



TRƯỜNG ĐẠI HỌC FPT

# Capstone Project Document

## DANDELION

Dandelion		
<b>Group Members</b>	Lưu Ngọc Mạnh	SE02619
	Vũ Công Chính	SE02585
	Nguyễn Minh Huy	SE02723
	Đương Đức Anh	SE02700
	Vũ Ngọc Trung	SE02967
	Cao Thị Phương Mai	SE02908
<b>Supervisor</b>	Mr. Nguyễn Văn Sang	
<b>Project code</b>	DDL	

- Hanoi, 12/2015 -

## TABLE CONTENTS

<b>1 INTRODUCTION .....</b>	<b>5</b>
1.1 Purpose .....	5
1.2 Acronyms and Definitions .....	5
1.3 Abstract.....	5
1.4 Literature Review .....	5
1.4.1 Kickstarter.....	6
1.4.2 Comicola.....	7
1.5 Proposal .....	7
1.5.1 The idea .....	7
1.5.2 The objective .....	7
1.5.3 Brief description about system .....	8
1.5.4 System features.....	9
1.6 Benefits from Project.....	10
1.6.1 For our group .....	10
1.6.2 For Community.....	10
1.7 Critical Assumption and Constraints .....	10
1.8 Potential Risks .....	11
<b>2 PROJECT MANAGEMENT .....</b>	<b>12</b>
2.1 Introduction .....	12
2.1.1 Purpose .....	12
2.1.2 Definitions and Acronyms.....	12
2.2 Project Overview .....	12
2.2.1 Project Description .....	12
2.2.2 Scope and Purpose.....	13
2.2.3 Assumptions and Constraints .....	15
2.2.4 Project Objectives.....	16
2.2.5 Critical Dependencies .....	17
2.2.6 Project Risk.....	18
2.3 Project Development Approach.....	19
2.3.1 Project Process.....	19
2.3.2 Requirement Change Management.....	21
2.3.3 Quality Management .....	21
2.4 Estimation.....	24
2.4.1 Size .....	24
2.4.2 Effort.....	24
2.4.3 Schedule.....	25

2.4.4 Resource .....	0
2.4.5 Infrastructure.....	0
2.4.6 Training Plan .....	1
2.4.7 Finance.....	1
2.5 Project Organization .....	2
2.5.1 Organization Structure.....	2
2.5.2 Project Team.....	2
2.6 Communication & Reporting .....	5
<b>3 SOFTWARE REQUIREMENT.....</b>	<b>7</b>
3.1 Introduction .....	7
3.1.1 Purpose .....	7
3.1.2 Definitions and Acronyms.....	7
3.2 User Requirement Specification .....	8
3.2.1 Business Process Overview .....	8
3.2.2 Product Features .....	19
3.2.3 User characteristic .....	21
3.2.4 Functional Requirements .....	21
3.2.5 Non-functional Requirement .....	22
3.3 Software Requirements Specification.....	23
3.3.1 Overall Description.....	23
3.3.2 Functional Specification .....	26
3.3.3 Non-Functional Specification .....	105
<b>4 SOFTWARE DESIGN.....</b>	<b>107</b>
4.1 Introduction .....	107
4.1.1 Purpose .....	107
4.1.2 Definitions, Acronyms, Abbreviations .....	107
4.2 Architecture design.....	107
4.2.1 Choice of Architecture Design .....	107
4.2.2 Architecture Representation .....	113
4.2.3 Architectural Goals and Constraints .....	114
4.2.4 Logical View .....	115
4.2.5 Process view .....	119
4.2.6 Deployment View .....	121
4.3 Detail Design .....	122
4.3.1 Package .....	122
4.3.2 Function .....	152
4.4 Data design .....	183

4.4.1 Entity Relationship Diagram .....	183
4.4.2 Table Diagram .....	185
4.5 Screen Design .....	193
4.5.1 User Client .....	193
4.5.2 Admin .....	225
<b>5 SOFTWARE TESTING .....</b>	<b>241</b>
5.1 Introduction .....	241
5.1.1 Purpose .....	241
5.1.2 Definitions and Acronyms.....	241
5.1.3 Scope of testing.....	242
5.1.4 Constraints .....	243
5.2 Test plan .....	244
5.2.1 Test type .....	244
5.2.2 Test stages.....	246
5.2.3 Test model .....	246
5.2.4 Acceptance test criteria.....	248
5.2.5 Futures to be tested .....	248
5.2.6 Feature not to be tested .....	248
5.2.7 Resources.....	248
5.2.8 Test milestones .....	249
5.2.9 Deliverables .....	250
5.3 Test case .....	250
5.4 Test Report .....	251
5.4.1 System test report .....	251
5.4.2 Integration test report.....	251
5.4.3 Test report.....	251
5.4.4 Defect report .....	251
<b>6 USER MANUAL .....</b>	<b>253</b>
6.1 Introduction .....	253
6.1.1 Purpose .....	253
6.1.2 Environment .....	253
6.2 Installation Guideline .....	253
6.2.1 Connecting an ASP.NET MVC Web App with SQL Azure .....	253
6.2.2 Create & Deploying the Azure Website .....	258
6.2.3 Connecting SQL Management Studio to SQL Azure.....	260
6.3 User Guideline.....	264
6.3.1 User Client .....	264

6.3.2 Admin .....	286
<b>7 PROJECT RESULT REPORT.....</b>	<b>293</b>
7.1 Product Judging .....	293
7.1.1 Advantage of Dandelion system.....	293
7.1.2 Current Limitations.....	293
7.1.3 Expectation in Future.....	293
7.2 Project Summary .....	293
7.3 Lesson Learnt .....	294
<b>8 REFERENCES .....</b>	<b>296</b>

## 1 INTRODUCTION

### 1.1 Purpose

This part of document is the introduction for project Dandelion – our Capstone Project at FPT University. In this document, we will describe the overview of some existing systems, the initial idea for our project, a brief description about our expected system and some potential risks, critical assumptions, constraints. Moreover, this document also shows opportunities what it offers for users.

### 1.2 Acronyms and Definitions

Acronym & Abbreviation	Definition
DDL	Dandelion
FU	FPT University
Q&A	Question and Answer

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

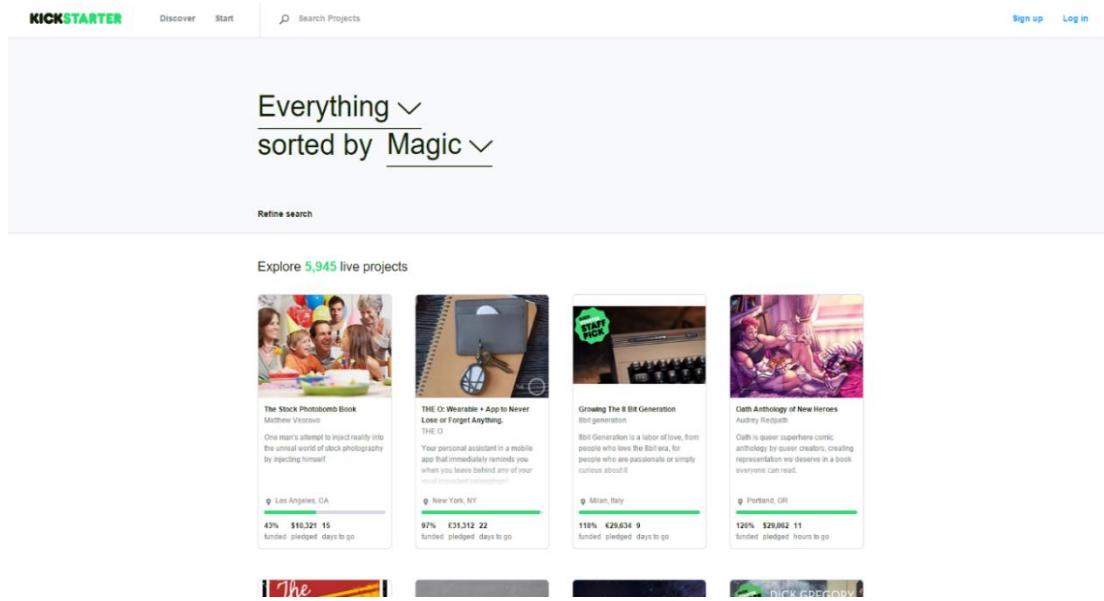
### 1.3 Abstract

We're a home for everything from films, games, and music to art, design, and technology. Dandelion is full of projects, big and small, that are brought to life through the direct support of people like you. Dandelion is a platform and a resource; we're not involved in the development of the projects themselves. Anyone can launch a project on Dandelion as long as it follows our rules. Project creators set a funding goal and deadline. If people like a project, they can pledge money to make it happen. Backers are supporting projects to help them come to life, not to profit financially. Instead, project creators offer rewards to thank backers for their support. We believe that creative projects make for a better world, and we're thrilled to help support new ones. Building a community of backers around an idea is an amazing way to make something new.

### 1.4 Literature Review

There are quite a few major platforms for crowdfunding your project idea. However, just have one crowdfunding in Vietnam and that major platform just aims to comic project. For example: kickstarter.com, comicola.com, indiegogo.com, etc.

### 1.4.1 Kickstarter

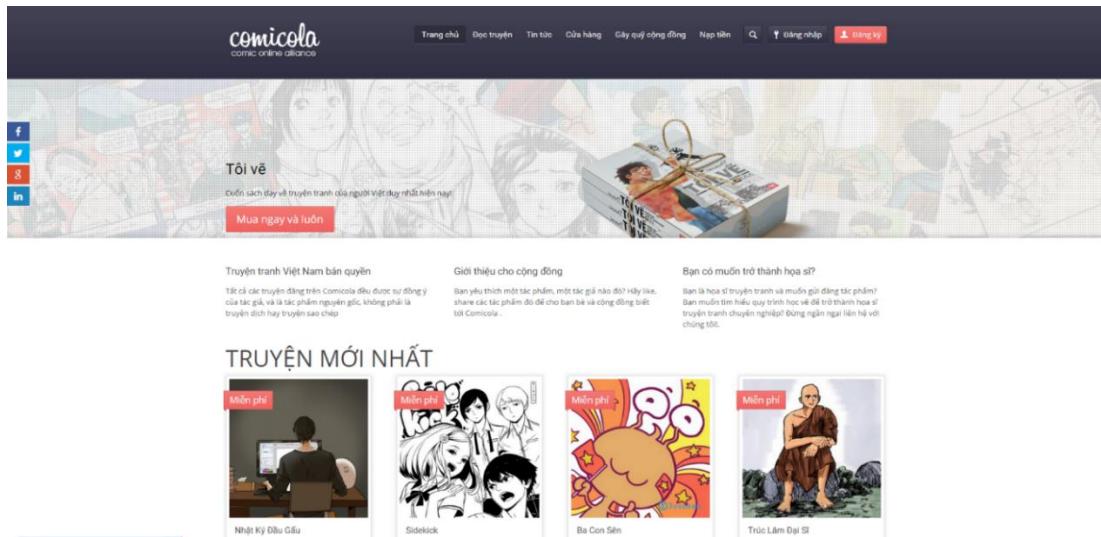


**Figure 1-1: Website Kickstarter**

Kickstarter is the biggest major platform for crowdfunding your project idea, users can create a project, Kickstarter staff look at your proposed project and provide feedback. Then they (hopefully) approve your project and it's posted on the site. If that project don't hit the funding goal in the specified time frame, no one's cards get charged and user don't receive any of the funds.

- **Advantages:**
  - The biggest major platform for crowdfunding
  - Kickstarter positions itself mostly as a community for creative projects.
  - Kickstarter staff provide feedback to help your project succeed.
- **Disadvantages:**
  - You have to be based in the US, Canada, the UK, Australia, New Zealand or the Netherlands.
  - Only support PayPal, MasterCard, etc. These are not popular in Vietnam.

### 1.4.2 Comicola



**Figure 1-2: Website Comicola**

Comicola is a major platform for crowdfunding your comic project idea in Vietnam.

- **Advantages:**

- Has friendly user interface – Vietnamese.
- Making payments easier for creators and backers.

- **Disadvantages:**

- Just for comic project.

## 1.5 Proposal

### 1.5.1 The idea

Nowadays people have a lot of idea to make things better. But lots of people are short of funds to raise their idea. Many thanks for internet they can easy to show these ideas to everyone. Our project – Dandelion is a major platform for crowdfunding. Anybody could launch a project on Dandelion as long as it follows our rules. We believe that creative projects making a better world, and we're thrilled to help support new ones. Building a community of backers around an idea is an amazing way to make something new.

### 1.5.2 The objective

This project is the Capstone Project in FPT University studying program.

First, this project is responsibility for all team members. So we must complete all requirements from teacher and FU. During the time doing project, we can learn how to develop a project, how to manage project, how to communicate better between team member, 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 project is very necessary for us to improve skills, knowledge which we studied in FU. This project also makes our CV more beautiful after graduating from FU. This is also an opportunity for us to prove capacity with recruiters.

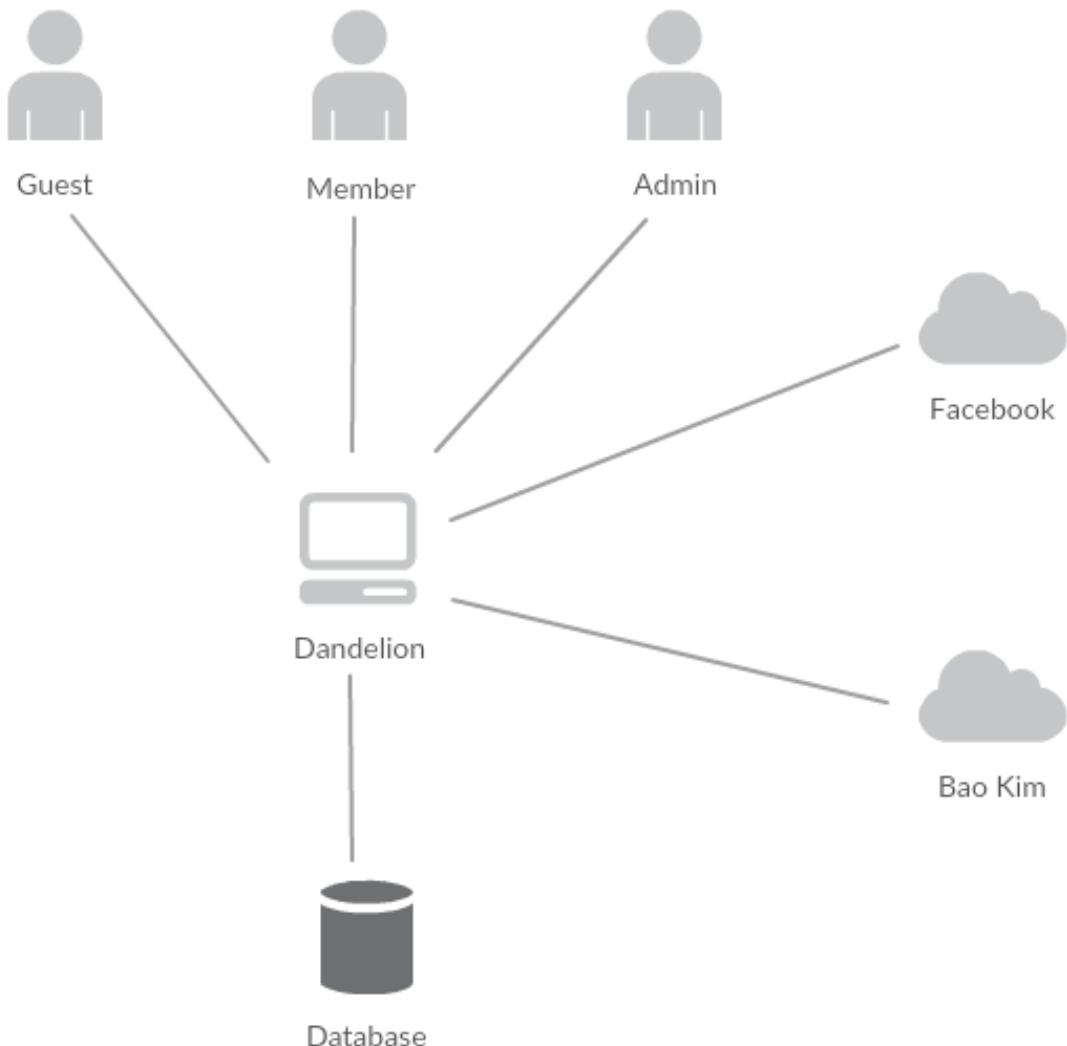
Second, because we'd love to support people's dream, especially Vietnamese users to create something that they want to see exist in the world. Everything from films, games, and music to

art, design, and technology. And we want to build a community where people rally around their friends' projects, fans support people they admire, and others simply come to DDL to be inspired by new ideas. Anyone can create a new project, update, share a project to social network or back a project. Specially, our website can display completely in Vietnamese, so that even users do not know English, they can easily use it.

### 1.5.3 Brief description about system

This software is a major platform for crowdfunding with friendly and attractive interface, users can easily find out interesting project to back or create a new project with their idea. User also can send opinion to admin if they need help.

DDL 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 DDL system**

### 1.5.4 System features

About the features of DDL, we focus to provide for users main features as searching, filter project, create new project, back project, etc. Admin can manage members, manage projects, manage database, manage contribute database and manage Q&A.

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

#### 1.5.4.1 Client features

- Register: User can register an account and login to use all features of DDL.
- Login/Logout: Users login/logout an account to use or exit system DDL.
- Search: User and guest can search with code of project, name of project, name of creator; and system will display all of project relate to keywords.
- Filter: User can filter with category (Art, Comics, Game, etc.), Status (Popularity, Most Funded, etc.). And system will display projects relate to these selected options.
- Project
  - Create New Project: Users can create new projects with basic info, reward, timeline, clip and some pictures.
  - Update Project Information: User can change and update information of projects which created.
  - Comment/Remind: Users can comment/remind a project to follow and discuss with another users.
  - Report: User can report a project which violate regulation to administrator.
- Back Project: User can back another project to donate money to help project success and get rewards.
- Profile:
  - Change password: User can change password to keep security.
  - Update profile: User can change or update information.
  - Manage created projects, backed projects, reminded projects: User can follow process of projects, follow project backed, project reminded. View statistical projects had created, backed, reminded.
  - Sent/Receive message: Users can send to or receive from other users to discuss about projects.

#### 1.5.4.2 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 project:
  - Search project :
    - o Search with project name: Admin can search with name of project and system will display simple information about that project.
    - o Search with category: Admin can search with category of project and system will display all project with that category
    - o Search with Creator name: Admin can search with Creator name of project and system will display all project of that user.

- Edit project status: Admin can set status of a project: pending to approved, pending to suspend, approved to suspend, etc.
- Manage Backing:
  - Admin can view users who backed a project. Also, Admin can view simple information about that user and which reward they chose.
- Manage Categories:
  - Add new category: Admin can add new category.
  - Inactive/active category: Admin can set category's status to inactive or active
- 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.

## 1.6 Benefits from Project

### 1.6.1 For our group

After developing and implementing this project, our group will get some benefits:

- Have more experiences of software project 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.

### 1.6.2 For Community

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

- Easy join to a community of crowdfunding
- Easy search for interesting project information and back them
- Easy to start a new project idea

## 1.7 Critical Assumption and Constraints

- Critical assumption:
  - Training: All developers can train .NET and AngularJS in 2 weeks. Before starting this project, just 1 member in our team have knowledge about .NET and no one 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 40 and Firefox 30 or higher.
- Process: We have to follow the software processing of FPT Software
- Human resources: There are 6 member in our team, each member have to study 2 subjects (Japanese and JFE) at school.

## 1.8 Potential Risks

After studying about this project, 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

## 2 PROJECT MANAGEMENT

### 2.1 Introduction

#### 2.1.1 Purpose

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

#### 2.1.2 Definitions and Acronyms

Acronym	Definition	Note
<b>BA</b>	Business Analyst	
<b>BU</b>	Business Unit	
<b>CC</b>	Infrastructure Configuration Controller	
<b>CM</b>	Configuration Management	
<b>DDL</b>	Dandelion	
<b>DEV</b>	Developer	
<b>PIC</b>	Person in charge	
<b>PM</b>	Project Manager	
<b>PTL</b>	Project Technical Leader	
<b>QA</b>	Quality Assurance Officer	
<b>SRS</b>	Software Requirement Specification	
<b>TC</b>	Test Case	
<b>PCB</b>	Process Capability Baseline	

**Table 2-1: Definitions and acronyms**

### 2.2 Project Overview

#### 2.2.1 Project Description

<b>Project Code</b>	DDL	<b>Contract Type</b>	None
<b>Customer</b>	FPT University	<b>2nd Customer</b>	None
<b>Project Level</b>	Group	<b>Project Rank</b>	None
<b>Application Type</b>	Website	<b>Project Manager</b>	Luu Ngoc Manh
<b>Project Category</b>	Development	<b>Business Domain</b>	E-commerce

**Table 2-2: Project Description**

## 2.2.2 Scope and Purpose

### 2.2.2.1 Purpose

Nowadays people have a lot of idea to make things better. But many of people are short of funds to raise their idea. However, many thanks for internet they can easy to show these ideas to everyone. Our project – Dandelion is a major platform for crowdfunding. Anybody could launch a project on Dandelion as long as it follows our rules. We believe that creative projects making a better world, and we're thrilled to help support new ones. Building a community of backers around an idea is an amazing way to make something new.

### 2.2.2.2 Scope

Our project scope included:

- Developing user requirement and software requirement specification.
- Developing architecture and detailed design documents.
- Coding and unit test.
- Developing test case and execute system test.
- The developing software will have these following functions.

### 2.2.2.3 The functions of Project

These are the functions of DDL's project:

#### 2.2.2.3.1 Client Module.

- **Register:** User can register an account and login to use all features of DDL.
- **Login/Logout:** Users login/logout an account to use or exit system DDL.
- **Search:** User and guest can search with code of project, name of project, name of creator; and system will display all of project relate to keywords.
- **Filter:** User can filter with category (Art, Comics, Game, etc.), place (Vietnam, etc.), Status (Magic, Popularity, Most Funded, etc.). And system will display projects relate to these selected options.
- **Project:**
  - **Create New Project:** Users can create new projects with basic info, reward, timeline, clip and some pictures.
  - **Update Project Information:** User can change and update information of projects which created.
  - **Comment/Remind:** Users can comment/remind a project to follow and discuss with another users.
  - **Report:** User can report a project which violate regulation to administrator.
- **Back Project:** User can back another project to donate money to help project success and get rewards.
- **Profile:**
  - **Change password:** User can change password to keep security.
  - **Update profile:** User can change or update information.
  - **Manage created projects, backed projects, reminded projects:** User can follow process of projects, follow project backed, project reminded. View statistical projects had created, backed, reminded.
  - **Sent/Receive message:** Users can send or receive from another users to discuss about project information.

### 2.2.2.3.2 Admin Module

- **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 project:**
  - Search project :
    - o Search with project name: Admin can search with name of project and system will display simple information about that project.
    - o Search with category: Admin can search with category of project and system will display all project with that category
    - o Search with Creator name: Admin can search with Creator name of project and system will display all project of that user.
  - Edit project status: Admin can set status of a project: pending to approved, pending to suspend, approved to suspend, etc.
- **Manage Backing:** Admin can view users who backed a project. Also, Admin can view simple information about that user and which reward they chose.
- **Manage Categories:**
  - Add new category: Admin can add new category.
  - Inactive/active category: Admin can set category's status to inactive or active
- **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

### 2.2.3 Assumptions and Constraints

No	Description	Note
<b>Assumptions</b>		
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
<b>Constraints</b>		
1	This project must be completed and delivered before 25/12/2015	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 completed within 12 days since 10/09/2015  <b>Deadline:</b> 28/09/2015	Schedule
4	Design Document (include Architecture Design, Screen Design, Database Design) must be completed within 18 days since 25/09/2015  <b>Deadline:</b> 20/10/2015	Schedule
5	Integration Test Plan (include test plan and test case...) must be completed within 15 days since 22/10/2015  <b>Deadline:</b> 16/11/2015	Schedule
6	Completed coding activity and have unit test result within 20 days since 03/11/2015  <b>Deadline:</b> 30/11/2015	Schedule
8	Deliver report about User manual, software package and installation guide on 5 days since 13/11/2015  <b>Deadline:</b> 16/12/2015	Schedule

<b>9</b>	Complete all of document and application before finishing the project on 23/12/2015	Schedule
<b>10</b>	Project contains 6 members	Resource

**Table 2-3: Project Description****2.2.4 Project Objectives****2.2.4.1 Standard Objectives**

Metrics	Unit	Committed	Re-committed	Note
<b>Start Date</b>		07-09-2015		
<b>End Date</b>		23-12-2015		
<b>Duration</b>	Day	78		
<b>Team Size</b>	Person	6		
<b>Billable Effort</b>	Person-day	480		1 Person-day = 5 hours
<b>Calendar effort</b>	Person-day	480		1 Person-day = 5 hours
<b>Effort Usage</b>	%	100		1 Person-day = 5 hours

**Table 2-4: Standard Objectives**

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

**Table 2-5: Standard Objectives 2**

#### 2.2.4.2 Specific Objectives

Metrics	Unit	Basic for setting Goals	
		Plan	Actual
Training technology: MVC, Bootstrap, jQuery, AngularJS	Person-day	15	15
Execute group review	Person-day	8	5
Training requirements, process before coding	Person-day	8	4

**Table 2-6: Specific Objectives**

#### 2.2.5 Critical Dependencies

No	Dependency	Expected delivery date	Note
1	This project must be completed and delivered to FPT University.	25/12/2015	
2	Project Plan and SRS must be completed and delivered to Supervisor.	28/09/2015	
3	User manual, Software Package and Installation Guide must be completed and delivered to Supervisor and FPT University.	18/12/2015	
4	Beside Capstone Project, Team members have to joining in Japanese class and Japanese Fundamental Exam class.	07/09/2015	

**Table 2-7: Critical Dependencies**

## 2.2.6 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 [DDL\\_Risk Management Plan\\_v1.0\\_EN.xlsx](#)

No	Risk	Root Cause	Triggers	Avoidance plan	Contingency plan	Fallback plan
1	Conflict among team members	Team member doesn't understand each others	Has issue inside team	Plan some team buildings to improve teammate	Have a meeting to resolve conflict problem	Switch member to other team.
2	Requirement is changed after project is started	SRS not good (not realistic, not feasibility, not meet customer needs)	Customer (user) make change request. Or project team cannot develop the system as description in SRS.	Requirement should be paid a lot of attention. SRS should be reviewed carefully by PM and the supervisor.	With each requirement changes, the project team should have a meeting to analysis the change requests and take actions.	Re-estimate and update plan.
3	Team member do not meet the deadline	Undisciplined and habit of team member, loose management	Have a deadline	Setting rule and minus of do not meet the deadline: lost money and named at meeting minute.	Punish the member do not meet the deadline. Working over time.	Ask for help from out of team, supervisors.
4	Inability to verify/validate products against requirements.		Inability to verify/validate products against requirements.	Print out the result to check by normal eyes. Debug into code to see how the code is running	Print out the result to check by normal eyes. Debug into code to see how the code is running	Ask for help from supervisors

5	The server has defective parts.	Defective power supply...	Server is down...	Ensure the supplier provides with the adequate warranties.	Replace or fix the defective parts quickly.	Buy a new one.
6	Delivery project doesn't meet requirement	Team member not understand requirement	Can't delivery project	Teamlead have to review sourcecode and follow requirement	Fix in next version	Extend dealine

Table 2-8: Risk Management plan

## 2.3 Project Development Approach

### 2.3.1 Project Process

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

#### 2.3.1.1 FPT Software Process Model

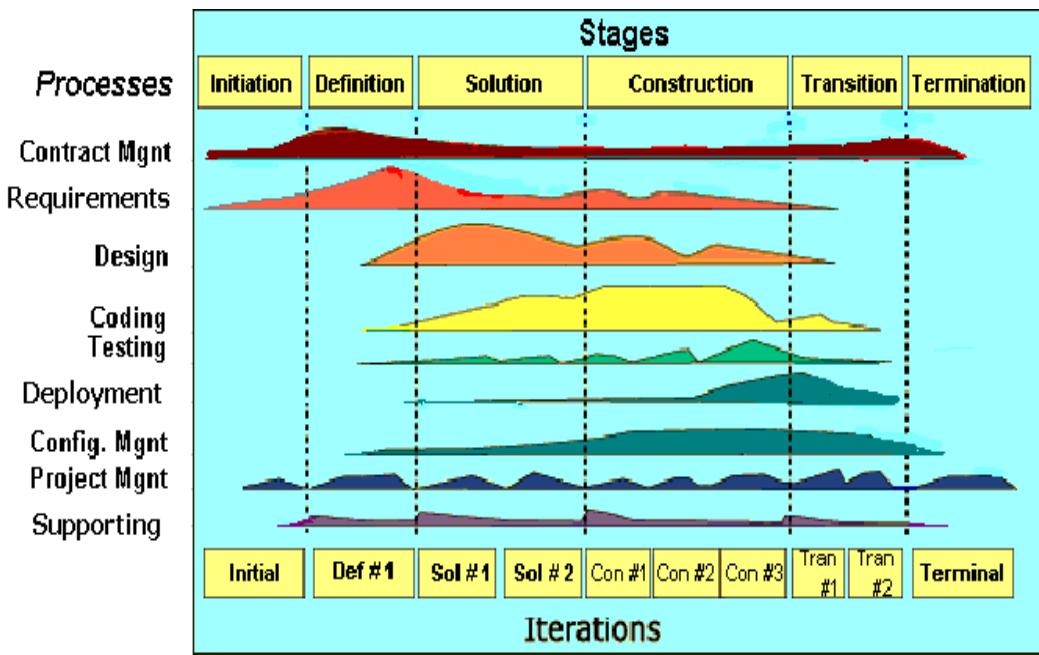


Figure 2-1: FPT Software process model

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

### 2.3.1.2 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
- ❖ **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 complete Requirement 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 #2
- ❖ **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 #3, #4
- ❖ **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 #5, #6

### 2.3.2 Requirement Change Management

<b>Who logs the change request?</b>	<b>Any team members</b>
<b>Who reviews the change request?</b>	PM or who is PM assign
<b>Who approves the change request?</b>	PM by default. PTL if: <ul style="list-style-type: none"> <li>▪ Changes to project scope</li> <li>▪ Changes in delivery plan of project deliverables</li> <li>▪ Changes to assignment for key roles (PM, PTL)</li> </ul>

**Table 2-9: Requirement Change Management**

### 2.3.3 Quality Management

#### 2.3.3.1 Defect Prevention Strategy

<b>Item (Process/Product)</b>	<b>Strategy</b>	<b>Expected Benefits</b>
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.

**Table 2-10:Defect Prevention Strategy**

### 2.3.3.2 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	

**Table 2-11: Review Strategy**

### 2.3.3.3 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	<ul style="list-style-type: none"> <li>- Number of UTC/KLOC: 20 UTC/KLOC</li> <li>- Number defects/KLOC: 3-4 defects/KLOC</li> <li>- Statement coverage: 97%</li> <li>- Branch coverage: 100%</li> <li>- Path coverage: 100%</li> </ul>

**Table 2-12: Unit Testing Strategy**

### 2.3.3.4 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

**Table 2-13: Integration Testing Strategy**

### 2.3.3.5 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

**Table 2-14: System Testing Strategy**

### 2.3.3.6 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%	Referenced to similar project estimations
Design review	15	11%	Referenced to similar project estimations
Code review	30	22%	Referenced to similar project estimations
Unit Test	50	38%	Referenced to similar project estimations
Integration Test	15	11%	Referenced to similar project estimations
System Test	10	7%	Referenced to similar project estimations
User Acceptance Test	5	4%	Referenced to similar project estimations
Total	135	100%	

**Table 2-15: Estimates of Defects**

### 2.3.3.7 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

**Table 2-16: Measurements Program**

## 2.4 Estimation

### 2.4.1 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.

### 2.4.2 Effort

The Effort estimation is documented in the table below:

	Initiation	Solution	Construction	Termination	Total
Effort(person/day)	72	168	186	54	<b>480</b>
Total % budgeted Effort Usage (%)	100	100	100	100	

**Table 2-17: Effort Estimation**

### 2.4.3 Schedule

#### 2.4.3.1 Project Milestone & Deliverables

No	Stage	Committed Delivery date	Description of Deliverable	Delivery media
<b>Initiation</b>			Requirements agreed, Report 1 reviewed	
1	Develop project idea	08-09-2015	Project goals and scope defined, milestone description defined, resource committed	
2	Q&A Management Sheet	14-09-2015	Criteria: Documentation reviewed	
3	Submit report no.1 final	22-09-2015	Completed report no.1	
4	Project Plan	21-09-2015	Criteria: Documentation reviewed	
5	Submit report no.2 final	19-10-2015	Completed report no.2	
<b>Solution</b>				
1	Screen Prototype	05-10-2015	Criteria: Documentation reviewed	
2	Architecture Design	08-10-2015	Criteria: Documentation reviewed	
3	Screen Design	09-10-2015	Criteria: Documentation reviewed	
4	Class Design	16-10-2015	Criteria: Documentation reviewed	
5	Detail Data Design	05-10-2015	Criteria: Documentation reviewed	
6	SRS	28-09-2015	Criteria: Documentation reviewed	
7	Submit Report no.3 Final	28-10-2015	Completed report no.3	
8	Submit Test Plan Final	28-10-2015	Criteria: Documentation reviewed	
9	Create Test Case	12-11-2015	Criteria: Documentation reviewed	
10	Submit Report no.4 Final	25-11-2015	Completed report no.4	
<b>Construction</b>			Product developed & tested and released to supervisor, documentation reviewed.	
1	Review Test Document	12-11-2015	Criteria: Documentation reviewed	

2	Submit report no.5 Final	10-12-2015	Completed report no.5	
3	Complete Coding and Unit Test	24-11-2015	Source code Acceptance criteria: Product unit tested	
4	Complete Testing	04-12-2015	Completed Test	
5	Submit report no.6 Final	14-12-2015	Completed report no.6	
6	Submit the last document and CD source code	15-12-2015	Final Documents and Source Code	
<b>Termination</b>			Project post-mortem is conducted, Project assets archived and released to supervisor	
1	Lesson learned	11-12-2015	Criteria: Completed	
2	Complete Presentation Slide	14-12-2015	Criteria: Completed	
3	Represent capstone project	24-12-2015	Criteria: Completed	
4	Project Complete	25-12-2015	Criteria: Completed	

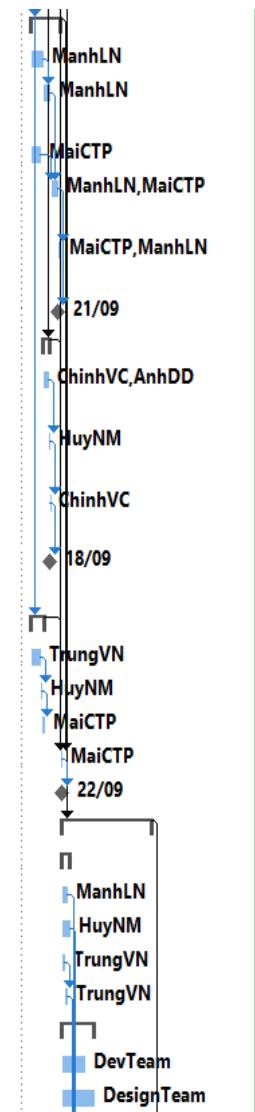
**Table 2-18: Project Milestone and Deliverables**

### 2.4.3.2 Detail Schedule

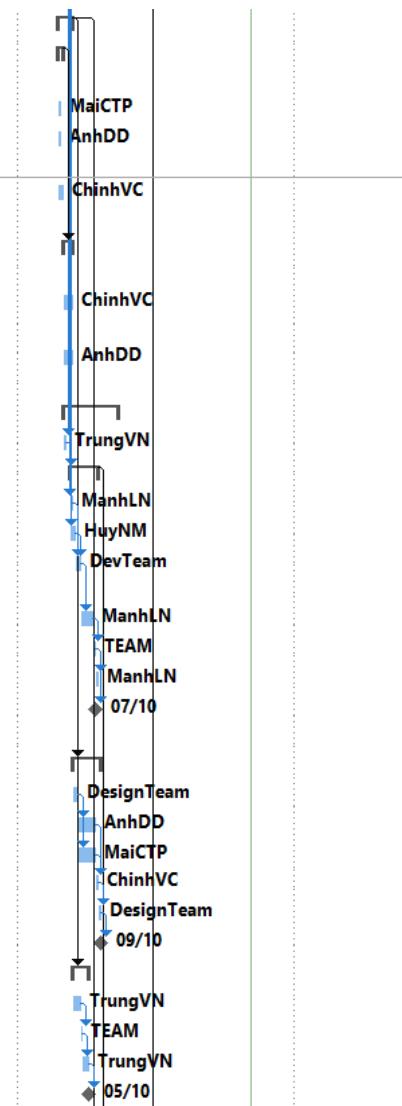
The detail project schedule is available in file [DDL\\_ProjectSchedule\\_v1.0\\_EN.mpp](#).

		Task Mode	Task Name	Duration	Start	Finish	Predecessor	Resource Names	2 Jul '15	16 Aug '15	20 Sep '15	25 Oct '15	29 Nov '15	03 Jan '16	07 Feb '16
									F	S	T	T	S	M	W
1			▫ Dandelion (DDL)	80 days?	Mon 07/09/15	Fri 25/12/15									
2			▫ Initiation	12 days?	Mon 07/09/15	Tue 22/09/15									
3			Form up team roles	0.5 days	Mon 07/09/15	Mon 07/09/15		TEAM							
4			Select project idea	0.5 days	Mon 07/09/15	Mon 07/09/15	3	TEAM							
5			▫ Develop project idea	1 day	Tue 08/09/15	Tue 08/09/15	4								
6			Define project scope	0.5 days	Tue 08/09/15	Tue 08/09/15		TEAM							
7			Define project features	0.5 days	Tue 08/09/15	Tue 08/09/15	6	TEAM							
8			Kick-off meeting	1 day?	Wed 09/09/15	Wed 09/09/15	5	TEAM							
9			▫ Environment	2 days?	Thu 10/09/15	Fri 11/09/15	8								
10			Create guideline to install and use tools & environments	0.5 days?	Thu 10/09/15	Thu 10/09/15		ChinhVC							
11			▫ Install tools & environments	1 day	Thu 10/09/15	Fri 11/09/15	10								
12			Install TortoiseGit	0.5 hrs	Thu 10/09/15	Thu 10/09/15		TEAM							
13			Install Visual Studio 2013	3 hrs	Thu 10/09/15	Thu 10/09/15	12	TEAM							
14			Install Microsoft SQL Server 2010	2 hrs	Thu 10/09/15	Fri 11/09/15	13	TEAM							
15			Install Microsoft Project 2013	2 hrs	Fri 11/09/15	Fri 11/09/15	14	TEAM							
16			Install Sublime Text	0.5 hrs	Fri 11/09/15	Fri 11/09/15	15	TEAM							
17			Tranning using tools & environments	0.5 days	Fri 11/09/15	Fri 11/09/15	11	TEAM							
18			▫ Technology Research (Part 1)	7 days	Mon 14/09/15	Tue 22/09/15	9								
19				Study .net MVC 5	7 days	Mon 14/09/15	Tue 22/09/15		DevTeam						
20			Study AngularJS	7 days	Mon 14/09/15	Tue 22/09/15		DevTeam							
21			Study Bootstraps, HTML5 & CSS3	5 days	Mon 14/09/15	Fri 18/09/15		DesignTeam							
22			SRS/PP Review checklist	0.5 days?	Thu 10/09/15	Thu 10/09/15	8	ChinhVC							
23			▫ Project Introduction	3 days	Thu 10/09/15	Tue 15/09/15	22								
24			Create Project Introduction	2 days	Thu 10/09/15	Mon 14/09/15		HuyNM							
25			Review Project Introduction	0.5 days	Mon 14/09/15	Mon 14/09/15	24	TrungVN,MaiCTP							
26			Update Project Introduction	0.5 days	Tue 15/09/15	Tue 15/09/15	25	HuyNM							
27			Complete Project Introduction	0 days	Tue 15/09/15	Tue 15/09/15	26								

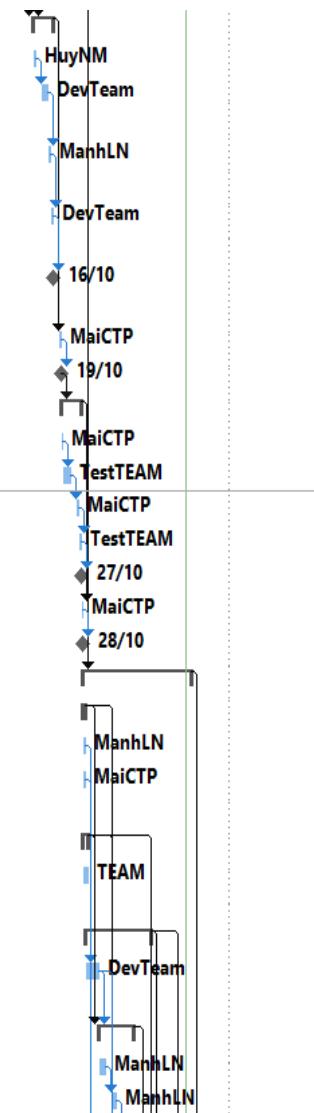
28		☛ Project Plan	8 days?	Thu 10/09/15	Tue 22/09/15	22		
29		Create Project Plan in MS Word	3 days	Thu 10/09/15	Tue 15/09/15			ManhLN
30		Create Project Schedule in .mmp file	3 days	Tue 15/09/15	Fri 18/09/15	29		ManhLN
31		Create CM Plan	2 days	Thu 10/09/15	Mon 14/09/15			MaiCTP
32		Review Project Plan, CMPlan, Schedule	1 day	Fri 18/09/15	Mon 21/09/15	31,30		ManhLN,MaiCTP
33		Update Project Plan, CMPlan, Schedule	1 day?	Mon 21/09/15	Tue 22/09/15	32		MaiCTP,ManhLN
34		Complete Project Plan	0 days	Mon 21/09/15	Mon 21/09/15	32		
35		☛ User Requirement Specification	3 days?	Tue 15/09/15	Fri 18/09/15	23		
36		Create User Requirement Sepcification	2 days	Tue 15/09/15	Thu 17/09/15			ChinhVC,AnhDD
37		Review User Requirement Specification	0.5 days?	Thu 17/09/15	Thu 17/09/15	36		HuyNM
38		Update User Requirement Specification	0.5 days?	Fri 18/09/15	Fri 18/09/15	37		ChinhVC
39		Complete User Requirement Specification	0 days	Fri 18/09/15	Fri 18/09/15	38		
40		☛ Q&A Sheet	3.5 days?	Thu 10/09/15	Tue 15/09/15	22		
41		Create Q&A sheet	2 days	Thu 10/09/15	Mon 14/09/15			TrungVN
42		Review Q&A sheet	0.5 days?	Mon 14/09/15	Mon 14/09/15	41		HuyNM
43		Translate Q&A sheet to Japanese	1 day	Tue 15/09/15	Tue 15/09/15	42		MaiCTP
44		Create progress report 1	0.5 days	Tue 22/09/15	Tue 22/09/15	23,28,35,40		MaiCTP
45		Submit progress report 1	0 days	Tue 22/09/15	Tue 22/09/15	44		ManhLN
46		☛ Solution	26 days?	Wed 23/09/15	Wed 28/10/15	2		
47		☛ Technology Research (Part 2)	3 days?	Wed 23/09/15	Fri 25/09/15			
48		Code first migration	2 days	Wed 23/09/15	Thu 24/09/15			ManhLN
49		Restful API on .net MVC	3 days	Wed 23/09/15	Fri 25/09/15			HuyNM
50		Payment by BaoKim API	1 day?	Wed 23/09/15	Wed 23/09/15			TrungVN
51		Facebook Login API	1 day?	Thu 24/09/15	Thu 24/09/15	50		TrungVN
52		☛ Demo & Prototypes	9 days	Wed 23/09/15	Mon 05/10/15			
53	!	Create Demo	7 days	Wed 23/09/15	Thu 01/10/15			DevTeam
54	!	Create Prototype	9 days	Wed 23/09/15	Mon 05/10/15			DesignTeam



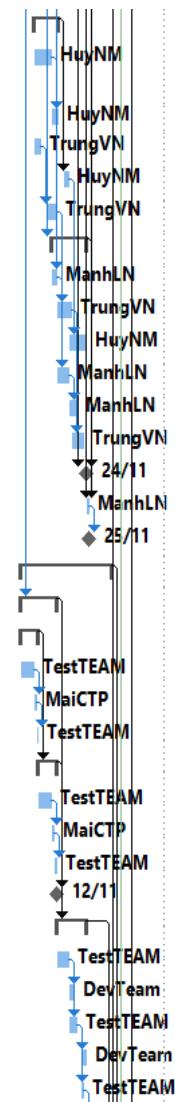
55		☛	☛ Software Requirement Specification	4 days	Wed 23/09/15	Mon 28/09/15			
56		☛	☛ Non-functional Requirements Specification	2 days	Wed 23/09/15	Thu 24/09/15			
57		☛	Define Security Requirements	1 day	Wed 23/09/15	Wed 23/09/15		MaiCTP	
58		☛	Define Performance Requirements	1 day	Wed 23/09/15	Wed 23/09/15		AnhDD	
59		☛	Define User-interface Requirements	2 days	Wed 23/09/15	Thu 24/09/15		ChinhVC	
60		☛	☛ Functional Requirements Specification	2 days	Fri 25/09/15	Mon 28/09/15	56		
61		☛	Define User Client Module functional Requirements	2 days	Fri 25/09/15	Mon 28/09/15		ChinhVC	
62		☛	Define Admin Module functional Requirements	2 days	Fri 25/09/15	Mon 28/09/15		AnhDD	
63		☛	☛ System Design	16 days?	Fri 25/09/15	Fri 16/10/15			
64		☛	Create design review checklist	1 day?	Fri 25/09/15	Fri 25/09/15	51	TrungVN	
65		☛	☛ Architectural Design	9 days?	Mon 28/09/15	Thu 08/10/15	64		
66		☛	Define architectural design	1 day	Mon 28/09/15	Mon 28/09/15	48	ManhLN	
67		☛	Define system features	2 days	Mon 28/09/15	Tue 29/09/15	49	HuyNM	
68	!	☛	Study and choose suitable architectural design	2 days	Wed 30/09/15	Thu 01/10/15	67,66	DevTeam	
69		☛	Create Architectural design	3 days	Fri 02/10/15	Tue 06/10/15	68	ManhLN	
70		☛	Review architectural design	1 day	Wed 07/10/15	Wed 07/10/15	69	TEAM	
71		☛	Update Architectural Design	1 day?	Thu 08/10/15	Thu 08/10/15	70	ManhLN	
72		☛	Complete Architectural Design version 1.0	0 days	Wed 07/10/15	Wed 07/10/15	70		
73		☛	☛ Screen design	9 days?	Tue 29/09/15	Fri 09/10/15	55		
74	!	☛	Create screen flow document	2 days	Tue 29/09/15	Wed 30/09/15		DesignTeam	
75		☛	Create client screen design	5 days	Thu 01/10/15	Wed 07/10/15	74	AnhDD	
76		☛	Create admin screen design	5 days	Thu 01/10/15	Wed 07/10/15	74	MaiCTP	
77		☛	Review screen design	1 day?	Thu 08/10/15	Thu 08/10/15	76,75	ChinhVC	
78		☛	Update screen design	1 day?	Fri 09/10/15	Fri 09/10/15	77	DesignTeam	
79		☛	Complete screen design	0 days	Fri 09/10/15	Fri 09/10/15	78		
80		☛	☛ Database design	4.5 days?	Tue 29/09/15	Mon 05/10/15	55		
81		☛	Create database design	3 days	Tue 29/09/15	Thu 01/10/15		TrungVN	
82		☛	Review database design	0.5 days?	Fri 02/10/15	Fri 02/10/15	81	TEAM	
83		☛	Update database design	1 day	Fri 02/10/15	Mon 05/10/15	82	TrungVN	
84		☛	Complete database design	0 days	Mon 05/10/15	Mon 05/10/15	83		



85		↳	↳ Class/Component Design	6 days?	Fri 09/10/15	Fri 16/10/15	65,55	
86		↳	Define Class and Component	1 day	Fri 09/10/15	Fri 09/10/15		HuyNM
87		↳	Create Class/component design	3 days	Mon 12/10/15	Wed 14/10/15	86	DevTeam
88		↳	Review Class/component design	1 day?	Thu 15/10/15	Thu 15/10/15	87	ManhLN
89		↳	Update Class/component design	1 day?	Fri 16/10/15	Fri 16/10/15	88	DevTeam
90		↳	Complete Class/component design	0 days	Fri 16/10/15	Fri 16/10/15	89	
91		↳	Create progress report 2	1 day	Mon 19/10/15	Mon 19/10/15	85	MaiCTP
92		↳	Submit progress report 2	0 days	Mon 19/10/15	Mon 19/10/15	91	ManhLN
93		↳	↳ Testing	6 days?	Tue 20/10/15	Tue 27/10/15	92	
94		↳	Create test plan/case checklists	1 day	Tue 20/10/15	Tue 20/10/15		MaiCTP
95		↳	Create test plan	3 days	Wed 21/10/15	Fri 23/10/15	94	TestTEAM
96		↳	Review test plan	1 day	Mon 26/10/15	Mon 26/10/15	95	MaiCTP
97		↳	Update test plan	1 day?	Tue 27/10/15	Tue 27/10/15	96	TestTEAM
98		↳	Test plan complete	0 days	Tue 27/10/15	Tue 27/10/15	97	
99		↳	Create progress report 3	1 day?	Wed 28/10/15	Wed 28/10/15	93	MaiCTP
100		↳	Submit progress report 3	0 days	Wed 28/10/15	Wed 28/10/15	99	ManhLN
101		↳	↳ Construction	31 days?	Thu 29/10/15	Thu 10/12/15	46	
102		↳	↳ Define Coding Standards	1 day?	Thu 29/10/15	Thu 29/10/15		
103		↳	Create Coding Convention	1 day?	Thu 29/10/15	Thu 29/10/15		ManhLN
104		↳	Create Code & UT Test review checklist	1 day?	Thu 29/10/15	Thu 29/10/15		MaiCTP
105		↳	↳ Screen Design	2 days	Thu 29/10/15	Fri 30/10/15		
106		↳	Refine screen design document and prototype	2 days	Thu 29/10/15	Fri 30/10/15		TEAM
107		↳	↳ Coding & Unit test	18 days	Fri 30/10/15	Tue 24/11/15		
108		↳	Structure core modules & source code common	3 days	Fri 30/10/15	Tue 03/11/15	103	DevTeam
109		↳	↳ Client User Module	10 days	Wed 04/11/15	Tue 17/11/15	108,105,102	
110		↳	Home Page Module	3 days	Wed 04/11/15	Fri 06/11/15		ManhLN
111		↳	Login/ Logout Module	2 days	Mon 09/11/15	Tue 10/11/15	110	ManhLN



112			<b>Project module</b>	<b>8 days</b>	<b>Wed 04/11/15</b>	<b>Fri 13/11/15</b>			
113			Create/manage a project Module	5 days	Wed 04/11/15	Tue 10/11/15			HuyNM
114			View/Back a project Module	3 days	Wed 11/11/15	Fri 13/11/15	113		HuyNM
115			User Personal Page Module	3 days	Wed 04/11/15	Fri 06/11/15			TrungVN
116			Search Module	2 days	Mon 16/11/15	Tue 17/11/15	112		HuyNM
117			Message Module	4 days	Mon 09/11/15	Thu 12/11/15	115		TrungVN
118			<b>Admin Module</b>	<b>10 days</b>	<b>Wed 11/11/15</b>	<b>Tue 24/11/15</b>	<b>108,105,102</b>		
119			Login/ Logout module	2 days	Wed 11/11/15	Thu 12/11/15	111		ManhLN
120			Manage Users Module	4 days	Fri 13/11/15	Wed 18/11/15	117		TrungVN
121			Manage Projects Module	5 days	Wed 18/11/15	Tue 24/11/15	116		HuyNM
122			Manage Backings Module	3 days	Fri 13/11/15	Tue 17/11/15	119		ManhLN
123			Manage Categories Module	3 days	Wed 18/11/15	Fri 20/11/15	122		ManhLN
124			Mangage Silder Module	3 days	Thu 19/11/15	Mon 23/11/15	120		TrungVN
125			<i>Complete Coding</i>	0 days	Tue 24/11/15	Tue 24/11/15	109,118		
126			Create Progress Report 4	1 day?	Wed 25/11/15	Wed 25/11/15	105,107		ManhLN
127			Submit Progress Report 4	0 days	Wed 25/11/15	Wed 25/11/15	126		ManhLN
128			<b>Testing</b>	<b>26 days?</b>	<b>Fri 30/10/15</b>	<b>Fri 04/12/15</b>			
129	i		<b>Writing test case (continue)</b>	<b>10 days?</b>	<b>Fri 30/10/15</b>	<b>Thu 12/11/15</b>	<b>104</b>		<b>MaiCTP</b>
130			<b>System test case</b>	<b>5 days?</b>	<b>Fri 30/10/15</b>	<b>Thu 05/11/15</b>			
131			Create system test case	3 days	Fri 30/10/15	Tue 03/11/15			TestTEAM
132	i		Review system test case	1 day?	Wed 04/11/15	Wed 04/11/15	131		MaiCTP
133			Update system test case	1 day?	Thu 05/11/15	Thu 05/11/15	132		TestTEAM
134			<b>Integration test case</b>	<b>5 days?</b>	<b>Fri 06/11/15</b>	<b>Thu 12/11/15</b>	<b>130</b>		
135			Create integration test case	3 days	Fri 06/11/15	Tue 10/11/15			TestTEAM
136	i		Review integration test case	1 day?	Wed 11/11/15	Wed 11/11/15	135		MaiCTP
137			Update integration test case	1 day?	Thu 12/11/15	Thu 12/11/15	136		TestTEAM
138			<i>Writing test case complete</i>	0 days	Thu 12/11/15	Thu 12/11/15	134		
139			<b>Execute Test Round 1</b>	<b>8 days</b>	<b>Fri 13/11/15</b>	<b>Tue 24/11/15</b>	<b>129</b>		
140			Execute Integration Test	3 days	Fri 13/11/15	Tue 17/11/15			TestTEAM
141			Fix bug Integration Test	2 days	Wed 18/11/15	Thu 19/11/15	140		DevTeam
142			Execute System Test	3 days	Wed 18/11/15	Fri 20/11/15	140		TestTEAM
143			Fix bug System Test	2 days	Mon 23/11/15	Tue 24/11/15	142		DevTeam
144			Update Test Case	1 day	Mon 23/11/15	Mon 23/11/15	142		TestTEAM



145		Execute Test Round 2	8 days	Tue 24/11/15	Thu 03/12/15	144		
146		Execute Integration Test	3 days	Tue 24/11/15	Thu 26/11/15		TestTEAM	
147		Fix bug Integration Test	2 days	Fri 27/11/15	Mon 30/11/15	146	DevTeam	
148		Execute System Test	3 days	Fri 27/11/15	Tue 01/12/15	146	TestTEAM	
149		Fix bug System Test	2 days	Wed 02/12/15	Thu 03/12/15	148	DevTeam	
150		Create Test Report	1 day	Fri 04/12/15	Fri 04/12/15	139,145	MaiCTP	
151		<i>Testing Complete</i>	0 days	Fri 04/12/15	Fri 04/12/15	150		
152		Quality Control	3 days	Mon 07/12/15	Wed 09/12/15	128,107		
153		Create Quality Report	2 days	Mon 07/12/15	Tue 08/12/15		MaiCTP	
154		Review Quality Report	0.5 days	Wed 09/12/15	Wed 09/12/15	153	ManhLN	
155		Update Quality Report	0.5 days	Wed 09/12/15	Wed 09/12/15	154	MaiCTP	
156		Create Progress Report 5	1 day?	Thu 10/12/15	Thu 10/12/15	152	MaiCTP	
157		Submit Progress Report 5	0 days	Thu 10/12/15	Thu 10/12/15	156	ManhLN	
158		Termination	11 days	Fri 11/12/15	Fri 25/12/15	101		
159		Create lesson learned	1 day	Fri 11/12/15	Fri 11/12/15		TEAM	
160		Final Report Document	2 days	Mon 14/12/15	Tue 15/12/15	159		
161		Create Final Report document	2 days	Mon 14/12/15	Tue 15/12/15		ManhLN	
162		Create Installing Guide	2 days	Mon 14/12/15	Tue 15/12/15		HuyNM,TrungVN	
163		Create User Guideline	2 days	Mon 14/12/15	Tue 15/12/15		ChinhVC,AnhDD	
164		Create Project Result Assessment	2 days	Mon 14/12/15	Tue 15/12/15		MaiCTP	
165		Create Slides for Final Presentation	2 days	Mon 14/12/15	Tue 15/12/15		TEAM	
166		<i>Submit all Final Documents and Source codes</i>	0 days	Tue 15/12/15	Tue 15/12/15	160	ManhLN	
167		Pre-closing	5 days	Wed 16/12/15	Tue 22/12/15			
168		Trainning presentation	5 days	Wed 16/12/15	Tue 22/12/15	160	TEAM	
169		Closing	3 days	Wed 23/12/15	Fri 25/12/15	168		
170		Presentation for Capstone Project	3 days	Wed 23/12/15	Fri 25/12/15		TEAM	
171		<i>Complete Project</i>	0 days	Fri 25/12/15	Fri 25/12/15	170		

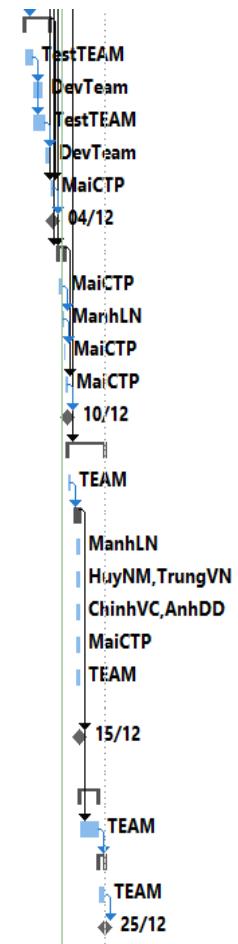


Figure 2-2: Project Schedule

#### **2.4.4 Resource**

Specified as in the section [2.5.2 Project Team](#)

#### **2.4.5 Infrastructure**

Item	Description	Expected Availability by	Note
<b>Development Environment</b>			
Operating System	Window 8.1 (32 bit, 64 bit)		
Browser	Chrome (40 or above), Firefox (30 or above)		
Development language	.NET C#		
<b>Technology</b>			
Development language	.NET C#, MVC Model, AngularJS		
Database	SQL Server 2012		
<b>Hardware Requirement</b>			
Hardware Configuration	2GB workspaces on server		
<b>Equipment &amp; Tools</b>			
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	

**Table 2-19: Infrastructure**

#### 2.4.6 Training Plan

Training Area	Participants	Duration	Waiver Criteria
<b>Technical</b>			
.NET MVC5	HuyNM, TrungCV	1 week	Mandatory
AngularJS, jQuery	HuyNM, TrungCV	1 week	Mandatory
Bootstrap	ChinhVC, AnhDD, MaiCTP	1 week	
<b>Process</b>			
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

**Table 2-20: Training Plan**

#### 2.4.7 Finance

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

## 2.5 Project Organization

### 2.5.1 Organization Structure

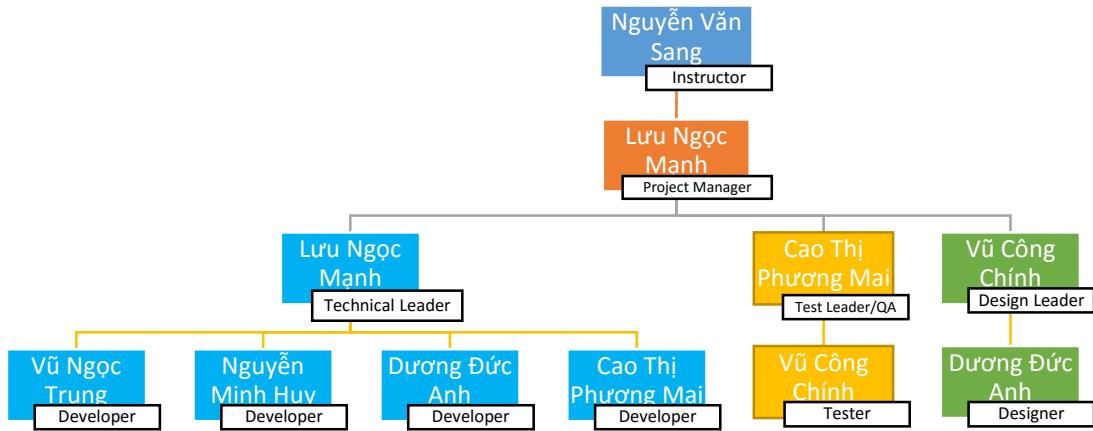


Figure 2-3: Organization Structure

### 2.5.2 Project Team

Role	Responsibility	Full name	Effort (%)	Start date	End date
PM	Have overall responsibility of the project: <ul style="list-style-type: none"><li>- Project planning and scheduling</li><li>- Task assignment and tracking processing</li><li>- Review documents</li><li>- Reporting to supervisor</li></ul>	Luu Ngoc Manh	50	07-09-2015	25-12-2015
PTL	PTL is responsible for the technical project execution	Luu Ngoc Manh	50	07-09-2015	25-12-2015
Programmer #1	<ul style="list-style-type: none"><li>- Study technique (C#, MVC, AngularJS, etc.)</li><li>- Coding functions and modules of system.</li><li>- Peer-review source code of others members</li></ul>	Nguyen Minh Huy, Vu Ngoc Trung,	100	07-09-2015	25-12-2015
Programmer #2	<ul style="list-style-type: none"><li>- Support coding functions and modules of system.</li></ul>	Duong Duc Anh	50	07-09-2015	25-12-2015

Programmer #3	- Support coding functions and modules of system.	Cao Thi Phuong Mai	100	07-09-2015	30-10-2015
Test Leader	- Create test plan, test case, test report, quality report - Execute test.	Cao Thi Phuong Mai	100	30-10-2015	25-12-2015
Tester	- Support creating test plan, test case, test report, quality report Execute test.	Vu Cong Chinh	100	30-10-2015	25-12-2015
Design Lead	- Create screen design, prototype - Review design of others member	Vu Cong Chinh	100	07-09-2015	25-12-2015
Designer #1	- Support creating screen design	Duong Duc Anh	50	07-09-2015	25-12-2015

**Table 2-21: Project Team description**

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

Role	Name	W 2 -	W 3 -	W 4 -	W 1 -	W 2 -	W 3 -	W 4 -	W 1- N ov	W 2- N ov	W 3- N ov	W 4- D ec	W 1- D ec	W 2- D ec	W 3- D ec	Total (pd)
		S e e p	S e e p	S e e p	O c t	O c t	O c t	O c t	N ov	N ov	N ov	D ec	D ec	D ec		
PM/ PTL	Luu Ngoc Manh	1 0 0 %	10 0%	80												
Dev	Nguyen Minh Huy	1 0 0 %	10 0%	80												
Dev	Vu Ngoc Trung	1 0 0 %	10 0%	80												
Test Lead er/De v	Cao Thi Phuong Mai	1 0 0 %	10 0%	80												
Teste r/Des igner /Dev	Vu Cong Chinh	1 0 0 %	10 0%	80												
Teste r/Des igner /Dev	Duong Duc Anh	1 0 0 %	10 0%	80												
<b>Total</b>																<b>480</b>

**Table 2-22: Human Resource Budget Allocation**

## 2.6 Communication & Reporting

Communication Type	Method/Tool	When	Information	Participants/Responsible
<b>Project Task Tracking</b>				
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
<b>Project Meeting</b>				
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
<b>Supervisor Communication and Reporting:</b>				

<b>Communication with Supervisor</b>				
Transfer/Sharing of project documentation/info rmation	TortoiseGit	When available	All project documentation and information	PM, Project Team Members, QA
Project Report	Agreed FPT Software and FU standard format	5pm Monday, Weekly	Project status report, Issue requiring clarifications, escalation, if any	PM
Project Meetings with supervisor	Face to face	12h30 Wednesday, Weekly	As above	PM
Requirement gathering/clarificati on	Face to face meeting	During requirement analysis phase	As in Q&A list	PM
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/Divisi on 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

**Table 2-23: Communication and Reporting Plan**

### 3 SOFTWARE REQUIREMENT

#### 3.1 Introduction

##### 3.1.1 Purpose

In this chapter, we provide the describing of general requirement and non-functional requirements. These requirements will assure that the system will correctly and reliably perform its intended functionality. This specification will provide general, as well as specific requirements to be used in the design, testing and validation of the system. It is intended for both stakeholders and the project team of the system.

