**MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Office Task Management**

|  |  |
| --- | --- |
| **Group 3** | |
| **Group member** | Nguyen Cu Dat – SE61010  Lu Cam Toan – SE60633  Le Hai Trieu - SE61146  Tran Thi Kim Anh – SE60909 |
| **Supervisor** | Mr. Lai Duc Hung |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | OTM |

-Ho Chi Minh City, 13/05/2015-

# Table of Contents

[Table of Contents 3](#_Toc421276935)

[List of Tables 6](#_Toc421276936)

[List of Figures 7](#_Toc421276937)

[Definitions, Acronyms, and Abbreviations 8](#_Toc421276938)

[A. Introduction 9](#_Toc421276939)

[1. Project Information 9](#_Toc421276940)

[2. Introduction 9](#_Toc421276941)

[3. Current Situation 9](#_Toc421276942)

[4. Problem Definition 9](#_Toc421276943)

[5. Proposed Solution 9](#_Toc421276944)

[5.1 Feature functions 10](#_Toc421276945)

[5.2 Advantage and disadvantage 10](#_Toc421276946)

[6. Functional Requirement 10](#_Toc421276947)

[6.1 User Management 10](#_Toc421276948)

[6.2 Tracking task 10](#_Toc421276949)

[6.3 Review and approve document 10](#_Toc421276950)

[6.4 Collaborate in real-time 10](#_Toc421276951)

[6.5 Statistic 10](#_Toc421276952)

[7. Role and Responsibility 10](#_Toc421276953)

[B. Software Project Management Plan 12](#_Toc421276954)

[1. Problem Definition 12](#_Toc421276955)

[1.1 Name of this Capstone project 12](#_Toc421276956)

[1.2 Problem Abstract 12](#_Toc421276957)

[1.3 Project Overview 12](#_Toc421276958)

[2. Project organization 13](#_Toc421276959)

[2.1 Software Process Model 14](#_Toc421276960)

[2.2 Roles and responsibilities 14](#_Toc421276961)

[2.3 Tools and Techniques 15](#_Toc421276962)

[3. Project Management Plan 15](#_Toc421276963)

[3.1 Software development life cycle 15](#_Toc421276964)

[3.2 Phase Detail 17](#_Toc421276965)

[3.3 All Meeting Minutes 18](#_Toc421276966)

[4. Coding Convention 18](#_Toc421276967)

[C. Software Requirement Specification 19](#_Toc421276968)

[1. User Requirement Specification 19](#_Toc421276969)

[1.1 Admin Requirement 19](#_Toc421276970)

[1.2 Director Requirement 19](#_Toc421276971)

[1.3 Manager Requirement 19](#_Toc421276972)

[1.4 Staff Requirement 20](#_Toc421276973)

[2. System Requirement Specification 20](#_Toc421276974)

[2.1 External Interface Requirement 20](#_Toc421276975)

[2.2 System Overview Use Case 21](#_Toc421276976)

[2.3 List of Use Case 21](#_Toc421276977)

[3. Software System Attribute 51](#_Toc421276978)

[3.1 Usability 51](#_Toc421276979)

[3.2 Reliability 51](#_Toc421276980)

[3.3 Availability 51](#_Toc421276981)

[3.4 Security 51](#_Toc421276982)

[3.5 Maintainability 51](#_Toc421276983)

[3.6 Portability 51](#_Toc421276984)

[3.7 Performance 51](#_Toc421276985)

[4. Conceptual Diagram 52](#_Toc421276986)

[D. Software Design Description 53](#_Toc421276987)

[1. Design Overview 53](#_Toc421276988)

[2. System Architectural Design 53](#_Toc421276989)

[2.1 Web application architecture description 54](#_Toc421276990)

[2.2 Mobile application architecture description 54](#_Toc421276991)

[3. Component Diagram 54](#_Toc421276992)

[4. Detailed Description 54](#_Toc421276993)

[4.1 Class Diagram 54](#_Toc421276994)

[4.2 Class Diagram Explanation 55](#_Toc421276995)

[4.3 Interaction Diagram 55](#_Toc421276996)

[5. User Interface Design 57](#_Toc421276997)

[5.1 Guest Interface Design 57](#_Toc421276998)

[6. Database Design 58](#_Toc421276999)

[6.1 Logical Diagram 58](#_Toc421277000)

[6.2 Data Dictionary 58](#_Toc421277001)

[7. Algorithms 59](#_Toc421277002)

[7.1 Document Breakdown 59](#_Toc421277003)

[7.2 String Comparison 60](#_Toc421277004)

[E. System Implementation & Test 61](#_Toc421277005)

[1. Introduction 61](#_Toc421277006)

[1.1 Overview 61](#_Toc421277007)

[1.2 Test Approach 63](#_Toc421277008)

[2. Database Relationship Diagram 63](#_Toc421277009)

[2.1 Physical Diagram 63](#_Toc421277010)

[2.2 Data Dictionary 63](#_Toc421277011)

[3. Performance Measures 63](#_Toc421277012)

[3.1 Clustering Performance 64](#_Toc421277013)

[4. Test Plan 64](#_Toc421277014)

[4.1 Features to be tested 64](#_Toc421277015)

[4.2 Features not to be tested 64](#_Toc421277016)

[5. System Testing Test Case 64](#_Toc421277017)

[5.1 Guest Test Case 66](#_Toc421277018)

[F. Software User’s Manual 66](#_Toc421277019)

[1. Installation Guide 66](#_Toc421277020)

[1.1 Setting up environment at server side 66](#_Toc421277021)

[1.2 Deployment at server side 67](#_Toc421277022)

[1.3 Setting up the environment at client side 67](#_Toc421277023)

[2. User Guide 67](#_Toc421277024)

[G. Appendix 67](#_Toc421277025)

# List of Tables

[Table 1: Role and Responsibility 12](#_Toc421283525)

[Table 2: <Hardware Requirements> Server 14](#_Toc421283526)

[Table 3: <Hardware Requirement> Mobile 14](#_Toc421283527)

[Table 4: Role and Reponsibility (2) 16](#_Toc421283528)

[Table 5: Software Deverlopment Life Cycle 18](#_Toc421283529)

[Table 6: <Phase Detail> Phase 1 18](#_Toc421283530)

[Table 7: <Phase Detail> Phase 2 18](#_Toc421283531)

[Table 8: <Phase Detail> Phase 3 19](#_Toc421283532)

[Table 9: <Phase Detail> Phase 4 19](#_Toc421283533)

[Table 10: <Phase Detail> Phase 5 19](#_Toc421283534)

[Table 11: <Phase Detail> Phase 6 19](#_Toc421283535)

[Table 12 <User> Login 24](#_Toc421283536)

[Table 13: <Admin> Reset Password 24](#_Toc421283537)

[Table 14: <Admin> Edit Profile 24](#_Toc421283538)

[Table 15: <Admin> Logout 24](#_Toc421283539)

[Table 16: <Admin> Search Accounts 24](#_Toc421283540)

[Table 17: <Admin> Add Account 24](#_Toc421283541)

[Table 18: <Admin> Edit Account 24](#_Toc421283542)

[Table 19: <Admin> Activate/Deactivate Account 24](#_Toc421283543)

[Table 20: <Admin> Search List Skills 24](#_Toc421283544)

[Table 21: <Admin> Add Skill List 24](#_Toc421283545)

[Table 22: <Admin> Update Skill List 24](#_Toc421283546)

[Table 23: <Admin> Add Category 24](#_Toc421283547)

[Table 24: <Admin> Search Categories 24](#_Toc421283548)

[Table 25: <Admin> Update Category 24](#_Toc421283549)

[Table 26: <Admin> Create Period Assessment 24](#_Toc421283550)

[Table 27: <Admin> Search Period Assessment 24](#_Toc421283551)

[Table 28: <Admin> Edit Period Assessment 24](#_Toc421283552)

[Table 29: <Admin> Create Criterion 24](#_Toc421283553)

[Table 30: <Admin> Edit Criterion 24](#_Toc421283554)

[Table 31: <Admin> Search Criterions 24](#_Toc421283555)

[Table 32: <Director> Add Manager into Project 24](#_Toc421283556)

[Table 33: <Director> Remove Manager form Project 24](#_Toc421283557)

[Table 34: <Director> Create Project 24](#_Toc421283558)

[Table 35: <Director> Update Project 24](#_Toc421283559)

[Table 36: <Director> Close Project 24](#_Toc421283560)

[Table 37: <Director> Remove Project 24](#_Toc421283561)

[Table 38: <Director> View Productivity Assessment 24](#_Toc421283562)

[Table 39: <Manager> Add Staffs into Project 24](#_Toc421283563)

[Table 40: <Manager> Remove Staffs from Project 24](#_Toc421283564)

[Table 41: <Manager> Create Task 24](#_Toc421283565)

[Table 42: <Manager> Assign Task 24](#_Toc421283566)

[Table 43: <Manager> Task Review 24](#_Toc421283567)

[Table 44: <Manager> Edit Task 24](#_Toc421283568)

[Table 45: <Manager> Remove Task 24](#_Toc421283569)

[Table 46: <Manager> View Project Details 24](#_Toc421283570)

[Table 47: <Manager> Search Assessment 24](#_Toc421283571)

[Table 48: <Manager> Add Assessment 24](#_Toc421283572)

[Table 49: <Manager> Edit Assessment 24](#_Toc421283573)

[Table 50:<Staff> Search Topic 24](#_Toc421283574)

[Table 51: <Staff> Add Topic 24](#_Toc421283575)

[Table 52: <Staff> Edit Topic 24](#_Toc421283576)

[Table 53: <Staff> Close Topic 24](#_Toc421283577)

[Table 54: <Staff> View Report Status of Work 24](#_Toc421283578)

[Table 55: <Staff> Search Tasks 24](#_Toc421283579)

[Table 56: <Staff> View Employee Assessments 24](#_Toc421283580)

[Table 57: <Staff> View All Tasks 24](#_Toc421283581)

[Table 58: <Staff> View Task Detail 24](#_Toc421283582)

[Table 59: <Staff> Search Project 24](#_Toc421283583)

[Table 60: <Staff> Change Task Status 24](#_Toc421283584)

