|  |
| --- |
|  |
| Capstone Project Document |

**WingS**

|  |  |  |
| --- | --- | --- |
| **Dandelion** | | |
| **Group Members** | Lưu Ngọc Mạnh | SE02619 |
| Vũ Công Chính | SE02585 |
| Nguyễn Minh Huy | SE02723 |
| Dương Đức Anh | SE02700 |
| Vũ Ngọc Trung | SE02967 |
| Cao Thị Phương Mai | SE02908 |
| **Supervisor** | Mr. Nguyễn Văn Sang | |
| **Project code** | DDL | |

**- Hanoi, 12/2015 -**

TABLE CONTENTS

[1 INTRODUCTION 5](#_Toc437975466)

[1.1 Purpose 5](#_Toc437975467)

[1.2 Acronyms and Definitions 5](#_Toc437975468)

[1.3 Abstract 5](#_Toc437975469)

[1.4 Literature Review 5](#_Toc437975470)

[1.4.1 Kickstarter 6](#_Toc437975471)

[1.4.2 Comicola 7](#_Toc437975472)

[1.5 Proposal 7](#_Toc437975473)

[1.5.1 The idea 7](#_Toc437975474)

[1.5.2 The objective 7](#_Toc437975475)

[1.5.3 Brief description about system 8](#_Toc437975476)

[1.5.4 System features 9](#_Toc437975477)

[1.6 Benefits from Project 10](#_Toc437975478)

[1.6.1 For our group 10](#_Toc437975479)

[1.6.2 For Community 10](#_Toc437975480)

[1.7 Critical Assumption and Constraints 10](#_Toc437975481)

[1.8 Potential Risks 11](#_Toc437975482)

[2 PROJECT MANAGEMENT 12](#_Toc437975483)

[2.1 Introduction 12](#_Toc437975484)

[2.1.1 Purpose 12](#_Toc437975485)

[2.1.2 Definitions and Acronyms 12](#_Toc437975486)

[2.2 Project Overview 12](#_Toc437975487)

[2.2.1 Project Description 12](#_Toc437975488)

[2.2.2 Scope and Purpose 13](#_Toc437975489)

[2.2.3 Assumptions and Constraints 15](#_Toc437975490)

[2.2.4 Project Objectives 16](#_Toc437975491)

[2.2.5 Critical Dependencies 17](#_Toc437975492)

[2.2.6 Project Risk 18](#_Toc437975493)

[2.3 Project Development Approach 19](#_Toc437975494)

[2.3.1 Project Process 19](#_Toc437975495)

[2.3.2 Requirement Change Management 21](#_Toc437975496)

[2.3.3 Quality Management 21](#_Toc437975497)

[2.4 Estimation 24](#_Toc437975498)

[2.4.1 Size 24](#_Toc437975499)

[2.4.2 Effort 24](#_Toc437975500)

[2.4.3 Schedule 25](#_Toc437975501)

[2.4.4 Resource 33](#_Toc437975502)

[2.4.5 Infrastructure 33](#_Toc437975503)

[2.4.6 Training Plan 34](#_Toc437975504)

[2.4.7 Finance 34](#_Toc437975505)

[2.5 Project Organization 35](#_Toc437975506)

[2.5.1 Organization Structure 35](#_Toc437975507)

[2.5.2 Project Team 35](#_Toc437975508)

[2.6 Communication & Reporting 38](#_Toc437975509)

[3 SOFTWARE REQUIREMENT 40](#_Toc437975510)

[3.1 Introduction 40](#_Toc437975511)

[3.1.1 Purpose 40](#_Toc437975512)

[3.1.2 Definitions and Acronyms 40](#_Toc437975513)

[3.2 User Requirement Specification 41](#_Toc437975514)

[3.2.1 Business Process Overview 41](#_Toc437975515)

[3.2.2 Product Features 52](#_Toc437975516)

[3.2.3 User characteristic 54](#_Toc437975517)

[3.2.4 Functional Requirements 54](#_Toc437975518)

[3.2.5 Non-functional Requirement 55](#_Toc437975519)

[3.3 Software Requirements Specification 56](#_Toc437975520)

[3.3.1 Overall Description 56](#_Toc437975521)

[3.3.2 Functional Specification 59](#_Toc437975522)

[3.3.3 Non-Functional Specification 138](#_Toc437975523)

[4 SOFTWARE DESIGN 140](#_Toc437975524)

[4.1 Introduction 140](#_Toc437975525)

[4.1.1 Purpose 140](#_Toc437975526)

[4.1.2 Definitions, Acronyms, Abbreviations 140](#_Toc437975527)

[4.2 Architecture design 140](#_Toc437975528)

[4.2.1 Choice of Architecture Design 140](#_Toc437975529)

[4.2.2 Architecture Representation 146](#_Toc437975530)

[4.2.3 Architectural Goals and Constraints 147](#_Toc437975531)

[4.2.4 Logical View 148](#_Toc437975532)

[4.2.5 Process view 152](#_Toc437975533)

[4.2.6 Deployment View 154](#_Toc437975534)

[4.3 Detail Design 155](#_Toc437975535)

[4.3.1 Package 155](#_Toc437975536)

[4.3.2 Function 185](#_Toc437975537)

[4.4 Data design 216](#_Toc437975538)

[4.4.1 Entity Relationship Diagram 216](#_Toc437975539)

[4.4.2 Table Diagram 218](#_Toc437975540)

[4.5 Screen Design 226](#_Toc437975541)

[4.5.1 User Client 226](#_Toc437975542)

[4.5.2 Admin 258](#_Toc437975543)

[5 SOFTWARE TESTING 274](#_Toc437975544)

[5.1 Introduction 274](#_Toc437975545)

[5.1.1 Purpose 274](#_Toc437975546)

[5.1.2 Definitions and Acronyms 274](#_Toc437975547)

[5.1.3 Scope of testing 275](#_Toc437975548)

[5.1.4 Constraints 276](#_Toc437975549)

[5.2 Test plan 277](#_Toc437975550)

[5.2.1 Test type 277](#_Toc437975551)

[5.2.2 Test stages 279](#_Toc437975552)

[5.2.3 Test model 279](#_Toc437975553)

[5.2.4 Acceptance test criteria. 281](#_Toc437975554)

[5.2.5 Futures to be tested 281](#_Toc437975555)

[5.2.6 Feature not to be tested 281](#_Toc437975556)

[5.2.7 Resources 281](#_Toc437975557)

[5.2.8 Test milestones 282](#_Toc437975558)

[5.2.9 Deliverables 283](#_Toc437975559)

[5.3 Test case 283](#_Toc437975560)

[5.4 Test Report 284](#_Toc437975561)

[5.4.1 System test report 284](#_Toc437975562)

[5.4.2 Integration test report 284](#_Toc437975563)

[5.4.3 Test report 284](#_Toc437975564)

[5.4.4 Defect report 285](#_Toc437975565)

[6 USER MANUAL 286](#_Toc437975566)

[6.1 Introduction 286](#_Toc437975567)

[6.1.1 Purpose 286](#_Toc437975568)

[6.1.2 Environment 286](#_Toc437975569)

[6.2 Installation Guideline 286](#_Toc437975570)

[6.2.1 Connecting an ASP.NET MVC Web App with SQL Azure 286](#_Toc437975571)

[6.2.2 Create & Deploying the Azure Website 291](#_Toc437975572)

[6.2.3 Connecting SQL Management Studio to SQL Azure 293](#_Toc437975573)

[6.3 User Guideline 297](#_Toc437975574)

[6.3.1 User Client 297](#_Toc437975575)

[6.3.2 Admin 319](#_Toc437975576)

[7 PROJECT RESULT REPORT 326](#_Toc437975577)

[7.1 Product Judging 326](#_Toc437975578)

[7.1.1 Advantage of Dandelion system 326](#_Toc437975579)

[7.1.2 Current Limitations 326](#_Toc437975580)

[7.1.3 Expectation in Future 326](#_Toc437975581)

[7.2 Project Summary 326](#_Toc437975582)

[7.3 Lesson Learnt 327](#_Toc437975583)

[8 REFERENCES 329](#_Toc437975584)

# INTRODUCTION

## Purpose

This document is created as the introduction for event WingS – our Capstone Event at FPT University. In this document, we will descript the overview of some existing systems, the initial idea for our event, a brief description about our expected system and some potential risks, critical assumptions, constrains. Moreover, this document also shows opportunities what it offers for users.

## Acronyms and Definitions

|  |  |
| --- | --- |
| **Acronym & Abbreviation** | **Definition** |
| WS | WingS |
| FU | FPT University |
| Q&A | Question and Answer |

**Table 1-1:** Definitions and Acronyms

## Abstract

WS makes connections that enable people to help themselves and each other. Website allows users (who created organization) create charity topics or event as long as it follows our rules to help other people. Donors can also take part in a event by donating money through a single donation or supporting event as a regular donor publicly or privately, not to profit financially. Instead, charity topic creators offer rewards to thank backers for their support. We hope our event can help as much people as we can in our country.

## Literature review

There are many charity organizations in Vietnam, but few of them can make connections between the organizations and people who need help. Some kind of charity organization website: childrenofvietnam.org, lovingkindnessvietnam.org.

* **Childrenofvietnam**



**Figure 1-1:** Website Childrenofvietnam

Childrenofvietnam is a website works solely as a volunteer, and does not receive a salary or compensation. They spend nothing on marketing, advertising, or solicitations.

* **Advantages:**
* Has friendly user interface
* **Disadvantages:**
* Only support PayPal. The language is English.
* No marketing or advertising so it is not popular in Vietnam.
* Only about children.
* **Lovingkindnessvietnam**



**Figure 1-2:** Website lovingkindnessvietnam

The mission of Lovingkindness Vietnam is to directly aid those in need throughout the area surround Nha Trang, Vietnam. Seeking out and visit the wonderful orphanages, charity schools, and communities, identify how best to improve the health and wellbeing of the children, and put into place programs to bring about lasting change. As they are a small charity dedicated to one area, they can oversee customized from donors overseas.

* **Advantages:**
  + Has friendly user interface
* **Disadvantages:**
  + Only support PayPal. The language is English.
  + No marketing or advertising so it is not popular in Vietnam.

## Proposal

### The idea

Nowadays in our country, there are many people have difficult circumstances. Besides that, many organizations and personals have financial condition and possibilities for helping. But the problem is they can’t meet the other. Thanks for Internet, WS can solve this problem by making connections that enable people to help themselves and each other.

### Objectives

This event is the Capstone Event in FPT University studying program.

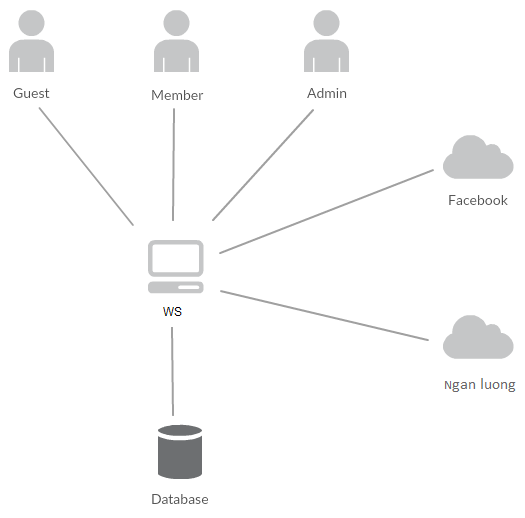
First, this event is responsibility for all team members. So we must complete all requirements from teacher and FU. During the time doing event, we can learn how to develop a event, how to manage event, how to working in group, how to control time, how to perform teamwork effectively. Moreover, we also have experience in .NET technology, website designing, software testing. Therefore, we think this event is very necessary for us to improve skills, knowledge which we studied in FU. This event also makes our CV more beautiful after graduating from FU. This is also an opportunity for us to prove capacity with recruiters.

Second, we hope that out event can help the other people and we hope to make our country better.

### Brief description about system

This software has friendly and attractive interface, users can easily find out the event to donate or create a new event when they have enough capacity and capability. User also can send opinion to admin if they need help.

WS uses familiar technologies as: Bootstraps, .NET MVC5, AngularJS…, and we are planning to develop a mobile version for easily connecting.



**Figure 1-3:** Brief descriptions diagram of WS system

### System features

About the features of WS, we focus to provide for users main features as searching, create new event, create thread, donate, message etc. Admin can manage members, manage organization, manage database, manage contribute database and manage Q&A.

In here, we focus on features which can help users can understand the best way what user want to have with features of WS. There are the detailed features of WS:

#### Client features

* Register: User can register an account and login to use all features of WS.
* Login/Logout: Users login/logout an account to use or exit system WS.
* Event
  + Create new event: Organization can create new program with basic info, contact, timeline and content.
  + Update program Information: User can change and update information of program which has been created.
  + Chat room: An event has a room chat where all user whom connected in can chat with the others.
  + Comment/Remind: Users can comment/remind an program to follow and discuss with another users.
  + Report: User can report a event which violate regulation to administrator.
* Thread:
  + Create new thread: Users can create new thread for discuss with other or get ideal about creating an event
  + Comment/Remind: Users can comment/remind a thread to follow and discuss with another users.
  + Report: User can report a thread which violate regulation to administrator
* Donate: User can donate money to help event and get point rewards.
* Organization:
  + Create new organization: Users can request create new organization.
  + Create a new event to raise donate.
  + Update profile: Organization can change or update information.
  + Manage created event.
* Profile:
  + Change password: User can change password to keep security.
  + Update profile: User can change or update information.
  + Manage created event, donated event, created thread. View statistical event had created, backed.
  + Sent/Receive message: Users can send to or receive from other users to discuss about event.
* Search:
  + Users can search any what they want follow: account, thread, event, organization.
* Message:
  + Users can message with others: members or admin.
* Report:
  + Users can report others users, events, threads, organizations.

#### Admin features

* Manage user:
  + Search user: Admin can search with name of user and system will display simple information about that user(name, email, phone number)
  + Inactive/active user: admin can set user’s account to inactive or active.
* Manage event:
  + Search event:
    - Search with event name: Admin can search with name of event and system will display simple information about that event.
  + Edit status: Admin can set status of an event: Inactive/active depend on report.
* Manage thread:
  + Search thread :
    - Search with thread name: Admin can search with name of thread and system will display simple information about that thread.
  + Edit status: Admin can set status of an thread: Inactive/active depend on report.
* Manage organization:
  + Admin manages creating organization requests from users. It means admin can accept or not a request.
  + Edit status: Admin can set status of an organization: Inactive/active depend on report.
* Manage report:
  + Admin recives reports of users. It means admin can accept or not a request.
* Message: Admin can view received messages/sent messages.
  + Compose new message: Admin can compose new message to send to user.

## Benefits from project

### For our team

After developing and implementing this event, our team will get some benefits:

* Have more experiences and knowledge about software event management how to manage plan, time, member and risk.
* Have more knowledge and skill about .NET MVC5 and AngularJS programming
* Know how to communicate with team members and how to teamwork more effective.

### For Community

Just need to have a device that connect to the internet and any web browser, user can:

* Easy join to a charity event to help other people who need.
* Easy to create a new charity event

## Critical Assumption and Constrains

* Critical assumption:
  + Training: All developers can train .NET and AngularJS in 7-10 days. Before starting this project, only 2 members in our team have knowledge about .NET MVC5 and one member have knowledge about AngularJS, but with programming techniques which we studied at FU, we have to try the best to not missing any deadline in project plan.
  + Human resources: Assume that all members in team have a good healthy to do their tasks.
* Constraints:
  + Time and deadline: We must complete task on time. We work on 14 weeks, each member works 5 hours/day and 5 days/week. We do not have more time for us to complete developing and deliver application to teachers. Besides, we have to submit report documents before deadline to teacher can review.
  + Quality: The products must be run well when users perform main functions in Google Chrome 53
  + Process: We have to follow the software processing of FPT Software
  + Human resources: There are 5 members in our team, each member have to study 2 subjects (Japanese and JIT) at school.

## Potential Risks

After studying about this event, we find out some problem that we may be encountered:

* Under-estimate scope and time or miss deadline because lack of experience in group working, managing and controlling work.
* Loss data, source code before uploading data to server because of careless.
* Equipment got broken because of careless or accident.
* Human resources: Team member cannot complete their works because of health reasons, key member leave team or un-cooperating on team.
* Change requirements: Requirement changed when some functions cannot be completed or some technologies is not suitable.

# PROJECT MANAGEMENT

## Introduction

### Purpose

This part is the project management plan of WingS (WS) Project – our Capstone Project  
in FPT University. It is included the project overview, project organization, tools and  
infrastructures, schedule of this project.

### Definitions and Acronyms

|  |  |  |
| --- | --- | --- |
| Acronym | Definition | Note |
| BA | Business Analyst |  |
| BU | Business Unit |  |
| CC | Infrastructure Configuration Controller |  |
| CM | Configuration Management |  |
| WS | WingS |  |
| ORG | Organization |  |
| DEV | Developer |  |
| PIC | Person in charge |  |
| PM | Project Manager |  |
| PTL | Project Technical Leader |  |
| QA | Quality Assurance Officer |  |
| SRS | Software Requirement Specification |  |
| TC | Test Case |  |
| PCB | Process Capability Baseline |  |

1. Definitions and acronyms

## Project overview

### Project Description

|  |  |  |  |
| --- | --- | --- | --- |
| Project Code | WS | Contract Type | None |
| Customer | FPT University | 2nd Customer | None |
| Project Level | Group | Project Rank | None |
| Application Type | Website | Project Manager | Đào Trọng Nghĩa |
| Project Category | Development | Business Domain | E-commerce |

1. Project Description

### Scope and Purpose

#### Purpose of Project

WS makes connections that enable people to help themselves and each other. Website allows users (person or organizations) create charity topics or raise fund project as long as it follows our rules to help other people. Backers can also take part in a project by donating money through a single donation or supporting project as a regular donor publicly or privately, not to profit financially. Instead, project creators offer rewards to thank backers for their support. We hope our project can help as much people as we can in our country.

#### Scope of Project

The scope of this project contains: Requirement Analysis, Design, Coding and Testing (Unit Test, Integration Test, and System Test).

##### The functions of Project

These are the functions of WS’s project:

##### Client User Module.

* **Register**: User can register an account and login to use all features of WS.
* **Login/Logout**: Users login/logout an account to use or exit system WS.
* **Search**: User and guest can search with: name of users, title of threads, event , name of organization, and system will display all of post relate to keywords.
* **Raise Request:** User can request to create an Organization. Admin can accept or refuse.
* **Thread:**
  + **Create New Thread:** User (ORG) can create new post with basic info, timeline, clip and some pictures.
  + **Update Post Information**: User can change and update information of post which created.
  + **Comment /Like/Share**: Users can comment/remind/like/share a post to follow and discuss with another users.
  + **Report:** User can report a post or Organization which violate regulation to administrator.
* **Event:**
  + **Create New Event:** ORG can create new event with basic info, timeline, clip and some pictures.
  + **Update Event Information**: ORG can change and update information of post which created.
  + **Comment /Like/Share**: Users can comment/remind/like/share a post to follow and discuss with another users.
  + **Report:** User can report a post or Organization which violate regulation to administrator.
* **Donate**: User can donate money, clothes or others for Organization to charity.
* **Profile:**
  + **Change password:** User can change password to keep security.
  + **Update profile:** User can change or update information.
  + **Manage created posts, reminded posts:** User (ORG) can follow process of post, post reminded. View statistical post had created, reminded.
  + **Sent/Receive message:** Users can send or receive from organizations or admin to discuss about post or problem.

##### Admin Module

* **Login/Logout**: Admin login/logout an account to use or exit system WS
* **Manage user:**
  + Search user:Admin can search with name of user and system will display simple information about that user (name, email,...)
  + Ban/Unban user: admin can set user’s account to block or unblock.
* **Manage Thread:** 
  + Search post :
    - Search with thread title:Admin can search with title of post and system will display all of post relate to keywords.
  + Edit thread status: Admin can set status of a thread: Inactive/active depend on report.
* **Manage Event :**.
  + Search event :
    - Search with event title:Admin can search with title of event and system will display all of event relate to keywords.
  + Edit event status: Admin can set status of a event: Inactive/active depend on report.
* **Manage Organization :**.
  + Search Organization :
    - Search with organization title:Admin can search with name of organization and system will display all of event relate to keywords.
  + Edit organization status: Admin can set status of a organization: Inactive/active depend on report.
* **Manage Donate:** Admin can view users who donated . Also, Admin can view amount of money, numbers of clothes.. to rank for users.
* **Manage Requests Of Users:** 
  + View/Accept/Decline: Admin can View/Accept/Decline requests : raise a ORG, report a post of users.
* **Message:** Admin can view received messages/sent messages.
  + Compose new message: Admin can compose new message to send to user.
  + Delete message: Admin can delete received messages/sent messages.

