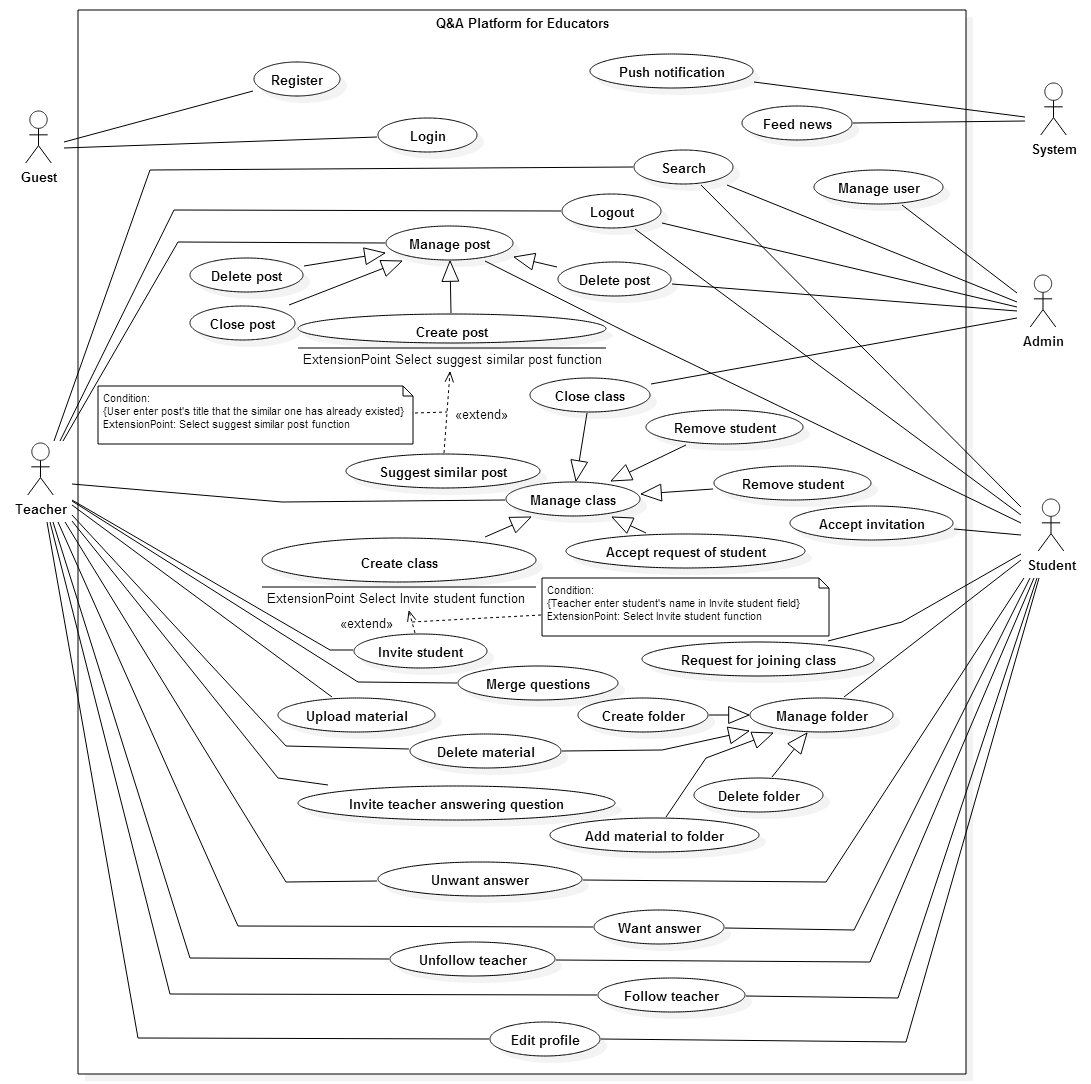
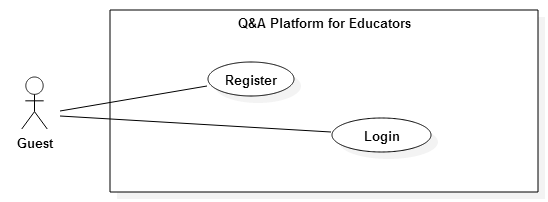