##### 3.1.2 Definitions and Acronyms

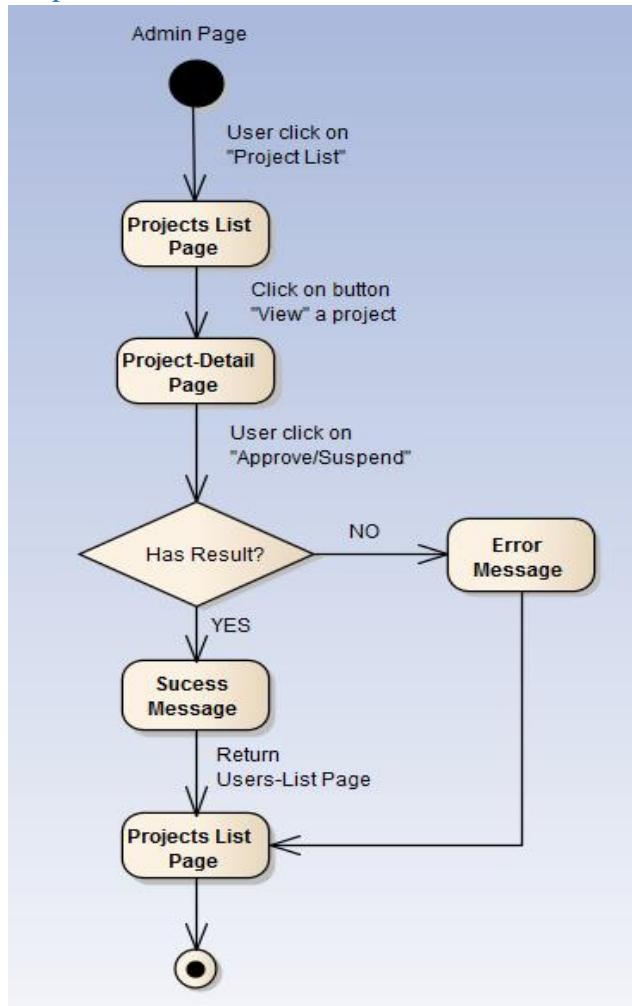
Acronym	Definition	Note
DDL	Dandelion system	
SRS	Software Requirements Specifications	
Guest	DDL's user who did not register	
Registered	DDL's user who registered	
Admin	DDL's administrator	
User	DDL's user	

**Table 3-1: Definition and Acronyms**

## 3.2 User Requirement Specification

### 3.2.1 Business Process Overview

#### 3.2.1.1 Approve/Suspend workflow

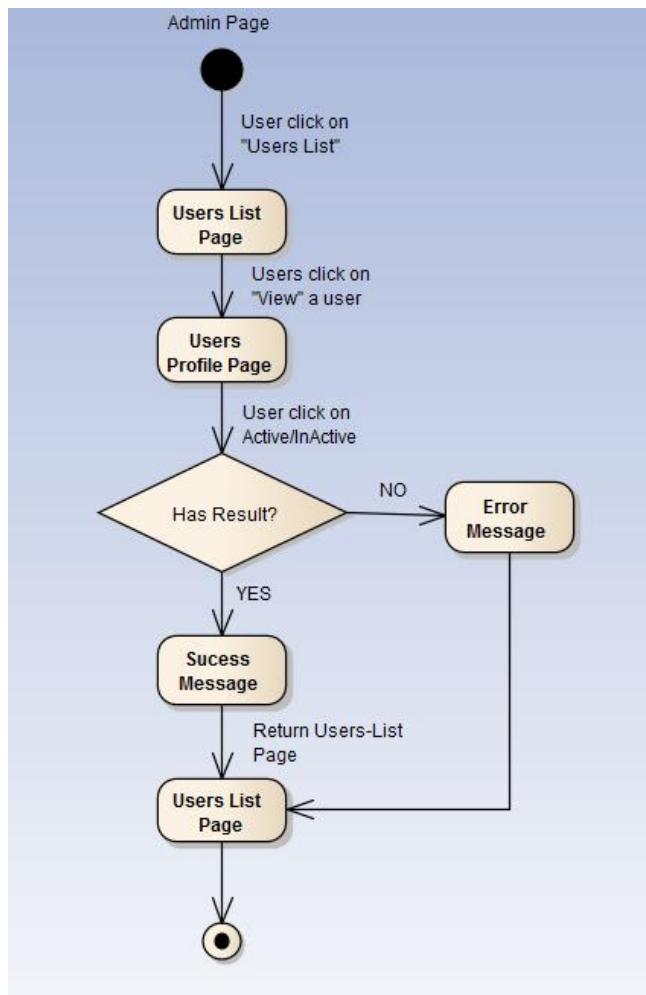


**Figure 3-1: Approve/Suspend workflow**

Brief description: Admin can use this function. Steps:

- Step 1: User click on projects list.
- Step 2: System return projects list page.
- Step 3: User click on button view
- Step 4: System return project detail page.
- Step 5: User click on button Approved/Suspended
- Step 6: System return success message or error message.
- Step 7: System return projects list page.

### 3.2.1.2 Active/Inactive workflow

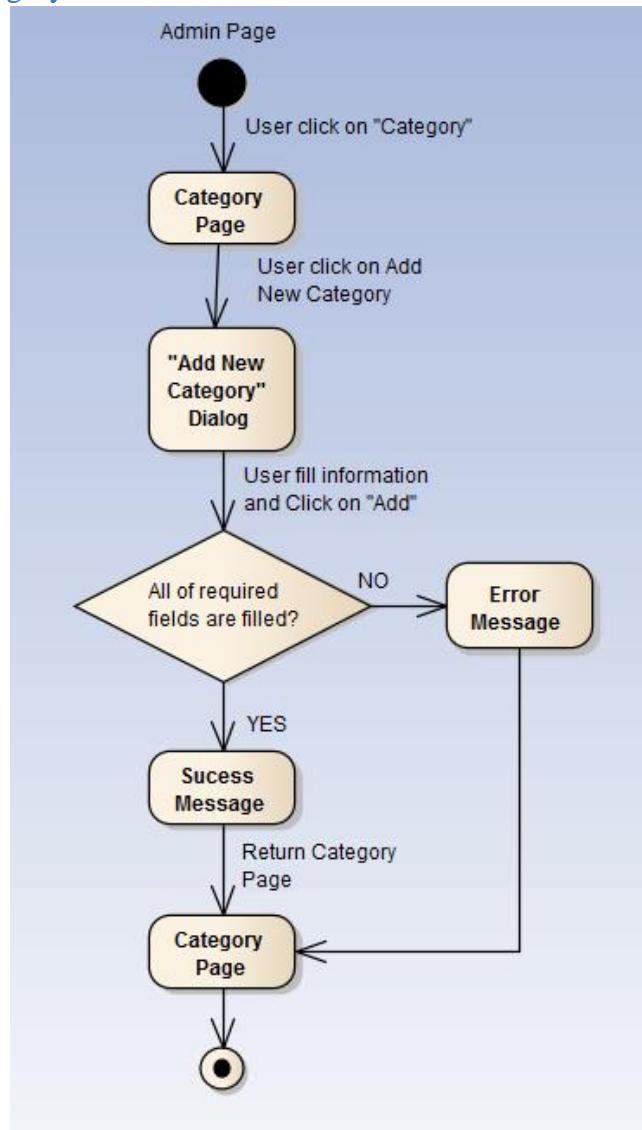


**Figure 3-2: Active/Inactive workflow**

Brief description: Admin can use this function. Steps:

- Step 1: User click on users list.
- Step 2: System return users list page.
- Step 3: User click on button view a user
- Step 4: System return user profile page.
- Step 5: User click on button Active/Inactive
- Step 6: System return success message or error message.
- Step 7: System return users list page.

### 3.2.1.3 Add category workflow

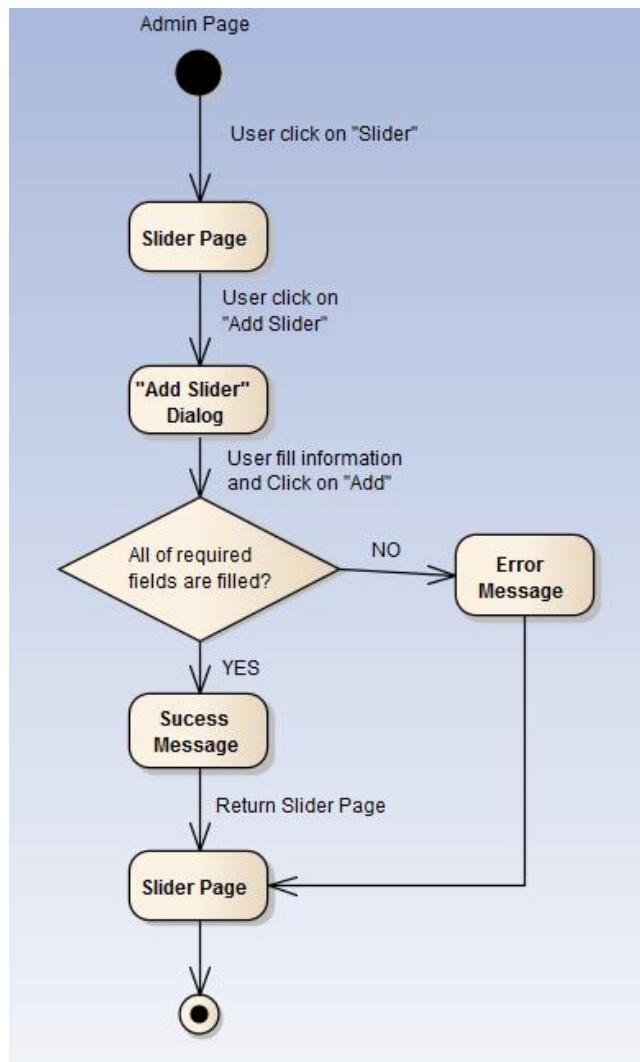


**Figure 3-3: Add category workflow**

Brief description: Admin can use this function. Steps:

- Step 1: User click on Category.
- Step 2: System return category page.
- Step 3: User click on button new category
- Step 4: System return dialog “Add New Category”.
- Step 5: User fill information and click “Add”
- Step 6: System return success message or error message.
- Step 7: System return category page.

### 3.2.1.4 Add slider workflow

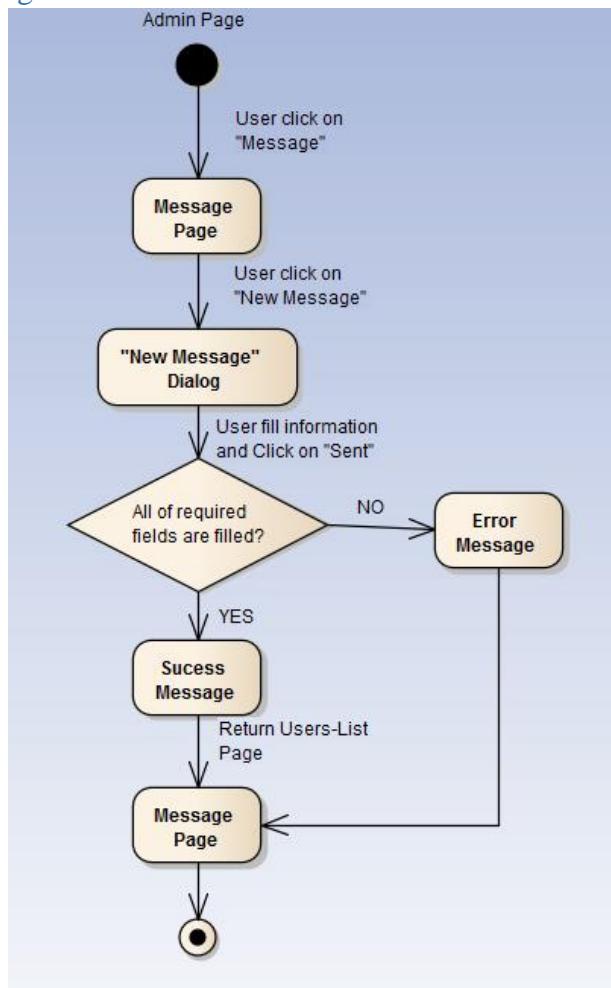


**Figure 3-4: Add slider workflow**

Brief description: Admin can use this function. Steps:

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

### 3.2.1.5 Sent message workflow

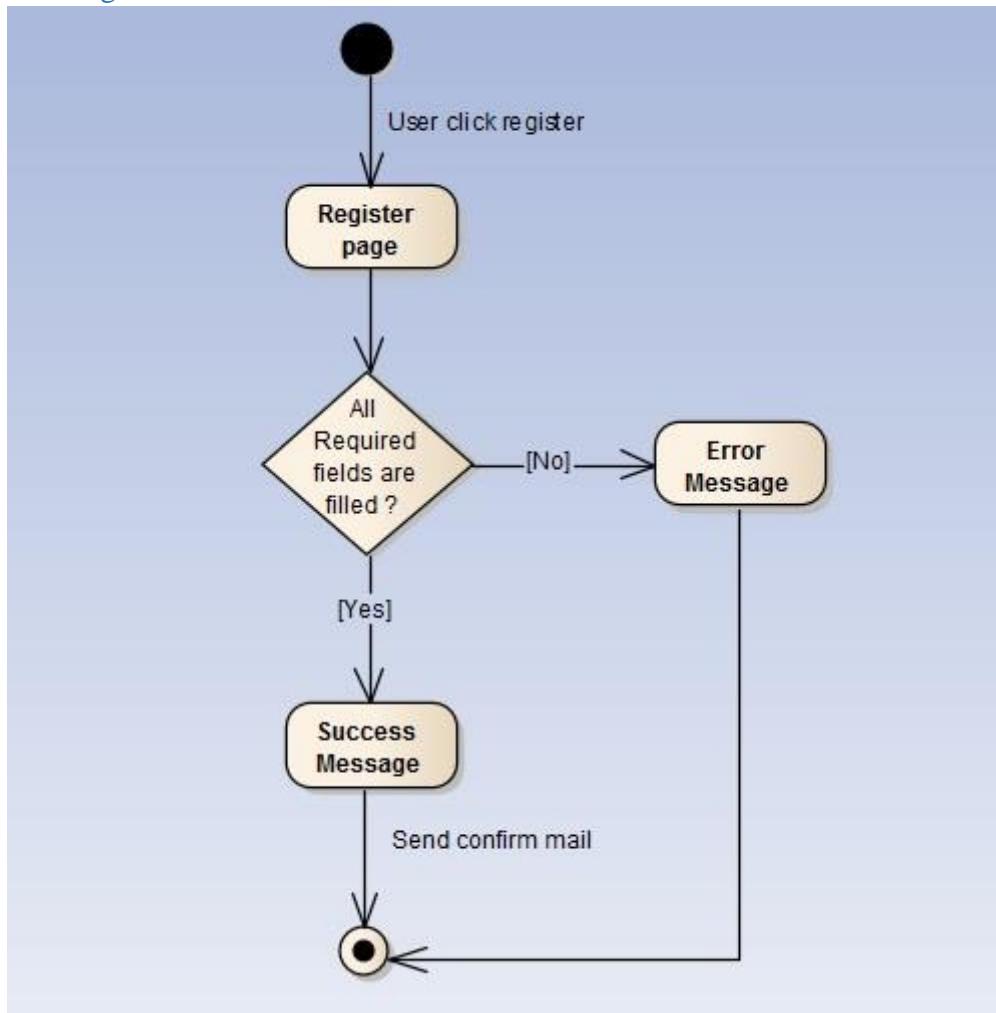


**Figure 3-5: Sent message workflow**

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” button.
- Step 4: System return dialog “New Message”.
- Step 5: User fill information and click “Sent”
- Step 6: System return success message or error message.
- Step 7: System return message page.

### 3.2.1.6 Register workflow

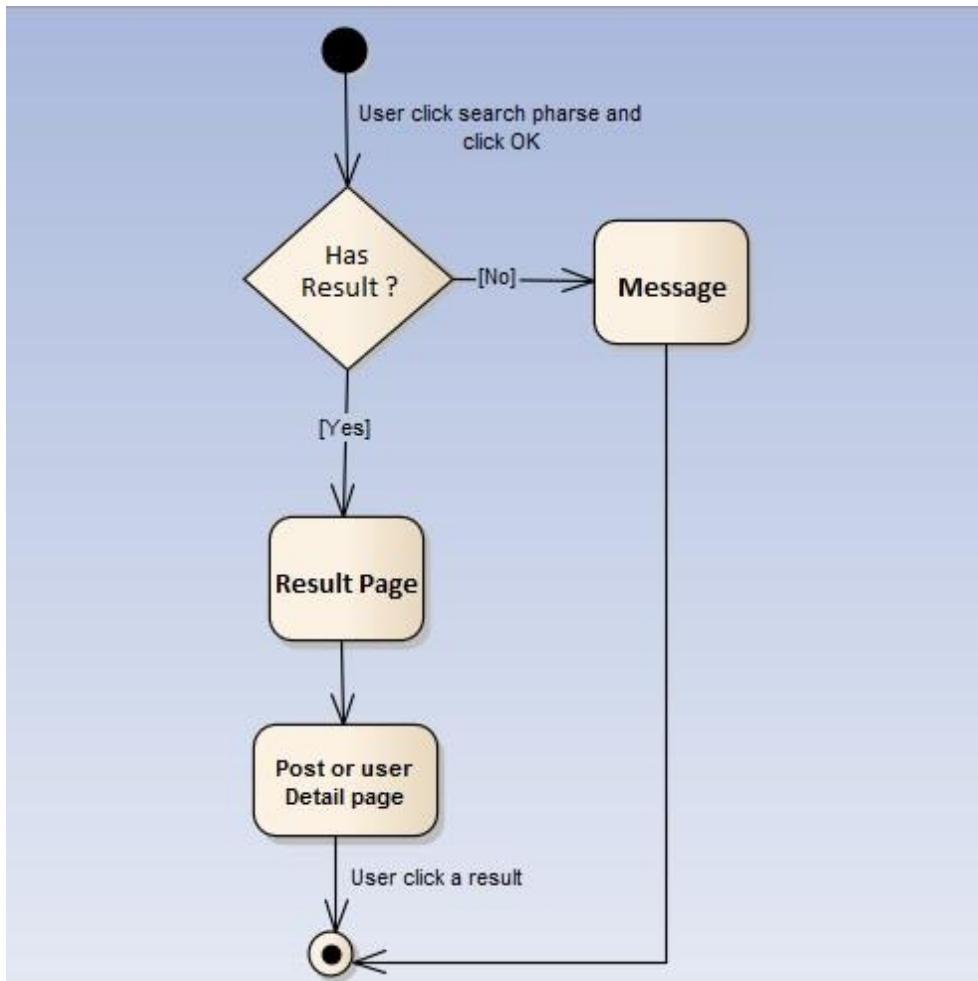


**Figure 3-6: Register workflow**

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.

### 3.2.1.7 Search workflow

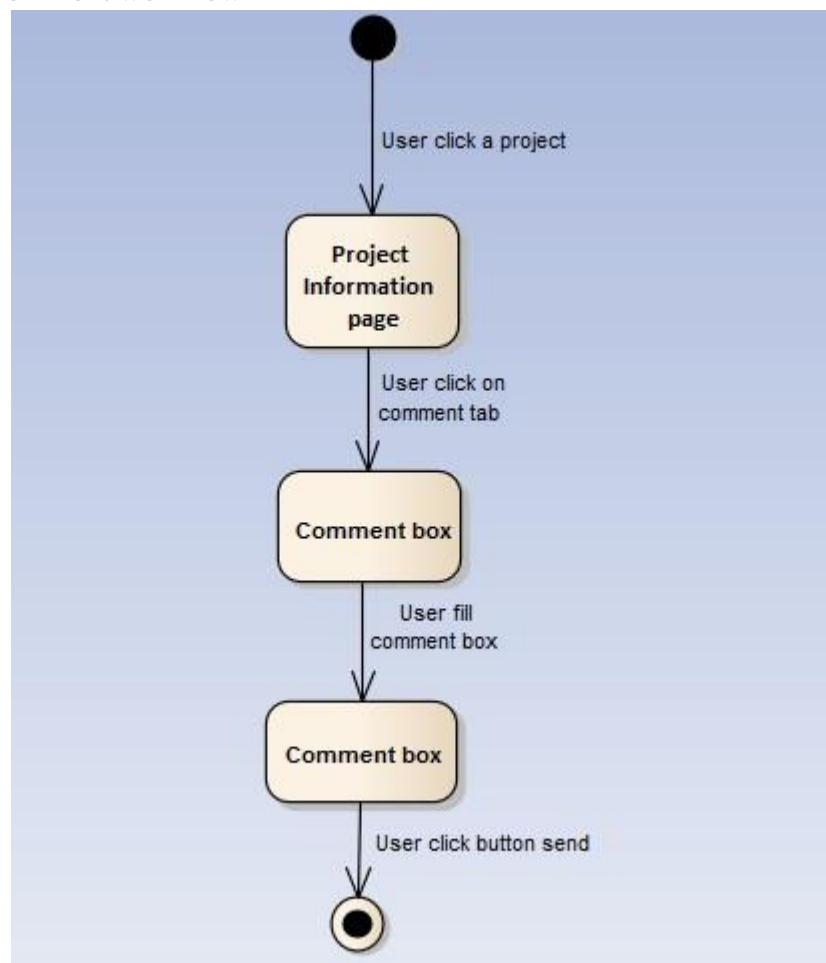


**Figure 3-7: Search workflow**

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 nothing found.
- Step 3: User can click returned results to view.

### 3.2.1.8 Comment workflow

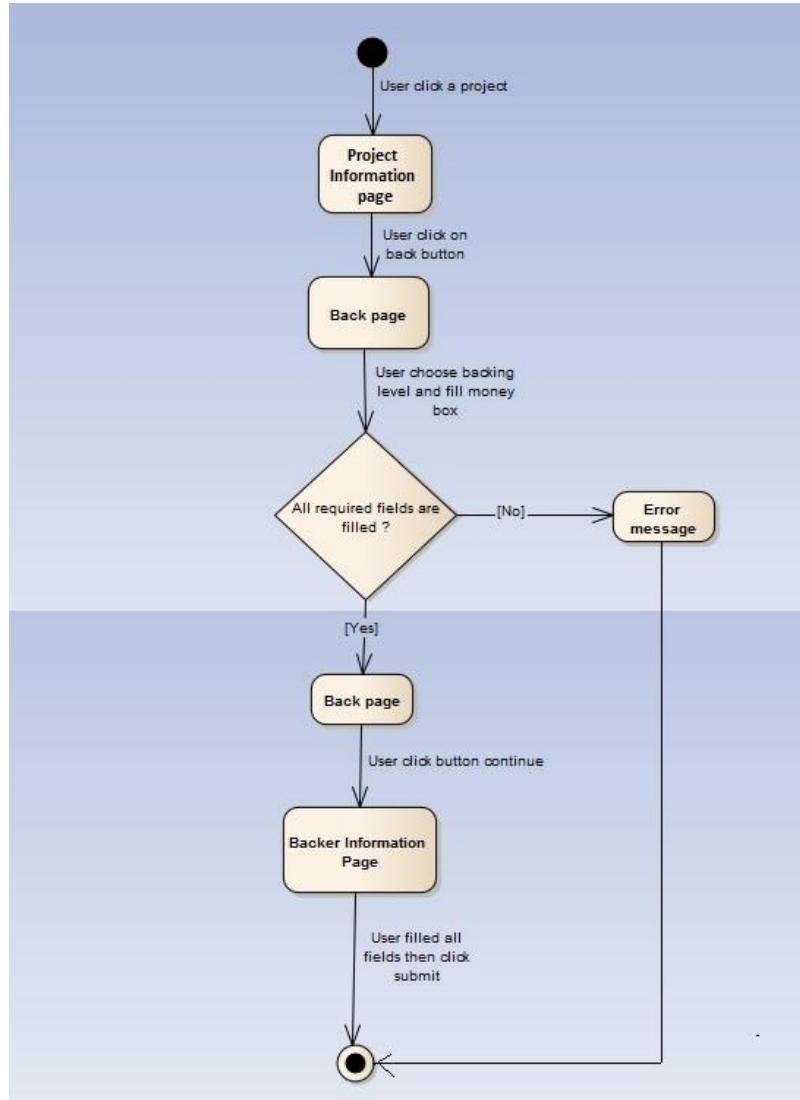


**Figure 3-8: Comment workflow**

Brief description: Member can use this function. Steps:

- Step 1: User click on a project.
- Step 2: System return Project information page.
- Step 3: User click on comment tab.
- Step 4: System show comment tab.
- Step 5: User fill comment box.
- Step 6: User click sent.

### 3.2.1.9 Back a project workflow

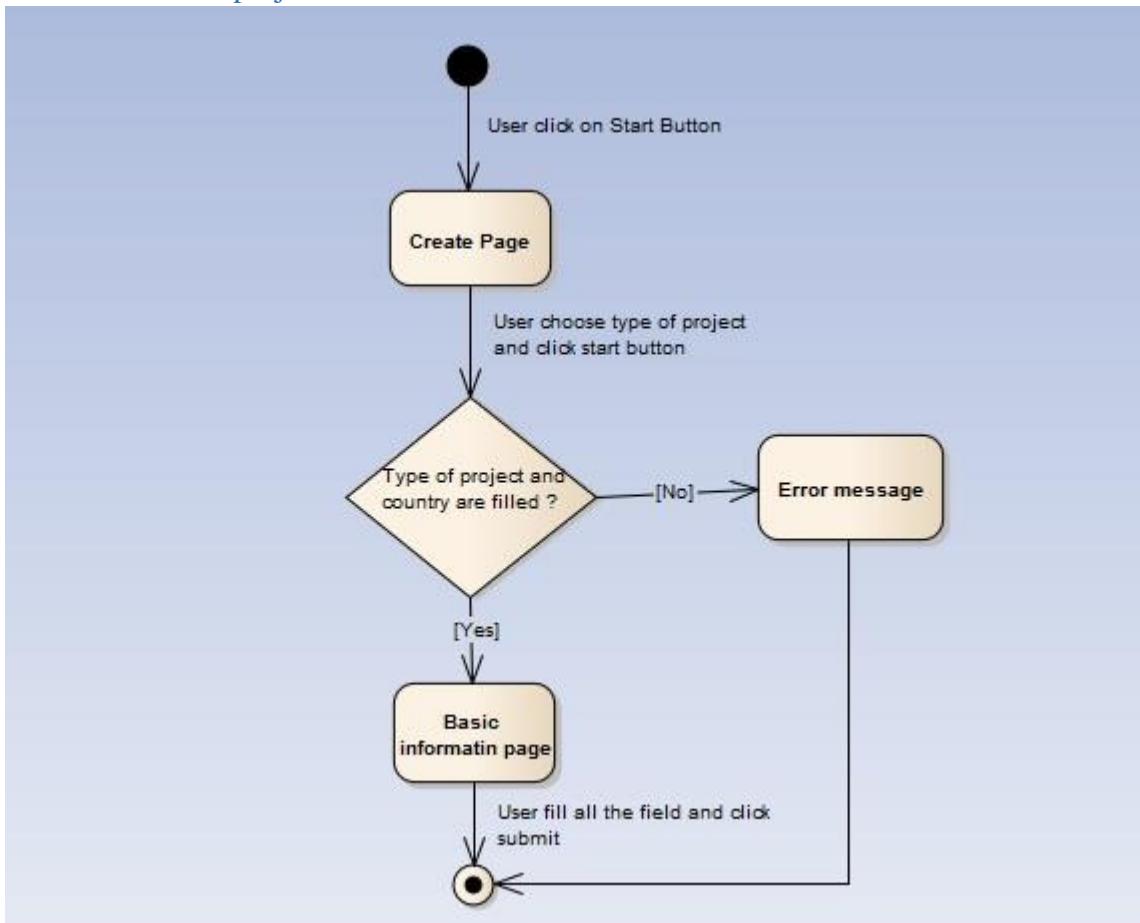


**Figure 3-9: Back a project workflow**

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

- Step 1: User click on a project.
- Step 2: System return Project information page.
- Step 3: User click on button back.
- Step 4: System return Back page.
- Step 5: User choose back level then click continue.
- Step 6: System return Backer information page.
- Step 7: User fill all fields then click submit.

### 3.2.1.10 Create a project workflow

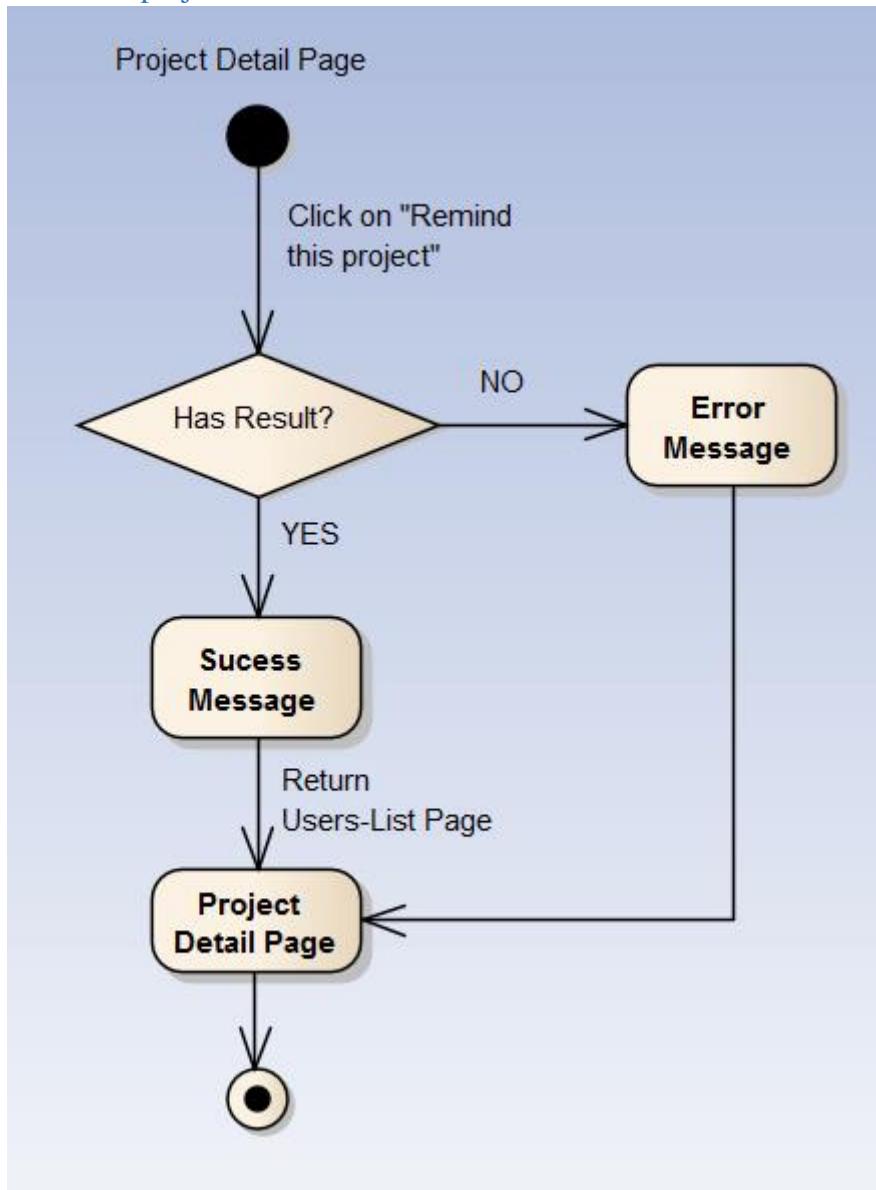


**Figure 3-10:** Create a project workflow

Brief description: Member can use this function. Steps:

- Step 1: User click on start button.
- Step 2: System return Create page.
- Step 3: User choose type and country then click start.
- Step 4: System return Basic page.
- Step 5: User fill all the fields then click submit.
- Step 6: System return Backer information page.

### 3.2.1.11 Remind a project workflow

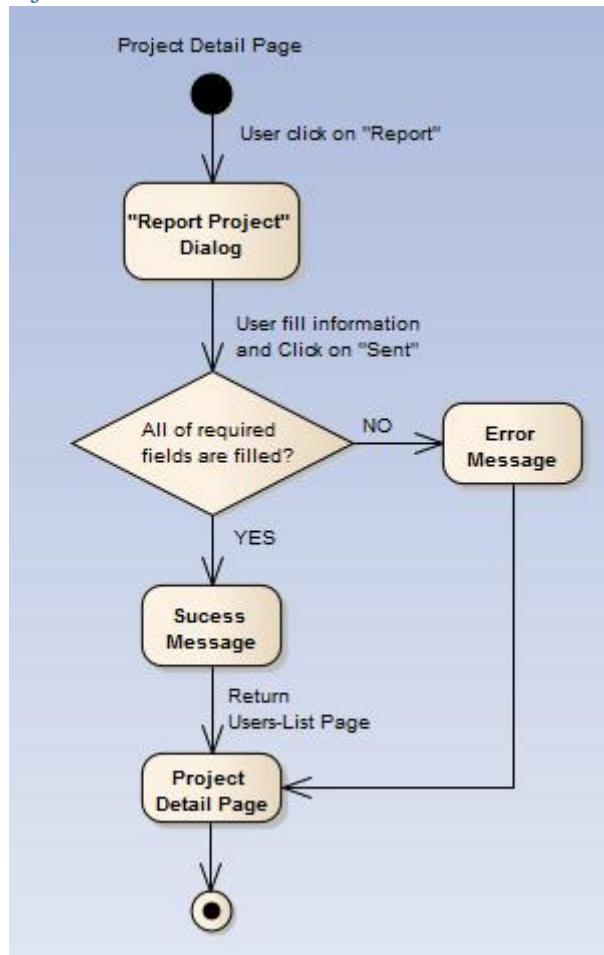


**Figure 3-11:** Remind a project workflow

Brief description: Member can use this function. Steps:

- Step 1: User click on “Remind this project” button.
- Step 2: System return success message or error message.
- Step 3: System return Project Detail page.

### 3.2.1.12 Report a project workflow



**Figure 3-12: Report a project workflow**

Brief description: Member can use this function. Steps:

- Step 1: User click on report button.
- Step 2: System return “Report project” dialog.
- Step 3: User fill information and click “Sent”
- Step 4: System return success message or error message.
- Step 5: System return project detail page.

## 3.2.2 Product Features

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

### 3.2.2.1 Client features

- **Register:** User can register an account and login to use all features of DDL.
- **Login/Logout:** Users login/logout an account to use or exit system DDL.
- **Search:** User and guest can search with code of project, name of project, name of creator; and system will display all of project relate to keywords.

- **Filter:** User can filter with category (Art, Comics, Game, etc.), place (Earth, Vietnam, etc.), Status (Magic, Popularity, Most Funded, etc.). And system will display projects relate to these selected options.
- **Project**
  - Create New Project: Users can create new projects with basic info, reward, timeline, clip and some pictures.
  - Update Project Information: User can change and update information of projects which created.
  - Comment/Remind: Users can comment/remind a project to follow and discuss with another users.
  - Report: User can report a project which violate regulation to administrator.
- **Back Project:** User can back another project to donate money to help project success and get rewards.
- **Profile:**
  - Change password: User can change password to keep security.
  - Update profile: User can change or update information.
  - Manage created projects, backed projects, reminded projects: User can follow process of projects, follow project backed, project reminded. View statistical projects had created, backed, reminded.
  - Sent/Receive message: Users can send to or receive from other users to discuss about projects.

### 3.2.2.2 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 project:**
  - Search project :
  - Search with project name: Admin can search with name of project and system will display simple information about that project.
  - Search with category: Admin can search with category of project and system will display all project with that category
  - Search with Creator name: Admin can search with Creator name of project and system will display all project of that user.
  - Edit project status: Admin can set status of a project: pending to approved, pending to suspend, approved to suspend, etc.
- **Manage Backing:** Admin can view users who backed a project. Also, Admin can view simple information about that user and which reward they chose.
- **Manage Categories:**
  - Add new category: Admin can add new category.
  - inactive/active category: Admin can set category's status to inactive or active
- **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/send messages.
  - Compose new message: Admin can compose new message to send to user.
  - Delete message: Admin can delete received messages/sent messages.

### 3.2.3 User characteristic

This website is a major platform for crowdfunding. Anybody could launch a project on Dandelion as long as it follows our rules. We believe that creative projects making a better world, and we're thrilled to help support new ones. Building a community of backers around an idea is an amazing way to make something new. The website requires users to have the ability to use a browser with an internet connection. 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 project, back project, etc...
- **Administrator:** They can use all functions of normal member and have an Administrator control panel to manage all configuration of system.

### 3.2.4 Functional Requirements

#### 3.2.4.1 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 projects, creator, bankers or members' name.

#### 3.2.4.2 Discover Projects

##### **Access right: Guest and member**

This function is used to view an overview of the projects. The system will display a Discover Screen for user. In this screen, have many category, favorite projects, popular projects. Users can use many options such as sort by category, place, popularity; link to Project Details Screen to view projects, and can back this project.

#### 3.2.4.3 Create Project

##### **Access right: Member**

This function is used to create new project. The system will display a Suggest Create Project Screen for user to create. In this screen, users choose category projects, input name projects and click on button create. The chosen will be passed to Create Project Screen. In this screen, it devices on many tabs such as basic, reward, story, update, etc. And users have to fill in the fields. After finish, users can click button "submit to review" and wait for project approved.

#### 3.2.4.4 Back Project

##### **Access right: Member**

This function is used to back project. The system will display a Project Details Screen for user to back project. In this screen, users press on a reward or click on button "back this project". The chosen will be passed to Back Project Screen. In this screen, have many rewards to users can choose and fill money to back. Then click on button "Continue", the chosen will be passed

to Payment Screen. In this screen, users can fill information and choose type payment and Bank to donate project (back project).

#### 3.2.4.5 Send/Receive messages

**Access right:** Member and Administrator.

This function is used to send and receive messages between users, 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.

#### 3.2.4.6 Management profile

**Access right:** Member

This function is used to management 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 view backed project, created project, starred projects, favorite bankers.

#### 3.2.4.7 Management Member's account

**Access right:** Administrator

This function is used to management 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's account temporarily, then these account cannot continue to use functions of website. Locked account will be unlocked when lock is expired. 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.

#### 3.2.4.8 Management Project

**Access right:** Administrator

This function is used to management project. The system will display a Management Project Screen. In this screen, admin can view information about project of creator, status of projects, and give decision to approve or suspend project of member.

#### 3.2.4.9 Management Dashboard Overview

**Access right:** Administrator

This function is used to management report. The system will display a Dashboard Screen to have overview all of system. In this screen, admin can view information about user and projects such as the number of accessing, new users, active users, inactive users, number of created projects, backed project, bankers, etc.

### 3.2.5 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**

### 3.3 Software Requirements Specification

#### 3.3.1 Overall Description

##### 3.3.1.1 System use case

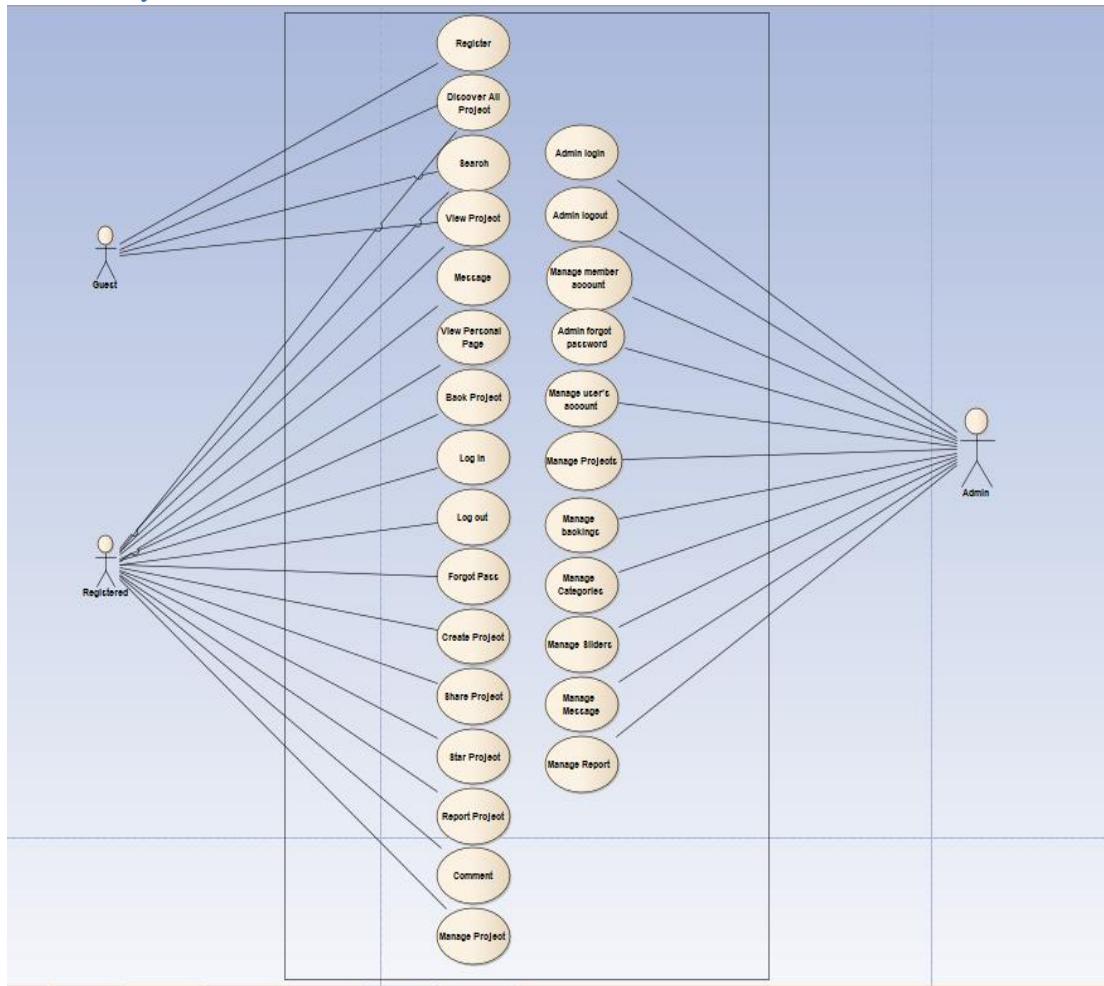


Figure 3-13: Use case diagram of DDL system

### 3.3.1.2 Product Features

UC No.	Group Of Functions	Function	Glossary
<b>Guest/Registered</b>			
UC001	Common	Discover all project	
UC002		Search project	
UC003	View project	View campaign	
UC004		View update	
UC005		View comments	
<b>Guest</b>			
UC006	Common	Register	
<b>Registered</b>			
UC007	Common	Login	
UC008		Logout	
UC009		Forgot password	
UC010		Create project	
UC011		Share project	
UC012		Star project	
UC013		Report project	
UC014	View Personal page	View backed project	
UC015		View created project	
UC016		View reminded project	
UC017		View Public profile	
UC018		Edit profile	
UC019		Change password	
UC020	Message	Search message	
UC021		Send message	
UC022		Delete message	
UC023	Comment	View list comment	
UC024		Post comment	
UC025		Edit comment	
UC026		Delete comment	
UC027	Manage Project	Edit project	
UC028		View list backing	
UC029		Add update project	
UC030		Back project	
<b>Administrator</b>			
UC031	Common	Admin login	
UC032		Admin logout	
UC033	Manage user's account	View list user	
UC034		View user profile	
UC035		Activate user's account	
UC036		Deactivate user's account	
UC037		Search user's account	
UC038		View user's dashboard	

UC039	Manage projects	View list projects	
UC040		Search project	
UC041		View(review) project	
UC042		Approve project	
UC043		Suspend project	
UC044		Export list projects	
UC045		View project's dashboard	
UC046	Manage backings	View list backing	
UC047		Search backing	
UC048		View backing's information	
UC049		Export list backings	
UC050	Manage Categories	View list categories	
UC051		Search category	
UC052		Activate category	
UC053		Deactivate category	
UC054		Add new category	
UC055		Edit category	
UC056	Manage Slides	View list slides	
UC057		Activate slide	
UC058		Deactivate slide	
UC059		Add new slide	
UC060		Edit slide	
UC061		Delete slide	
UC062	Manage Messages	View list messages	
UC063		Search message	
UC064		Send message	
UC065		View message	
UC066		Delete message	
UC067	Manage Reports	View list reports	
UC068		View report detail	
UC069		Change report's status	

**Table 3-2: List of use case diagram**

### 3.3.1.3 User Characteristics

DDL website towards to anyone wants to launch a project on Dandelion as long as it follows our rules. Project creators set a funding goal and deadline. If people like a project, they can pledge money to make it happen. Backers are supporting projects to help them come to life, not to profit financially. Instead, project creators offer rewards to thank backers for their support. DDL has three objective users:

- **Guest:** Users did not register, they can use some function like Register, Discover project, Search, etc...
- **Registered:** Users have account on DDL, they can use almost functions of DDL for user like Login, Logout, Create, Manage project, Comment, Report, etc...
- **Administrator:** They can use all functions of normal member and have an Administrator control panel to manage all information, project and configuration of DDL.

### 3.3.2 Functional Specification

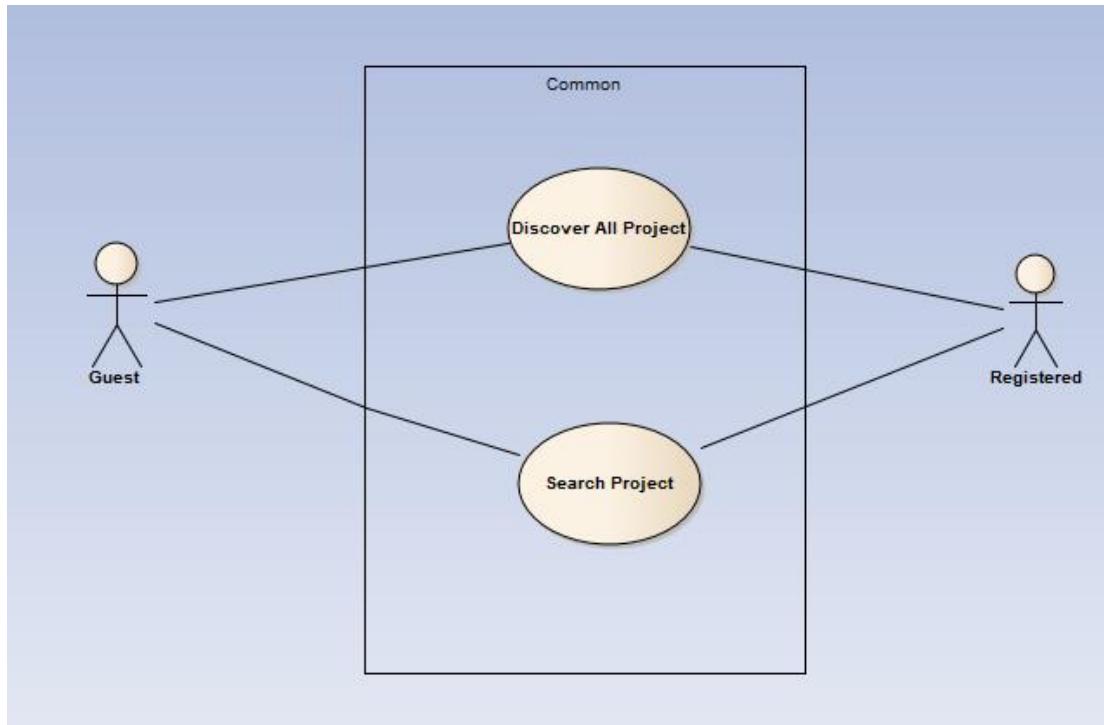
#### 3.3.2.1 Business Rules

No	Description
B1	Search textbox cannot be empty
B2	Content is limited to 1000 characters
B3	Chat box cannot be empty
B4	Subject is limited to 1000 characters
B5	Name cannot be empty
B6	If user accesses to nonexistent page, DDL displays error message “Page not found”
B7	Email must contain characters ‘@’ and ‘.’
B8	Date of birth must be chosen
B9	Displayed date is mm/dd/yyyy format
B10	Date time input format is datetime
B11	Number is rounded to 2 decimal number
B12	Unit of money is VND
B13	Displayed money is rounded to 2 decimal number
B14	Allowed file extension: .jpg, .png, .bmp
B15	File size is limited to 100 MB
B16	Guest could back project without login
B17	Message displayed like a Gmail format
B18	Rank is calculated depending on funding of the project

**Table 3-3: Business rules**

### 3.3.2.2 Guest/ Registered Group Function

#### 3.3.2.2.1 Common Module



**Figure 3-14:** Common use case

##### 3.3.2.2.1.1 UC001-Discover

<b>Use Case ID</b>	UC001	<b>Version</b>	1.1
<b>Use Case Name</b>	Discover list of all project		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Guest & Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to show all existing project stored on DDL		
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website		
<b>Trigger:</b>	User click Discover		
<b>Post conditions:</b>	List all existing project will be displayed in the Discover screen		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	User	Click Discover	
2	DDL	List all existing project	

<b>Alternative Flows:</b>		
<b>AT1</b>	At step 1 in the main flows, if user click other hyperlinks or buttons	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
1.1	DDL	Redirects user to chosen hyperlink or button
<b>Exceptions: N/A</b>		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

Table 3-4: Discover Use case

## 3.3.2.2.1.2 UC002-Search Project

<b>Use Case ID</b>	UC002	<b>Version</b>	1.1
<b>Use Case Name</b>	Search project		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Guest & Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to search existing information about project stored on DDL by name		
<b>Preconditions:</b>	1.DDL website is available 2.User browsed DDL website		
<b>Trigger:</b>	User enters search phrase and click Search or press Enter		
<b>Post conditions:</b>	List all results that matched with the keyword user enters will be displayed in the Search Details screen		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	User	Enter search phrase to text box	
2	User	Click Search or press Enter	
3	DDL	List all results that matched with the keyword user entered	
<b>Alternative Flows:</b>			

<b>AT1</b>	At step 2 in the main flows, if user click other hyperlinks or buttons	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
2.1	DDL	Redirects user to chosen hyperlink or button
<b>Exceptions:</b> N/A		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

Table 3-5: Search project Use case

### 3.3.2.2.2 View Project Module

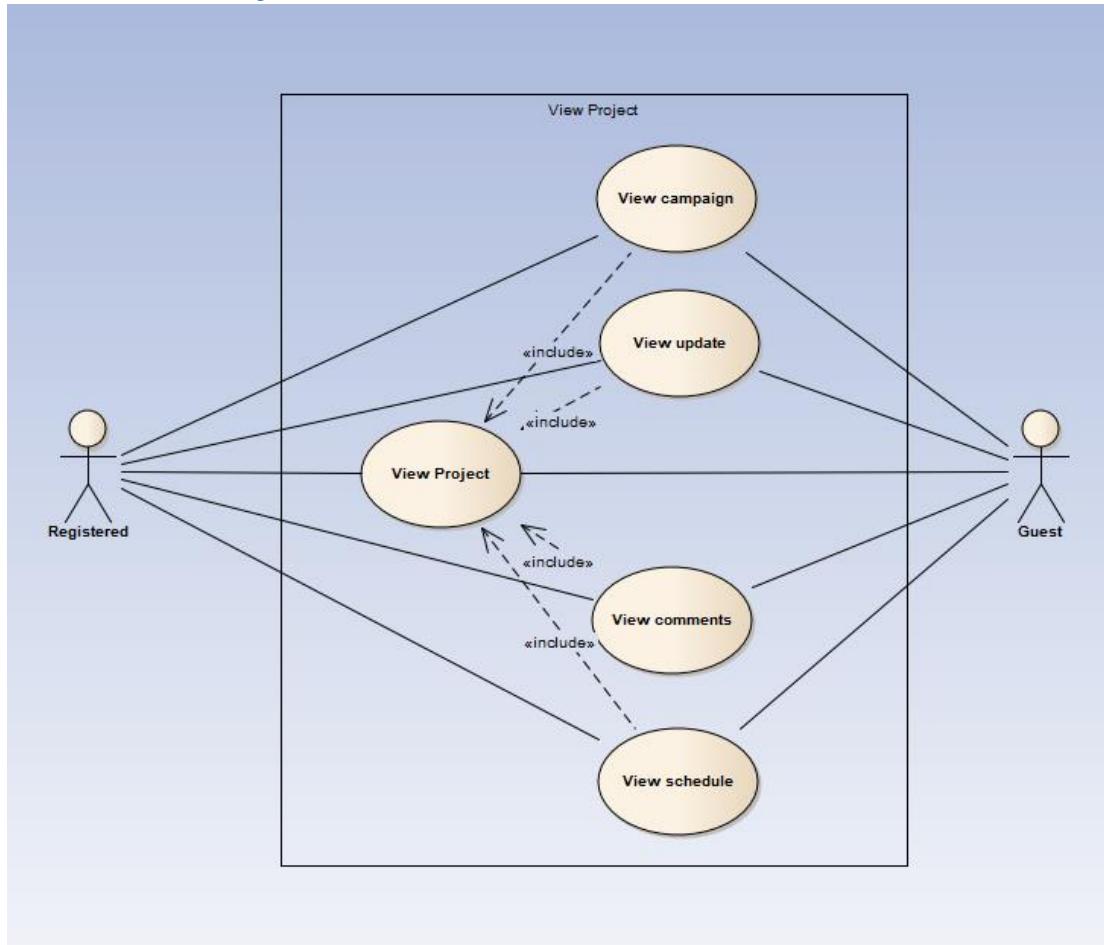


Figure 3-15: View Project Module use case

## 3.3.2.2.1 UC003-View campaign

<b>Use Case ID</b>	UC003	<b>Version</b>	1.1		
<b>Use Case Name</b>	View campaign				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user Guest	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows user to view all common existing information of the project(description, risks, Q&A, ...)				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User choose the project that user want to see				
<b>Trigger:</b>	User clicks on View Project button/ Name of the project.				
<b>Post conditions:</b>	Common information of this project is displayed				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	User	Click on View Project button or Click on name of the project			
2	DDL	Display View Project page includes 5 tabs: - Campaign - Update - Comments With Campaign tab is default.			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

Table 2-24:View campaign Use case

## 3.3.2.2.2 UC004-View update

<b>Use Case ID</b>	UC004	<b>Version</b>	1.1		
<b>Use Case Name</b>	View update				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user Guest	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows user to view all update existing information of the project				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User choose the project that user want to see				
<b>Trigger:</b>	User clicks on Update tab				
<b>Post conditions:</b>	Update information of this project is displayed				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	User	Click on View Project button or Click on name of the project			
2	DDL	Display View Project page includes 5 tabs: - Campaign - Update - Comments With Campaign tab is default.			
3	User	Click on Update tab			
4	DDL	Display Update page with all updated information.			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>		High			
<b>Frequency of Use:</b>		High			
<b>Business Rules:</b>		<a href="#">B1</a>			
<b>Other Information:</b>		N/A			

Table 3-6: View update Use case

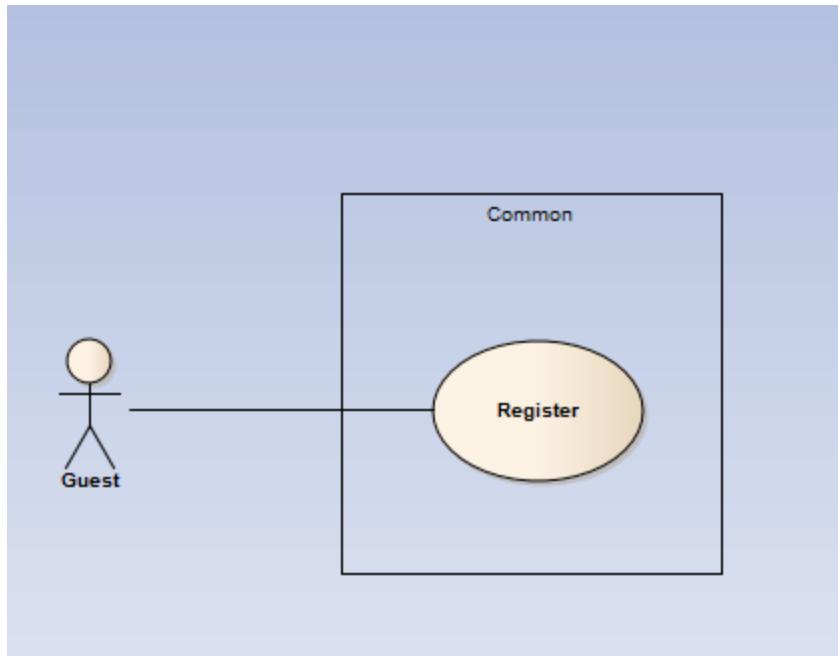
## 3.3.2.2.3 UC005-View comments

<b>Use Case ID</b>	UC005	<b>Version</b>	1.1		
<b>Use Case Name</b>	View comments				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user Guest	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows user to view all existing comment of the project				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User choose the project that user want to see				
<b>Trigger:</b>	User clicks on Comments tab				
<b>Post conditions:</b>	All comments of this project is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Click on View Project button or Click on name of the project			
2	DDL	Display View Project page includes 5 tabs: - Campaign - Update - Comments With Campaign tab is default.			
3	User	Click on Comments tab			
4	DDL	Display Comments page with all messages of this project.			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>		High			
<b>Frequency of Use:</b>		High			
<b>Business Rules:</b>		<a href="#">B1</a>			
<b>Other Information:</b>		N/A			

Table 3-7: View comments Use case

### 3.3.2.3 Guest Group Function

#### 3.3.2.3.1 Common Module



**Figure 3-16:** Register Use case

##### 3.3.2.3.1.1 UC006-Register

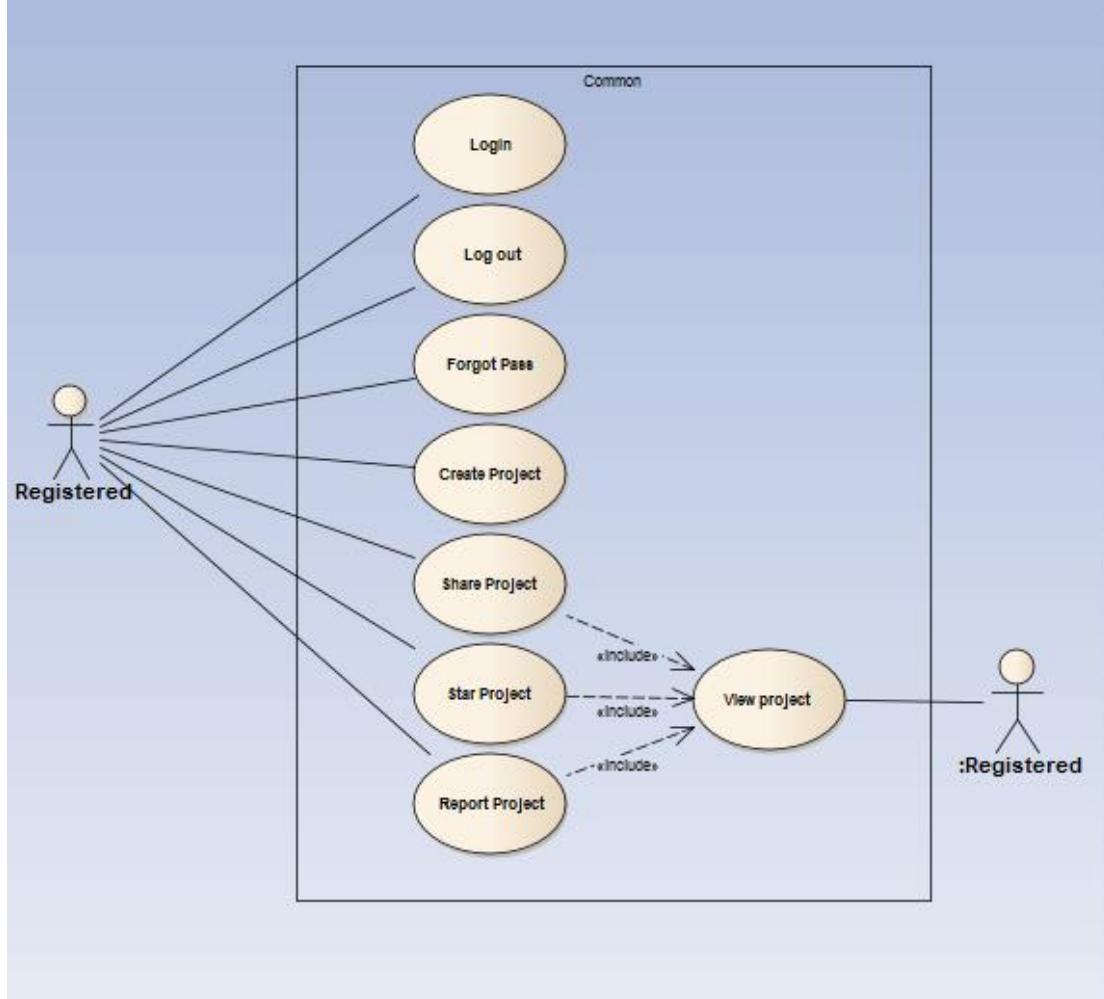
<b>Use Case ID</b>	UC006	<b>Version</b>	1.1
<b>Use Case Name</b>	Register		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Guest	<b>Secondary Actor</b>	
<b>Description:</b>	Visitor can create new account to use more function as a register user		
<b>Preconditions:</b>	1. DDL must be connected to the Internet 2. User browsed DDL website		
<b>Trigger:</b>	Click on register link on the website		
<b>Post conditions:</b>	1. Add account information to database 2. Send confirm email 3. Logs user into system 4. Redirect user to Homepage and display as Member		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	User	Click Register in header	
2	DDL	Display Register page includes: - Header - User name text box - Password text box - Email text box - Date of Birth date picker - Phone number text box	

		- Register button - Cancel button
3	User	Enters data in fields
4	User	Click Register
5	DDL	Send confirm email
6	DDL	Logs user into system
7	DDL	Display Homepage as Member
<b>Alternative Flows</b>		
<b>AT1</b>	At step 4, user click Cancel	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
4.1	DDL	Redirects user to previous page
<b>AT2</b>	At step 4 in the main flows, if user click other hyperlinks or buttons	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
4.1	DDL	Redirects user to chosen hyperlink or button
<b>Exceptions:</b>		
<b>EC1</b>	At step 4, required fields are not entered	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
4.1	DDL	Display Register page with message: with ID MSG5
4.2	DDL	Mark error fields
<b>EC2</b>	At step 4, fields are entered with wrong type of data	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
4.1	DDL	Display Register page with message: with ID MSG6
4.2	DDL	Mark error fields
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>		
<b>Other Information:</b>	N/A	