### Assumptions and Constraints

|  |  |  |
| --- | --- | --- |
| No | Description | Note |
| Assumptions | | |
| 1 | Customer reviewers will get seven days to approve a milestone document. If no comments are received within this time period, it will be considered as approved. | External Interfaces |
| Constraints | | |
| 1 | This project must be completed and delivered before 16/12/2016 | Schedule |
| 2 | In doing project processing, PM must submit report (include 6 reports) on certain date. | Schedule |
| 3 | Software Requirement Specification Document and Project Plan must be completedwithin10dayssince7/09/2016  **Deadline**: 16/09/2016 | Schedule |
| 4 | Design Document (include Architecture Design, Screen Design, Database Design) must be completed within 17 days since 22/09/2016  **Deadline**:20/10/2016 | Schedule |
| 5 | Completed coding activity and have unit test result within 18 days since 20/10/2016  **Deadline**: 18/11/2016 | Schedule |
| 6 | Integration TestPlan (include test plan and test case…) must be completed within20dayssince1/11/2016  **Deadline**: 28/11/2016 | Schedule |
| 8 | Deliver report about User manual, software package and installation guide on 2 days since 6/12/2016  **Deadline**: 10/12/2016 | Schedule |
| 9 | Complete all of document and application before finishingtheprojecton12/12/2016 | Schedule |
| 10 | Project contains 6 members | Resource |

1. Project Description

### Project Objectives

#### Standard Objectives

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metrics | Unit | Committed | Re-committed | Note |
| Start Date |  | 05-09-2016 |  |  |
| End Date |  | 16-12-2016 |  |  |
| Duration | Day | 75 |  |  |
| Team Size | Person | 5 |  |  |
| Billable Effort | Person-day | 375 |  |  |
| Calendar effort | Person-day | 375 |  |  |
| Effort Usage | % | 100 |  |  |

1. Standard Objectives 1

|  |  |  |
| --- | --- | --- |
| Metrics | Unit | Basic for setting Goals |
| Average |
| Customer Satisfaction | Point | 9.5 |
| Leakage | Wdef/mm | 5 |
| Effort Efficiency | % | 95 |
| Timeliness | % | 100 |

1. Standard Objectives 2

#### Specific Objectives

|  |  |  |  |
| --- | --- | --- | --- |
| Metrics | Unit | Basic for setting Goals | |
| Plan | Actual |
| Training technology: MVC, Bootstrap, jQuery, AngularJS, Ajax | Person-day | 15 | 15 |
| Execute group review | Person-day | 8 | 5 |
| Training requirements, process before coding | Person-day | 8 | 5 |

1. Specific Objectives

### Critical Dependencies

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Dependency | Expected delivery date | | Note |
| 1 | This project must be completed and delivered to FPT University. | 16/12/2016 |  | |
| 2 | Project Plan and SRS must be completed and delivered to Supervisor. | 20/09/2016 |  | |
| 3 | User manual, Software Package and Installation Guide must be completed and delivered to Supervisor and FPT University. | 14/12/2016 |  | |
| 4 | Beside Capstone Project, Team members have to joining in Japanese class and Japanese Fundamental Exam class. | 05/09/2016 |  | |

1. Critical Dependencies

### Project Risk

PM identifies risks in the Risk Management Plan. The document is updated to trigger each milestone, each event also. The document is updated weekly by the PM, Risk Management Plan will be notified to all of the stakeholders affected. Status of risk is reported to supervisor at Project Milestones Report.

Reference to WS\_Risk Management Plan\_v1.0\_EN.xlsx

## Project Development Approach

### Project Process

Process of this project is performed follow to Software Development Process of FPT Software.

#### FPT Software Process Model



Figure 1‑1: FPT Software process model

The software lifecycle is broken into *cycles*, each cycle working on a new generation of the product. The FPT Software process divides one development cycle in six consecutive *phases*:

1. Initiation phase
2. Definition phase
3. Solution phase
4. Construction phase
5. Transition
6. Termination

#### Project Life Cycle

Basing on FPT Software process and real-world project, we decided to divide the project into 4 phases: Initiation, Solution, Construction, and Termination:

* **Initiation Phase:** This is the explanatory phase of the project. Project objective and description is described at this stage. The purpose of this phase is to collect and understand business requirements, detail the project plan and agree upon a high level statement of work. Our primary objectives are complete project identification and project plan. After these are completed, the project is checked against the following criteria:
  + Identify business functions of the system
  + Determining the scope, conditions and limitations of the project
  + List the main functions of the system
  + List one or more suitable architecture for the system
  + Identify project risks
  + Complete Report #1, and Report #2
* **Solution Phase**: In this phase, the architecture of the system is designed. The goal is to translate requirements and specification into a technical solution to produce Technical Design.
  + Our *primary objectives* are completeRequirement Specification, Architecture Design and Database Design.
  + Finally, the plan must be provided (including estimates of cost and time) for the construction phase. The plan must ensure proper and accurate based on experience.
  + Complete Report #3 and Report #4
* **Construction Phase**: This is the longest phase of a project life cycle.
  + In this phase, all functions of the system will be installed. The installation will be divided into small stages, each stage of the installation a few functions. The results of each phase will be the release of the module function can be executed.
  + Construction and improvement of products until the final product is ready to deliver to the user. During this phase, all the components and other features of the application is developed and integrated into the product.
  + This phase emphasizes the resource management and control operations to optimize cost, time and quality.
  + Complete software packages and Report #5
* **Termination Phase**: This is the final phase in the life cycle of a project.
  + Their products will be deployed to the client. The feedback received during the transfer process will be recorded and put on the new functional requirements or functionality enhancements in the next version of the product.
  + Phase transfer switch also includes the training system and the new system for the user.
  + Complete software packages and Report #6

### Requirement Change Management

|  |  |
| --- | --- |
| Who logs the change request? | Any team members |
| Who reviews the change request? | PM or who is PM assign |
| Who approves the change request? | PM by default. PTL if:   * Changes to project scope * Changes in delivery plan of project deliverables * Changes to assignment for key roles (PM, PTL) |

1. Requirement Change Management

### Quality Management

#### Defect Prevention Strategy

|  |  |  |
| --- | --- | --- |
| Item (Process/Product) | Strategy | Expected Benefits |
| Requirement missing | List up all of requirement into SRS document. | 10–20% reduction in defect injection rate and about 2% improvement in productivity |
| Careless mistake in Design Document Format/Template wrong | After designing, QA will review Document Format base on checklist review design | Improvement in quality as overall defect removal efficiency will improve; some benefits in productivity as defects will be detected early |
| Use wrong template | Have a meeting to disseminate all template that is used in this project for all member | All member will use right template when do document |
| Coding application does not match with User Requirement. | Develop Team must study about Requirement/Design within 1 weeks since project is assigned.  PM and PTL has responsibility to review task results and explain User Requirement for Develop Team | Coding Application match with User Requirement. |

1. Defect Prevention Strategy

#### Review Strategy

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Review Item | Reviewer | Review Type | Review Method | Completion Criteria |
| Project Plan  Project Schedule  CM Plan | PM,QA, Supervisor | Group review  Group review  One-person review | Use checklist and Self-review |  |
| Business analysis and requirements specification document, Use Case catalog | PM,QA, Supervisor | Group review and One-person review | Use checklist |  |
| Design document, object model | Self-review, PM,QA Supervisor | One-person Review | Use checklist |  |
| Stage plans | PM,QA, Supervisor | One-person review | Use checklist |  |
| Complex/first time generated program specs incl. test cases, interactive diagrams |  | Group review |  |  |
| Source code | Self-review, Peer review, PM, Supervisor | One-person review and Group review | Self-review and use checklist |  |

1. Review Strategy

#### Unit Testing Strategy

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item to be Unit Tested | Unit Test Type | Unit Test Technique | Tool Used | Unit Test Completion Criteria |
| Source Code | White-Box Test | Using unit test case and test script | None | - Number of UTC/KLOC: 20 UTC/KLOC  - Number defects/KLOC: 3-4 defects/KLOC  - Statement coverage: 97%  - Branch coverage: 100%  - Path coverage: 100% |

1. Unit Testing Strategy

#### Integration Testing Strategy

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item to be Integration Tested | Integration Test Type | Integration Test Technique | Tool Used | Completion Criteria |
| Do test by flow of functions and items which have concern each other | Black-Box Test |  | Checklist, Boundary | - Number of UTC/KLOC: 20  - Number of defects/KLOC: 2-3 |

1. Integration Testing Strategy

#### System Testing Strategy

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item to be System Tested | System Test Type | System Test Technique | Tool Used | Completion Criteria |
| Test whole system | Black-Box Test |  | None | -Number of UTC/KLOC: 60  -Number of defects/KLOC: 4-6 |

1. System Testing Strategy

#### Estimates of Defects to be detected

|  |  |  |  |
| --- | --- | --- | --- |
| Review/Testing Stage | Targeted No. of Defects to be detected | % of Defects to be detected | Basic for Estimation |
| Requirements review | 10 | 7.5% | Referenced to similar project estimations |
| Design review | 15 | 11.5% | Referenced to similar project estimations |
| Code review | 30 | 23% | Referenced to similar project estimations |
| Unit Test | 45 | 35.5% | Referenced to similar project estimations |
| Integration Test | 15 | 11.5% | Referenced to similar project estimations |
| System Test | 10 | 7.5% | Referenced to similar project estimations |
| User Acceptance Test | 5 | 3.5% | Referenced to similar project estimations |
| Total | 130 | 100% |  |

1. Estimates of Defects

#### Measurements Program

|  |  |  |  |
| --- | --- | --- | --- |
| Data to be collected | Purpose | PIC | When |
| Size: No. of KLOC | Achieve target | PM | At the end of stages |
| Effort: No. person-day | Monitor and controlling team member to keep plan. | Team members | Daily |
| Quality: No. defects detected | Managing product’s quality. | Reviewer  Tester | Right after the review/test |
| Schedule | Monitor and controlling software developing processing keep plan. | PM | Weekly and at the end of stages |

1. Measurements Program

## Estimation

### Size

This project is performed and must complete all requirements from teacher and FPT University. So size of our project is in Capstone Project limit.

### Effort

The Effort estimation is documented in the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Initiation | Solution | Construction | Termination | Total |
| Effort(person/day) | 82 | 163 | 200 | 65 | **510** |
| Total % budgeted Effort Usage (%) | 100 | 100 | 100 | 100 |  |

1. Effort Estimation

### Schedule

#### Project Milestone & Deliverables

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Stage | Committed Delivery date | Description of Deliverable | Delivery media |
| Initiation | |  | Requirements agreed, Report 1 reviewed | |
| 1 | Develop project idea | 06-09-2016 | Project goals and scope defined, milestone description defined, resource committed | Commit TortoiseGit |
| 2 | Q&A Management Sheet | 12-09-2016 | Criteria: Documentation reviewed | Commit TortoiseGit |
| 3 | Complete User Requirement Specification | 16-09-2016 | Complete User Requirement Specification | Commit TortoiseGit |
| 4 | Submit report no.1 final | 16-09-2016 | Completed report no.1 | Commit TortoiseGit |
| 5 | Project Plan | 15-09-2016 | Criteria: Documentation reviewed | Commit TortoiseGit |
| Solution | |  |  | |
| 1 | Screen Prototype | 02-10-2016 | Criteria: Documentation reviewed | Commit TortoiseGit |
| 2 | Architecture Design | 04-10-2016 | Criteria: Documentation reviewed | Commit TortoiseGit |
| 3 | Screen Design | 06-10-2016 | Criteria: Documentation reviewed | Commit TortoiseGit |
| 4 | Class Design | 04-10-2016 | Criteria: Documentation reviewed | Commit TortoiseGit |
| 5 | Detail Data Design | 07-10-2016 | Criteria: Documentation reviewed | Commit TortoiseGit |
| 6 | Submit progress report 2 | 5-10-2016 | Completed report no.2 | Commit TortoiseGit |
| 7 | Create Test Plan Final | 13-10-2016 | Criteria: Documentation reviewed | Commit TortoiseGit |
| 8 | Submit Report no.3 Final | 14-10-2016 | Completed report no.3 | Commit TortoiseGit |
| Construction | |  | Product developed & tested and released to supervisor, documentation reviewed. | |
| 1 | Complete Coding and Unit Test | 14-11-2016 | Source code  Acceptance criteria: Product unit tested | Commit TortoiseGit |
| 4 | Complete Testing | 28-11-2016 | Completed Test | Commit TortoiseGit |
| 5 | Submit report no.5 Final | 02-12-2016 | Completed report no.5 | Commit TortoiseGit |
| 6 | Submit the last document and CD source code | 10-12-2016 | Final Documents and Source Code | Commit TortoiseGit |
| Termination | |  | Project post-mortem is conducted, Project assets archived and released to supervisor | |
| 1 | Lesson learned | 05-12-2015 | Criteria: Completed | Commit TortoiseGit |
| 2 | Complete Presentation Slide | 07-12-2016 | Criteria: Completed | Commit TortoiseGit |
| 3 | Present capstone project | 16-12-2016 | Criteria: Completed | Commit TortoiseGit |
| 4 | Project Complete | 16-12-2016 | Criteria: Completed | Commit TortoiseGit |

1. Project Milestone and Deliverables

#### Activity Schedule

The detail project schedule is available in file WS\_ProjectSchedule\_v1.0\_EN.mpp. The Project Schedule is weekly updated by the Project Manager.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Activity | Start date | End date | Responsible |
| Defect Prevention | | | | |
| 1 | Training coding convention C#, Javascript, CSS | 12-09-2016 | 16-09-2016 | Lê Hồng Nhiên |
| 2 | Training for Q&A and tester to use checklist | 06-10-2016 | 10-10-2016 | Tạ Ngọc Duy |
| **Quality Control** | | | | |
| 1 | Group review requirement | 14-10-2016 | 16-10-2016 | Đào Trọng Nghĩa |
| 2 | Group review design | 07-10-2016 | 09-10-2016 | Tạ Ngọc Duy |
| 3 | Group review coding | 16-11-2016 | 24-11-2016 | Lê Hồng Nhiên |
| Project Tracking | | | | |
| 1 | Solution: Milestone review meeting | 15-10-2016 | 16-10-2016 | Đào Trọng Nghĩa |
| 2 | Construction: Milestone review meeting | 3-12-2016 | 04-12-2016 | Đào Trọng Nghĩa |
| 3 | Transition: Milestone review meeting | 10-12-2016 | 11-12-2016 | Đào Trọng Nghĩa |
| Configuration Management | | | | |
| 1 | Baseline code | 17-10-2016 | 20-10-2016 | Đào Trọng Nghĩa |
| 2 | Base line test report, test case and test plan | 22-10-2016 | 29-10-2016 | Đỗ Văn Tuấn |
| Q&A | | | | |
| 1 | Final Inspection: Report 1 | 05-09-2016 | 16-09-2016 | Đào Trọng Nghĩa |
| 2 | Final Inspection: Report 2 | 19-09-2016 | 05-10-2016 | Đào Trọng Nghĩa |
| 3 | Final Inspection: Report 3 | 06-10-2016 | 14-10-2016 | Đào Trọng Nghĩa |
| 4 | Final Inspection: Report 4 | 17-10-2016 | 15-11-2016 | Đào Trọng Nghĩa |
| 5 | Final Inspection: Report 5 | 1-11-2016 | 02-12-2016 | Đào Trọng Nghĩa |
| 6 | Final Inspection: Report 6 | 05-12-2016 | 10-12-2016 | Đào Trọng Nghĩa |

1. Activity Schedule

### Resource

Specified as in the section 4.2. [Project Team](#_Project_team)

### Infrastructure

|  |  |  |  |
| --- | --- | --- | --- |
| Item | Description | Expected Availability by | Note |
| Development Environment | | | |
| Operating System | Window 8.1, 10 (32 bit, 64 bit) |  |  |
| Browser | Chrome (40 or above), Firefox (30 or above), Cốc Cốc |  |  |
| Development language | .NET C# |  |  |
| Technology | | | |
| Development language | .NET C#, MVC Model, AngularJS |  |  |
| Database | SQL Server 2008 |  |  |
| Hardware Requirement | | | |
| Hardware Configuration | 2GB workspaces on server |  |  |
| Equipment & Tools | | | |
| Source Version Control | TortoiseGit | Definition stage |  |
| Task Tracking | MS Project Professional 2013 | Initiation stage |  |
| SRS | Microsoft Office Word 2013, Microsoft Office Excel 2013, Microsoft Office Visio 2013 | Initiation stage |  |

1. Infrastructure

### Training Plan

|  |  |  |  |
| --- | --- | --- | --- |
| Training Area | Participants | Duration | Waiver Criteria |
| **Technical** | | | |
| .NET MVC5 | Team | 1 week | Mandatory |
| AngularJS, jQuery | Team | 1 week | Mandatory |
| Bootstrap | Team | 1 week |  |
| **Process** | | | |
| Quality system |  | 3 hours | If already trained |
| Configuration management |  | 2 hours | If already trained for CC. For others, on-the-job training |
| Group review |  | 2 hours | If already trained |
| Defect prevention |  | 2 hours | Mandatory |

1. Training Plan

### Finance

Because this project is non-business, it is a Capstone Project at FPT University. So we do not estimate about finance.