Figure 2: System Overview Use Case

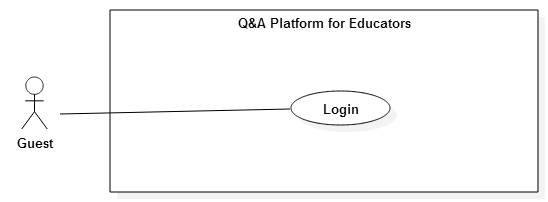
### List of Use Case

#### <Guest> Overview Use Case



**Figure 3: <Guest> Overview Use Case**

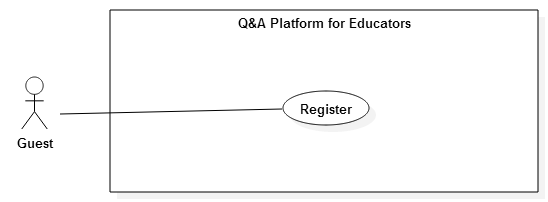
##### <Guest> Login Use Case Diagram

****

**Figure 4: <Guest> Login Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -1 SPECIFICATION** | | | |
| **Use-case No.** | QAPE001 | **Use-case Version** | 2.0 |
| **Use-case Name** | Login | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | High |
| **Actor:**   * Guest   **Summary:**   * This use case allows guest login to the system   **Goal:**   * Allows guest login to the system   **Triggers**   * Guest clicks Login button on Welcome screen.   **Preconditions:**   * N/A   **Post Conditions:**   * **On Success**: Guest logged in to the system. * **On Failure**: Show Login screen with error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 |  | System displays Welcome screen:  - “Email”: textbox, min length 8, max length 255.  - “Password”: textbox, min length: 8, max length 50.  -“Login”: button. | | 2 | Guest inputs email and password |  | | 3 | Click “Login” button | System navigates to News feed screen.  [Exceptions 1, 2, 3] |   **Alternative Scenario:**   * -N/A   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest lefts “Email” or “Password” textboxes blank. | Show login screen with message: “Invalid email or password”. | | 2 | Guest logins with wrong username or password. | Show login screen with message: “Invalid email or password”. | | 3 | Guest logins with non-existed account. | Show login screen with message: “Invalid email or password”. |   **Relationships:**   * N/A   **Business Rules:**   * Guest have to login with Student account, Teacher account or Admin account to access the web application. | | | |