# List of Figures

[Figure 1: <Software Process Model> Waterfall 16](file:///E:\Study\Capstone%20Projecct\Project\trunk\Document\CapstoneProject.docx#_Toc421283585)

[Figure 2: Overview Usecase 23](#_Toc421283586)

[Figure 3 <User>Overview Use Case 24](#_Toc421283587)

[Figure 4 <User> Login 24](#_Toc421283588)

[Figure 5: <Admin> Overview Use case 25](#_Toc421283589)

[Figure 6: <Admin> Reset Password 25](#_Toc421283590)

[Figure 7: <Admin> Edit Profile 25](#_Toc421283591)

[Figure 8: <Admin> Logout 25](#_Toc421283592)

[Figure 9: <Admin> Search Accounts 25](#_Toc421283593)

[Figure 10: <Admin> Add Account 25](#_Toc421283594)

[Figure 11: <Admin> Edit Account 25](#_Toc421283595)

[Figure 12: <Admin> Activate/Deactivate Account 25](#_Toc421283596)

[Figure 13: <Admin> Search List Skills 25](#_Toc421283597)

[Figure 14: <Admin> Add Skill List 25](#_Toc421283598)

[Figure 15: <Admin> Update Skill List 25](#_Toc421283599)

[Figure 16: <Admin> Add Category 25](#_Toc421283600)

[Figure 17: <Admin> Search Categories 25](#_Toc421283601)

[Figure 18: <Admin> Update Category 25](#_Toc421283602)

[Figure 19: <Admin> Create Period Assessment 25](#_Toc421283603)

[Figure 20: <Admin> Search Period Assessment 25](#_Toc421283604)

[Figure 21: <Admin> Edit Period Assessment 25](#_Toc421283605)

[Figure 22: <Admin> Create Criterion 25](#_Toc421283606)

[Figure 23: <Admin> Edit Criterion 25](#_Toc421283607)

[Figure 24: <Admin> Search Criterions 25](#_Toc421283608)

[Figure 25: <Director> Overview Use Case 25](#_Toc421283609)

[Figure 26: <Director> Add Manager into Project 25](#_Toc421283610)

[Figure 27: <Director> Remove Manager from Project 25](#_Toc421283611)

[Figure 28: <Director> Create Project 25](#_Toc421283612)

[Figure 29: <Director> Update Project 25](#_Toc421283613)

[Figure 30: <Director> Close Project 25](#_Toc421283614)

[Figure 31: <Director> Remove Project 25](#_Toc421283615)

[Figure 32: <Director> View Productivity Assessment 25](#_Toc421283616)

[Figure 33: <Manager> Overview Use Case 25](#_Toc421283617)

[Figure 34: <Manager> Add Staffs into Project 25](#_Toc421283618)

[Figure 35: <Manager> Remove Staffs from Project 25](#_Toc421283619)

[Figure 36: <Manager> Create Task 25](#_Toc421283620)

[Figure 37: <Manager> Assign Task 25](#_Toc421283621)

[Figure 38: <Manager> Task Review 25](#_Toc421283622)

[Figure 39: <Manager> Edit Task 25](#_Toc421283623)

[Figure 40: <Manager> Remove Task 25](#_Toc421283624)

[Figure 41: <Manager> View Project Details 25](#_Toc421283625)

[Figure 42: <Manager> Search Assessments 25](#_Toc421283626)

[Figure 43: <Manager> Add Assessment 25](#_Toc421283627)

[Figure 44: <Manager> Edit Assessment 25](#_Toc421283628)

[Figure 45: <Staff> Overview Use Case 25](#_Toc421283629)

[Figure 46: <Staff> Search Topics 25](#_Toc421283630)

[Figure 47: <Staff> Add Topic 25](#_Toc421283631)

[Figure 48: <Staff> Edit Topic 25](#_Toc421283632)

[Figure 49: <Staff> Close Topic 25](#_Toc421283633)

[Figure 50: <Staff> View Report Status of Work 25](#_Toc421283634)

[Figure 51: <Staff> Search Tasks 25](#_Toc421283635)

[Figure 52: <Staff> View Employee Assessments 25](#_Toc421283636)

[Figure 53: <Staff> View All Tasks 25](#_Toc421283637)

[Figure 54: <Staff> View Task Detail 25](#_Toc421283638)

[Figure 55: <Staff> Search Project 25](#_Toc421283639)

[Figure 56: <Staff> Change Task Status 25](#_Toc421283640)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Name** | **Definition** |
|  |  |

# Introduction

## Project Information

* Project name: **Office Task Management**
* Project Code: **OTM**
* Product Type: **Website, Mobile app**
* Start Date: **May 13th, 2015**
* End Date: **August 22th, 2015**

## Introduction

Nowadays, business organizations have a very large number of tasks every day, so managers may cause mistakes in tracking work and managing tasks. To solve problems of using handwriting to track tasks and manage works, a solution that will support manager to manage and monitor progress tasks and help members track project’s status in needed.

From actual needs, we give the idea of developing a product that helps manage office tasks for business uses. Besides website version, we will develop one more mobile version. Specifically, this version will run based on Android OS.

With this product, we hope that user will follow up tasks more easily.

## Current Situation

Current systems can support users follow up and manage tasks on website, but don’t have mobile version or vice versa. Besides, current systems have comment function on website, but cannot discuss with real-time collaboration.

All of current systems are web service that allows users create project then add tasks into project simply and freely. This model is not suitable with complex groups like a company.

## Problem Definition

Below are the advantages and disadvantages of current office task management software:

* Advantages:
* Have a nice and simple user-interface, so user can manage tasks more easy.
* Have support review document, download or upload files, and manage groups.
* Disadvantages:
  + - Don’t have support report progress.
    - Lack of productivity assessments.
    - Don’t have discuss function with real-time collaboration on website and mobile.

## Proposed Solution

Our project includes website and mobile app not only let managers track tasks and manage processes, but also help assessing productivity of work and creating convenient communication environment.

### **Feature functions**

* Assign task
* Document review and approval
* Task review and approval
* Group discussion
* Productivity assessment
* Report

### **Advantage and disadvantage**

* Advantages:
  + Easy to use
  + Simple interface
  + Easy to divide task
  + Approve document fast and easily
  + Discuss group in real-time
  + Productivity assessment based on effort and difficult-level of task
  + Report progress of work
* Disadvantages:
  + Only for small companies
  + Report feature is simple
  + Business constraint of this system may not be applicable to some others kind of company.

## Functional Requirement

Function requirements of the system are listed as below:

### **User Management**

* Add, edit, delete account
* Manage user privileges

### Tracking task

* Assign task
* Monitor task
* View, update, delete task
* Search task

### Review and approve document

* Upload/ Download document
* Approve document

### Collaborate in real-time

* Chat group
* See history

### Statistic

* Report
* Productivity assessment

## Role and Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Full Name | Role | Position | Contact |
| 1 | Lại Đức Hùng | Project Managers | Instructor | [Hungld@fpt.edu.vn](mailto:Hungld@fpt.edu.vn) |
| 2 | Nguyễn Cự Đạt | Developer | Leader | [Datncse601010@fpt.edu.vn](mailto:Datncse601010@fpt.edu.vn) |
| 3 | Lư Cẩm Toàn | Developer | Member | [Toanlcse60633@fpt.edu.vn](mailto:Toanlcse60633@fpt.edu.vn) |
| 4 | Lê Hải Triều | Developer | Member | [Trieulhse61146@fpt.edu.vn](mailto:Trieulhse61146@fpt.edu.vn) |
| 5 | Trần Thị Kim Anh | Developer | Member | [Anhttkse60909@fpt.edu.vn](mailto:Anhttkse60909@fpt.edu.vn) |

Table : Role and Responsibility

# Software Project Management Plan

## Problem Definition

### Name of this Capstone project

* **Office Task Management (OTM)**

### Problem Abstract

* Nowadays, there is a large number of routine works in a company or a division, and most of the company work in documents that lead they lose material and manage the assessment difficulty. Office Task Management is system that helps users manage their works, the staffs follow up the works entrusted, the managers watch for their team and project, the director manages the company’s employee and support to evaluate project and employee more specific.

### Project Overview

#### Current Situation

There are many of office work management systems in recent time.

For example: Mantis Bug Trasker.

* Advantages:
  + Having friendly interface with users.
  + Supporting real-time collaboration.
* Disadvantages:
  + Not supporting forum to the employee of a team to discuss their works.
  + Not supporting tasks and projects assessment to attend to the estimation of an employee in the period of the year.
  + Their system is developed for group working, not for a business organization.
  + Use Bug management system to manage Tasks.

#### The proposed system

To deal this situation, our team suggest the **Office Task Management** system project.

##### Web Site

Web application has these main core:

* Director and manager can manage the projects and tasks in a company, give assessment about member in a projct in period time, update status of projects and tasks in real-time.
* Beside, director still give productivity assessment.
* Member in a project can discuss together at forum of each project, beside they can give comment in each task.
* Staff can track tasks based on reports, they support to manage status of tasks and number tasks.

##### Mobile Application

* Director and manager can track projects and tasks when they don’t stay at company.

#### Boundaries of the System

* The system can be used by everybody who have basic knowledge about IT and have an Android smartphone.
* The language of system is English.
* The completed product includes:
  + Website version for all roles of user.
  + Mobile version for director and manager.
  + All the process document involved.

#### Development Environment

##### Hardware requirements

**For Server**

|  |  |  |
| --- | --- | --- |
| **Windows** | **Minimum Requirements** | **Recommended** |
| **Internet Connection** | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (12 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 : <Hardware Requirements> Server

**For Mobile**

|  |  |  |
| --- | --- | --- |
| **Windows** | **Minimum Requirements** | **Recommended** |
| **Internet Connection** | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (12 Mbps) |
| **Operating System** | Android 4.1.2 | Android 4.4.2 |
| **Hardware** | Intel® Xeon ® 1.4GHz | Intel® Xeon ® Quad Core  (12M Cache, 2.50 GHz) |
| **Computer Memory** | 1GB RAM | 2GB or more |

Table : <Hardware Requirement> Mobile

##### Software requirements

* Window Server 2008: operating system and platform for development.
* SQL Server 2008 Enterprise R2: used to create and manage the database for system.
* Eclipse Luna 4.4.2: used to implement website and web service.
* GitHub & TortoiseSVN: used for source control.
* StarUML, [www.lucidchart.com](http://www.lucidchart.com): used to create models and diagrams.
* Justinmind: used to create GUI prototypes.
* Skype, Garena: used for communication and meeting.