## Project Organization

### Organization Structure

Figure 1-2: Organization Structure

### Project Team

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Role | Responsibility | Full name | Effort (%) | Start date | End date |
| PM | Have overall responsibility of the project:  - Project planning and scheduling  - Task assignment and tracking processing  - Review documents  - Reporting to supervisor | Đào Trọng Nghĩa | 50 | 05-09-2016 | 15-12-2016 |
| PTL | PTL is responsible for the technical project execution | Lê Hồng Nhiên | 50 | 05-09-2016 | 15-12-2016 |
| Programmer #1 | - Study technique (C#, MVC, AngularJS, etc.)  - Coding functions and modules of system.  - Peer-review source code of others members | Đỗ Văn Tuấn,  Hoàng Anh Tuấn,  Tạ Ngọc Duy,  Lê Hồng Nhiên,  Đào Trọng Nghĩa | 100 | 05-09-2016 | 15-12-2016 |
| Programmer #2 | - Support coding functions and modules of system. | Lê Hồng Nhiên | 50 | 05-09-2016 | 15-12-2016 |
| Programmer #3 | - Support coding functions and modules of system. | Hoàng Anh Tuấn | 100 | 05-09-2016 | 15-12-2016 |
| Test Leader | - Create test plan, test case, test report, quality report  - Execute test. | Đỗ Văn Tuấn | 100 | 1-11-2016 | 15-12-2016 |
| Tester | - Support creating test plan, test case, test report, quality report  Execute test. | Tạ Ngọc Duy | 100 | 1-11-2016 | 15-12-2016 |
| Design  Lead | - Create screen design, prototype  - Review design of others member | Tạ Ngọc Duy | 100 | 05-09-2016 | 15-12-2016 |
| Designer #1 | - Support creating screen design | Đào Trọng Nghĩa | 50 | 05-09-2016 | 15-12-2016 |

1. Project Team description

The detail of Human resource budget allocation over the whole project life is in the below table:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Role | Name | W2-  Sep | W3-  Sep | W4-  Sep | W1-  Oct | W2-  Oct | W3-  Oct | W4-  Oct | W1-  Nov | W2-  Nov | W3-  Nov | W4-  Nov | W1-  Dec | W2-  Dec | W3-  Dec | Total (pd) |
| PM/Designer | Đào Trọng Nghĩa | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 80 |
| PTL/Dev | Lê Hồng Nhiên | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 80 |
| Dev/Tester | Hoàng Anh Tuấn | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 80 |
| Test Leader/Dev | Đỗ Văn Tuấn | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 80 |
| Tester/Designer/Dev | Tạ Ngọc Duy | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 80 |
| **Total** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **400** |

1. Human Resource Budget Allocation

### External Interfaces

#### FPT University’s Interfaces

|  |  |  |  |
| --- | --- | --- | --- |
| Department | Contact Person  (name-position) | Contact address  (email, telephone) | Responsibility |
| Teacher | Nguyen Van Sang | [SangNV@fpt.edu.vn](mailto:SangNV@fpt.edu.vn) | - Review and accept documents during project  - Review and accept products of the project.  - Resolve escalated issues and receive project reports. |
| Training Department |  | [acad.hn@fpt.edu.vn](mailto:acad.hn@fpt.edu.vn) | Management course of student |

1. FPT University’s Interfaces

## Communication and Reporting

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Communication Type | Method/Tool | When | Information | Participants/ Responsible |
| Project Task Tracking | | | | |
| Task scheduling | MS Project Professional 2013 | At the beginning of every stage, and weekly  Refinement and rescheduling as necessary |  | PM |
| Task assignment | MS Project Professional 2013 | Weekly |  | PTL |
| Task status reporting | Daily Report | Daily |  | Project Team members |

|  |
| --- |
| Project Meeting |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Kick-off Meeting | Face to face | Initiation stage | Project introduction; Project plan review; Risk identification; stakeholders identify. | PM, Project Team Members |
| Project Progress Review Meetings | Face to face | Weekly and on event | Communicate project status  Communicate and resolve any open issue, risks, and changes  Discuss any suggested improvement | PM, Project Team Members |
| Milestone Meetings | Face to face | 5 days after the completion of stages: Definition, Solution & Construction | Project objective review, evaluate project performance (quality, schedule, effort), Causal analysis, update project plan for next stage | PM, Project Team Members, QA, Supervisor |
| Transfer/Sharing of project documentation/information | TortoiseGit | When available | All project documentation and information | PM, Project Team Members, QA |

|  |
| --- |
| Supervisor Communication and Reporting: |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Report | Agreed FPT Software and FU standard format | Daily | Project status report, Issue requiring clarifications, escalation, if any | PM |
| Project Meetings with supervisor | Face to face | 13h45 Wednesday, Weekly | As above | PM |
| Requirement gathering/clarification | Face to face meeting | During requirement analysis phase | As in Q&A list | PM |

|  |
| --- |
| Communication with Supervisor |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Review Project Plan & Project schedule | By attend project meeting | Significant changes to WO, PP and Project schedule (scope, objectives Organization, HR, major milestone, deliverables ) |  | PM |
| Project Progress Review | By email and/or via Operation meeting at Group/Division level | Weekly | Project status report, Issue requiring clarifications, escalation, if any | PM |
| Project Milestone Review | By email and via project milestone review meeting | End of every stage | Project objective review, evaluate project performance (quality, schedule, effort), Causal analysis, update project plan for next stage | PM |

1. Communication and Reporting Plan

# SOFTWARE REQUIREMENT

## Introduction

### Purpose

This document will provide the describing of general requirement and non-functional requirements of our WS system. It also describes the requirement scope of each phase in the project.

### Definition and Acronyms

|  |  |
| --- | --- |
| Terms | Description |
| WS | WingS |
|  |  |

Table 1‑1: Definition and Acronyms

## User Requirement Specification

### Business Process Overview

WS is a website. Therefore, that system will implement the following processes:

#### Accept/Decline Request workflow

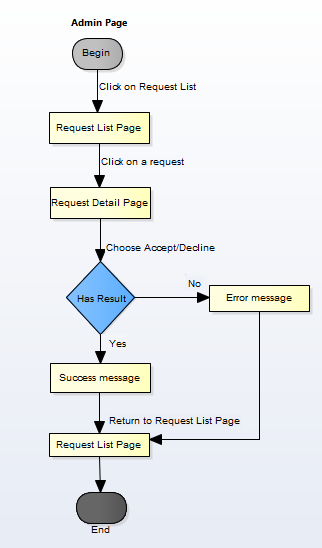


Figure 1‑1: Accept/Decline Request workflow

Brief description: Admin can use this function. Steps:

* Step 1: Admin Click on Request List.
* Step 2: System return Request list page.
* Step 3: Admin click on a Request.
* Step 4: System return Request detail page.
* Step 5: Admin click on button Accept/Decline.
* Step 6: System return success message or error message.
* Step 7: System return Request list page.

#### Ban/Unban User workflow

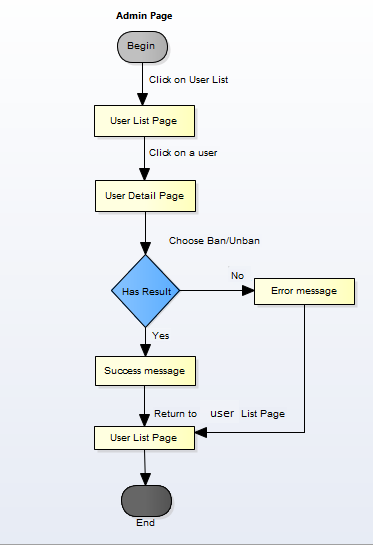


Figure 1‑2: Ban/Unban User workflow

Brief description: Admin can use this function. Steps:

* Step 1: Admin click on users list.
* Step 2: System return users list page.
* Step 3: Admin click on a user
* Step 4: System return user profile page.
* Step 5: Admin click on button Ban/Unban
* Step 6: System return success message or error message.
* Step 7: System return users list page.

#### Report a Post workflow

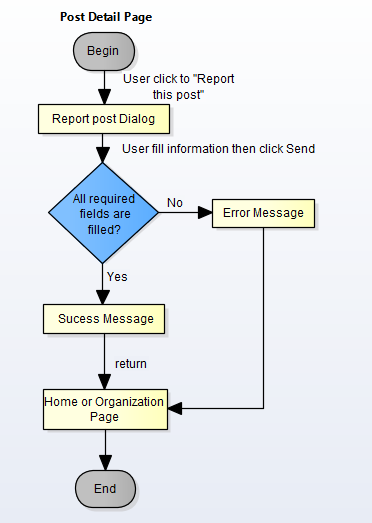


Figure 1‑3: Report a Post workflow

Brief description: Member can use this function. Steps:

* Step 1: User click to “report this post”.
* Step 2: System return “Report this post” dialog.
* Step 3: User fill information and click “Send”
* Step 4: System return success message or error message.
* Step 5: System return Home page or organization page.

#### Add slider workflow

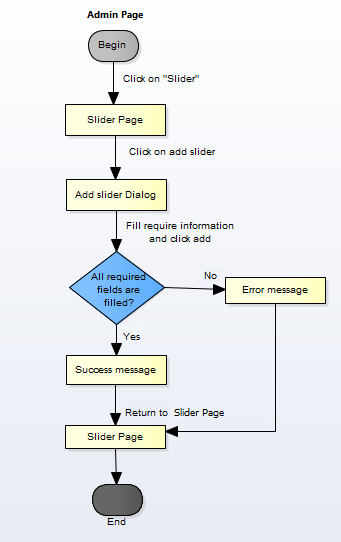


Figure 1‑4: Add slider workflow

Brief description: Admin can use this function. Steps:

* Step 1: Admin click on Slider.
* Step 2: System return slider page.
* Step 3: Admin click on button add slider.
* Step 4: System return dialog “Add Slider”.
* Step 5: Admin fill information and click “Add”
* Step 6: System return success message or error message.
* Step 7: System return slider page.

#### Send message workflow

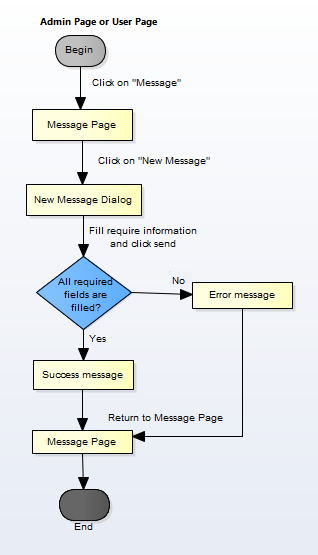


Figure 1‑5: Send message workflow

Brief description: Admin or user can use this function. Steps:

* Step 1: User click on Message.
* Step 2: System return message page.
* Step 3: User click on “new message”.
* Step 4: System return dialog “New Message”.
* Step 5: User fill information and click “Send”
* Step 6: System return success message or error message.
* Step 7: System return message page.

#### Register workflow

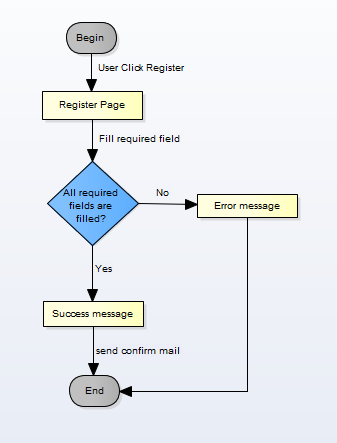


Figure 1‑6: Register workflow

Brief description: Guest can use this function. Steps:

* Step 1: User click register.
* Step 2: System return register page.
* Step 3: User fill all required fields and click Register or press Enter.
* Step 4: System return complete message.
* Step 5: System send confirm mail to user’s entered email.

#### Search workflow

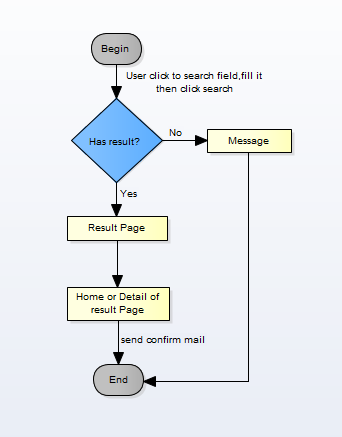


Figure 1‑7: Search workflow

Brief description: Guest and Member can use this function. Steps:

* Step 1: User input keyword into search text box and click Search or press Enter
* Step 2: System return search results or message if found nothing.
* Step 3: User can click return to Home page or click to result to Detail Page of the result.

#### Comment workflow

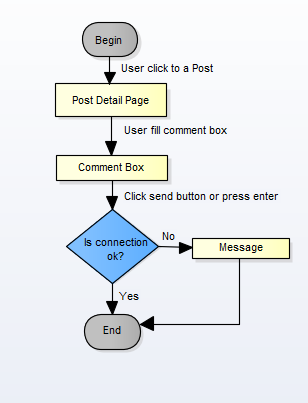


Figure 1‑8: Comment workflow

Brief description: Member can use this function. Steps:

* Step 1: User click on a post.
* Step 2: System return Post detail page.
* Step 3: User fill comment box.
* Step 4: User click send.

#### Login workflow

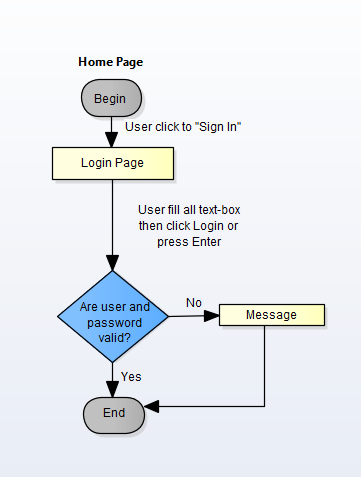


Figure 1‑9: Login workflow

Brief description: Guest can use this function. Steps:

* Step 1: At Home page click to “Sign in”.
* Step 2: System display Login Page.
* Step 3: User fill user name and password textbox then click login button.
* Step 4: System show message if login failed. System return Home page if login success

#### Create a Post workflow

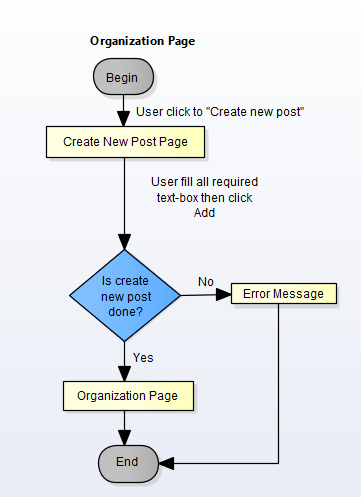


Figure 1‑10: Create a post workflow

Brief description: Member can use this function. Steps:

* Step 1: At organization user click to “Create new post”.
* Step 2: System return Create New Post page.
* Step 3: User fill all required textbox then click to “Add” button
* Step 4: System return message.
* Step 5: System return to Organization Page if success.

#### Join an organization workflow

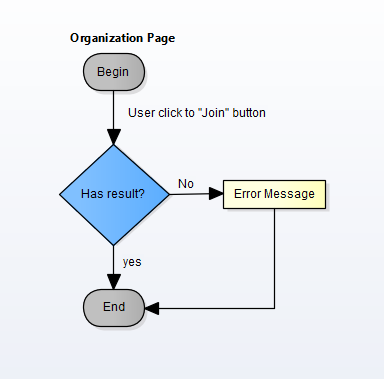


Figure 1‑11: Join an organization workflow

Brief description: Member can use this function. Steps:

* Step 1: At Organization Page click on “Join” button.
* Step 2: System return success message or error message.
* Step 3: System return Organization Page.

#### Donate workflow

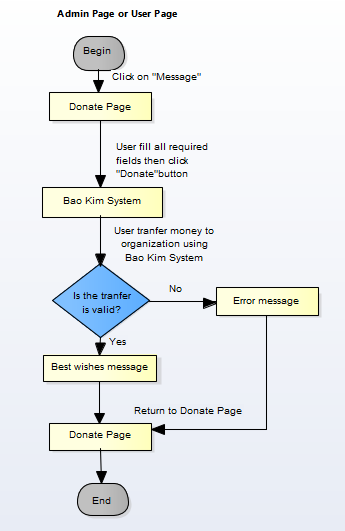
****

Figure 1‑12: Donate workflow

Brief description: Member can use this function. Steps:

* Step 1: At Home Page click on “Donate” button.
* Step 2: System display Donate Page.
* Step 3: User fill all required fields then click to Donate button.
* Step 4: System will automatically linked with Bao Kim
* Step 5: User transfer money to organization by using Bao Kim System.
* Step 6: System display message.
* Step 7: Return to Home Page.

### Product Features

WS is a website, user only connect to internet to use website. System provides these main functions:

#### Client features

* **Register:** User can register an account and login to use all features of WS.
* **Login/Logout**: Users login/logout an account to use or exit system WS.
* **Search**: User and guest can search with title of posts, name of organization, and system will display all of post relate to keywords.
* **Search Advanced**: Search (Event, Discussion, Organization, User) follow location.
* **Raise Request:** User can request to create an Organization. Admin can accept or refuse.
* **Post:**
  + **Create New Post:** User (ORG) can create new post with basic info, timeline, clip and some pictures.
  + **Update Post Information**: User can change and update information of post which created.
  + **Comment/Remind/Like/Share**: Users can comment/remind/like/share a post to follow and discuss with another users.
  + **Report:** User can report a post or Organization which violate regulation to administrator.
* **Donate**: User can donate money, clothes or others for Organization to charity
* **Profile:**
  + Change password: User can change password to keep security.
  + Update profile: User can change or update information.
  + Manage created posts, reminded posts: User (ORG) can follow process of post, post reminded. View statistical post had created, reminded.
  + Sent/Receive message: Users can send to or receive from other users to discuss about events or activity.

#### Admin features

* **Login/Logout**: Admin login/logout an account to use or exit system WS
* **Manage user:**
  + Search user:Admin can search with name of user and system will display simple information about that user (name, email,...)
  + Ban/Unban user: admin can set user’s account to block or unblock.
* **Manage Post:** 
  + Search post :
    - Search with post title:Admin can search with title of post and system will display all of post relate to keywords.
    - Search with name of ORG: Admin can search with name of ORG and system will display all of post relate to keywords.
  + Edit Post status: Admin can set status of a Post: Inactive/active depend on report.
* **Manage Donate:** Admin can view users who donated . Also, Admin can view amount of money, numbers of clothes.. to rank for users.
* **Manage Requests Of Users:** 
  + View/Accept/Decline: Admin can View/Accept/Decline requests : raise a ORG, report a post of users.
* **Manage Slider:** Slider list includes images what running at homepage.
  + Add new image: Admin can add new image to slider
  + Inactive/active: Admin can set image’s status to inactive or active on slider.
  + Delete image: Admin can delete image.
* **Message:** Admin can view received messages/sent messages.
  + Compose new message: Admin can compose new message to send to user.
  + Delete message: Admin can delete received messages/sent messages.

### User characteristic

WS makes connections that enable people to help themselves and each other. Website allows users (person or organizations) create charity topics or raise fund project as long as it follows our rules to help other people. Backers can also take part in a project by donating money through a single donation or supporting project as a regular donor publicly or privately, not to profit financially. Instead, project creators offer rewards to thank backers for their support. We hope our project can help as much people as we can in our country. If users want to use all function, they must be login.

There are 2 main actors that system support:

* **Member:** Have account on system, they can use all functions of website. For example: Searching, create post, donate etc…
* **Administrator:** They can use all functions of normal member and have an Administrator control panel to manage all configuration of system.

### Functional Requirements

#### Common Function

**Access right: Guest and member**

This function is used to by guest and member. Guest or Member can search for their needs, results may be organization, post, discussion or members’ name.

#### View

**Access right: Guest and member**

This function is used to view an overview of the post, event. They can read information then decided join that event, donate items… or not.

#### Create Thread

**Access right: Member**

This function is used to create new post. In this screen, users input thread title, add media, information about thread… and click on button create.

#### Create Event

**Access right: Member**

This function is used to create new event. In this screen, users input event name, add media, information about event… and click on button create.

#### Send/Receive messages

**Access right: Member and Administrator.**

This function is used to send and receive messages between users and users, users and admin. The system will display a Message Screen for users and admin to view message. In this screen, users can compose new message, search, filter and delete received messages/sent messages.

#### Management profile

**Access right: Member**

This function is used to manage profile. The system will display a Profile Screen. In this screen, users can update, modify their information such as: name, password, age, phone number and choose which information would be public; users also can view many information such as recent joined event, donated money, user class,…

#### Donate

**Access right: Member and Administrator.**

This function is used to donate money for organization. The system will display a Donate Screen that contain amount of money, information of donor, donate to which organization,…  
then click Donate button. The system will display another page that linked with Bao Kim to help user transfer money to organization through WS system.

#### Management Member’s account

**Access right: Administrator**

This function is used to manage member’s account. The system will display a Management Member Screen. In this screen, admin can view information about member, status of member (active or Inactive). When a member violates some rules of website, admin can block these member’ account temporarily, then these account cannot continue to use some functions of website. Locked account will be unlocked when lock is expired or they can send message to admin to request unban. If these’ account violate rules one more time, Admin can lock account permanently.

Warn: If member violate the rules (reported by others), admin can send message to warn this people from stopping violation and delete/modify violated post. If member still violate the rules, admin will ban them.

#### Management Post

**Access right: Member(Organization), Administrator**

This function is used to manage post. User (ORG) can follow process of post, post reminded. View statistical post had created, reminded. Admin can delete a post that violate the rules.

#### Management Overview

**Access right: Administrator**

This function is used to manage report. The system will display a Dashboard Screen to have overview all of system. In this screen, admin can view information about user and post such as the number of accessing, new users, active users, inactive users, number of post, amount of donate…

#### Management Request

**Access right: Administrator**

This function is used to manage user request. The system will display a Request screen that contains all users’ request. Admin will check that request then decide accept or decline request.