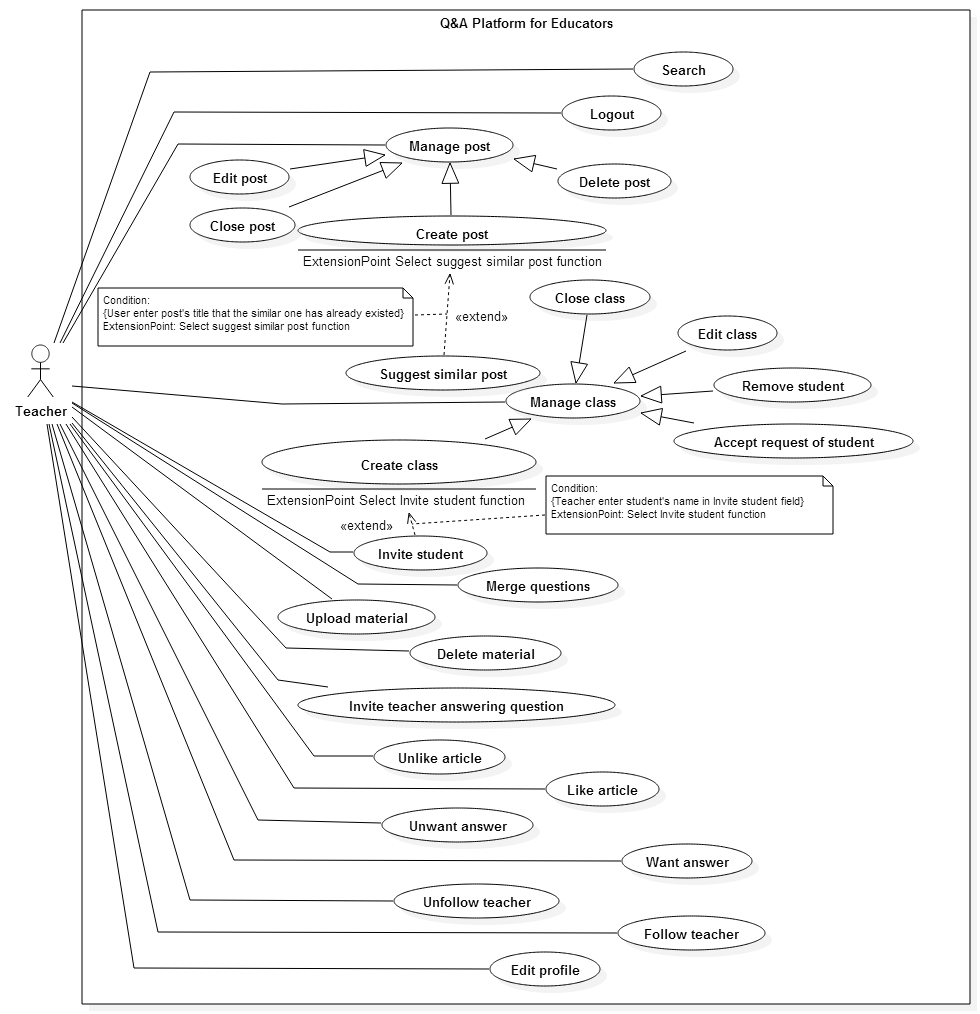
##### <Guest> Register Use Case Diagram



**Figure 5: <Guest> Register Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -2 SPECIFICATION** | | | |
| **Use-case No.** | QAPE002 | **Use-case Version** | 2.0 |
| **Use-case Name** | Register | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | High |
| **Actor:**   * Guest   **Summary:**   * This use case allows guest register an account   **Goal:**   * Allows guest login register an account   **Triggers**   * Guest clicks Register button on Welcome screen.   **Preconditions:**   * N/A   **Post Conditions:**   * **On Success**: New account is created and guest logged in to the system. * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 |  | System displays Welcome screen:  - “Email\*”: textbox, min length 8, max length 255, regular expression: “^[a-zA-Z0-9\_.+-]+@[a-zA-Z0-9-]+\.[a-zA-Z0-9-.]+$" required.  - “Password\*”: textbox, min length: 8, max length 50, required.  - “Confirm password\*”: textbox, min length: 8, max length 50, required.  - “Knowledge about\*”: drop down list, required.  - “You are\*”: select radio button, required.  -“Register”: button. | | 2 | Guest inputs all required fields |  | | 3 | Click “Register” button | New account is created and system navigates to News feed screen.  [Exceptions 1, 2, 3, 4, 5]  [Alternative 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest enters existed email address and clicks Register button | Show message: “This email has been used”. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest lefts “Email\*” blank or enters one not matched regular expression | Show error message: “Please enter a valid email address.” | | 2 | Guest lefts “Password\*” or “Confirm password\*” blank | Show error message: “Please provide a password. | | 3 | Guest enters “Password\*” or “Confirm password\*” is not matched min and max length | Show error message: “Your password must be between 8 and 50 characters long.” | | 4 | Guest enters “Password\*” and “Confirm password\*” are not matched | Show error message: “Please enter the same password as above.” | | 5 | Guest lefts “Knowledge about\*” as default | Show error message: “Please choose your knowledge.” |   **Relationships:**   * N/A   **Business Rules:**   * All required fields must be filled. * Email address must be unique. * Password and confirm password must be same. | | | |

#### <Teacher> Overview Use Case

****

**Figure 6: <Teacher> Overview Use Case**

##### <Teacher> Create class Use Case Diagram

****

**Figure 7: <Teacher> Create class Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -3 SPECIFICATION** | | | |
| **Use-case No.** | QAPE003 | **Use-case Version** | 2.0 |
| **Use-case Name** | Create class | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | High |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher create the classroom and invite student to join classroom   **Goal:**   * Create new classroom   **Triggers**   * Teacher clicks “Create class” button in Dashboard screen.   **Preconditions:**   * User must login as Teacher role.   **Post Conditions:**   * **On Success**: New classroom is created. * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Dashboard screen and clicks “Create class” button | System displays Create classroom screen:  - “Class name\*”: textbox, min length 20, max length 255, required.  - “Tag\*”: textbox, required.  - “Invite student”: textbox.  - “Category\*”: drop down list, required.  - “Class’s Description\*”: text-area, , min length 20, max length 350, required.  - “Create class”: button. | | 2 | Teacher inputs all required fields |  | | 3 | Click “Create class” button | New classroom created and system navigates to classroom screen.  [Exceptions 1, 2, 3, 4, 5, 6] |   **Alternative Scenario:**   * N/A   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher lefts “Class name\*” blank | Show error message: “Please provide the class's name.” | | 2 | Teacher enters “Class name\*” is not match min and max length | Show error message: “The class name must be between 20 and 255 characters long.” | | 3 | Teacher lefts “Tag\*” blank | Show error message: “Please provide at least one tag.” | | 4 | Teacher lefts “Category\*” as default | Show error message: “Please select category.” | | 5 | Teacher lefts “Class’s description\*” blank | Show error message: “Please provide the description.” | | 6 | Teacher enters “Class’s description\*” is not match min and max length | Show error message: “The description must be between 20 and 255 characters long.” |   **Relationships:**   * Extend for Invite student (Select invite student function)   **Business Rules:**   * All required fields must be filled. * Number of Tag is not over 5. * Duplicated tags in Tag or students in Invite student do not allow. * Invite student can be blank. | | | |

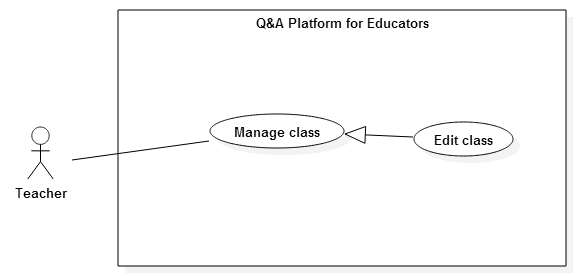
##### <Teacher> Invite student Use Case Diagram