## Project organization

### Software Process Model

Project is developed under modified waterfall model.

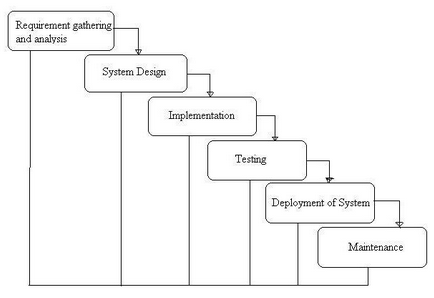


Figure : <Software Process Model> Waterfall

For more information: <http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/>

### Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Lại Đức Hùng | Project manager | * Specify user requirement * Control the development process * Give out technique and business analysis |
| **2** | Nguyễn Cự Đạt | Team Leader,  BA, Dev, Tester | * Managing process * Clarifying requirements * Prepare documents * GUI Design * Designing database * Coding * Create test plan * Testing |
| **3** | Lư Cẩm Toàn | Team Member,  BA, Dev, Tester | * Clarifying requirements * Prepare documents * GUI Design * Designing database * Coding * Create test plan * Testing |
| **4** | Lê Hải Triều | Team Member,  BA, Dev, Tester | * Clarifying requirements * Prepare documents * GUI Design * Designing database * Coding * Create test plan * Testing |
| **5** | Trần Thị Kim Anh | Team Member,  BA, Dev, Tester | * Clarifying requirements * Prepare documents * GUI Design * Designing database * Coding * Create test plan * Testing |

Table : Role and Reponsibility (2)

### Tools and Techniques

* Front-end technologies: HTML5, CSS3, JavaScript, jQuery, AJAX.
* Back-end technologies:
  + Website: Java EE
  + Web service: Java + Jersey
  + Third-party libraries:
* Mobile: Android JellyBean 4.1.2 & Android KitKat 4.4.2 –Java 7
* WebServer: Apacha Tomcat 7.0.61
* Database Management System: MSSQL Server 2008 R2 Enterprise

## Project Management Plan

### Software development life cycle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Phase** | **Description** | **Deliverables** | **Resource needed** | **Dependencies and Constrains** | **Risks** |
| **Requirement gathering and Analysis** | - Gather requirement from customers.  - Study business in reality to determine business rules.  - Identify and clarify requirement base on real business. | - Introduction of system.  - Software project management plan.  - Software requirement spectification.  - Business rules.  - Protoypes. |  | N/A | - Members in team don’t identify with the requirement understanding .  - Misunderstanding about business. |
| **System Design** | - Determine architecture style which will be used.  - Design architecture for the system.  - Solve problems and plan solutions. | - Software design document.  - Architectual style for code.  - Technologies for the system. |  | Depend on “Requirement gathering and analysis”. | - Lack of knowledge and experience.  - Lack of understanding requirement |
| **Implementation** | - Code GUI.  - Code core functions and other features.  - Unit test and debug. | - Web application.  - Mobile application on Android. |  | Depend on “System design”. | - Mistake while implementing |
| **Testing** | - Unit testing.  - Integration testing.  - System testing.  - Fix bugs. | - Test document. |  | Depend on “Implemention”. | - Lack of experience.  - Lack of test cases. |
| **Deployment of System** | - Deploy on web server and mobile. | - Installation guide.  - User Manual |  | Depend on “Testing”. | - Lack of experience. |
| **Maintenance** | - Maintain and update the system. | N/A |  | Depend on “Deployment of system”. | N/A |

Table : Software Deverlopment Life Cycle

### Phase Detail

#### Phase 1: Requirement gathering and analysis

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Gather requirement | - Collect requirement from customer.  - Study from current similar system. | DatNC, ToanLC, TrieuLH, AnhTTK |
| 2. Study business | - Define strengths and weakness of current similar system.  - Interview some real PMs about business. | DatNC, ToanLC, TrieuLH, AnhTTK |
| 3. Identify and clarify requirement | - Define main scope of system. | DatNC, ToanLC, TrieuLH, AnhTTK |
| 4. Create system introduction, SRS document | - Complete introduction of system  - Write SRS document | DatNC, TrieuLH |
| 5. Design prototypes | - Design GUI prototypes by tools | ToanLC, AnhTTK |

Table : <Phase Detail> Phase 1

#### Phase 2: System design

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Research technologies | - Research third-party’s APIs support real-time collaboration and push notification.  - Study Android application development.  - Study web services using Jersey | DatNC, ToanLC, TrieuLH, AnhTTK |
| 2. Design ERD | - Create ER diagram. | DatNC, ToanLC, TrieuLH, AnhTTK |
| 3. Design database | - Design logical and create physical database | DatNC, TrieuLH |
| 4. Design Class diagram | - Create Class diagram | DatNC, ToanLC, TrieuLH, AnhTTK |
| 5. Detail design and create design document | - Top-down design.  - Create and update software design document | DatNC, ToanLC, TrieuLH, AnhTTK |

Table : <Phase Detail> Phase 2

#### Phase 3: Implemention

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Implement web front-end | - Code GUI for web application | DatNC, ToanLC, TrieuLH, AnhTTK |
| 2. Implement web back-end | - Code functions for web application | DatNC, ToanLC, TrieuLH, AnhTTK |
| 3. Implement mobile front-end | - Code GUI for mobile application |  |
| 4. Implement mobile back-end | - Code functions for mobile application |  |
| 5. Unit testing | - Unit test  - Fix bugs |  |

Table : <Phase Detail> Phase 3

#### Phase 4: Testing

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Unit testing | - Write test cases  - Test and fix bugs. |  |
| 2. Integration testing | - Write test cases  - Test and fix bugs. |  |
| 3. System testing | - Write test cases  - Test and fix bugs. |  |
| 3. Create test document | - Create and update test document |  |

Table : <Phase Detail> Phase 4

#### Phase 5: Deployment of System

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Deployment | - Deploy system on server and mobile |  |
| 2. Create installation guide | - Write installation guide |  |
| 3. Create user manual | - Write user manual |  |

Table : <Phase Detail> Phase 5

#### Phase 6: Maintenance

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Maintain and update system | - Maintain current system  - Update new features |  |

Table : <Phase Detail> Phase 6

### All Meeting Minutes

Place at Meeting Minutes folder.

## Coding Convention

Use Java coding convention to develop website, web service and mobile application.

Covention:

* Naming convention:
  + Use camel case for name of variables and functions.
  + Use Pascal case for class
* Indentation:
  + Avoid lines longer than 80 characters, since they are not handled well by many terminals and tools.
* Declaration:
  + In absolutely no case should variables and functions be declared on the same line.
  + Do not put different types on the same line.

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>

# Software Requirement Specification

## User Requirement Specification

In this project, user will be devide into roles:

### Admin Requirement

Admin is a person who manage directly system and database. Admin can use some functions in the system. To use that functions, user must login with Admin Role. There are some functions admin can use:

* Mange account: create, edit, and search, change status of account
* Manage skill: add, update, and search skill.
* Manage skillCategory: add, update, and search category.
* Manage Period Assessment: create, edit, and search assessment.
* Manage Criterions: create, edit, search, and remove criterion.

### Director Requirement

Director is a person who manage projects and task and have highest power in this system. Director has some functions in the system. Beside, director can also use function of manager and staff. There are some functions:

* Manage Project: create, update, and remove project
* Add/Remove manager into/from project.
* View Productivity Assessment.

### Manager Requirement

Manager is a person who receive projects from director and manage tasks. Manager has some functions in the system. Beside, manager can also use function of staff. There are some functions:

* Manage Task: create, update, remove, assign, and review task.
* Add/Remove staff into/from project.
* View project detail
* Manage Employee Assessment: add, edit, and search assessment.

### Staff Requirement

Staff is a person who join in project. Staff has some functions in the system.

* View all tasks
* View task detail
* View report status of task
* Manage discussion: add, edit, search, and close topic
* Edit profile
* View employee assessment
* Change status of task
* Search task, project.

## System Requirement Specification

### External Interface Requirement

#### User Interface

<Các yêu cầu về giao diện>

**Ví dụ**

* 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.
* Some design principles will be taken into consideration:
  + UI for businesss web applications - Janko Jovanovic [Ref: [http://www.smashingmagazine.com/2010/02/25/designing-](http://www.smashingmagazine.com/2010/02/25/designing-user-interfaces-for-business-web-applications/) [user-interfaces-for-business-web-applications/]](http://www.smashingmagazine.com/2010/02/25/designing-user-interfaces-for-business-web-applications/)
  + Ten principles of effective web design – Vitaly Friedman [Ref: [http://www.smashingmagazine.com/2008/01/31/10-](http://www.smashingmagazine.com/2008/01/31/10-principles-of-effective-web-design/) [principles-of-effective-web-design/]](http://www.smashingmagazine.com/2008/01/31/10-principles-of-effective-web-design/)
  + Principles of mobile interface design – Jonathan Stark [Ref: <http://www.oreilly.com/pub/e/2144>]

#### Hardware Interface

#### Software Interface

<Yêu cầu về phần mềm chú ý ghi rõ phiên bản cũng như kích thước màn

hình>

#### Communication Protocol

<Yêu cầu về giao tiếp giữa các thành phần trong ứng dụng>

### System Overview Use Case



Figure : Overview Usecase

### List of Use Case

#### <User>Overview Use Case

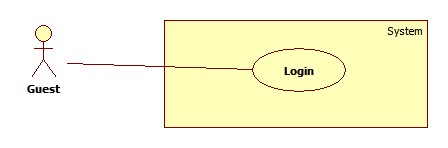


Figure <User>Overview Use Case

##### <User> Login

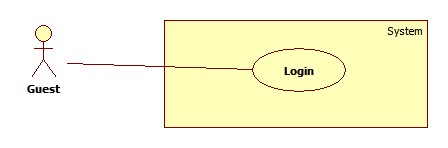


Figure <User> Login

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC001** | | | |
| **Use Case No.** | UC001 | **Use Case Version** | 2.0 |
| **Use Case Name** | Log in | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table <User> Login