### Non-functional Requirement

The system has to satisfy these principles:

* **Learnable**: The user interface must be easy to learn, even at the first time looking at the screen, user may immediately understand the function of each button or icon.
* **Efficiency**: The user interface must minimize the number of steps that the system takes to complete its task.
* **Memorable**: System contains some complex screens. Therefore, interface should be easier to use each time the user interacts with it.
* **Visibility**: Important information (example: text in nodes) should be clearly visible.
* **Consistency**: Like items should always be displayed and act the same way through the entire application.
* **Easy to maintain and upgrade**
* **Ensure data security capabilities, high performance**
* **Scalability system**

## Software Requirement Specification

### OVERALL DESCRIPTION

#### **System overview**

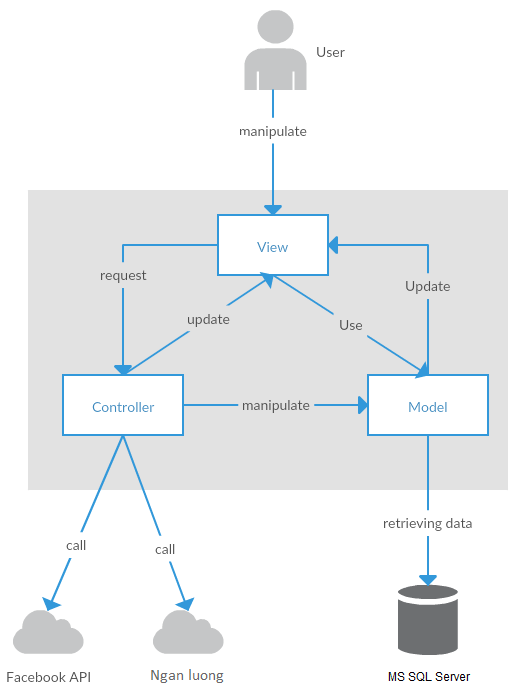


Figure 2-1: System Overview

#### **System Use Case**

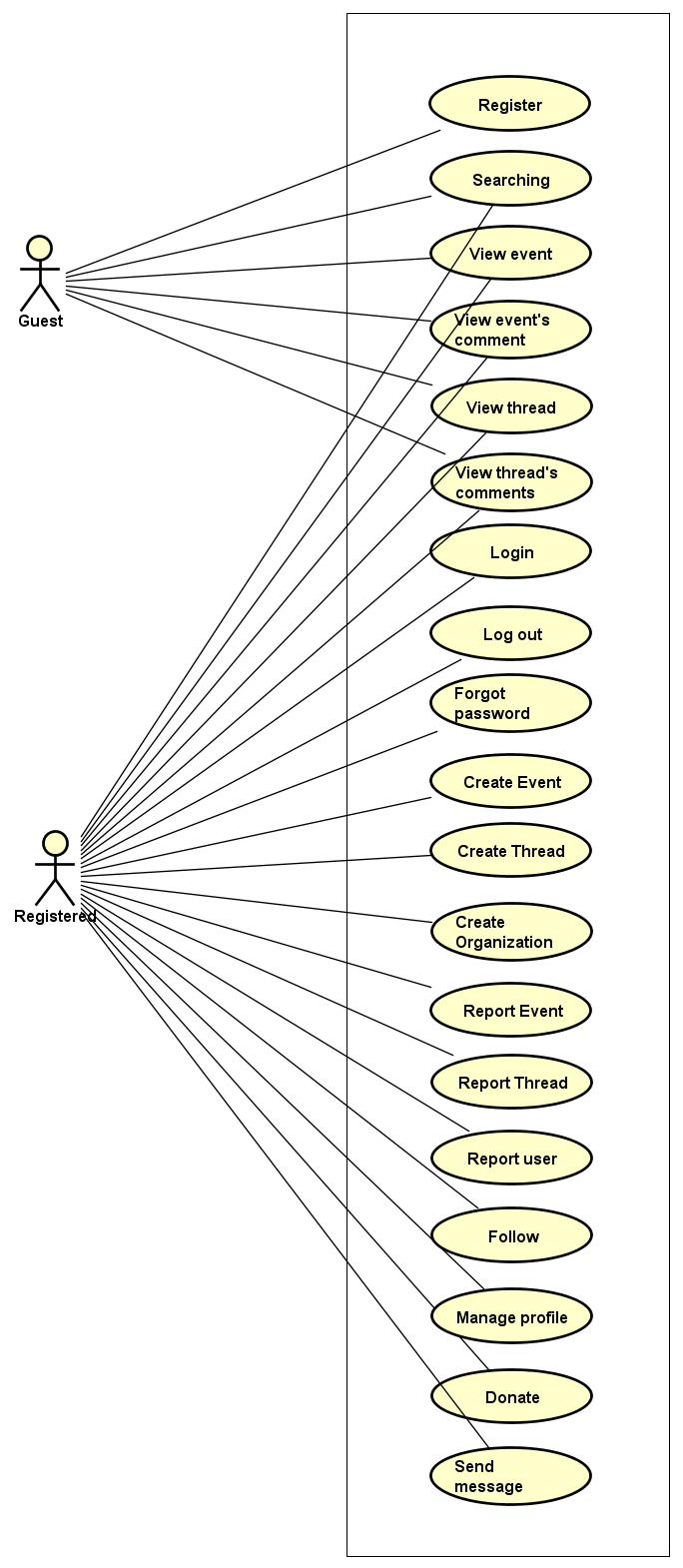


Figure 2-2: User use case diagram of WS system

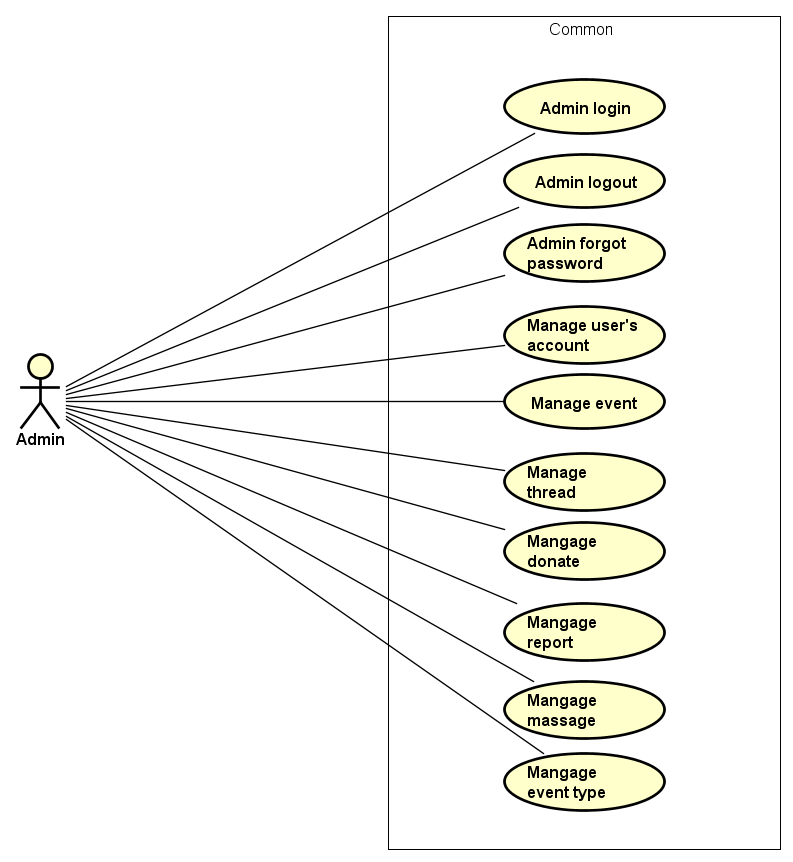
****

Figure 2-3: Admin use case diagram of WS system

#### **Product Features**

|  |  |  |  |
| --- | --- | --- | --- |
| UC No. | Group Of Functions | Function | Glossary |
| Guest/Registered | | | |
|  | Common | Searching (User, Event, thread or organization) |  |
|  | View event | View event |  |
|  | View event’s comments |  |
|  | View thread |  |
|  | View thread’s comments |  |
| Guest | | | |
|  | Common | Register |  |
| Registered | | | |
|  | Common | Login |  |
|  | Logout |  |
|  | Forgot password |  |
|  | Create event |  |
|  | Create thread |  |
|  | Create Organization |  |
|  | Report Event |  |
|  | Report Thread |  |
|  | Report User |  |
|  | Follow Event |  |
|  |  | Donate |  |
|  | View Personal page | View other profile |  |
|  | View profile |  |
|  | View number created thread |  |
|  | View participated event |  |
|  | View point |  |
|  | Edit profile |  |
|  | Message | Send message |  |
|  | Group message |  |
|  | Receive message |  |
|  | Comment | View list comment |  |
|  | Post comment |  |
|  | Reply comment |  |
|  | Delete comment |  |
|  | Chat room | Go to room chat |  |
|  | View all other’s message |  |
|  | Receive message |  |
|  | Send message |  |
| Administrator | | | |
|  | Common | Admin login |  |
|  | Admin logout |  |
|  | Admin forgot password |  |
|  | Manage user's account | View list user |  |
|  | View user profile |  |
|  | Activate user's account |  |
|  | Deactivate user's account |  |
|  | View user’s dashboard |  |
|  | Manage event | View event’s dashboard |  |
|  | Active event |  |
|  | De-active event |  |
|  | View event detail |  |
|  | Search event |  |
|  | Manage donate | View list backing |  |
|  | Search backing |  |
|  | View backing's information |  |
|  | Export list backings |  |
|  | Manage Thread | View list thread |  |
|  | Activate thread |  |
|  | Deactivate thread |  |
|  | Search thread |  |
|  | View thread detail |  |
|  | Manage Messages | View list messages |  |
|  | Receive message |  |
|  | Send message |  |
|  | Delete message |  |
|  | Delete message |  |
|  | Manage Organization | View list Organization |  |
|  | Activate thread |  |
|  | Deactivate thread |  |
|  | Manage Reports | View list reports |  |
|  | View report detail |  |
|  | Change report's status |  |

Table 2-2: List of use case diagram

##### User Characteristics

WS website towards to anyone wants to create an event on WingS as long as it follows our rules. Event creators set a donation goal and timeline. If have interested in event, they can donate this event by money or efforts. Donors are supporting event and help it to success and help as much people as they can, not to profit financially. Instead, WS system offer rewards to thank donors for their support by activities point. WS has three main objective users:

* **Guest:** Users did not register, they can use some function like Register, view event or thread, Search, etc...
* **Registered**: Users have account on WS, they can use almost functions of WS for user like Login, Logout, Create, Edit event, Comment, Report, etc…
* **Administrator:** They can use all functions of normal member and have an Administrator control panel to manage all information, event, user and configuration of WS.

##### User Documentation

|  |  |
| --- | --- |
| Name | Description |
| User Guide | Provide detailed explanation about the system, screens and guide users how to use all features of WS. |

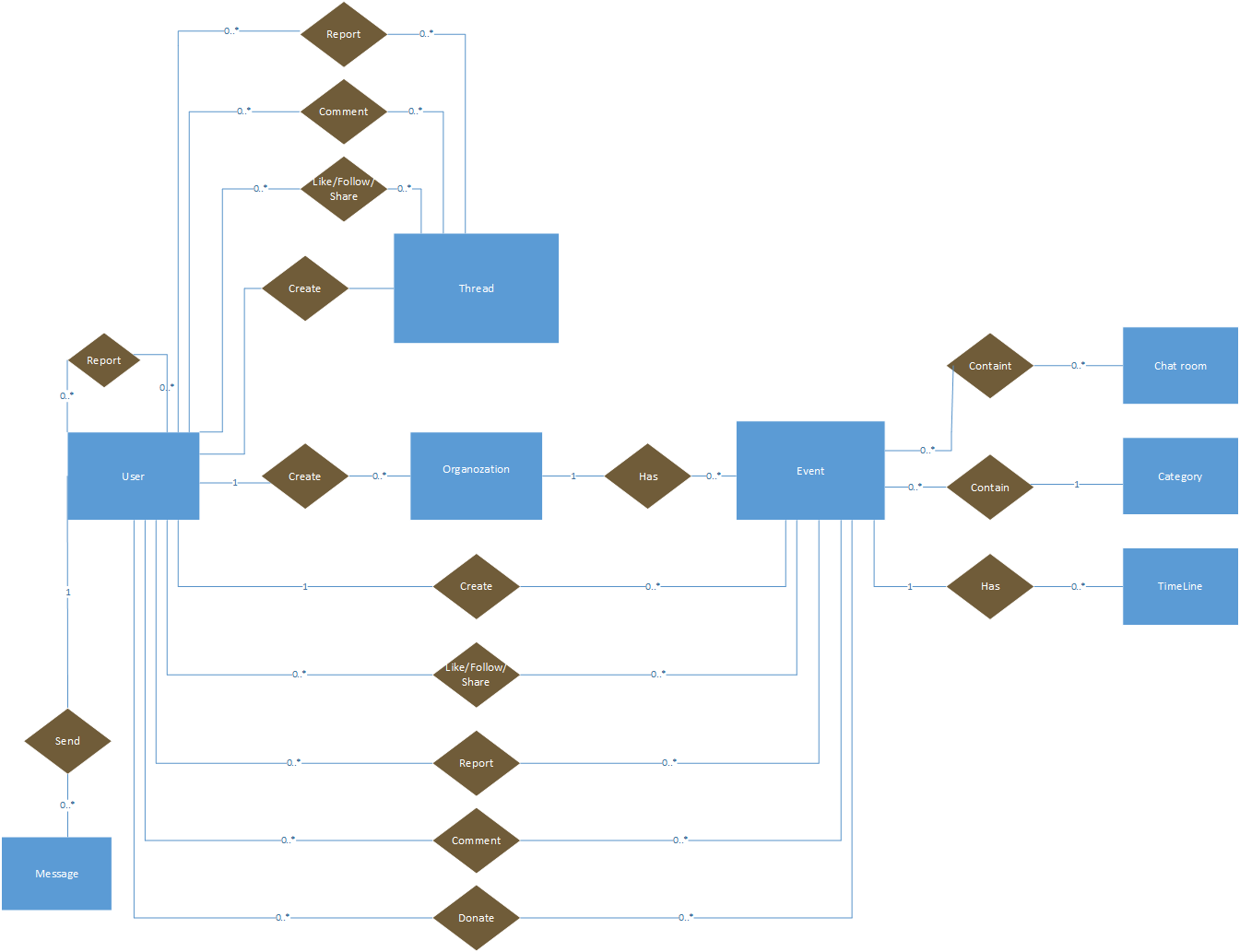
Table 2-3: User Documentation

##### Assumption

* Receive the guidance and good support of teacher.
* No one of member got sick in the software development process.
* In the development process, no device got fail.
* Every report, task meets the deadline.
* Team member do not conflict in the work process.

#### **Entity Relation Model**

##### Entity-Relationship Diagram



**Figure 2-3:** Entity Relationship Diagram

##### Entity Detail

###### WS\_User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
|  | UserID | Int | Y | User’s ID |
|  | UserName | String | Y | User’s name |
|  | Password | String | Y | User’s password |
|  | AccountType | String | Y | User’s type (admin or normal user) |
|  | IsActive | Bit | Y | User’s active status (inactive, active) |
|  | IsVerify | Bit | Y | User’ email verify status |
|  | CreatedDate | Date | Y | User’s created date |
|  | LastLogin | Date | Y | User’s last login |
|  | Email | String |  | User’s email |
|  | VerifyCode | String |  | User’s verify code |

Table 2-4: WS-User

###### Organization

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
|  | OrganizationID | Int | Y | Organization’s |
|  | OrganizationName | String | Y | Organization’s Name |
|  | EOrganizationName | String |  | EOrganization’s Name |
|  | Introduction | String |  | Organization’s introduction |
|  | LogoUrl | String |  | Organization’s image link |
|  | Phone | String |  | Organization’s phone |
|  | Email | String |  | Organization’s email |
|  | Address | String |  | Organization’s address |
|  | Status | Bit | Y | Is Organization expired |
|  | Point | Int | Y |  |

Table 2-5: Organization

###### Thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | ThreadIdID | Int | Y | Thread’s ID |
| 2 | UserId | Int | Y | UserID who created thread |
| 3 | Title | String | Y | Thread’s title |
| 4 | Etitle | String | Y | Thread’s ETitle |
| 5 | [Content] | String | Y | Thread’s content |
| 6 | VideoUrl | String | Y | Thread’s video link |
| 7 | CreatedDate | Date | Y | Thread’s created date |
| 8 | UpdatedDate | Date | Y | Thread’s updated date |
| 9 | Status | Bit | Y | Thread’s status: active or banned |

Table 2-5: Thread

###### Event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | EventID | Int | Y | Event’s ID |
| 2 | CreatorID | Int | Y | UserID who created event |
| 3 | EventType | String | Y | Event’s type |
| 4 | EventName | String | Y | Event’s name |
| 5 | EEventName | String | Y | EEvent’s name |
| 6 | Created\_Date | Date | Y | Event’s created date |
| 7 | Start\_Date | Date | Y | Event’s start date |
| 8 | Finish\_Date | Date | Y | Event’s finish date |
| 9 | UpdatedDate | Date | Y | Event’s updated date |
| 10 | Contact | String | Y | Contact of event’s creator |
| 11 | ShortDescription | String | Y | Event’s short description |
| 12 | Description | String | Y | Event’s description |
| 13 | Location | String | Y | Event’s location |
| 14 | VideoUrl | String | Y | Event’s video link |
| 15 | ExpectedMoney | Float | Y | Amount Money which event expected |
| 16 | TotalPoint | Int | Y | Total point of event |
| 17 | Status | Bit | Y | Event’s status |

Table 2-5: Event

###### Event Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1. 1 | EventTypeID | Int | Y | Event type’s ID |
| 1. 2 | EventName | String | Y | Event type’s name |
|  | [Content] | String | Y | Event type’s description |

Table 2-6: Event type

###### Event Timeline

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1. 1 | TimelineID | Int | Y | Timeline’s ID |
| 1. 2 | EventId | Int | Y | Event’s id |
| 1. 3 | [Content] | String | Y | Timeline’s contents |
|  | FromDate | Date | Y | Timeline’s start date |
|  | ToDate | Date | Y | Timeline’s end date |
|  | Status | Bit | Y | Timeline’s status |

Table 2-8: Event Timeline

###### Event chat room

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
|  | PublicRoomId | Int | Y | Conversation’s ID |
|  | ConnectionId | Int | Y | Conversation’s creator id |
|  | EventId | Int | Y | Conversation’s receiver |

Table 2-13: Event chat room

###### Message table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1. 1 | MessageID | Int | Y | Message’s ID |
| 1. 2 | ConversationID | Int | Y | Message’s conversation id |
| 1. 3 | UserID | Int | Y | Message’s user id |
|  | CreatedDate | Date | Y | Message’s sent time |
|  | Content | String | Y | Message’s content |
|  | Status | Bit | Y | Message’s status |

Table 2-11: Message table

### FUNCTIONAL SPECIFICATION

#### Business Rules

|  |  |
| --- | --- |
| **No** | **Description** |
|  | Search textbox cannot be empty |
|  | Content is unlimited characters |
|  | Chat box cannot be empty |
|  | Subject is limited to 1000 characters |
|  | Name cannot be empty |
|  | If user accesses to nonexistent page, WS displays error message “Error” |
|  | Email must content characters ‘@’ and ‘.’ |
|  | Date of birth must be chosen |
|  | Displayed date is mm/dd/yyyy format |
|  | Date time input format is datetime |
|  | Number is rounded to 2 decimal number |
|  | Unit of money is VND |
|  | Displayed money is rounded to 2 decimal number |
|  | Allowed file extension: .jpg, .png, .bmp |
|  | File size is limited to 100 MB |
|  | Message displayed like a Gmail format |
|  | Rank is calculated depending on point which gain from donating event or creating thread |