Table 3-8: Register Use case

### 3.3.2.4 Registered Group Function

#### 3.3.2.4.1 Common module



**Figure 3-17:** Registered common use case

##### 3.3.2.4.1.1 UC007- Login

<b>Use Case ID</b>	UC007	<b>Version</b>	1.1
<b>Use Case Name</b>	Login by social network's account Login by registered account		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered User	<b>Secondary Actor</b>	
<b>Description:</b>	When User want to login by Facebook/Google/Register account to use more functions in website		
<b>Preconditions:</b>	1. DDL must be connected to the Internet 2. User browsed DDL website 3. Account Facebook/Google/Register must be exist and correctly		

	4. User are viewing Login page	
<b>Trigger:</b>	User enter flyawayplus.com	
<b>Post conditions:</b>	1. Log user into system 2. Redirect user to previous page and display as Member	
<b>Normal Flow</b>		
Step	Actor	Action
1	User	Click Login in Homepage
2	DDL	Display Login popup includes: <ul style="list-style-type: none"><li>- User name text box</li><li>- Password text box</li><li>- Login button</li><li>- Facebook button</li><li>- Forgot password hyperlink</li></ul>
3	User	Enters User name and Password
4	User	Click Login
5	DDL	Logs user into system
6	DDL	Close popup
7	DDL	Display previous page as Member
<b>Alternative Flows</b>		
<b>AT1</b>	At step 3 in the main flows, if user click Facebook or Google+ button	
Step	Actor	Action
3.1	DDL	Redirects user to confirm site
3.2	DDL	Logs user into system with Facebook
<b>AT2</b>	At step 4, user choose Cancel:	
Step	Actor	Action
4.1	DDL	Closes popup.
<b>AT3</b>	At step 8 in the main flows, if user click other hyperlinks or buttons	
Step	Actor	Action

8.1	DDL	Redirects user to chosen hyperlink or button
<b>Exceptions:</b>		
<b>EC1</b>	At step 4 in main flow, if user entered wrong User name or Password	
Step	Actor	Action
3.1	DDL	Display Login popup message with ID MSG14
3.2	DDL	Mark error fields
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	N/A	
<b>Other Information:</b>	N/A	

Table 3-9: Login Use case

## 3.3.2.4.1.2 UC008- Logout

<b>Use Case ID</b>	UC008	<b>Version</b>	1.1
<b>Use Case Name</b>	Logout		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	User	<b>Secondary Actor</b>	
<b>Description:</b>	When User want to logout their account		
<b>Preconditions:</b>	1. DDL must be connected to the Internet 2. User browsed DDL website 3. User logged in to system		
<b>Trigger:</b>	User click Logout in header		
<b>Post conditions:</b>	1. Log user out from system. 2. Redirect user to login page		
<b>Normal Flow</b>			
Step	Actor	<b>Action</b>	
1	User	Click Logout	
2	DDL	Display page as Guest	
<b>Alternative Flows: N/A</b>			

<b>Exceptions: N/A</b>	
<b>Priority</b>	High
<b>Frequency of Use:</b>	High
<b>Business Rules:</b>	N/A
<b>Other Information:</b>	N/A

**Table 3-10:Logout Use case****3.3.2.4.1.3 UC009- Forgot password**

<b>Use Case ID</b>	UC009	<b>Version</b>	1.1		
<b>Use Case Name</b>	Forgot password				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	User	<b>Secondary Actor</b>			
<b>Description:</b>	When Member forgot their password to login the system, this function will receiving their password to email				
<b>Preconditions:</b>	1. DDL must be connected to the Internet 2. User browsed DDL website 3. User logged in to system				
<b>Trigger:</b>	Click Forgot password in Login popup				
<b>Post conditions:</b>	1. Send confirm email				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Click Forgot password in Login popup			
2	DDL	Display Email text box			
3	User	Enter registered email			
4	User	Click Send Code			
5	DDL	Send reset code to email			
6	User	Enter reset code			
7	User	Click Reset			
8	DDL	Send new password to email			
<b>Alternative Flows</b>					
<b>AT2</b>	At step 3 in the main flow, user clicks on Cancel button				

<b>Step</b>	<b>Actor</b>	<b>Action</b>
3.1	DDL	Return to previous screen
<b>AT2</b>		At step 4 in the main flows, if user click other hyperlinks or buttons
<b>Step</b>	<b>Actor</b>	<b>Action</b>
4.1	DDL	Redirects user to chosen hyperlink or button
<b>Exceptions:</b>		
<b>EC1</b>	At step 4, entered email is invalid	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
4.1	DDL	Displays Login popup with message: with ID MSG15
4.2	DDL	Marks email text box
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	N/A	
<b>Other Information:</b>	N/A	

**Table 3-11: Forgot password Use case****3.3.2.4.1.4 UC010-Create Project**

<b>Use Case ID</b>	UC010	<b>Version</b>	1.1
<b>Use Case Name</b>	Create project		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to create a project		
<b>Preconditions:</b>	1.DDL website is available 2.User browsed DDL website 3.User logged in to system 4.User are on project information page		
<b>Trigger:</b>	User click on button start		

<b>Post conditions:</b>		Create a project page is displayed
<b>Normal Flow</b>		
Step	Actor	Action
1	User	Click start
2	DDL	Show create project page basic
3	User	Choose rank , type and fill project name
4	User	Click button Start
5	DDL	Show create project page detail
6	User	Fill all information about project
7	User	Click Submit for review
8	DDL	Send message to admin for review
<b>Alternative Flows: N/A</b>		
<b>Exceptions:</b>		
<b>EC1</b>	At step 3 and 6 in the main flows, if user don't fill all information of a project	
Step	Actor	Action
3.1	DDL	Display Dialog to remind user
<b>Priority</b>	Normal	
<b>Frequency of Use:</b>	Normal	
<b>Business Rules:</b>	N/A	
<b>Other Information:</b>	N/A	

**Table 3-12:Create project Use case****3.3.2.4.1.5 UC011- Share project**

<b>Use Case ID</b>	UC011	<b>Version</b>	1.1
<b>Use Case Name</b>	Share project		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to share a project		
<b>Preconditions:</b>	1.DDL website is available		

	2.User browsed DDL website 3.User logged in to system 4.User are on project information page	
<b>Trigger:</b>	User click on button report	
<b>Post conditions:</b>	Share a project page is displayed	
<b>Normal Flow</b>		
Step	Actor	Action
1	User	Choose a project
2	User	Click Share
3	DDL	Show dialog ask User confirm for share on Facebook
4	User	Click share to confirm
<b>Alternative Flows: N/A</b>		
<b>Exceptions: N/A</b>		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>		
<b>Other Information:</b>	N/A	

Table 3-13: Share project Use case

## 3.3.2.4.1.6 UC012- Remind project

<b>Use Case ID</b>	UC012	<b>Version</b>	1.1
<b>Use Case Name</b>	Remind project		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to remind a project		
<b>Preconditions:</b>	1.DDL website is available 2.User browsed DDL website 3.User logged in to system 4.User are on project information page		
<b>Trigger:</b>	User click on button report		

<b>Post conditions:</b>		Project page information is displayed
<b>Normal Flow</b>		
Step	Actor	Action
1	User	Choose a project
2	User	Click on icon Star
3	DDL	Save this project into User's Starred Project list
<b>Alternative Flows: N/A</b>		
<b>Exceptions: N/A</b>		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		
<b>Other Information:</b>		N/A

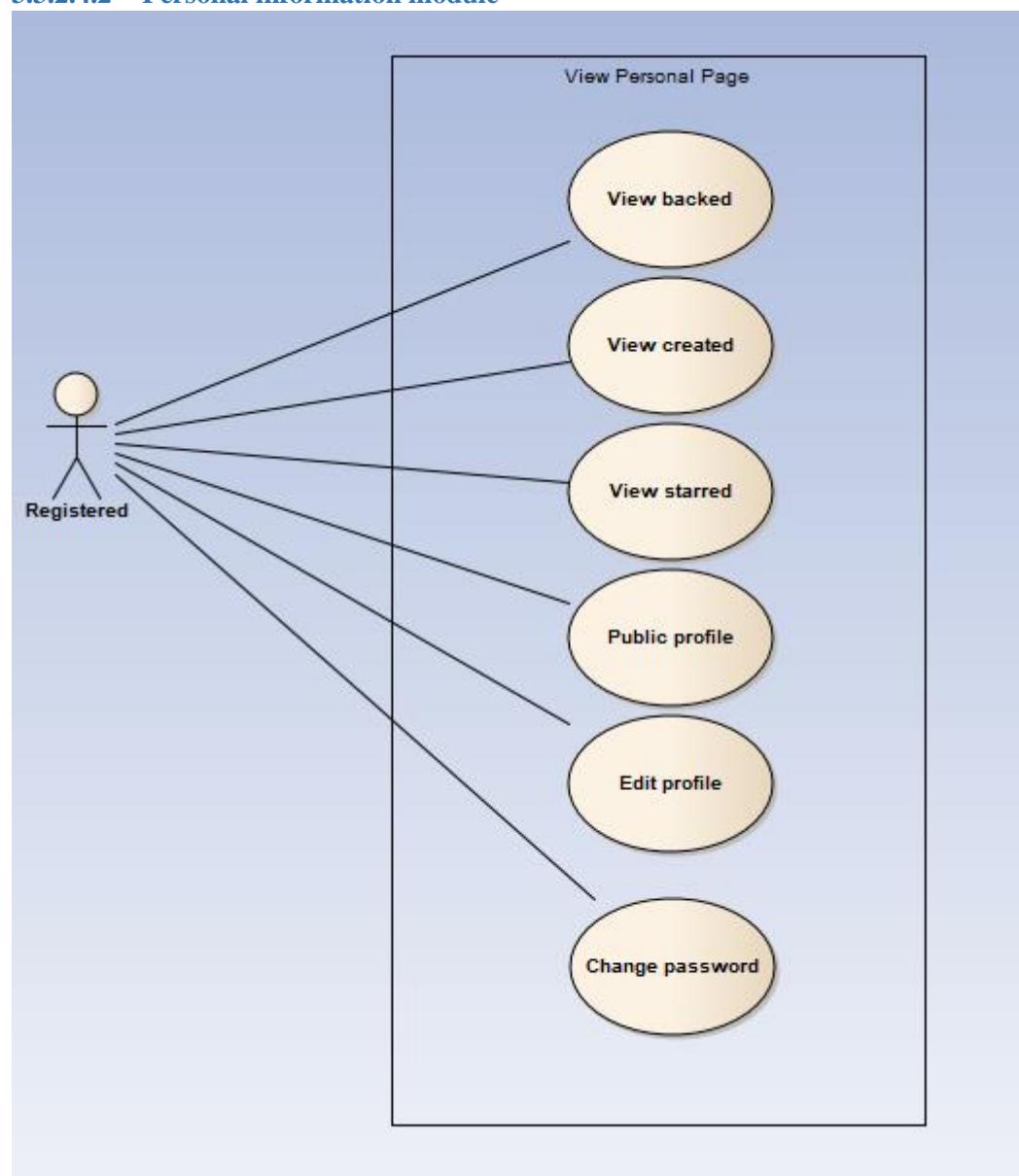
**Table 3-14:Star project Use case****3.3.2.4.1.7 UC013- Report project**

<b>Use Case ID</b>	UC013	<b>Version</b>	1.1
<b>Use Case Name</b>	Report project		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to report a project		
<b>Preconditions:</b>	1.DDL website is available 2.User browsed DDL website 3.User logged in to system 4.User are on project information page		
<b>Trigger:</b>	User click on button report		
<b>Post conditions:</b>	Project page information is displayed		
<b>Normal Flow</b>			
Step	Actor	Action	
1	User	Choose a project	
2	User	Click Report	

3	DDL	Show confirm dialog with question ask about reason
4	User	Fill reason then confirm
5	DDL	Send report
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b> N/A		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		
<b>Other Information:</b>		N/A

**Table 3-15: Report project Use case**

### 3.3.2.4.2 Personal information module



**Figure 3-18:** Personal information module use cases

## 3.3.2.4.2.1 UC014- View backed

<b>Use Case ID</b>	UC014	<b>Version</b>	1.1		
<b>Use Case Name</b>	View backed project				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	DDL records all project which user backed				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system				
<b>Trigger:</b>	User click View Backed Project				
<b>Post conditions:</b>	Backed Project Page is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Click Avatar image button in header			
2	DDL	Display a menu includes: - View Profile menu item - Logout menu item			
3	User	Click View Profile			
4	DDL	Redirect user to Profile page includes: - Header - Backed project thumbnail			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

Table 3-16: View backed project Use case

## 3.3.2.4.2.2 UC015- View created project

<b>Use Case ID</b>	UC015	<b>Version</b>	1.1		
<b>Use Case Name</b>	Display created project				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	DDL records all project which user backed				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system				
<b>Trigger:</b>	User click View Created Project				
<b>Post conditions:</b>	Created Project Page is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Click Avatar image button in header			
2	DDL	Display a menu includes: - View Profile menu item - Logout menu item			
3	User	Click View Profile			
4	DDL	Redirect user to Profile page includes: - Header - Created project thumbnail			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

Table 3-17: View created project Use case

## 3.3.2.4.2.3 UC016- View starred project

<b>Use Case ID</b>	UC016	<b>Version</b>	1.1		
<b>Use Case Name</b>	Display starred project				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	DDL records all project which user starred				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system				
<b>Trigger:</b>	User click View Starred Project				
<b>Post conditions:</b>	Starred Project Page is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Click Avatar image button in header			
2	DDL	Display a menu includes: - View Profile menu item - Logout menu item			
3	User	Click View Profile			
4	DDL	Redirect user to Profile page includes: - Header - Starred project thumbnail			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

Table 3-18: View reminded project Use case

## 3.3.2.4.2.4 UC017- Public profile

<b>Use Case ID</b>	UC017	<b>Version</b>	1.1		
<b>Use Case Name</b>	View Public profile				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	DDL records all of user's activities and display				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system				
<b>Trigger:</b>	User click View Profile				
<b>Post conditions:</b>	Personal Page is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Click Avatar image button in header			
2	DDL	Display a menu includes: - View Profile menu item - Logout menu item			
3	User	Click View Profile			
4	DDL	Redirect user to Profile page includes: - Header - Cover image label - Profile panel - Facebook connect status - Contact button			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

Table 3-19: Public profile Use case

## 3.3.2.4.2.5 UC018- Edit profile

<b>Use Case ID</b>	UC018	<b>Version</b>	1.1		
<b>Use Case Name</b>	Edit Profile				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows user to edit profile				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system				
<b>Trigger:</b>	User click Edit profile				
<b>Post conditions:</b>	Edit profile Page is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Click Avatar image button in header			
2	DDL	Display a menu includes: - View Profile menu item - Logout menu item			
3	User	Click View Profile			
4	DDL	Redirect user to Profile page includes: - Header - Profile text box			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

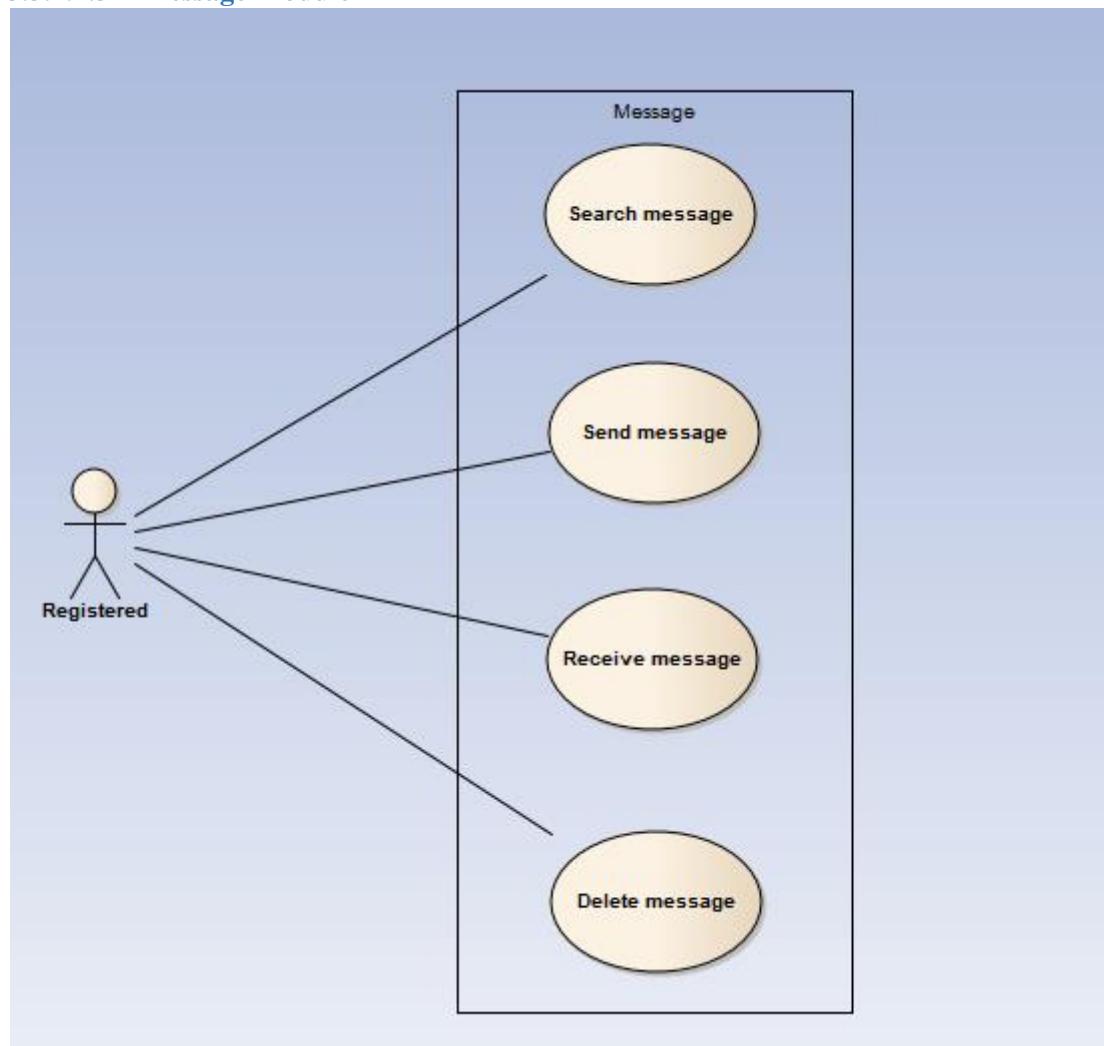
Table 3-20: Edit Profile Use case

## 3.3.2.4.2.6 UC019- Change password

<b>Use Case ID</b>	UC019	<b>Version</b>	1.1		
<b>Use Case Name</b>	Change password				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows user to change password				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system				
<b>Trigger:</b>	User click Change password				
<b>Post conditions:</b>	Change password Page is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Click Avatar image button in header			
2	DDL	Display a menu includes: - View Profile menu item - Logout menu item			
3	User	Click View Profile			
4	DDL	Redirect user to Profile page includes: - Header - Password and new password text box - Mail confirm text box			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

Table 3-21: Change password

### 3.3.2.4.3 Message Module



**Figure 3-19:** Message module use cases

### 3.3.2.4.3.1 UC020-Search Message

<b>Use Case ID</b>	UC020	<b>Version</b>	1.1
<b>Use Case Name</b>	Search message by content , sender		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	DDL records all project which user backed		
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system		

<b>Trigger:</b>	User click Message then enters search phrase and click Search or press Enter	
<b>Post conditions:</b>	List all results that matched with the keyword user enters will be displayed in the Search Details screen	
<b>Normal Flow</b>		
Step	Actor	Action
1	User	Enter search phrase to text box
2	User	Click Search or press Enter
3	DDL	List all results that matched with the keyword user entered
<b>Alternative Flows: N/A</b>		
<b>Exceptions:</b>		
<b>EC1</b>	At step 3 in the main flows, if there is no result matched the keywords	
Step	Actor	Action
3.1	DDL	Display Basic Guide page with text: "Nothing found :("
<b>Priority</b>	Normal	
<b>Frequency of Use:</b>	Normal	
<b>Business Rules:</b>	N/A	
<b>Other Information:</b>	N/A	

**Table 3-22: Search message Use case**

### 3.3.2.4.3.2 UC021-Send Message

<b>Use Case ID</b>	UC021	<b>Version</b>	1.1		
<b>Use Case Name</b>	Send message				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows user to send a message				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system 4. User are on message page				
<b>Trigger:</b>	User enters message and click Send				
<b>Post conditions:</b>	Message page is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Filled all information ( subject , to , content )			
2	User	Click Send			
3	DDL	Send this message			
4	DDL	Displays message			
<b>Alternative Flows:</b>					
<b>AT1</b>	At step 2 in the main flows, if user click other hyperlinks or buttons				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
2.1	DDL	Redirects user to chosen hyperlink or button			
<b>AT2</b>	At step 2 in the main flows, if user click other hyperlinks or buttons				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
2.1	DDL	Redirects user to chosen hyperlink or button			
<b>Exceptions:</b>					
<b>EC1</b>	Username of who user want to send a message is not exist				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			

1	DDL	Show a dialog error
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>		
<b>Other Information:</b>	N/A	

**Table 3-23: Send message use case****3.3.2.4.3.3 UC022-Delete Message**

<b>Use Case ID</b>	UC022	<b>Version</b>	1.1		
<b>Use Case Name</b>	Delete message				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows user to delete a message				
<b>Preconditions:</b>	1.DDL website is available 2.User browsed DDL website 3.User logged in to system 4.User are on message page				
<b>Trigger:</b>	User choose a message then click delete				
<b>Post conditions:</b>	Message page is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	User	Choose a message			
2	User	Click Delete			
3	DDL	Show confirm dialog			
4	User	Confirm			
5	DDL	Delete message			
<b>Alternative Flows:</b>					
<b>AT1</b>	At step 2 in the main flows, if user click other hyperlinks or buttons				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			

2.1	DDL	Redirects user to chosen hyperlink or button
<b>Exceptions: N/A</b>		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		
<b>Other Information:</b>		N/A

Table 3-24: Delete message Use case

### 3.3.2.4.4 Comment module

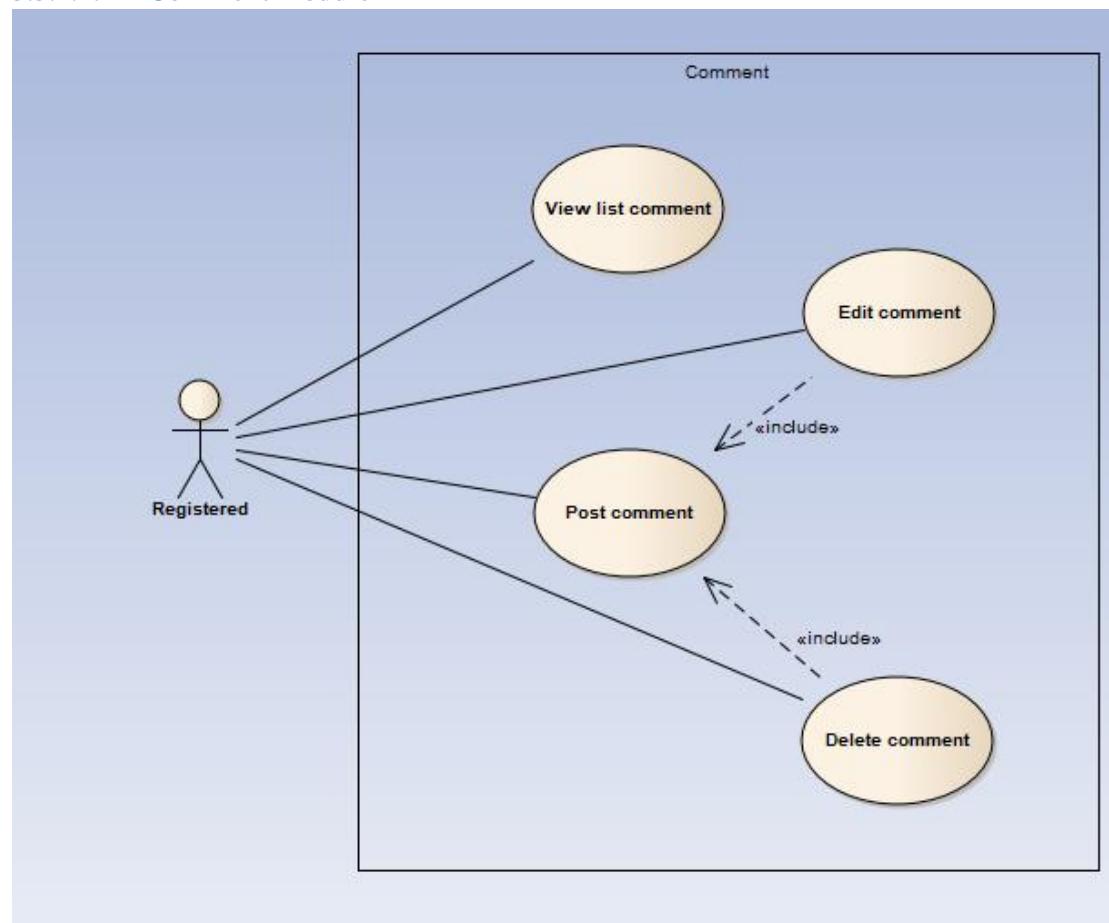


Figure 3-20: Comment module use cases

## 3.3.2.4.4.1 UC023- View list comment

<b>Use Case ID</b>	UC023	<b>Version</b>	1.1		
<b>Use Case Name</b>	View list comment				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	DDL show all comment of a project				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system 4. User clicked on a project				
<b>Trigger:</b>	User click on comment tab in project information page				
<b>Post conditions:</b>	Comment tab is displayed				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	User	Click on a project			
2	DDL	Display project information page			
3	User	Click comment tab			
4	DDL	Show comment tab include all comment of project			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

Table 3-25: View list comment Use case

## 3.3.2.4.4.2 UC024- Post comment

<b>Use Case ID</b>	UC024	<b>Version</b>	1.1		
<b>Use Case Name</b>	Post comment				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	User post a comment				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system 4. User clicked on a project				
<b>Trigger:</b>	User click on comment tab in project information page				
<b>Post conditions:</b>	Comment tab is displayed				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	User	Click on a project			
2	DDL	Display project information page			
3	User	Click comment tab			
4	DDL	Show comment tab include all comment of project			
5	User	Fill the comment text box then click post			
6	DDL	Post the comment user sent			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

Table 3-26: Post comment Use case

## 3.3.2.4.4.3 UC025- Edit comment

<b>Use Case ID</b>	UC025	<b>Version</b>	1.1		
<b>Use Case Name</b>	Edit comment				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	User edit comment				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system 4. User clicked on a project				
<b>Trigger:</b>	User click on comment tab in project information page				
<b>Post conditions:</b>	Comment tab is displayed				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	User	Click on a project			
2	DDL	Display project information page			
3	User	Click comment tab			
4	DDL	Show comment tab include all comment of project			
5	User	Click edit on where user's comment posted			
6	DDL	Show edit comment text box			
7	User	Edit then click post			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

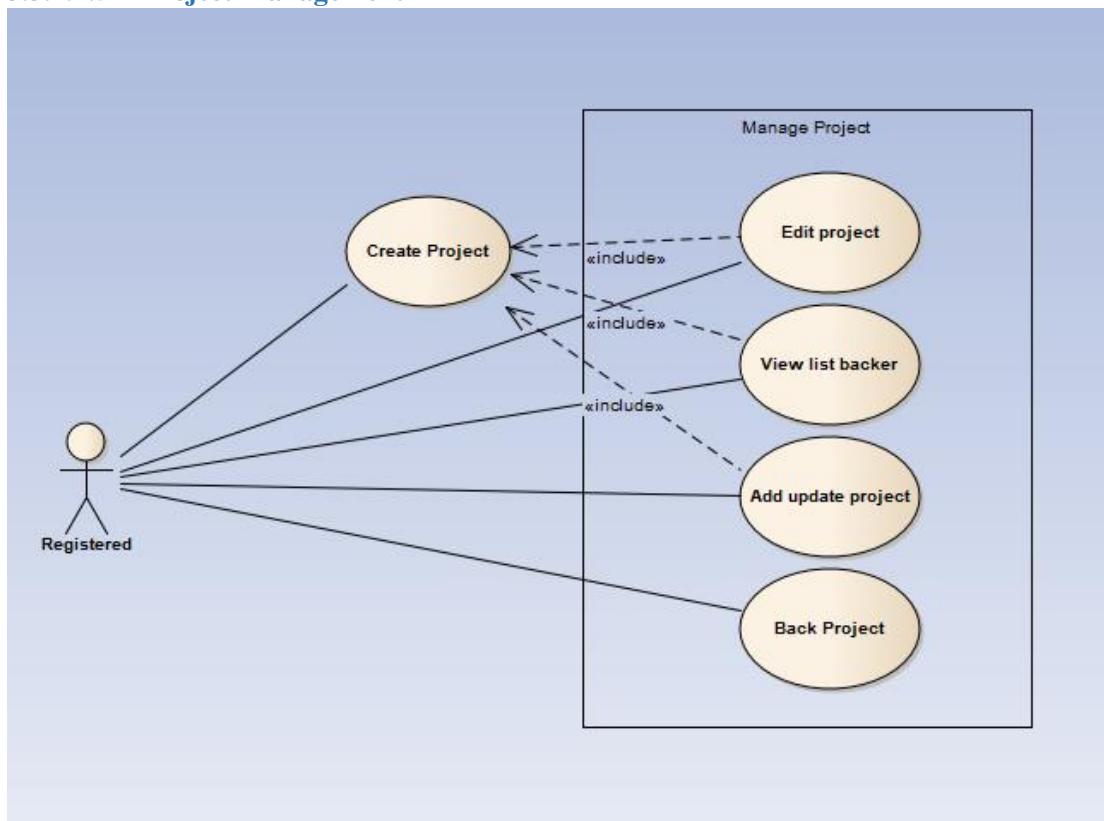
Table 3-27: Registered user Use case

## 3.3.2.4.4.4 UC026- Delete comment

<b>Use Case ID</b>	UC026	<b>Version</b>	1.1		
<b>Use Case Name</b>	Delete comment				
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015		
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>			
<b>Description:</b>	User edit comment				
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system 4. User clicked on a project				
<b>Trigger:</b>	User click on comment tab in project information page				
<b>Post conditions:</b>	Comment tab is displayed				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	User	Click on a project			
2	DDL	Display project information page			
3	User	Click comment tab			
4	DDL	Show comment tab include all comment of project			
5	User	Click delete on where user's comment posted			
6	DDL	Delete the comment			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	N/A				
<b>Other Information:</b>	N/A				

Table 3-28: Delete comment Use case

### 3.3.2.4.5 Project Management



**Figure 3-21:** Project management use cases

#### 3.3.2.4.5.1 UC027- Edit Project

Use Case ID	UC027	Version	1.1
Use Case Name	Edit Project		
Created by:	AnhDD	Date Created:	24/09/2015
Primary Actor	Registered user	Secondary Actor	
Description:	This function allows user to edit Project		
Preconditions:	1. DDL website is available 2. User browsed DDL website 3. User logged in to system		
Trigger:	User click Edit		
Post conditions:	Edit project Page is displayed		
<b>Normal Flow</b>			
Step	Actor	Action	
1	User	Choose a project what user's own	

2	User	Click edit
3	User	Edit information
4	User	Click save
5	DDL	Save change
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b>		
<b>EC1</b>	At step 3 in the main flows, if user miss fill 1 or more information	
Step	Actor	Action
3.1	DDL	Display Dialog to remind user
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	N/A	
<b>Other Information:</b>	N/A	

**Table 3-29: Edit Project Use case****3.3.2.4.5.2 UC028-View list Backing**

<b>Use Case ID</b>	UC028	<b>Version</b>	1.1
<b>Use Case Name</b>	View list backer		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to edit Project		
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system		
<b>Trigger:</b>	User click list backer tab		
<b>Post conditions:</b>	Project information page is displayed		
<b>Normal Flow</b>			
Step	Actor	<b>Action</b>	
1	User	Choose a project what user's own	

2	User	Click on list backer tab
3	DDL	Show list backer table
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b> N/A		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		N/A
<b>Other Information:</b>		N/A

**Table 3-30:View list backing Use case****3.3.2.4.5.3 UC029-Add update project**

<b>Use Case ID</b>	UC029	<b>Version</b>	1.1
<b>Use Case Name</b>	Add update project		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to edit Project		
<b>Preconditions:</b>	1. DDL website is available 2. User browsed DDL website 3. User logged in to system		
<b>Trigger:</b>	User click add update		
<b>Post conditions:</b>	Project information page is displayed		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	User	Choose a project what user's own	
2	User	Click on update tab	
3	User	Click add update	
4	User	Fill information about update	
5	User	Click save	
6	DDL	Add this update into update tab	
<b>Alternative Flows:</b> N/A			

<b>Exceptions: N/A</b>		
<b>EC1</b>	At step 4 in the main flows, if user don't fill information	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
3.1	DDL	Display Dialog to remind user
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	N/A	
<b>Other Information:</b>	N/A	

**Table 3-31: Add update project Use case****3.3.2.4.5.4 UC030- Back Project**

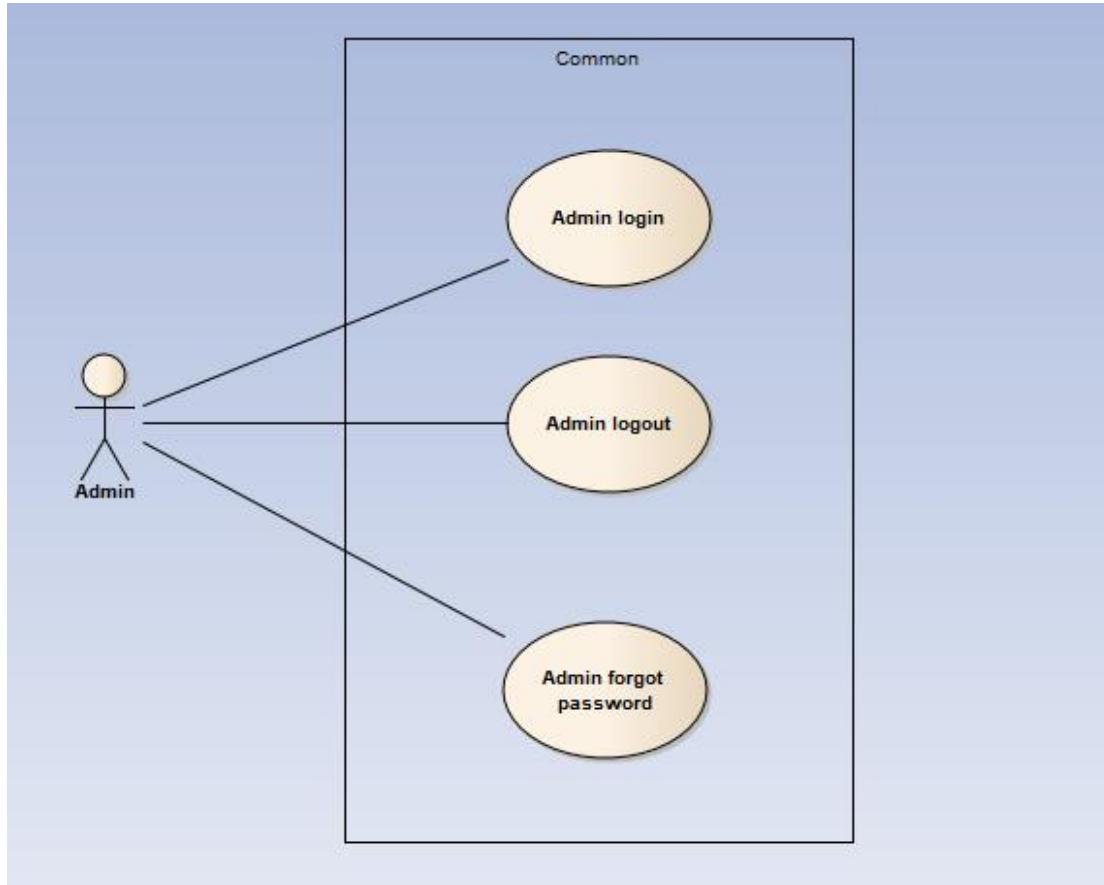
<b>Use Case ID</b>	UC030	<b>Version</b>	1.1
<b>Use Case Name</b>	Back project		
<b>Created by:</b>	AnhDD	<b>Date Created:</b>	24/09/2015
<b>Primary Actor</b>	Registered user	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows user to back a project		
<b>Preconditions:</b>	1.DDL website is available 2.User browsed DDL website 3.User logged in to system 4.User are on project information page		
<b>Trigger:</b>	User click on button back		
<b>Post conditions:</b>	Back a project page is displayed		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	User	Choose a project	
2	DDL	Show project information page	
3	User	Click Back this project	
4	User	Choose reward	
5	User	Click continue	
6	DDL	Show Backer's information page	

7	User	Fill all information
8	User	Can check or don't check the Anonymous check box for hide info
9	User	Choose the bank user want to pay
10	User	Click submit
11	DDL	Submit this payment
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b>		
<b>EC1</b>	At step 7,8,9 in the main flows, if user miss 1 or more information	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
3.1	DDL	Display Dialog to remind user
<b>Priority</b>	Normal	
<b>Frequency of Use:</b>	Normal	
<b>Business Rules:</b>	N/A	
<b>Other Information:</b>	N/A	

**Table 3-32:Back project Use case**

### 3.3.2.5 Administrator Group Function

#### 3.3.2.5.1 Common module



**Figure 3-22:** Common use case of administrator group function

##### 3.3.2.5.1.1 UC031- Admin login

Use Case ID	UC031	Version	1.1
Use Case Name	Admin login		
Created by:	MaiCTP	Date Created:	28/09/2015
Primary Actor	Administrator(Admin)	Secondary Actor	
Description:	This function allows Admin logins into website		
Preconditions:	3. DDL website is available 4. Admin browsed DDL website 5. Admin are viewing Login page for admin		
Trigger:	Admin clicks on Login button		
Post conditions:	3. Log Admin into system 4. Redirect Admin to Admin home page		
Normal Flow			

<b>Step</b>	<b>Actor</b>	<b>Action</b>
1	DDL	Display Admin Login page. If Admin choose Remember Me function before, fill all needed information
2	Admin	Type email address and password if Admin does not choose Remember Me function before
3	DDL	Verify: a. Email address b. Password
4	DDL	- Log Admin into system - Redirect Admin to Admin home page
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b>		
<b>EC1</b>	At step 2 in main flow, if Admin entered wrong Email address or Password	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
2.1	DDL	Display Login popup message point out error field
2.2	DDL	Mark error fields
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		<a href="#">B1</a>
<b>Other Information:</b>		N/A

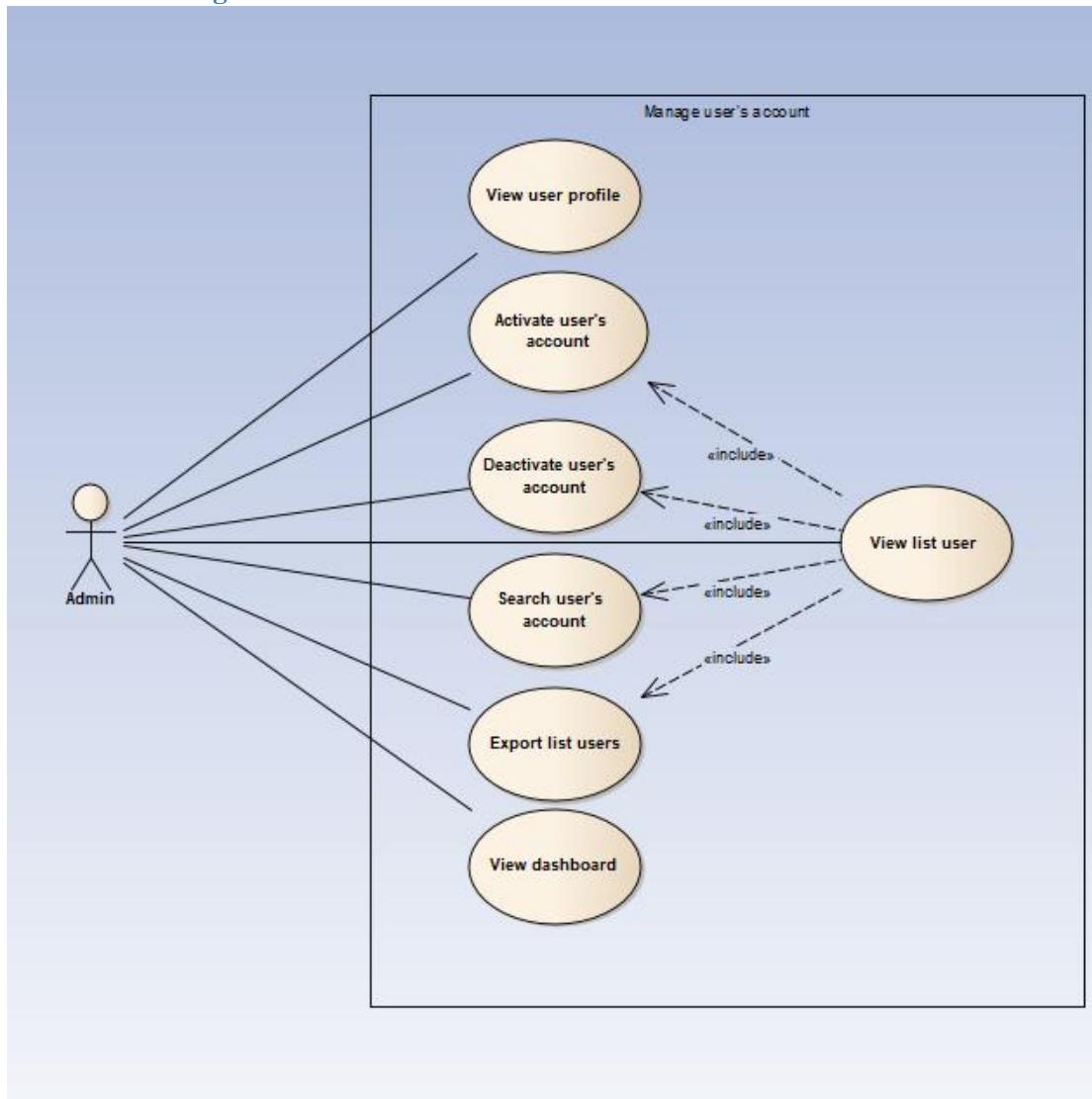
**Table 3-33:Admin login Use case**

## 3.3.2.5.1.2 UC032- Admin logout

<b>Use Case ID</b>	UC032	<b>Version</b>	1.1		
<b>Use Case Name</b>	Admin logout				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Admin logouts of DDL website				
<b>Preconditions:</b>	1. Admin logged in DDL website as Administrator				
<b>Trigger:</b>	Admin clicks on Logout option				
<b>Post conditions:</b>	Admin is logged out from DDL website				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	<ul style="list-style-type: none"> <li>- Click on dropdown at right side of DDL website</li> <li>- Choose Logout option</li> </ul>			
2	DDL	<ul style="list-style-type: none"> <li>- Logout Admin</li> <li>- Redirect to Admin Login page</li> </ul>			
<b>Alternative Flows:N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

Table 3-34:Admin logout Use case

### 3.3.2.5.2 Manage user's account



#### 3.3.2.5.2.1 UC033- View user list

<b>Use Case ID</b>	UC033	<b>Version</b>	1.1
<b>Use Case Name</b>	View user list		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to view User list of the project		
<b>Preconditions:</b>	1. DDL website is available 2. Admin browsed DDL website 3. Admin logged in as Administrator		
<b>Trigger:</b>	Admin clicks on Users List		
<b>Post conditions:</b>	Users List is displayed		

<b>Normal Flow</b>		
<b>Step</b>	<b>Actor</b>	<b>Action</b>
1	Admin	<ul style="list-style-type: none"> <li>- Click on Users at left side</li> <li>- Click on Users List</li> </ul>
2	DDL	Display list of users
<b>Alternative Flows:</b>		
<b>AT1</b>	At step 1 in the main flows, if do not have any user	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
2.1	DDL	Display “Do not have any user” message
<b>Exceptions: N/A</b>		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

**Table 3-35:View user list Use case****3.3.2.5.2.2 UC034- View user profile**

<b>Use Case ID</b>	UC034	<b>Version</b>	1.1
<b>Use Case Name</b>	View user profile		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to view profile of an user		
<b>Preconditions:</b>	Admin is viewing Users List		
<b>Trigger:</b>	Admin clicks on View button		
<b>Post conditions:</b>	Profile of this user is displayed		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	Admin	Click on View button at Action column of the user whom want to see profile	

2	DDL	Display profile of this user
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b> N/A		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

**Table 3-36:View user profile Use case****3.3.2.5.2.3 UC035- Activate user's account**

<b>Use Case ID</b>	UC035	<b>Version</b>	1.1		
<b>Use Case Name</b>	Activate user's account				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to activate an user's account				
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>- Admin is viewing Users List</li> <li>- Status of this user is deactivate</li> </ul>				
<b>Trigger:</b>	Admin clicks on Deactivate button				
<b>Post conditions:</b>	User's account is activated				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on Deactivate button at Status column of the user whom want to activate			
2	DDL	<ul style="list-style-type: none"> <li>- Change displayed status from Deactivate to Activate</li> <li>- Change status of this user's account to Activate on database</li> </ul>			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				

<b>Other Information:</b>	N/A		
---------------------------	-----	--	--

**Table 3-37: Activate user's account Use case****3.3.2.5.2.4 UC036- Deactivate user's account**

<b>Use Case ID</b>	UC036		<b>Version</b>	1.1			
<b>Use Case Name</b>	Deactivate user's account						
<b>Created by:</b>	MaiCTP		<b>Date Created:</b>	28/09/2015			
<b>Primary Actor</b>	Administrator(Admin)		<b>Secondary Actor</b>				
<b>Description:</b>	This function allows Administrator to deactivate an user's account						
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>- Admin is viewing Users List</li> <li>- Status of this user is activate</li> </ul>						
<b>Trigger:</b>	Admin clicks on Activate button						
<b>Post conditions:</b>	User's account is deactivated						
<b>Normal Flow</b>							
<b>Step</b>	<b>Actor</b>	<b>Action</b>					
1	Admin	Click on Activate button at Status column of the user whom want to deactivate					
2	DDL	<ul style="list-style-type: none"> <li>- Change displayed status from Activate to Deactivate</li> <li>- Change status of this user's account to Deactivate on database</li> </ul>					
<b>Alternative Flows:</b> N/A							
<b>Exceptions:</b> N/A							
<b>Priority</b>		High					
<b>Frequency of Use:</b>		High					
<b>Business Rules:</b>		<a href="#">B1</a>					
<b>Other Information:</b>		N/A					

**Table 3-38: Deactivate user's account Use case**

## 3.3.2.5.2.5 UC037- Search user's account

<b>Use Case ID</b>	UC037	<b>Version</b>	1.1		
<b>Use Case Name</b>	Search user's account				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to search an user's account				
<b>Preconditions:</b>	Admin is viewing Users List				
<b>Trigger:</b>	Admin types name of user at Search textbox				
<b>Post conditions:</b>	Display list of searched user				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	- Type name of user whom want to search			
2	DDL	- Display list of users who have the name is typed			
<b>Alternative Flows:</b>					
<b>AT1</b>	At step 1 in the main flows, if the typed name is not exist				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
2.1	DDL	Display "No matching records found" message			
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

Table 3-39:Search user's account Use case

## 3.3.2.5.2.6 UC038- View user's dashboard

<b>Use Case ID</b>	UC038	<b>Version</b>	1.1
<b>Use Case Name</b>	View user's dashboard		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015

<b>Primary Actor</b>	Administrator(Admin)		<b>Secondary Actor</b>				
<b>Description:</b>	This function allows Administrator to view dashboard about user of the project						
<b>Preconditions:</b>	1. DDL website is available 2. Admin browsed DDL website 3. Admin logged in as Administrator						
<b>Trigger:</b>	Admin clicks on Dashboard						
<b>Post conditions:</b>	Dashboard about user is displayed						
<b>Normal Flow</b>							
<b>Step</b>	<b>Actor</b>	<b>Action</b>					
1	Admin	<ul style="list-style-type: none"> <li>- Click on Users at left side</li> <li>- Click on Dashboard</li> </ul>					
2	DDL	Display User Dashboard page					
<b>Alternative Flows:</b> N/A							
<b>Exceptions:</b> N/A							
<b>Priority</b>	High						
<b>Frequency of Use:</b>	High						
<b>Business Rules:</b>	<a href="#">B1</a>						
<b>Other Information:</b>	N/A						

Table 3-40:View user's dashboard Use case

### 3.3.2.5.3 Manage projects

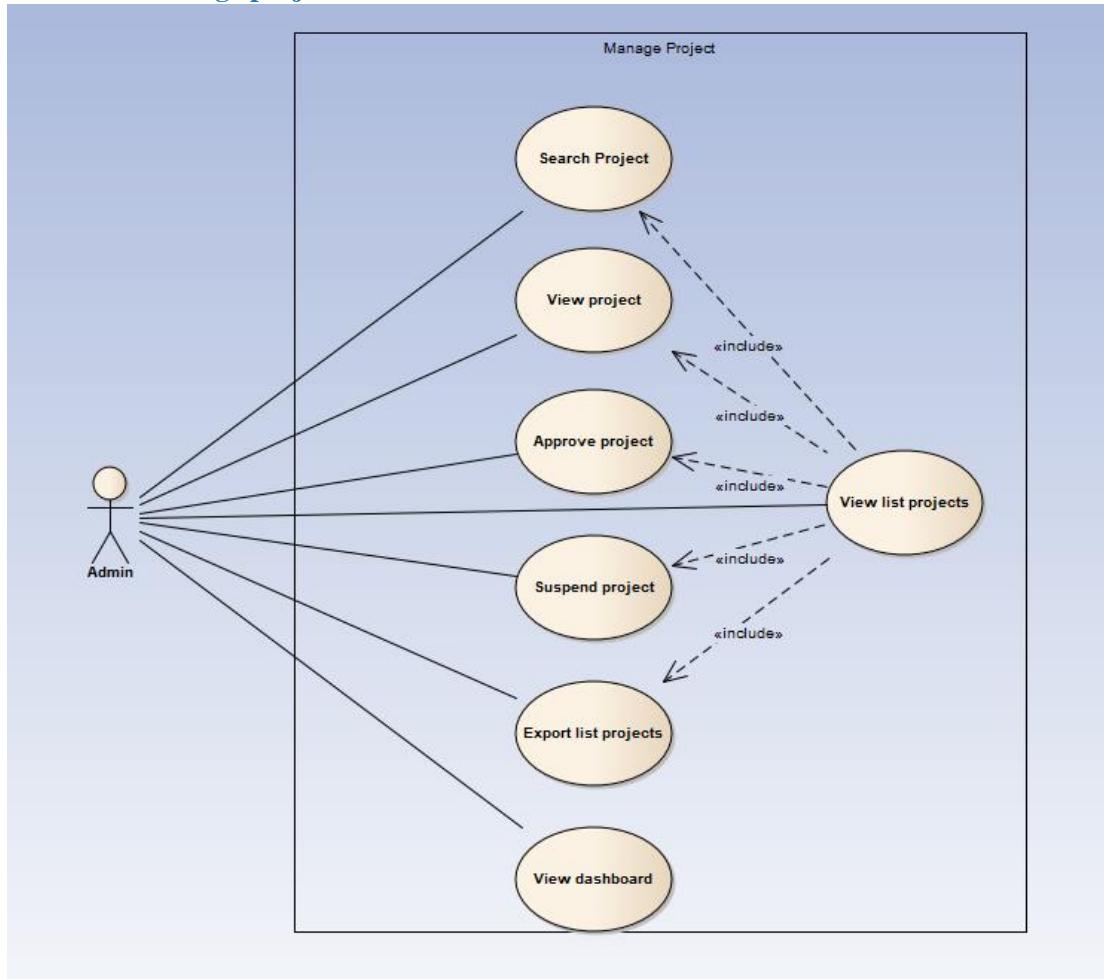


Figure 3-23: Manage project use cases

#### 3.3.2.5.3.1 UC039- View project list

<b>Use Case ID</b>	UC039	<b>Version</b>	1.1
<b>Use Case Name</b>	View project list		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to view project list of the project		
<b>Preconditions:</b>	1. DDL website is available 2. Admin browsed DDL website 3. Admin logged in as Administrator		
<b>Trigger:</b>	Admin clicks on Projects List		
<b>Post conditions:</b>	Project List is displayed		
<b>Normal Flow</b>			

Step	Actor	Action
1	Admin	<ul style="list-style-type: none"> <li>- Click on Project at left side</li> <li>- Click on Projects List</li> </ul>
2	DDL	Display list of projects
<b>Alternative Flows:</b>		
<b>AT1</b>	At step 1 in the main flows, if do not have any project	
Step	Actor	Action
2.1	DDL	Display "Do not have any project" message
<b>Exceptions: N/A</b>		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

Table 3-41: View project list Use case

## 3.3.2.5.3.2 UC040- Search project

<b>Use Case ID</b>	UC040	<b>Version</b>	1.1
<b>Use Case Name</b>	Search project		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to search a project		
<b>Preconditions:</b>	Admin is viewing Projects List		
<b>Trigger:</b>	Admin types name of project on Search textbox		
<b>Post conditions:</b>	Display list of searched project		
<b>Normal Flow</b>			
Step	Actor	<b>Action</b>	
1	Admin	<ul style="list-style-type: none"> <li>- Type name of project which want to search</li> </ul>	
2	DDL	<ul style="list-style-type: none"> <li>- Display list of projects which have the name is typed</li> </ul>	

<b>Alternative Flows:</b>		
<b>AT1</b>	At step 1 in the main flows, if the typed name is not exist	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
2.1	DDL	Display “No matching records found” message
<b>Exceptions: N/A</b>		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

**Table 3-42:Search project Use case****3.3.2.5.3.3 UC041- View (review) project**

<b>Use Case ID</b>	UC041	<b>Version</b>	1.1		
<b>Use Case Name</b>	View (review) project				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to view information of a project				
<b>Preconditions:</b>	Admin is viewing Project List				
<b>Trigger:</b>	Admin clicks on button at Action column				
<b>Post conditions:</b>	Information of this project is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Admin clicks on button at Action column of the project which want to view information			
2	DDL	Display Project Detail page of this project			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				

<b>Frequency of Use:</b>	High
<b>Business Rules:</b>	<a href="#">B1</a>
<b>Other Information:</b>	N/A

**Table 3-43: View (review) project Use case****3.3.2.5.3.4 UC042- Approve project**

<b>Use Case ID</b>	UC042	<b>Version</b>	1.1		
<b>Use Case Name</b>	Approve project				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to approve a project				
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>- Admin is viewing Project List</li> <li>- Status of this Project is Suspend</li> </ul>				
<b>Trigger:</b>	Admin clicks on button at Status column				
<b>Post conditions:</b>	This project is approved				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on button at Status column of the project which want to approve			
2	DDL	<ul style="list-style-type: none"> <li>- Change displayed status to Approve</li> <li>- Change status of this user's account to Approve on database</li> </ul>			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

**Table 3-44: Approve project Use case**

### 3.3.2.5.3.5 UC043- Suspend project

<b>Use Case ID</b>	UC043	<b>Version</b>	1.1		
<b>Use Case Name</b>	Suspend project				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to suspend a project				
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>- Admin is viewing Project List</li> <li>- Status of this Project is Approve</li> </ul>				
<b>Trigger:</b>	Admin clicks on button at Status column				
<b>Post conditions:</b>	This project is suspended				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on button at Status column of the project which want to suspend			
2	DDL	<ul style="list-style-type: none"> <li>- Change displayed status to Suspend</li> <li>- Change status of this user's account to Suspend on database</li> </ul>			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

**Table 3-45:Suspend project Use case**

## 3.3.2.5.3.6 UC044- Export project list

<b>Use Case ID</b>	UC044	<b>Version</b>	1.1		
<b>Use Case Name</b>	Export users list				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to export list of projects				
<b>Preconditions:</b>	- Admin is viewing Project List				
<b>Trigger:</b>	Admin clicks on Export button				
<b>Post conditions:</b>	List of projects is exported				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on Export button at right side Project list table			
2	DDL	Export list of projects to file			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

Table 3-46:Export users list Use case

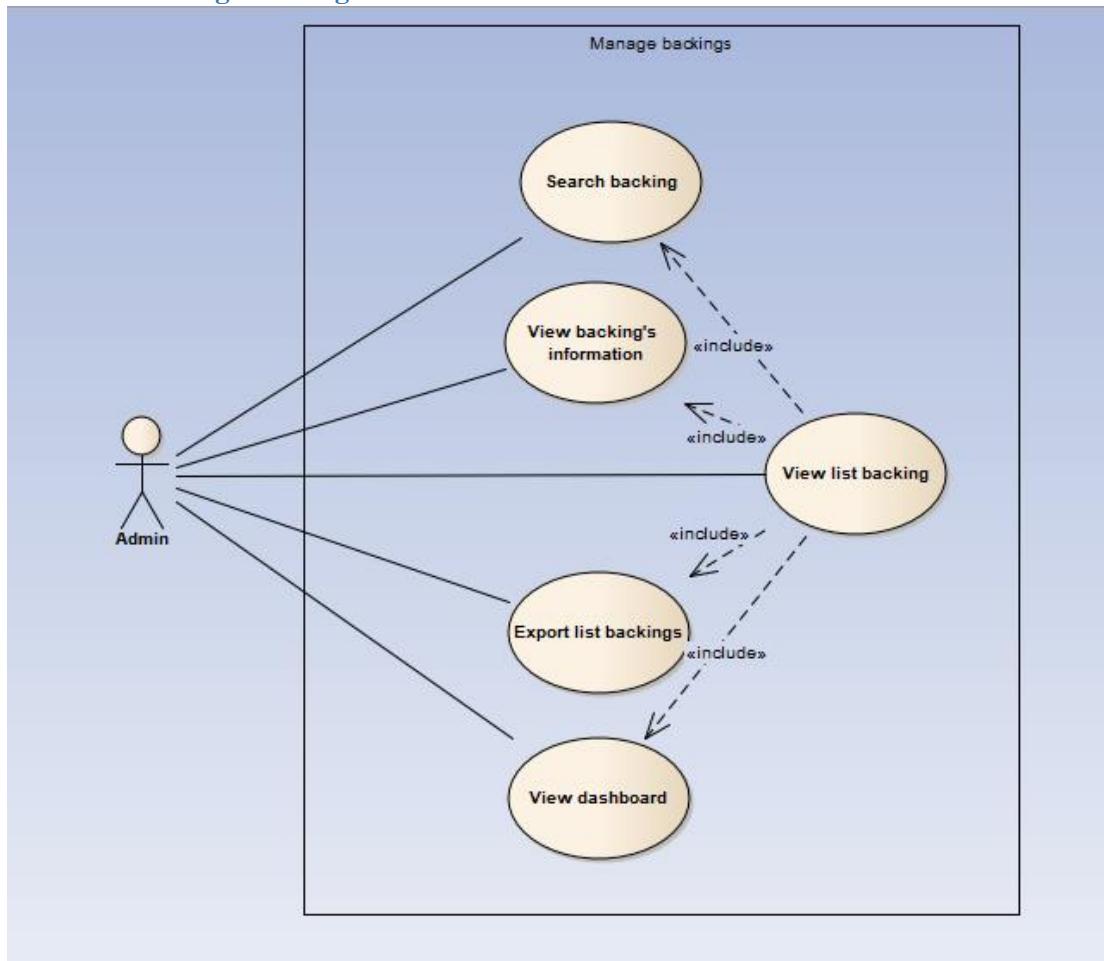
## 3.3.2.5.3.7 UC045- View project's dashboard

<b>Use Case ID</b>	UC045	<b>Version</b>	1.1
<b>Use Case Name</b>	View project's dashboard		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to view dashboard about project of the project		
<b>Preconditions:</b>	1. DDL website is available 2. Admin browsed DDL website 3. Admin logged in as Administrator		

<b>Trigger:</b>	Admin clicks on Dashboard	
<b>Post conditions:</b>	Dashboard about project is displayed	
<b>Normal Flow</b>		
Step	Actor	Action
1	Admin	<ul style="list-style-type: none"> <li>- Click on Project at left side</li> <li>- Click on Dashboard</li> </ul>
2	DDL	Display Project Dashboard page
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b> N/A		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

Table 3-47: View project's dashboard Use case

### 3.3.2.5.4 Manage backings



**Figure 3-24:** Manage Backing Use cases

#### 3.3.2.5.4.1 UC046- View backing list

Use Case ID	UC046	Version	1.1
Use Case Name	View backing list		
Created by:	MaiCTP	Date Created:	28/09/2015
Primary Actor	Administrator(Admin)	Secondary Actor	
Description:	This function allows Administrator to view Backing list of the project		
Preconditions:	1. DDL website is available 2. Admin browsed DDL website 3. Admin logged in as Administrator		
Trigger:	Admin clicks on Backing		
Post conditions:	Backing List is displayed		
Normal Flow			

<b>Step</b>	<b>Actor</b>	<b>Action</b>
1	Admin	Click on Backing at left side
2	DDL	Display list of backings
<b>Alternative Flows:</b>		
<b>AT1</b>	At step 1 in the main flows, if do not have any backing	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
2.1	DDL	Display "Do not have any backing" message
<b>Exceptions: N/A</b>		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

**Table 3-48:View backing list Use case****3.3.2.5.4.2 UC047- Search project**

<b>Use Case ID</b>	UC047	<b>Version</b>	1.1
<b>Use Case Name</b>	Search backing		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to search a backing		
<b>Preconditions:</b>	Admin is viewing Backing List		
<b>Trigger:</b>	Admin types name of backing on Search textbox		
<b>Post conditions:</b>	Display list of searched backing		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	Admin	- Type name of backing which want to search	
2	DDL	- Display list of backings which have the name is typed	
<b>Alternative Flows:</b>			

<b>AT1</b>	At step 1 in the main flows, if the typed name is not exist	
<b>Step</b>	<b>Actor</b>	<b>Action</b>
2.1	DDL	Display “No matching records found” message
<b>Exceptions: N/A</b>		
<b>Priority</b> High		
<b>Frequency of Use:</b> High		
<b>Business Rules:</b> <a href="#">B1</a>		
<b>Other Information:</b> N/A		

Table 3-49:Search backing Use case

## 3.3.2.5.4.3 UC048- View backing's information

<b>Use Case ID</b>	UC048	<b>Version</b>	1.1		
<b>Use Case Name</b>	View backing's information				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to view information of a backing				
<b>Preconditions:</b>	Admin is viewing Backing List				
<b>Trigger:</b>	Admin clicks on button at Action column				
<b>Post conditions:</b>	Information of this backing is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Admin clicks on button at Action column of the backing which want to view information			
2	DDL	Display Backing's information popup			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>		High			
<b>Frequency of Use:</b>		High			