****

**Figure 7: <Teacher> Invite student Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -4 SPECIFICATION** | | | |
| **Use-case No.** | QAPE004 | **Use-case Version** | 2.0 |
| **Use-case Name** | Invite student | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | High |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher invite students to join classroom   **Goal:**   * Invite students to join classroom   **Triggers**   * Teacher clicks “Invite” button in Classroom screen. * Teacher enters student’s name in Invite student when Create classroom.   **Preconditions:**   * User must login as Teacher role. * In case of invite student in Classroom screen, classroom must be already created, in “open” state and owned by teacher.   **Post Conditions:**   * **On Success**: Students are invited received invitation. * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks “Invite” button | System displays popup “Invite student to join class”:  - “Student\*”: textbox, required  - “Invite”: button.  - “x” icon: button.  [Alternative 1] | | 2 | Teacher inputs student’s name | System show drop down list with suggest students base on student’s name | | 3 | Teacher chooses student from drop down list |  | | 4 | Click “Invite” button | Popup is closed and invited students receive invitations.  [Alternative 2]  [Exceptions 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks “x” icon | Popup is closed and no student is invited | | 2 | Teacher types in text box instead of chooses from dropdown list | Popup is closed and no student is invited |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher lefts “Student\*” blank | Show error message: “Please provide at least one student.” |   **Relationships:**   * Extend by Create class   **Business Rules:**   * Teacher must choose student from suggest drop down list instead of types text. * Suggested students must be active accounts. | | | |

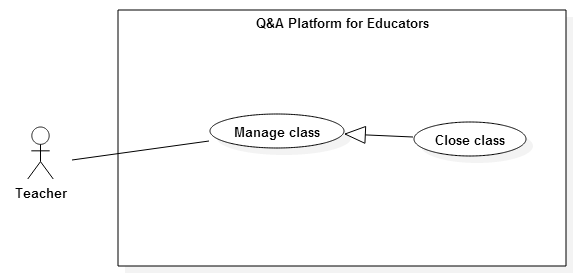
##### <Teacher> Edit class Use Case Diagram

****

**Figure 9: <Teacher> Edit class Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -5 SPECIFICATION** | | | |
| **Use-case No.** | QAPE005 | **Use-case Version** | 2.0 |
| **Use-case Name** | Edit class | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | Medium |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher update classroom’s information   **Goal:**   * Update classroom’s information   **Triggers**   * Teacher clicks setting icon and clicks “Update class” in Classroom screen.   **Preconditions:**   * User must login as Teacher role. * Classroom must be in “open” state and owned by teacher.   **Post Conditions:**   * **On Success**: Classroom’s information is updated * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks setting icon | System displays drop down list | | 2 | Teacher clicks “Update class”  [Alternative 1] | System displays Update classroom screen:  - “Tag\*”: textbox, required.  - “Category\*”: drop down list, required.  - “Class’s Description\*”: text-area, , min length 20, max length 350, required.  - “Update class”: button. | | 3 | Teacher updates classroom’s information |  | | 4 | Click “Update class” button | Classroom’s information is updated and system navigates to classroom screen.  [Exceptions 1, 2, 3, 4] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks different place in page or clicks setting icon again | Drop down list is hidden |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher lefts “Tag\*” blank | Show error message: “Please provide at least one tag.” | | 2 | Teacher lefts “Category\*” as “Select a type” | Show error message: “Please select category.” | | 3 | Teacher lefts “Class’s description\*” blank | Show error message: “Please provide the description.” | | 4 | Teacher enters “Class’s description\*” is not match min and max length | Show error message: “The description must be between 20 and 255 characters long.” |   **Relationships:**   * N/A   **Business Rules:**   * All required fields must be filled. * Number of Tag is not over 5. * Teacher cannot input duplicated tags. * Teacher cannot edit classroom’s name. | | | |

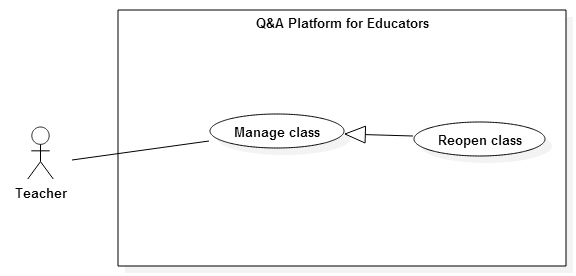
##### <Teacher> Close class Use Case Diagram

****

**Figure 10: <Teacher> Close class Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -6 SPECIFICATION** | | | |
| **Use-case No.** | QAPE006 | **Use-case Version** | 2.0 |
| **Use-case Name** | Close class | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | Medium |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher close classroom   **Goal:**   * Close the classroom to prevent more request, question, article or material   **Triggers**   * Teacher clicks setting icon and clicks “Close class” in Classroom screen.   **Preconditions:**   * User must login as Teacher role. * Classroom must be in “open” state and owned by teacher.   **Post Conditions:**   * **On Success**: Classroom status is changed to close * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks setting icon | System displays drop down list | | 2 | Teacher clicks “Close class”  [Alternative 1] | Classroom status is changed to close  System displays Classroom screen with no “Create post” button and show “Classroom is closed” in “About classroom” widget |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks different place in page or clicks setting icon again | Drop down list is hidden. Nothing change |   **Exceptions:**   * N/A   **Relationships:**   * N/A   **Business Rules:**   * Student cannot create new article/question in closed classroom * Student cannot request to join closed classroom * Teacher cannot edit classroom information when it is closed * Teacher can manage question, article, material, request, student in closed classroom except create new question or article, upload material and invite student. | | | |