Table 2-13: Business rules

#### Use Cases

##### Guest/ Registered Group Function

###### Common Module

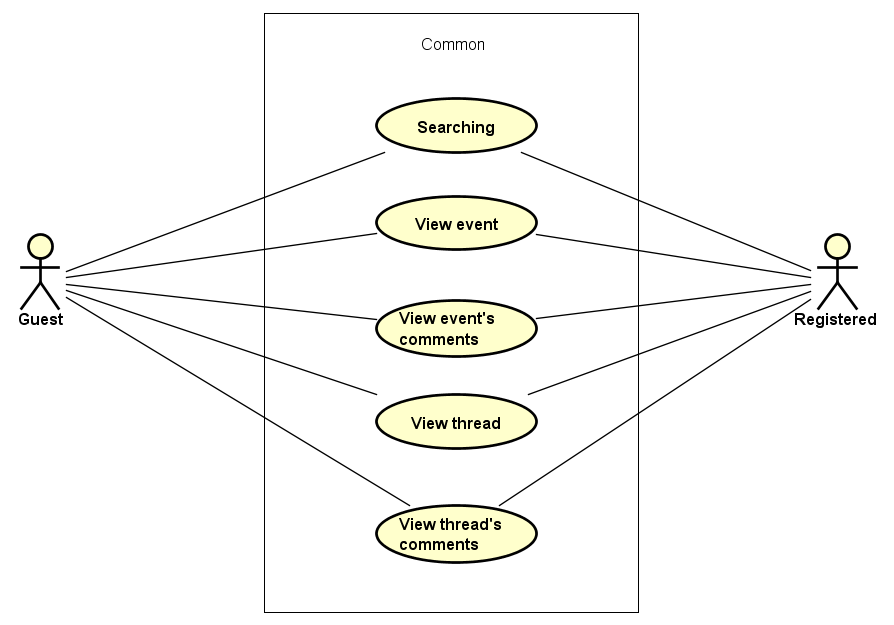


Figure 3-1: Common module use case

UC001-Searching

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC001 | **Version** | 1.0 |
| **Use Case Name** | | Search event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user  Guest | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to search existing information about Search existing information about User, Account, Event, Thread or  Organization | | |
| **Preconditions:** | | 1. WS website is available 2. User browsed WS website | | |
| **Trigger:** | | User enters search phrase, choose the search type (User, Account, Event, Thread or  Organization) and click Search or press Enter | | |
| **Post conditions:** | | List all results that matched with the keyword user enters will be displayed bellow Search screen | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Enter search phrase to text box | | |
| 2 | User | Choose one of bellow search type (User, Account, Event, Thread or  Organization) | | |
| 3 | User | Click Search icon or press Enter | | |
| 4 | WS | List all results that matched with the keyword user entered in. | | |
| **Alternative Flows:** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | [B1](#_Business_Rules) | | |
| **Other Information:** | | N/A | | |

Table 2- 15: Searching Use case

UC002-View event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC002 | **Version** | 1.0 |
| **Use Case Name** | | View event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user  Guest | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to show all existing events and event detail stored on WS | | |
| **Preconditions:** | | 1. WS website is available 2. User browsed WS website | | |
| **Trigger:** | | User click “Event” | | |
| **Post conditions:** | | Event information will be displayed in the Event detail screen | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Event | | |
| 2 | WS | List all existing event | | |
| 3 | User | Click on one event | | |
| 4 | WS | Redirect to event detail page | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 3 in the main flows, if user click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 1.0 | WS | Redirects user to chosen hyperlink or button | | |
|  | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 14: View event Use case

UC003-View event’s comment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC003 | **Version** | 1.0 |
| **Use Case Name** | | View event’s comment | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user  Guest | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to show all comment stored in a event | | |
| **Preconditions:** | | 1. WS website is available 2. User browsed WS website | | |
| **Trigger:** | | User click “Event” | | |
| **Post conditions:** | | List all Event’s comment will be displayed in the Event detail screen | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Event | | |
| 2 | WS | List all existing event | | |
| 3 | User | Click on 1 event and scroll down | | |
| 4 | WS | Display all comment in event | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if user click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 1.0 | WS | Redirects user to chosen hyperlink or button | | |
|  | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 14: View comment in Event Use case

UC004-View thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC004 | **Version** | 1.0 |
| **Use Case Name** | | View thread | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user  Guest | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to show all existing event stored on WS | | |
| **Preconditions:** | | 1. WS website is available 2. User browser WS is available | | |
| **Trigger:** | | User click “Thread” | | |
| **Post conditions:** | | Thread information will be displayed in the Thread detail screen | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Thread | | |
| 2 | WS | List all existing Thread | | |
| 3 | User | Click on one thread in the list | | |
| 4 | WS | Redirect and Display information of thread in thread detail page | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if user click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 1.0 | WS | Redirects user to chosen hyperlink or button | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 14: View thread Use case

UC005-View thread’s comment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC005 | **Version** | 1.0 |
| **Use Case Name** | | View event’s comment | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user  Guest | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to show all comment stored in event | | |
| **Preconditions:** | | 1. WS website is available 2. User browsed WS website | | |
| **Trigger:** | | User click “Thread” | | |
| **Post conditions:** | | List all Thread’s comment will be displayed in the Thread detail screen | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Thread | | |
| 2 | WS | List all existing thread | | |
| 3 | User | Click on 1 thread and scroll down | | |
| 4 | WS | Display all comment in thread | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if user click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 1.0 | WS | Redirects user to chosen hyperlink or button | | |
|  | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 14: View comment in thread Use case

##### Guest Group Function

###### Common Module

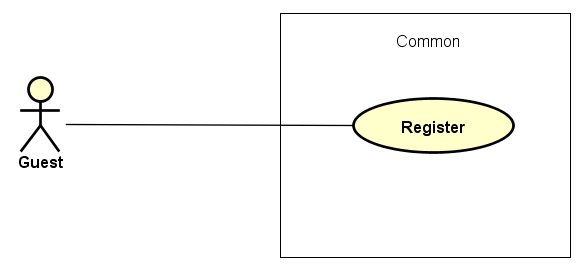


Figure 3-2: Guest common use case

UC006-Register

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use Case ID** | | | UC006 | **Version** | 1.0 |
| **Use Case Name** | | | Register | | |
| **Created by:** | | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | | Guest | **Secondary Actor** |  |
| **Priority** | | | High | | |
| **Frequency of Use:** | | | High | | |
| **Description:** | | | Visitor can create new account to use more function as a register user | | |
| **Preconditions:** | | | 1. WS must be connected to the Internet 2. User browsed WS website | | |
| **Trigger:** | | | Click on register link on the website | | |
| **Post conditions:** | | | 1. Add account information to database 2. Send confirm email 3. Save user information into system 4. Redirect user to confirm validate code page 5. Display login screen | | |
| **Normal Flow** | | | | | |
| **Step** | **Actor** | | **Action** | | |
|  | User | | Click Register in header | | |
|  | WS | | Display Register page includes:   * Header * User name text box * Password text box * Confirm password text box * Email text box * Register button * Cancel button | | |
|  | User | | Enters data in fields | | |
|  | User | | Click Register | | |
|  | WS | | Send confirm validate code email | | |
|  | WS | | Logs user into system | | |
|  | WS | | Display validate code page | | |
|  | WS | | Display login page | | |
| **Alternative Flows** | | | | | |
| **AT1** | At step 4, user click Cancel | | | | |
| **Step** | **Actor** | | **Action** | | |
| 4.1 | WS | | Redirects user to previous page | | |
|  | | | | | |
| **AT2** | At step 4 in the main flows, if user click other hyperlinks or buttons | | | | |
| **Step** | **Actor** | **Action** | | | |
| 4.1 | WS | Redirects user to chosen hyperlink or button | | | |
|  | | | | | |
| **Exceptions:** | | | | | |
| **EC1** | At step 4, required fields are not entered | | | | |
| **Step** | **Actor** | | **Action** | | |
| 4.1 | WS | | Display Register page with message: with ID MSG5 | | |
| 4.2 | WS | | Mark error fields | | |
|  | | | | | |
| **EC2** | At step 4, fields are entered with wrong type of data | | | | |
| **Step** | **Actor** | | **Action** | | |
| 4.1 | WS | | Display Register page with message: with ID MSG6 | | |
| 4.2 | WS | | Mark error fields | | |
| **Business Rules:** | | |  | | |
| **Other Information:** | | | N/A | | |

Table 2- 30: Register Use case

##### Registered Group Function

###### Common module

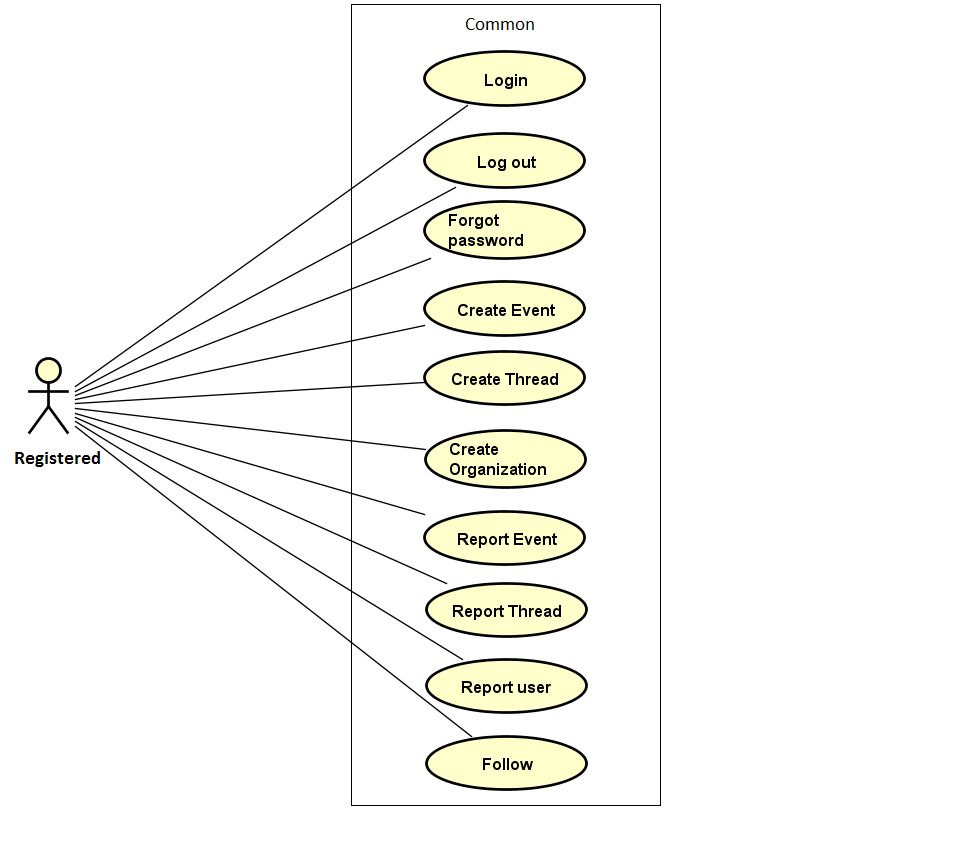


Figure 3-3: Registered common use case

UC007- Login

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC007 | **Version** | 1.0 |
| **Use Case Name** | | Login by social network’s account  Login by registered account | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered User | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | When User want to login by Facebook/Register account to use more functions in website | | |
| **Preconditions:** | | 1. WS must be connected to the Internet 2. User browsed WS website 3. Account Facebook/Register must be exist and correctly 4. User is viewing Login page | | |
| **Trigger:** | | User enter wings.com | | |
| **Post conditions:** | | 1. Log user into system 2. Redirect user to previous page and display as Member | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
|  | User | Click Login in Homepage | | |
|  | WS | Redirect to Login page includes:   * User name text box * Password text box * Login button * Cancel button * Facebook button * Forgot password hyperlink * Remember me check box | | |
|  | User | Enters User name and Password | | |
|  | User | Click Login | | |
|  | WS | Logs user into system | | |
|  | WS | Redirect to Homepage | | |
|  | WS | Display previous page as Member | | |
| **Alternative Flows** | | | | |
| **AT1** | At step 3 in the main flows, if user click Facebook button | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Redirects user to confirm site | | |
| 3.2 | WS | Logs user into system with Facebook account | | |
|  | | | | |
| **AT2** | At step 4, user choose Cancel: | | | |
| **Step** | **Actor** | **Action** | | |
| 4.1 | WS | Closes popup. | | |
|  | | | | |
| **AT3** | At step 8 in the main flows, if user click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 8.1 | WS | Redirects user to chosen hyperlink or button | | |
|  | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 4 in main flow, if user entered wrong User name or Password | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display Login popup message with ID MSG14 | | |
| 3.2 | WS | Mark error fields | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 31: Login Use case

UC008- Logout

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC008 | **Version** | 1.0 |
| **Use Case Name** | | Logout | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | User | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | When User want to logout their account | | |
| **Preconditions:** | | 1. WS must be connected to the Internet 2. User browsed WS website 3. User logged in to system | | |
| **Trigger:** | | User click Logout in user menu | | |
| **Post conditions:** | | 1. Log user out from system. 2. Redirect user to guest homepage view | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Logout | | |
| 2 | WS | Display page as Guest | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 32: Logout Use case

UC009- Forgot password

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC009 | **Version** | 1.0 |
| **Use Case Name** | | Forgot password | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | User | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | Normal | | |
| **Description:** | | When Member forgot their password to login the system, this function will receiving their password to email | | |
| **Preconditions:** | | 1. WS must be connected to the Internet 2. User browsed WS website 3. User logged in to system | | |
| **Trigger:** | | Click Forgot password in Login popup | | |
| **Post conditions:** | | 1. Send confirm email | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
|  | User | Click Forgot password in Login page | | |
|  | WS | Display Email text box | | |
|  | User | Enter registered email | | |
|  | User | Click Get password | | |
|  | WS | Sends confirm email | | |
|  | WS | Display message: | | |
| **Alternative Flows** | | | | |
| **AT2** | At step 3 in the main flow, user clicks on Cancel button | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Return to previous screen | | |
|  | | | | |
| **AT2** | At step 4 in the main flows, if user click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 4.1 | WS | Redirects user to chosen hyperlink or button | | |
|  | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 4, entered email is invalid | | | |
| **Step** | **Actor** | **Action** | | |
| 4.1 | WS | Displays Login popup with message: with ID MSG15 | | |
| 4.2 | WS | Marks email text box | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 33: Forgot password Use case

UC010-Create Event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC010 | **Version** | 1.0 |
| **Use Case Name** | | Create event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to create an event | | |
| **Preconditions:** | | 1. WS website is available  2. User browsed WS website  3. User logged in to system  4. User has their own organization  5. User is on homepage or event page | | |
| **Trigger:** | | User click on button “Create Even” | | |
| **Post conditions:** | | Create an event page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click create event | | |
| 2 | WS | Show create event page | | |
| 3 | User | Fill all information about event | | |
| 4 | WS | Create button enable | | |
| 5 | User | Click on Create Event | | |
| 6 | WS | Event created and waiting for administrator accept. | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 3 in the main flows, if user don’t fill all information of event | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Warning label display to remind user | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 34: Create event Use case

UC011- Create Thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC011 | **Version** | 1.0 |
| **Use Case Name** | | Share event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to create a thread to discussion | | |
| **Preconditions:** | | 1.WS website is available  2.User browsed WS website  3.User logged in to system  4.User is on homepage or thread page | | |
| **Trigger:** | | User click on create thread button | | |
| **Post conditions:** | | Create an thread page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click create thread | | |
| 2 | WS | Show create thread page | | |
| 3 | User | Fill all information about thread | | |
| 4 | WS | Create button enable | | |
| 5 | User | Click on Create Thread | | |
| 6 | WS | Thread created has been created | | |
| **Alternative Flows: N/A** | | | | |
| **EC1** | At step 3 in the main flows, if user don’t fill all information of thread | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Warning label display to remind user | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | |  | | |
| **Other Information:** | | N/A | | |

Table 2- 35: Create Thread Use case

UC012- Create Organization

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC012 | **Version** | 1.0 |
| **Use Case Name** | | Star event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to create an organization | | |
| **Preconditions:** | | 1.WS website is available  2.User browsed WS website  3.User logged in to system  4.User is on organization page | | |
| **Trigger:** | | User click on Create Organization button | | |
| **Post conditions:** | | Organization page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click create organization | | |
| 2 | WS | Show create thread page | | |
| 3 | User | Fill all information about thread | | |
| 4 | WS | Create button enable | | |
| 5 | User | Click on Create Thread | | |
| 6 | WS | Thread created has been created | | |
| **Alternative Flows: N/A** | | | | |
| **EC1** | At step 3 in the main flows, if user don’t fill all information of organization | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Warning label display to remind user | | |
|  | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 36: Create Organization Use case

UC013- Report event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC013 | **Version** | 1.0 |
| **Use Case Name** | | Report event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to report an event | | |
| **Preconditions:** | | 1.WS website is available  2.User browsed WS website  3.User logged in to system  4.User is on event detail page | | |
| **Trigger:** | | User click on button report | | |
| **Post conditions:** | | Event detail page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | WS | Display all information of event | | |
| 2 | User | Click Report Button | | |
| 3 | WS | Show confirm dialog with question ask about reason | | |
| 4 | User | Fill reason then click send report | | |
| 5 | WS | Send report | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | |  | | |
| **Other Information:** | | N/A | | |

Table 2- 37: Report Event Use case

UC014- Report thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC014 | **Version** | 1.0 |
| **Use Case Name** | | Report thread | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to report a thread | | |
| **Preconditions:** | | 1.WS website is available  2.User browsed WS website  3.User logged in to system  4.User is on thread detail page | | |
| **Trigger:** | | User click on button report | | |
| **Post conditions:** | | Thread detail page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | WS | Display all information of event | | |
| 2 | User | Click Report | | |
| 3 | WS | Show confirm dialog with question ask about reason | | |
| 4 | User | Fill reason then confirm | | |
| 5 | WS | Send report | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 37: Report thread Use case

UC015- Report user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC015 | **Version** | 1.0 |
| **Use Case Name** | | Report user | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to report an user | | |
| **Preconditions:** | | 1.WS website is available  2.User browsed WS website  3.User logged in to system  4.User is on User profife page | | |
| **Trigger:** | | User click on button report | | |
| **Post conditions:** | | User profile page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | WS | Display all information of user | | |
| 2 | User | Click Report | | |
| 3 | WS | Show confirm dialog with question ask about reason | | |
| 4 | User | Fill reason then confirm | | |
| 5 | WS | Send report | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | |  | | |
| **Other Information:** | | N/A | | |

Table 2- 37: Report user Use case

###### Manage Personal Information page

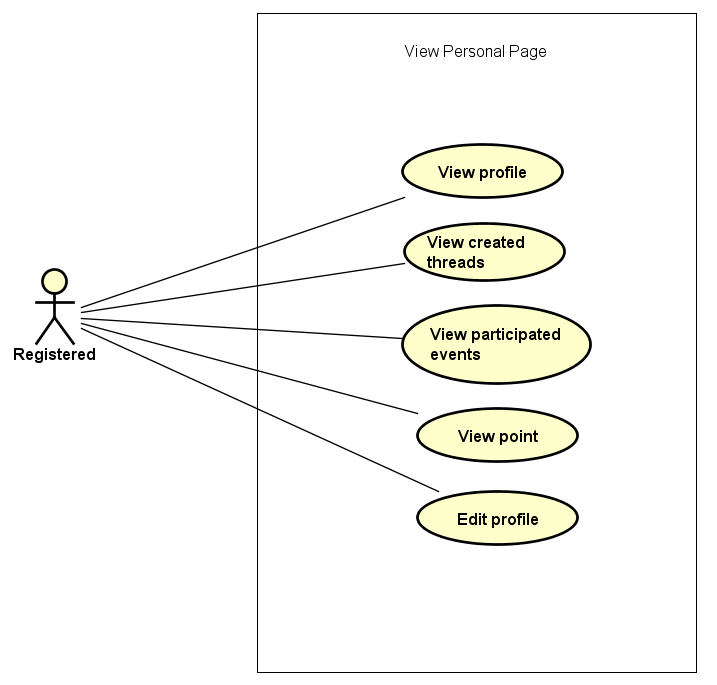


Figure 3-4: Manage Personal Information use case

UC016- View profile

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC016 | **Version** | 1.0 |
| **Use Case Name** | | Display user profile | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | WS display the information of user on user profile page | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User move on Avatar image icon | | |
| **Post conditions:** | | The user profile is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click View Profile | | |
| 4 | WS | Redirect user to Profile page includes:   * Header * User information tab * Donation tab * Thread tab * Change password tab | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 24: View profile Use case

UC017- View created thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC017 | **Version** | 1.0 |
| **Use Case Name** | | Display created event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | WS records all thread which user created | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User move on Avatar image icon | | |
| **Post conditions:** | | All created thread is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click Profile | | |
| 4 | WS | Redirect user to Profile page includes:   * Header * User information tab * Donation tab * Thread tab * Change password tab | | |
| 5 | User | Click Created Thread tab | | |
| 6 | WS | Display all thread created | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 25: View Created thread Use case