#### <Admin>Overview Use Case

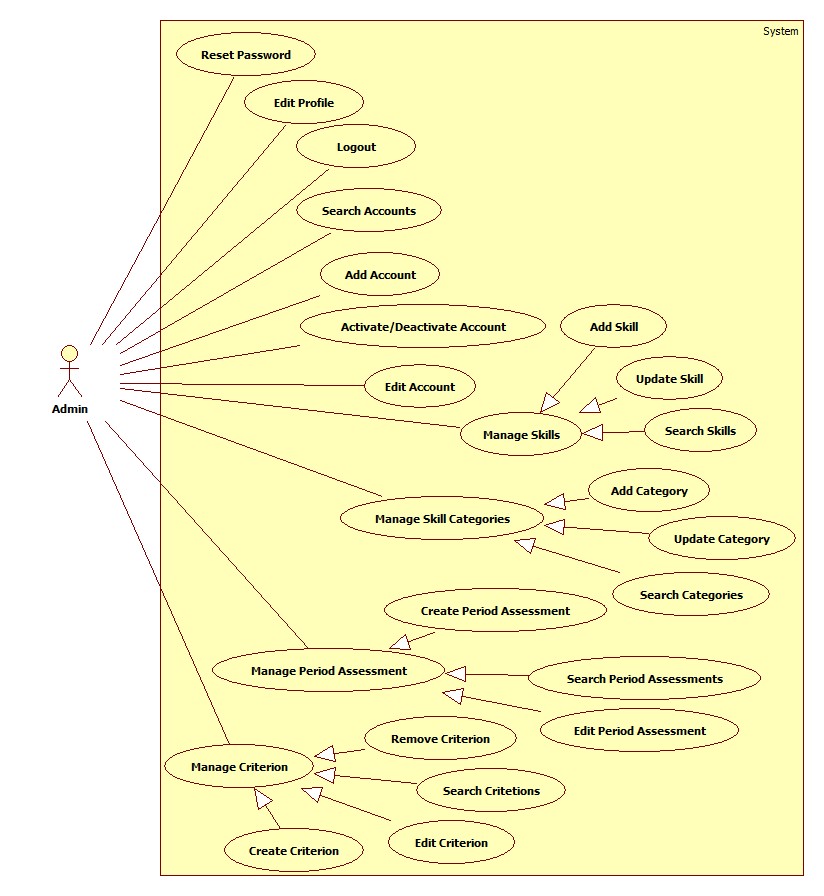


Figure : <Admin> Overview Use case

##### <Admin> Reset Password

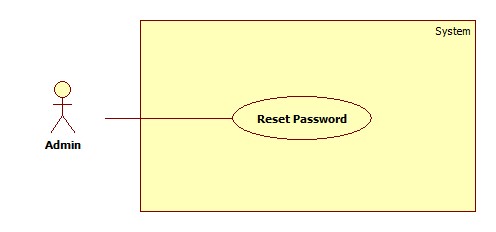


Figure : <Admin> Reset Password

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC002** | | | |
| **Use Case No.** | UC002 | **Use Case Version** | 2.0 |
| **Use Case Name** | Reset Password | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Reset Password

##### <Admin> Edit Profile

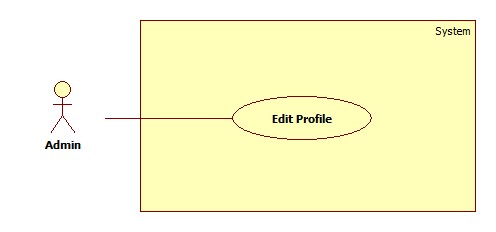


Figure : <Admin> Edit Profile

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC003** | | | |
| **Use Case No.** | UC003 | **Use Case Version** | 2.0 |
| **Use Case Name** | Edit Profile | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Edit Profile

##### <Admin> Logout

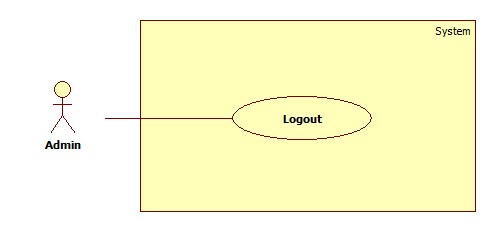


Figure : <Admin> Logout

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC004** | | | |
| **Use Case No.** | UC004 | **Use Case Version** | 2.0 |
| **Use Case Name** | Logout | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Logout

##### <Admin>Search Accounts

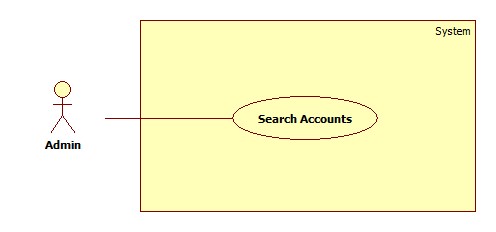


Figure : <Admin> Search Accounts

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC005** | | | |
| **Use Case No.** | UC005 | **Use Case Version** | 2.0 |
| **Use Case Name** | Search Accounts | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Search Accounts

##### <Admin>Add Account

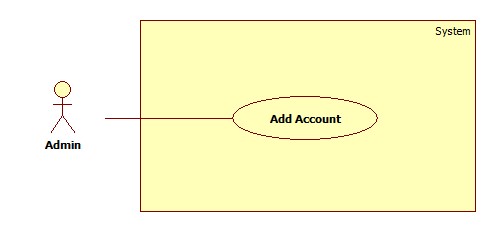


Figure : <Admin> Add Account

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Add Account

##### <Admin>Edit Account

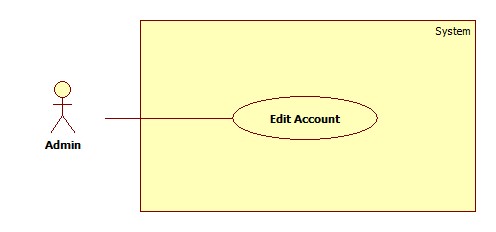


Figure : <Admin> Edit Account

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC007** | | | |
| **Use Case No.** | UC007 | **Use Case Version** | 2.0 |
| **Use Case Name** | Edit Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Edit Account

##### <Admin>Activate/Deactivate Account

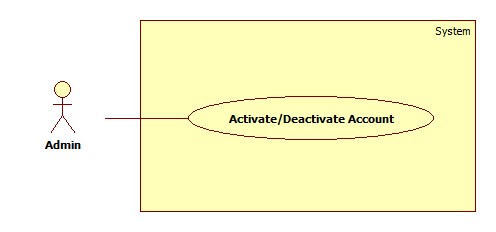


Figure : <Admin> Activate/Deactivate Account

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC008** | | | |
| **Use Case No.** | UC008 | **Use Case Version** | 2.0 |
| **Use Case Name** | Activate/Deactivate Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Activate/Deactivate Account

##### <Admin> Search List Skills

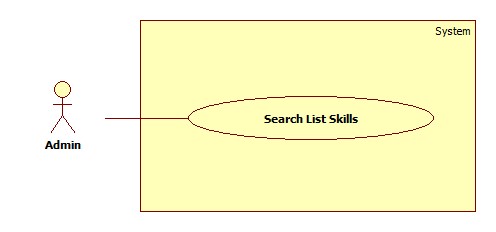


Figure : <Admin> Search List Skills

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC009** | | | |
| **Use Case No.** | UC009 | **Use Case Version** | 2.0 |
| **Use Case Name** | Search List Skills | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Search List Skills

##### <Admin> Add Skill List

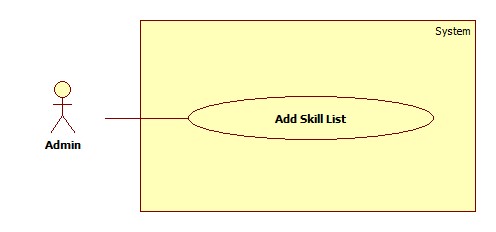


Figure : <Admin> Add Skill List

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC010** | | | |
| **Use Case No.** | UC010 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Skill List | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Add Skill List