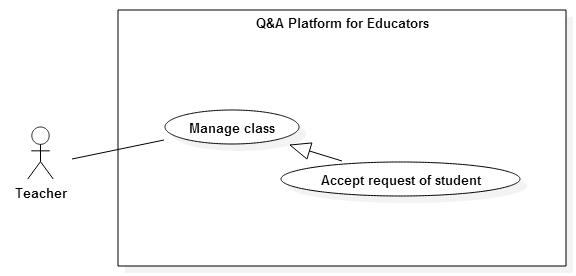
##### <Teacher> Reopen class Use Case Diagram

****

**Figure 11: <Teacher> Reopen class Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -7 SPECIFICATION** | | | |
| **Use-case No.** | QAPE007 | **Use-case Version** | 2.0 |
| **Use-case Name** | Reopen class | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | Medium |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher reopen closed classroom   **Goal:**   * Reopen closed classroom   **Triggers**   * Teacher clicks setting icon and clicks “Reopen class” in Classroom screen.   **Preconditions:**   * User must login as Teacher role. * Classroom must be in “close” state and owned by teacher.   **Post Conditions:**   * **On Success**: Classroom status is changed to open * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks setting icon | System displays drop down list | | 2 | Teacher clicks “Reopen class”  [Alternative 1] | System displays Classroom screen with full classroom management functions |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks different place in page or clicks setting icon again | Drop down list is hidden. Nothing change |   **Exceptions:**   * N/A   **Relationships:**   * N/A   **Business Rules:**   * Teacher has full management functions with owned opened classroom * Student can send request to join opened classroom | | | |

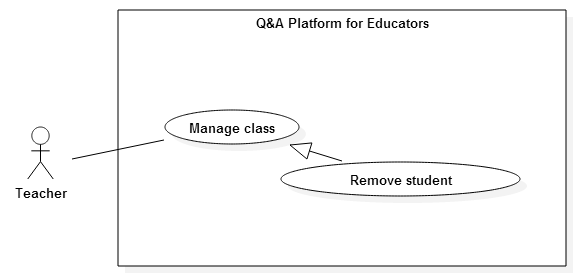
##### <Teacher> Accept request of student Use Case Diagram

****

**Figure 12: <Teacher> Accept request of student Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -8 SPECIFICATION** | | | |
| **Use-case No.** | QAPE008 | **Use-case Version** | 2.0 |
| **Use-case Name** | Accept request of student | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | High |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher accept joining requests of students   **Goal:**   * Accept students’ requests   **Triggers**   * Teacher clicks Confirm button in Join Request tab.   **Preconditions:**   * User must login as Teacher role. * Classroom must be owned by teacher and have at least one joining request from student.   **Post Conditions:**   * **On Success**: Request is removed and student is added to classroom * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks tab “Join Request” | System displays list of requests from students. | | 2 | Teacher clicks “Confirm” button  [Alternative 1] | The request in list changes to “You have confirmed the request of <student’s name>” and this student is added to classroom. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks “Ignore” button | The request changes to “You have ignored the request of <student’s name>” and the request is removed |   **Exceptions:**   * N/A   **Relationships:**   * N/A   **Business Rules:**   * Teacher can confirm or ignore only with students’ requests to joined owned classrooms * Teacher cannot confirm the request after ignoring * Teacher can invite students after ignoring their requests | | | |

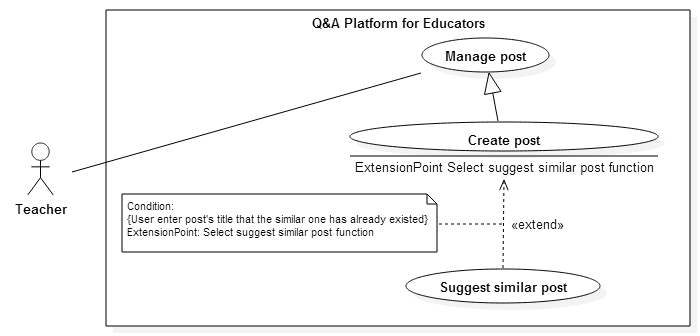
##### <Teacher> Remove student Use Case Diagram

****

**Figure 13: <Teacher> Remove student Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -9 SPECIFICATION** | | | |
| **Use-case No.** | QAPE009 | **Use-case Version** | 2.0 |
| **Use-case Name** | Remove student | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | High |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher remove students from classroom   **Goal:**   * Remove students   **Triggers**   * Teacher clicks “Remove” button in Student tab.   **Preconditions:**   * User must login as Teacher role. * Classroom must be owned by teacher and have at least one student.   **Post Conditions:**   * **On Success**: Student is removed from classroom * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks tab “Student” | System displays list of students in classroom. | | 2 | Teacher clicks “Remove” button | System displays “Remove student” popup:  - “OK”: button  - “Cancel”: button | | 3 | Teacher clicks “OK” button  [Alternative 1] | The student in list changes to “You have remove <student’s name> from classroom.” and student is removed from classroom |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks “Cancel” button | Popup disappears and no student is removed. |   **Exceptions:**   * N/A   **Relationships:**   * N/A   **Business Rules:**   * Teacher can only remove students in owned classrooms. * Teacher cannot add students after removing them. * Teacher can invite students after removing them. | | | |