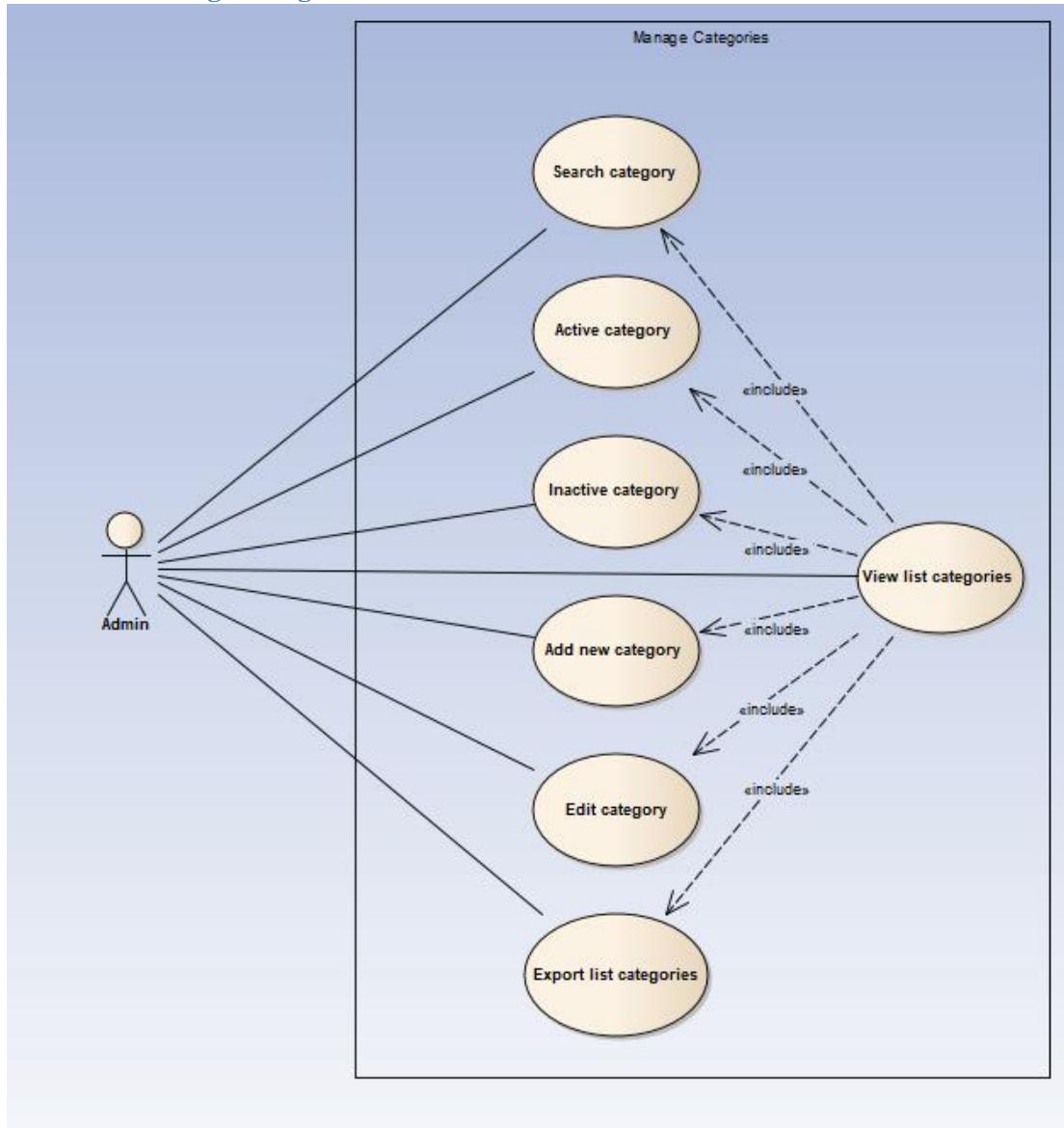
<b>Business Rules:</b>	<a href="#">B1</a>
<b>Other Information:</b>	N/A

**Table 3-50:View backing's information Use case****3.3.2.5.4.4 UC049- Export backings list**

<b>Use Case ID</b>	UC049	<b>Version</b>	1.1		
<b>Use Case Name</b>	Export backings list				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to export list of backings				
<b>Preconditions:</b>	Admin is viewing Backings List				
<b>Trigger:</b>	Admin clicks on Export button				
<b>Post conditions:</b>	List of backings is exported				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on Export button at right side Backings list table			
2	DDL	Export list of backings to file			
<b>Alternative Flows: N/A</b>					
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

**Table 3-51:Export backings list Use case**

### 3.3.2.5.5 Manage Categories



**Figure 3-25:** Manage categories use cases

#### 3.3.2.5.5.1 UC050- View category list

<b>Use Case ID</b>	UC050	<b>Version</b>	1.1
<b>Use Case Name</b>	View category list		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to view Category list of the project		
<b>Preconditions:</b>	1. DDL website is available 2. Admin browsed DDL website		

		3. Admin logged in as Administrator
<b>Trigger:</b>		Admin clicks on Category
<b>Post conditions:</b>		Category List is displayed
<b>Normal Flow</b>		
Step	Actor	Action
1	Admin	Click on Category at left side
2	DDL	Display list of categories
<b>Alternative Flows:</b>		
<b>AT1</b>	At step 1 in the main flows, if do not have any category	
Step	Actor	Action
2.1	DDL	Display “Do not have any category” message
<b>Exceptions: N/A</b>		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		<a href="#">B1</a>
<b>Other Information:</b>		N/A

Table 3-52: View category list Use case

## 3.3.2.5.5.2 UC051- Search category

<b>Use Case ID</b>	UC051	<b>Version</b>	1.1
<b>Use Case Name</b>	Search category		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to search a category		
<b>Preconditions:</b>	Admin is viewing Category List		
<b>Trigger:</b>	Admin types name of category at Search textbox		
<b>Post conditions:</b>	Display list of searched category		
<b>Normal Flow</b>			

<b>Step</b>	<b>Actor</b>	<b>Action</b>
1	Admin	Type name of category which want to search
2	DDL	Display list of categories which have the name is typed
<b>Alternative Flows:</b>		
<b>AT1</b>		At step 1 in the main flows, if the typed name is not exist
<b>Step</b>	<b>Actor</b>	<b>Action</b>
2.1	DDL	Display “No matching records found” message
<b>Exceptions: N/A</b>		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		<a href="#">B1</a>
<b>Other Information:</b>		N/A

**Table 3-53:Search category Use case****3.3.2.5.5.3 UC052- Activate category**

<b>Use Case ID</b>	UC052	<b>Version</b>	1.1
<b>Use Case Name</b>	Activate category		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to activate a category		
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>- Admin is viewing Category List</li> <li>- Status of this category is deactivate</li> </ul>		
<b>Trigger:</b>	Admin clicks on Deactivate button		
<b>Post conditions:</b>	Category is activated		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	Admin	Click on Deactivate button at Status column of the category which want to activate	
2	DDL	- Change displayed status from Deactivate to Activate	

		- Change status of this category to Activate on database
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b> N/A		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

**Table 3-54:Activate category Use case****3.3.2.5.5.4 UC053- Deactivate category**

<b>Use Case ID</b>	UC053	<b>Version</b>	1.1		
<b>Use Case Name</b>	Deactivate category				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to deactivate a category				
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>- Admin is viewing Category List</li> <li>- Status of this category is activate</li> </ul>				
<b>Trigger:</b>	Admin clicks on Activate button				
<b>Post conditions:</b>	Category is deactivated				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on Activate button at Status column of the category which want to deactivate			
2	DDL	<ul style="list-style-type: none"> <li>- Change displayed status from Activate to Deactivate</li> <li>- Change status of this category to Deactivate on database</li> </ul>			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				

<b>Other Information:</b>	N/A
---------------------------	-----

**Table 3-55:Deactivate category Use case****3.3.2.5.5.5 UC054- Add new category**

<b>Use Case ID</b>	UC054	<b>Version</b>	1.1		
<b>Use Case Name</b>	Add new category				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to add new a category				
<b>Preconditions:</b>	Admin is viewing Category List				
<b>Trigger:</b>	Admin clicks on Add button				
<b>Post conditions:</b>	New category is added				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on Add Category button at right side of Category list table			
2	DDL	Display Add new category popup			
3	Admin	<ul style="list-style-type: none"> <li>- Type Category Name</li> <li>- Type Description</li> <li>- Choose picture from Admin's computer</li> <li>- Click Add button</li> </ul>			
<b>Alternative Flows:</b>					
<b>AT1</b>	At step 3 in the main flows, if Admin click on Close button				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
3.1	DDL	Turn off Add new category popup			
<b>Exceptions:</b>					
<b>EC1</b>	At step 4 in main flow, if Admin does not type Category Name/ Description or does not choose picture				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
3.1	DDL	Display Add popup message with error message			
3.2	DDL	Mark error fields			
<b>Priority</b>		High			
<b>Frequency of Use:</b>		High			

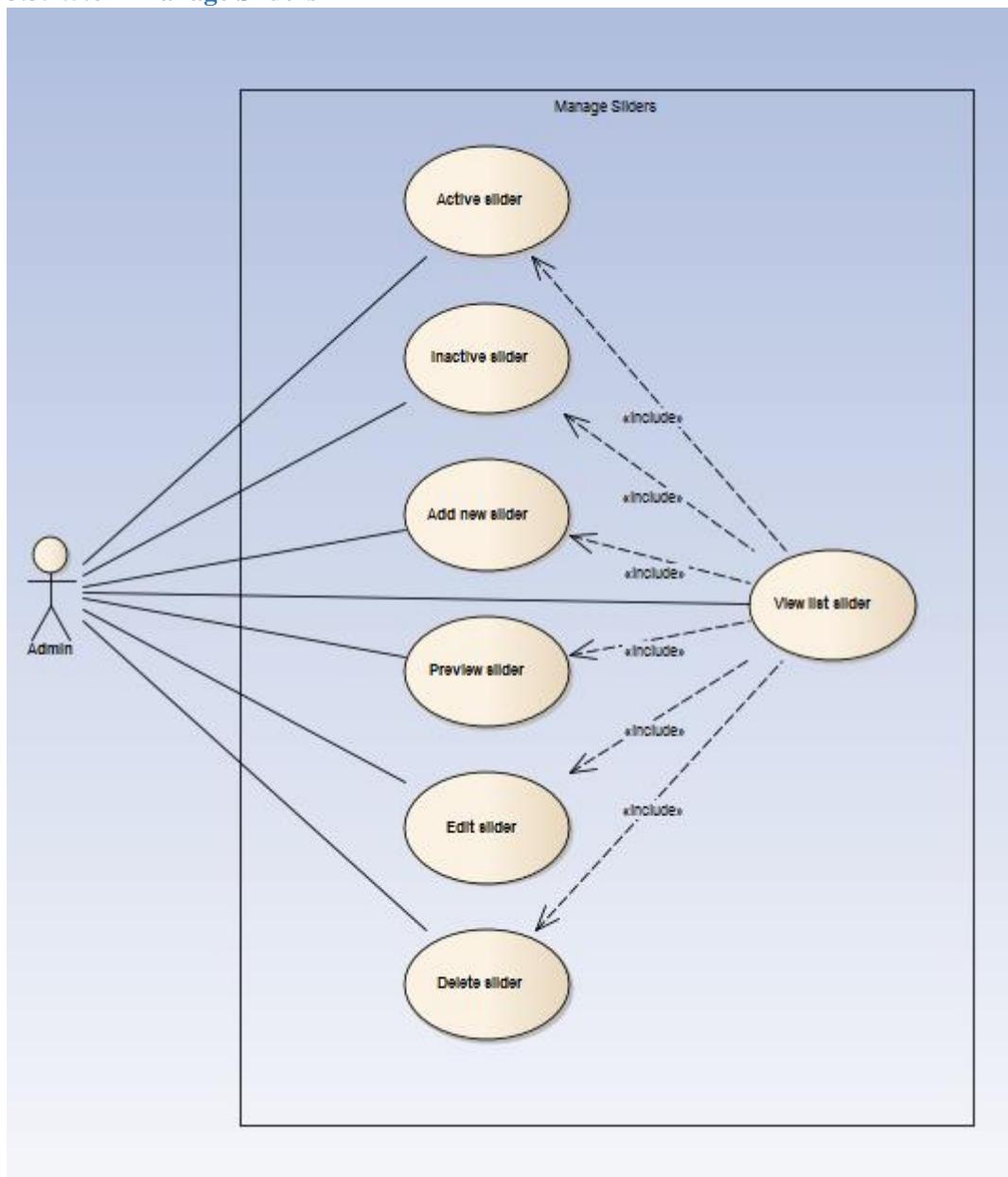
<b>Business Rules:</b>	<a href="#">B1</a>
<b>Other Information:</b>	N/A

**Table 3-56: Add new category Use case****3.3.2.5.5.6 UC055- Edit category**

<b>Use Case ID</b>	UC055	<b>Version</b>	1.1		
<b>Use Case Name</b>	Edit category				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to edit a category				
<b>Preconditions:</b>	Admin is viewing Category List				
<b>Trigger:</b>	Admin clicks on Edit button				
<b>Post conditions:</b>	Category is edited				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on Edit button at Edit column			
2	DDL	Display Edit category popup			
3	Admin	<ul style="list-style-type: none"> <li>- Type Category Name</li> <li>- Type Description</li> <li>- Choose picture from Admin's computer</li> <li>- Click Add button</li> </ul>			
<b>Alternative Flows:</b>					
<b>AT1</b>	At step 3 in the main flows, if Admin click on Close button				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
3.1	DDL	<ul style="list-style-type: none"> <li>- Turn off Edit category popup</li> <li>- Keep old information of this category</li> </ul>			
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

**Table 3-57: Edit category Use case**

### 3.3.2.5.6 Manage Sliders



**Figure 3-26:** Manage slides use cases

#### 3.3.2.5.6.1 UC056- View slider list

Use Case ID	UC056	Version	1.1
Use Case Name	View slider list		
Created by:	MaiCTP	Date Created:	28/09/2015
Primary Actor	Administrator(Admin)	Secondary Actor	
Description:	This function allows Administrator to view Slider list of the project		
Preconditions:	1. DDL website is available 2. Admin browsed DDL website		

	3. Admin logged in as Administrator	
<b>Trigger:</b>	Admin clicks on Slider	
<b>Post conditions:</b>	Slider List is displayed	
<b>Normal Flow</b>		
Step	Actor	Action
1	Admin	Click on Slider at left side
2	DDL	Display list of sliders
<b>Alternative Flows:</b>		
<b>AT1</b>	At step 1 in the main flows, if do not have any slider	
Step	Actor	Action
2.1	DDL	Display “Do not have any slider” message
<b>Exceptions: N/A</b>		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

**Table 3-58:View slider list Use case****3.3.2.5.6.2 UC057- Active slider**

<b>Use Case ID</b>	UC057	<b>Version</b>	1.1
<b>Use Case Name</b>	Active slider		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to activate a slider		
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>- Admin is viewing Slider List</li> <li>- Status of this slider is deactivate</li> </ul>		
<b>Trigger:</b>	Admin clicks on Deactivate button		
<b>Post conditions:</b>	Slider is activated		
<b>Normal Flow</b>			

Step	Actor	Action
1	Admin	Click on Deactivate button at Status column of the slider which want to activate
2	DDL	<ul style="list-style-type: none"> <li>- Change displayed status from Deactivate to Activate</li> <li>- Change status of this slider to Activate on database</li> </ul>
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b> N/A		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		<a href="#">B1</a>
<b>Other Information:</b>		N/A

**Table 3-59:Active slider Use case****3.3.2.5.6.3 UC058- Deactivate slider**

<b>Use Case ID</b>	UC058	<b>Version</b>	1.1		
<b>Use Case Name</b>	Deactivate slider				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to deactivate a slider				
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>- Admin is viewing Slider List</li> <li>- Status of this slider is activate</li> </ul>				
<b>Trigger:</b>	Admin clicks on Activate button				
<b>Post conditions:</b>	Slider is deactivated				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	Admin	Click on Activate button at Status column of the slider which want to deactivate			
2	DDL	<ul style="list-style-type: none"> <li>- Change displayed status from Activate to Deactivate</li> <li>- Change status of this slider to Deactivate on database</li> </ul>			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					

<b>Priority</b>	High
<b>Frequency of Use:</b>	High
<b>Business Rules:</b>	<a href="#">B1</a>
<b>Other Information:</b>	N/A

**Table 3-60:Deactivate slider Use case****3.3.2.5.6.4 UC059- Add new slider**

Use Case ID	UC059	Version	1.1		
<b>Use Case Name</b>	Add new slider				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to add new a slider				
<b>Preconditions:</b>	Admin is viewing Slider List				
<b>Trigger:</b>	Admin clicks on Add button				
<b>Post conditions:</b>	New slider is added				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	Admin	Click on Add Slider button at right side of Slider list table			
2	DDL	Display Add new slider popup			
3	Admin	<ul style="list-style-type: none"> <li>- Type Slider Title</li> <li>- Type Slider Description</li> <li>- Type Slider Url</li> <li>- Choose picture from Admin's computer</li> <li>- Choose Button Color</li> <li>- Choose Slider Order</li> <li>- Choose status</li> <li>- Click Add button</li> </ul>			
<b>Alternative Flows:</b>					
<b>AT1</b>	At step 3 in the main flows, if Admin click on Close button				
Step	Actor	<b>Action</b>			
3.1	DDL	Turn off Add new slider popup			
<b>Exceptions:</b>					

<b>EC1</b>	- At step 3 in main flow, if Admin does not type Slider Title/ Slider Description/ Slider Url or does not choose picture/ Button Color/ Slider Order/ status	
Step	Actor	Action
3.1	DDL	Display Add popup message with error message
3.2	DDL	Mark error fields
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<u>B1</u>	
<b>Other Information:</b>	N/A	

**Table 3-61: Add new slider Use case****3.3.2.5.6.5 UC060- Edit slider**

<b>Use Case ID</b>	UC060	<b>Version</b>	1.1		
<b>Use Case Name</b>	Edit slider				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to edit a slider				
<b>Preconditions:</b>	Admin is viewing Slider List				
<b>Trigger:</b>	Admin clicks on Edit button				
<b>Post conditions:</b>	Slider edited				
<b>Normal Flow</b>					
Step	Actor	<b>Action</b>			
1	Admin	Click on Edit button at Action column			
2	DDL	Display Edit slider popup			
3	Admin	Type all needed information			
<b>Alternative Flows:</b>					
<b>AT1</b>	At step 3 in the main flows, if Admin click on Close button				
Step	Actor	<b>Action</b>			
3.1	DDL	- Turn off Edit slider popup - Keep old information of this slider			
<b>Exceptions:</b> N/A					

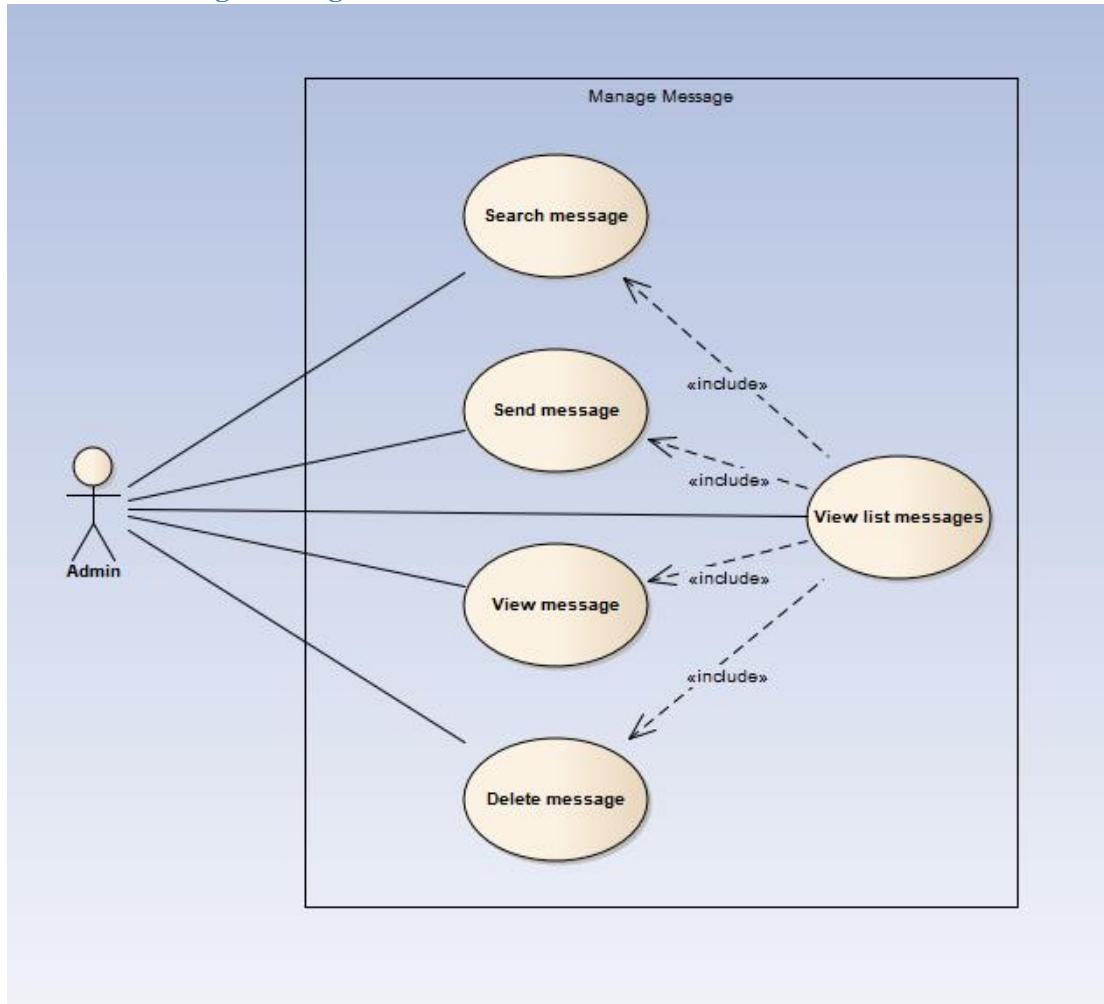
<b>Priority</b>	High
<b>Frequency of Use:</b>	High
<b>Business Rules:</b>	<a href="#">B1</a>
<b>Other Information:</b>	N/A

**Table 3-62:Edit slider Use case****3.3.2.5.6.6 UC061- Delete slider**

<b>Use Case ID</b>	UC061	<b>Version</b>	1.1		
<b>Use Case Name</b>	Delete slider				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to delete a slider				
<b>Preconditions:</b>	Admin is viewing Slider List				
<b>Trigger:</b>	Admin clicks on Delete button				
<b>Post conditions:</b>	Slider deleted				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on Delete button at Action column			
2	DDL	Display Deleted confirm popup			
3	Admin	Click on Delete button			
<b>Alternative Flows:</b>					
<b>AT1</b>	At step 3 in the main flows, if Admin click on Close button				
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
3.1	DDL	<ul style="list-style-type: none"> <li>- Turn off Delete slider popup</li> <li>- Keep all information of this slider</li> </ul>			
<b>Exceptions: N/A</b>					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

**Table 3-63:Delete slider Use case**

### 3.3.2.5.7 Manage Message



**Figure 3-27:** Manage message use case

#### 3.3.2.5.7.1 UC062- View message list

<b>Use Case ID</b>	UC062	<b>Version</b>	1.1
<b>Use Case Name</b>	View message list		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to view Message list of the project		
<b>Preconditions:</b>	1. DDL website is available 2. Admin browsed DDL website 3. Admin logged in as Administrator		
<b>Trigger:</b>	Admin clicks on Message		
<b>Post conditions:</b>	Message List is displayed		

<b>Normal Flow</b>		
<b>Step</b>	<b>Actor</b>	<b>Action</b>
1	Admin	Click on Message at left side
2	DDL	Display list of messages
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b> N/A		
<b>Priority</b>	High	
<b>Frequency of Use:</b>	High	
<b>Business Rules:</b>	<a href="#">B1</a>	
<b>Other Information:</b>	N/A	

**Table 3-64: View message list Use case****3.3.2.5.7.2 UC063- Search message**

<b>Use Case ID</b>	UC063	<b>Version</b>	1.1
<b>Use Case Name</b>	Search message		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to search a message		
<b>Preconditions:</b>	Admin is viewing Message List		
<b>Trigger:</b>	Admin clicks on Search button		
<b>Post conditions:</b>	Display list of searched message		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	Admin	<ul style="list-style-type: none"> <li>- Type name of category which want to search</li> <li>- Click on Search button</li> </ul>	
2	DDL	Display list of messages which have the name is typed	
<b>Alternative Flows:</b> N/A			
<b>Exceptions:</b>			
<b>AT1</b>	At step 1 in the main flows, if the typed name is not exist		
<b>Step</b>	<b>Actor</b>	<b>Action</b>	

2.1	DDL	Display “No matching records found” message
<b>Priority</b>	High	

**Table 3-65:Search message Use case****3.3.2.5.7.3 UC064- Send message**

<b>Use Case ID</b>	UC064		<b>Version</b>	1.1			
<b>Use Case Name</b>	Send message						
<b>Created by:</b>	MaiCTP		<b>Date Created:</b>	28/09/2015			
<b>Primary Actor</b>	Administrator(Admin)		<b>Secondary Actor</b>				
<b>Description:</b>	This function allows Administrator to send a message						
<b>Preconditions:</b>	Admin is viewing Message List						
<b>Trigger:</b>	Admin clicks on Send button						
<b>Post conditions:</b>	Message is send						
<b>Normal Flow</b>							
<b>Step</b>	<b>Actor</b>	<b>Action</b>					
1	Admin	Click on New message button					
2	DDL	Display Send Message popup					
3	Admin	<ul style="list-style-type: none"> <li>- Type all needed information at popup</li> <li>- Click on Send button</li> </ul>					
4	DDL	Send message to To email address					
<b>Alternative Flows:</b>							
<b>AT1</b>	At step 3 in the main flows, if Admin click on Close button						
<b>Step</b>	<b>Actor</b>	<b>Action</b>					
3.1	DDL	<ul style="list-style-type: none"> <li>- Close Send Message popup</li> </ul>					
<b>Exceptions: N/A</b>							
<b>Priority</b>	High						

**Table 3-66:Send message Use case**

## 3.3.2.5.7.4 UC065- View message

<b>Use Case ID</b>	UC065	<b>Version</b>	1.1		
<b>Use Case Name</b>	View message				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to view a message				
<b>Preconditions:</b>	Admin is viewing Message List				
<b>Trigger:</b>	Admin clicks on a message				
<b>Post conditions:</b>	Message is viewed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Click on a message			
2	DDL	Display View Message popup			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>	High				

Table 3-67: View message Use case

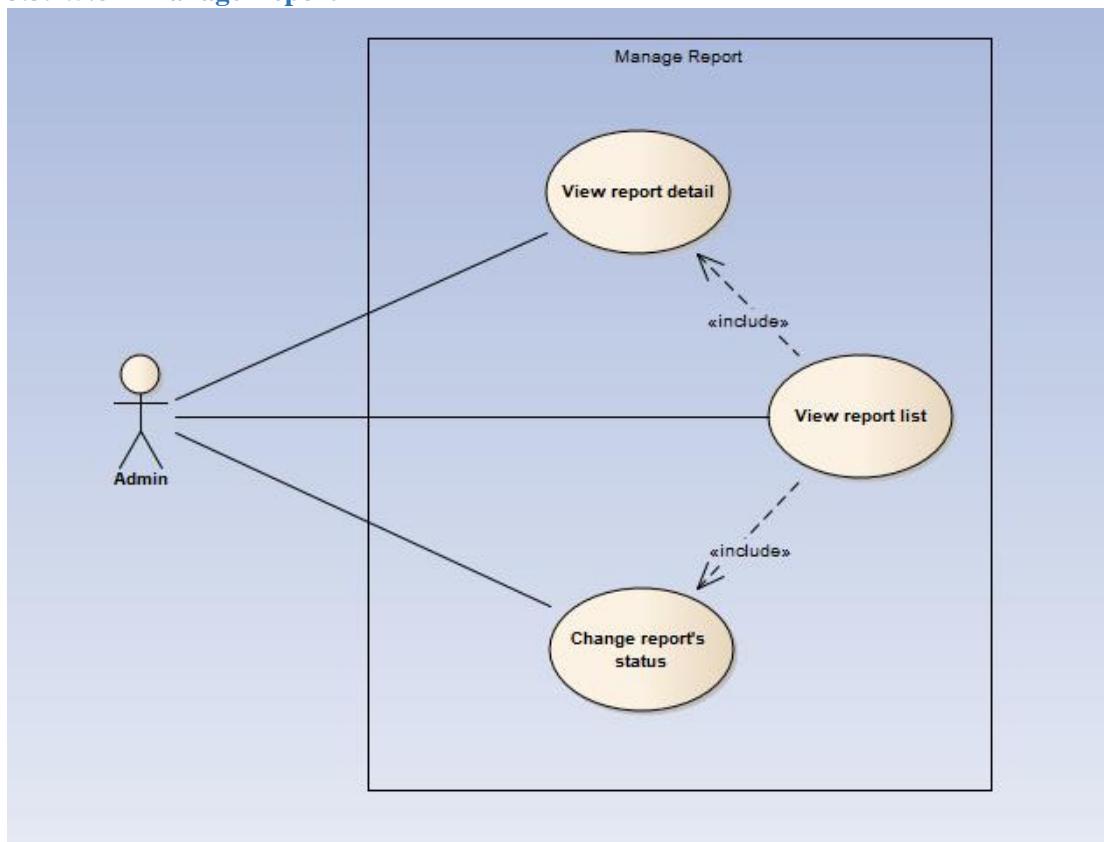
## 3.3.2.5.7.5 UC066- Delete message

<b>Use Case ID</b>	UC066	<b>Version</b>	1.1
<b>Use Case Name</b>	Delete message		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to delete a message		
<b>Preconditions:</b>	Admin is viewing Message List		
<b>Trigger:</b>	Admin clicks on Delete button		
<b>Post conditions:</b>	Message deleted		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	Admin	Click on checkbox in front of message which want to delete	

2	DDL	Display Deleted confirm popup
3	Admin	Click on Delete button
<b>Alternative Flows:</b>		
<b>AT1</b>		At step 3 in the main flows, if Admin click on Close button
Step	Actor	Action
3.1	DDL	<ul style="list-style-type: none"> <li>- Turn off Delete message popup</li> <li>- Keep all information of this message</li> </ul>
<b>Exceptions: N/A</b>		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		<a href="#">B1</a>
<b>Other Information:</b>		N/A

**Table 3-68:Delete message Use case**

### 3.3.2.5.8 Manage Report



**Figure 3-28:** Manage report use cases

#### 3.3.2.5.8.1 UC067- View report list

<b>Use Case ID</b>	UC067	<b>Version</b>	1.1
<b>Use Case Name</b>	View report list		
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>	
<b>Description:</b>	This function allows Administrator to view Report list of the project		
<b>Preconditions:</b>	1. DDL website is available 2. Admin browsed DDL website 3. Admin logged in as Administrator		
<b>Trigger:</b>	Admin clicks on User/ Project		
<b>Post conditions:</b>	Report List is displayed		
<b>Normal Flow</b>			
<b>Step</b>	<b>Actor</b>	<b>Action</b>	
1	Admin	- Click on Report at left side	

		- Click on User/ Project
2	DDL	Display list of reports
<b>Alternative Flows:</b> N/A		
<b>Exceptions:</b> N/A		
<b>Priority</b>		High
<b>Frequency of Use:</b>		High
<b>Business Rules:</b>		<a href="#">B1</a>
<b>Other Information:</b>		N/A

**Table 3-69: View report list Use case****3.3.2.5.8.2 UC068- View report detail**

<b>Use Case ID</b>	UC068	<b>Version</b>	1.1		
<b>Use Case Name</b>	View report detail				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to view information of a report				
<b>Preconditions:</b>	Admin is viewing Report List				
<b>Trigger:</b>	Admin clicks on button View at Action column				
<b>Post conditions:</b>	Detail of this report is displayed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Admin clicks on button View at Action column of the user/ project which want to view report's detail information			
2	DDL	Display View Report Content popup			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>		High			
<b>Frequency of Use:</b>		High			
<b>Business Rules:</b>		<a href="#">B1</a>			

<b>Other Information:</b>	N/A
---------------------------	-----

**Table 3-70:View report detail Use case****3.3.2.5.8.3 UC069- Change report's status**

<b>Use Case ID</b>	UC069	<b>Version</b>	1.1		
<b>Use Case Name</b>	Change report's status				
<b>Created by:</b>	MaiCTP	<b>Date Created:</b>	28/09/2015		
<b>Primary Actor</b>	Administrator(Admin)	<b>Secondary Actor</b>			
<b>Description:</b>	This function allows Administrator to change status of a report				
<b>Preconditions:</b>	Admin is viewing Report List				
<b>Trigger:</b>	Admin clicks on button at Status column				
<b>Post conditions:</b>	Status of this report is changed				
<b>Normal Flow</b>					
<b>Step</b>	<b>Actor</b>	<b>Action</b>			
1	Admin	Admin clicks on button Waiting/ Cancel/ Confirmed at Status column of the user/ project which want to change report's status			
2	DDL	<ul style="list-style-type: none"> <li>- Display changed status as content of the button</li> <li>- Change status of report in database</li> </ul>			
<b>Alternative Flows:</b> N/A					
<b>Exceptions:</b> N/A					
<b>Priority</b>	High				
<b>Frequency of Use:</b>	High				
<b>Business Rules:</b>	<a href="#">B1</a>				
<b>Other Information:</b>	N/A				

**Table 3-71:Change report's status Use case**

### 3.3.3 Non-Functional Specification

#### 3.3.3.1 Reliability

- Mean time between failures: The mean time between each failure is expected to be about 480 hours of using.
- 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.
- The database must be backed up daily and can be recovered if necessary

#### 3.3.3.2 Performance Requirements

##### 3.3.3.2.1 Availability

The application must be available 95% of time. Users can access to it everywhere from their Web browser with internet connection.

##### 3.3.3.2.2 Response time

In term of response time, waiting time of processing will be 2 to 5 seconds; time to process any functions will not exceed 7 seconds.

##### 3.3.3.2.3 Maintainability

- **Coding standards and naming conventions:**
  - Output of the project 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.

##### 3.3.3.2.4 Usability

- Usability Requirements support the following from the perspective of its primary users:
  - Efficiency of use: user can complete each function in less than 12 actions
  - Intuitiveness: User can understand approximately 90% help/ error messages

##### 3.3.3.2.5 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.

##### 3.3.3.2.6 Hardware/Software Requirements

- **Hardware requirement:**

- Server: Azure server, IIS8 or higher
- Client: PC/Laptop (window 7, window 8, window 10)
- **Software requirements:**
  - Web browser (Chrome 40.0, Firefox 30.0) with internet connection

## 4 SOFTWARE DESIGN

### 4.1 Introduction

#### 4.1.1 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.

#### 4.1.2 Definitions, Acronyms, Abbreviations

Acronym	Definition	Note
DDL	Dandelion	
DB	Database	
MVC	Model view control	
IDE	Integrated development environment	
Q&A	Question and answer	
GUI	Graphic user interface	
FU	FPT University	

Table 4-1: Definitions and Acronyms

### 4.2 Architecture design

#### 4.2.1 Choice of Architecture Design

##### 4.2.1.1 MVC Model

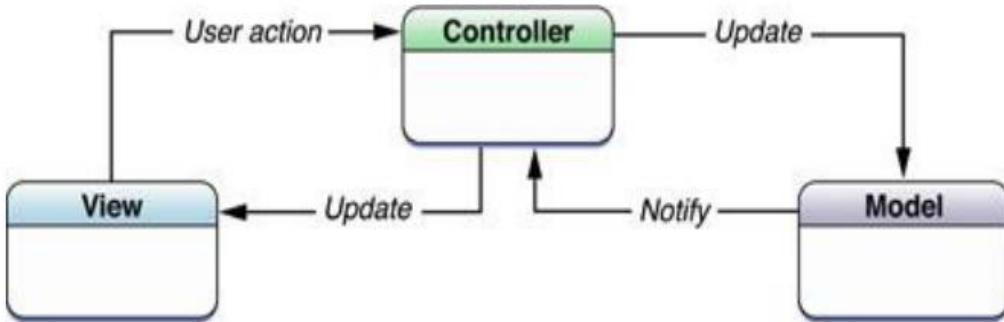
The purpose of DDL is developing as a crowdfunding website, where people can start their ideas and make it real. The system of DDL is structured based on MVC combined with layered architecture.

###### 4.2.1.1.1 MVC Model Overview

The **model-view-controller** or **MVC** is software architecture commonly used for creating web applications or software. In other words, it's a structure for web applications to follow in order to ensure efficiency and consistency. Many of the most popular frameworks use the MVC architecture, including ASP.NET, CodeIgniter, Zend, Django, and Ruby on Rails. At the same time, there are many web developers who don't use a coding framework yet still set up their applications to follow the MVC structure.

The Model-View-Controller (MVC) design pattern assigns objects in an application one of three roles: model, view, or controller. The pattern defines not only the roles objects play in the application, it defines the way objects communicate with each other. Each of the three types of objects is separated from the others by abstract boundaries and communicates with

objects of the other types across those boundaries. The collection of objects of a certain MVC type in an application is sometimes referred to as a layer—for example, model layer.



**Figure 4-1: MVC Model**

In addition to dividing the application into three kinds of components, the MVC design defines the interactions between them:

- **Controller:** The controller can be considered the "middle man" of the application. It works with the user, taking in data, and then working with the model to get the appropriate data or calculation, and then working with the view to show the response to the user.
- **Model:** A model is simply a representation of something we need to deal within our application. It is a "model" for something we must represent in code, such as a book, user, bank account, or whatever. The model is responsible for holding the functions and variables that are involved with whatever it's representing. You can think of a model's logic as the core concept to object oriented programming — models are just our "classes". However, don't let this confuse you as controllers are technically structured as classes as well.
- **View:** Finally, after the controller requests information from the model it sends it to a view. A view is just like the application's templating system — there might be a view for a certain type of page layout (*profile page*), a mobile view, or a view for a particular theme/skin. A view will contain all of the markup, CSS, and etc. that you traditionally use with creating a static web page.

#### 4.2.1.1.2 Advantages and disadvantages of MVC Model

- Advantages:
  - MVC separates system into components, which can be developed, maintained and upgraded individually without pausing system.
  - Develop tools is useful and easy to use.
  - Large of documentary sources.
- Disadvantages:
  - For small projects that apply MVC model caused cumbersome, time consuming in development process.
  - Time consuming to transits data between components.
  - Not suitable for agent-oriented applications such as interactive mobile and robotics applications.

- Multiple pairs of controllers and views based on the same data model make data model change expensive.
- The division between the View and the Controller is not clear in some cases.

#### **4.2.1.1.3 The reasons for choosing MVC Model**

- MVC makes parts of system can be developed individually and simultaneously to reduce developing time.
- Better support for test-driven development.
- Tools is useful and documentary source is large makes MVC is easy to develop.
- DDL system is not complete system, now. We built the system that towards extensibility and maintainability in the future.

#### **4.2.1.2 .Net Framework**

##### **4.2.1.2.1 .Net Framework Overview**

.NET Framework is a software framework developed by Microsoft that runs primarily on Microsoft Windows. It includes a large class library known as Framework Class Library (FCL) and provides language interoperability (each language can use code written in other languages) across several programming languages.

The .NET Framework is a technology that supports building and running the next generation of applications and XML Web services. The .NET Framework is designed to fulfill the following objectives:

- To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
- To provide a code-execution environment that minimizes software deployment and versioning conflicts.
- To provide a code-execution environment that promotes safe execution of code, including code created by an unknown or semi-trusted third party.
- To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
- To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.
- To build all communication on industry standards to ensure that code based on the .NET Framework can integrate with any other code.

##### **4.2.1.2.2 Advantages and disadvantages of .Net Framework**

- Advantages
  - Less time to produce product
  - Less Complexity.
  - Easily to access complex O.S functions
  - Easily to build Data Oriented Project, support huge DB functions.
  - Managed
  - Support Both Windows and Web Application.
  - Easy to create Dynamic sites.

- Disadvantages
  - Not suitable for High End Application
  - Low performance compare to C, C++.
  - Unavailability of build in methods.
  - .NET framework is free to download but Code Editor is costly.
  - Only few O.S supports .NET.

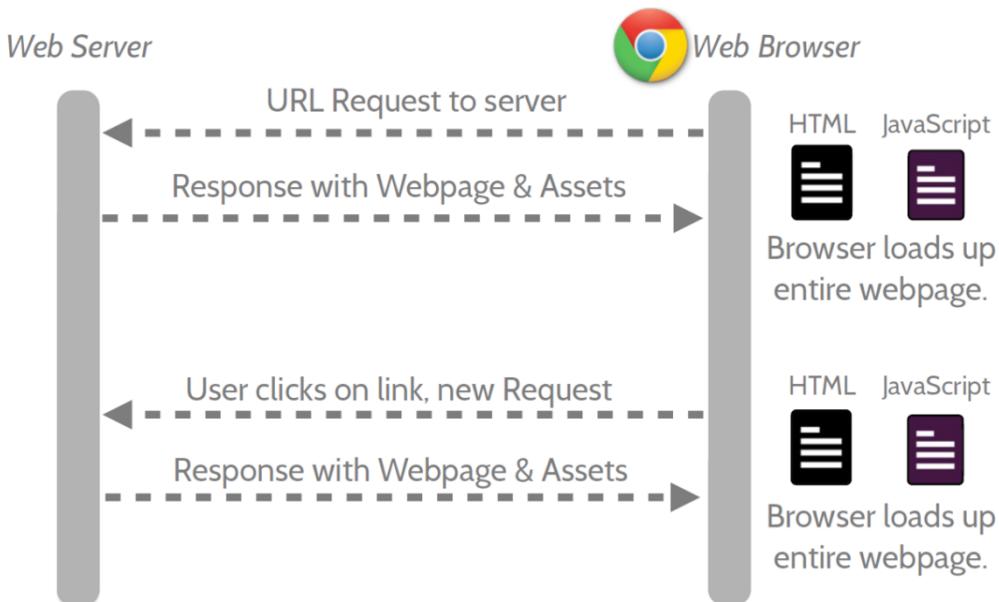
#### 4.2.1.2.3 The reasons for choosing .Net Framework

- Consistent with DDL system.
- There are many plugins and resources which support creating a website using MVC model.
- Many members can use and have experiences using C# (a language of .Net Framework)

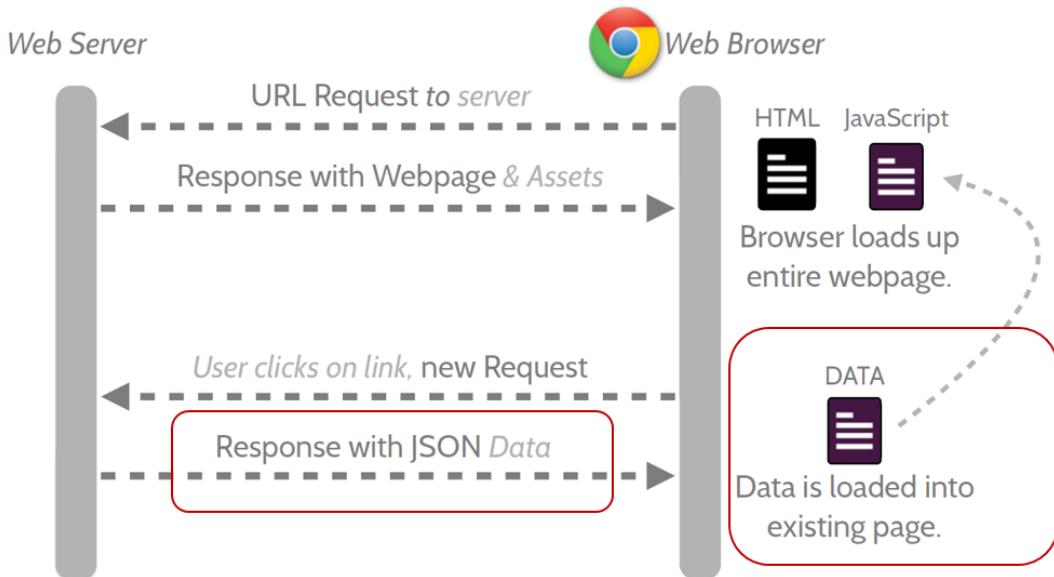
#### 4.2.1.3 AngularJS

##### 4.2.1.3.1 AngularJS Overview

AngularJS is a structural framework for dynamic web apps. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application's components clearly and succinctly. Angular's data binding and dependency injection eliminate much of the code you would otherwise have to write. And it all happens within the browser, making it an ideal partner with any server technology.



**Figure 4-1:** Multi Page Application



**Figure 4-2:** Single Page Application

❖ Features:

- AngularJS is a powerful JavaScript based development framework to create RICH Internet Application (RIA).
- AngularJS provides developers options to write client side application (using JavaScript) in a clean MVC (Model View Controller) way.
- Application written in AngularJS is cross-browser compliant. AngularJS automatically handles JavaScript code suitable for each browser.
- AngularJS is open source, completely free, and used by thousands of developers around the world. It is licensed under the Apache License version 2.0.

Overall, AngularJS is a framework to build large scale and high performance web application while keeping them as easy-to-maintain.

❖ Core futures: following are most important core features of AngularJS:

- **Data-binding:** It is the automatic synchronization of data between model and view components.
- **Scope:** These are objects that refer to the model. They act as a glue between controller and view.
- **Controller:** These are JavaScript functions that are bound to a particular scope.
- **Services:** AngularJS come with several built-in services for example \$http to make XMLHttpRequests. These are singleton objects which are instantiated only once in app.
- **Filters:** These select a subset of items from an array and returns a new array.

- **Directives:** Directives are markers on DOM elements (such as elements, attributes, css, and more). These can be used to create custom HTML tags that serve as new, custom widgets. AngularJS has built-in directives (ngBind, ngModel...)
- **Templates:** These are the rendered view with information from the controller and model. These can be a single file (like index.html) or multiple views in one page using "partials".
- **Routing:** It is concept of switching views.
- **Model View Whatever:** MVC is a design pattern for dividing an application into different parts (called Model, View and Controller), each with distinct responsibilities. AngularJS does not implement MVC in the traditional sense, but rather something closer to MVVM (Model-View-ViewModel). The Angular JS team refers it humorously as Model View Whatever.
- **Deep Linking:** Deep linking allows you to encode the state of application in the URL so that it can be bookmarked. The application can then be restored from the URL to the same state.
- **Dependency Injection:** AngularJS has a built-in dependency injection subsystem that helps the developer by making the application easier to develop, understand, and test.

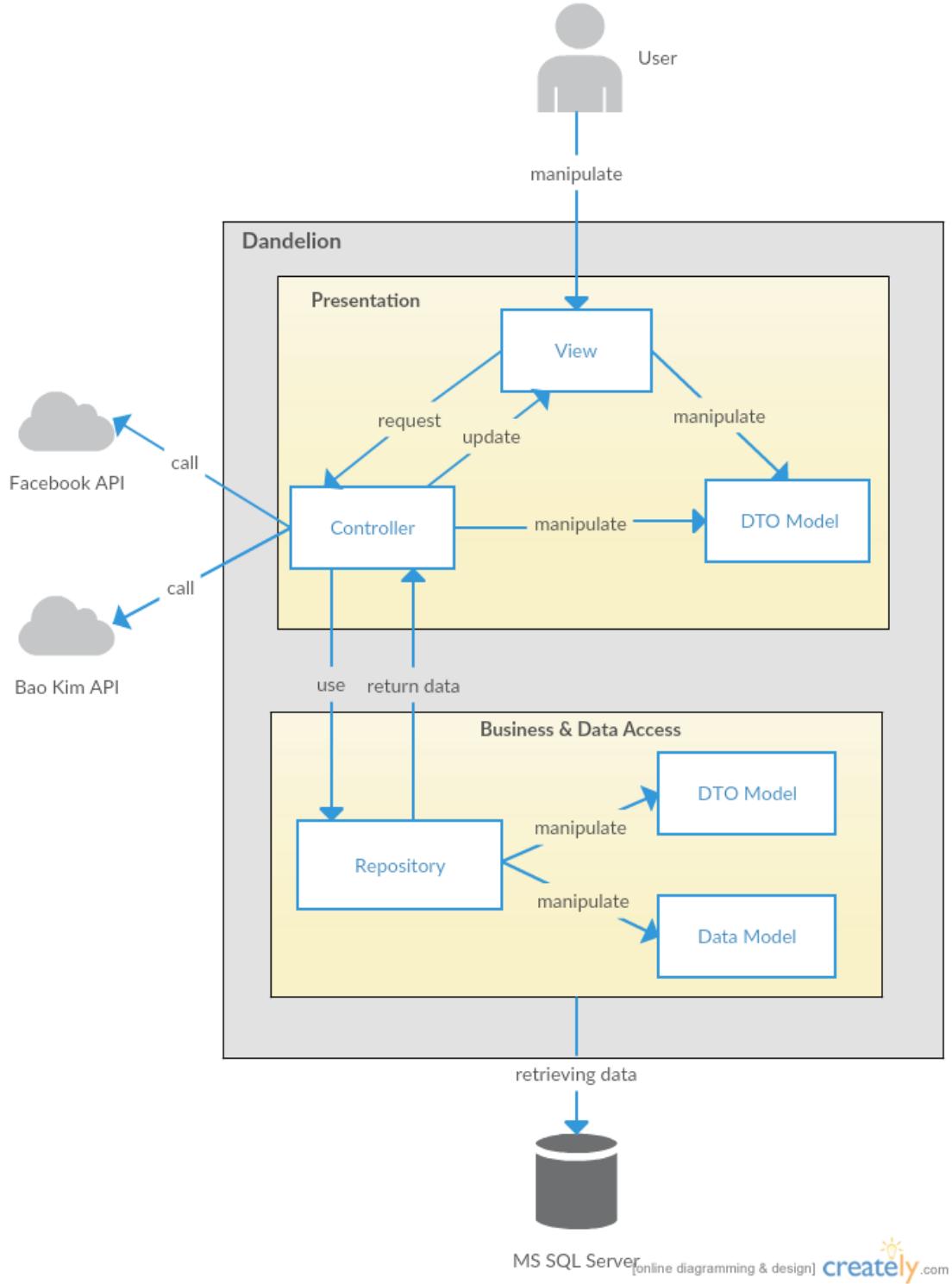
#### 4.2.1.3.2 Advantages and disadvantages of AngularJS

- Advantages:
  - AngularJS provides capability to create Single Page Application in a very clean and maintainable way.
  - AngularJS provides data binding capability to HTML thus giving user a rich and responsive experience
  - AngularJS code is unit testable.
  - AngularJS uses dependency injection and make use of separation of concerns.
  - AngularJS provides reusable components.
  - With AngularJS, developer write less code and get more functionality.
  - In AngularJS, views are pure html pages, and controllers written in JavaScript do the business processing.
- Disadvantages:
  - **Not Secure:** Being JavaScript only framework, application written in AngularJS are not safe. Server side authentication and authorization is must to keep an application secure.
  - **Not degradable:** If your application user disables JavaScript then user will just see the basic page and nothing more.

#### 4.2.1.3.3 The reason for choosing AngularJS

- AngularJS supports to create a website faster via RestApi and MVC Model.
- Provide a more fluid user experience akin to a desktop application.
- All team members want to study a new language.

#### 4.2.2 Architecture Representation



**Figure 4-2: System Overview**

We follow MVC architecture to implement the DDL Project. MVC offers architectural benefits over standard Jquery and AngularJS — it helps us write better-organized and therefore more maintainable code.

- **DTO Model (data transfer object model)** is an object that defines how the data will be sent over the network to remove circular references from data model, Hide particular properties that clients are not supposed to view, omit some properties in order to reduce payload size, flatten object graphs that contain nested objects, to make them more convenient for clients, avoid “over-posting” vulnerabilities and decouple your service layer from your database layer.
- **View** is what is presented to the users and how users interact with the system. The view is expected to render the model in a meaningful way to the user. In DDL, the view is made with .cshtml file including css, AngularJS and jQuery, it sends user gestures to controller and allows controller to select view.
- **Controller** is the decision maker and the glue between the model and view; it handles user actions and gestures, and responds to user events. For example, in CMS, when a user clicks the “Create” button to create a new contract, the controller for that action is invoked. The controller will then make changes to the contract model. The view will then render the modified contract model to the display so that user can view the new contract he added in the contract list.
- **Data Model** is where the application’s data objects are stored. A model object is in charge of encapsulating application state and one object could be related to other objects establishing a one-to-one or one-to-many relationship.
- **Repository** is intermediate layer which used to separate the controller and the data access layer (database context). It queries the data source for data, maps it to DTO models, processes data and returns data to controller.

#### 4.2.3 Architectural Goals and Constraints

❖ **Availability:**

- The application must be available 95% of time. Users can access to it everywhere from there .Web browser with internet connection.

❖ **Maintainability:**

- Coding standards and naming conventions:
  - Output of the project must include coding standards and naming conventions documentations. Implementation code must be easy to maintain.
  - All code must be clearly commented, including class, method documentations.
  - 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 changes 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.
- ❖ Usability:
  - Intuitiveness: all help/error messages are simple to understand; user can know exactly how to do each feature after one time using it.
- ❖ Capacity and scalability:
  - Throughput, storage and growth requirements.

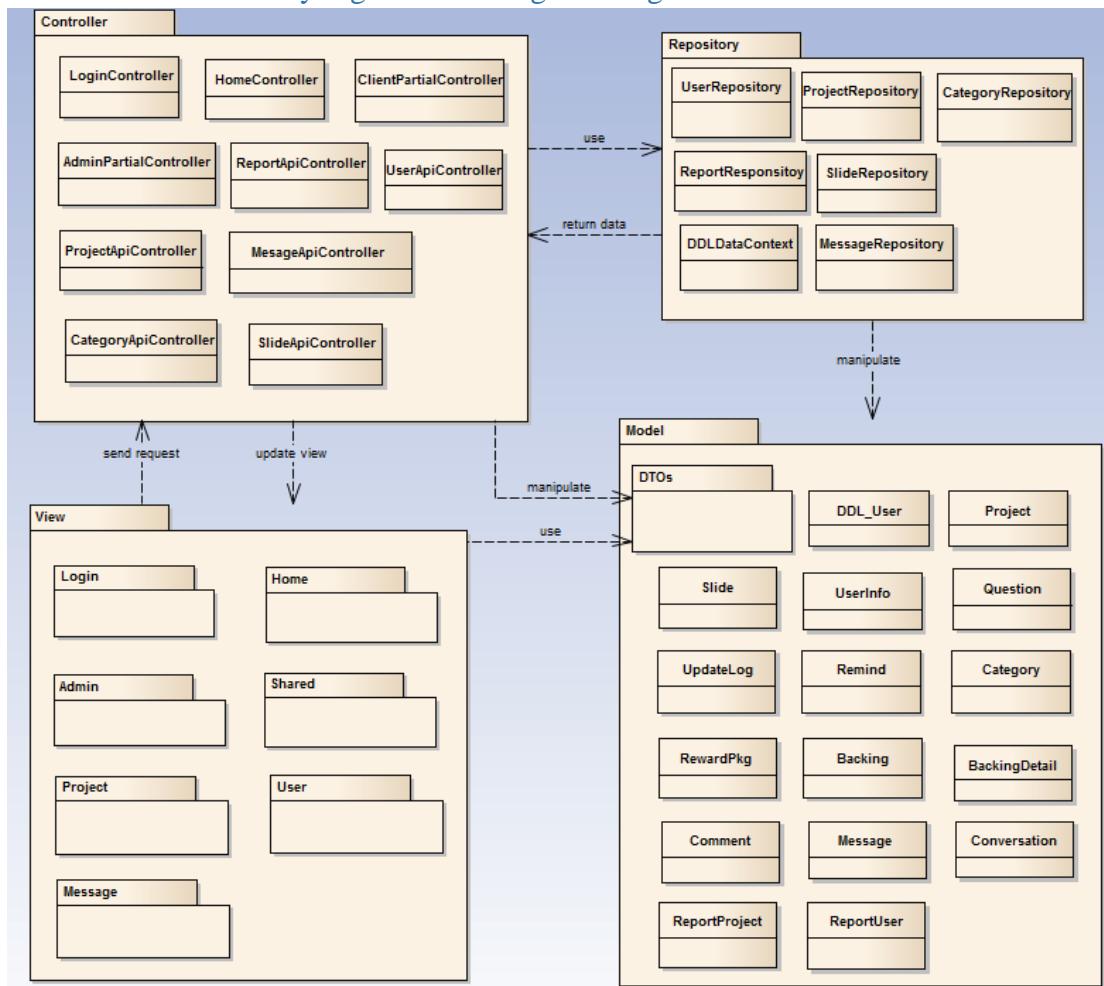
#### 4.2.4 Logical View

##### 4.2.4.1 Overview

Logical View includes Package diagram and Class diagram. Package diagram describes the organization of packages and elements. Class Diagram provides an overview of the target system by describing the objects and classes inside the system and the relationships between them. It provides a wide variety of usages; from modeling the domain-specific data structure to detailed design of the target system

- Controller contain the interface between:
  - Associated models
  - Associated views
  - The input devices (e.g., keyboard, pointing device, time).
  - Send commands to the model to update the model's state.
- Model is:
  - the domain-specific software simulation
  - Or implementation of the application's central structure.
- View: deal with everything graphical:
  - Requests data from their model
  - Display the data.
- Repository:
  - Create queries to DB.
  - Process data.
  - Return to controller.

#### 4.2.4.2 Architecturally Significant Design Packages



**Figure 4-3: Package Diagram**

❖ **Model:**

No	Model class	Role
1	DDL_User	Description entity of User in database
2	Project	Description entity of Project in database
3	Conversation	Description entity of Conversation in database
4	UpdateLog	Description entity of UpdateLog in database
5	Remind	Description entity of Remind in database
6	Category	Description entity of Category in database
7	Reward	Description entity of Reward in database
8	Backing	Description entity of Backing in database

9	BackingDetail	Description entity of BackingDetail in database
10	Comment	Description entity of Comment in database
11	QuestionAnswer	Description entity of QuestionAnswer in database
12	Message	Description entity of Message in database
13	ReportProject	Description entity of ReportProject in database
14	ReportUser	Description entity of ReportUser in database
15	Slide	Description entity of Slide in database
16	Userinfo	Description entity of Userinfo in database

**Table 4-2: Model Package**

❖ **Controller:**

No	Controller class	Role
1	LoginController	Receive request login, logout from client. Call method login, logout. Respond login view and login, logout status.
2	HomeController	Receive request to home page from client. Handle request from client and call methods to get all data of home page. Respond data back to Home View.
3	ClientParitalController	Receive request get client partial views of from client. Generate partial views and respond to client.
4	AdminParitalController	Receive request get admin partial views of from client. Generate partial views and respond to client.
5	UserApiController	Receive requests (get, post, push, delete) about User from client. Respond status, message and json data to client.
6	ProjectApiController	Receive requests (get, post, push, delete) about Project from client. Respond status, message and json data to client.
7	MessgaeApiController	Receive requests (get, post, push, delete) about Message from client. Respond status, message and json data to client.

8	SliderApiController	Receive requests (get, post, push, delete) about Slider from client. Respond status, message and json data to client.
9	ReportApiController	Receive requests (get, post, push, delete) about Report User or Project from client. Respond status, message and json data to client.
10	CategoryApiController	Receive requests (get, post, push, delete) about Category from client. Respond status, message and json data to client.

**Table 4-3: Controller Package**

❖ **Repository:**

No	Repository class	Role
1	DDLDDataContext	Connect to DB, create queries to select, update, deletes data.
2	UserRepository	Process user data and return to controller.
3	SlideRepository	Process slide data and return to controller.
4	ReportResponsitoy	Process report data and return to controller.
5	ProjectRepository	Process project data and return to controller.
6	MessageRepository	Process message data and return to controller.
7	CategoryRepository	Process category data and return to controller.

**Table 4-4: Repository Package**

❖ **DTOs:**

Include many data transfer object classes.