UC018- View donated event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC018 | **Version** | 1.0 |
| **Use Case Name** | | Display donated event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | WS records all event which user donated | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User move on Avatar image icon | | |
| **Post conditions:** | | All donated event are displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click Profile | | |
| 4 | WS | Redirect user to Profile page includes:   * Header * User information tab * Donation tab * Thread tab * Change password tab | | |
| 5 | User | Click on donated event tab | | |
| 6 | WS | Display all donated event | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 26: View craeted event Use case

UC019- View point

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC019 | **Version** | 1.0 |
| **Use Case Name** | | Display user activity point | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | WS record every user activity and base on it to point or rank each user | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User move on Avatar image icon | | |
| **Post conditions:** | | Display activity point of user | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click Profile | | |
| 4 | WS | Redirect user to Profile page includes:   * Header * User information | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 27: View activity point Use case

UC020- Edit profile

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC020 | **Version** | 1.0 |
| **Use Case Name** | | Edit Profile | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to edit profile | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click Edit button | | |
| **Post conditions:** | | Edit profile screen is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click Profile | | |
| 4 | WS | Redirect user to Profile page includes:   * Header * User information | | |
| 5 | User | Click Edit profile button | | |
| 6 | WS | Some text box are enable to edit | | |
| 7 | User | Edit information and click Save button | | |
| 8 | WS | Display message successful  All information has been saved and display. All text box are disable to edit. | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 7 in main flow, if user do not click Save and go to other page | | | |
| **Step** | **Actor** | **Action** | | |
| 8.1 | WS | All the information which has been changed do not save | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 28: Edit Profile Use case

UC021- Change password

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC021 | **Version** | 1.1 |
| **Use Case Name** | | Change password | | |
| **Created by:** | | AnhDD | **Date Created:** | 24/09/2015 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | Normal | | |
| **Description:** | | This function allows user to change password | | |
| **Preconditions:** | | 1. DDL website is available  2.User browsed DDL website  3. User logged in to system | | |
| **Trigger:** | | User click Change password tab | | |
| **Post conditions:** | | Password has been changed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click Profile | | |
| 4 | WS | Redirect user to Profile page includes:   * Header * User information tab * Donation tab * Thread tab * Change password tab | | |
| 5 | User | Click change password tab | | |
| 6 | WS | Display change password tab | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 29: Change password

###### Message

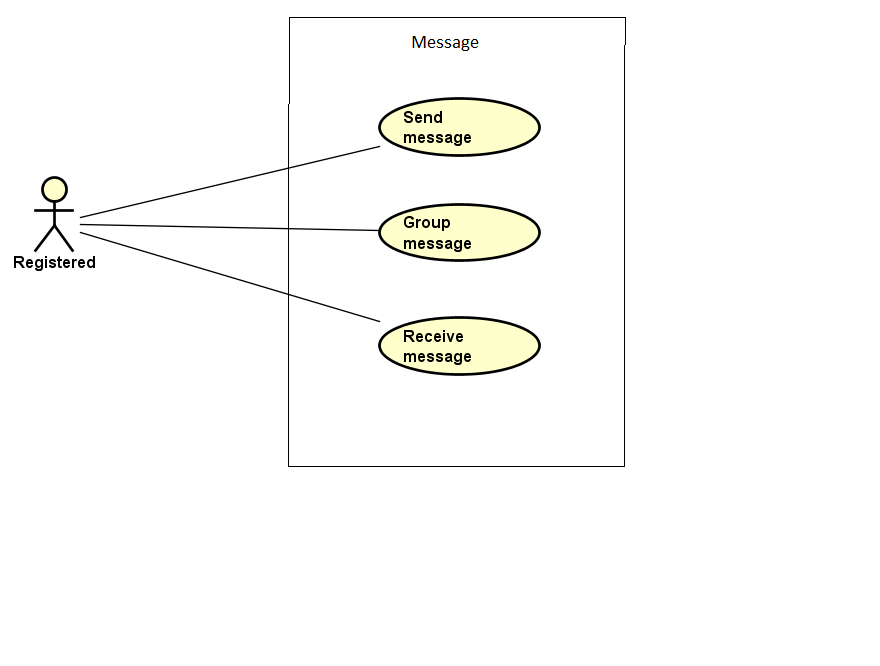


Figure 3-5: Message Modal use case

UC022- Group message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC022 | **Version** | 1.0 |
| **Use Case Name** | | Group message | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | WS records all message of user | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click Message then choose message type in drop box | | |
| **Post conditions:** | | List all results that matched with the message type and on screen | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click Message | | |
| 4 | WS | Display message page | | |
| 5 | User | Choose message type in drop box | | |
| 6 | WS | List all results that matched with the message type and on screen | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 5 in the main flows, if there is no result matched the message type | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display Nothing | | |
|  | | | | |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | Normal | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 22: Group message Use case

UC023-Send Message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC023 | **Version** | 1.0 |
| **Use Case Name** | | Send message | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to send a message | | |
| **Preconditions:** | | 1. WS website is available 2. User browsed WS website 3. User logged in to system 4. User are on message page | | |
| **Trigger:** | | User enters message and click Send | | |
| **Post conditions:** | | Message sent | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click Message | | |
| 4 | WS | Display message page | | |
| 5 | User | Choose 1 message of other user | | |
| 6 | WS | Display conversation of 2 user | | |
| 7 | User | Enter message and click Send or Press Enter | | |
| 8 | WS | Message send to user | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 5 in the main flows, if user enter user URL linl | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | WS | Redirects user to other user profile | | |
| 2 | User | Click on Send Message button | | |
| 3 | WS | Display message pop-up | | |
| 4 | User | Write message and click Send | | |
| 5 | WS | Send message and display notification | | |
|  | | | | |
| **AT2** | At step 2 in the main flows, if user click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Redirects user to chosen hyperlink or button | | |
|  | | | | |
| **Exceptions:** | | | | |
| **EC1** | Message can not sent | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | WS | Show a dialog error | | |
|  | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | |  | | |
| **Other Information:** | | N/A | | |

UC024-Receive Message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC024 | **Version** | 1.0 |
| **Use Case Name** | | Receive message | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to receive message from other user | | |
| **Preconditions:** | | 1.WS website is available  2.User browsed WS website  3.User logged in to system | | |
| **Trigger:** | | User choose a message button | | |
| **Post conditions:** | | All message is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click Avatar image button in header | | |
| 2 | WS | Display a menu includes:   * Message * Notification * Profile * Organization * Logout | | |
| 3 | User | Click Message | | |
| 4 | WS | Display message page | | |
| 5 | User | Choose 1 of message type | | |
| 6 | WS | Display all message which has been sent from other | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 2 in the main flows, if user click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Redirects user to chosen hyperlink or button | | |
|  | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | |  | | |
| **Other Information:** | | N/A | | |

Table 2- 23: Receive message Use case

###### Chat room

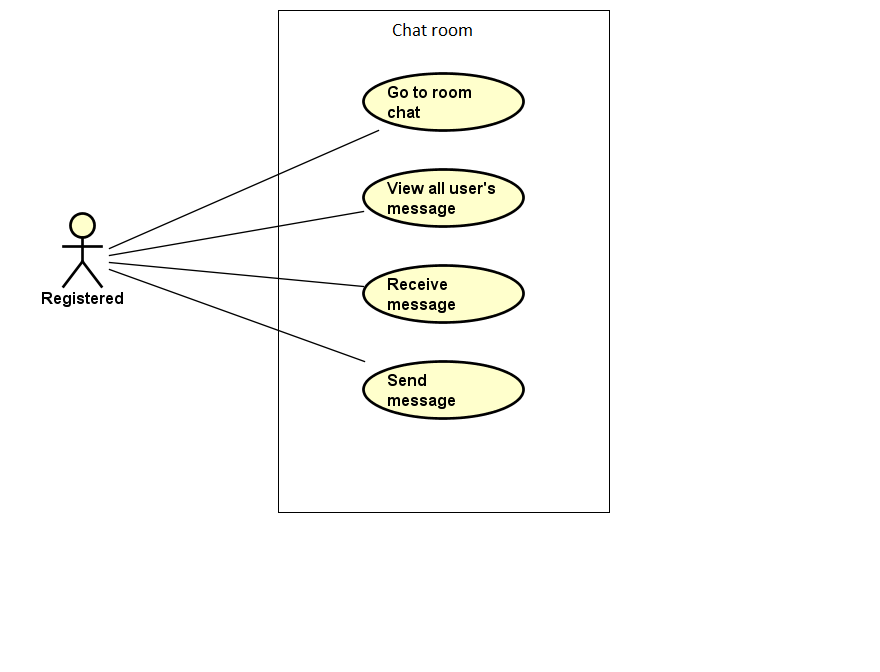


Figure 3-6: Message Modal use case

UC025- Go to Room chat

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC025 | **Version** | 1.0 |
| **Use Case Name** | | Enter event chat room | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | User enter to event chat room | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click Enter chat room | | |
| **Post conditions:** | | User enter chat room | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event on homepage | | |
| 2 | WS | Redirect to event detail | | |
| 3 | User | Click on pop-up Chat and click Enter Room chat | | |
| 4 | WS | Move to event room chat and display all chat message of all user | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 3 in the main flows, if user cannot enter room chat | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display Error | | |
|  | | | | |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 22: Enter room chat Use case

UC026- View all user message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC026 | **Version** | 1.0 |
| **Use Case Name** | | View all user message before | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | User enter to event chat room | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click Enter chat room | | |
| **Post conditions:** | | Display all message of users before | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event on homepage | | |
| 2 | WS | Redirect to event detail | | |
| 3 | User | Click on pop-up Chat and click Enter Room chat | | |
| 4 | WS | Move to event room chat and display all chat message of all user | | |
| **Alternative Flows** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 3 in the main flows, if user cannot enter room chat | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display Error | | |
|  | | | | |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 22: View all user message Use case

UC027- Receive Message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC027 | **Version** | 1.0 |
| **Use Case Name** | | Enter event chat room | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | User enter to event chat room | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click Enter chat room | | |
| **Post conditions:** | | User receive message when other send | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event on homepage | | |
| 2 | WS | Redirect to event detail | | |
| 3 | User | Click on pop-up Chat and click Enter Room chat | | |
| 4 | WS | Display message when other send | | |
| **Alternative Flows** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 3 in the main flows, if user cannot enter room chat | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display Error | | |
|  | | | | |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 22: Receive message Use case

UC028- Send Message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC028 | **Version** | 1.0 |
| **Use Case Name** | | Enter event chat room | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | User enter to event chat room | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click Enter chat room | | |
| **Post conditions:** | | Message has been sent and display on room chat | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event on homepage | | |
| 2 | WS | Redirect to event detail | | |
| 3 | User | Click on pop-up Chat and click Enter Room chat | | |
| 4 | WS | Move to event room chat and display all chat message of all user | | |
| 5 | User | Write message and click Send (Or press Enter) | | |
| 6 | WS | Send message and display on room chat pop-up | | |
| **Alternative Flows** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 3 in the main flows, if user cannot enter room chat | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display Error | | |
|  | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 22: Send message Use case

###### Like

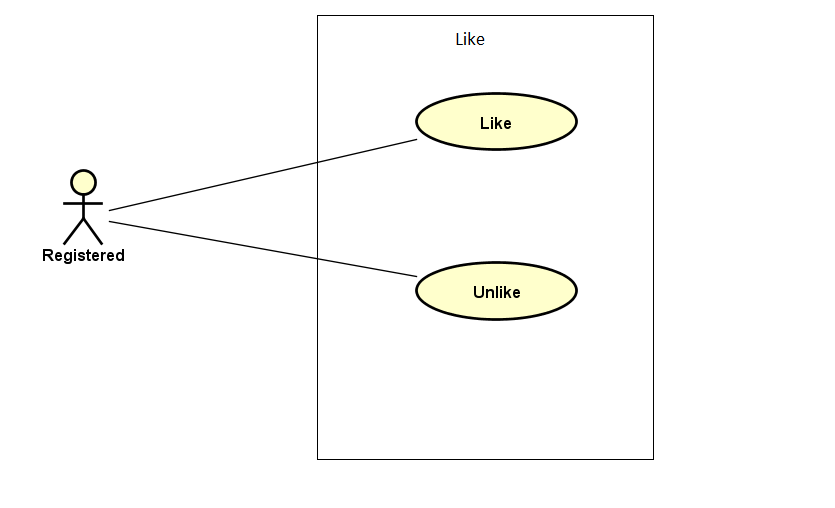


Figure 3-7: Like use case

UC029- Like

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC029 | **Version** | 1.0 |
| **Use Case Name** | | Like comment or thread | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | User like an comment or thread which they interested | | |
| **Preconditions:** | | 1. WS website is available  2. User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click like | | |
| **Post conditions:** | | Event or thread has been liked | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event or thread | | |
| 2 | WS | Go to event or thread information detail page | | |
| 3 | User | Like | | |
| 4 | WS | Show like number | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 38: Like Use case

UC030- Unlike

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC030 | **Version** | 1.0 |
| **Use Case Name** | | Unlike comment or thread | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | User like an comment or thread which they interested | | |
| **Preconditions:** | | 1. WS website is available  2. User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click like (already liked this before) | | |
| **Post conditions:** | | Event or thread has been unliked | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event or thread | | |
| 2 | WS | Go to event or thread information detail page | | |
| 3 | User | Click on like button (already like it before) | | |
| 4 | WS | Unlike this event or thread | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 38: Like Use case

###### Comment

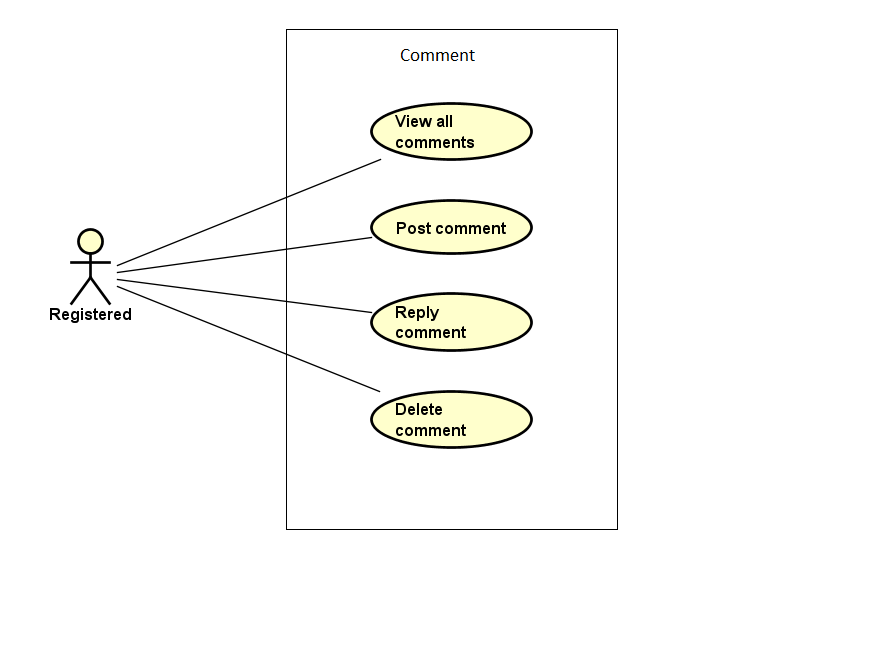


Figure 3-8: Comment use case

UC031- View all comment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC031 | **Version** | 1.0 |
| **Use Case Name** | | View list comment | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | WS show all comment of an event or thread | | |
| **Preconditions:** | | 1. WS website is available  2. User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click on event or thread detail | | |
| **Post conditions:** | | Comment is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event or thread | | |
| 2 | WS | Go to event or thread detail page | | |
| 3 | User | Scroll down to comment area | | |
| 4 | WS | Display list comment | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 38: View all comment Use case

UC032- Post comment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC032 | **Version** | 1.0 |
| **Use Case Name** | | Post comment | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | User post a comment | | |
| **Preconditions:** | | 1. WS website is available  2. User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click on event or thread | | |
| **Post conditions:** | | Comment area is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event or thread | | |
| 2 | WS | Go to event or thread detail page | | |
| 3 | User | Scroll down to comment area | | |
| 4 | WS | Display list comment | | |
| 5 | User | Fill the comment text box then click post | | |
| 6 | WS | Post the user comment | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 39: Post comment Use case

UC033- Reply comment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC033 | **Version** | 1.0 |
| **Use Case Name** | | Edit comment | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | User edit comment | | |
| **Preconditions:** | | 1. WS website is available  2. User browsed WS website  3. User logged in to system  4. User clicked on a event | | |
| **Trigger:** | | User click on comment tab in event information page | | |
| **Post conditions:** | | Comment tab is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event or thread | | |
| 2 | WS | Go to event or thread detail page | | |
| 3 | User | Scroll down to comment area | | |
| 4 | WS | Display list comment | | |
| 5 | User | Choose a comment which want to reply and click Reply | | |
| 6 | WS | Show comment text box | | |
| 7 | User | Write the reply in text box and press Enter | | |
| 8 | WS | Display the comment as reply comment | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 40: Registered user Use case

UC034- Delete comment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC034 | **Version** | 1.0 |
| **Use Case Name** | | Delete comment | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | Normal | | |
| **Description:** | | User edit comment | | |
| **Preconditions:** | | 1. WS website is available  2. User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click on thread or event detail page | | |
| **Post conditions:** | | Comment area is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Click on an event or thread | | |
| 2 | WS | Go to event or thread detail page | | |
| 3 | User | Scroll down to comment area | | |
| 4 | WS | Display list comment | | |
| 5 | User | Click on own comment and click delete | | |
| 6 | WS | Delete the comment | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 41: Delete comment Use case

###### Manage Event

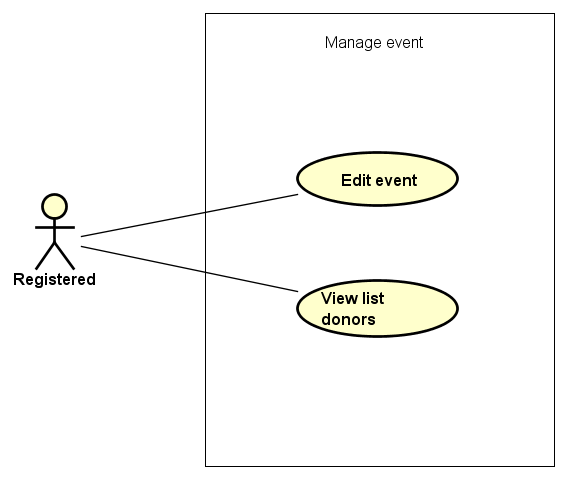


Figure 3-9: Manage event use case

UC035- Edit Event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC035 | **Version** | 1.0 |
| **Use Case Name** | | Edit Event | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to edit Event | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click Edit | | |
| **Post conditions:** | | Edit event Page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Choose a event what user’s own | | |
| 2 | User | Click edit | | |
| 3 | User | Edit information | | |
| 4 | User | Click save | | |
| 5 | WS | Save change | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 3 in the main flows, if user miss fill 1 or more information | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display Dialog to remind user | | |
|  | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 42: Edit Event Use case

UC036-View list donor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC036 | **Version** | 1.0 |
| **Use Case Name** | | View list donor | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to edit Event | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click list backer tab | | |
| **Post conditions:** | | Event information page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Choose a event what user’s own | | |
| 2 | User | Click on list donor tab | | |
| 3 | WS | Show list donor table | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 43: View list donors Use case