##### <Admin> Update Skill List

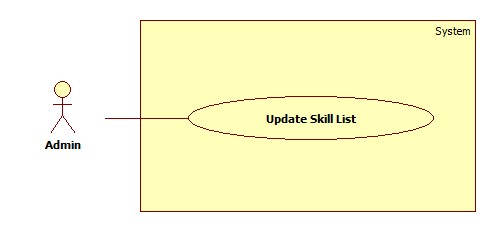


Figure : <Admin> Update Skill List

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Update Skill List

##### <Admin> Add Category

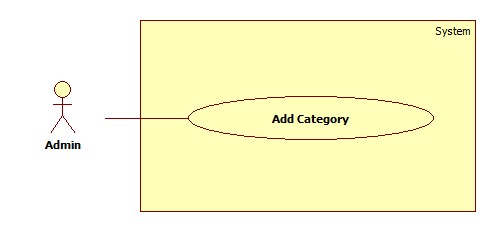


Figure : <Admin> Add Category

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Add Category

##### <Admin> Search Categories

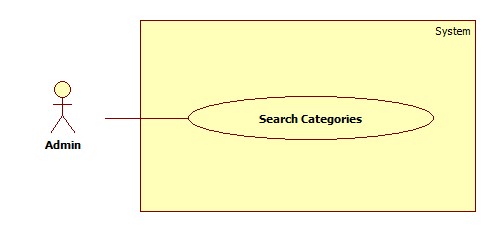


Figure : <Admin> Search Categories

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Search Categories

##### <Admin> Update Category

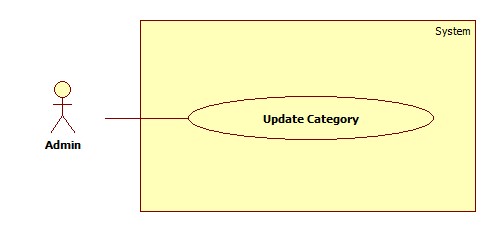


Figure : <Admin> Update Category

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Update Category

##### <Admin> Create Period Assessment

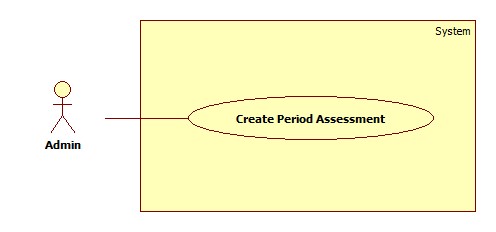


Figure : <Admin> Create Period Assessment

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Create Period Assessment

##### <Admin> Search Period Assessments

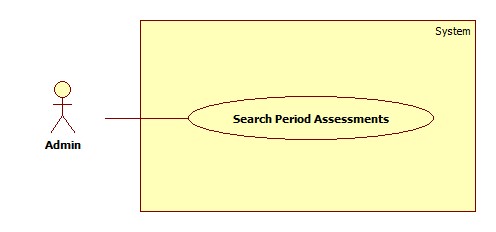


Figure : <Admin> Search Period Assessment

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Search Period Assessment

##### <Admin> Edit Period Assessment

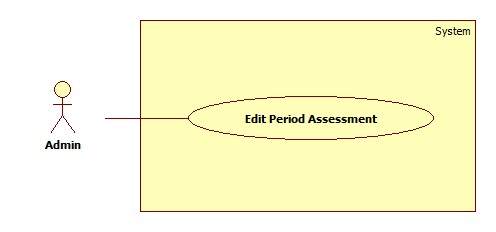


Figure : <Admin> Edit Period Assessment

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Edit Period Assessment

##### <Admin> Create Criterion

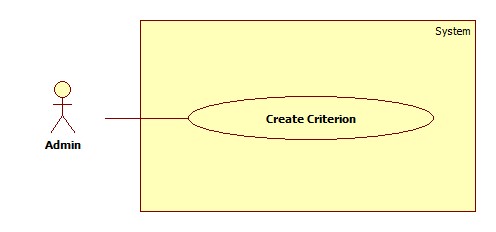


Figure : <Admin> Create Criterion

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Create Criterion

##### <Admin> Edit Criterion

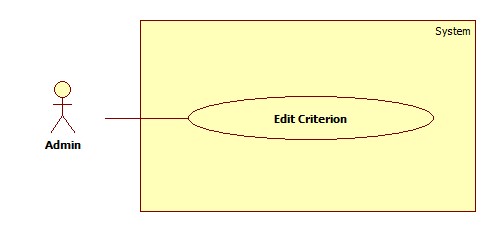


Figure : <Admin> Edit Criterion

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Edit Criterion

##### <Admin> Search Criterions

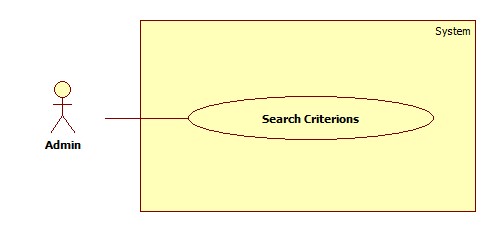


Figure : <Admin> Search Criterions

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Admin> Search Criterions

#### <Director>Overview Use Case

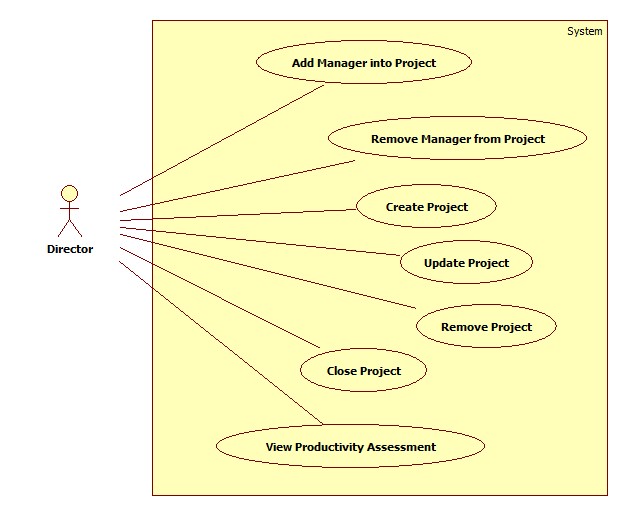


Figure : <Director> Overview Use Case

##### <Director> Add Manager into Project

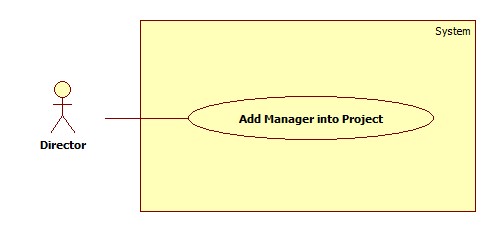


Figure : <Director> Add Manager into Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Director> Add Manager into Project

##### <Director> Remove Manager from Project

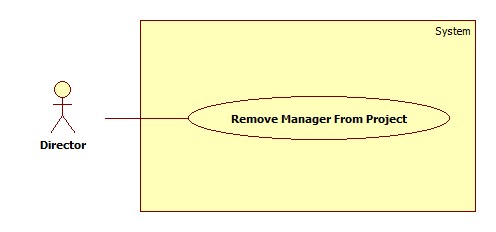


Figure : <Director> Remove Manager from Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Director> Remove Manager form Project

##### <Director> Create Project

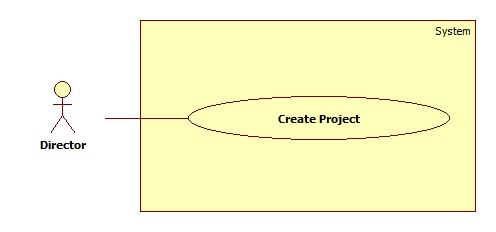


Figure : <Director> Create Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Director> Create Project

##### <Director> Update Project

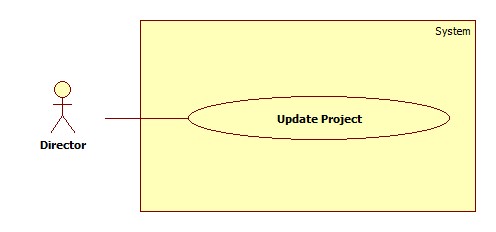


Figure : <Director> Update Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Director> Update Project

##### <Director> Close Project

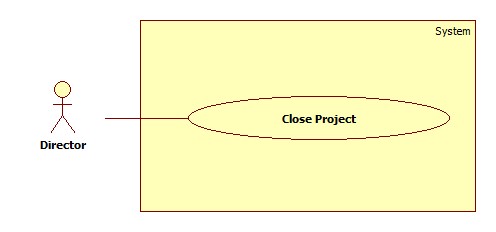


Figure : <Director> Close Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Director> Close Project

##### <Director> Remove Project

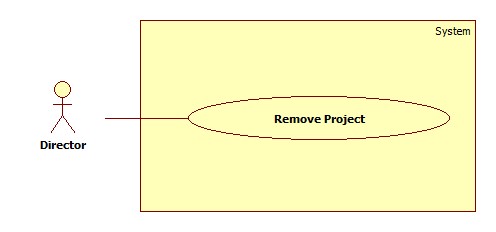


Figure : <Director> Remove Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Director> Remove Project

##### <Director>View Productivity Assessment

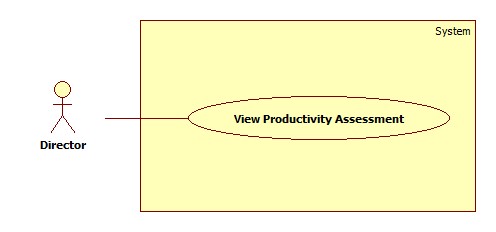


Figure : <Director> View Productivity Assessment

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Director> View Productivity Assessment

#### <Manager>Overview Use Case

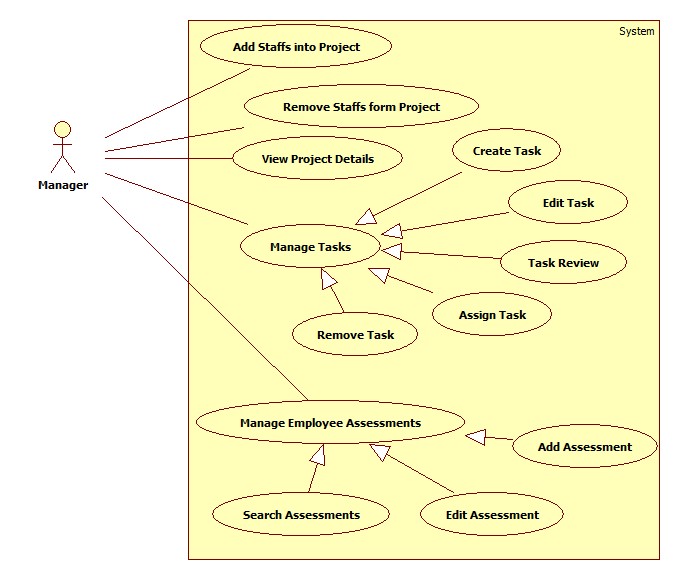


Figure : <Manager> Overview Use Case

##### <Manager> Add Staffs into Project

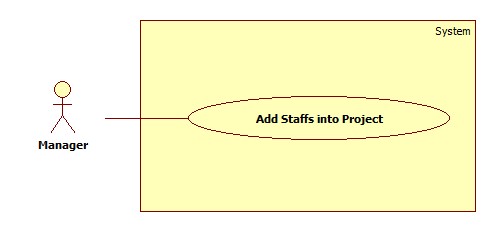


Figure : <Manager> Add Staffs into Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Add Staffs into Project

##### <Manager> Remove Staffs form Project

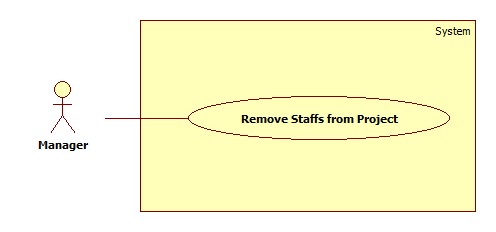


Figure : <Manager> Remove Staffs from Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Remove Staffs from Project