❖ **View:**

Include many .cshtml file

#### 4.2.5 Process view

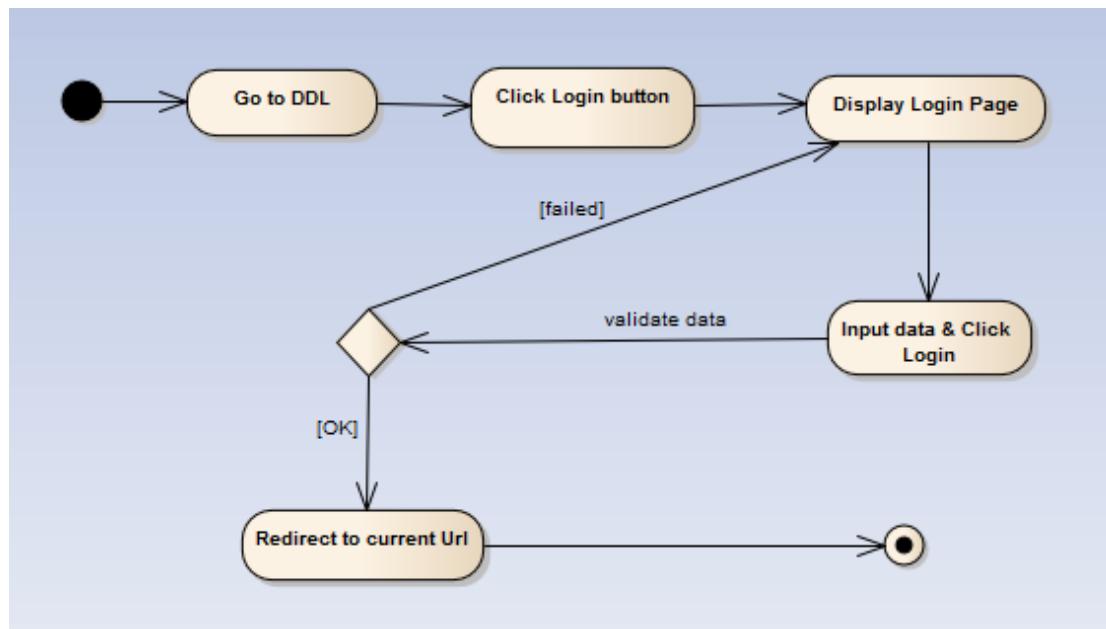


Figure 4-4: Login activity diagram

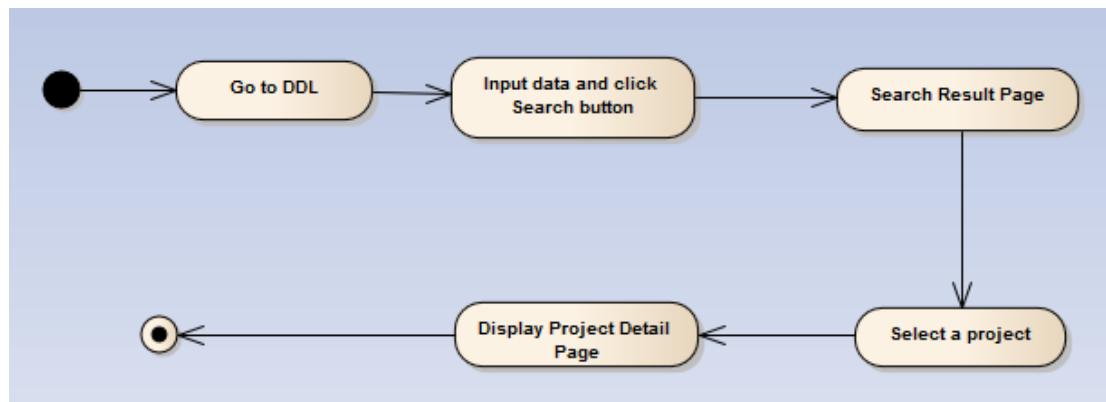


Figure 4-5: Search Project activity diagram

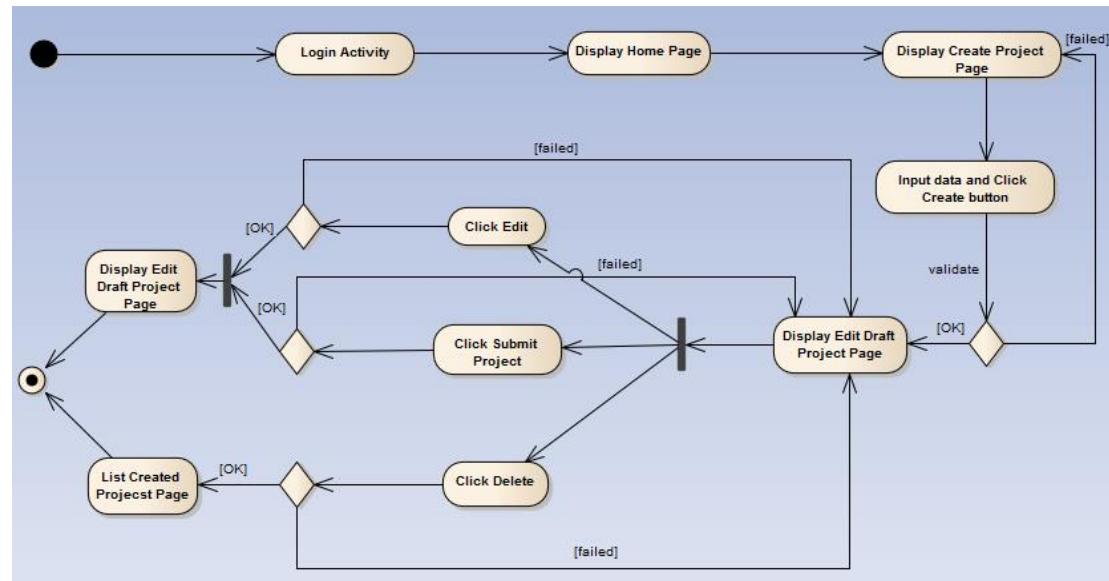


Figure 4-6: Create Project activity diagram

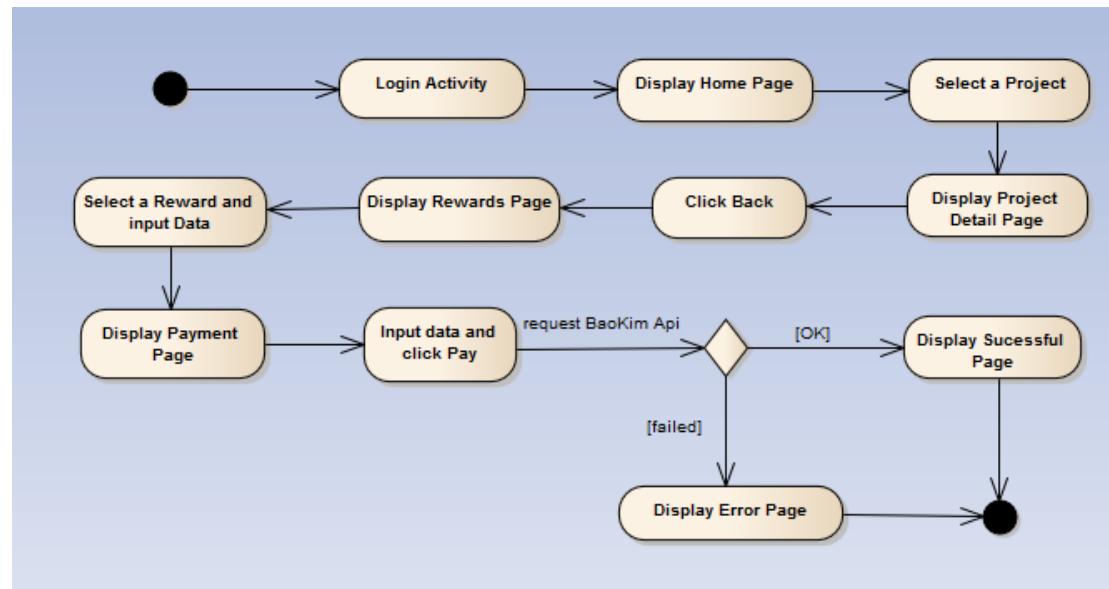


Figure 4-7: Back Project activity diagram

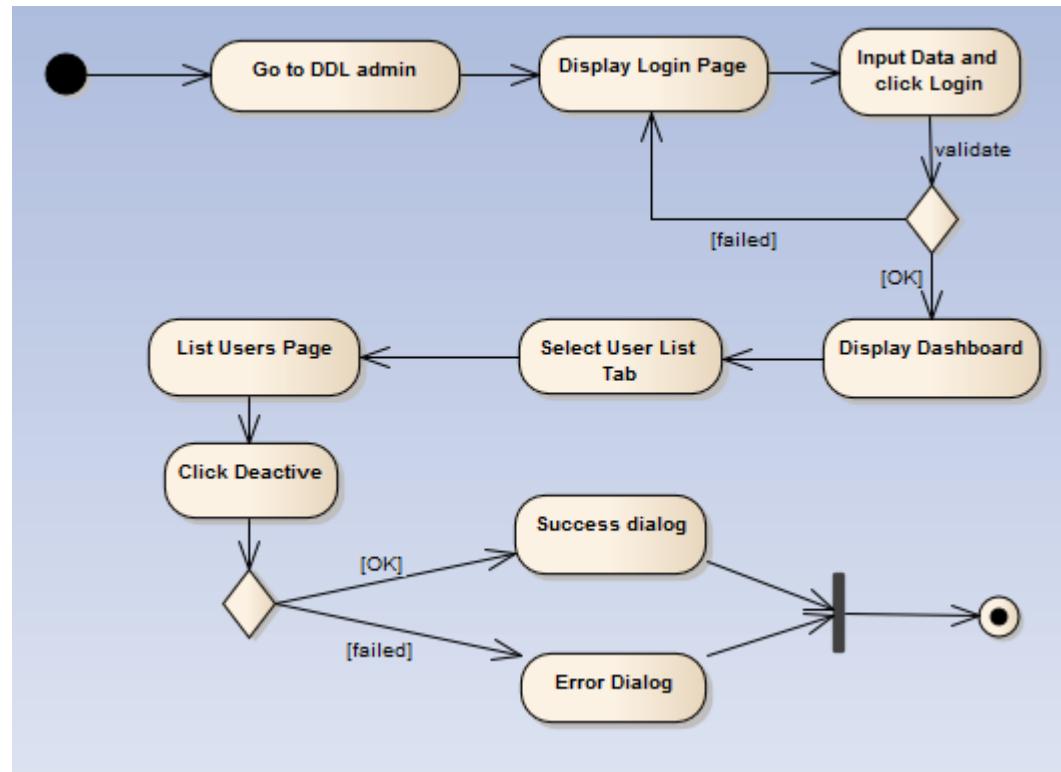


Figure 4-8: De-active User activity diagram

#### 4.2.6 Deployment View

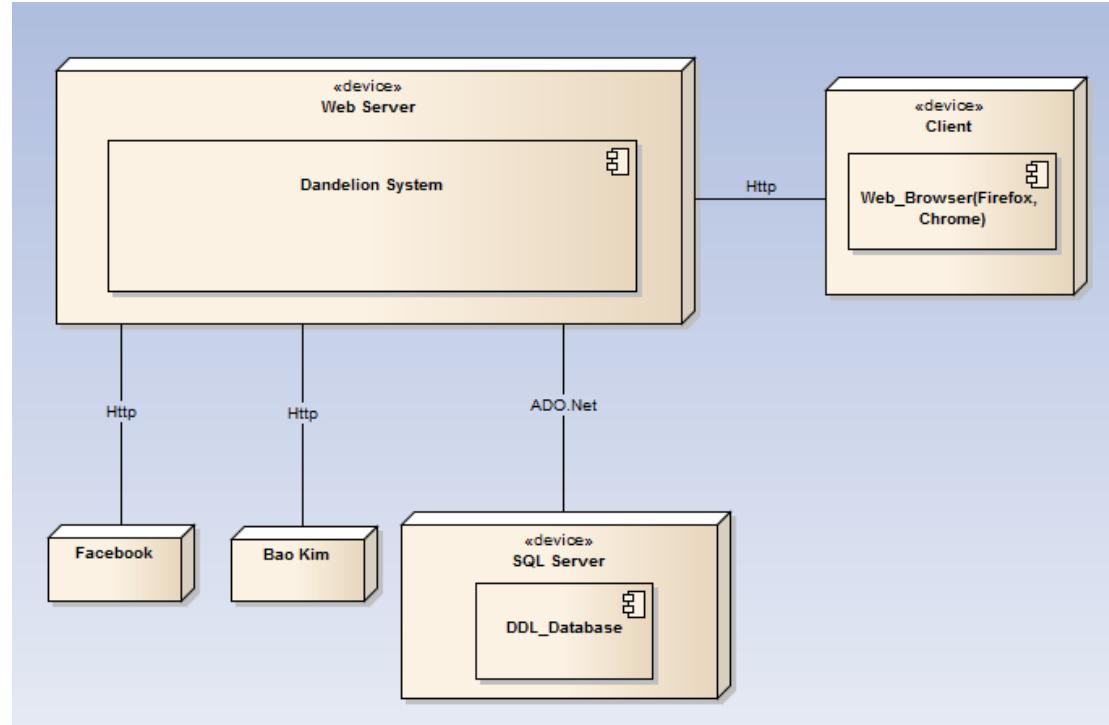


Figure 4-9: Deployment Diagram

No	Name	Description
1	SQL Server	SQL Server use to store system's data. Using SQL Server 2010 or higher.
2	Facebook	Facebook Api server
3	Bao Kim	Bao Kim Api server
4	Client	Client is web browser to use system. Firefox 30, Chrome 40 or higher.
5	Web Server	Web server is hosted by Azure Web Service.

Table 4-5: Deployment Diagram Description

## 4.3 Detail Design

### 4.3.1 Package

#### 4.3.1.1 Package Diagram

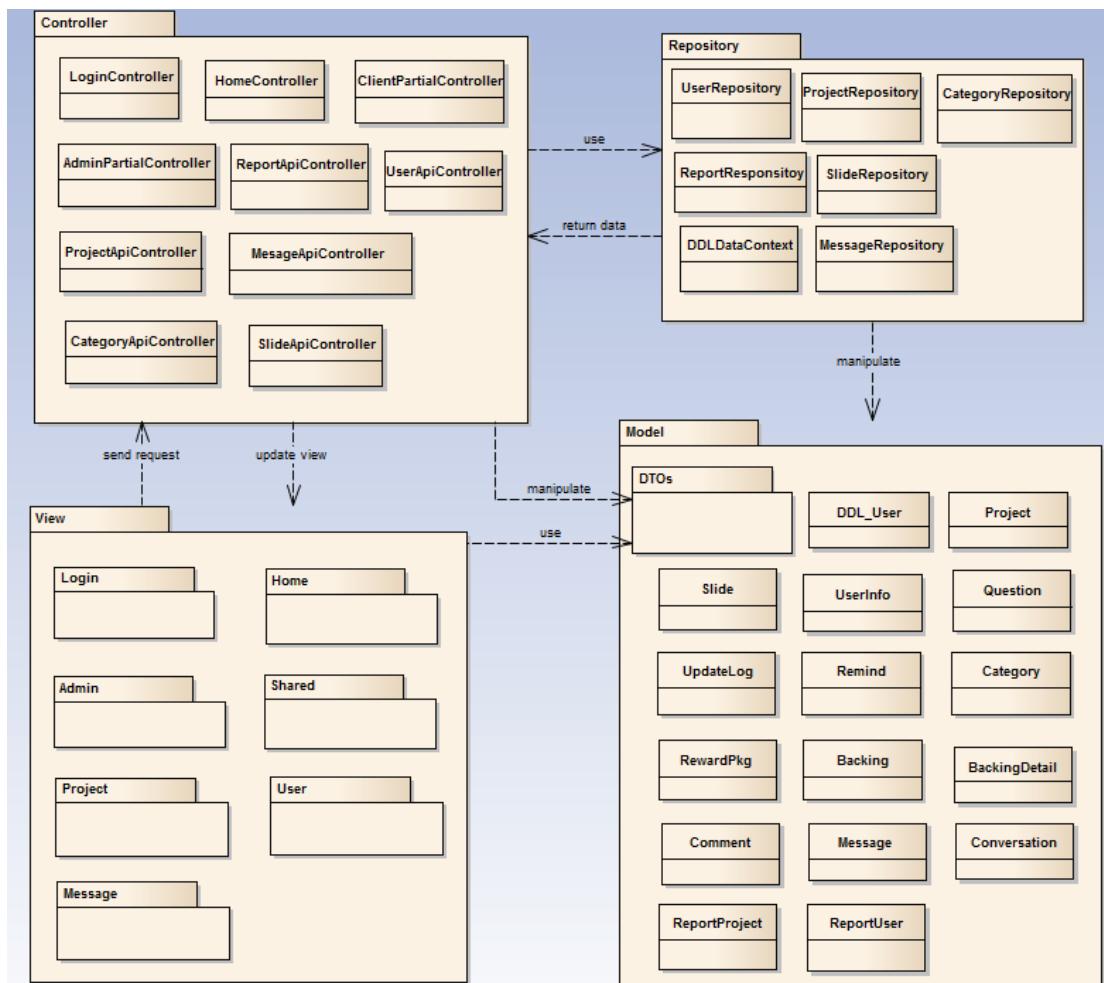


Figure 4-10: Package Diagram

#### 4.3.1.2 Package Description

##### 4.3.1.2.1 Model

###### 4.3.1.2.1.1 DDL\_User

No	Field Name	Type	Default	Note	Description
1	DDLUserID	int		private	User's ID
2	UserName	String		private	User's name
3	Password	String		private	User's password
4	IsActive	bool	false	private	User's active status (inactive, active)
5	CreatedDate	Datetime		private	User's created date
6	LoginType	String	normal	private	User's created type includes: normal, Facebook or both
7	Email	string		private	User's email
8	LastLogin	Datetime		private	User's last login
9	UserType	String	normal	private	User's type (admin or nomal user)
10	IsVerify	bool	false	private	User' email verify status
11	VerifyCode	string		private	User's verify code
12	UserInfo	UserInfo		private	User's Information
13	CreatedConversations	ICollection<Conversation>		private	User's created conversations
14	ReceivedConversations	ICollection<Conversation>		private	User's received conversations
15	SentMessages	ICollection<Message>		private	User's sent messages
16	CreatedReportUsers	ICollection<ReportUser>		private	User's created user reports
17	ReportedReportUsers	ICollection<ReportUser>		private	User's reported reports
18	CreatedProjects	ICollection<Project>		private	user's created projects

19	CreatedReportProjects	ICollection<ReportProject>		private	User's reported projects
20	Comments	ICollection<Comment>		private	User's comments
21	Reminds	ICollection<Remind>		private	User's reminds
22	Backings	ICollection<Backing>		private	User's backings

**Table 4-6: DDL\_User model****4.3.1.2.1.2 userinfo**

No	Field Name	Type	Default	Note	Description
1	DDL.UserID	int		private	Person's ID
2	FullName	String		private	Person's full name
3	ProfileImage	String		private	Person's profile image
4	Biography	String		private	Person's biography
5	Address	String		private	Person's address
6	Gender	String	male	private	Person's gender
7	PhoneNumber	String		private	Person's phone number
8	DateOfBirth	Datetime		private	Person's date of birth
9	Website	String		private	Person's website
10	FacebookURL	String		private	Person's facebook link
11	Country	string		private	Person's country
12	DDL_User	DDL_User		private	Object of DDL_User

**Table 4-7: UserInfo model****4.3.1.2.1.3 Project**

No	Field Name	Type	Default	Note	Description
1	ProjectID	int		private	Project's ID
2	ProjectCode	String		private	Project's code with fixed format
3	CategoryID	int		private	Project's category
4	CreatorID	int		private	Project's UserID who created project

5	Title	String		private	Project's title
6	Risk	String		private	Project's risk
7	ImageURL	String		private	Project's image link
8	SubDescription	String		private	Project's sub description
9	Location	String		private	Project's location created
10	IsExpired	bool		private	Is Project's expired
11	CurrentFunded	decimal		private	Project's funded status
12	IsFunded	bool		private	Is Project's funded
13	CreatedDate	Datetime		private	Project's created date
14	UpdatedDate	Datetime		private	Project's updated date
15	ExpireDate	datetime		private	Project's expired date
16	FundingGoal	decimal		private	Project's funding goal
17	Description	String		private	Project's description
18	VideoUrl	String		private	Project's video link
19	PopularPoint	int	0	private	Project's popular point
20	PointOfTheDay	int	0	private	Project's popular point of day
21	Status	String	draft	private	Project's status ( draft, pending, approved, suspended)
22	Category	Category		private	Project's category
23	Creator	DDL_User		private	Creator of project
24	Questions	ICollection<Question>		private	Project's questions
25	ReportCollection	ICollection<ReportProject>		private	Project's reported
26	UpdateLogs	ICollection<UpdateLog>		private	Project's updatelog
27	RewardPkgs	ICollection<RewardPkg>		private	Project's rewardPkg

28	Reminds	ICollection<Remind>		private	Project's reminds
29	Comments	ICollection<Comment>		private	Project's comments
30	Backings	ICollection<Backing>		private	Project's backings list

**Table 4-8: Project model****4.3.1.2.1.4 Question Model**

No	Field Name	Type	Default	Note	Description
1	QuestionID	Int		private	question's ID
2	ProjectID	Int		private	question's project id
3	QuestionContent	String		private	question's content
4	Answer	String		private	question's answer
5	CreatedDate	Datetime		private	question's created date
6	Project	Project		private	Project that includes question

**Table 4-9: Question Model****4.3.1.2.1.5 Category model**

No	Field Name	Type	Default	Note	Description
1	CategoryID	int		private	Category's ID
2	Name	String		private	Category's name
3	IsActive	bool	false	private	Category's active status
4	Description	String		private	Category's description
5	Projects	ICollection<Project>		private	List project of category

**Table 4-10:Category Model**

#### 4.3.1.2.1.6 UpdateLog Model

No	Field Name	Type	Default	Note	Description
1	UpdateLogID	int		private	Update's ID
2	ProjectID	Int		private	Update's project id
3	Title	String		private	Update's title
4	Description	String		private	Update's description
5	CreatedDate	datetime		private	Update's created date
6	Project	Project		private	Project that includes updatelog

**Table 4-11:UpdateLog Model**

#### 4.3.1.2.1.7 ReportProject Model

No	Field Name	Type	Default	Note	Description
1	ReportID	int		private	Report 's ID
2	ProjectID	int		private	Report 's project id
3	ReporterID	int		private	Report 's user id, who report project
4	Subject	String		private	Report 's subject
5	ReportContent	String		private	Report 's content
6	ReportedDate	datetime		private	Report 's report date
7	Status	String	new	private	Report's status (new, viewed, done)
8	Reporter	DDL_User		private	Reporter Object
9	Project	Project		private	Project Object

**Table 4-12:ReportProject Model**

#### 4.3.1.2.1.8 Remind Model

No	Field Name	Type	Default	Note	Description
1	RemindID	int		private	Remind's ID
2	ProjectID	int		private	Remind's project id
3	UserID	int		private	Remind's user id
4	User	DDL_User		private	User object
5	Project	Project		private	Project object

**Table 4-13:Remind Model**

#### 4.3.1.2.1.9 Backing Model

No	Field Name	Type	Default	Note	Description
1	BackingID	int		private	Back's ID
2	UserID	int		private	Back's user id
3	ProjectID	int		private	Back's project id
4	BackedDate	Datetime		private	Back's date
5	Description	String		private	Back's description
6	IsPublic	bool	true	private	Is back public
7	BackingDetail	BackingDetail		private	Detail object of backing
8	User	DDL_User		private	User object
9	Project	Project		private	Project object

**Table 4-14:Backing Model**

#### 4.3.1.2.1.10 BackingDetail Model

No	Field Name	Type	Default	Note	Description
1	BackingID	int		private	Backing's ID
2	BackerName	string		private	Name of backer
3	RewardPkgID	int		private	Identify number of reward package
4	PledgedAmount	decimal	0	private	total pledged amount
5	Quantity	int	1	private	quantity of reward

6	Description	string		private	Backing's description
7	Address	string		private	Backer's address
8	PhoneNumber	string		private	Backer's phone number
9	Email	string		private	Backer's email

**Table 4-15:BackingDetail Model****4.3.1.2.1.11 Comment Model**

No	Field Name	Type	Default	Note	Description
1	CommentID	int		private	Comment's ID
2	UserID	int		private	Comment's user id
3	ProjectID	int		private	Comment's project id
4	CommentContent	String		private	Comment's content
5	CreatedDate	Datetime		private	Comment's date
6	IsHide	bool	false	private	Is Comment's hide
7	UpdatedDate	Datetime		private	Comment's updated datetime
8	IsEdited	bool	false	private	Is comment edited
9	User	DDL_User		private	User who comments
10	Project	Project		private	Project which includes comment

**Table 4-16:Comment Model****4.3.1.2.1.12 ReportUser Model**

No	Field Name	Type	Default	Note	Description
1	ReportID	int		private	Report's ID
2	ReporterID	int		private	Report's report user ID
3	ReportedUserID	int		private	Report's reported user ID

4	Status	String		private	Report's status (new, viewed, done)
5	Subject	String		private	Report's Subject
6	ReportContent	String		private	Report's content
7	ReportedDate	datetime		private	Report's date
8	Reporter	DDL_User		private	User who report
9	ReportedUser	DDL_User		private	User who is reported

**Table 4-17: ReportUser Model****4.3.1.2.1.13 Slide Model**

No	Field Name	Type	Default	Note	Description
1	SlideID	int		private	Slide's ID
2	Title	String		private	Slide's title
3	Order	int		private	Slide's order
4	Description	String		private	Slide's description
5	ImageUrl	String		private	Slide's image link
6	VideoUrl	String		private	Slide's video
7	IsActive	bool	false	private	Slide's isActive status
8	SliderURL	String		private	Slide's slide URL
9	ButtonColor	String	green	private	Slide's button color
10	CreatedDate	datetime		private	Slide's created date
11	TextColor	string	light	private	slide's text color
12	ButtonText	String		private	Slide's button text

**Table 4-18:Slide Model****4.3.1.2.1.14 Conversation Model**

No	Field Name	Type	Default	Note	Description
1	ConversationID	int		private	Conversation's ID
2	CreatorID	int		private	Conversation's creator id

3	ReceiverID	int		private	Conversation's receiver
4	Subject	String		private	Conversation's subject
5	CreatedDate	Datetime		private	Conversation's created date
6	UpdatedDate	Datetime		private	Conversation's update date
7	DeleteStatus	String	no	private	Conversation's user delete status
8	ViewStatus	String	creator	private	Conversation's user view status
9	Creator	DDL_User		private	User who create conversation
10	Receiver	DDL_User		private	User who receive
11	Messages	ICollection<Message>		private	List message of conversation

**Table 4-19:Conversation Model****4.3.1.2.1.15 Message Model**

No	Field Name	Type	Default	Note	Description
1	MessageID	int		private	Message's ID
2	ConversationID	int		private	Message's conversation id
3	UserID	int		private	Message's user id
4	MessageContent	String		private	Message's content
5	SentTime	Datetime		private	Message's sent time
6	Conversation	Conversation		private	Conversation which includes message
7	Sender	DDL_User		private	User who send message

**Table 4-20:Message Model****4.3.1.2.1.16 RewardPkg Model**

No	Field Name	Type	Default	Note	Description
1	RewardID	int		private	Reward's ID
2	ProjectID	int		private	Reward's project id

3	Quantity	int		private	Reward's limited quantity
4	CurrentQuantity	int		private	Reward's current quantity
5	Type	String		private	Reward's type (noReward, limited, unlimited)
6	Description	String		private	Reward's description
7	EstimatedDeliveryDate	datetime		private	Reward's estimated delivery date
8	IsHide	bool	true	private	Is Reward hide
9	PledgeAmount			private	Min amount for one package
10	Project	Project		private	Project that includes reward
11	BackingDetails	ICollection<BackingDetail>		private	List backing

**Table 4-21:Reward package table**

#### 4.3.1.2.2 DTOs

Contain many data transfer object classes.

#### 4.3.1.2.3 View

Contain many .cshtml files to display user interface.

#### 4.3.1.2.4 Repository

##### 4.3.1.2.4.1 UserRepository

No	Method	Parameter	Return	Description
1	GetBackingFullInforListForExport		List<Object>	Get backing list for export
2	GetUserTop	string	Dictionary<Object>	Get top 10 user
3	GenerateNewPassword		string	Generate new password from random string

4	GenerateResetCode		string	Generate reset password code from random string
5	GenerateRandomString	int	string	Generate random string
6	GetByUserNameOrEmail	string, string	DDL_User	Get user by username, email and password
7	GetByUserNameOrEmail	string	DDL_User	Get user by username or email
8	GetBasicInfo	string	UserBasicInfoDTO	get basic information of a user
9	InsertUser	DDL_User	DDL_User	Insert user into DB
10	RegisterFacebook	dynamic	DDL_User	register new user from facebook information
11	Register	UserRegisterDTO	DDL_User	Register new user by normal way
12	VerifyAccount	string, string		Verify account by code
13	UpdateUser	DDL_User	DDL_User	Update information of a user
14	SendCodeResetPassword	string	bool	Update reset code into DB

15	ResetPassword	string, string	bool	Reset password of a user
16	GetListUserName	string	List<UserNameDT O>	Get all username of user in system.
17	GetUserPublicInfo	string	UserPublicInfoDTO	Get public information of a user.
18	GetUserEditInfo	string	UserEditInfoDTO	Get information of a user to display in edit profile page.
19	EditUserInfo	UserEditIn foDTO, string		Edit profile of a user.
20	GetUserPassword	string	EditPasswordDTO	Get password of a user.
21	ChangePassword	string, EditPassw ordDTO	bool	Change password of a user.
22	GetUserListForAdmin		AdminUserListDTO	Get list user for admin page.
23	ChangeUserStatus	string		Active or deactive a user
24	GetUserDashboardForAdmi n		AdminUserDashboa rdDTO	Get user dash board for admin page.
25	AdminGetTopBacker		List<TopBackerDT O>	Get top 10 backer of dandelion.

26	AdminGetRecentUser		List<RecentUserDTO>	Get last login user.
----	--------------------	--	---------------------	----------------------

**Table 4-22: User Repository****4.3.1.2.4.2 ProjectRepository**

No	Method	Parameter	Return	Description
1	projectTitleList	string	List<ProjectTitleDTO>	get list title project
2	getStatisticsInfo r		Dictionary<string, int>	Get statistic information
3	GetProjectTop	string	List<ProjectBasicViewDTO>	Get top 10 project
4	GetProject	int, int, string, string, string, string	List<ProjectBasicViewDTO>	Get project by options
5	orderBy	string, IQueryable<ProjectBasicViewDTO>	IQueryable<ProjectBasicViewDTO>	order list project by options
6	GetProjectByCategory		List<ProjectBasicViewDTO>	Get list project by category
7	GetTotalFund		List<ProjectBasicViewDTO>	Get total funded
8	CreatProject	ProjectCreateDTO, string	string	Create new project
9	EditProjectBasic	ProjectEditDTO, string, string	ProjectEditDTO	edit basic information of a project

10	EditProjectStory	ProjectStoryDTO , string	ProjectStoryDTO	edit advanced information of a project
11	GetProjectBasic	string, string	ProjectEditDTO	Get basic information of a project
12	GetProjectStory	int, string	ProjectStoryDTO	Get advanced information of a project
13	SubmitProject	ProjectEditDTO, string	List<string>	Submit project for review
14	GetBackProjectI nfo	string	ProjectInfoBackDTO	Get backing information of a project
15	BackProject	ProjectBackDTO	string	Back a project
16	CaculateProject Point	string, int		Calculate popular point
17	AdminGetBacki ngDetail	int	ProjectBackDTO	Get backing detail of a backing
18	GetPendingProj ectList		List<ProjectBasicListDT O>	Get list pending project
19	GetProjectList		List<ProjectBasicListDT O>	Get list project
20	AdminChangePr ojectStatus	ProjectEditDTO, string	bool	Change project status( approve, reject)
21	AdminDashboar dInfo		AdminDashboardInfoD TO	Get

22	AdminGetTopProjectList		List<ProjectBasicListDTO>	Get top 10 project
23	AdminProjectStatistic	int	AdminProjectStatisticDTO	Get statistic information of project
24	AdminStatisticTable	string	List<AdminDashboardInfoDTO>	Get statistic information
25	AdminGetProjectDetail	string	ProjectDetailDTO	Get project detail for admin
26	GetQuestion	int	List<QuestionDTO>	Get list question of a project
27	CreateQuestion	int, QuestionDTO, string	QuestionDTO	create a question
28	EditQuestion	List<QuestionDTO>, string	bool	edit list question
29	EditSingleQuestion	QuestionDTO, string	QuestionDTO	Edit a question
30	DeleteQuestion	int, string	bool	Delete a question
31	GetRewardPkg	int	List<RewardPkgDTO>	Get list rewardPkg
32	GetRewardPkgByCode	string	List<RewardPkgDTO>	Get list rewardPkg by project code
33	CreateRewardPkg	int, RewardPkgDTO, string	RewardPkgDTO	Create a rewardPkg
34	EditRewardPkg	RewardPkgDTO, string	bool	Edit a rewardPkg

35	DeleteRewardPk g	int, string	bool	Delete a rewardPkg
36	GetUpdateLog	int	List<UpdateLogDTO>	Get updateLog
37	CreateUpdateLo g	int, UpdateLogDTO, string	UpdateLogDTO	Create updateLog
38	EditUpdateLog	List<UpdateLog DTO>, string	bool	Edit list updateLog
39	EditSingleUpdat eLog	UpdateLogDTO, string	UpdateLogDTO	Edit a updateLog
40	DeleteUpdateLo g	int, string	bool	Delete a updateLog
41	GetProjectByCo de	string, string	ProjectDetailDTO	Get a project by Code
42	GetListCommen t	string, DateTime, string	List<CommentDTO>	Get list comment of a project
43	GetListUpdateL og	string, string	List<UpdateLogDTO>	Get list updateLog of a project
44	GetBackedProje ctHistory	string	List<ProjectBasicViewD TO>	Get backed history.
45	BackingInfo	string	List<BackingInfoDTO>	Get backing information
46	GetBackedProje ct	string	List<ProjectBasicViewD TO>	Get list backed project
47	GetStarredProje ct	string	List<ProjectBasicViewD TO>	Get list reminded project
48	GetCreatedProje ct	string	List<ProjectBasicViewD TO>	Get list created project

49	DeleteProjectReminded	int	bool	Unremind a project
50	DeleteProjectDraft	int	bool	Delete a draft project
51	RemindProject	string, string	bool	Remind a project
52	GetComments	int, bool, Datetime	List<CommentDTO>	Get list comment of a project
53	AddComment	string, CommentDTO, DateTime	List<CommentDTO>	Add comment
54	ShowHideComment	int, string	CommentDTO	Set show/hide comment
55	EditComment	int, string, string	CommentDTO	Edit comment
56	DeleteComment	int, string	bool	Delete a comment

**Table 4-23:Project Repository****4.3.1.2.4.3 ReportRepository**

No	Method	Parameter	Return	Description
1	ReportProject	string, string, string		Create a project report function
2	ReportUser	string, string, string		Create a user report function
3	GetReportProjects		List<ReportProjectDTO>	Get list project report
4	changeReportProjectStatus	int, string		Change status of a project report
5	changeReportUserStatus	int, string		Change status of a user report

**Table 4-24: Report Repository**

#### 4.3.1.2.4.4 MessageRepository

No	Method	Parameter	Return	Description
1	GetListConversation	string	List<ConversationBasicDTO>	List list conversation of a user.
2	GetListSentConversation	string	List<ConversationBasicDTO>	Get list sent conversation of a user.
3	GetListReceivedConversation	string	List<ConversationBasicDTO>	Get list receiver conversation of a user
4	GetConversation	int, string	ConversationDetailDTO	Get conversation detail
5	ReplyMessage	ReplyDTO, string	MessageDTO	Reply a conversation
6	Delete	int, string	bool	Delete a conversation
7	DeleteMessageList	int[], string	bool	Delete a list conversation

**Table 4-25:Message Repository**

#### 4.3.1.2.4.5 Slide Repository

No	Method	Parameter	Return	Description
1	GetSlides		List<SlideDTO>	Get list slide.
2	CreateSlide	SlideDTO	SlideDTO	create new slide
3	EditSlide	SlideDTO	SlideDTO	edit a slide
4	DeleteSlide	int	bool	delete a slide
5	ChangeSlideStatus	int	SlideDTO	active or deactivate a slide
6	ChangeOrder	int, string	List<SlideDTO>	change slide's order

**Table 4-26:Slide Repository**

#### 4.3.1.2.5 Controller

##### 4.3.1.2.5.1 LoginController

No	Method	Parameter	Return	Description
1	AuthenFacebook		ActionResult	Use to request login to Facebook
2	FacebookCallback	string	ActionResult	Function callback, get data from Facebook
3	ForgotPassword		ActionResult	Set user's password to a random string and send to email of user
4	Login	String	ActionResult	Return login page for user
5	Login	Object, String	ActionResult	Receive request login with form data included user name and password.
6	Logout		ActionResult	Logout current user from DDL.
7	AdminLogin	string		Return login page for admin.
8	AdminLogin	Object, string	ActionResult	Receive request login with form data included user name and password.
9	AdminLogout		ActionResult	Logout for admin page.
10	Active	string, string	ActionResult	Verify email account, active new account with code.

Table 4-27:Login Controller

##### 4.3.1.2.5.2 HomeController

No	Method	Parameter	Return	Description
1	Index		ActionResult	Display main page of DDL client.
2	AdminIndex		ActionResult	Display main page of DDL admin.
3	Error		ActionResult	Display error page

Table 4-28: Home Controller

##### 4.3.1.2.5.3 ClientPartialController

No	Method	Parameter	Return	Description

<b>1</b>	Home		ActionResult	Display home page
<b>2</b>	Register		ActionResult	Display Register page
<b>3</b>	RegisterSuccess		ActionResult	Display register successful page
<b>4</b>	ForgotPassword		ActionResult	Display ForgotPassword page
<b>5</b>	Discover		ActionResult	Display discovery page
<b>6</b>	Message		ActionResult	Display message list page
<b>7</b>	MessageDetail		ActionResult	Display message detail page
<b>8</b>	CreateProject		ActionResult	Display create project page
<b>9</b>	ProjectDetail		ActionResult	Display project detail page
<b>10</b>	EditProject		ActionResult	Display Edit Project page
<b>11</b>	Statistics		ActionResult	Display Statistics page
<b>12</b>	Search		ActionResult	Display Search result page page
<b>13</b>	BackProject		ActionResult	Display first Backing page with list rewards of project.
<b>14</b>	PaymentProject		ActionResult	Display second Backing page to input backing information.
<b>15</b>	PublicProfile		ActionResult	Display Public Profile page
<b>16</b>	EditProfile		ActionResult	Display Edit Profile page
<b>17</b>	EditPassword		ActionResult	Display Edit Password page
<b>18</b>	Message		ActionResult	Display Message page
<b>19</b>	BackedProject		ActionResult	Display Backed Project page
<b>20</b>	BackedProjectHistory		ActionResult	Display backed history page
<b>21</b>	CreatedProject		ActionResult	Display Created Project page
<b>22</b>	StarredProject		ActionResult	Display Reminded Project page
<b>23</b>	ListBacker		ActionResult	Display page with list backing of a project.
<b>24</b>	Error		ActionResult	Display Error page.

<b>25</b>	NotFound		ActionResult	Display 404 not found page.
-----------	----------	--	--------------	-----------------------------

**Table 4-29: ClientPartialController**

## 4.3.1.2.5.4 AdminPartialController

No	Method	Parameter	Return	Description
1	Dashboard		ActionResult	Display dashboard page.
2	ProjectDashboard		ActionResult	Display Project Dashboard page
3	ProjectList		ActionResult	Display Project List page
4	ProjectDetail		ActionResult	Display project detail page
5	Slide		ActionResult	Display slide page.
6	MessageList		ActionRes	Display message list for admin.
7	MessageDetail			Display message detail for admin.
8	UserDashboard		ActionResult	Display User Dashboard project page
9	UserList		ActionResult	Display User List page
10	UserProfile		ActionResult	Display profile of a user page
11	BackingList			Display backing list page.
12	Category			Display category page.
13	ReportProject			Display report user page.
14	ReportUser			Display report project page
15	NotFound		ActionResult	Display Error page.
16	Error		ActionResult	Display 404 not found page.

**Table 4-30: AdminPartialController**

## 4.3.1.2.5.5 UserApiController

No	Method	Parameter	Return	Description
1	Register	Object	IHttpActionResult	Create new user api
2	ResetPassword	string	IHttpActionResult	Reset password to random string for a

				user and send to user's email.
3	GetListUserName	String	IHttpActionResult	Api to get list username which matched with input string. This function is used by message page.
4	CheckLoginStatus		IHttpActionResult	Check login status of current user.
5	GetPublicInfo	string	IHttpActionResult	Get public information of a user.
6	GetUserInfoEdit		IHttpActionResult	Get personal information of current user.
7	EditUserInfo		IHttpActionResult	Api to edit information of current user.
8	GetUserPasswordEdit		IHttpActionResult	Get password of current user.
9	ChangePassword	Object	IHttpActionResult	Change password of current user.
10	GetUserBackedProjectForAdmin	string	IHttpActionResult	Get list backed project of a user.
11	GetUserCreatedProjectForAdmin	string	IHttpActionResult	Get list created project of a user.
12	GetUserDashboardForAdmin		IHttpActionResult	
13	GetUserListForAdmin		IHttpActionResult	Get list user for admin.

14	GetUserProfileForAdmin	string	IHttpActionResult	Get profile of a user for admin page.
15	ChangeUserStatus	string	IHttpActionResult	Set active or deactivate a user.
16	GetUserTop		IHttpActionResult	Get top user for statistic
17	AdminGetTopBacker		IHttpActionResult	Get top backer for admin
18	AdminGetRecentUser		IHttpActionResult	Get last 10 logged user.
19	AdminGetRecentBackin g		IHttpActionResult	Get last 10 backing

**Table 4-31: UserAPIController****4.3.1.2.5.6 ProjectApiController**

No	Method	Parameter	Return	Description
1	CreateProject	Object	IHttpActionResult	Create a new project
2	EditProjectBasic	Object	IHttpActionResult	Edit basic information of a draft project
3	EditProjectStory	Object	IHttpActionResult	Edit advanced information of a project
4	GetProjectBasic	int	IHttpActionResult	Get basic information of a project
5	GetProjectStory	int	IHttpActionResult	Get advanced information of a project.

6	GetRewardPkg	int	IHttpActionResult	Get list reward package of a project
7	GetRewardPkgByCode	String	IHttpActionResult	Get list reward package of a project by code.
8	CreateRewardPkg	Int, Object	IHttpActionResult	Create new reward package for a project
9	EditRewardPkg	Object	IHttpActionResult	Create a reward package of project
10	DeleteRewardPkg	Int	IHttpActionResult	Delete a reward package
11	GetUpdateLog	int	IHttpActionResult	Get list update log of a project
12	CreateUpdateLog	Object	IHttpActionResult	Add an update log for a project
13	EditUpdateLog	Object	IHttpActionResult	Edit an update log of project.
14	DeleteUpdateLog	Int	IHttpActionResult	Delete an update log of project.
15	GetQuestion	Int	IHttpActionResult	Get list question and answer of a project
16	CreateQuestion	Object	IHttpActionResult	Create a Q&A
17	EditQuestion	Object	IHttpActionResult	Edit a Q&A
18	DeleteQuestion	Int	IHttpActionResult	Delete a Q&A

19	SubmitProject	Int	IHttpActionResult	Submit project for admin to review
20	BackProject	Object	IHttpActionResult	Back for a project
21	GetBackProjectInfo	string	IHttpActionResult	Get
22	GetPendingProjectList		IHttpActionResult	Get list pending, rejected project
23	AdminGetProjectGeneralInfo		IHttpActionResult	Get dashboard project information
24	AdminGetProjectList		IHttpActionResult	Get project list for admin
25	AdminGetProjectDetail	string	IHttpActionResult	Get project detail for admin
26	AdminGetProjectComment	string	IHttpActionResult	Get list comment of a project
27	AdminGetUpdateLogList	string	IHttpActionResult	Get update log list of a project
28	AdminChangeProjectStatus	Object	IHttpActionResult	Change project status
29	AdminGetListBacker	String	IHttpActionResult	Get list backing of a project.
30	AdminGetDashboardInfo		IHttpActionResult	Get dashboard information for admin.
31	AdminGetTopProjectList		IHttpActionResult	Get top 10 project success

32	AdminGetProjectStatistic	int	IHttpActionResult	Get statistic by year
33	AdminGetStatisticTable	Int	IHttpActionResult	Get statistic table by year
34	GetProjectTop	string	IHttpActionResult	Get top 10 project success by category
35	GetProject	Int, int, string, string, string, string, bool, string	IHttpActionResult	Get project with multi option
36	GetProjectByCategory		IHttpActionResult	Get project by category
37	GetProjectStatisticList		IHttpActionResult	Get project statistic
38	GetStatisticListForHome		IHttpActionResult	Get statistic for home page.
39	GetProjectDetail	string	IHttpActionResult	Get project detail by project code
40	GetCommentList	string, string	IHttpActionResult	Get list comment of a project
41	GetUpdateLogList	string	IHttpActionResult	Get list update log of a project
42	GetBackedProject		IHttpActionResult	Get list backed project of current user.

43	GetBackedProjectHistory		IHttpActionResult	Get backed history of current user.
44	GetStarredProject		IHttpActionResult	Get list reminded projects of current user.
45	GetCreatedProject		IHttpActionResult	Get list created projects of current user
46	DeleteProjectDraft	int	IHttpActionResult	Delete a draft project
47	DeleteProjectReminded	int	IHttpActionResult	Delete a reminded project
48	GetBackingInfo	string	IHttpActionResult	Get backing detail information
49	RemindProject	string	IHttpActionResult	Remind a project
50	GetListBacker	string	IHttpActionResult	Get backings list of a project
51	ReportProject	string, string	IHttpActionResult	Report a bad project
52	Comment	string, Object, string	IHttpActionResult	Add new comment to a project
53	ShowHideComment	int	IHttpActionResult	Set show/hide status of a comment
54	EditComment	int, string	IHttpActionResult	edit a comment

55	DeleteComment	int	IHttpActionResult	Delete a comment
56				

**Table 4-32: ProjectApiController****4.3.1.2.5.7 MessageApiController**

No	Method	Parameter	Return	Description
1	NewMessage	Object	IHttpActionResult	Create new message
2	GetListConversations		IHttpActionResult	Get list message of current user
3	GetListSentConversations		IHttpActionResult	Get list sent message of current user.
4	GetListReceivedConversations		IHttpActionResult	Get list received message of current user.
5	GetConversation	int	IHttpActionResult	Get message detail.
6	Reply	Object	IHttpActionResult	Reply to a message
7	DeleteMessage	int	IHttpActionResult	Delete a message
8	DeleteMessageList	int[]	IHttpActionResult	Delete a list of messages.
9	GetNumberNewMessage		IHttpActionResult	Get number new message of current user.

**Table 4-33: MessageApiController**

#### 4.3.1.2.5.8 SlideApiController

No	Method	Parameter	Return	Description
1	GetSlides		IHttpActionResult	Get list active slides
2	GetSlidesForAdmin		IHttpActionResult	Get list slides for admin page.
3	CreateSlide	Object	IHttpActionResult	Create new slide
4	DeleteSlide	int	IHttpActionResult	Delete a slide
5	ChangeSlideStatus	int	IHttpActionResult	Active or deactivate a slide.
6	EditSlide	Object	IHttpActionResult	Edit a slide information
7	ChangeOrder	int, string	IHttpActionResult	Change order of a slide.

**Table 4-34:MessageApiController**

#### 4.3.1.2.5.9 ReportApiController

No	Method	Parameter	Return	Description
1	GetReportProjects		IHttpActionResult	Get list project report for admin.
2	GetReportUsers		IHttpActionResult	Get list user report for admin.
3	changeReportStatus		IHttpActionResult	Change status of a report

**Table 4-35: ReportApiController**

#### 4.3.1.2.5.10 CategoryApiController

No	Method	Parameter	Return	Description
1	listDataForStatistic		IHttpActionResult	Get list data for statistic page.
2	getAllCategories		IHttpActionResult	Get all active categories
3	GetCategoryProjectCount		IHttpActionResult	Get all active categories with project count.

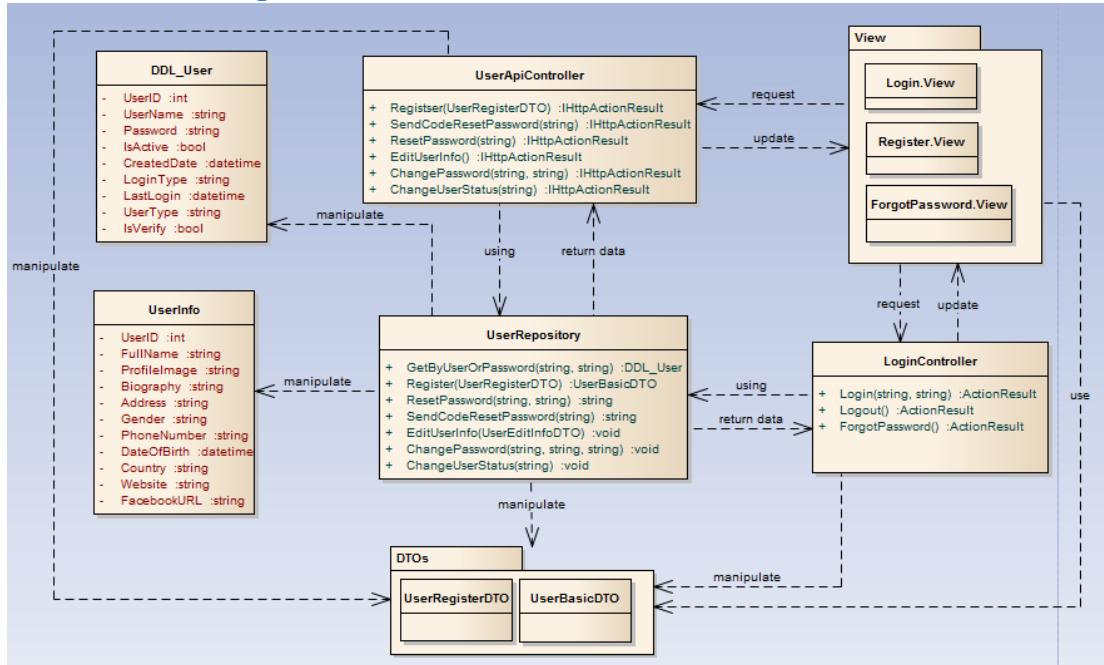
4	GetCategoriesForCreate		IHttpActionResult	Get list category for create project page.
5	GetCategoriesForAdmin		IHttpActionResult	Get list categories for admin page.
6	ChangeCategoryStatus	int	IHttpActionResult	Active, deactivate a category
7	AddNewCategory	Object	IHttpActionResult	Add new category
8	EditCategory	Object	IHttpActionResult	Edit information of a category

**Table 4-36:CategoryApiController**

### 4.3.2 Function

#### 4.3.2.1 Common

##### 4.3.2.1.1 Class diagram

**Figure 4-11: Common functions class diagram**

### 4.3.2.1.2 Sequence diagram

#### 4.3.2.1.2.1 Login

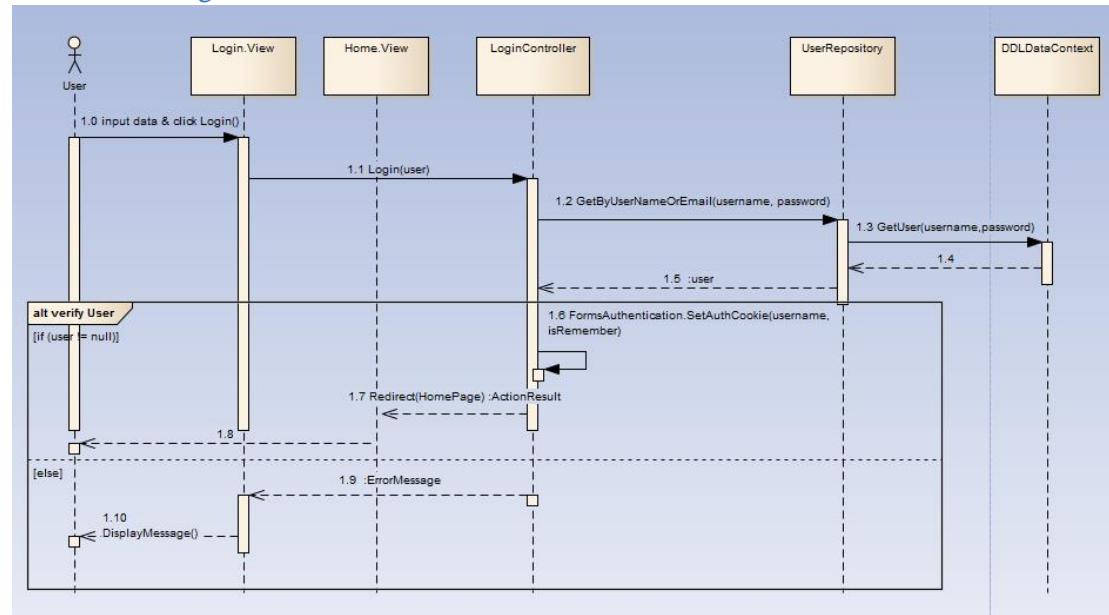


Figure 4-12: Login sequence diagram

#### 4.3.2.1.2.2 Logout

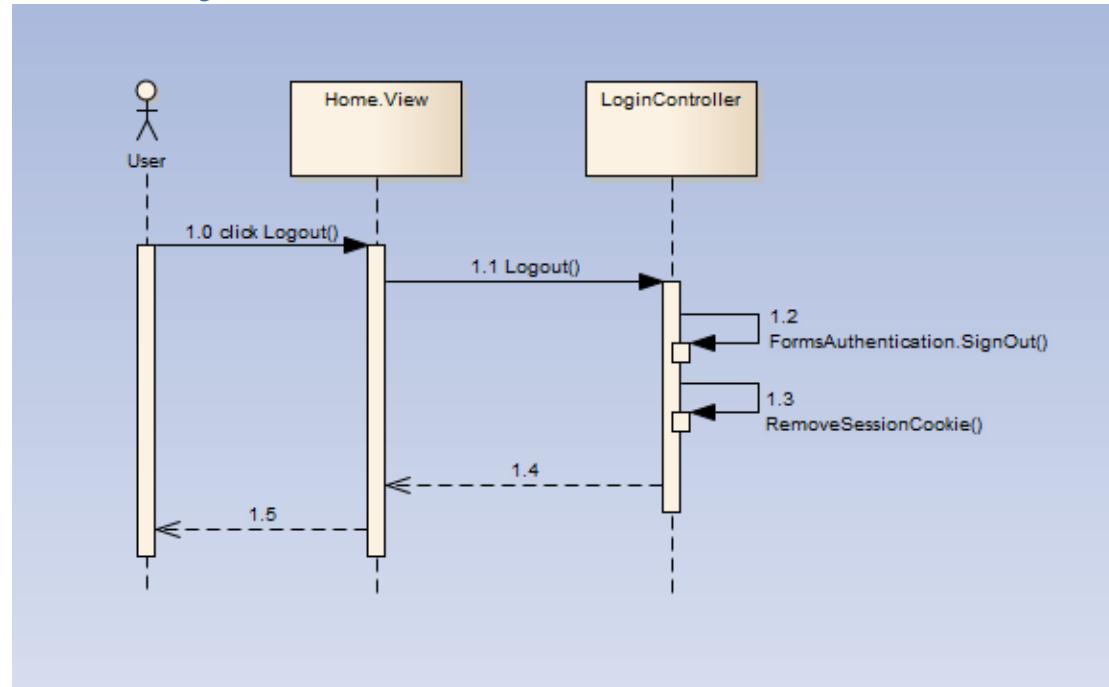


Figure 4-13: Logout sequence diagram

#### 4.3.2.1.2.3 Register

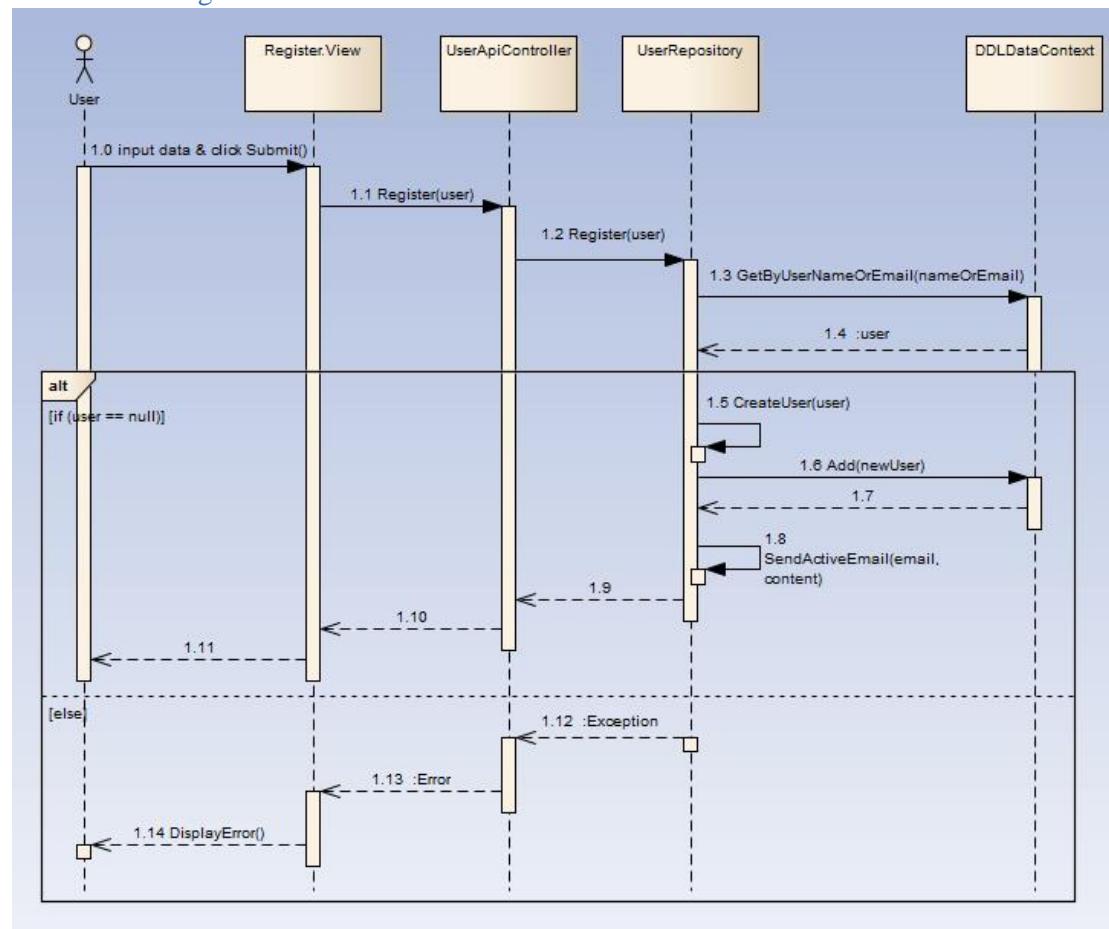


Figure 4-14: Register sequence diagram

#### 4.3.2.1.2.4 Forgot Password

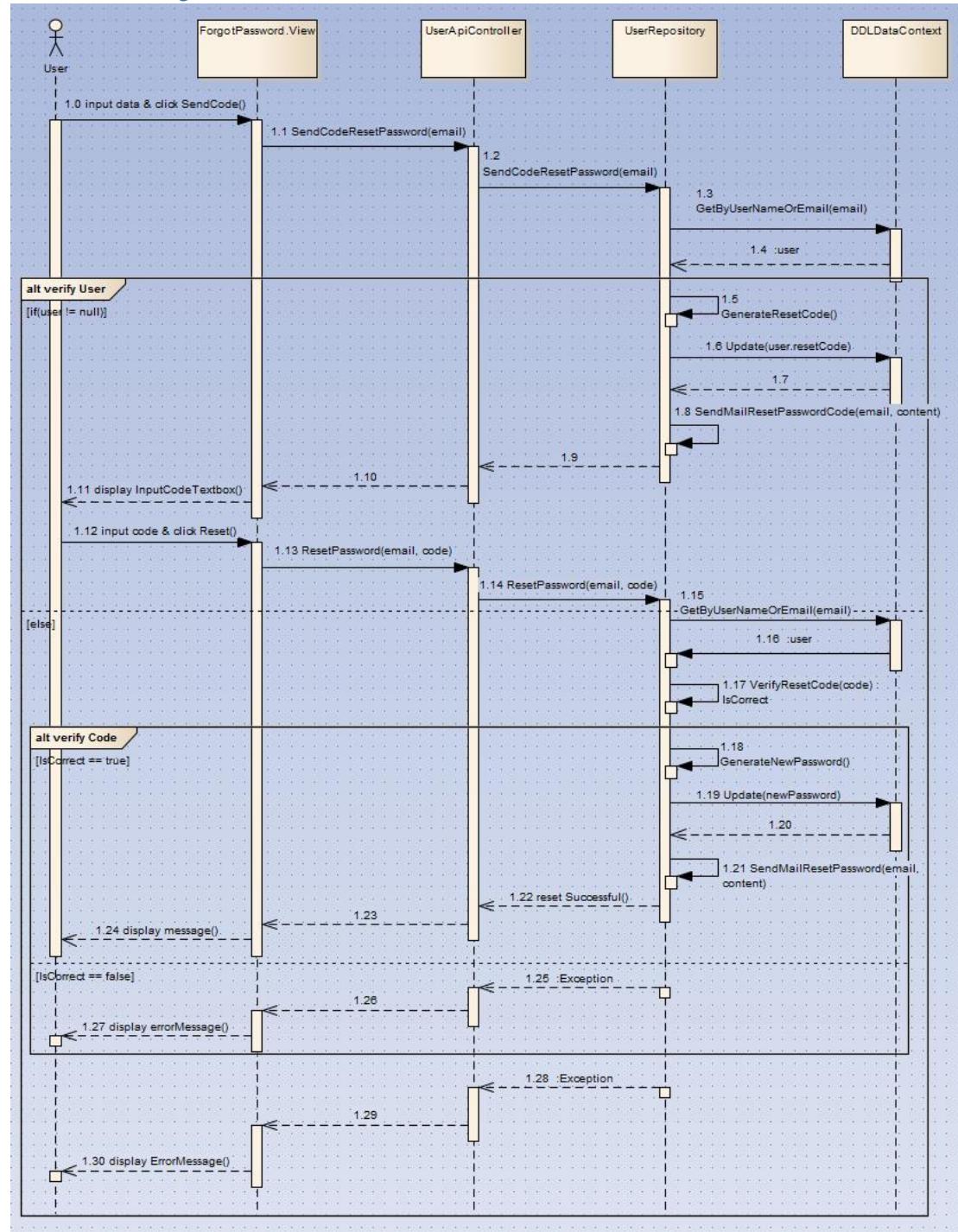


Figure 4-15: Forgot password sequence diagram

### 4.3.2.2 Project management

#### 4.3.2.2.1 Class diagram

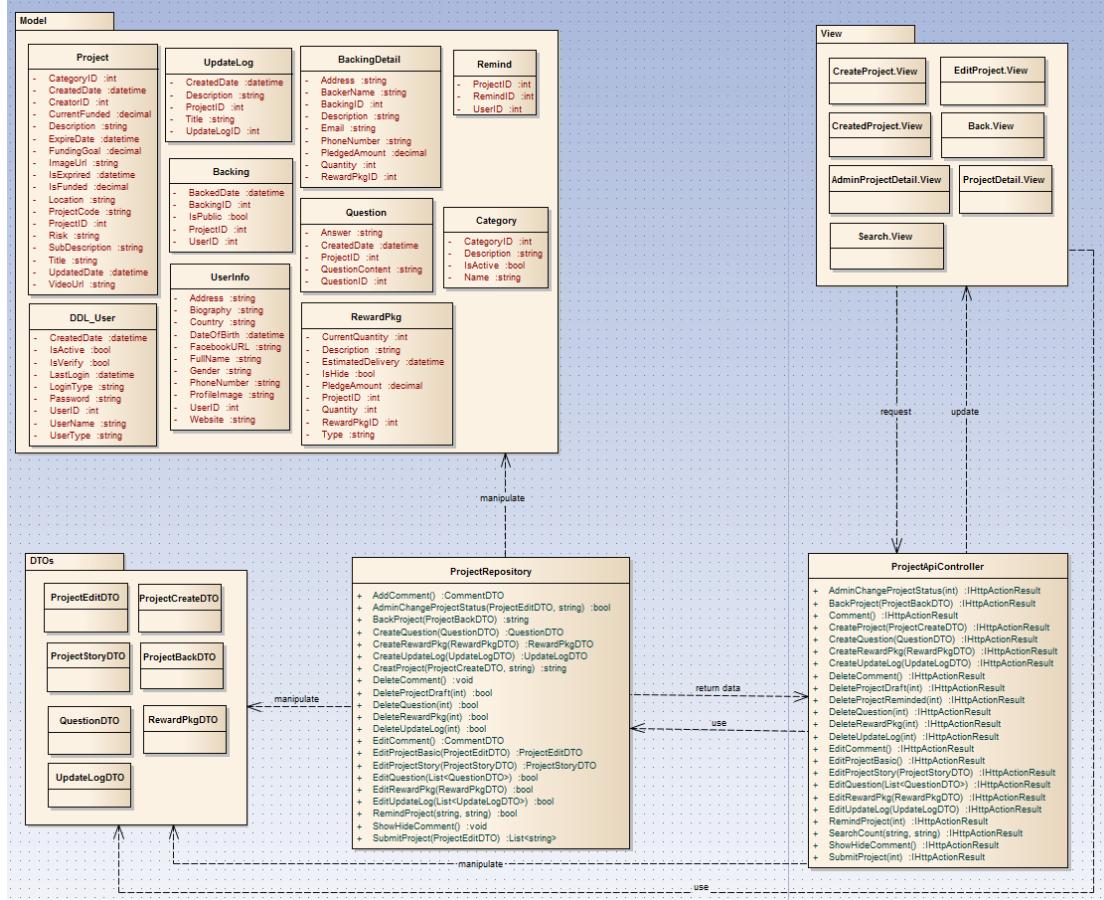
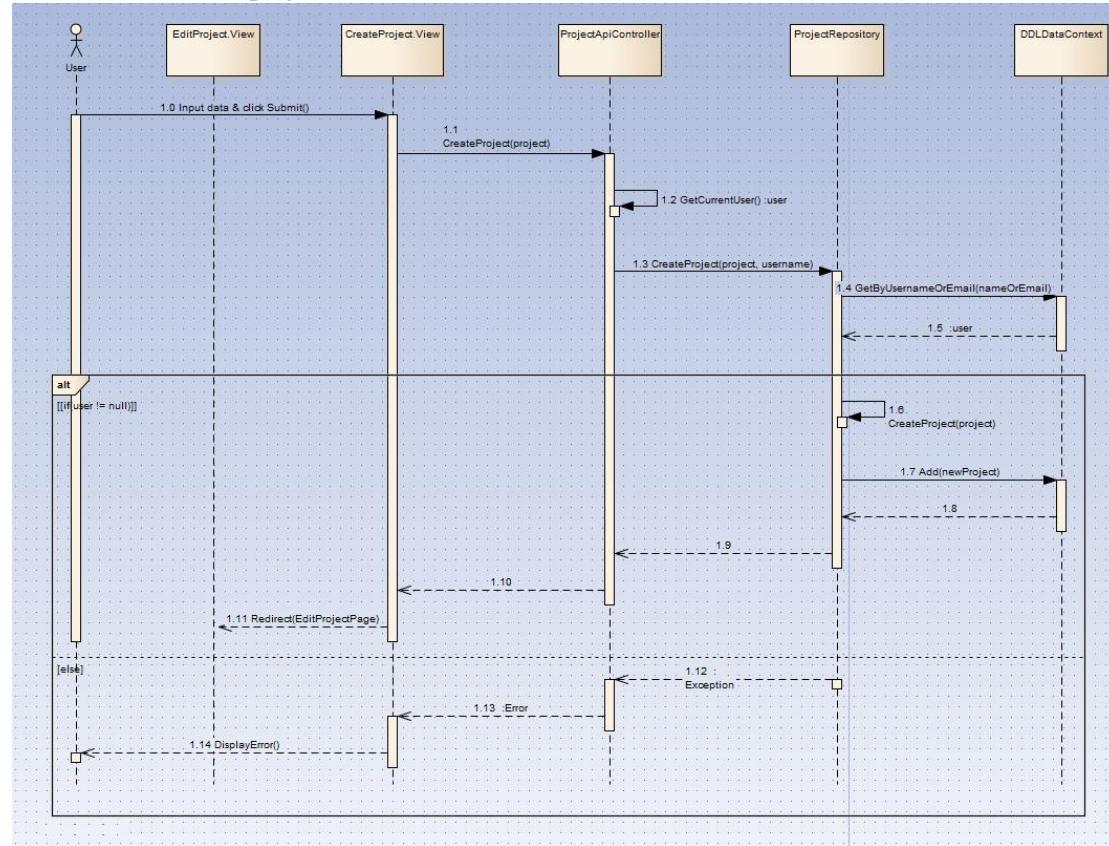