###### Manage Thread

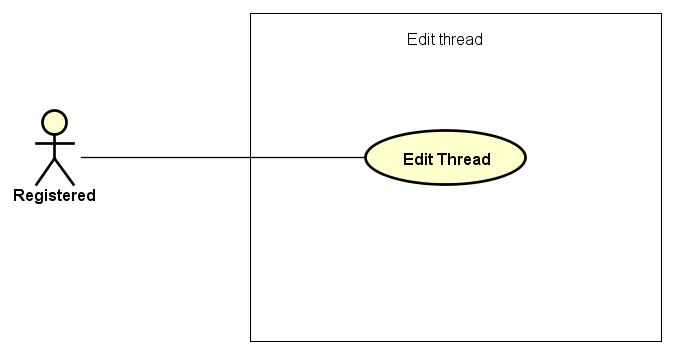


Figure 3-9: Manage thread use case

UC037- Edit Thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC037 | **Version** | 1.0 |
| **Use Case Name** | | Edit Thread | | |
| **Created by:** | | DuyTN | **Date Created:** | 05/11/2016 |
| **Primary Actor** | | Registered user | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows user to edit Thread | | |
| **Preconditions:** | | 1. WS website is available  2.User browsed WS website  3. User logged in to system | | |
| **Trigger:** | | User click Edit | | |
| **Post conditions:** | | Edit thread Page is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | User | Choose a thread what user’s own | | |
| 2 | User | Click edit | | |
| 3 | User | Edit information | | |
| 4 | User | Click save | | |
| 5 | WS | Save change | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 3 in the main flows, if user miss fill 1 or more information | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display Dialog to remind user | | |
|  | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 42: Edit Thread Use case

##### Administrator Group Function

###### Common module

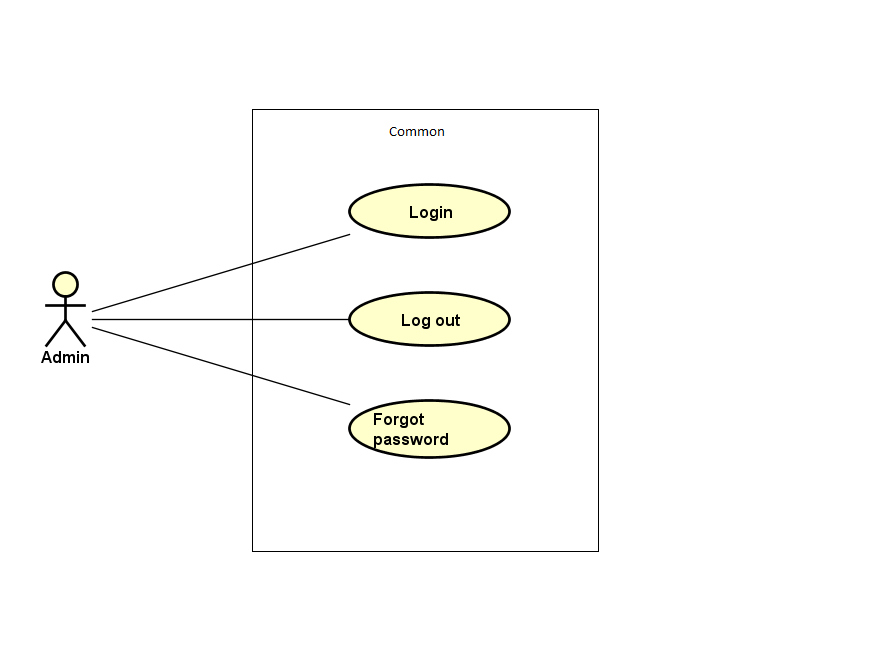


Figure 3-10: Admin common module use case

UC038- Admin login

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use Case ID** | | UC038 | **Version** | 1.0 | |
| **Use Case Name** | | Admin login | | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 | |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  | |
| **Priority** | | High | | | |
| **Frequency of Use:** | | High | | | |
| **Description:** | | This function allows Admin logins into website | | | |
| **Preconditions:** | | 1. WS website is available  2. Admin browsed WS website | | | |
| **Trigger:** | | Admin clicks on Login button | | | |
| **Post conditions:** | | 1. Log Admin into system 2. Redirect Admin to Admin home page | | | |
| **Normal Flow** | | | | | |
| **Step** | **Actor** | **Action** | | | |
| 1 | Admin | Enter URL link: wings.com/Admin | | | |
| 2 | WS | Display Admin Login page. | | | |
| 3 | Admin | Type username and password | | | |
| 4 | WS | Verify:   * 1. Email address   2. Password | | | |
| 5 | WS | * Log Admin into system * Redirect Admin to Admin home page | | | |
| **Alternative Flows: N/A** | | | | | |
| **Exceptions:** | | | | | |
| **EC1** | At step 3 in main flow, if Admin entered wrong UserName or Password | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display warning error access failed | | |
| **Business Rules:** | | N/A | | | |
| **Other Information:** | | N/A | | | |

Table 2- 45: Admin login Use case

UC039- Admin logout

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC039 | **Version** | 1.0 |
| **Use Case Name** | | Admin logout | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Admin logouts of WS website | | |
| **Preconditions:** | | Admin logged in WS website as Administrator | | |
| **Trigger:** | | Admin clicks on Logout | | |
| **Post conditions:** | | Admin is logged out from WS website | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on dropdown at right side of WS website * Choose Logout option | | |
| 2 | WS | * Logout Admin * Redirect to Admin Login page | | |
| **Alternative Flows:N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 46: Admin logout Use case

UC040- Admin forgot password

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use Case ID** | | UC040 | **Version** | 1.0 | |
| **Use Case Name** | | Admin forgot password | | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 | |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  | |
| **Priority** | | Normal | | | |
| **Frequency of Use:** | | Normal | | | |
| **Description:** | | This function allows Admin receiving her/ his password to email | | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website | | | |
| **Trigger:** | | Click Forgot password in Login page | | | |
| **Post conditions:** | | Admin get new password on her/ his email | | | |
| **Normal Flow** | | | | | |
| **Step** | **Actor** | **Action** | | | |
| 1 | Admin | Click Forgot password in Login page | | |
| 2 | WS | Display Email text box | | |
| 3 | Admin | Enter registered email | | |
| 4 | Admin | Click Get Password | | |
| 5 | WS | Sends confirm email | | |
| 6 | WS | Redirect to guest homepage | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 3 in the main flows, if Admin click other hyperlinks or buttons | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Redirects Admin to chosen hyperlink or button | | |
|  | | | | |
| **Exceptions:** | | | | |
| **EC1** | At step 3, mail input is invalid | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Displays warning message | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 47: Admin forgot password Use case

###### Manage user's account

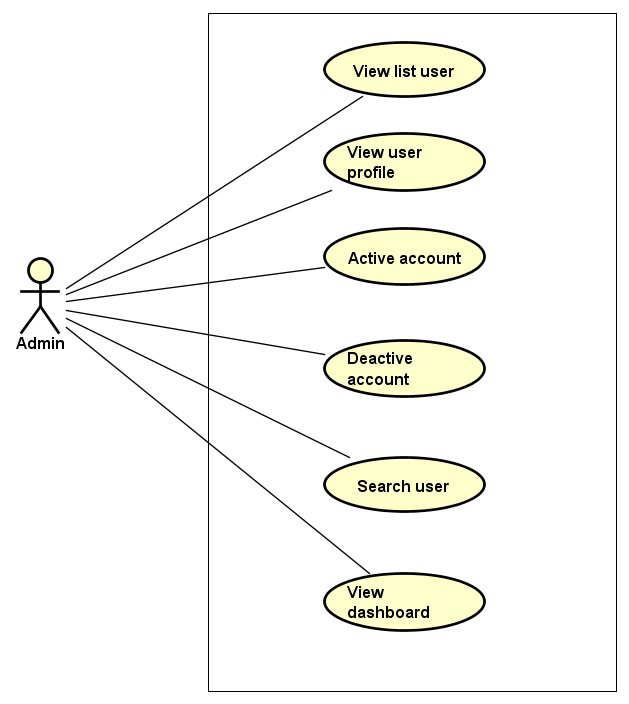


Figure 3-11: Manage user account use case

UC041- View list user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC041 | **Version** | 1.0 |
| **Use Case Name** | | View user list | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view User list of who has registered. | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on Users List | | |
| **Post conditions:** | | Users List is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Users at left side * Click on Users List | | |
| 2 | WS | Display list of users | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if do not have any user | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “Do not have any user” message | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 48: View user list Use case

UC042- View user profile

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC042 | **Version** | 1.0 |
| **Use Case Name** | | View user profile | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view profile of an user | | |
| **Preconditions:** | | Admin is viewing Users List | | |
| **Trigger:** | | Admin clicks on View button | | |
| **Post conditions:** | | Profile of this user is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on View detail button at Action column of the user whom want to see profile | | |
| 2 | WS | Display profile of this user | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 49: View user profile Use case

UC043- Activate user's account

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC043 | **Version** | 1.0 |
| **Use Case Name** | | Activate user's account | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to activate an user’s account | | |
| **Preconditions:** | | * Admin is viewing Users List * Status of this user is deactivate | | |
| **Trigger:** | | Admin clicks on Deactivate button on status column | | |
| **Post conditions:** | | User’s account is activated | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on Deactivate button at Status column of the user whom want to activate | | |
| 2 | WS | Display pop-up confirm change status | | |
| 3 | Admin | Confirm and click YES | | |
| 4 | WS | * Change displayed status from Deactivate to Activate * Change status of this user’s account to Activate on database * Display message success | | |
| 5 | Admin | Click OK | | |
| 6 | WS | Close pop-up | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 3 in the main flows, if Admin click NO | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display “Cancel Action” message and Status doesn’t chage | | |
| 3.2 | Admin | Click OK | | |
| 3.3 | WS | Close pop-up | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 50: Activate user's account Use case

UC044- Deactivate user's account

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC044 | **Version** | 1.0 |
| **Use Case Name** | | Deactivate user's account | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to deactivate an user’s account | | |
| **Preconditions:** | | * Admin is viewing Users List * Status of this user is activate | | |
| **Trigger:** | | Admin clicks on Active button on status column | | |
| **Post conditions:** | | User’s account is deactivated | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on Active button at Status column of the user whom want to activate | | |
| 2 | WS | Display pop-up confirm change status | | |
| 3 | Admin | Confirm and click YES | | |
| 4 | WS | * Change displayed status from Active to Deactive * Change status of this user’s account to Deactive on database * Display message success | | |
| 5 | Admin | * Click OK | | |
| 6 | WS | * Close pop-up | | |
| **Alternative Flows** | | | | |
| **AT1** | At step 3 in the main flows, if Admin click NO | | | |
| **Step** | **Actor** | **Action** | | |
| 3.1 | WS | Display “Cancel Action” message and Status doesn’t chage | | |
| 3.2 | Admin | Click OK | | |
| 3.3 | WS | Close pop-up | | |
|  | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 51: Deactivate user's account Use case

UC045- Search user's account

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC045 | **Version** | 1.0 |
| **Use Case Name** | | Search user's account | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to search an user’s account | | |
| **Preconditions:** | | Admin is viewing Users List | | |
| **Trigger:** | | Admin types name of user at Search textbox | | |
| **Post conditions:** | | Display list of searched user | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Type keyword of user whom want to search | | |
| 2 | WS | * Display list of users who have the name is typed | | |
| **Alternative Flows:** | | | | |
| **F** | At step 1 in the main flows, if the typed name is not exist | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “No matching records found” message | | |
|  | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 52: Search user's account Use case

UC046- View user’s dashboard

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC046 | **Version** | 1.0 |
| **Use Case Name** | | View user’s dashboard | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view dashboard about user of the event | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on Dashboard | | |
| **Post conditions:** | | Dashboard about user is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Users at left side * Click on Dashboard | | |
| 2 | WS | Display User Dashboard page | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 54: View user’s dashboard Use case

###### Manage event

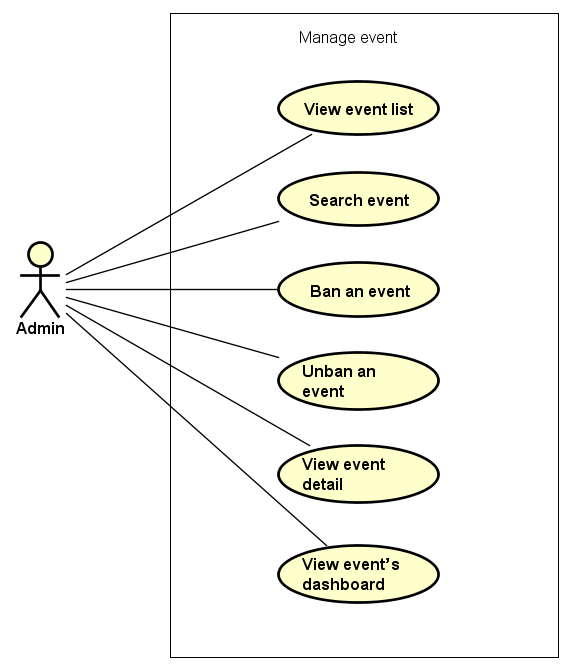


Figure 3-12: Manage event use case

UC047- View event list

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC047 | **Version** | 1.0 |
| **Use Case Name** | | View event list | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view event list | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on Events List | | |
| **Post conditions:** | | Event List is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Event at left side * Click on Events List | | |
| 2 | WS | Display list of events | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if do not have any event | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “Do not have any event” message | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 55: View event list Use case

UC048- Search event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC048 | **Version** | 1.0 |
| **Use Case Name** | | Search event | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to search an event | | |
| **Preconditions:** | | Admin is viewing Event List | | |
| **Trigger:** | | Admin types name of event on Search textbox | | |
| **Post conditions:** | | Display list of searched event | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Type name of event which want to search | | |
| 2 | WS | * Display list of event which have the name is typed | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if the typed name is not exist | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “No matching records found” message | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | [B1](#_Business_Rules) | | |
| **Other Information:** | | N/A | | |

Table 2- 56: Search event Use case

UC049- Ban Event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC049 | **Version** | 1.0 |
| **Use Case Name** | | Ban event | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to ban an event | | |
| **Preconditions:** | | Admin is viewing Event List and chosen event is not be banned | | |
| **Trigger:** | | Admin clicks on button Ban | | |
| **Post conditions:** | | Event has been banned | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on button ban at Action column | | |
| 2 | WS | Ban the chosen event | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 57: Ban event Use case

UC050- Unban Event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC050 | **Version** | 1.0 |
| **Use Case Name** | | Unban event | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to unban an event | | |
| **Preconditions:** | | Admin is viewing Event List and chosen event which is banned | | |
| **Trigger:** | | Admin clicks on button unban | | |
| **Post conditions:** | | Event has been unbanned | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on button ban at Action column | | |
| 2 | WS | Ban the chosen event | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 58: Unban event Use case

UC051- View event detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC051 | **Version** | 1.0 |
| **Use Case Name** | | View event detail | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator view information of event | | |
| **Preconditions:** | | Admin login | | |
| **Trigger:** | | Admin clicks on a event on thread list | | |
| **Post conditions:** | | Display event detail page | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on an event on event list | | |
| 2 | WS | Display Event detail page | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | |  | | |
| **Other Information:** | | N/A | | |

Table 2- 59: View event Use case

UC052- View event dashboard

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC052 | **Version** | 1.0 |
| **Use Case Name** | | View event dashboard | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view dashboard of event | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on Dashboard | | |
| **Post conditions:** | | Dashboard about event is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Event at left side * Click on Dashboard | | |
| 2 | WS | Display Event Dashboard page | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 61: View event’s dashboard Use case

###### Manage thread

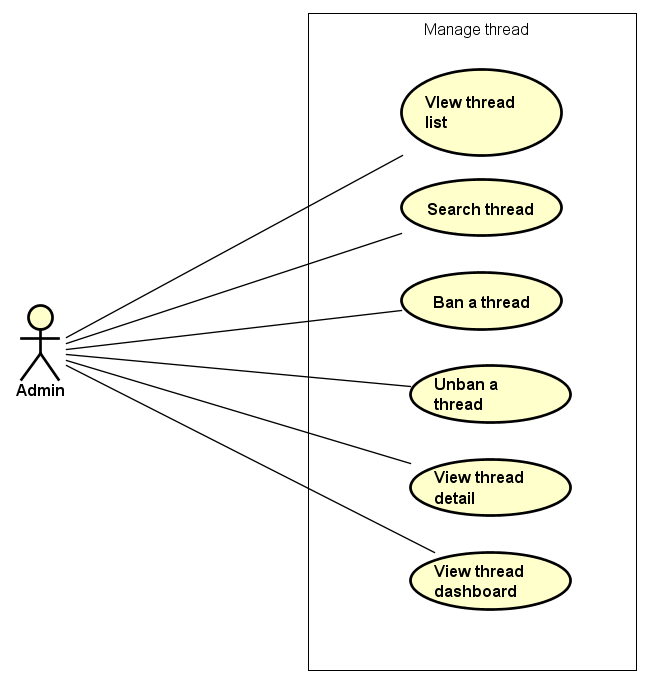


Figure 3-12: Manage thread use case

UC053- View thread list

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC053 | **Version** | 1.0 |
| **Use Case Name** | | View thread list | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view thread list | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on Thread List | | |
| **Post conditions:** | | Thread List is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Thread at left side * Click on Thread List | | |
| 2 | WS | Display list of thread | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if do not have any thread | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “Do not have any thread” message | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 55: View thread list Use case

UC054- Search thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC054 | **Version** | 1.0 |
| **Use Case Name** | | Search Thread | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to search a Thread | | |
| **Preconditions:** | | Admin is viewing Thread List | | |
| **Trigger:** | | Admin types name of thread on Search textbox | | |
| **Post conditions:** | | Display list of searched thread | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Type name of thread which want to search | | |
| 2 | WS | * Display list of threads which have the name is typed | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if the typed name is not exist | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “No matching records found” message | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | [B1](#_Business_Rules) | | |
| **Other Information:** | | N/A | | |

Table 2- 56: Search thread Use case

UC055- Ban Thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC055 | **Version** | 1.0 |
| **Use Case Name** | | Ban thread | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to ban a thread | | |
| **Preconditions:** | | Admin is viewing Thread List and chosen thread is not be banned | | |
| **Trigger:** | | Admin clicks on button Ban | | |
| **Post conditions:** | | Thread has been banned | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on button ban at Action column | | |
| 2 | WS | Ban the chosen thread | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 57: Ban thread Use case

UC056- Unban Thread

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC056 | **Version** | 1.0 |
| **Use Case Name** | | Unban thread | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to unban a thread | | |
| **Preconditions:** | | Admin is viewing Thread List and chosen thread which is banned | | |
| **Trigger:** | | Admin clicks on button unban | | |
| **Post conditions:** | | Thread has been unbanned | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on button ban at Action column | | |
| 2 | WS | Ban the chosen thread | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 58: Unban thread Use case

UC057- View thread detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC057 | **Version** | 1.0 |
| **Use Case Name** | | View event detail | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator view information of thread | | |
| **Preconditions:** | | * Admin login | | |
| **Trigger:** | | Admin clicks on a thread on thread list | | |
| **Post conditions:** | | Display thread detail page | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on a thread on thread list | | |
| 2 | WS | Display thread detail page | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 59: View thread detail Use case

UC058- View thread dashboard

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC058 | **Version** | 1.0 |
| **Use Case Name** | | View thread dashboard | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view dashboard of thread | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on Dashboard | | |
| **Post conditions:** | | Dashboard about thread is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Thread at left side * Click on Dashboard | | |
| 2 | WS | Display Thread Dashboard page | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 61: View thread’s dashboard Use case

###### Manage organization

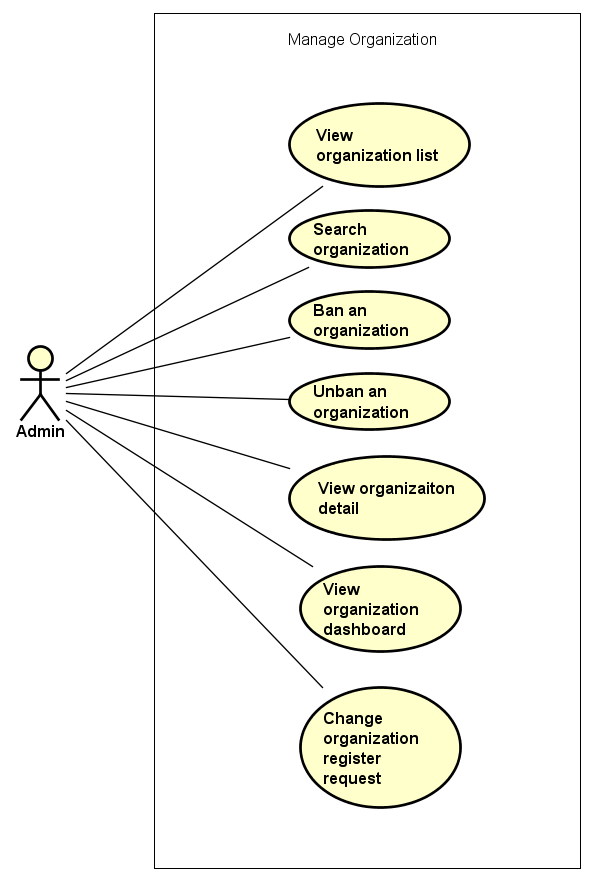


Figure 3-12: Manage organization use case

UC059- View organization list

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC059 | **Version** | 1.0 |
| **Use Case Name** | | View organization list | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view organization list | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on Organization List | | |
| **Post conditions:** | | Organization List is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Organization at left side * Click on Organization List | | |
| 2 | WS | Display list of Organization | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if do not have any thread | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “Do not have any thread” message | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 55: View organization list Use case