##### <Manager> Create Task

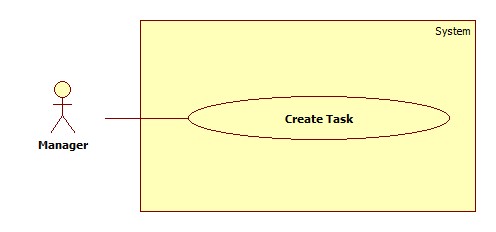


Figure : <Manager> Create Task

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Create Task

##### <Manager> Assign Task

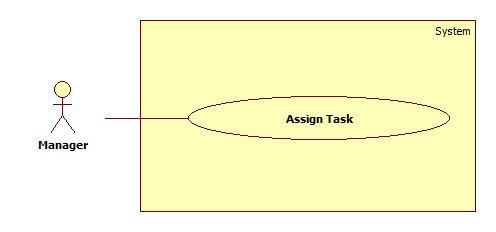


Figure : <Manager> Assign Task

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Assign Task

##### <Manager> Task Review

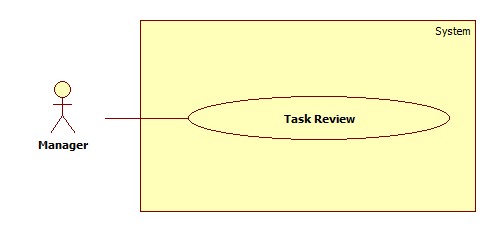


Figure : <Manager> Task Review

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Task Review

##### <Manager> Edit Task

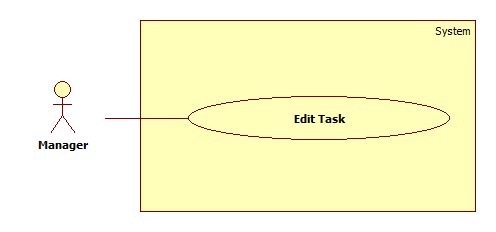


Figure : <Manager> Edit Task

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Edit Task

##### <Manager> Remove Task

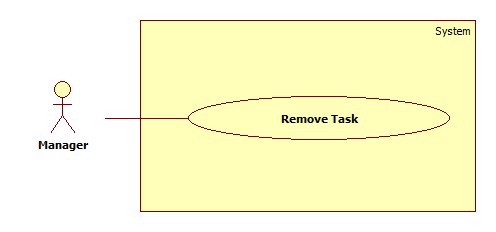


Figure : <Manager> Remove Task

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Remove Task

##### <Manager> View Project Details

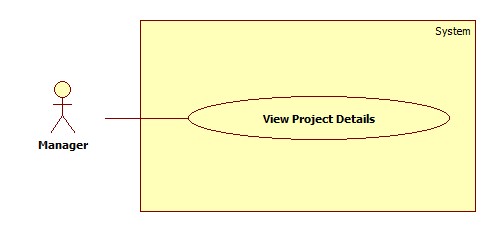


Figure : <Manager> View Project Details

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> View Project Details

##### <Manager> Search Assessments

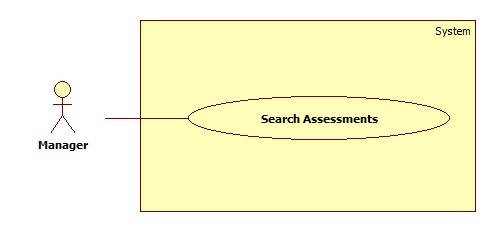


Figure : <Manager> Search Assessments

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Search Assessment

##### <Manager> Add Assessment

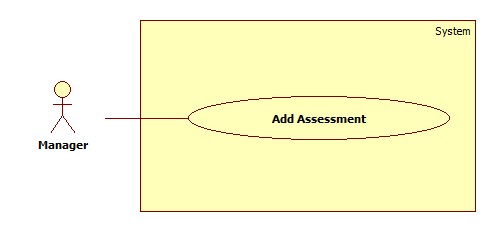


Figure : <Manager> Add Assessment

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Add Assessment

##### <Manager> Edit Assessment

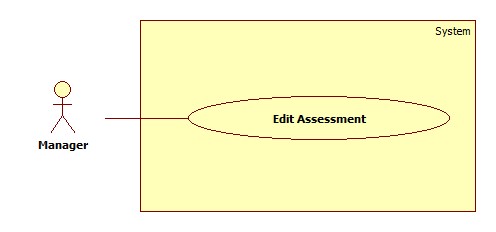


Figure : <Manager> Edit Assessment

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Manager> Edit Assessment

#### <Staff>Overview Use Case

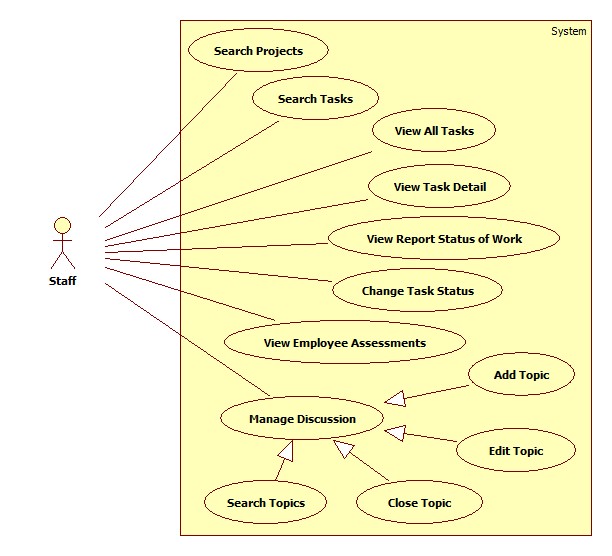


Figure : <Staff> Overview Use Case

##### <Staff> Search Topics

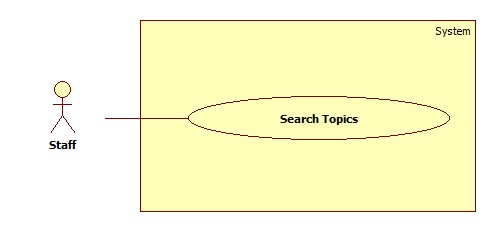


Figure : <Staff> Search Topics

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table :<Staff> Search Topic

##### <Staff> Add Topic

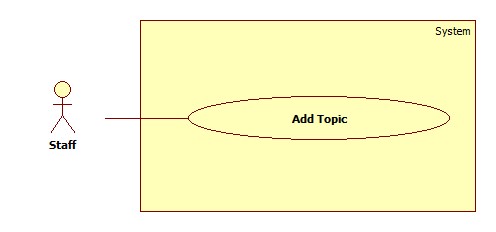


Figure : <Staff> Add Topic

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> Add Topic

##### <Staff> Edit Topic

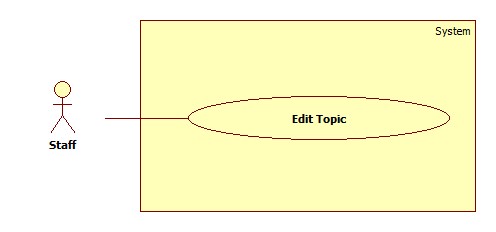


Figure : <Staff> Edit Topic

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> Edit Topic

##### <Staff> Close Topic

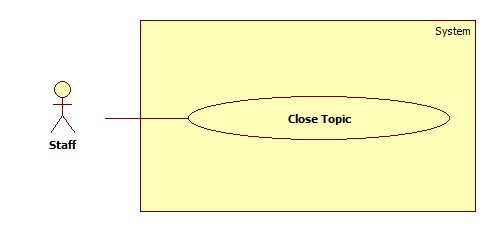


Figure : <Staff> Close Topic

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> Close Topic

##### <Staff> View Report Status of Work

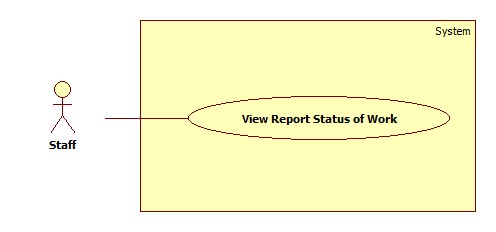


Figure : <Staff> View Report Status of Work

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> View Report Status of Work