Figure 4-16: Project management class diagram

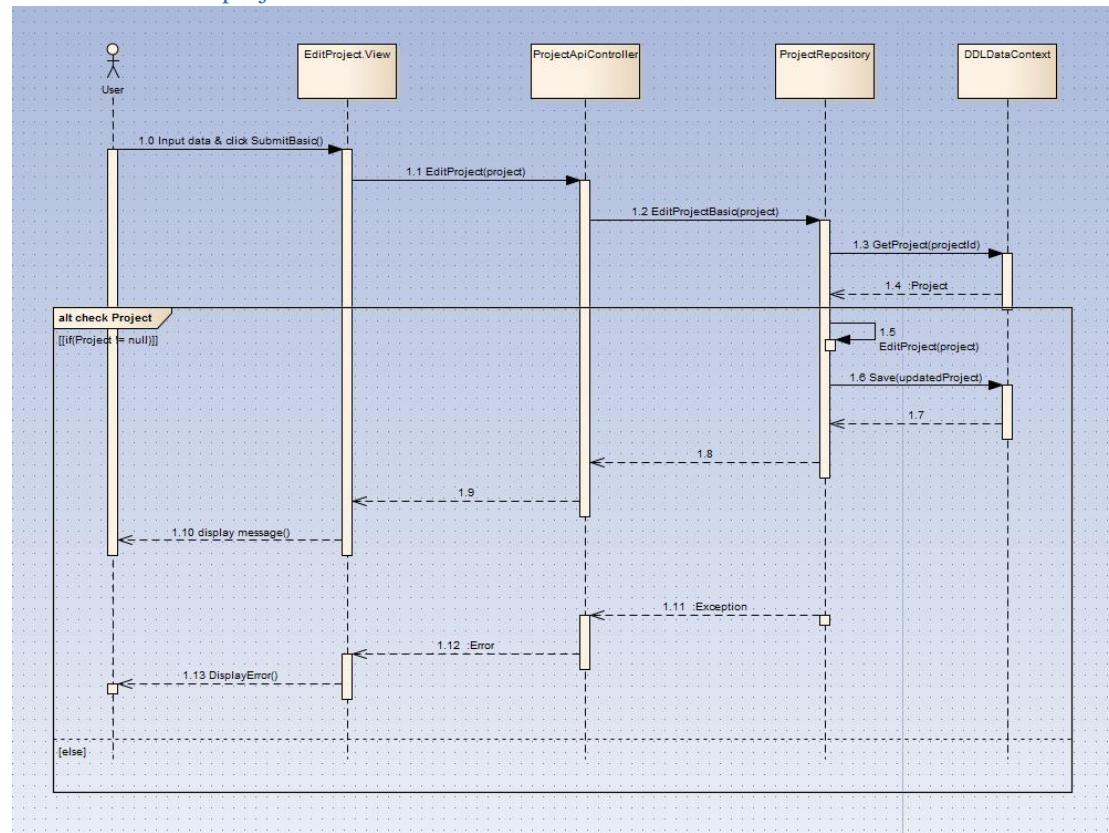
### 4.3.2.2.2 Sequence diagram

#### 4.3.2.2.1 Create project



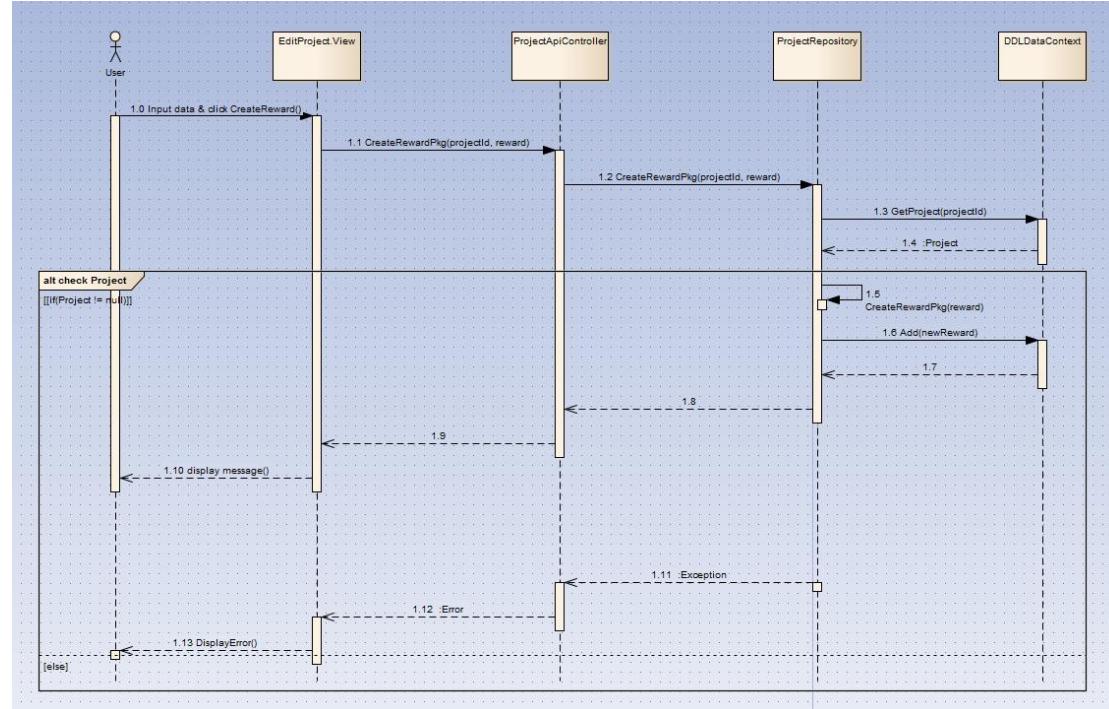
**Figure 4-17: Create project sequence diagram**

### 4.3.2.2.2 Edit project basic



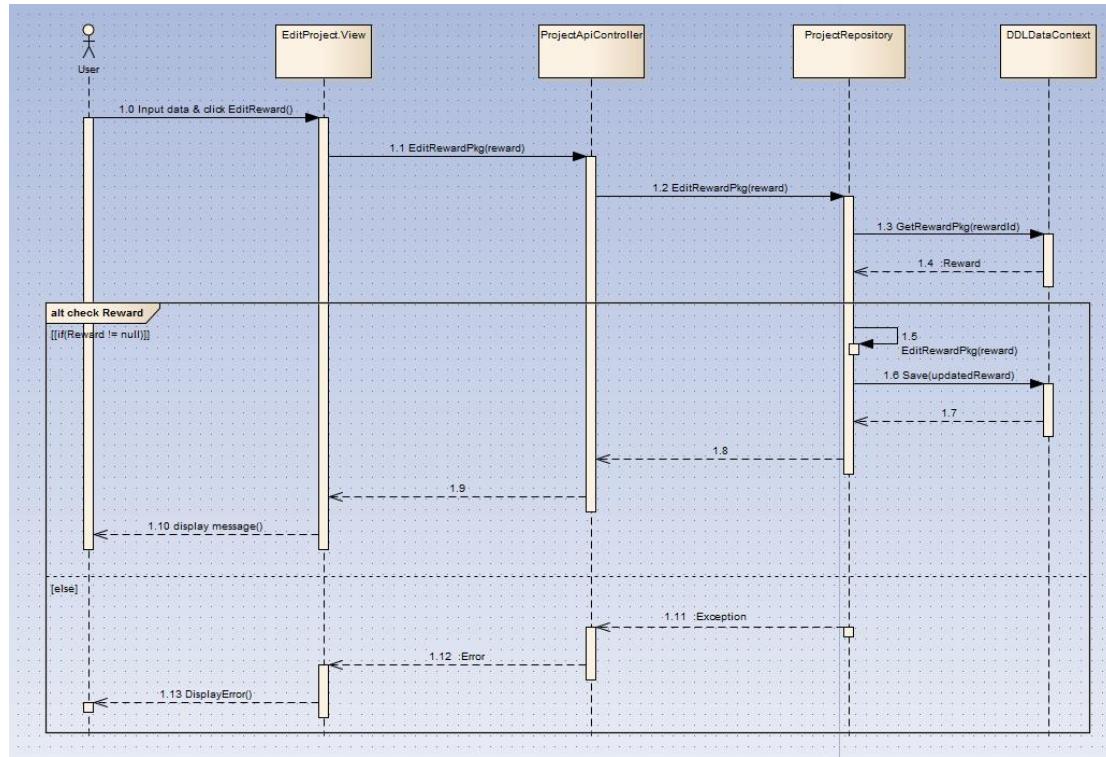
**Figure 4-18:** Edit project basic sequence diagram

### 4.3.2.2.3 Create reward package



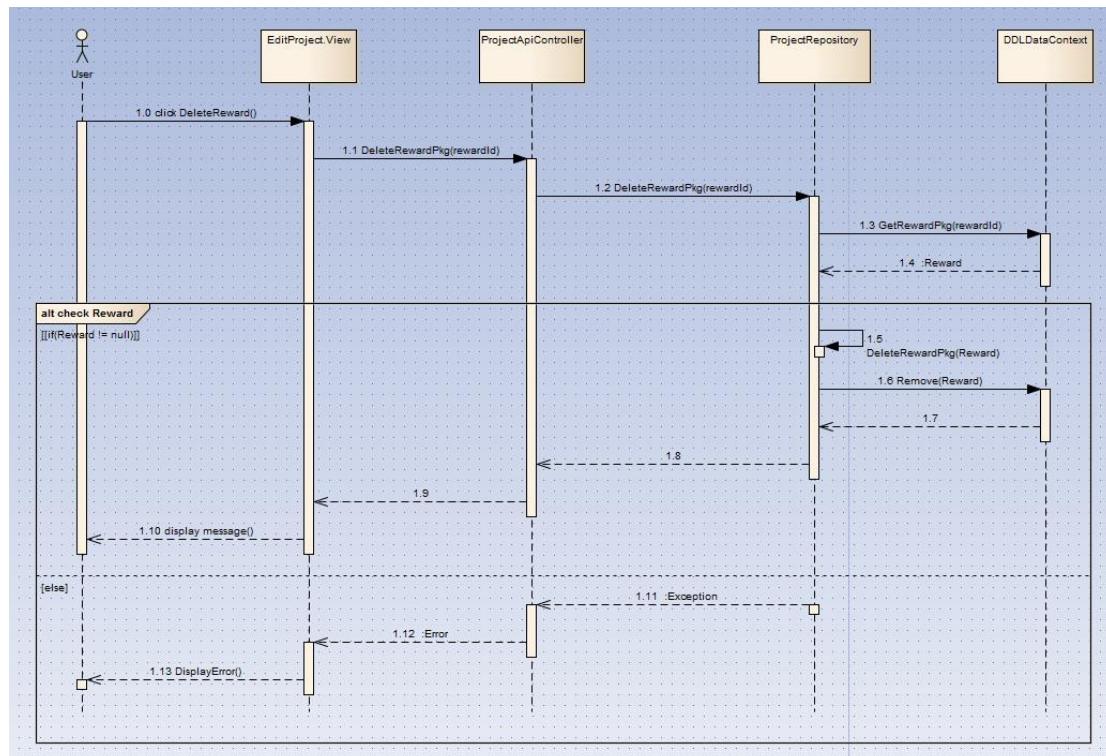
**Figure 4-19:** Create reward packet sequence diagram

#### 4.3.2.2.4 Edit reward package



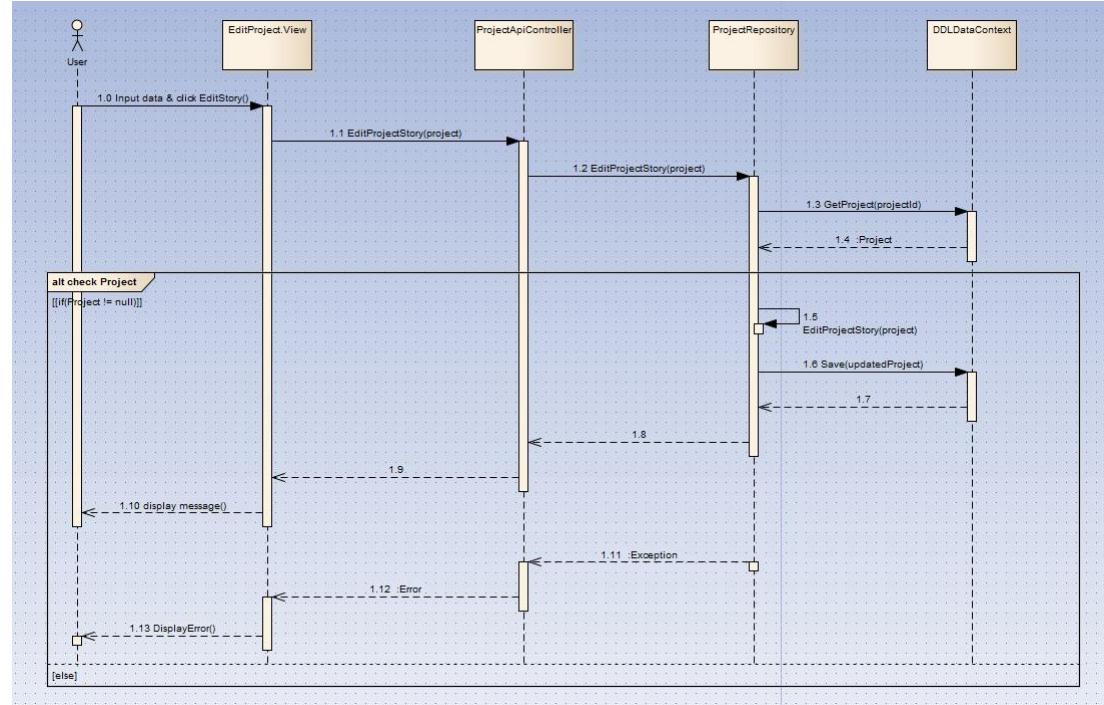
**Figure 4-20:** Edit reward packet sequence diagram

#### 4.3.2.2.5 Delete reward package

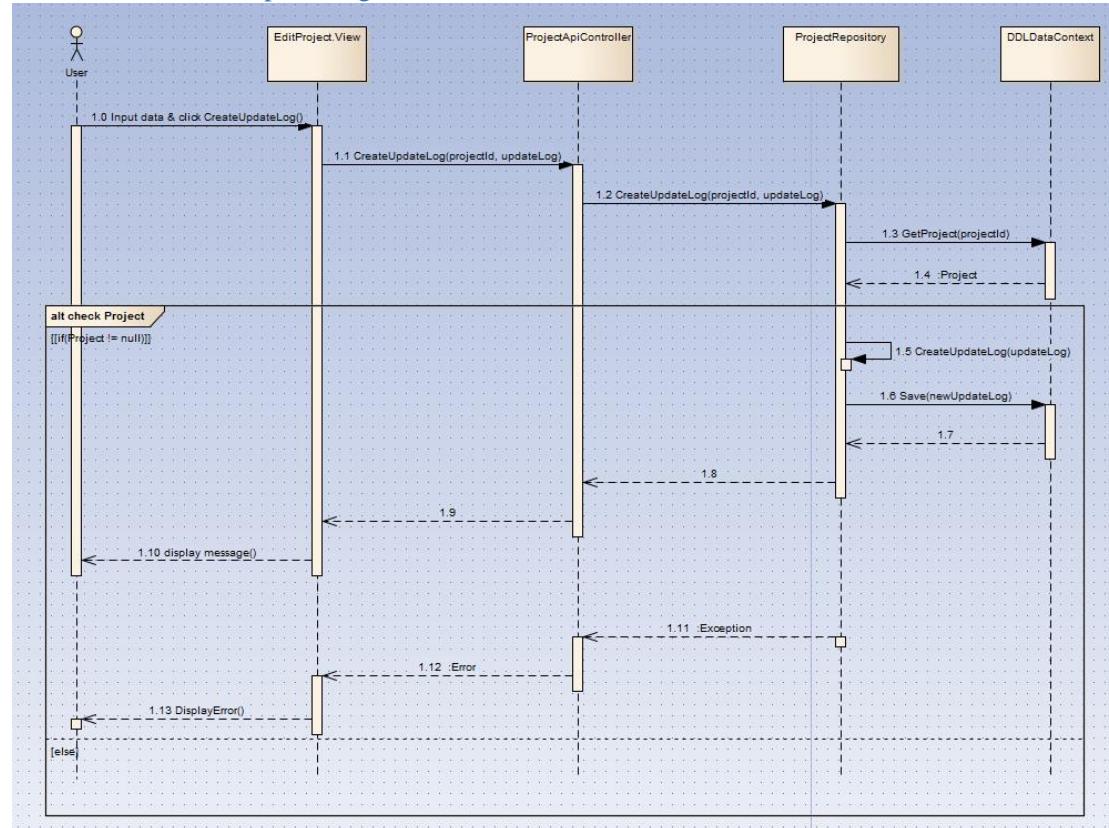


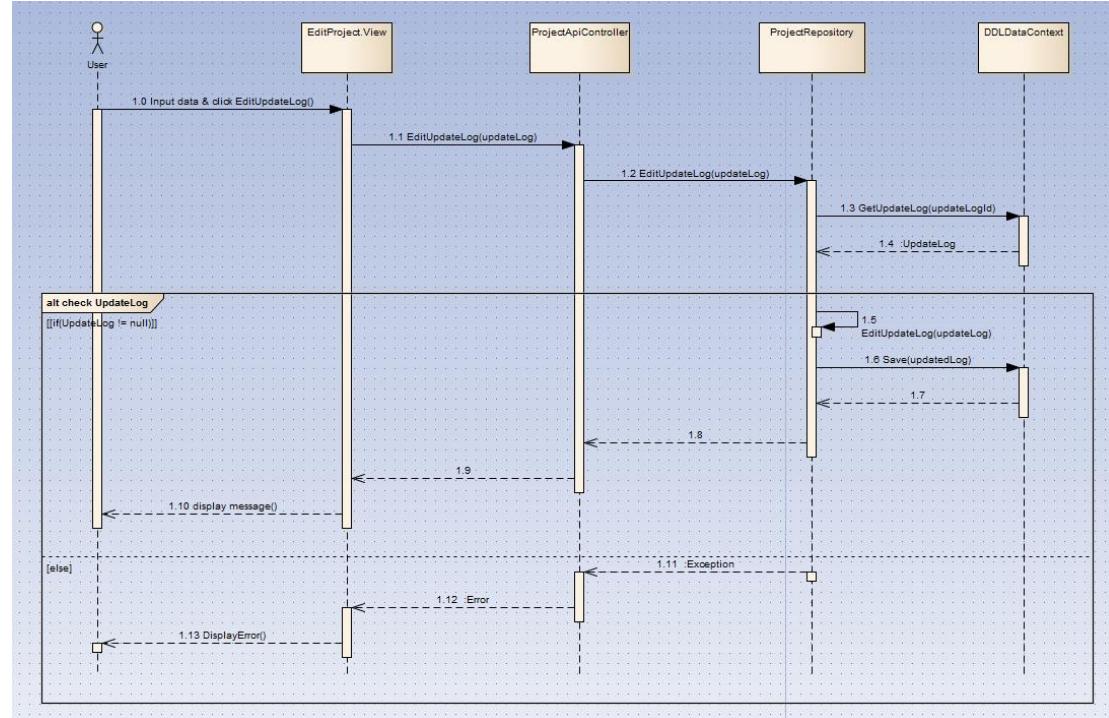
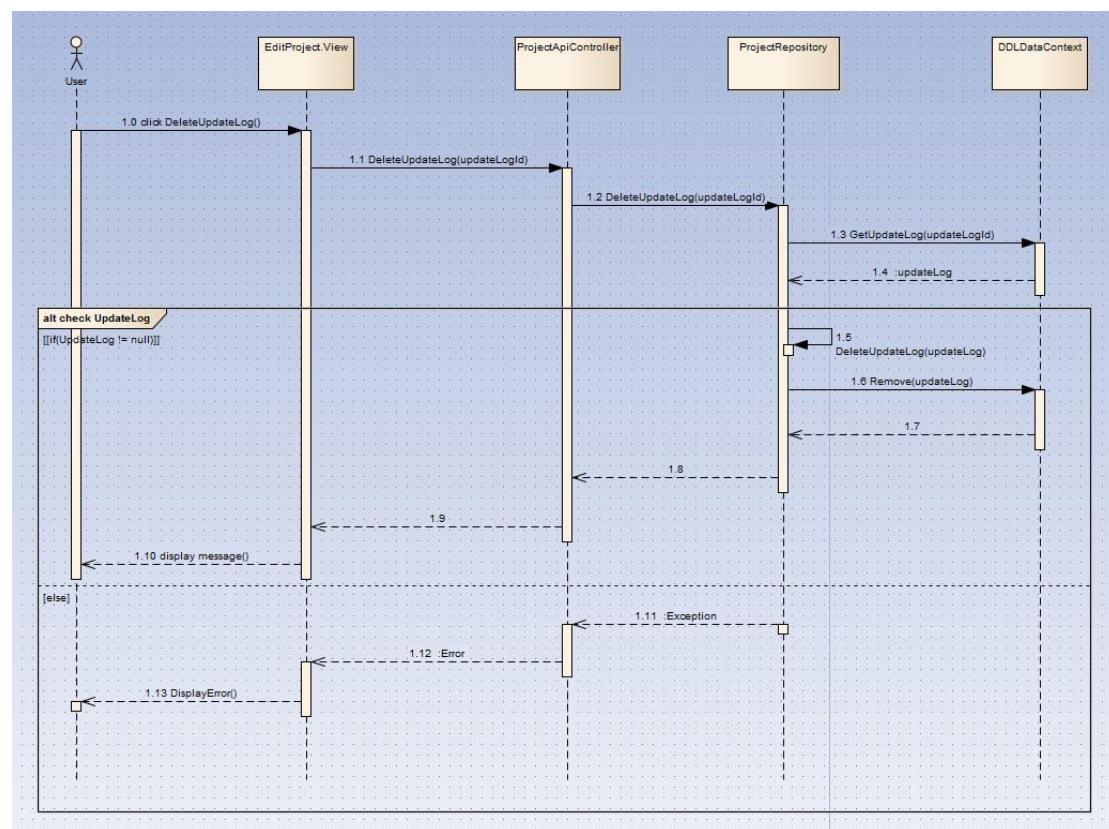
**Figure 4-21:** Delete reward packet sequence diagram

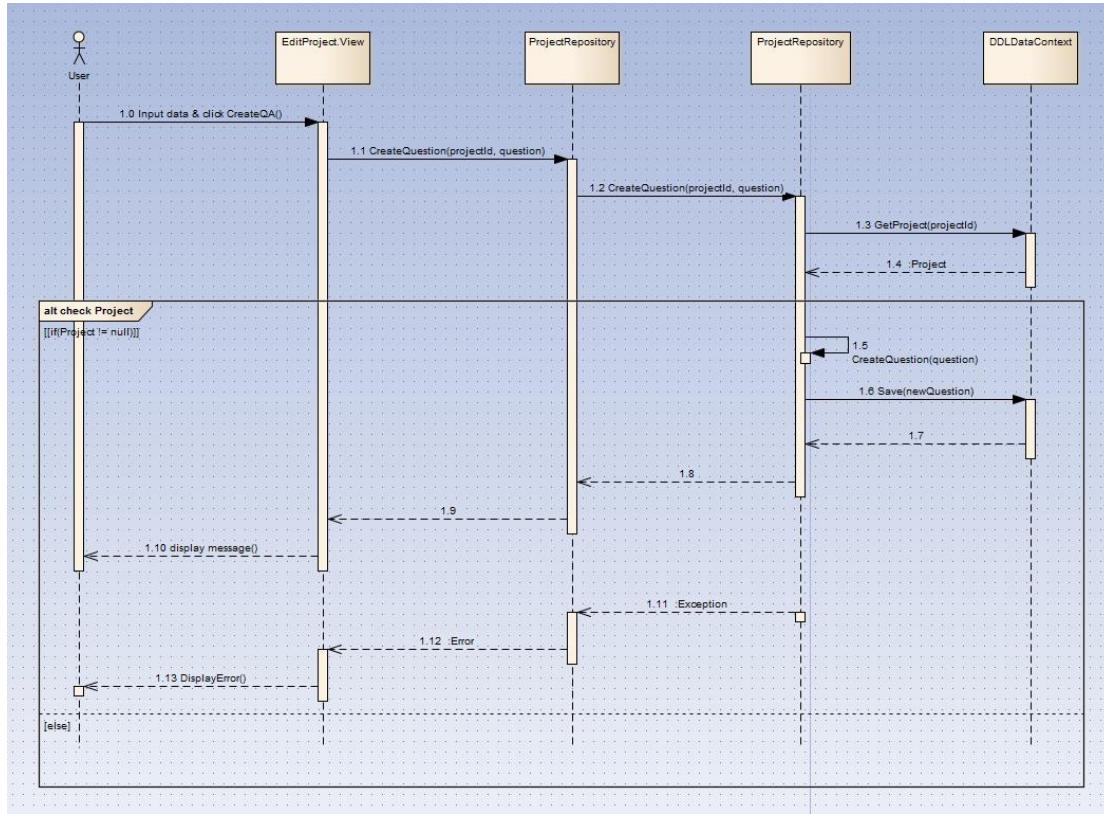
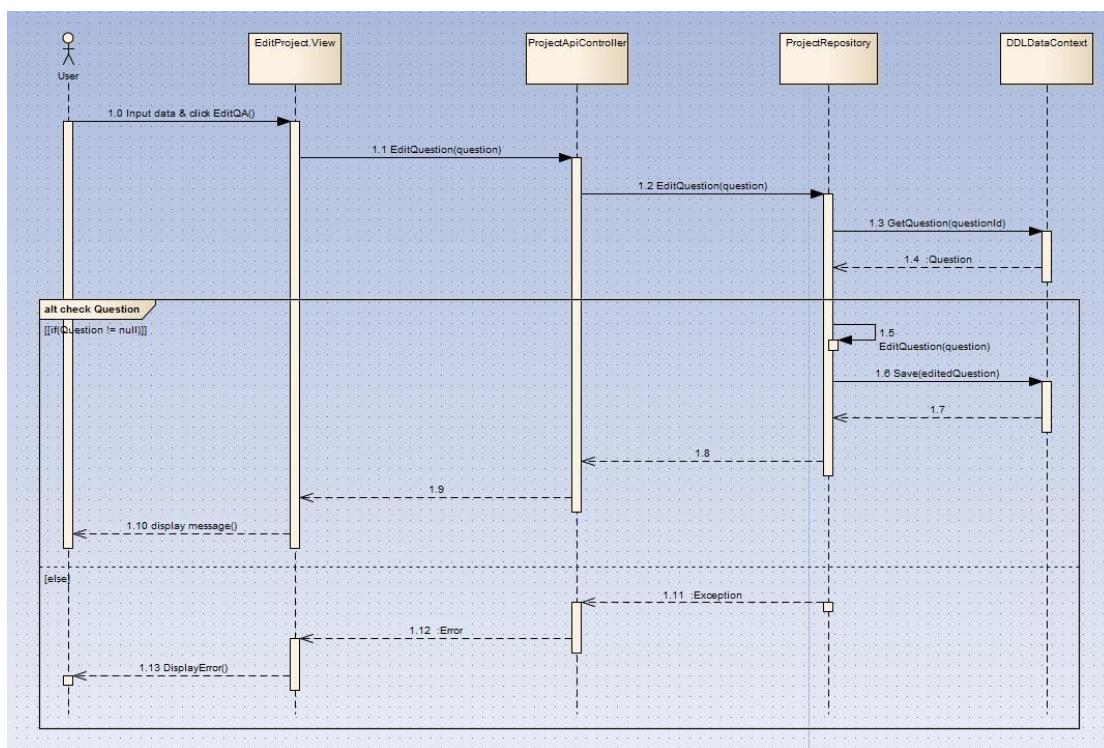
#### 4.3.2.2.6 Edit project story

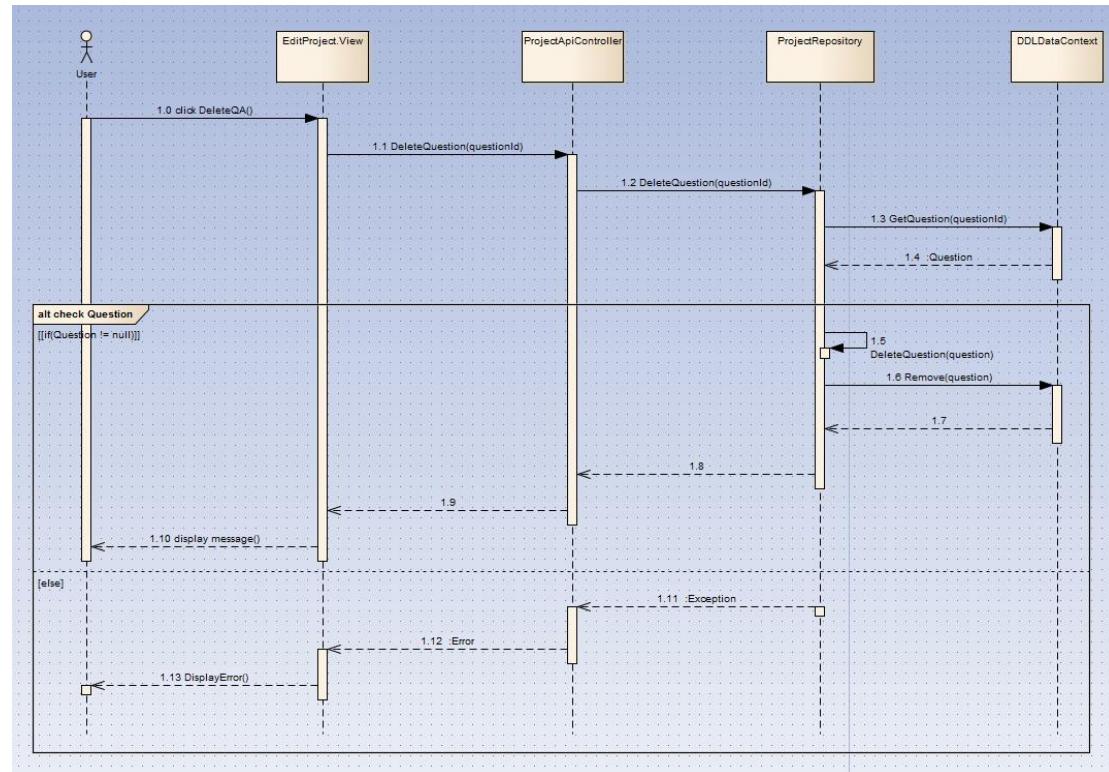
**Figure 4-22:** Edit project story sequence diagram

#### 4.3.2.2.7 Create update log

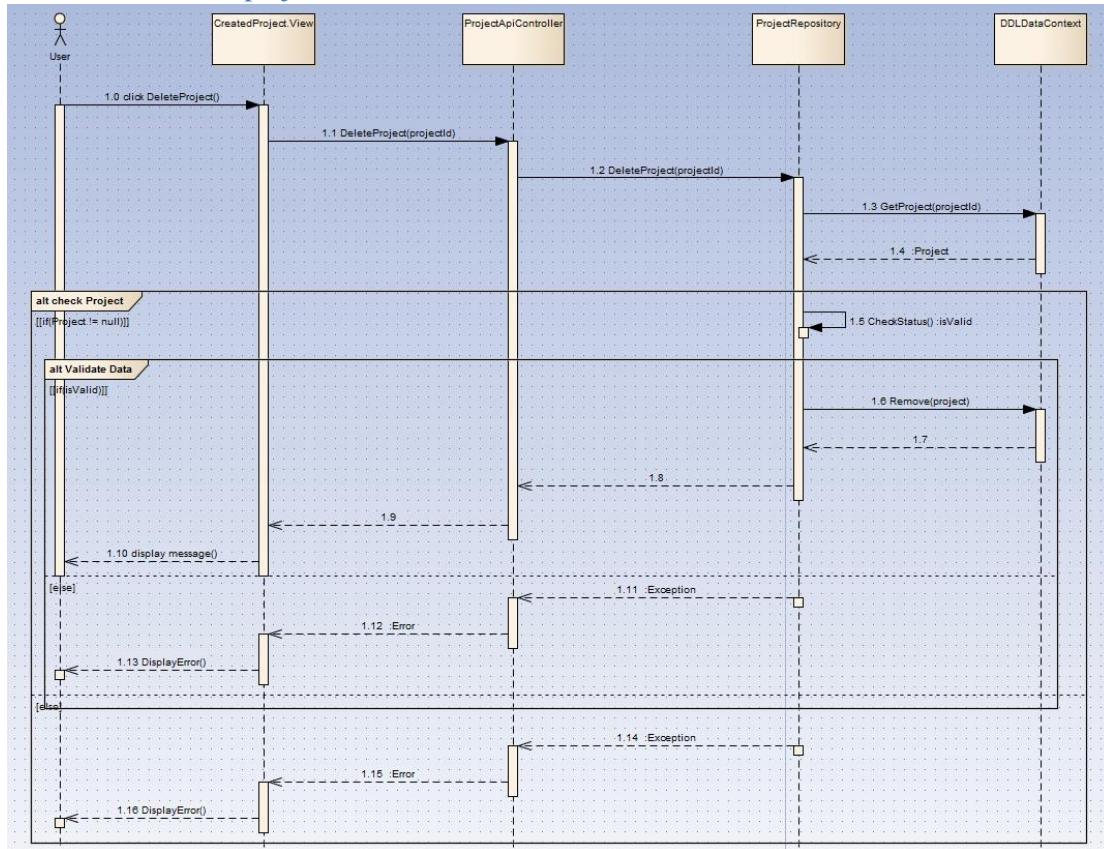


**Figure 4-23:** Create update log sequence diagram**4.3.2.2.8 Edit update log****Figure 4-24:** Edit update log sequence diagram**4.3.2.2.9 Delete update log**

**Figure 4-25:** Delete update log sequence diagram**4.3.2.2.10 Create Q&A****Figure 4-26:** Create Q&A sequence diagram**4.3.2.2.11 Edit Q&A**

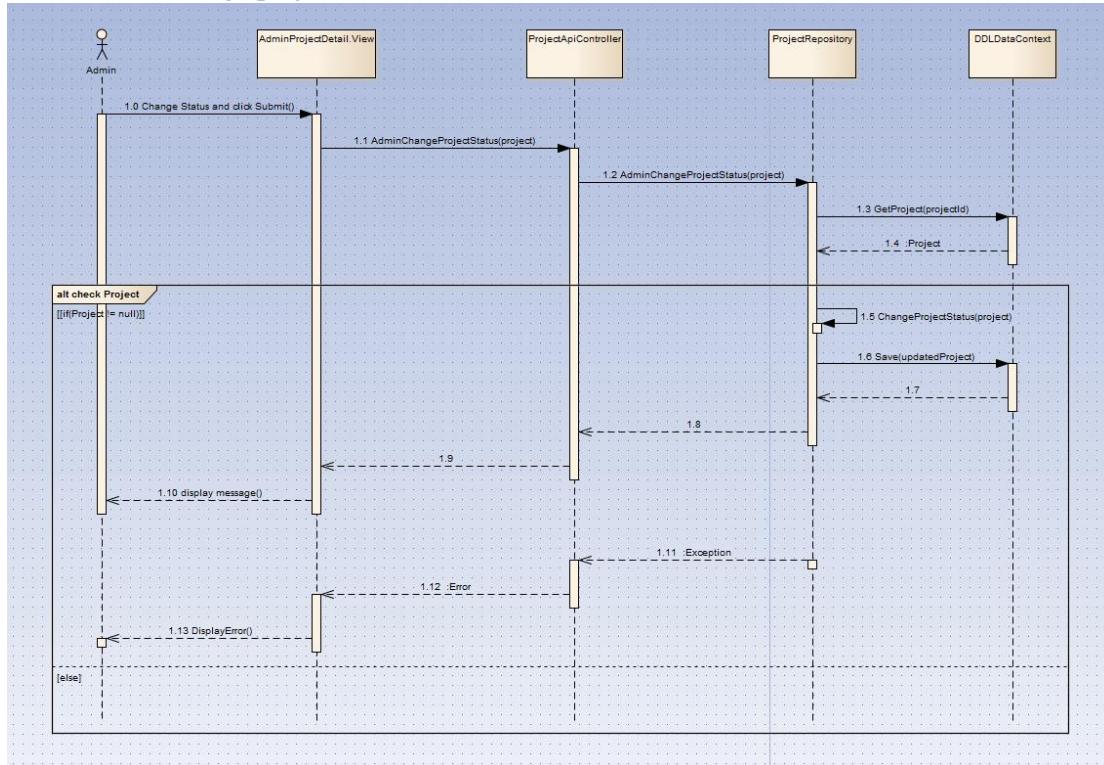
**Figure 4-27: Edit Q&A sequence diagram****4.3.2.2.12 Delete Q&A****Figure 4-28: Delete Q&A sequence diagram**

#### 4.3.2.2.13 Delete project



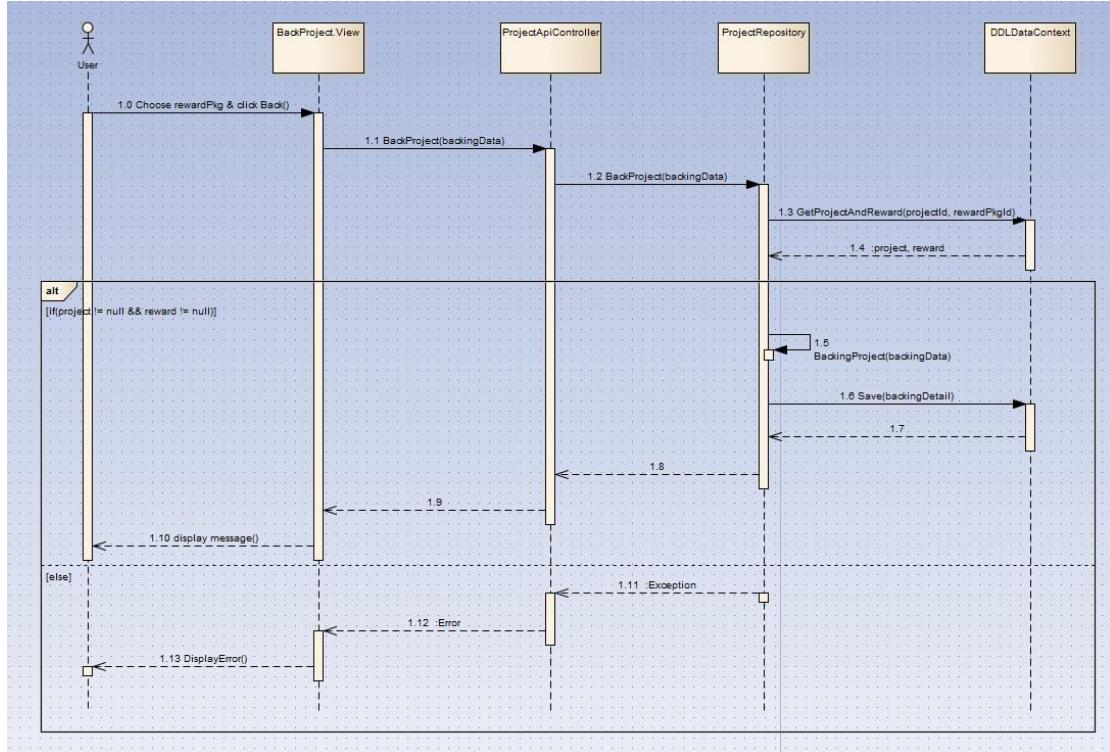
**Figure 4-29: Delete project sequence diagram**

#### 4.3.2.2.14 Change project status

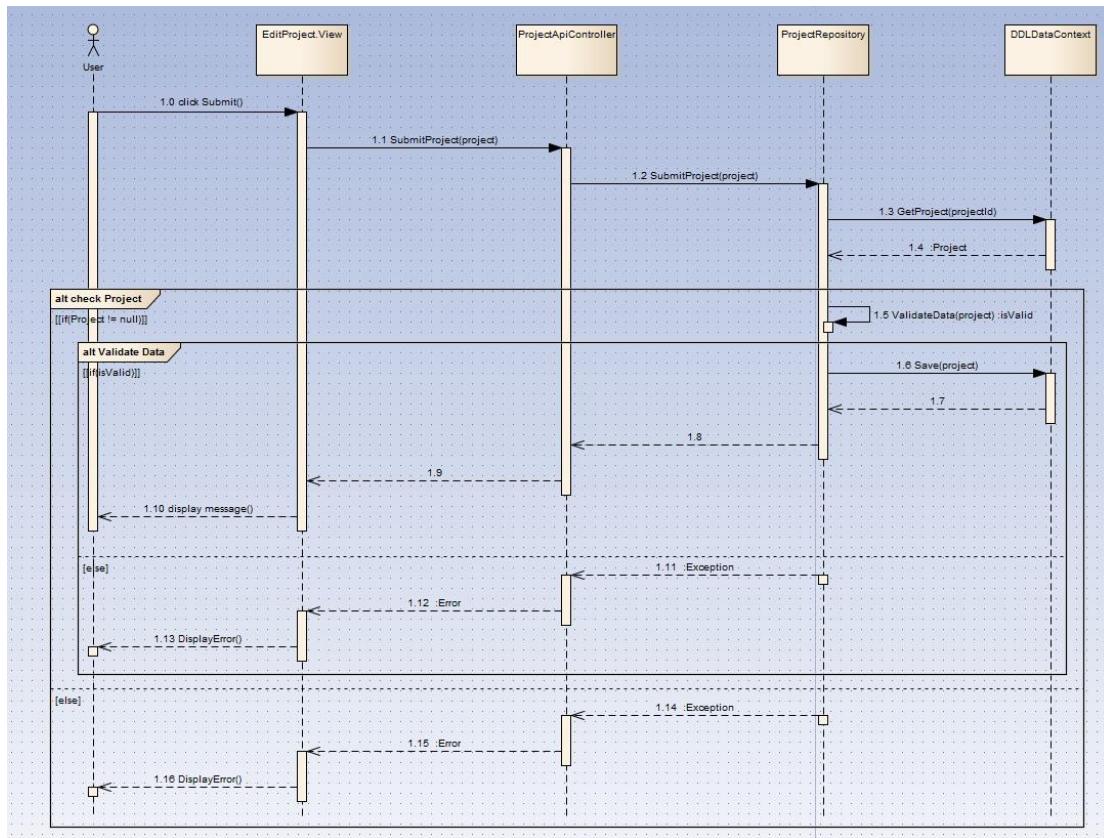


**Figure 4-30:** Change project status sequence diagram

#### 4.3.2.2.15 Back project

**Figure 4-31:** Back project sequence diagram

#### 4.3.2.2.16 Submit project



**Figure 4-32: Submit project sequence diagram**

#### 4.3.2.2.17 Remind Project

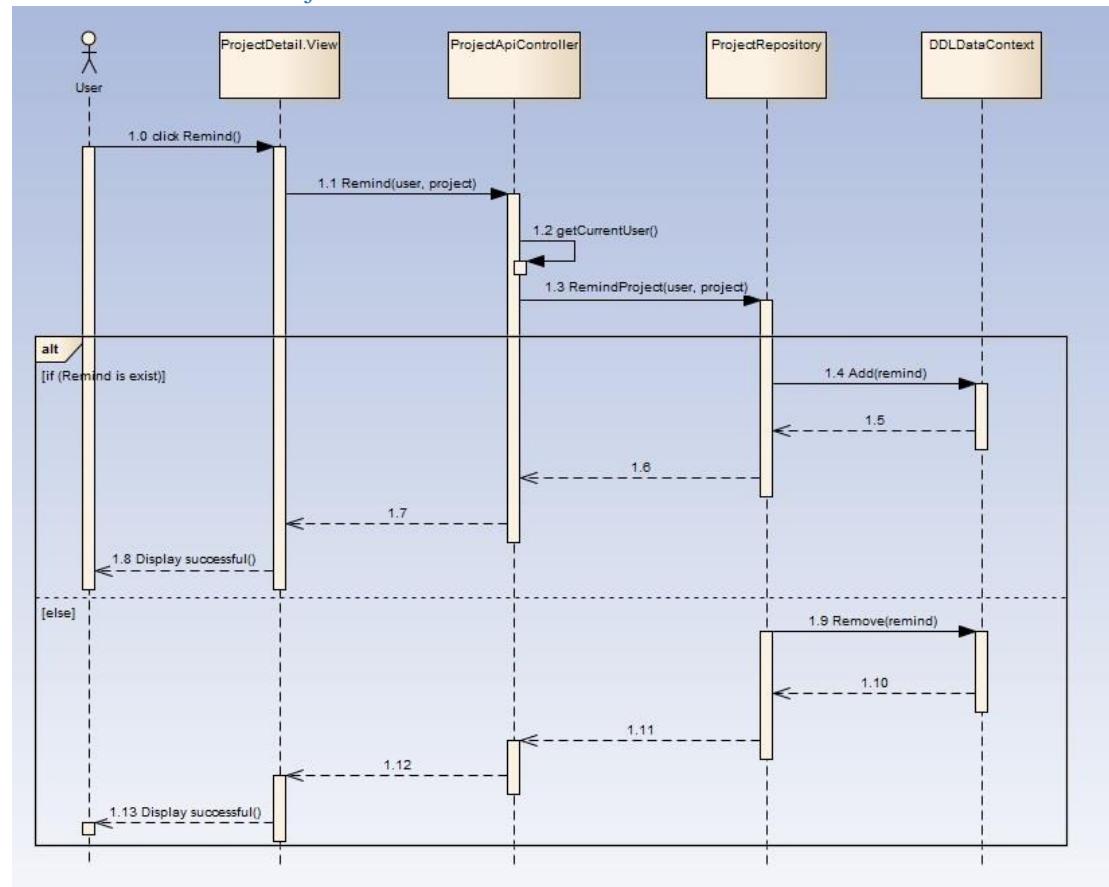


Figure 4-33: Remind project sequence diagram

### 4.3.2.3 Comment

#### 4.3.2.3.1 Class diagram

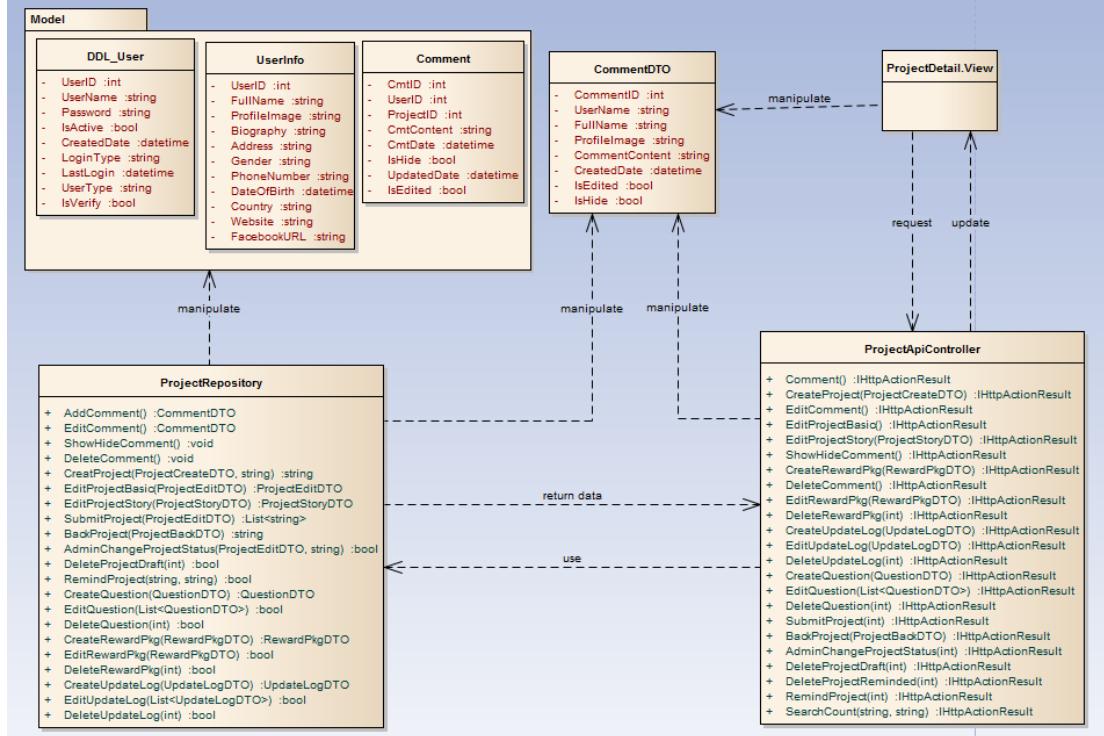


Figure 4-34: Comment function class diagram

#### 4.3.2.3.2 Sequence diagram

##### 4.3.2.3.2.1 Add new comment

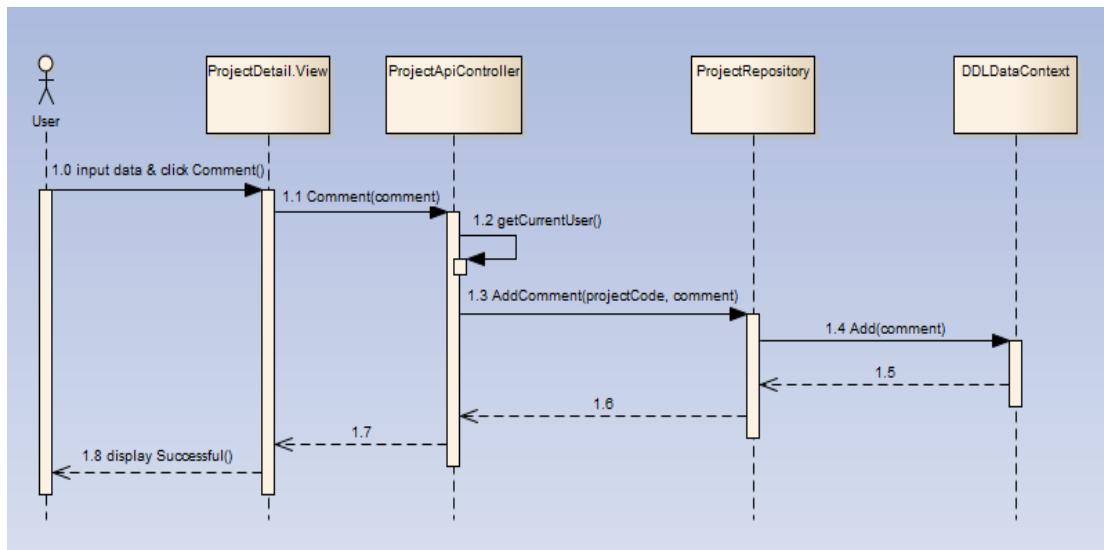
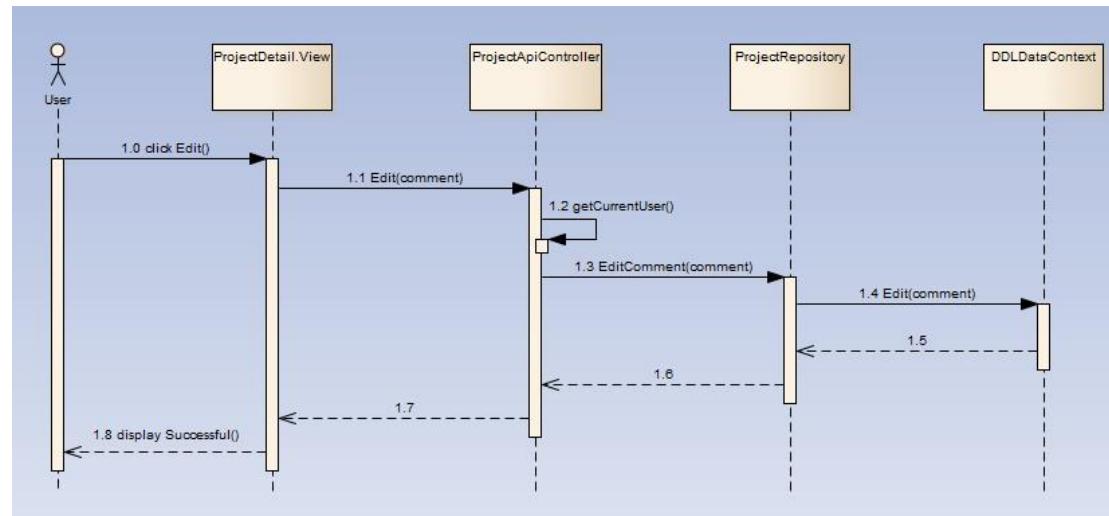


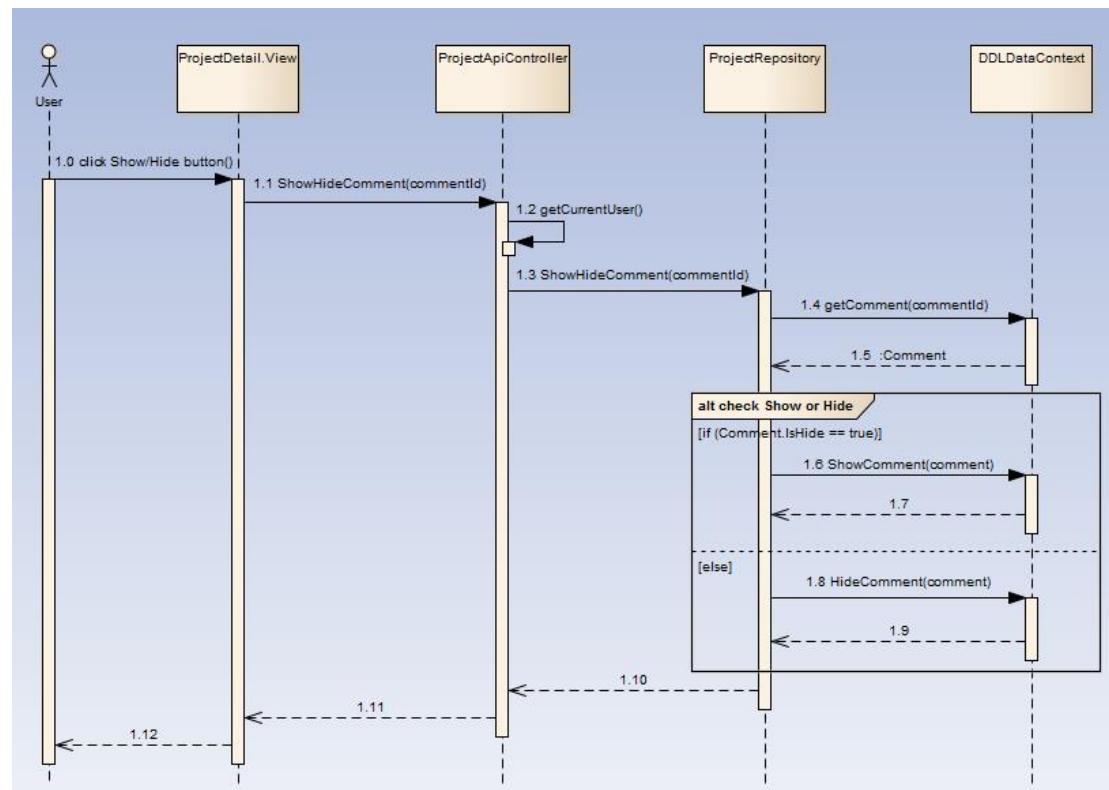
Figure 4-35: Add new comment sequence diagram