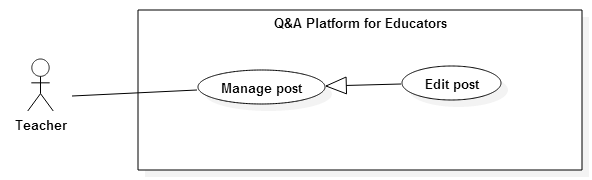
##### <Teacher> Create post Use Case Diagram

****

**Figure 14: <Teacher> Create post Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE 10 SPECIFICATION** | | | |
| **Use-case No.** | QAPE010 | **Use-case Version** | 2.0 |
| **Use-case Name** | Create post | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 16/07/2015 | **Priority** | High |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher create new question or article in owned classroom   **Goal:**   * Create new question or article   **Triggers**   * Teacher clicks “Create post” button in Classroom screen.   **Preconditions:**   * User must login as Teacher role. * Teacher is owner of classroom * Classroom is “open” state   **Post Conditions:**   * **On Success**: New question or article is created. * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks “Create post” button | System displays Create post screen:  - “Title\*”: textbox, min length 20, max length 255, required.  - “Tag\*”: textbox, required.  - “Type\*”: drop down list, required.  - “Detail\*”: text-area, min length 120, required.  - “Publish your post”: button. | | 2 | Teacher inputs all required fields | After inputting “Title\*”, system suggest posts have similar titles  [Alternative 1] | | 3 | Click “Publish your post” button  [Alternative 2] | New article/question (base on Type) is created and system navigates to article/question screen.  [Exceptions 1, 2, 3, 4, 5, 6] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 |  | Nothing is suggested when there are no post have similar title | | 2 | Teacher clicks post’s title in suggested list. | System opens new tab in browser and show this post |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher lefts “Title\*” blank | Show error message: “Please provide the title.” | | 2 | Teacher enters “Title\*” is not match min and max length | Show error message: “The class name must be between 20 and 255 characters long.” | | 3 | Teacher lefts “Tag\*” blank | Show error message: “Please provide at least one tag.” | | 4 | Teacher lefts “Type\*” as default | Show error message: “Please choose question or article.” | | 5 | Teacher lefts “Details\*” blank | Show error message: “Please provide the details.” | | 6 | Teacher enters “Details \*” is not match min length | Show error message: “The details must be at least 120 characters long.” |   **Relationships:**   * Extend for suggest similar post (Select suggest similar post function)   **Business Rules:**   * Post is only created if classroom is open. * Teacher can only create post in owned classroom. * All required fields must be filled. * Number of Tag is not over 5. * Teacher cannot input duplicated tags. | | | |

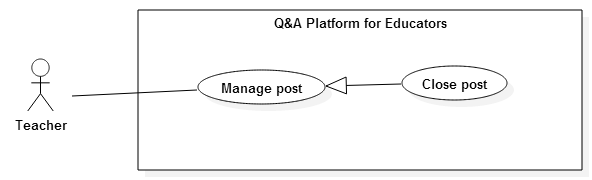
##### <Teacher> Edit post Use Case Diagram

****

**Figure 15: <Teacher> Edit post Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE 11 SPECIFICATION** | | | |
| **Use-case No.** | QAPE011 | **Use-case Version** | 2.0 |
| **Use-case Name** | Edit post | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 16/07/2015 | **Priority** | Medium |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher edit owned question or article   **Goal:**   * Edit question or article   **Triggers**   * Teacher clicks setting icon and clicks “Edit” button in Question or Article screen.   **Preconditions:**   * User must login as Teacher role. * Teacher is owner of question or article. * Article or question in “Open” state.   **Post Conditions:**   * **On Success**: Question or Article is updated * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks “Create post” button | System displays Update post screen:  - “Title\*”: textbox, min length 20, max length 255, required.  - “Tag\*”: textbox, required.  - “Type\*”: drop down list, required.  - “Detail\*”: text-area, min length 120, required.  - “Save”: button. | | 2 | Teacher inputs all required fields |  | | 3 | Click “Save” button | Question/Article is updated and system navigates to question/article screen  [Exceptions 1, 2, 3, 4, 5, 6] |   **Alternative Scenario:**   * N/A   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher lefts “Title\*” blank | Show error message: “Please provide the title.” | | 2 | Teacher enters “Title\*” is not match min and max length | Show error message: “The class name must be between 20 and 255 characters long.” | | 3 | Teacher lefts “Tag\*” blank | Show error message: “Please provide at least one tag.” | | 4 | Teacher lefts “Type\*” as default | Show error message: “Please choose question or article.” | | 5 | Teacher lefts “Details\*” blank | Show error message: “Please provide the details.” | | 6 | Teacher enters “Details \*” is not match min length | Show error message: “The details must be at least 120 characters long.” |   **Relationships:**   * N/A   **Business Rules:**   * Teacher can only edit owned post. * All required fields must be filled. * Number of Tag is not over 5. * Teacher cannot input duplicated tags. | | | |