##### <Staff> Search Tasks

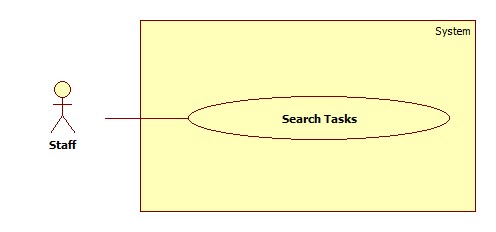


Figure : <Staff> Search Tasks

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> Search Tasks

##### <Staff> View Employee Assessments

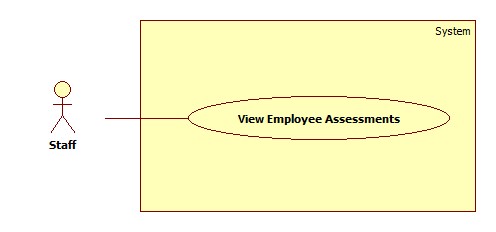


Figure : <Staff> View Employee Assessments

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> View Employee Assessments

##### <Staff> View All Tasks

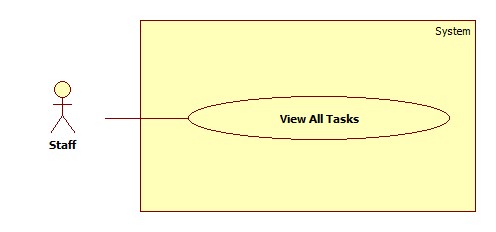


Figure : <Staff> View All Tasks

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> View All Tasks

##### <Staff> View Task Details

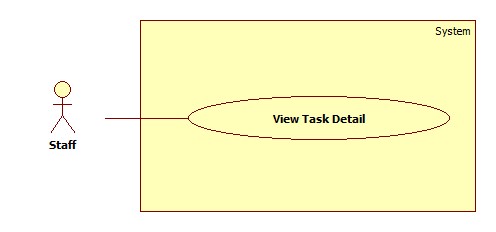


Figure : <Staff> View Task Detail

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> View Task Detail

##### <Staff> Search Projects

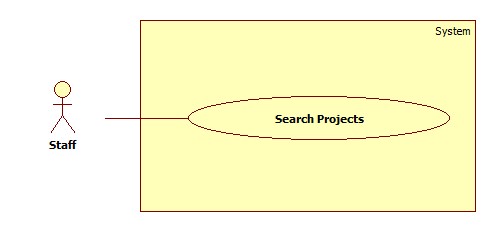


Figure : <Staff> Search Project

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> Search Project

##### <Staff> Change Task Status

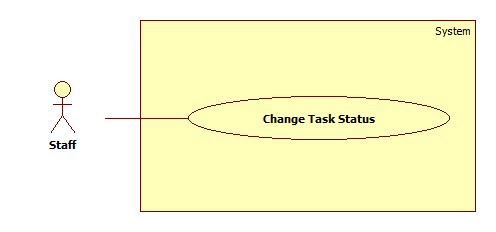


Figure : <Staff> Change Task Status

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE –UC006** | | | |
| **Use Case No.** | UC006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Account | | |
| **Author** | Pham Tran Tuan Vu | | |
| **Date** | 17/01/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use-case allows guest to register in the system.   **Goal:**   * Guest could log into the system to use system’s functions.   **Triggers:**   * Application start.   **Preconditions**:   * Guests have an account to sign in.   **Post Conditions:**   * **Success**: Guest log into the system. * **Fail**: System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Run application | System shows login window contains:   * “User Name”: textbox * “Password”: password textbox * “Login”: button * “Close”: button | | 2 | Guest fills username and password |  | | 3 | Guest clicks “Login” button  [Alternative 1] | Guest logs in to system and the system shows the Main Menu window.  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Guest clicks “Close” button | Close application. | | 2 | Wrong username or password. | System shows error message “Username or password is wrong, please try again”. |   **Exceptions:** N/A  **Relationships**: N/A  **Business Rules:**   * After login, the system automatically selects the most recent semester added | | | |

Table : <Staff> Change Task Status

## Software System Attribute

<Mô tả nonfunctional, biết thì ghi, ghi thì phải có dẫn chứng về việc đã đo đạc,

có định lượng như thế nào>

### Usability

### Reliability

### Availability

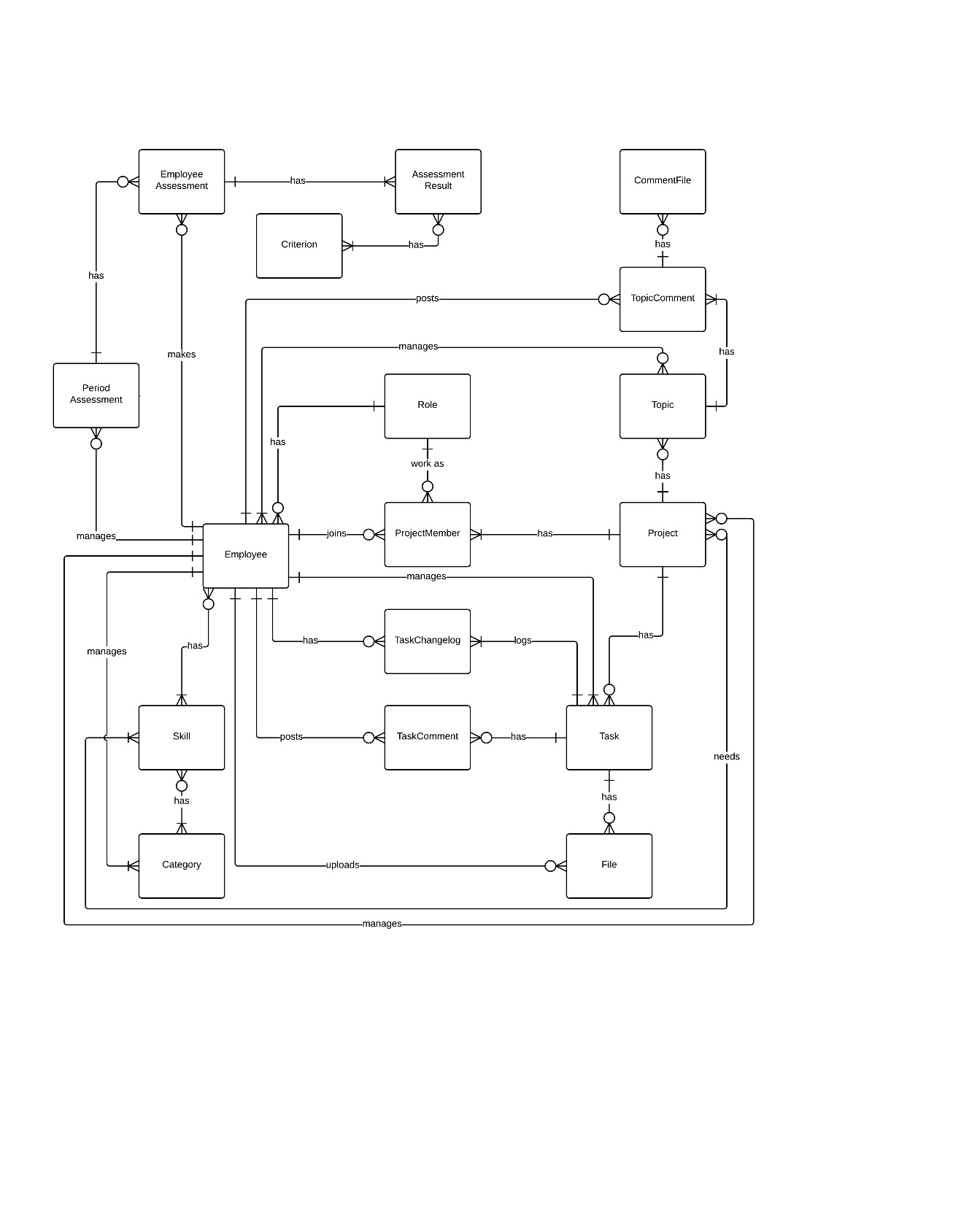
### Security

### Maintainability

### Portability

### Performance

## Conceptual Diagram



**Data Dictionary <Đặc tả các thực thể có trong hình>**

|  |  |
| --- | --- |
| **Entity Data dictionary: describe content of all entities** | |
| **Entity Name** | **Description** |
|  |  |

# Software Design Description

## Design Overview

<Tham khảo>

- This document describes the technical and user interface design of MSSC System. It includes the architectural design, the detailed design of common functions and business functions and the design of database model.

- The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.

- The detailed design describes static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.

- The database design describes the relationships between entities and details

of each entity.

- Document overview:

* Section 2: gives an overall description of the system architecture design.
* Section 3: gives component diagrams that describe the connection and integration of the system.
* Section 4: gives the detail design description which includes class diagram, class explanation, and sequence diagram to details the application functions.
* Section 5: describe screens design.
* Section 6: describe a fully attributed ERD.
* Section 7: describe algorithms **(HOW)**

## System Architectural Design

<Kiến trúc hệ thống mà nhóm xây dựng: chú ý hình trên mạng là không chính

xác tự vẽ sẽ là chính xác nhất, không sao chép, vay mượn và chế kí hiệu. Nếu dùng kí hiệu ngoài UML thì vui lòng ghi chú giải kí hiệu ngay cạnh hình vẽ.>

<Giải thích lý do tại sao lựa chọn mô hình này dựa trên SRS và Introduction của dự án đang thực hiện.>

<Mô tả các thành phần của kiến trúc theo dạng bảng, và sự tương tác giữa các thành phần theo kiến trúc.>

<Mô tả kiến trúc của từng thành phần trong ứng dụng nếu có.>

**Ví dụ**

Nếu ứng dụng làm web và mobile thì việc đặc tả theo mục 2.1 và 2.2

### Web application architecture description

<Mô tả cụ thể>

### Mobile application architecture description

<Mô tả cụ thể>

## Component Diagram

<Thể hiện góc nhìn hệ thống dưới sự tương tác của các thành phần component. Xem lại bộ quy ước kí hiệu của UML 2.0 trước khi vẽ các mối

quan hệ cũng như hiểu rõ thiết kế để vẽ chính xác. Nếu tool không phù hợp thì

nhóm nên dùng Paint để vẽ>

<Mô tả từng thành phần trong hình vẽ theo bảng biểu bên dưới.>

|  |  |
| --- | --- |
| **Component dictionary: describe component** | |
| **Component Name** | **Description** |
|  |  |

## Detailed Description

### Class Diagram

<Hình thiết kế class diagram: tham khảo các mối quan hệ giữa các lớp trong đặc tả UML, nắm rõ về dependency, association, composition, aggregation, inheritance>

Lời khuyên là tự vẽ, tránh lệ thuộc công nghệ

<Mô tả từng thành phần component theo bảng biểu bên dưới.>

|  |  |
| --- | --- |
| **Class dictionary: describe Class** | |
| **Class Name** | **Description** |
|  |  |

### Class Diagram Explanation

<Mô tả cụ thể cho các lớp>

Ví dụ

#### Role

Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| RoleID | int | Private | Unique identifier of a role |
| Name | string | Private | Role name |

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

### Interaction Diagram

#### Tên Interaction Diagram

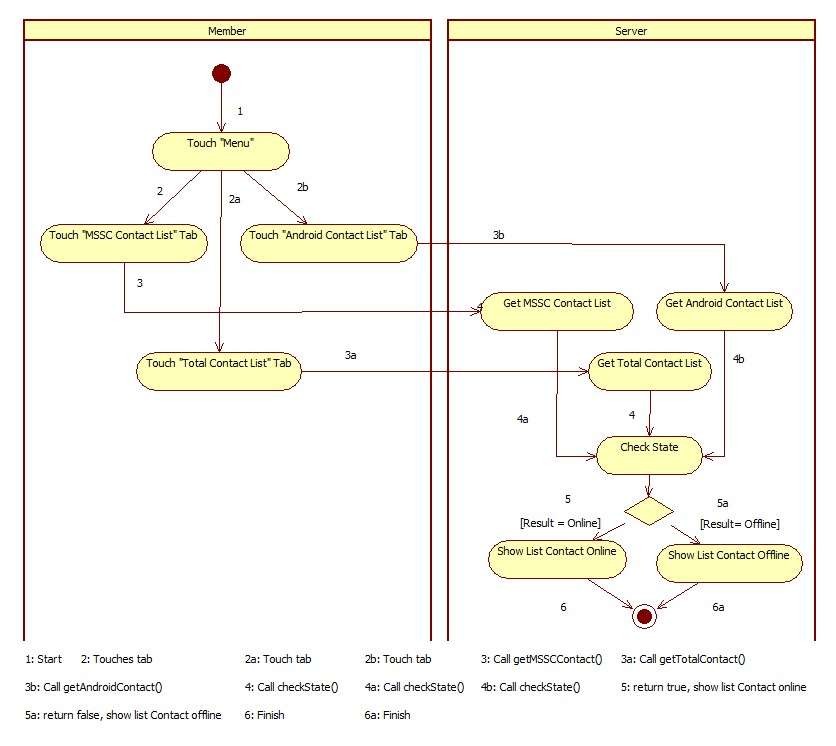
**Summary:** <Nên có phần tóm tắt trước 1 diagram nói về mục

đích của diagram trước khi thể hiện hình vẽ>.