#### 4.3.2.3.2.2 Edit comment



**Figure 4-36: Edit comment sequence diagram**

#### 4.3.2.3.2.3 Show/hide comment



**Figure 4-37: Show/hide comment sequence diagram**

#### 4.3.2.3.2.4 Delete comment

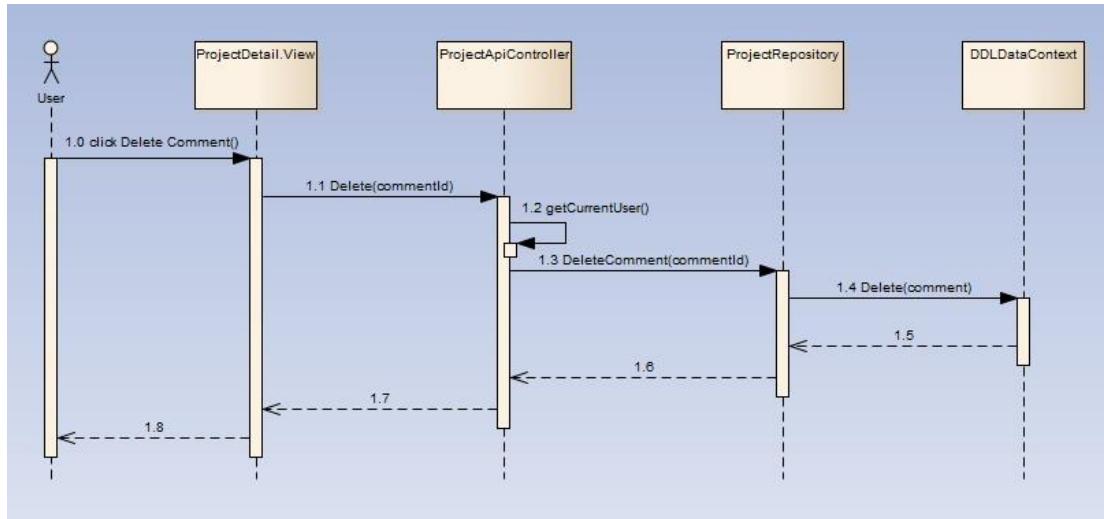


Figure 4-38: Delete comment sequence diagram

#### 4.3.2.4 Message

##### 4.3.2.4.1 Class diagram

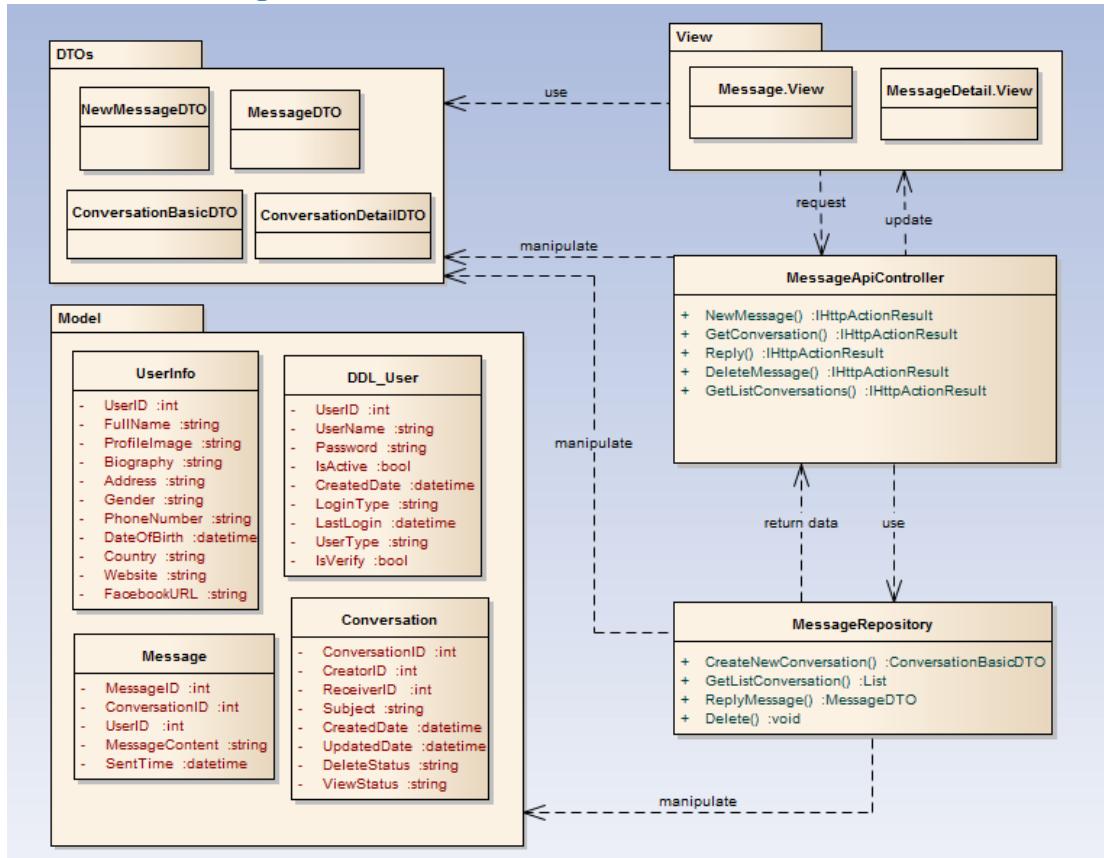


Figure 4-39: Message function class diagram

#### 4.3.2.4.2 Sequence diagram

##### 4.3.2.4.2.1 Send message

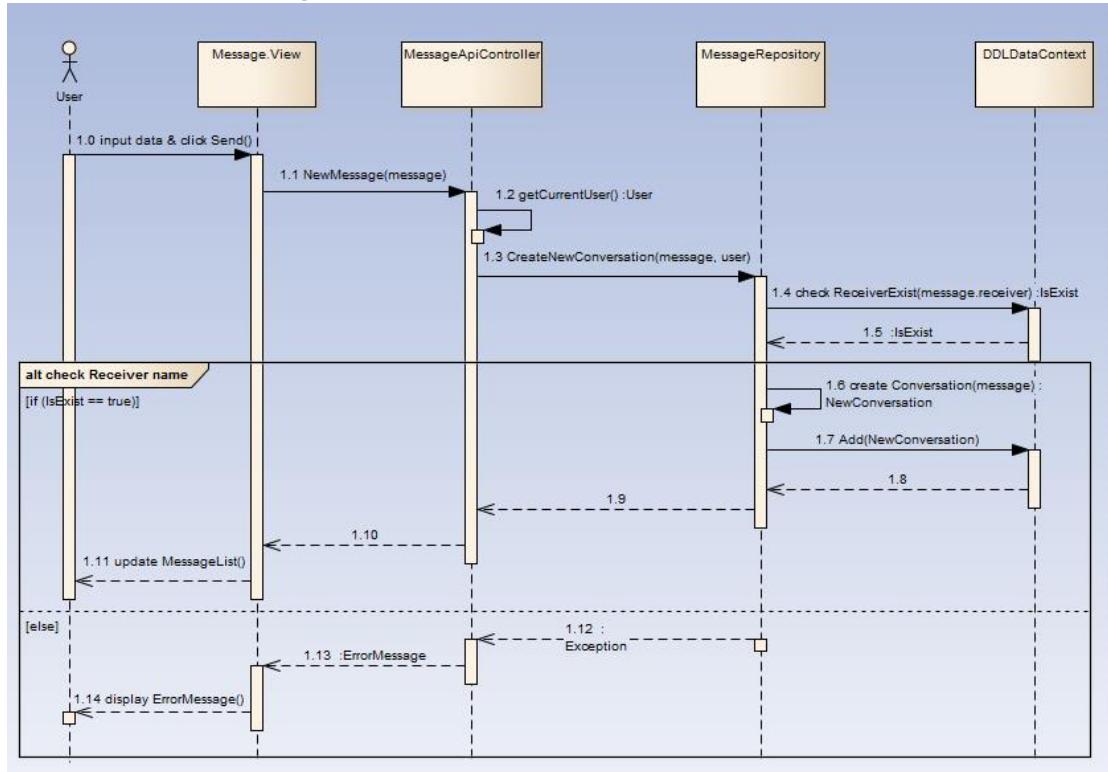


Figure 4-40: Send message sequence diagram

##### 4.3.2.4.2.2 Delete message

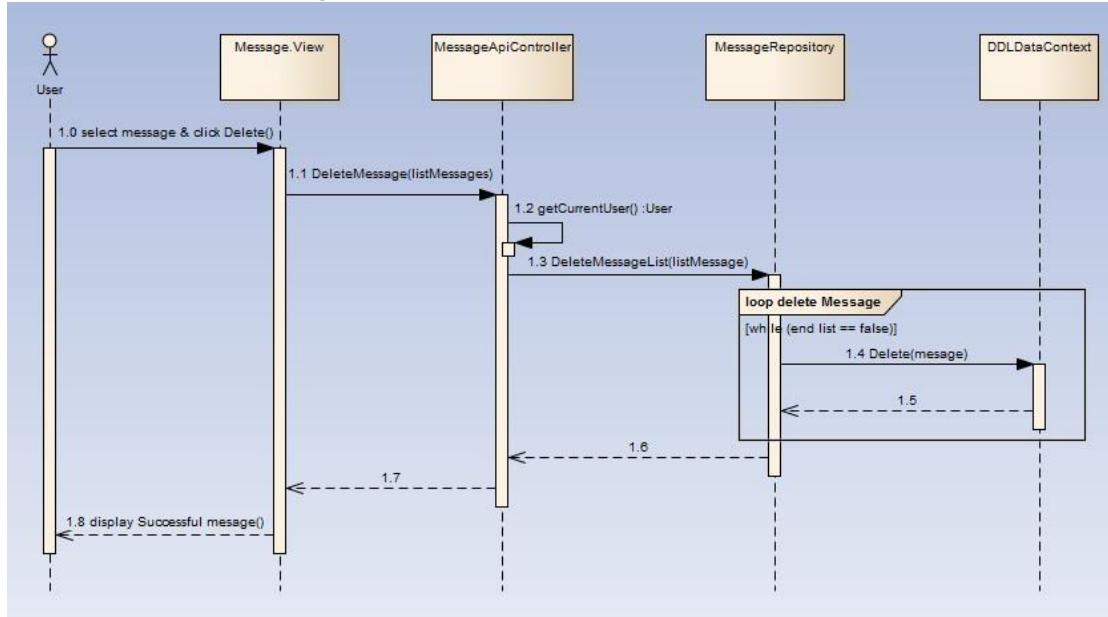


Figure 4-41: Delete message sequence diagram

#### 4.3.2.4.2.3 Reply message

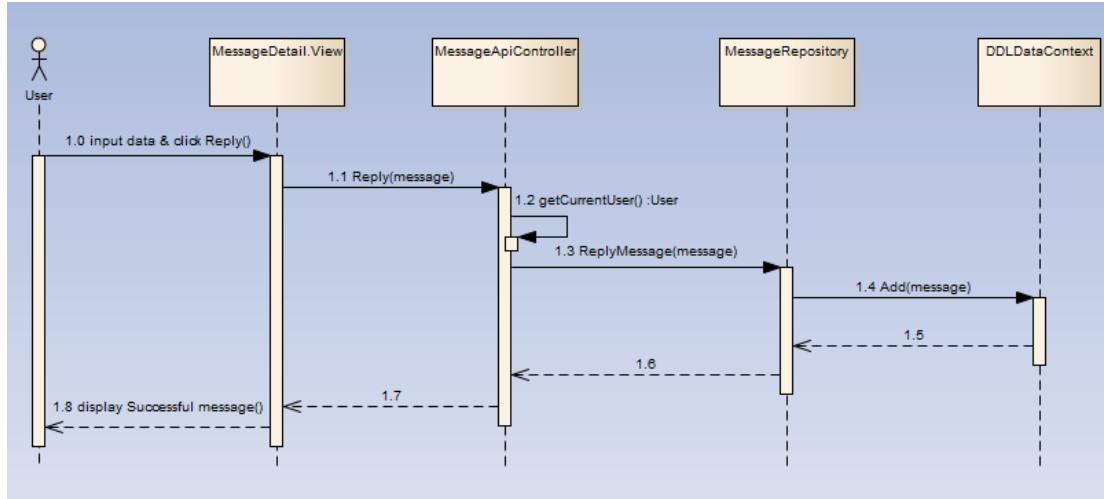


Figure 4-42: Reply message sequence diagram

#### 4.3.2.5 User Management

##### 4.3.2.5.1 Class diagram

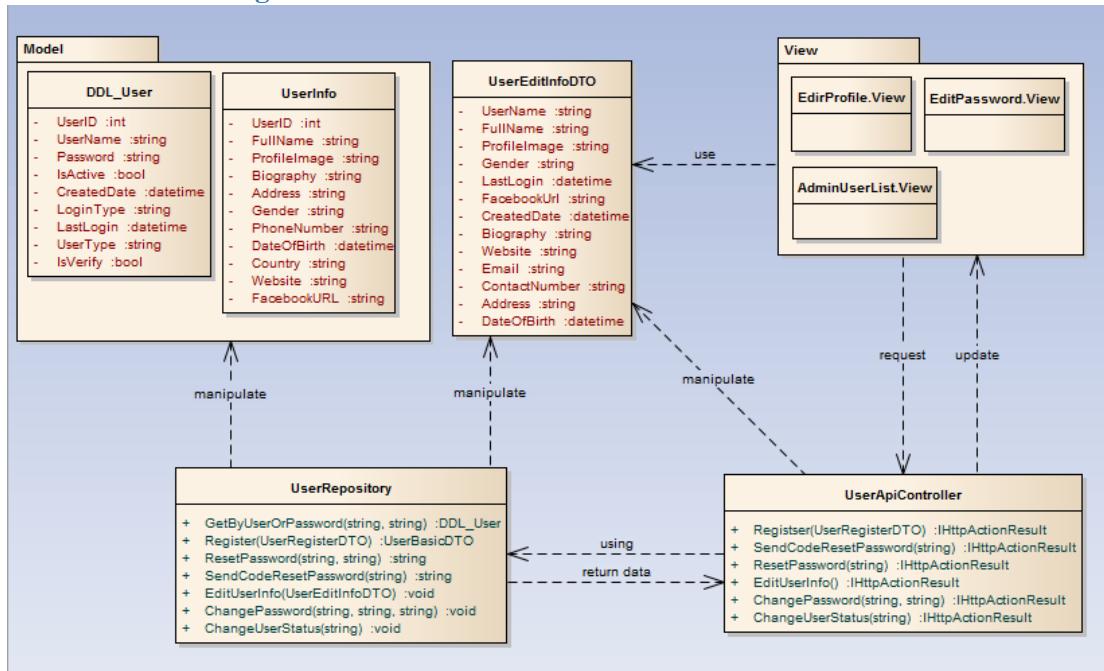


Figure 4-43: User management class diagram

### 4.3.2.5.2 Sequence diagram

#### 4.3.2.5.2.1 Edit profile

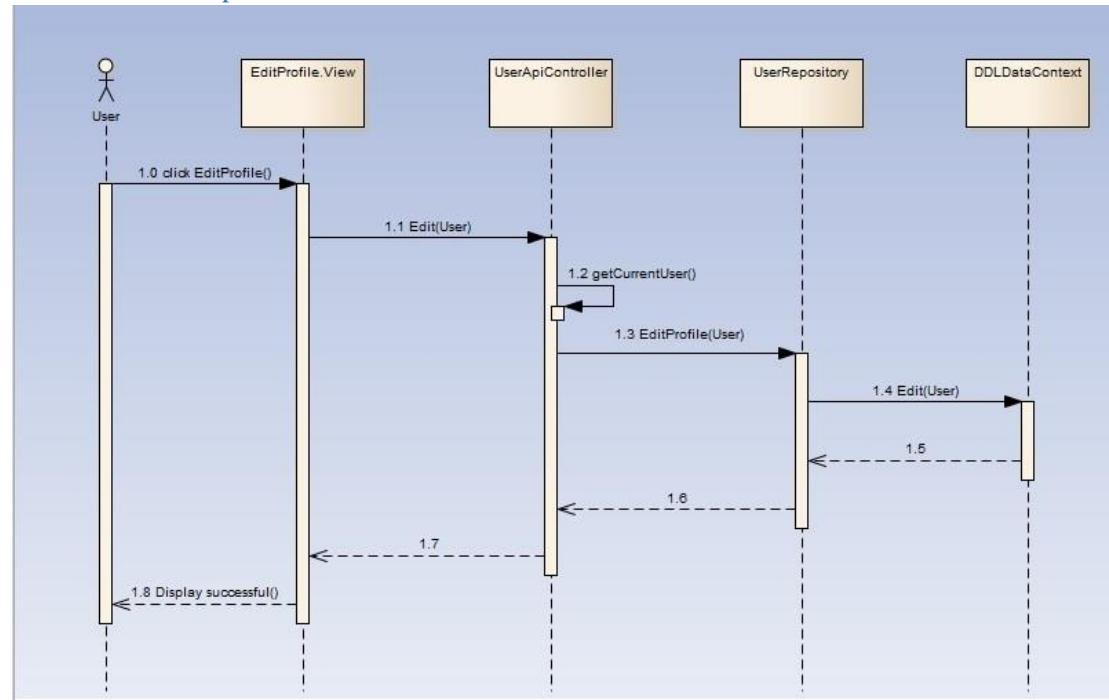


Figure 4-44: Edit profile sequence diagram

#### 4.3.2.5.2.2 Change password

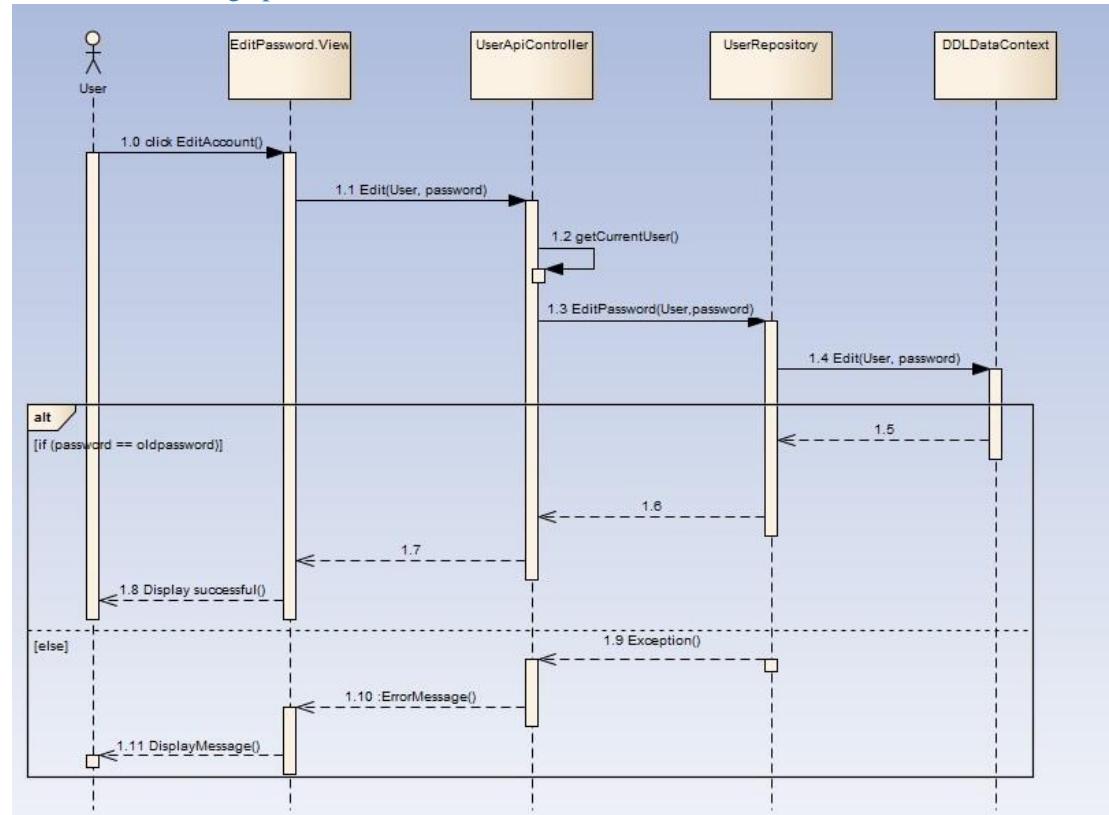
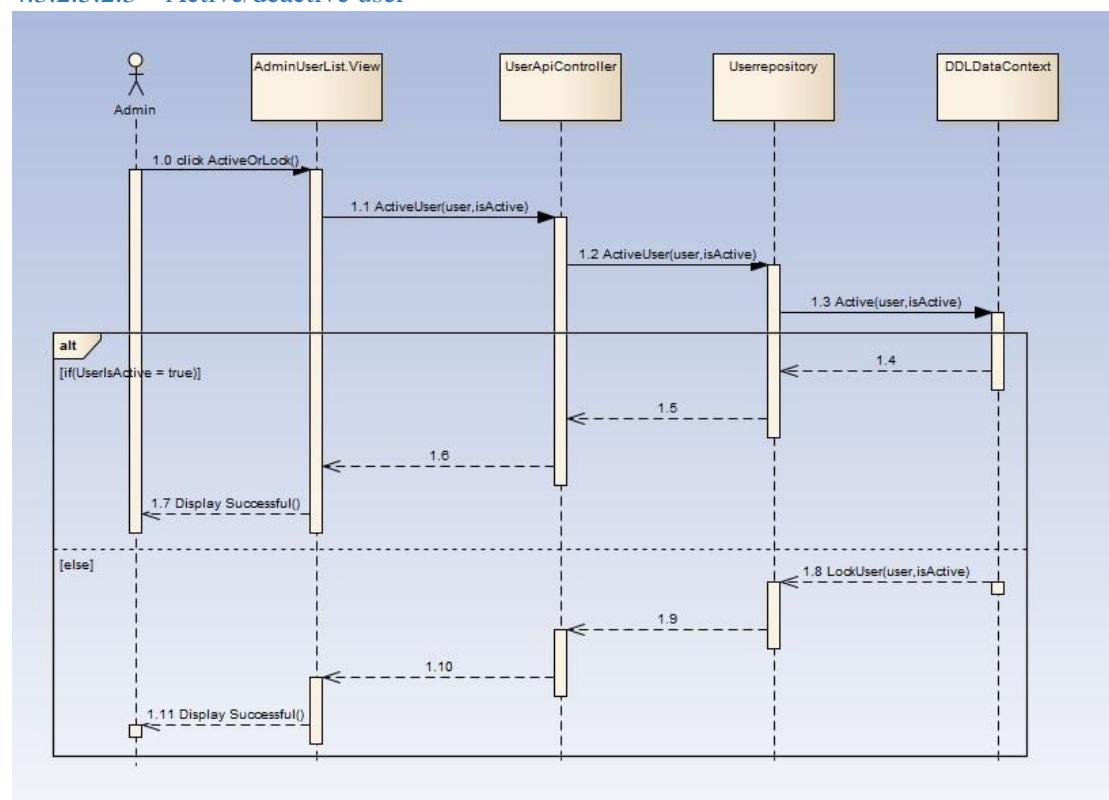


Figure 4-45: Change password sequence diagram

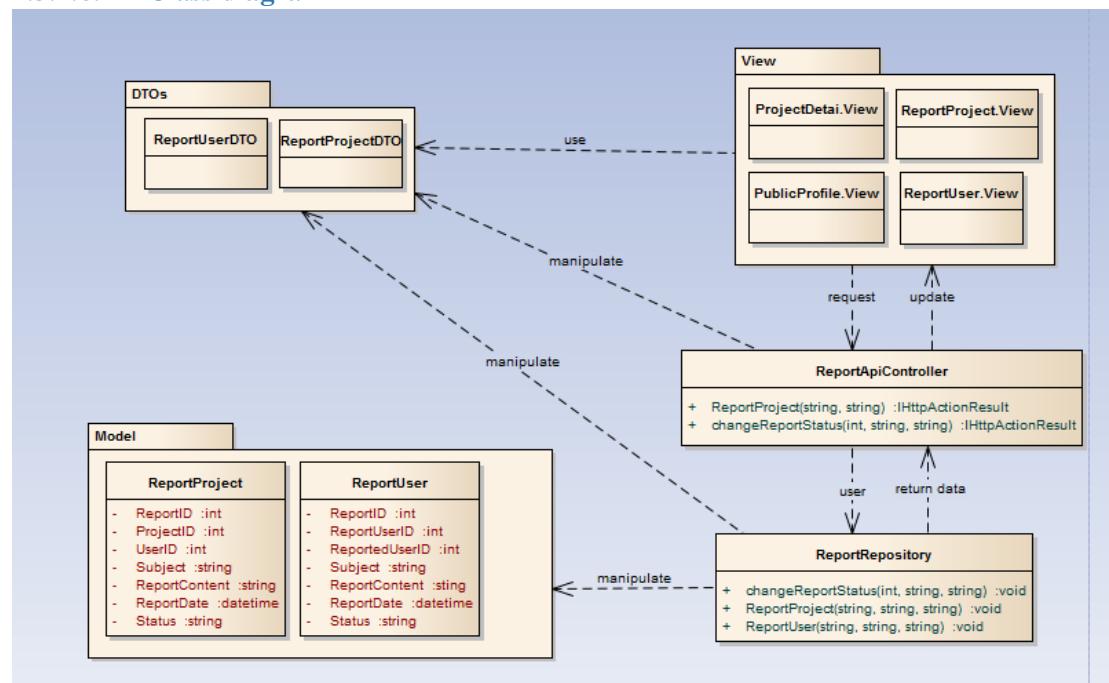
### 4.3.2.5.2.3 Active/deactive user



**Figure 4-46:** Active/ deactivate user sequence diagram

### 4.3.2.6 Report Management

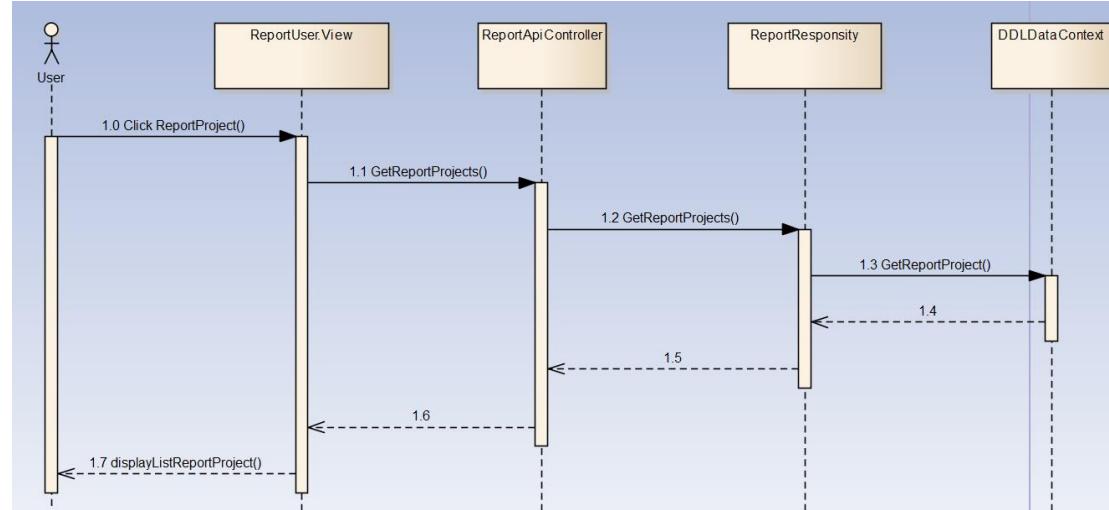
#### 4.3.2.6.1 Class diagram



**Figure 2-1:** Report management class diagram

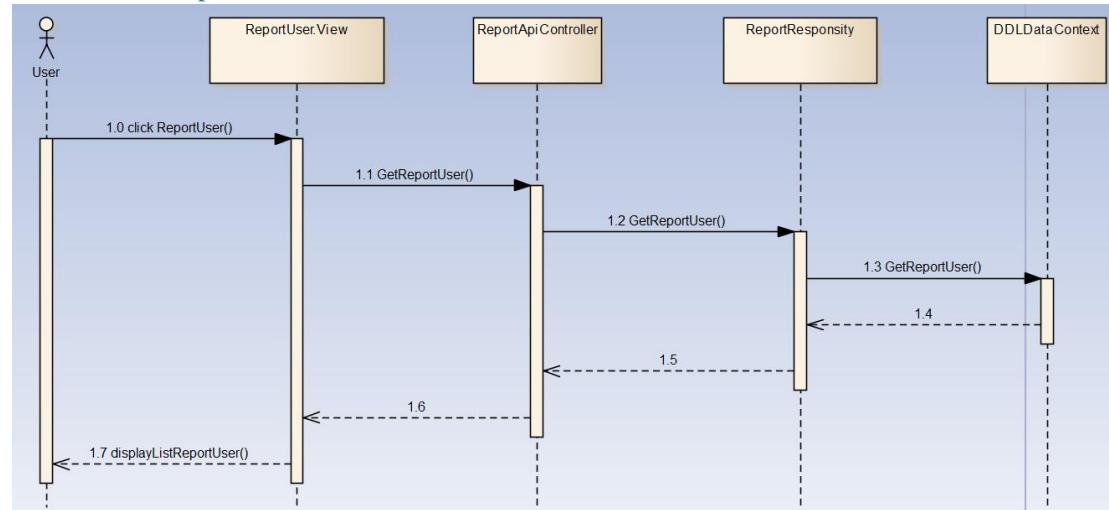
### 4.3.2.6.2 Sequence diagram

#### 4.3.2.6.2.1 Report Project



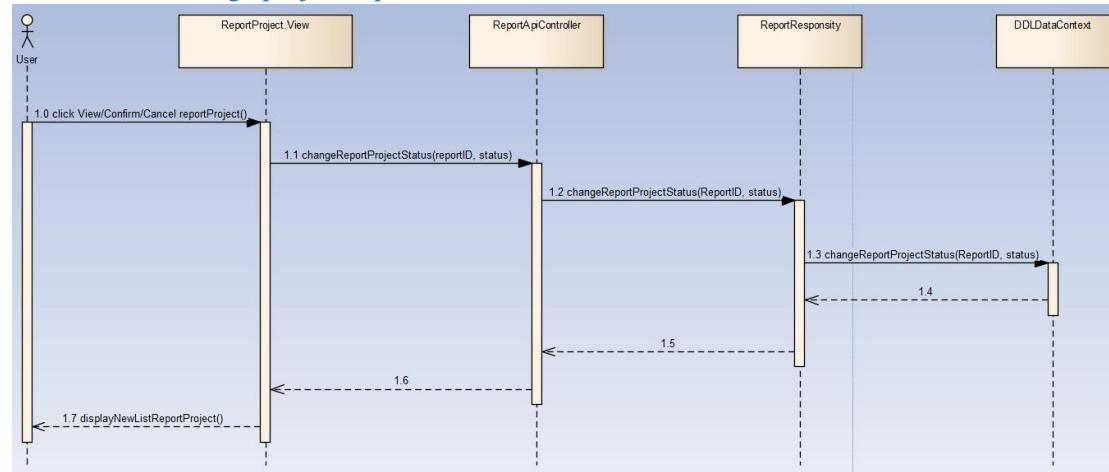
**Figure 4-47: Report project sequence diagram**

#### 4.3.2.6.2.2 Report User



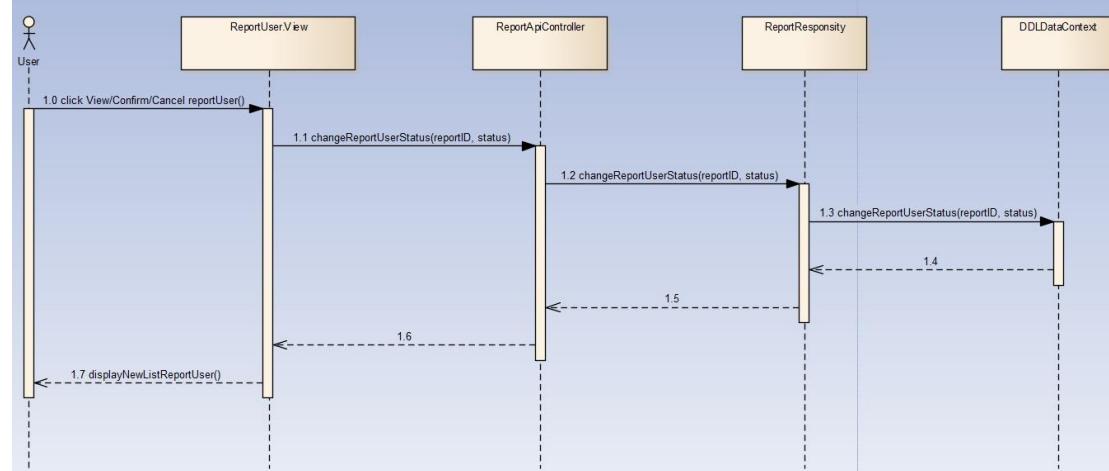
**Figure 4-48: Report user sequence diagram**

#### 4.3.2.6.2.3 Change project report status



**Figure 4-49:** Change project report status sequence diagram

#### 4.3.2.6.2.4 Change user report status



**Figure 4-50:** Change project report status sequence diagram

### 4.3.2.7 Manage Slide

#### 4.3.2.7.1 Class diagram

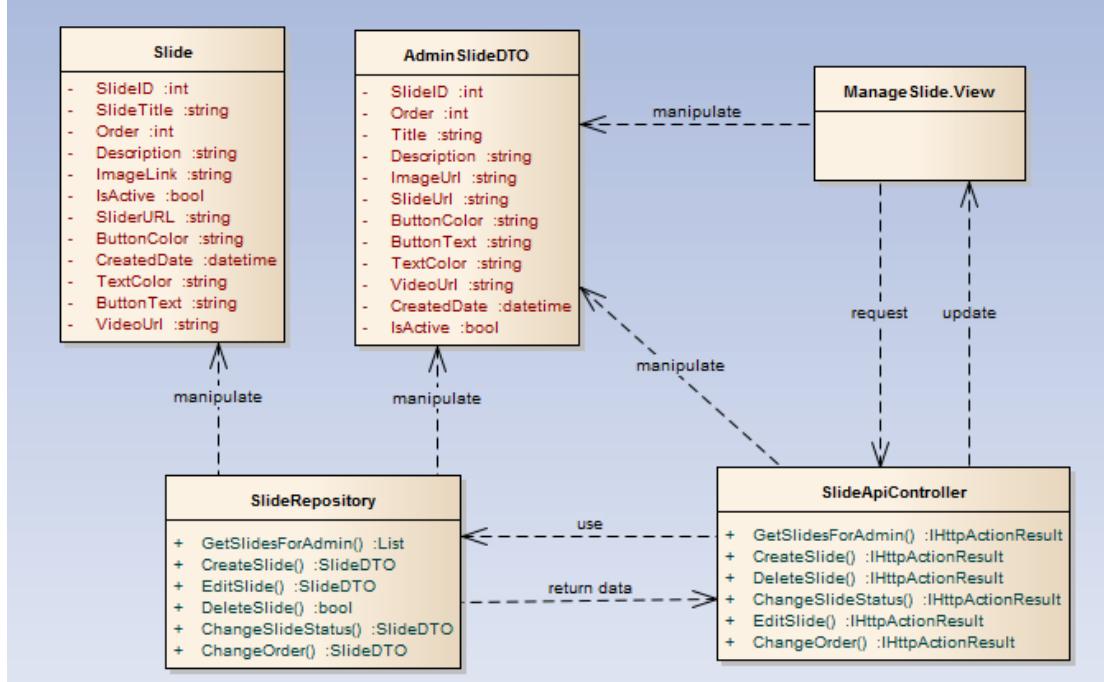


Figure 2-2: Manage Slide class diagram

#### 4.3.2.7.2 Sequence diagram

##### 4.3.2.7.2.1 Add slide

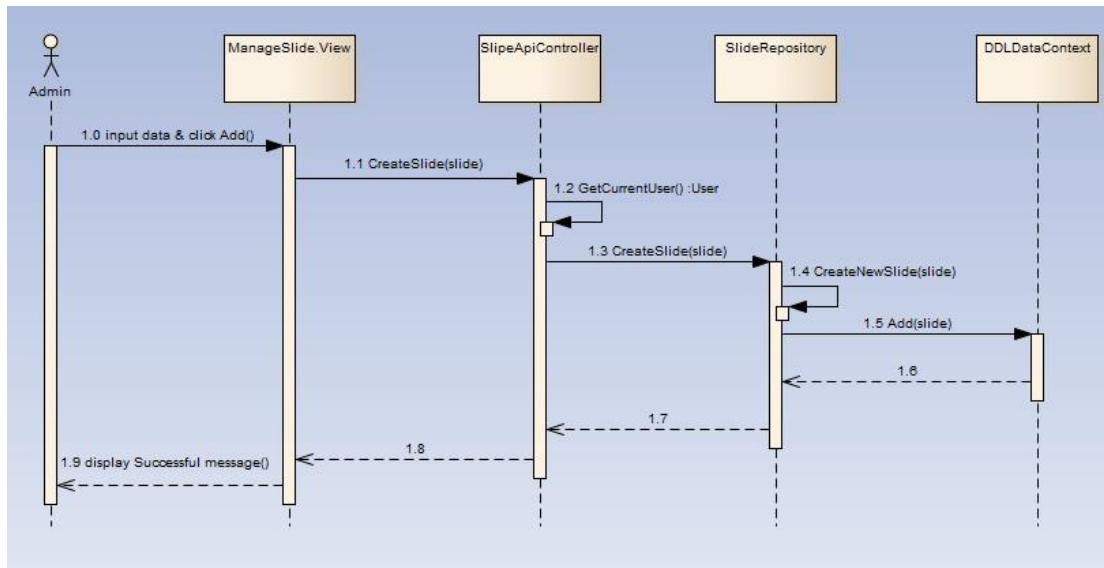


Figure 4-51: Add slide sequence diagram

#### 4.3.2.7.2.2 Edit slide

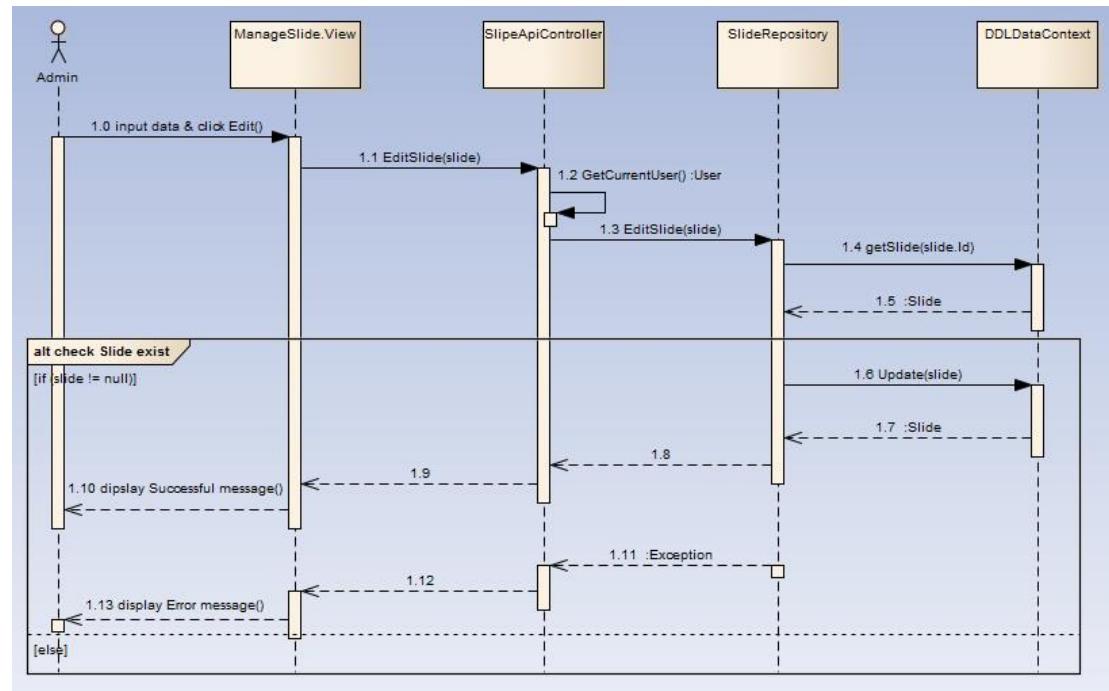


Figure 4-52: Edit slide sequence diagram

#### 4.3.2.7.2.3 Delete slide

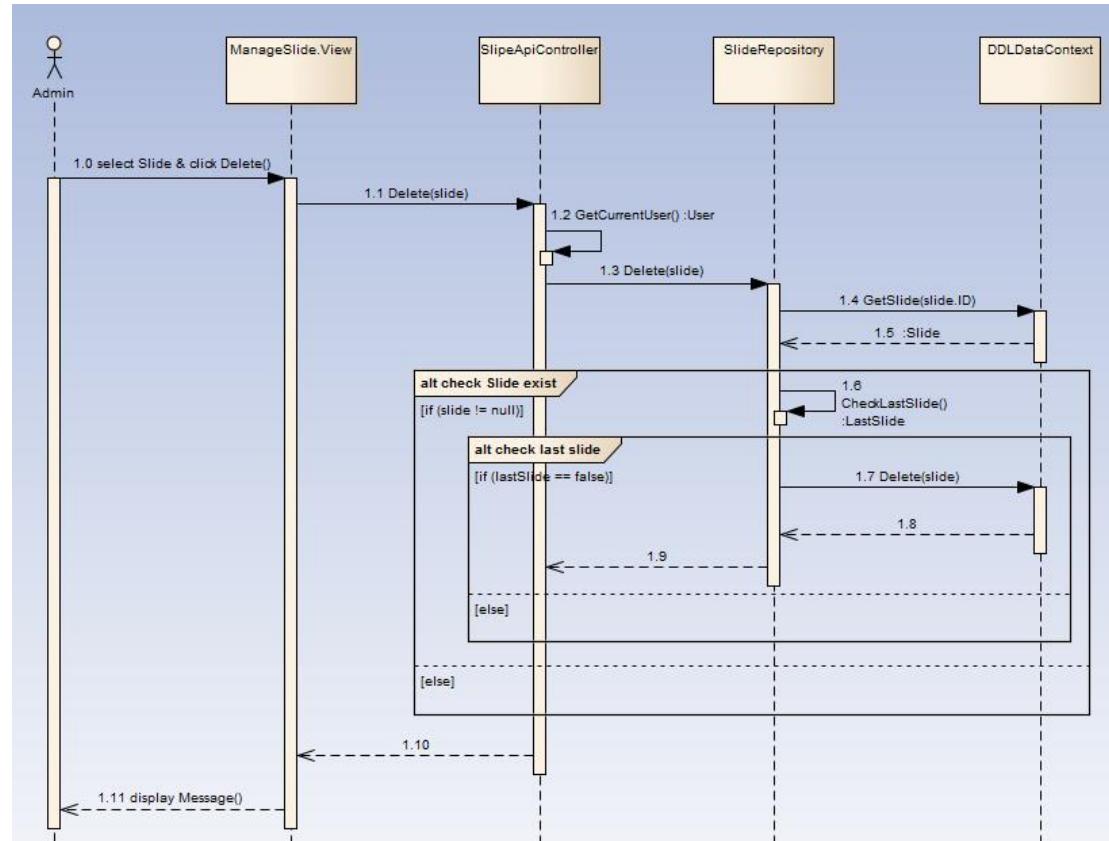
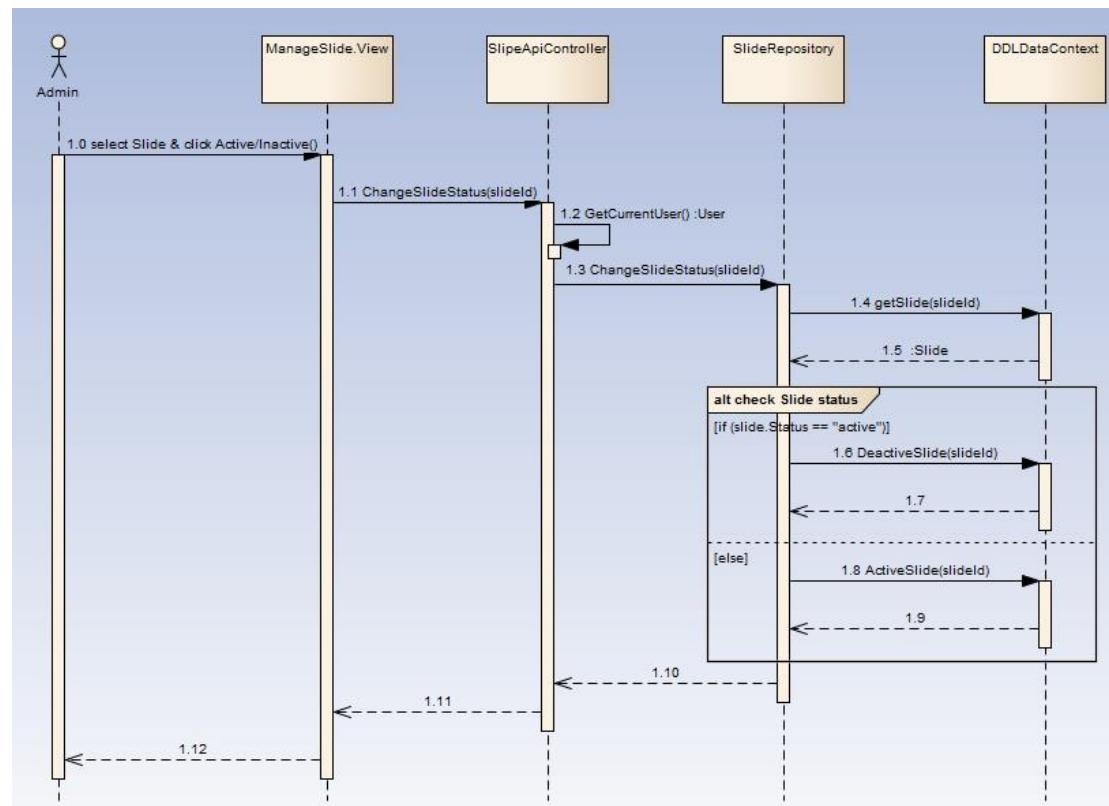


Figure 4-53: Delete slide sequence diagram

#### 4.3.2.7.2.4 Active/deactive slide



**Figure 4-54:** Active/deactive slide sequence diagram

#### 4.3.2.7.2.5 Change slide order

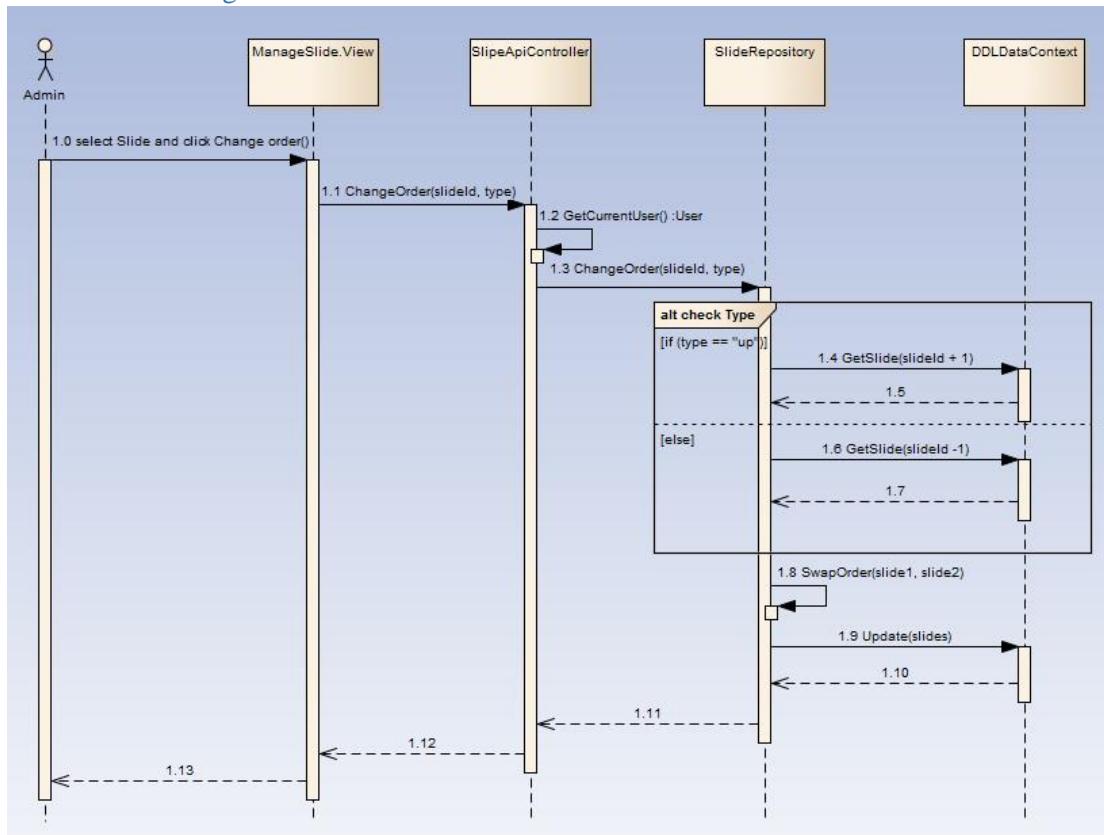


Figure 4-55: Change slide order sequence diagram

#### 4.3.2.8 Manage Category

##### 4.3.2.8.1 Class diagram

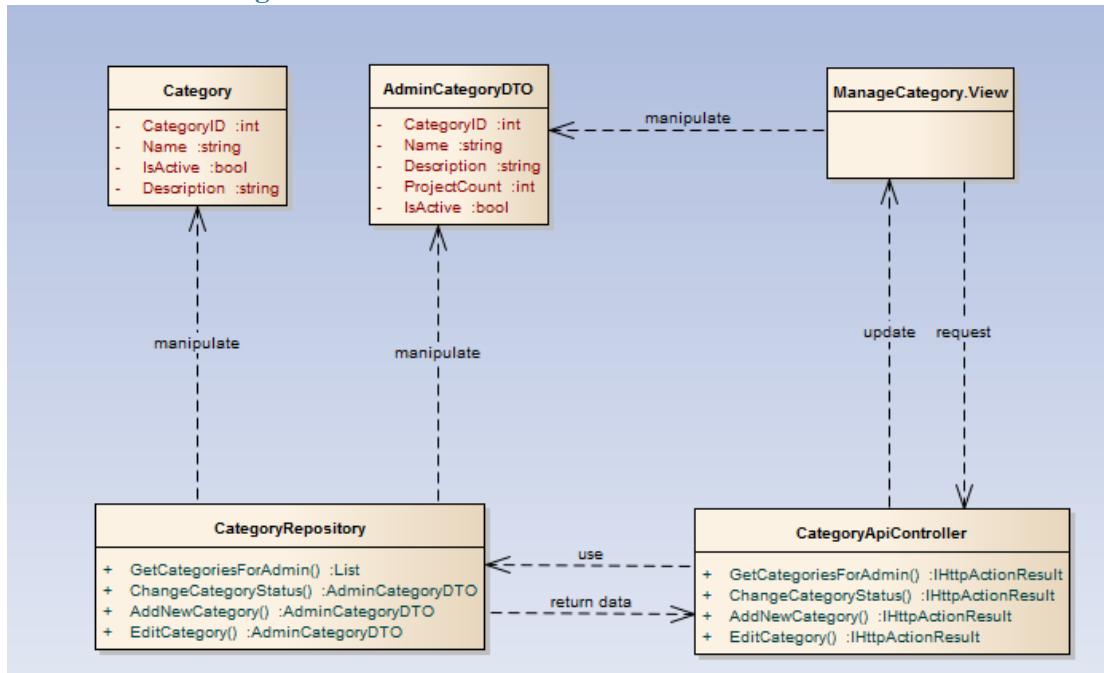


Figure 2-3: Manage category class diagram

### 4.3.2.8.2 Sequence diagram

#### 4.3.2.8.2.1 Add category

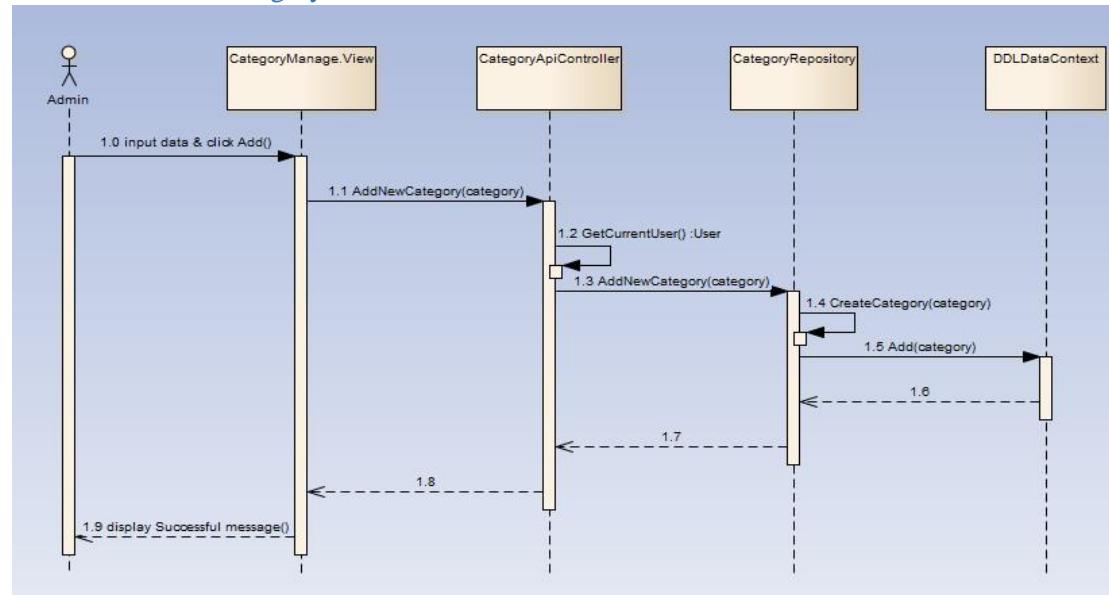


Figure 4-56: Add category sequence diagram

#### 4.3.2.8.2.2 Edit category

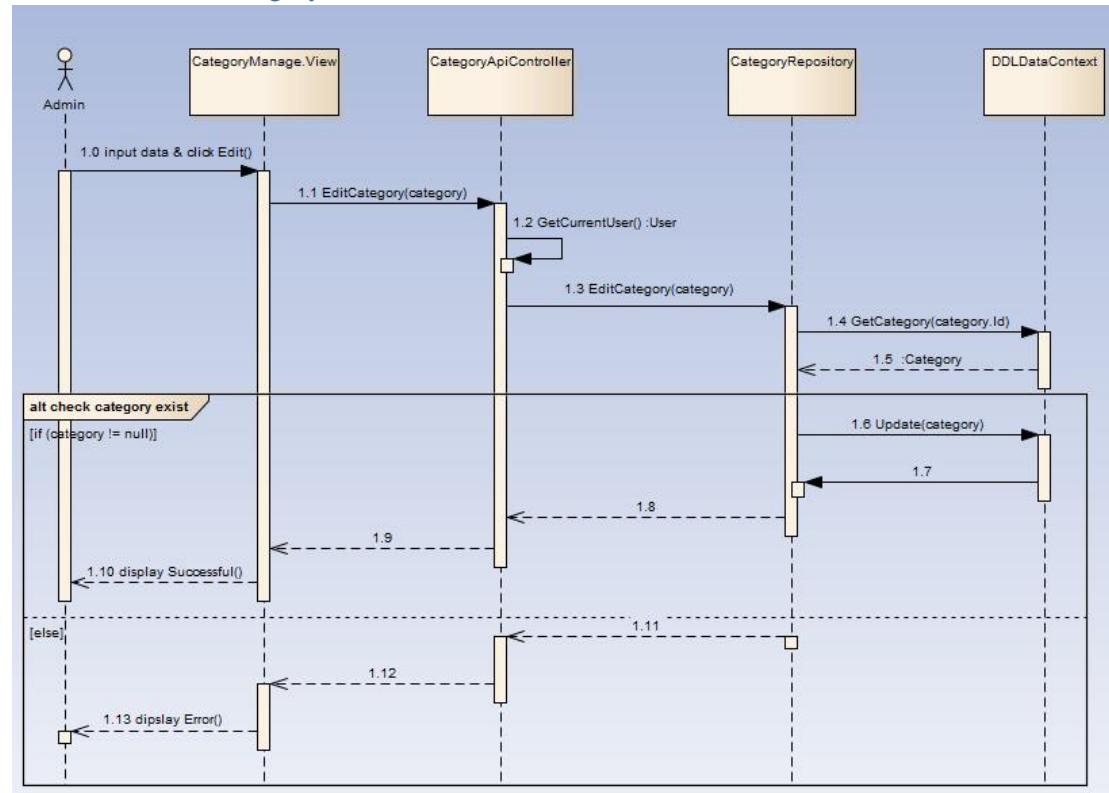
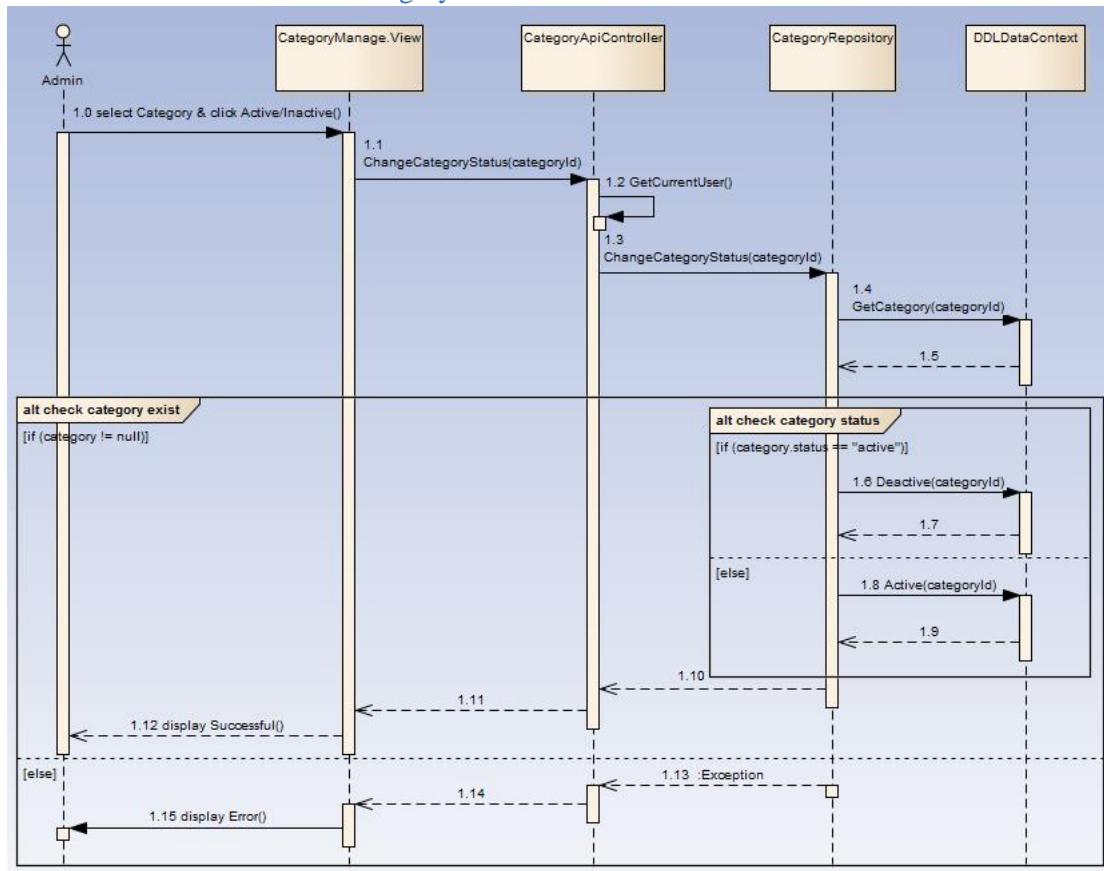


Figure 4-57: Edit category sequence diagram

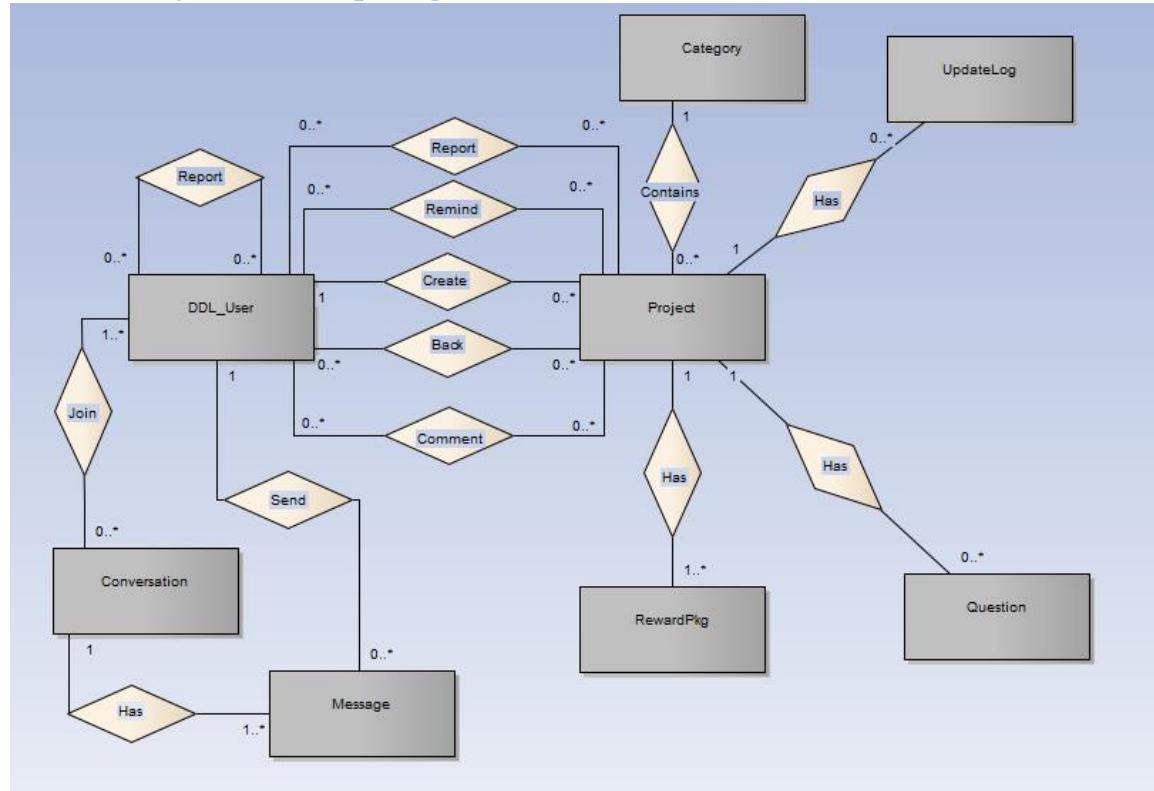
### 4.3.2.8.2.3 Active/deactive category



**Table 4-37:Active/deactive category sequence diagram**

## 4.4 Data design

#### 4.4.1 Entity Relationship Diagram



**Figure 4-58:** Entity Relationship Diagram

#### 4.4.1.1 Entity description

No	Entity	Description
1	DDL_User	User is all people who using DDL feature
2	Project	A project which user created, includes content, schedule, photo, video to describing.
3	Category	A category which user use to filter project
4	Update	A Update which user use to save/update change of project
5	QuestionAnswer	A QuestionAnswer which save questions and answers of project.
6	RewardPkg	A RewardPkg which user distribution rewards by packet
7	Conversation	A Conversation which user created and joined to send and receive message together

8	Message	A Message which user send or receive to talk together
---	---------	---

**Table 4-38: Entity description****4.4.1.2 Relationship**

No	Relation	Properties	From	To	Description
1	CONTAINS	None	Category	Project	A category contains many projects
2	HAS	None	Project	Update	A update of project which was changed by creator
3	HAS	None	Project	Question Answer	A common question and answer of use project which created by creator
4	HAS	None	Project	RewardP kg	A Reward packet of project which creator device to get funded
5	COMMENT	UserID ProjectID	DDL_User	Project	A Comment which user give to project
6	BACK	UserID ProjectID	DDL_User	Project	A project which user backing project
7	REMIND	UserID ProjectID	DDL_User	Project	A Remind which user bookmark a project.
8	REPORT	UserID ProjectID	DDL_User	Project	A Report which user report a project
9	CREATE	None	DDL_User	Project	A Create which user created project
10	REPORT	ReportUserID ReportedUserID	DDL_User		A Report which User report another user
11	JOIN	None	User	Conversation	A Join which user joined or created a conversation
12	HAS	None	Conversation	Message	A Has which conversation has message(s)

13	SEND	None	DDL_User	Message	A Send which User create and send message
----	------	------	----------	---------	---

Table 4-39: Relationship description

#### 4.4.2 Table Diagram

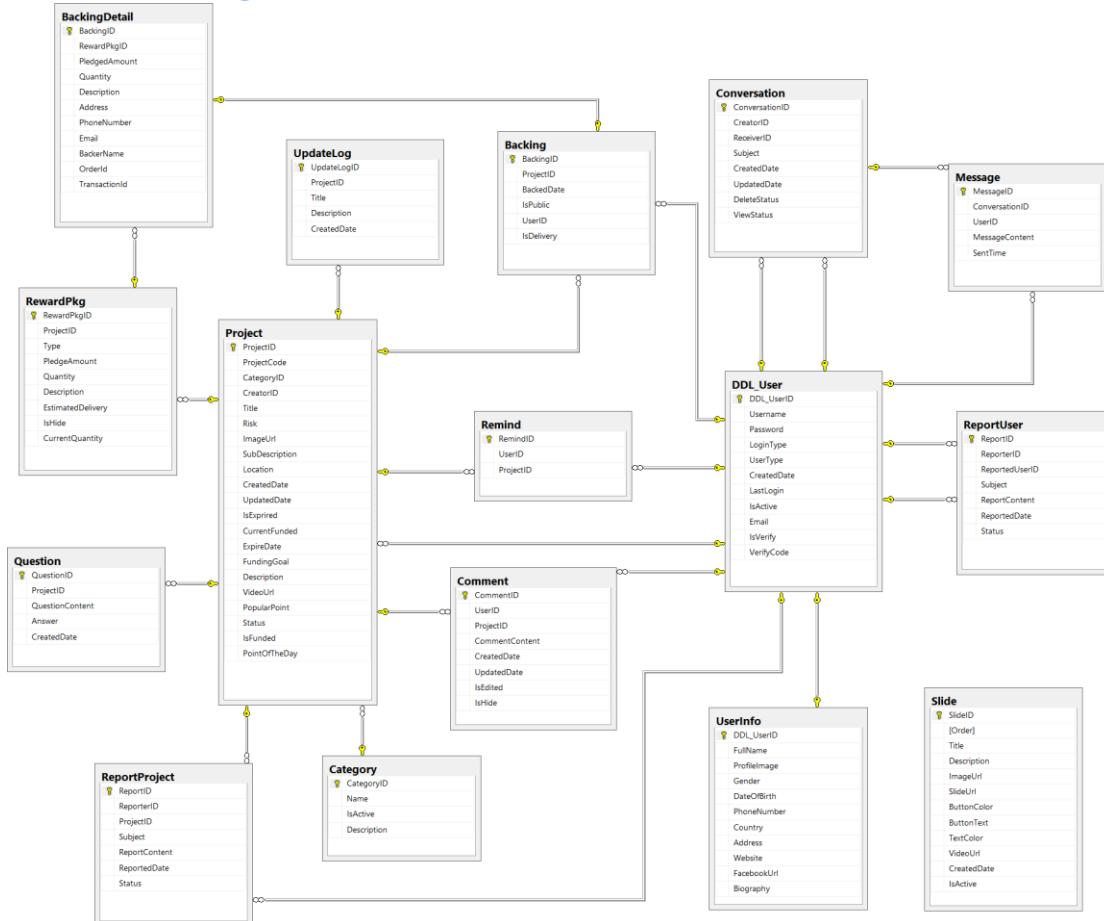


Figure 4-59: Table Diagram

##### 4.4.2.1 DDL\_User table

No	Field Name	Type	Allow Null	Description
1	UserID	int	No	User's ID
2	UserName	String	No	User's name
3	Password	String	No	User's password
4	IsActive	bool		User's active status (inactive, active)
5	CreatedDate	Datetime		User's created date
6	LoginType	String		User's created type includes: normal, Facebook or both

7	LastLogin	Datetime		User's last login
8	UserType	String		User's type (admin or nomal user)
9	IsVerify	bool		User' email verify status
10	VerifyCode	String		User's verify code
11	Email	String	No	User's email

**Table 4-40: User table****4.4.2.2 UserInfo table**

No	Field Name	Type	Allow Null	Description
1	UserID	int	No	Person's ID
2	FullName	String	No	Person's full name
3	ProfileImage	String		Person's profile image
4	Biography	String		Person's biography
5	Address	String		Person's address
6	Gender	String		Person's gender
7	PhoneNumber	String		Person's phone number
8	DateOfBirth	Datetime		Person's date of birth
9	Country	String		Person's country
10	Website	String		Person's website
11	FacebookURL	String		Person's facebook link

**Table 4-41: UserInfo table****4.4.2.3 Project table**

No	Field Name	Type	Allow Null	Description
1	ProjectID	int	No	Project's ID
2	ProjectCode	String	No	Project's code with fixed format
3	CategoryID	int	No	Project's category
4	CreatorID	int	No	Project's UserID who created project