UC060- Search organization

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC060 | **Version** | 1.0 |
| **Use Case Name** | | Search Organization | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to search an Organization | | |
| **Preconditions:** | | Admin is viewing Organization List | | |
| **Trigger:** | | Admin types name of thread on Search textbox | | |
| **Post conditions:** | | Display list of searched thread | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Type name of organization which want to search | | |
| 2 | WS | * Display list of organizations which have the name is typed | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if the typed name is not exist | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “No matching records found” message | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 56: Search organization Use case

UC061- Ban Organization

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC061 | **Version** | 1.0 |
| **Use Case Name** | | Ban organization | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to ban an Organization | | |
| **Preconditions:** | | Admin is viewing Organization List and chosen an organization which is not be banned | | |
| **Trigger:** | | Admin clicks on button Ban | | |
| **Post conditions:** | | Organization has been banned | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on button ban at Action column | | |
| 2 | WS | Ban the chosen thread | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 57: Ban organization Use case

UC062- Unban Organization

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC062 | **Version** | 1.0 |
| **Use Case Name** | | Unban Organization | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to unban an Organization | | |
| **Preconditions:** | | Admin is viewing Organization List and chosen organization which is banned | | |
| **Trigger:** | | Admin clicks on button unban | | |
| **Post conditions:** | | Organization has been unbanned | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on button ban at Action column | | |
| 2 | WS | Ban the chosen organization | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 58: Unban organization Use case

UC063- View organization detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC063 | **Version** | 1.0 |
| **Use Case Name** | | View organization detail | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator view information of Organization | | |
| **Preconditions:** | | * Admin login | | |
| **Trigger:** | | Admin clicks on an Organization on thread list | | |
| **Post conditions:** | | Display organization detail page | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on an organization on Organization list | | |
| 2 | WS | Display organization detail page | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 59: View organization detail Use case

UC064- View organization dashboard

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC064 | **Version** | 1.0 |
| **Use Case Name** | | View Organization dashboard | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view dashboard of Organization | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on Dashboard | | |
| **Post conditions:** | | Dashboard about Organization is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Organization at left side * Click on Dashboard | | |
| 2 | WS | Display Organization Dashboard page | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 61: View Organization’s dashboard Use case

UC065- Manage organization register request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC065 | **Version** | 1.0 |
| **Use Case Name** | | Manage organization register request | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to change the request register an organization | | |
| **Preconditions:** | | Admin is viewing Organization Register request List | | |
| **Trigger:** | | Admin clicks accept or not | | |
| **Post conditions:** | | Organization request status is changed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on Organization Register request List | | |
| 2 | WS | Dislay request list | | |
| 3 | Admin | Admin choose a request and change stauts | | |
| 4 | WS | Request status is changed and mail to creator | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 57: Ban organization Use case

###### Manage donate

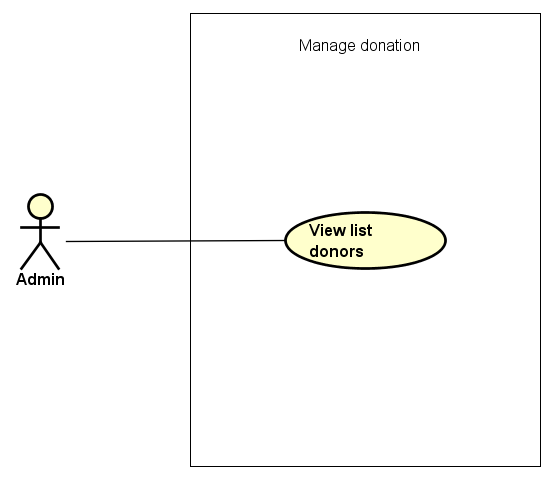


Figure 3-14: Manage donation use case

UC066- View donation list

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use Case ID** | | UC066 | **Version** | 1.0 | |
| **Use Case Name** | | View donation list | | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 | |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  | |
| **Priority** | | High | | | |
| **Frequency of Use:** | | High | | | |
| **Description:** | | This function allows Administrator to view Donation list of the event | | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | | |
| **Trigger:** | | Admin clicks on Donate | | | |
| **Post conditions:** | | Donation List is displayed | | | |
| **Normal Flow** | | | | | |
| **Step** | **Actor** | **Action** | | | |
| 1 | Admin | Click on Donate at left side | | | |
| 2 | WS | Display list of Donate | | | |
| **Alternative Flows:** | | | | | |
| **AT1** | At step 1 in the main flows, if do not have any donation | | | | |
| **Step** | **Actor** | **Action** | | | |
| 2.1 | WS | Display “Do not have any donate” message | | | |
|  | | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 62: View donation list Use case

###### Manage event categories

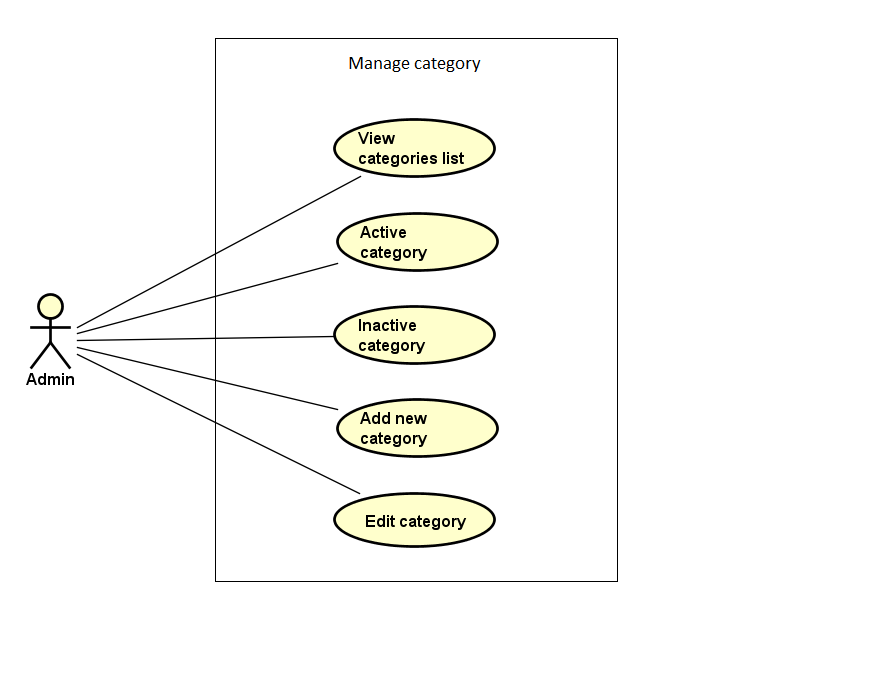


Figure 3-15: Manage event categories use case

UC067- View category list

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use Case ID** | | UC067 | **Version** | 1.0 | |
| **Use Case Name** | | View category list | | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 | |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  | |
| **Priority** | | Normal | | | |
| **Frequency of Use:** | | High | | | |
| **Description:** | | This function allows Administrator to view Category list of the event | | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | | |
| **Trigger:** | | Admin clicks on Category | | | |
| **Post conditions:** | | Category List is displayed | | | |
| **Normal Flow** | | | | | |
| **Step** | **Actor** | **Action** | | | |
| 1 | Admin | Click on Category at left side | | | |
| 2 | WS | Display list of categories | | | |
| **Alternative Flows:** | | | | | |
| **AT1** | At step 1 in the main flows, if do not have any category | | | | |
| **Step** | **Actor** | **Action** | | | |
| 2.1 | WS | Display “Do not have any category” message | | | |
|  | | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | [B1](#_Business_Rules) | | |
| **Other Information:** | | N/A | | |

Table 2- 67: View category list Use case

UC068- Search category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC068 | **Version** | 1.0 |
| **Use Case Name** | | Search category | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | Normal | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to search a category | | |
| **Preconditions:** | | Admin is viewing Category List | | |
| **Trigger:** | | Admin types name of category at Search textbox | | |
| **Post conditions:** | | Display list of searched category | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Type name of category which want to search | | |
| 2 | WS | Display list of categories which have the name is typed | | |
| **Alternative Flows:** | | | | |
| **AT1** | At step 1 in the main flows, if the typed name is not exist | | | |
| **Step** | **Actor** | **Action** | | |
| 2.1 | WS | Display “No matching records found” message | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 68: Search category Use case

UC069- Activate category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC069 | **Version** | 1.0 |
| **Use Case Name** | | Activate category | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to activate a category | | |
| **Preconditions:** | | * Admin is viewing Category List * Status of this category is deactivate | | |
| **Trigger:** | | Admin clicks on Deactivate button | | |
| **Post conditions:** | | Category is activated | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on Deactivate button at Status column of the category which want to activate | | |
| 2 | WS | * Change displayed status from Deactivate to Activate * Change status of this category to Activate on database | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 69: Activate category Use case

UC070- Deactivate category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC070 | **Version** | 1.0 |
| **Use Case Name** | | Deactivate category | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to deactivate a category | | |
| **Preconditions:** | | * Admin is viewing Category List * Status of this category is activate | | |
| **Trigger:** | | Admin clicks on Activate button | | |
| **Post conditions:** | | Category is deactivated | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on Activate button at Status column of the category which want to deactivate | | |
| 2 | WS | * Change displayed status from Activate to Deactivate * Change status of this category to Deactivate on database | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 70: Deactivate category Use case

UC071- Add new category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC071 | **Version** | 1.0 |
| **Use Case Name** | | Add new category | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to add new a category | | |
| **Preconditions:** | | Admin is viewing Category List | | |
| **Trigger:** | | Admin clicks on Add button | | |
| **Post conditions:** | | New category is added | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on Add Category button at right side of Category list table | | |
| 2 | WS | Display Add new category popup | | |
| 3 | Admin | * Type Category Name * Type Description * Choose picture from Admin’s computer * Click Add button | | |
| **Alternative Flows:** | | | | |

|  |  |  |
| --- | --- | --- |
| **AT1** | At step 3 in the main flows, if Admin click on Close button | |
| **Step** | **Actor** | **Action** |
| 3.1 | WS | Turn off Add new category popup |
|  | | |

|  |
| --- |
| **Exceptions:** |

|  |  |  |
| --- | --- | --- |
| **EC1** | At step 4 in main flow, if Admin does not type Category Name/ Description or does not choose picture | |
| **Step** | **Actor** | **Action** |
| 3.1 | WS | Display Add popup message with error message |
| 3.2 | WS | Mark error fields |

|  |  |
| --- | --- |
| **Business Rules:** | N/A |
| **Other Information:** | N/A |

Table 2- 71: Add new category Use case

UC072- Edit category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC072 | **Version** | 1.0 |
| **Use Case Name** | | Edit category | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to edit a category | | |
| **Preconditions:** | | Admin is viewing Category List | | |
| **Trigger:** | | Admin clicks on Edit button | | |
| **Post conditions:** | | Category is edited | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Click on Edit button at Edit column | | |
| 2 | WS | Display Edit category popup | | |
| 3 | Admin | * Type Category Name * Type Description * Choose picture from Admin’s computer * Click Add button | | |
| **Alternative Flows:** | | | | |

|  |  |  |
| --- | --- | --- |
| **AT1** | At step 3 in the main flows, if Admin click on Close button | |
| **Step** | **Actor** | **Action** |
| 3.1 | WS | * Turn off Edit category popup * Keep old information of this category |
|  | | |

|  |  |
| --- | --- |
| **Exceptions: N/A** | |
| **Business Rules:** | N/A |
| **Other Information:** | N/A |

Table 2- 72: Edit category Use case

###### Manage report

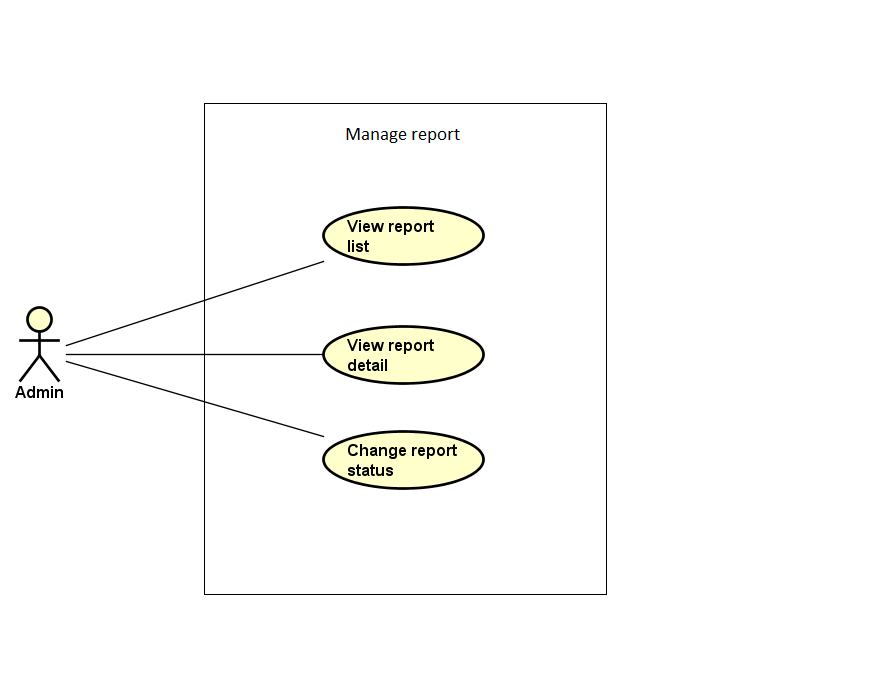


Figure 3-18: Manage report use case

UC083- View report list

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC083 | **Version** | 1.0 |
| **Use Case Name** | | View report list | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view Report list | | |
| **Preconditions:** | | 1. WS website is available 2. Admin browsed WS website 3. Admin logged in as Administrator | | |
| **Trigger:** | | Admin clicks on User/ Event | | |
| **Post conditions:** | | Report List is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | * Click on Report at left side * Click on User/ Event | | |
| 2 | WS | Display list of reports | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 86: View report list Use case

UC084- View report detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC084 | **Version** | 1.0 |
| **Use Case Name** | | View report detail | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to view information of a report | | |
| **Preconditions:** | | Admin is viewing Report List | | |
| **Trigger:** | | Admin clicks on button View at Action column | | |
| **Post conditions:** | | Detail of this report is displayed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on button View at Action column of the user/ event which want to view report’s detail information | | |
| 2 | WS | Display View Report Content popup | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 87: View report detail Use case

UC085- Change report's status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | UC085 | **Version** | 1.0 |
| **Use Case Name** | | Change report's status | | |
| **Created by:** | | DuyTN | **Date Created:** | 11/11/2016 |
| **Primary Actor** | | Administrator(Admin) | **Secondary Actor** |  |
| **Priority** | | High | | |
| **Frequency of Use:** | | High | | |
| **Description:** | | This function allows Administrator to change status of a report | | |
| **Preconditions:** | | Admin is viewing Report List | | |
| **Trigger:** | | Admin clicks on button at Status column | | |
| **Post conditions:** | | Status of this report is changed | | |
| **Normal Flow** | | | | |
| **Step** | **Actor** | **Action** | | |
| 1 | Admin | Admin clicks on button Waiting/ Cancel/ Confirmed at Status column of the user/ event which want to change report’s status | | |
| 2 | WS | * Display changed status as content of the button * Change status of report in database | | |
| **Alternative Flows: N/A** | | | | |
| **Exceptions: N/A** | | | | |
| **Business Rules:** | | N/A | | |
| **Other Information:** | | N/A | | |

Table 2- 89: Change report's status Use case

### NON-FUNCTIONAL SPECIFICATION

#### Reliability

1. Mean time between failures: The mean time between each failure is expected to be about 360 hours of using.
2. The maximum bugs per function are 2 bugs/a function when user uses software within 2 months. The average number of bugs in the software during system testing is about to be around 1 bugs/KLOC.
3. The database must be backed up daily and can be recovered if necessary

#### Performance Requirements

##### Availability

The application must be available 95% of time. Users can access to it everywhere from their

Web browser with internet connection.

##### Maintainability

* **Coding standards and naming conventions**
  + Output of the event must include coding standards and naming conventions documentations. Implementation code must be easy to maintain.
  + If some components are reused, the documentations of those components must also be included.
* **Design**
  + The design of the system must be loosely coupled that chances on some module will not affect others.
* **Logging**
  + All the errors should be logged, supporting for bug fixing and maintenance.
  + All strange or sensitive situations should also be logged.
* **Mean time to repair:** Immediately when Administrator finds out problem or website is attacked/ hacked by someone. Average 1 day.

##### Usability

Usability Requirements support the following from the perspective of its primary users:

● **Efficiency of use:** user can complete each function in less than 10 actions

● **Intuitiveness:** User can understand approximately 90% help/ error messages

##### Security Requirements

In order to avoid unwanted access by strangers, the security implemented in various ways:

* **User authorize**: Each function must be distributed system access and manipulate the individual parts. To edit information, user must declare their username and password before the operation. Functions will be show after users login dependencies user’s role and permission of each user.
* **Manage Data Layer**: Only Administrator can modify database system.

##### Hardware/Software Requirements

● **Hardware requirement:**

**- Server:** IIS Express

**- Client:** PC/Laptop (window 7, window 8, 8.1)

● **Software requirements:** Web browser (Chrome 53.0, Firefox 50.0) with internet connection

# SOFTWARE DESIGN

## Introduction

### Purpose

This chapter provides a guideline for the development team of what the system’s architect, how to implement functions.

This section of document is the software design of the project in order to develop the software, it includes four main parts:

* Architecture Design.
* Detail Design.
* Database design.
* Screen design.

### Definitions, Acronyms, Abbreviations

|  |  |  |
| --- | --- | --- |
| **Acronym** | **Definition** | **Note** |
| WS | WingS |  |
| DB | Database |  |
| MVC | Model view control |  |
| IDE | Integrated development environment |  |
| Q&A | Question and answer |  |
| GUI | Graphic user interface |  |
| FU | FPT University |  |

1. Definitions and Acronyms

## Architecture design

## Detail Design

## Data design

## Screen Design

# SOFTWARE TESTING

# USER MANUAL

# PROJECT RESULT REPORT

# REFERENCES

1. Kickstarter

<https://www.kickstarter.com/>

1. Comicola

<https://www.comicola.com/>

1. Indiegogo

<https://www.indiegogo.com/>

1. Wikipedia, 2012, IBM Rational Unified Process, viewed 30 September 2013

<http://en.wikipedia.org/wiki/IBM_Rational_Unified_Process>

1. AngularJS Overview

<http://www.tutorialspoint.com/angularjs/angularjs_overview.html>

1. Capstone Project Report Teamplate – FPT University, Ha Noi, Viet Nam.
2. Report document guidelines (2013) – FPT University, Ha Noi, Viet Nam.
3. Template\_Project\_Plan, (2010), FPT University, Hanoi, Vietnam
4. Template Software\_Requirements\_Specification, (2010), FPT Software, Hanoi, Vietnam
5. Template Software\_Design\_Template, (2010), FPT Software, Hanoi, Vietnam
6. Templates for Test case & Defect log, FPT Software, Hanoi, Vietnam
7. The Unified Modeling Language diagram

<http://www.uml-diagrams.org/>