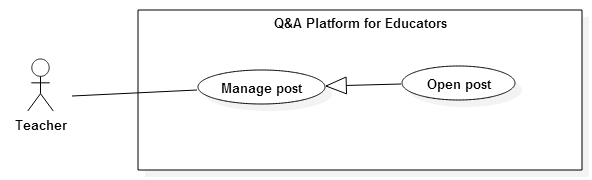
##### <Teacher> Close post Use Case Diagram

****

**Figure 16: <Teacher> Close post Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE 12 SPECIFICATION** | | | |
| **Use-case No.** | QAPE012 | **Use-case Version** | 2.0 |
| **Use-case Name** | Close post | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 16/07/2015 | **Priority** | Medium |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher close question or article   **Goal:**   * Close question or article to prevent more answers   **Triggers**   * Teacher clicks setting icon and clicks “Close” in Question or Article screen.   **Preconditions:**   * User must login as Teacher role. * Question or article is in teacher’s owned classroom. * Article or question in “Open” state.   **Post Conditions:**   * **On Success**: Question or Article is closed * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Question/Article screen and clicks setting icon. | System displays drop down list. | | 2 | Teacher clicks “Close”.  [Alternative 1] | System displays question/article screen with no “Leave a reply” area.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks different place in page or clicks setting icon again. | Drop down list is hidden. Nothing is changed. | | 2 |  | If post is question, “Ask teacher” widget will be hidden too. |   **Exceptions:**   * N/A   **Relationships:**   * N/A   **Business Rules:**   * Teacher cannot leave answer in closed post or edit closed post. * Teacher can delete closed owned post. * Teacher can close all posts in owned classroom | | | |

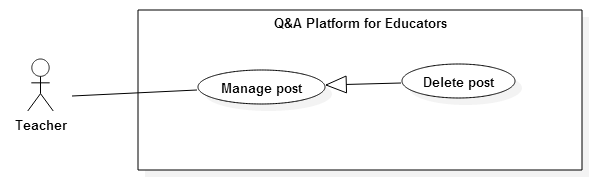
##### <Teacher> Open post Use Case Diagram

****

**Figure 17: <Teacher> Open post Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE 13 SPECIFICATION** | | | |
| **Use-case No.** | QAPE013 | **Use-case Version** | 2.0 |
| **Use-case Name** | Open post | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 16/07/2015 | **Priority** | Medium |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher open question or article   **Goal:**   * Open question or article   **Triggers**   * Teacher clicks setting icon and clicks “Open” in Question or Article screen.   **Preconditions:**   * User must login as Teacher role. * Question or article is in teacher’s owned classroom. * Article or question in “Close” state.   **Post Conditions:**   * **On Success**: Question or Article is opened * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Question/Article screen and clicks setting icon. | System displays drop down list. | | 2 | Teacher clicks “Open”.  [Alternative 1] | System displays question/article screen with “Leave a reply” area.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks different place in page or clicks setting icon again. | Drop down list is hidden. Nothing is changed. | | 2 |  | If post is question, “Ask teacher” will be display too. |   **Exceptions:**   * N/A   **Relationships:**   * N/A   **Business Rules:**   * Teacher can reply or if post is question, teacher can also invite other teachers to answer it. | | | |

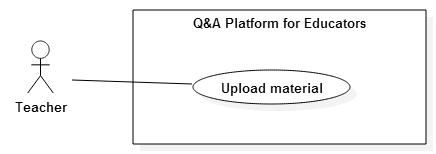
##### <Teacher> Delete post Use Case Diagram

****

**Figure 18: <Teacher> Delete post Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE 14 SPECIFICATION** | | | |
| **Use-case No.** | QAPE014 | **Use-case Version** | 2.0 |
| **Use-case Name** | Delete post | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 16/07/2015 | **Priority** | Medium |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher delete owned question or article   **Goal:**   * Delete question or article   **Triggers**   * Teacher clicks setting icon and clicks “Delete” button in Question or Article screen.   **Preconditions:**   * User must login as Teacher role. * Teacher is owner of question or article. * Article or question in “Open” state.   **Post Conditions:**   * **On Success**: Question or Article is deleted * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Question/Article screen and clicks setting icon | System displays drop down list | | 2 | Teacher clicks “Delete” | System displays “Delete post” popup:  - “OK”: button  - “Cancel”: button | | 3 | Teacher clicks “OK” button  [Alternative 1] | Post is removed and system navigates to post’s owned classroom screen. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks “Cancel” button | Popup disappears and no student is removed. |   **Exceptions:**   * N/A   **Relationships:**   * N/A   **Business Rules:**   * Teacher cannot leave answer in closed post or edit closed post. * Teacher cannot delete students’ posts. * Teacher can delete closed owned post. | | | |