**Ví dụ**

##### <Member> View Friend List

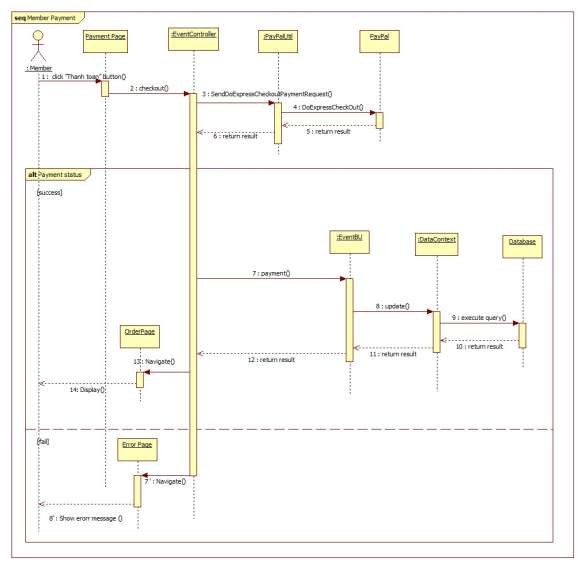
**Summary:** This diagram shows how member views all contacts that include MSSC contacts and android cell phone contacts.



**Figure 6: <Member> View Friend List**

#### <Member> Payment

**Summary:** This diagram shows how members make payment for a booking.



**Figure 7: <Member>Payment**

## User Interface Design

<Chụp và mô tả màn hình>. Lưu ý phải đánh số đặc tả các control trên giao diện cùng với các thành phần trong ràng buộc

Ví dụ

### Guest Interface Design

#### Login



**Figure 8: Login**

**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field**  **Name** | **Description** | **Read only** | **Mandatory** | **Control**  **Type** | **Data**  **Type** | **Length** |
| 1 | Username | Fill user name | No | Yes | Textbox | String | N/A |
| 2 | Password | Fill password | No | Yes | Password | String | N/A |

**Buttons/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 3 | Signin | Log-in into the system | N/A | Transfer to home page |

## Database Design

### Logical Diagram

<Thiết kế logical>

### Data Dictionary

<Mô tả về các thực thể>

|  |  |
| --- | --- |
| **Entity Data dictionary: describe content of all entities** | |
| **Entity Name** | **Description** |
|  |  |

**Table 8: Entity Data Dictionary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entity name** | **Attributes** | **Description** | **Domain** | **Null** |
| Role | RoleID {PK} | Unique identifier of role, auto increment. | int | No |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Name | Role name. | varchar(20) | No |

**Table 9: Detail Data Dictionary**

\* Business integrity constraint:

<Mô tả các ràng buộc về toàn vẹn dữ liệu để đảm bảo nghiệp vụ>

## Algorithms

<Các thành phần thuật toán mà nhóm đã áp dụng, chú ý không nhất thiết phải là thuật toán nổi tiếng mà có thể là cách tổ chức dữ liệu cũng như giải thuật do nhóm đang thực hiện ở bên trong hệ thống: ghi rõ bản chất, phân tích về độ phức tạp, nếu tham khảo phải ghi rõ nguồn>

Cách giải quyết hay cách áp dụng các qui trình nghiệp vụ hay cách chuyển đổi bài toán khi làm bằng tay - chưa áp dụng máy tính và chương trình để cho thấy việc áp dụng giải bài toán hay giải quyết vấn đề rồi chuyển đổi cách đó sang thành chương trình máy tính

**Ví dụ**

### Document Breakdown

#### Definition

Document breakdown is the way to break the document into many small parts. Each part has it own title and contents of it. And the final data has tree structure.

#### Define Problem

All content of document is quite difficute for manage so we must re- construc structure of document for using.

#### Solution

To solve this problem, we should follow these steps:

- Convert (save) document DOCX file as html type by using Microsoft

Word save as Web Filtered.

- Import both html file and directory that incluses all pictures of document.

- Using xpath to get data of html file as we need, include h1, h2, h3,…,

image, text content,..

- Save them with structure as below:

-TitleA: contentA

---TitleA1: contentA1

------TitleA1.1: contentA1.1

------TitleA1.2: contentA1.2

---TitleA2: contentA2

#### Complexity

- In total, the complexity of this algorithm is 

#### Flowchart

**Figure 9: Breakdown document flow chart**

### String Comparison

#### Define Problem

Given two strings. Calculate their matching percent.

#### Requirement

- Robustness to changes of word order: two strings which contain the same words, but in a different order, should be recognised as being similar.

- Language independence: the algorithm should work not only in

English, but in many different languages.

#### Solution

- If a string contains many words, break it into a list of words.

- For each word, we find out how many adjacent character pairs are contained in it.

- Create a function *pairs(s)* which returns a list of adjacent character pairs of string *s*.

- Then, we use below formula to calculate matching percent.



#### Example

Calculate the matching percent of 2 strings: France and French.

- Upper case 2 strings:

+ France FRANCE.



+ French FRENCH.

- Break string into list of adjacent character pairs:

+ FRANCE



+ FRENCH

- Calculate its matching percent.



# System Implementation & Test

## Introduction

### Overview

<Mô tả tống quát mục đích test chủ yếu với thời gian và scope và số lượng nhân lực

thì nhóm áp dụng phương pháp gì cho việc test>

**Ví dụ**

This section provides in detail all necessary information about implementation information and testing procedure of MSSC includes test

plans, test cases, test result and risks estimations.

### Test Approach

<Phương pháp kiểm thử của nhóm : black box, white box ...>

## Database Relationship Diagram

### Physical Diagram

<Vẽ database khi cài đặt vật lý: chú ý bố cục cũng nhu kích thước cho dễ đọc thể

hiện tôn trọng với người xem tài liệu>

### Data Dictionary

Mô tả thành phần theo bảng biểu bên dưới

|  |  |
| --- | --- |
| **Data dictionary: describe content of all tables** | |
| **Table Name** | **Description** |
| Tên | Explanation |

**Table 10: Data Dictionary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table name** | **Attributes** | **Description** | **Domain** | **Null** |
| Role | RoleID {PK} | Unique identifier of role, auto increment. | int | No |
| Name | Role name. | varchar(20) | No |

**Table 11: Attribute Data Dictionary**

## Performance Measures

<Cách nhóm ước lượng cũng nhu đo đạc hệ thống>

**Ví dụ**

### Clustering Performance

 Clustering is performed by running K Mean Algorithm which has complexity of : O(n \* k \* I \* d)

o n : number of points

o k : number of cluster

o I : number of iteration

o d : number of attributes (3)

Clustering take almost the time of process that we can ignore the time needed to load data from database, digitalize data.

The speed of clustering will vary and increase dramatically when n increase. The purpose of this project is not about optimizing K-Mean Algorithm so it is accepted to let the process run till it completes.

Moreover, the clustering is designed to run by staff, wait time is acceptable.

## Test Plan

<Đưa ra kế hoạch test>

Ví dụ

The purpose of this section is to verify and ensure that MSSC meets its design specification and other requirements from user. The following part will describe which features to be tested and which will not.

### Features to be tested

<Tính năng sẽ kiểm thử>

### Features not to be tested

<Tính năng sẽ không kiểm thử>

## System Testing Test Case

**<Nên vẽ các workflow tính năng sẽ test để dể hình dung, chú ý dàn trang in**

**ngang, chú ý đánh số, ngày tháng, kết quả, không sao chép>**

**Ví dụ**

**Figure 10: Guest, Member Core Flow**

MSSC - Introduction

### Guest Test Case

#### Search Event

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case**  **Description** | **Test Case Procedure** | **Expected output** | **Inter-test Case**  **Dependence** | **Result** | **Test Date** | **Note** |
|  |  |  |  |  |  |  |  |

MSSC - Introduction

# Software User’s Manual

## Installation Guide

### Setting up environment at server side

The following software must be installed into the server machine:

#### Hardware requirements

<Yêu cầu phần cứng server, chú ý xem lại các report trước để nhất quán>

#### Software requirements

<Yêu cầu phần mềm server, chú ý xem lại các report trước để nhất quán>

### Deployment at server side

<Mô tả quá trình triển khai lên server thực tế, gợi ý có thể gồm các

bước sau, chú ý khi làm phải chụp hình cụ thể để hướng dẫn cũng như so sánh kết quả thành công>

#### Prepare deployment package

#### Configure Server before deploy

#### Deploy web application on server

### Setting up the environment at client side

#### Setting up for computer

<Ghi rõ phiên bản tối thiểu để sử dụng>

## User Guide

<Viết hướng dẫn sử dụng cho người dùng>

# Appendix

<Các thành phần tham khảo của tài liệu chú ý tham khảo thêm cách ghi tại

[http://www.khoahocviet.info/meresci/vi/meresci03d4.html>](http://www.khoahocviet.info/meresci/vi/meresci03d4.html)