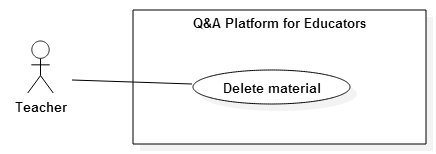
##### <Teacher> Upload material Use Case Diagram

****

**Figure 19: <Teacher> Upload material Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -15 SPECIFICATION** | | | |
| **Use-case No.** | QAPE015 | **Use-case Version** | 2.0 |
| **Use-case Name** | Upload material | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 16/07/2015 | **Priority** | Medium |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher upload material in owned classroom   **Goal:**   * Upload material   **Triggers**   * Teacher clicks Upload button in Material tab.   **Preconditions:**   * User must login as Teacher role. * Classroom must be in “open” state and owned by teacher.   **Post Conditions:**   * **On Success**: New material is uploaded * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks tab “Material” | System displays list of materials in classroom. | | 2 | Teacher clicks “Upload” button | System displays “Upload material” popup:  - “Tag\*”: textbox, required.  - “x” icon: button  - “Choose File”: button  - “Upload”: button | | 3 | Teacher inputs “Tag\*”  [Alternative 1] |  | | 4 | Teacher clicks “Choose file” and select file from computer | System displays name of selected material. | | 5 | Teacher clicks “Upload” button | Material is uploaded and systems show it in material list  [Exception 1, 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks “x” icon button | Popup disappears and no material is uploaded. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher lefts “Tag\*” blank | Show error message: “Please provide at least one tag.” | | 2 | Teacher doesn’t choose material to upload | Show error message: “Please choose material.” |   **Relationships:**   * N/A   **Business Rules:**   * Teacher can only upload material in owned opened classrooms. * Teacher can only upload one material at time. * Number of Tag is not over 5. * Duplicated tags in Tag do not allow. | | | |

##### <Teacher> Delete material Use Case Diagram

****

**Figure 13: <Teacher> Delete material Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -16 SPECIFICATION** | | | |
| **Use-case No.** | QAPE016 | **Use-case Version** | 2.0 |
| **Use-case Name** | Delete material | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | High |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher delete material in owned classroom   **Goal:**   * Delete materials   **Triggers**   * Teacher clicks “Delete” button in Material tab.   **Preconditions:**   * User must login as Teacher role. * Classroom must be owned by teacher and have at least one material.   **Post Conditions:**   * **On Success**: Material is deleted * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Classroom screen and clicks tab “Material” | System displays list of materials in classroom. | | 2 | Teacher clicks “Delete” button | System displays “Delete material” popup:  - “OK”: button  - “Cancel”: button | | 3 | Teacher clicks “OK” button  [Alternative 1] | Material is deleted and disappears in material list |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks “Cancel” button | Popup disappears and no material is deleted. |   **Exceptions:**   * N/A   **Relationships:**   * N/A   **Business Rules:**   * Teacher can only delete materials in owned classrooms. * Deleted material is only deleted from classrooms; it’s still available in student’s folder if it have already added to. | | | |

##### <Teacher> Invite teacher answering question Use Case Diagram

****

**Figure 7: <Teacher> Invite teacher answering question Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE -17 SPECIFICATION** | | | |
| **Use-case No.** | QAPE017 | **Use-case Version** | 2.0 |
| **Use-case Name** | Invite teacher answering question | | |
| **Author** | Kha Hoang Minh | | |
| **Date** | 15/07/2015 | **Priority** | High |
| **Actor:**   * Teacher   **Summary:**   * This use case allows teacher invite another teachers to join into discussion in question   **Goal:**   * Invite teacher to answering question   **Triggers**   * Teacher clicks “Ask teacher” button in Question screen.   **Preconditions:**   * User must login as Teacher role. * Question must be in “Open” state and in teacher’s owned classroom.   **Post Conditions:**   * **On Success**: Teachers are invited received invitation and can discuss in question. * **On Failure**: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Teacher goes to Question screen and clicks “Ask teacher” button | System displays popup “Ask teacher student to join class”:  - “Teacher\*”: textbox, required  - “Invite”: button.  - “x” icon: button.  [Alternative 1] | | 2 | Teacher inputs student’s name | System show drop down list with suggest teachers base on teacher’s name | | 3 | Teacher chooses another teacher from drop down list |  | | 4 | Click “Invite” button | Popup is closed, invited teachers receive invitations and they can answering this question.  [Alternative 2]  [Exceptions 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher clicks “x” icon | Popup is closed and no student is invited | | 2 | Teacher types in text box instead of chooses from dropdown list | Popup is closed and no student is invited |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Teacher lefts “Student\*” blank | Show error message: “Please provide at least one student.” |   **Relationships:**   * Extend by Create class   **Business Rules:**   * Teacher must choose student from suggest drop down list instead of types text. * Suggested students must be active accounts. | | | |