5	Title	String		Project's title
6	Risk	String		Project's risk
7	ImageURL	String		Project's image link
8	SubDescription	String		Project's sub description
9	Location	String		Project's location created
10	IsExpired	bool		Is Project's expired
11	CurrentFunded	decimal		Project's funded status
12	IsFunded	bool		Is Project's funded
13	CreatedDate	datetime		Project's createdDate
14	ExpireDate	datetime		Project's expired date
15	FundingGoal	decimal		Project's funding goal
16	Description	String		Project's description
17	VideoUrl	String		Project's video link
18	PopularPoint	int		Project's popular point
19	Status	String		Project's status ( draft, pending, approved, suspended)
20	UpdatedDate	Datetime		Project's updatedDate
21	PointOfTheDay	int		Project's popular point of the day

**Table 4-42: Project table****4.4.2.4 Question table**

No	Field Name	Type	Allow Null	Description
1	QuestionID	Int	No	question and answer's ID
2	ProjectID	Int	No	question and answer's project id
3	Question	String		question and answer's question
4	Answer	String		question and answer's answer
5	CreatedDate	Datetime		question and answer's created date

**Table 4-43: QuestionAnswer table**

#### 4.4.2.5 Category table

No	Field Name	Type	Allow Null	Description
1	CategoryID	int	No	Category's ID
2	Name	String		Category's name
3	IsActive	bool		Category's active status
4	Description	String		Category's description

Table 4-44: Category table

#### 4.4.2.6 UpdateLog table

No	Field Name	Type	Allow Null	Description
1	UpdateLogID	int	No	Update's ID
2	ProjectID	Int	No	Update's project id
3	Title	String		Update's title
4	Description	String		Update's description
5	CreatedDate	datetime		Update's created date

Table 4-45: UpdateLog table

#### 4.4.2.7 ReportProject table

No	Field Name	Type	Allow Null	Description
1	ReportID	int	No	Report 's ID
2	ProjectID	int	No	Report 's project id
3	ReporterID	int	No	Report 's user id, who report project
4	Subject	String		Report 's subject
5	ReportContent	String		Report 's report content
6	ReportedDate	datetime		Report 's report date
7	Status	String		Report's status

Table 4-1: ReportPost table

#### 4.4.2.8 Remind table

No	Field Name	Type	Allow Null	Description
1	RemindID	int	No	Remind's ID
2	ProjectID	int	No	Remind's project id
3	UserID	int	No	Remind's user id

Table 4-46:Remind table

#### 4.4.2.9 Backing table

No	Field Name	Type	Allow Null	Description
1	BackingID	int	No	Back's ID
2	UserID	int	No	Back's user id
3	ProjectID	int	No	Back's project id
4	BackedDate	Datetime		Back's date
5	Description	String		Back's description
6	IsPublic	bool		Is Back public

Table 4-47:Backing table

#### 4.4.2.10 Comment table

No	Field Name	Type	Allow Null	Description
1	CommentID	int	No	Comment's ID
2	UserID	int	No	Comment's user id
3	ProjectID	int	No	Comment's project id
4	CommentContent	String		Comment's content
5	CreatedDate	Datetime		Comment's date
6	IsHide	bool		Is Comment's hide
7	UpdatedDate	Datetime		Comment's updated date
8	IsEdited	bool		Is Comment's edit

Table 4-48:Comment table

#### 4.4.2.11 ReportUser table

No	Field Name	Type	Allow Null	Description
1	ReportID	int	No	Report's ID
2	ReporterID	int	No	Report's report user ID
3	ReportedUserID	int	No	Report's reported user ID
4	Status	String		Report's status
5	Subject	String		Report's subject
6	ReportContent	String		Report's content
7	ReportedDate	datetime		Report's date

Table 4-49:ReportUser table

#### 4.4.2.12 Slide table

No	Field Name	Type	Allow Null	Description
1	SlideID	int	No	Slide's ID
2	Title	String		Slide's title
3	Order	int		Slide's order
4	Description	String		Slide's description
5	ImageUrl	String		Slide's image link
6	IsActive	bool		Slide's is Active status
7	SliderUrl	String		Slide's slide URL
8	ButtonColor	String		Slide's button color
9	CreatedDate	datetime		Slide's created date
10	ButtonText	String		Slide's button text
11	TextColor	String		Slide's text color
12	VideoUrl	String		Slide's url video

Table 4-50: Slide table

#### 4.4.2.13 Conversation table

No	Field Name	Type	Allow Null	Description
1	ConversationID	int	No	Conversation's ID
2	CreatorID	int	No	Conversation's creator id
3	ReceiverID	int	No	Conversation's receiver
4	Subject	String		Conversation's subject
5	CreatedDate	Datetime		Conversation's created date
6	UpdatedDate	Datetime		Conversation's update date
7	DeleteStatus	String		Conversation's user delete status
8	ViewStatus	String		Conversation's user view status

**Table 4-51: Conversation table**

#### 4.4.2.14 Message table

No	Field Name	Type	Allow Null	Description
1	MessageID	int	No	Message's ID
2	ConversationID	int	No	Message's conversation id
3	UserID	int	No	Message's user id
4	MessageContent	String		Message's content
5	SentTime	Datetime		Message's sent time

**Table 4-52:Message table**

#### 4.4.2.15 RewardPkg table

No	Field Name	Type	Allow Null	Description
1	RewardID	int	No	Reward's ID
2	ProjectID	int	No	Reward's project id
3	Quantity	int		Reward's limited quantity
4	Type			Reward's type
5	Description	String		Reward's description

6	EstimatedDelivery	datetime		Reward's estimated delivery date
7	IsHide	bool		Is Reward hide
8	CurrentQuantity	int		Reward's current quantity
9	PledgeAmount	decimal		Reward's PledgeAmount

**Table 4-53: RewardPkg table****4.4.2.16 BackingDetail table**

No	Field Name	Type	Allow Null	Description
1	BackingID	int	No	Backing ID
2	RewardPkgID	int	No	RewardPkgID
3	PledgedAmount	Int	No	PledgedAmount
4	Quantity	String	No	Quantity
5	Description	String	No	Description
6	Address	Datetime	No	Backer's Address
7	PhoneNumber	Bool	No	Backer's Phone number
8	Email	String	No	Backer email
9	BackerName	String	No	Backer's name
10	OrderId	string	No	Identify string of backing order
11	TransactionId	string	No	Identify string of Baokim transaction

**Table 4-54: BackingDetail table**

## 4.5 Screen Design

### 4.5.1 User Client

#### 4.5.1.1 Screen flow

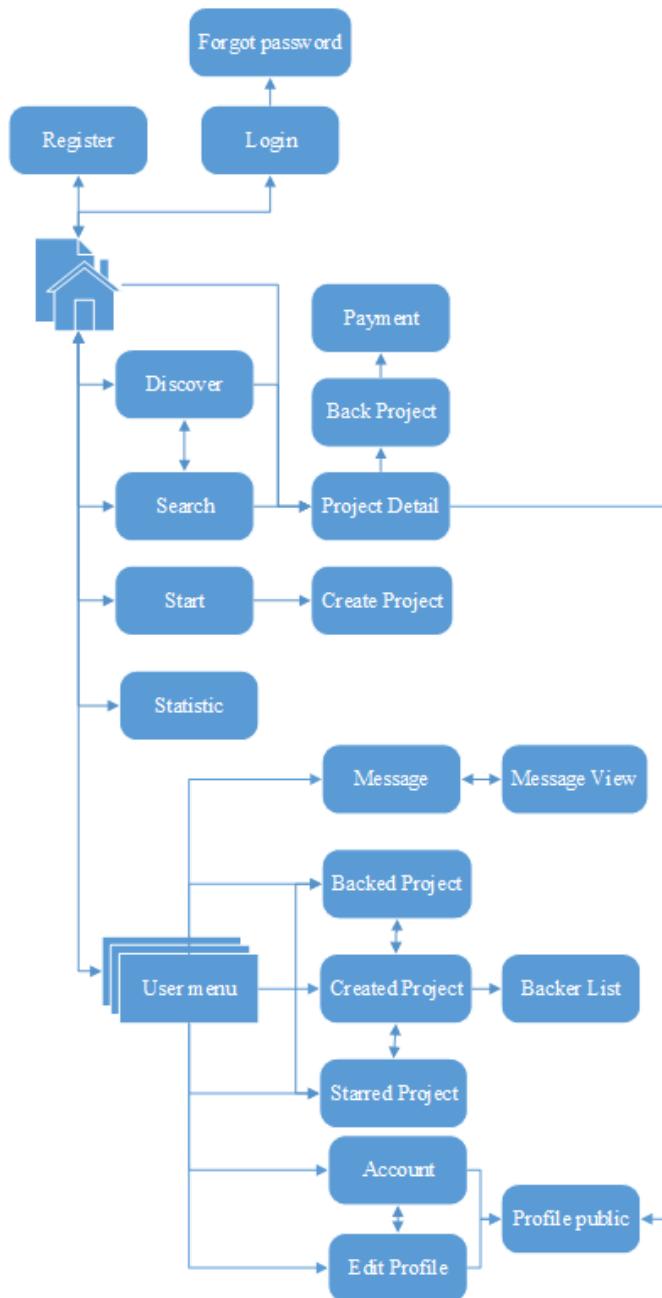


Figure 4-60: Client screen flow

#### 4.5.1.2 Register screen

**Figure 4-61: Register screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Register	Đăng ký	Label		Y		Display register form
2	Login with Facebook	Đăng nhập với facebook	Button	Click	Y		Login with facebook
3	Username	Tên tài khoản	Text box	Type	Y	8-20	Username of user
4	Password	Mật khẩu	Text box	Type	Y	8-50	Password of user
5	Confirm Password	Nhập lại mật khẩu	Text box	Type	Y		User retype password
6	Email	Email	Textbox	Type	Y		Email of user
7	Full name	Tên đầy đủ	Textbox	Type	Y		Full name of user
8	Sign up	Đăng ký	Button	Click	Y		User register a new account
9	Login now	Đăng nhập ngay	Hyperlink	Click	Y		Login when user have an account

**Table 4-55: Register screen**

#### 4.5.1.3 Login screen

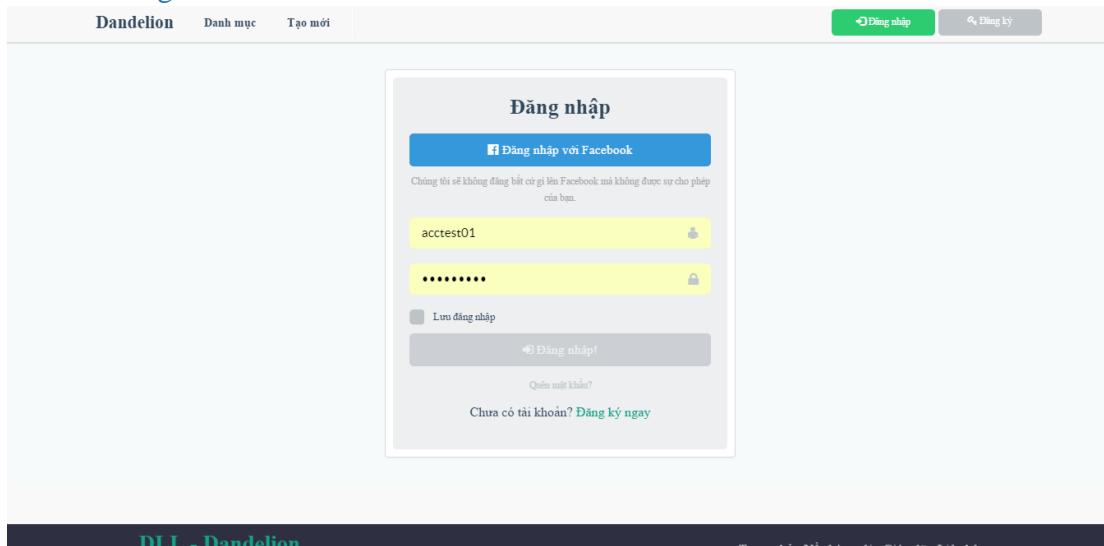
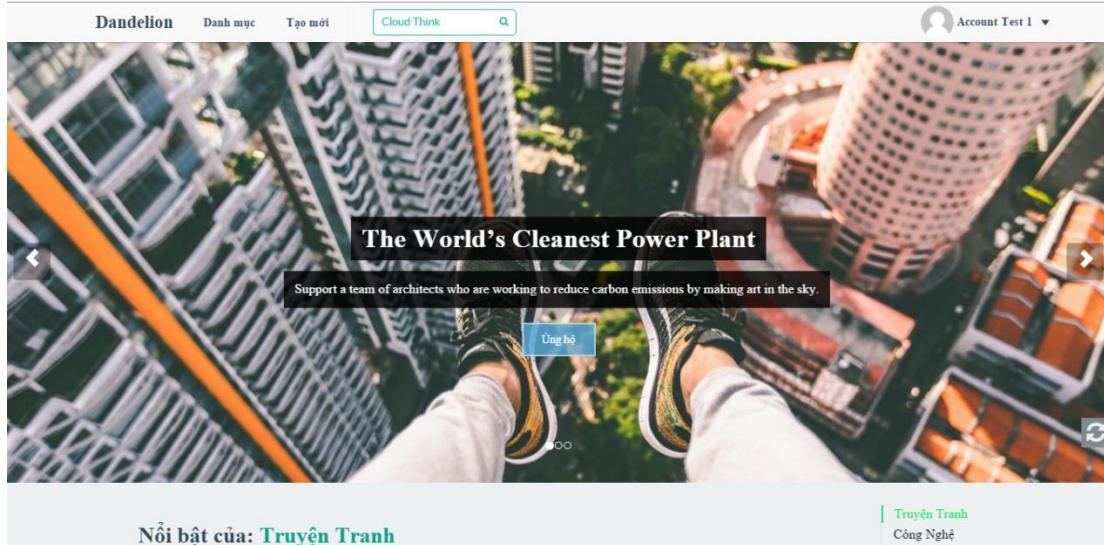


Figure 4-62: Login screen

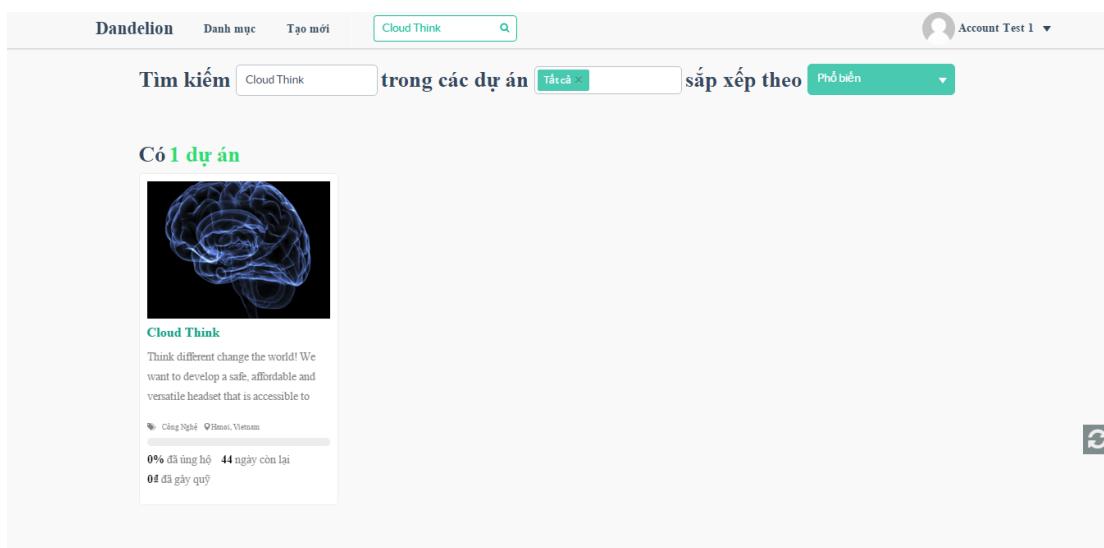
No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Login with Facebook	Đăng nhập qua Facebook	Button	Click	Y		Login with Facebook
2	Email	Email	Text box	Type	Y		Email of registered user
3	Password	Mật khẩu	Text box	Type	Y		Password of registered user
4	Remember Me	Lưu đăng nhập	Check box	Click	Y		Save login information
5	Login	Đăng nhập	Button	Click	Y		Login
6	Forgot Password	Quên mật khẩu	Hyperlink	Click	Y		Resend password to user
7	Register now	Đăng ký ngay	Hyperlink	Click	Y		Register when user have not an account

Table 4-56: Login screen

#### 4.5.1.4 Search screen



**Figure 4-63:** Search project screen design



**Figure 4-64:** Search project result screen design

No	Field name	Field name in Vietnamese	Type	Require	Man-datory	Max-Length	Description
1	Search box	Ô tìm kiếm	Text		Y		Display text
2	Search icon		Button	Click	Y		Search project

2	Search result box	Tìm kiếm	Text		Y		Display search text
3	Project category	Danh mục dự án	Dropdowlist	Click	Y		Filter by category of projects
4	Status project	Trạng thái dự án	Dropdow list	Click	Y		Filter by status of projects
5	Number project	Số dự án	Label		Y		Display number of projects as result search
6	List Project	Danh sách dự án	List				Display list of project as result

Table 4-57: Search screen

#### 4.5.1.5 Discover screen

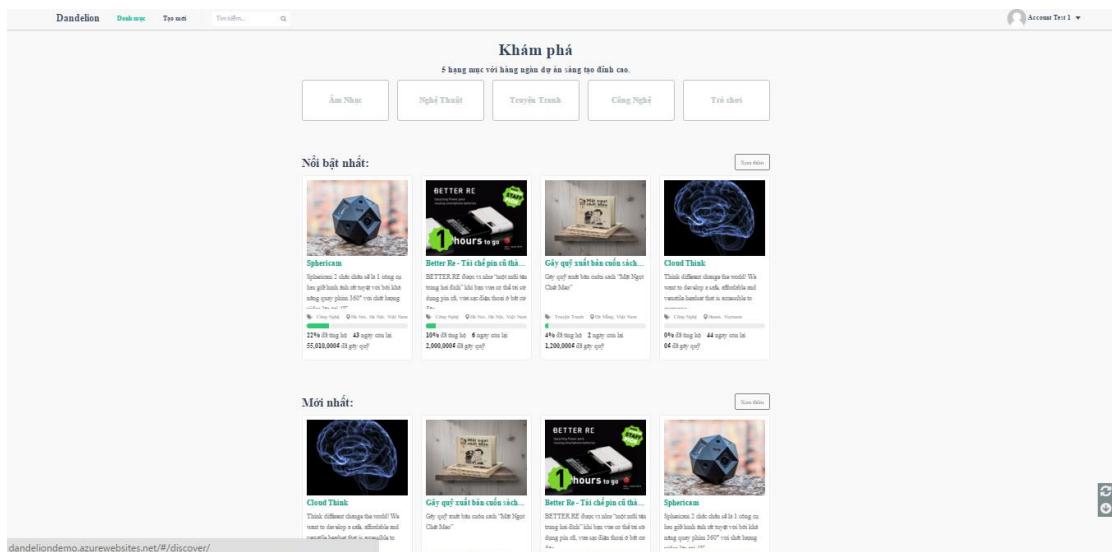


Figure 4-65: Discover screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List of Category	Danh sách dự án	List		Y		Display list of categories
2	What popular	Dự án nổi bật	Label		Y		Label display "Nổi bật nhất"
3	List popular project	Danh sách các dự án nổi bật	List		Y		Display list of popular project

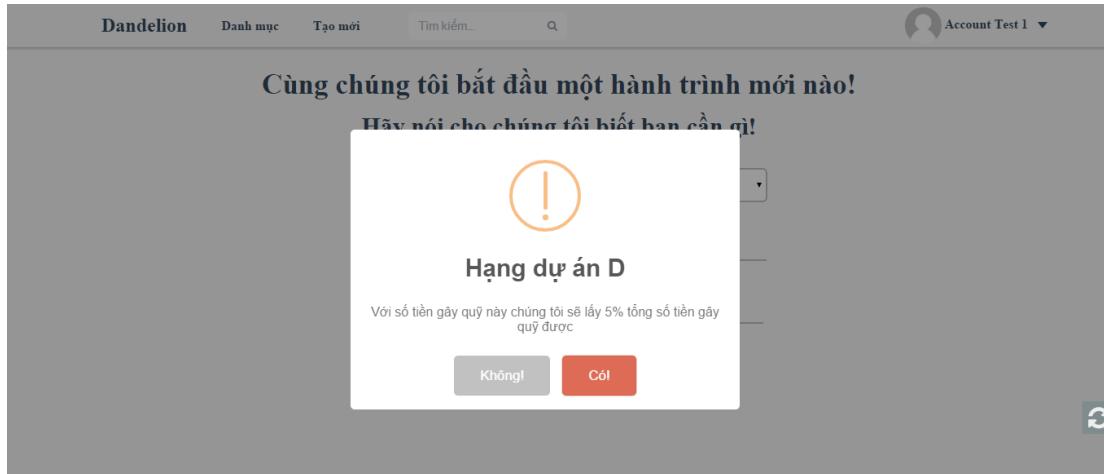
4	What newest	Dự án mới	Label 1		Y		Label display “Mới nhất”
5	List newest project	Danh sách các dự án mới nhất	List		Y		Display list of newest project
6	What most fund	Dự án ủng hộ nhiều nhất	Label 1		Y		Label display “Được ủng hộ nhiều nhất”
7	List most fund project	Danh sách các dự án được ủng hộ nhiều nhất	List		Y		Display list of most fund project
8	What end date	Dự án sắp hết hạn ủng hộ	Label 1		Y		Label display “Sắp hết hạn ủng hộ”
9	List end date project	Danh sách các dự án sắp hết hạn ủng hộ	List		Y		Display list of end date project

Table 4-58: Discover screen

#### 4.5.1.6 Create project screen

The screenshot shows a web application interface for creating a new project. At the top, there is a navigation bar with links for 'Dandelion', 'Danh mục', 'Tạo mới' (highlighted in green), 'Tim kiếm...', and a user account icon labeled 'Account Test 1'. Below the navigation bar, the main content area has a heading 'Cùng chúng tôi bắt đầu một hành trình mới nào!' and a sub-instruction 'Hãy nói cho chúng tôi biết bạn cần gì!'. There are two input fields: one for 'Tôi muốn bắt đầu với dự án về...' containing 'Âm Nhạc' with a dropdown arrow, and another for 'với số tiền gây quỹ là ...' followed by a blank input field. Below these fields is another input field for 'cho dự án ...' also followed by a blank input field. A large grey button labeled 'Bắt đầu' is positioned below the first input field. In the bottom right corner of the main area, there is a circular icon with a double-headed arrow. The footer of the page includes the text 'DLL - Dandelion' and a link 'dandeliondemo.azurewebsites.net/#/project/create/'. On the far right of the footer, there are links for 'Trang chủ', 'Về chúng tôi', 'Giúp đỡ', and 'Liên hệ'.

Figure 4-66: Create project screen



**Figure 4-67:** Popup create project screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Project category	Danh mục dự án	Drop down List	Click	Y		Choose a category of project
2	Project name	Tên dự án	Text		Y		Name of project
3	Project pledge	Số tiền gây quỹ	Number		Y		Pledge of project
4	Start	Bắt đầu	Button	Click	Y		Click to create a project
5	Confirm popup	Xác nhận thông tin	Pop up		Y		Display confirm popup
6	No	Hủy bỏ	Button	Button	Y		Not accept with rules
7	Yes	Có	Button	Button	Y		Accept with rules and Create a project

**Table 4-59:** Create project screen

#### 4.5.1.7 Edit project screen

##### 4.5.1.7.1 Basic tab screen

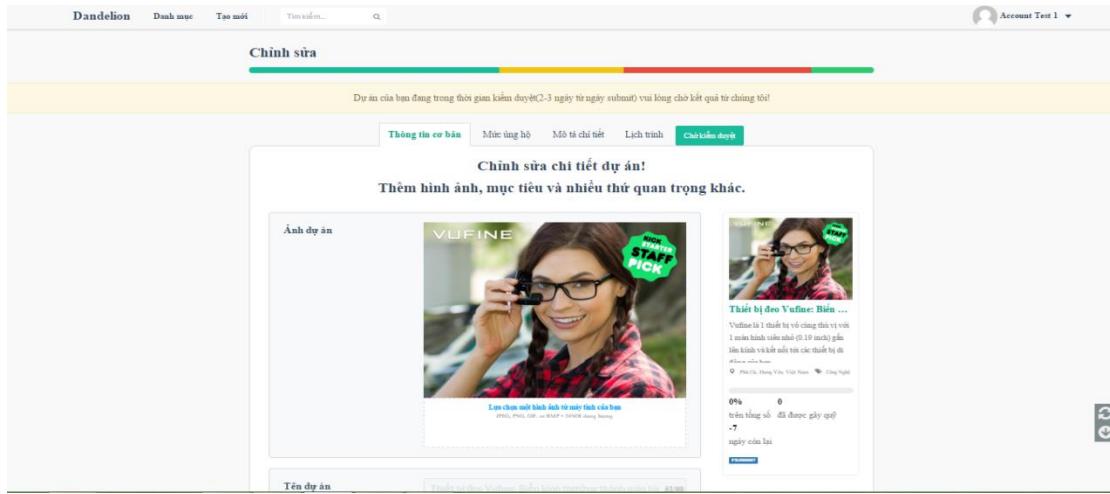


Figure 4-68: Basic tab screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Edit project	Chỉnh sửa dự án	Label		Y		Display label as “Chinh sửa”
2	Basic tab	Tab thông tin cơ bản	Tab	Click	Y		Display Basic edit form
3	Project image	Ảnh dự án	Image	Click	Y		Display project image
4	Project title	Tiêu đề dự án	Text		Y	10-60	Display title of project
5	Short blurb	Mô tả ngắn	Textarea		Y	30-135	Display short blurb of project
6	Project category	Lĩnh vực	Dropdownlist	Click	Y		Display category of project
7	Project location	Địa điểm	Date		Y		Display location of project
8	Funding duration	Thời hạn	Date		Y		Display duration of project
9	Funding goal	Mục tiêu gây quỹ	Text		Y		Display pledge of project
10	Save	Lưu	Button		Y		Save information of project
11	Reset	Khôi phục	Button		Y		Reset information of project

12	Submit for review	Gửi dự án	Button		Y		Submit project for admin
----	-------------------	-----------	--------	--	---	--	--------------------------

**Table 4-60:Basic tab****4.5.1.7.2 Reward tab screen**

The screenshot shows a web application interface for creating a reward. At the top, there's a navigation bar with 'Dandelion', 'Danh mục', 'Tạo mới', 'Tim kiếm...', and a user account icon. Below the navigation is a yellow header bar with the text 'Chinh sáu' and a message about a project currently in the review period (2-3 days). The main content area has tabs: 'Thông tin cơ bản', 'Mức ứng hộ' (which is selected), 'Mô tả chi tiết', 'Lịch trình', and 'Chỉ kiểm duyệt'. A large central box contains questions like 'Bạn nên cảm ơn những người đã ủng hộ bạn như thế nào?' and 'Đưa ra những gói quà tặng, ưu đãi... thì sao?'. It also includes a reward plan table with one row labeled 'Gói #1' and details like 'Ứng hộ 1,000đ hoặc hơn', '0 Đã ứng hộ', and a note about receiving a thank you email. At the bottom right of the central box are three small buttons: 'Xem', 'Sửa', and 'Xóa'. The footer shows the URL 'dandeliondemo.azurewebsites.net/#reward' and links to 'Trang chủ', 'Về chúng tôi', 'Giúp đỡ', 'Liên hệ'.

**Figure 4-69: Reward tab screen**

This screenshot shows the same reward creation form as Figure 4-69, but with validation errors. The 'Mức ứng hộ' field has an error message: 'Mức ứng hộ ít nhất là 10,000đ.' The 'Thời gian giao hàng' field has an error message: 'Ngày giao hàng cần đặt sau ngày hết hạn dự án.' The 'Giới hạn' dropdown has an error message: 'Số lượng phải lớn hơn 0.' The 'Tạo' button at the bottom right is highlighted in green, while the 'Bỏ' button is red.

**Figure 4-70: Reward tab screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Reward tab	Tab mức ứng hộ	Tab	Click	Y		Display Reward edit form

2	List reward	List reward	List				List of reward
3	Add new reward	Tạo gói mới	Button	Click	Y		Add new reward
4	Hidden reward	Ẩn gói	Button	Click			Hidden a reward
5	Delete reward	Xóa gói	Button	Click			Delete a reward
6	Edit reward	Sửa gói	Button	Click			Edit a reward
7	Popup edit,create reward	Popup Tạo và sửa gói	Popup				Popup create and edit reward
8	Reward pledge	Mức ủng hộ	Number		Y		Reward pledge
9	Reward description	Mô tả	Textarea		Y		Description of reward
10	Time ship	Thời gian giao hàng	Date		Y		Time ship of reward
11	Limit	Giới hạn					
12	Reward Type	Loại	Dropd ownlist		Y		Type of reward: No reward, Limit, No limit
13	Create	Tạo	Button		Y		Create a reward
14	Cancel	Hủy (Bỏ)	Button		Y		Cancel the Create a reward
15	Save	Lưu	Button		Y		Save information of project
16	Reset	Khôi phục	Button		Y		Reset information of project

**Table 4-61:Reward tab**

#### 4.5.1.7.3 Story tab screen

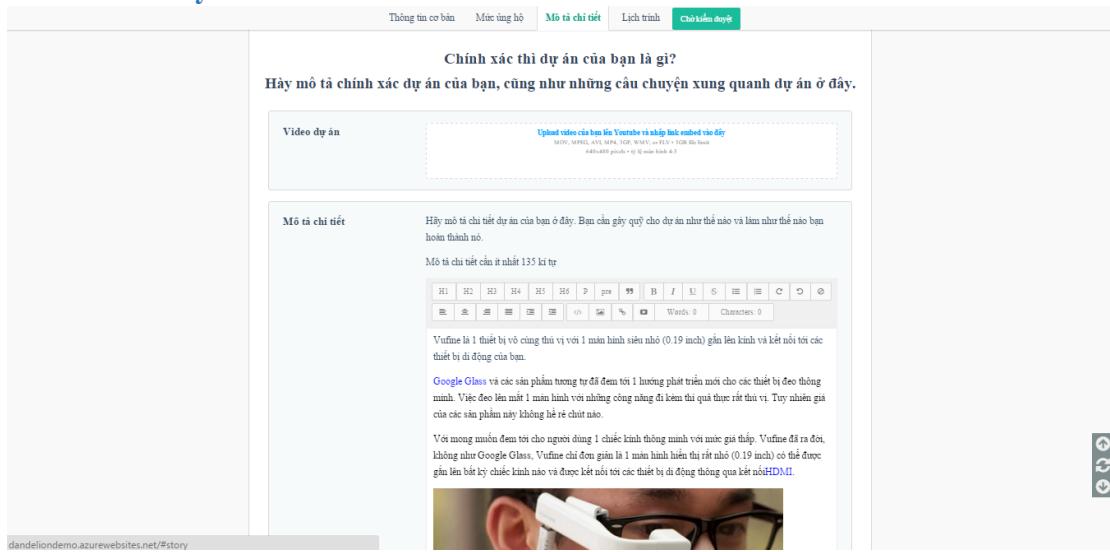


Figure 4-71: Story tab screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Story tab	Tab mô tả chi tiết	Tab	Click	Y		Display Story edit form
3	Project video	Video/Clip dự án	Video	Click			Display project video
4	Project description	Mô tả chi tiết dự án	Textarea		Y		Display description of project
5	Risk and challenges	Khó khăn và thử thách	Textarea		Y		Display risk and challenges of project
6	Save	Lưu	Button		Y		Save information of project
7	Reset	Khôi phục	Button		Y		Reset information of project

Table 4-62: Story tab

#### 4.5.1.7.4 Update tab screen

The screenshot shows a user interface titled 'Chỉnh sửa' (Edit) at the top. Below it is a navigation bar with tabs: Thông tin cơ bản, Mục ứng hò, Mô tả chi tiết, Cập nhật (highlighted in green), Hồi đáp, Lịch trình, and Đã xác nhận. A message in Vietnamese asks how to thank supporters. Below the message, there are three update entries:

- Cập nhật #1: Tên (Name) and Nội dung (Content).
- Cập nhật #2: Tên (Name) and Nội dung (Content).
- Cập nhật #3: Tên (Name) and Nội dung (Content). The content field contains two lines of placeholder text: 'tuyetthien' and 'tuyetthien/tuyetthien/Hoang'.

A green button labeled 'Tạo cập nhật mới' (Create new update) is located at the bottom right of the update list area.

Figure 4-72: Update tab screen

The screenshot shows a modal dialog box titled 'Tạo cập nhật' (Create update) overlaid on the main 'Update tab screen'. The dialog has fields for 'Tên:' (Name) and 'Mô tả:' (Description). At the bottom are 'Bó' (Save) and 'Làm' (Cancel) buttons, and a green 'Tạo cập nhật mới' (Create new update) button at the bottom right of the dialog.

Figure 4-73: New update dialog

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Update tab	Tab cập nhật	Tab	Click	Y		Display Update edit form
2	List updates	Danh sách cập nhật	List				List of update
3	Add new update	Tạo cập nhật mới	Button	Click	Y		Add new update
4	Save update	Lưu cập nhật	Button	Click	Y		Save a update

5	Delete update	Xóa cập nhật	Button	Click			Delete a update
6	Edit update	Sửa cập nhật	Button	Click	Y		Edit a update
7	Popup create update	Popup Tạo cập nhật	Popup				Popup create update
8	Title of update	Tiêu đề	Text		Y		Title of update
10	Date of update	Ngày cập nhật	Label		Y		Date of create update
12	Content of update	Nội dung	Textarea		Y		Content of update
13	Create	Tạo	Button		Y		Create a update
14	Cancel	Hủy (Bỏ)	Button		Y		Cancel the Create a update
15	Save	Lưu	Button		Y		Save information of project
16	Reset	Khôi phục	Button		Y		Reset information of project

Table 4-63: Update tab

#### 4.5.1.7.5 Question and Answer tab screen

The screenshot shows a user interface for managing questions and answers. At the top, there is a title in Vietnamese: "Đặt những câu hỏi và trả lời" (Ask questions and answers) and "Giải đáp những thắc mắc và biết thêm nhiều thông tin" (Answer questions and learn more information). Below the title, there is a section labeled "#1 Câu hỏi" (Question #1) containing a text input field with placeholder text "Đây là câu hỏi". Below it, there is a section labeled "Trả lời" (Answer) containing a text input field with placeholder text "Đây là câu trả lời". A red "Xóa" (Delete) button is located at the bottom of this section. To the right of the main content area, there are three small circular icons with arrows pointing up, down, and left-right. At the bottom center, there is a green "Thêm hỏi đáp" (Add question and answer) button.

Figure 4-74: Q&amp;A tab screen

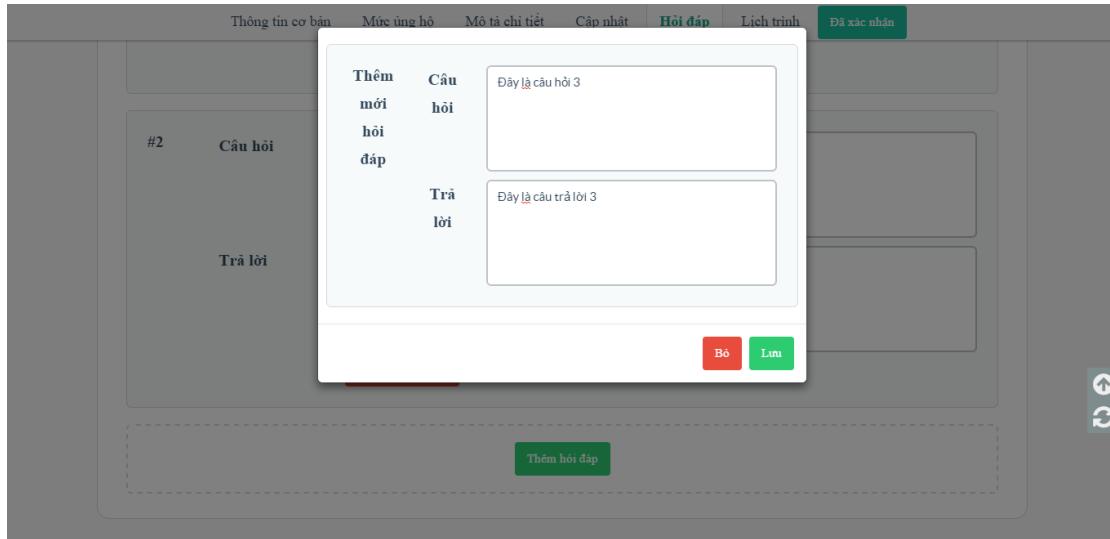


Figure 4-75: New Q&amp;A dialog

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Q&A tab	Tab hỏi đáp	Tab	Click	Y		Display Q&A edit form
2	List Q&A	Danh sách hỏi đáp	List				List of Q&A
3	Add new Q&A	Tạo cập hỏi đáp	Button	Click	Y		Add new Q&A
4	Save Q&A	Lưu hỏi đáp	Button	Click	Y		Save a Q&A
5	Delete Q&A	Xóa hỏi đáp	Button	Click			Delete a Q&A
6	Edit Q&A	Sửa hỏi đáp	Button	Click	Y		Edit a Q&A
7	Popup create Q&A	Popup Tạo hỏi đáp	Popup				Popup create Q&A
8	Question	Câu hỏi	Textarea		Y		Question
12	Answer	Trả lời	Textarea		Y		Answer for question
13	Create	Tạo	Button		Y		Create a Q&A
14	Cancel	Hủy (Bỏ)	Button		Y		Cancel the Create a Q&A

15	Save	Lưu	Button		Y		Save information of project
16	Reset	Khôi phục	Button		Y		Reset information of project

Table 4-64: Q&amp;A tab

#### 4.5.1.8 Statistic screen

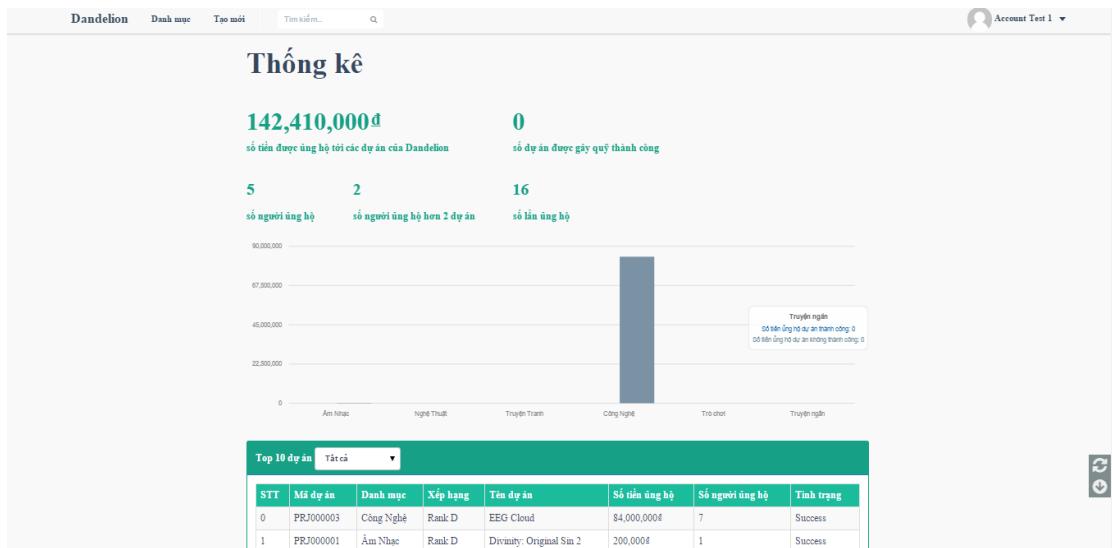


Figure 4-76: Statistic screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Statistic	Thống kê	Label		Y		Label of page
2	Total pledge	Tổng số tiền được ủng hộ	Label		Y		Display total pledge of projects in Dandelion
3	Total project	Tổng số dự án được gây quỹ thành công	Label		Y		Display total project success in Dandelion
4	Total backer	Tổng số người ủng hộ	Label		Y		Display total backer in Dandelion
5	Total backer more 2 times	Tổng số người ủng hộ 2 lần	Label		Y		Display total backer more 2 times in Dandelion

6	Total time backing	Tổng số lần ủng hộ	Label		Y		Display total time backing in Dandelion
7	Top 10 project	Top 10 dự án	Hyperlink	Click	Y		Top 10 project
8	Table top 10 project	Bảng Top 10 dự án	Table		Y		Table top 10 project
9	Top 10 backer	Top 10 người ủng hộ	Hyperlink	Click	Y		Top 10 backer
10	Table top 10 backer	Bảng Top 10 người ủng hộ	Table		Y		Table top 10 backer
11	Top 10 creator	Top 10 người gây quỹ	Hyperlink	Click	Y		Top 10 creator
12	Table top 10 creator	Bảng Top 10 người gây quỹ	Table		Y		Table top 10 creator

**Table 4-65: Statistic**

#### 4.5.1.9 Project detail screen

The screenshot shows the Dandelion platform's project detail screen for a campaign titled "Sphericam" by Vu Cong Chinh. The main visual is a video thumbnail showing a person inside a white spherical camera device. Key statistics displayed are: 6 people have backed the project, raising a total of 59,010,000đ, which is 23.55% of the 250,000,000đ goal. There are 41 days left. A summary message indicates the project will end on December 30, 2015. The creator's profile picture and basic information are also visible.

**Figure 4-77: Project detail screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Name of Project	Tên dự án	Hyperlink	Click	Y		Name of project

2	Name of creator	Tên người gây quỹ	Hyperlink	Click	Y		Display information of creator
3	Total backer	Số người ủng hộ	Label		Y		Total backer
4	Total back	Tổng số tiền được ủng hộ	Label		Y		Total back
5	Total pledge	Tổng số tiền cần gây quỹ	Label		Y		Total pledge
6	Total remain days	Tổng số ngày còn lại	Label		Y		Total remain days
7	Category of project	Danh mục của dự án	Hyperlink	Click	Y		Category of project
8	Location of project	Địa điểm của dự án	Label		Y		Location of project
9	Share this project	Chia sẻ dự án này	Button	Click	Y		Share project with facebook
10	Tab info	Tab thông tin	Tab	Click	Y		Display information of project
11	Tab update	Tab cập nhật	Tab	Click	Y		Display update of project
12	Tab comment	Tab bình luận	Tab	Click	Y		Display comment of project
13	Tab schedule	Tab tiến trình	Tab	Click	Y		Display schedule of project

**Table 4-66: Project detail**

#### 4.5.1.9.1 Information tab screen

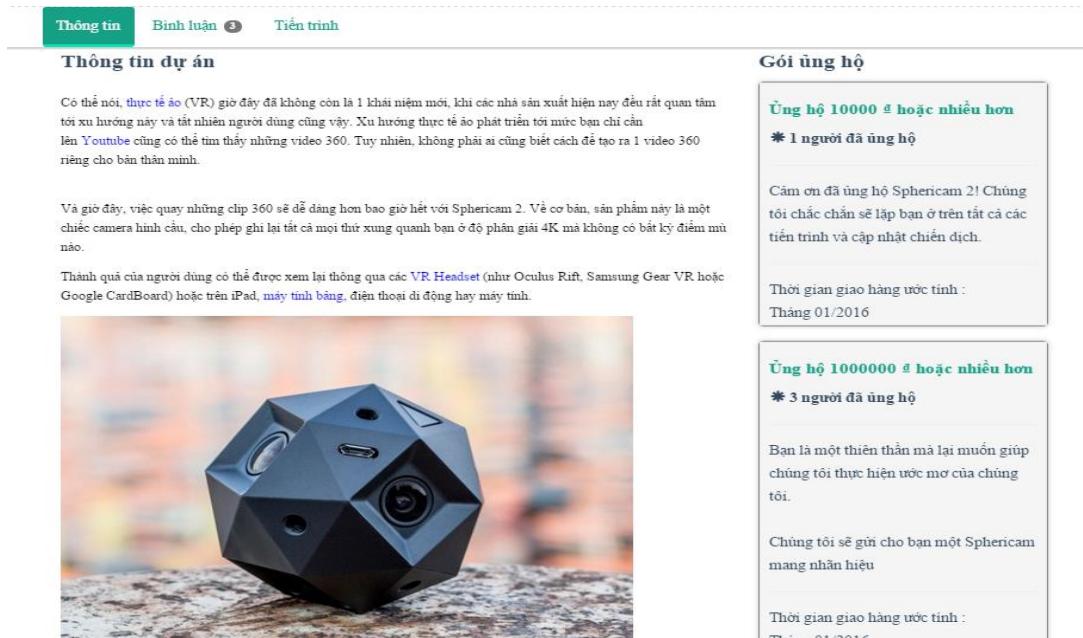


Figure 4-78: Information tab screen

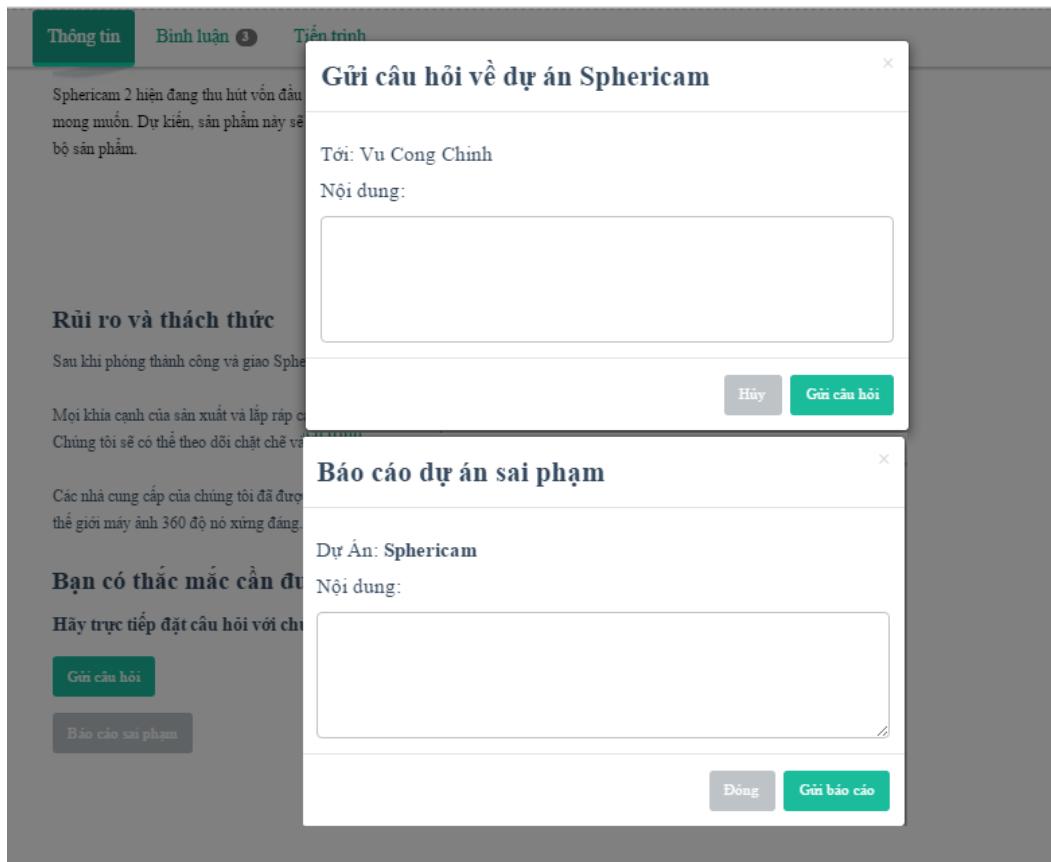


Figure 4-79: Sent report, question dialog

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List reward	Danh sách gói quà	Button	Click	Y		Display list reward
2	Sent a question	Gửi câu hỏi	Button	Click	Y		Sent a question to creator's project
3	Popup Sent a question	Popup Gửi câu hỏi	Popup				Display popup sent a question when click sent a question
4	Report this project	Báo cáo sai phạm	Button	Click	Y		Report project
5	Popup Report this project	Popup Báo cáo dự án sai phạm	Popup				Display popup report project when click Report this project

Table 4-67: Sent report, question

#### 4.5.1.9.2 Update tab screen



Figure 4-80: Update tab screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List updates	Danh sách các cập nhật	List		Y		Display list updates

Table 4-68: Update tab

#### 4.5.1.9.3 Comment tab screen

The screenshot shows a web-based application interface for managing comments. At the top, there are three tabs: 'Thông tin' (Information), 'Bình luận' (Comments) which is highlighted in green, and 'Tiến trình' (Progress). Below the tabs, a header bar says 'Bình luận về dự án này'. A user profile icon is visible. The main area contains a list of comments from different users:

- Lưu Ngọc Mạnh** - lúc 00:58 ngày 19/11/2015: bình luận cho lên top nào..  
up  
up up
- Vũ Ngọc Trung** - lúc 16:34 ngày 16/11/2015 - đã chỉnh sửa: quả là chán =="  
stressed
- Lưu Ngọc Mạnh** - lúc 00:30 ngày 30/10/2015: dự án chán vãi lúa

On the right side of the comment list, there is a small circular icon with a plus sign and a checkmark.

Figure 4-81: Comment tab screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List comments	Danh sách các bình luận	List		Y		Display list comment
2	Input comment	Nhập bình luận...	Textarea		Y	10-500	Display content of comment
3	Comment	Bình luận	Button	Click	Y		Sent a comment to project
4	Hide comment	Ẩn	Hyperlink	Click	Y		Comment of themselves is hided
5	Edit comment	Sửa	Hyperlink	Click	Y		Comment of themselves is edited
6	Delete comment	Xóa	Hyperlink	Click	Y		Comment of themselves is Deleted

Table 4-69: Comment tab

#### 4.5.1.10 Back screen

##### 4.5.1.10.1 Back screen

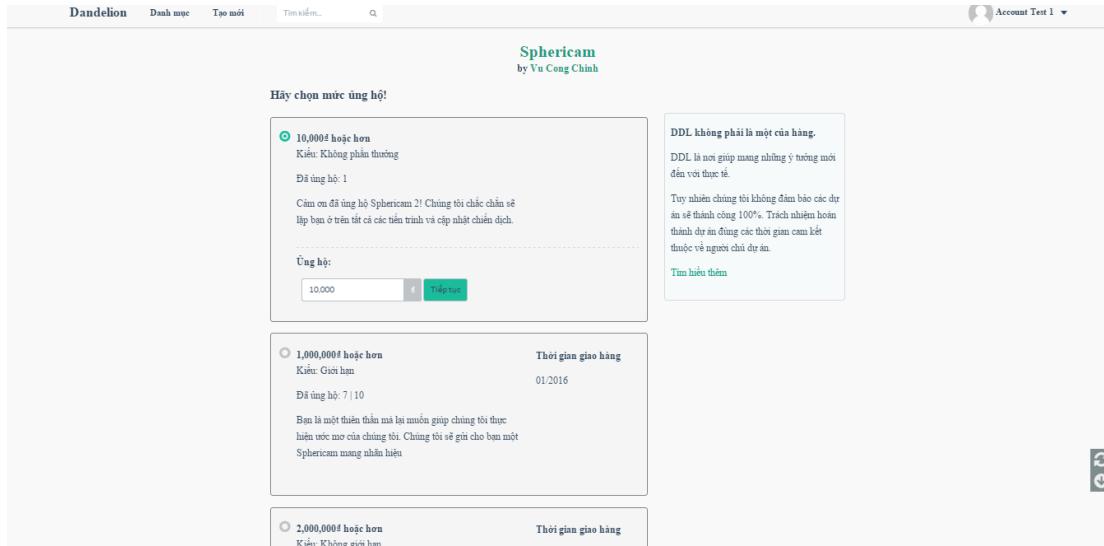


Figure 4-82: Back screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Name of Project	Tên dự án	Hyper link	Click	Y		Name of project
2	Name of creator	Tên người gây quỹ	Hyper link	Click	Y		Display information of creator
3	List reward	Danh sách gói quà	Radio	Click	Y		Display list reward
4	Pledge	Số tiền ủng hộ	Number		Y		Pledge for project
5	Quantity	Số lượng	Number		Y		Quantity reward
6	Continue	Tiếp tục	Button	Click	Y		Click to link to payment page

Table 4-70: Back project

#### 4.5.1.10.2 Payment screen

Figure 4-83: Payment screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Avatar	Ảnh đại diện	Image				Display avartar
2	Full name	Tên đầy đủ	Text		Y		Full name of backer
3	Email	Email	Text		Y		Email of backer
4	Address	Địa chỉ	Text		Y		Address of backer
5	Phone number	Số điện thoại	Text		Y		Phone number of backer
6	More info	Thông tin thêm	Textarea				More information about reward,...
7	Back	Üng hộ	Button		Y		Back project

Table 4-71: Payment

#### 4.5.1.11 Message screen

**Figure 4-84:** Message screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	New message	Soạn tin nhắn	Button	Click	Y		Display new message form
2	All message	Tất cả thư	Button	Click	Y		Filter table as all message
3	Inbox	Hộp thư đến	Button	Click	Y		Filter table as inbox message
4	Sent	Hộp thư đi	Button	Click	Y		Filter table as sent message
5	Toggle	Toggle icon	Button	Click	Y		Minimize/Maximun options of message
6	Delete message	Xóa tin nhắn	Button	Click	Y		Delete message selected
7	Table message	Danh sách tin nhắn	Table		Y		Table contain messages

**Table 4-72:** Message

#### 4.5.1.11.1 Send message screen

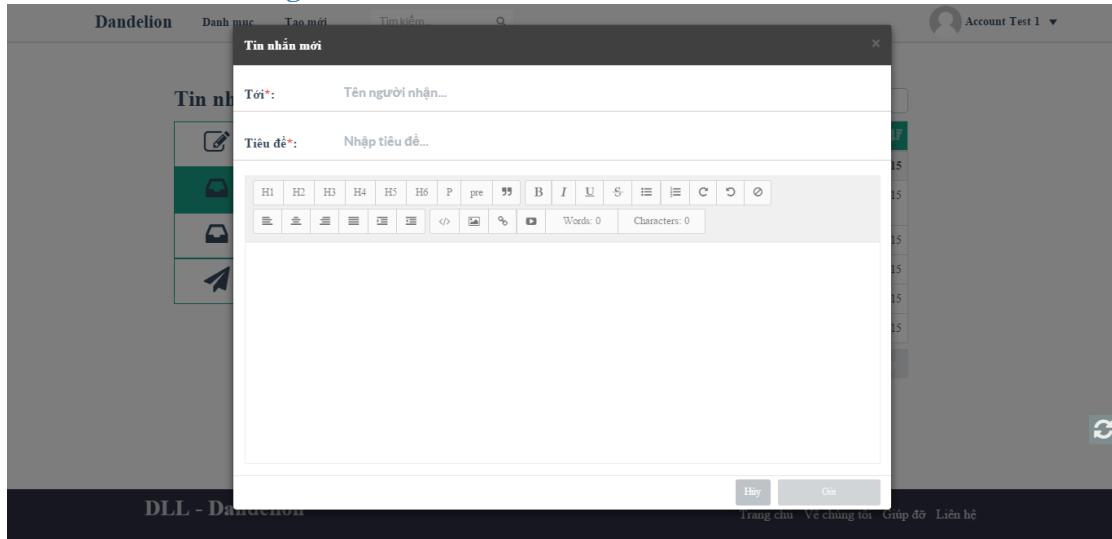


Figure 4-85: Send message screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Send to	Tên người nhận	Text		Y		Display suggest name want to message
2	Title	Tiêu đề	Text		Y		Title of new message
3	Content	Nội dung	Text area		Y		Content of message
4	Cancel	Hủy	Button		Y		Cancel the new message
5	Send	Gửi	Button		Y		Sent message

Table 4-73: Sent message

#### 4.5.1.11.2 Delete message screen

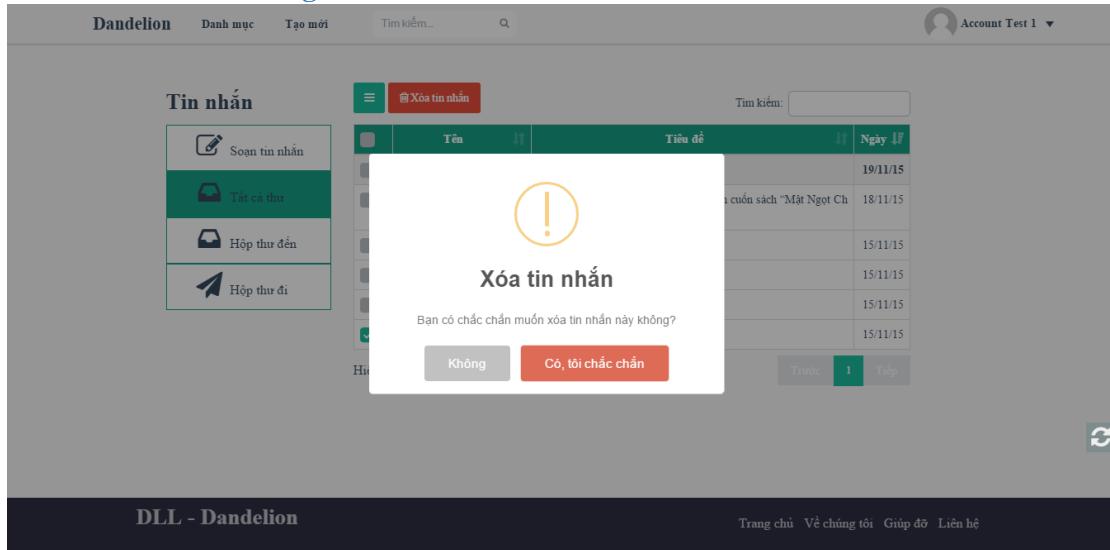
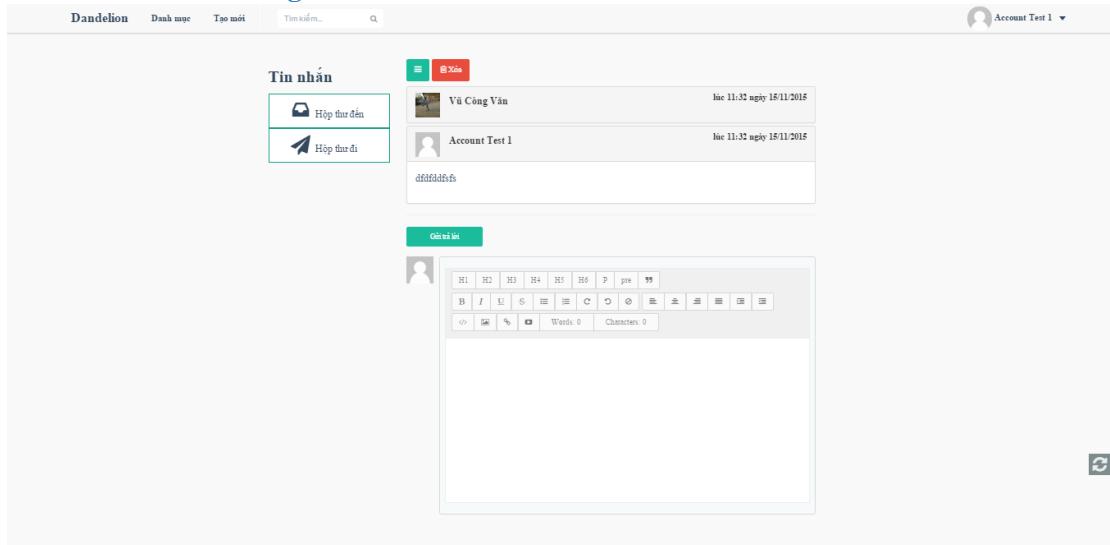


Figure 4-86: Delete message screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Confirm Label	Xóa tin nhắn Bạn có muốn xóa tin nhắn này không	Label	Label	Y		Label display “Xóa tin nhắn...”
2	No	Không	Button	Button	Y		Cancel delete message
3	Yes	Có, tôi chắc chắn	Button	Button	Y		Delete mesage

Table 4-74: Delete message

#### 4.5.1.11.3 View message screen



**Figure 4-87:** View Message screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Conversation	Hội thoại	List		Y		List of message in conversation
2	Header	Tiêu đề	Hyper link	Click	Y		Toggle hide/show message
3	Sent	Gửi	Button	Click	Y		Sent a message
4	Content	Nội dung	Textarea		Y		Content of message

**Table 4-75:** Message view

#### 4.5.1.12 Project management screen

##### 4.5.1.12.1 Created project screen

Dự án đã tạo:

Dự án nháp:

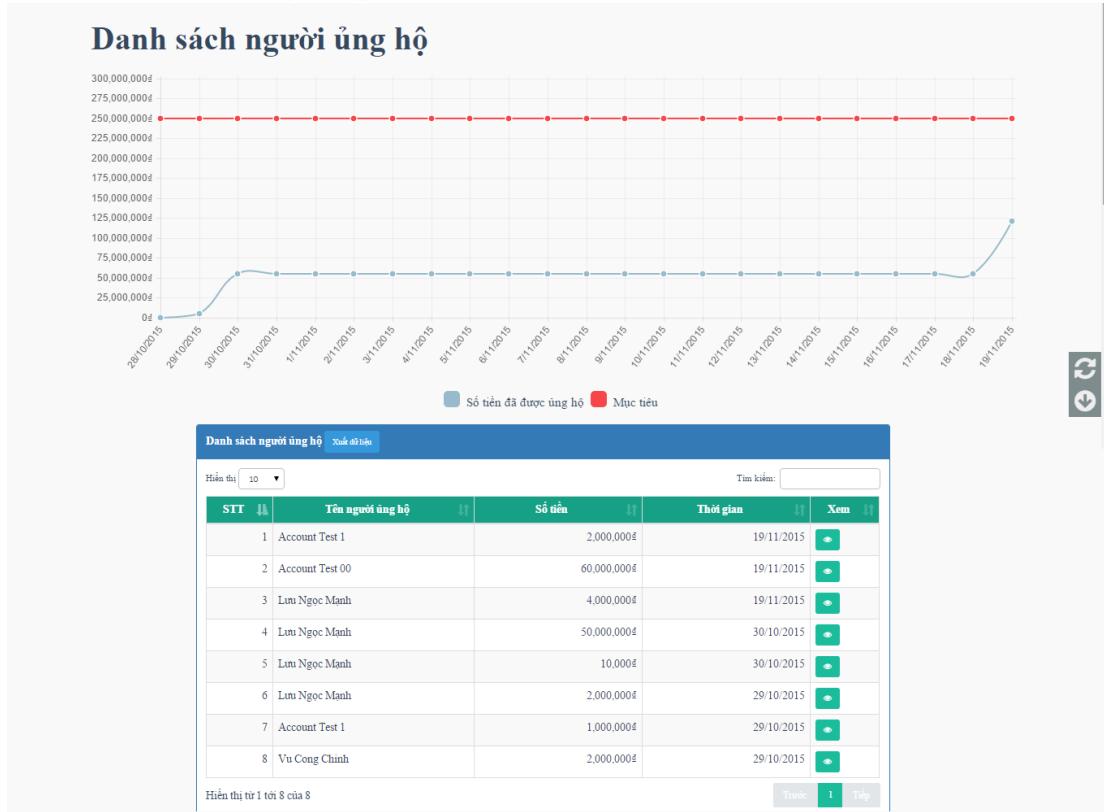
- Gây quỹ xuất bản Tuyệt Đỉnh...**  
Sửa Xóa Đang chờ kiểm duyệt  
Tuyệt Đỉnh Sinh Vật là bức danh của một nữ họa sĩ trẻ của Việt Nam, rất nổi tiếng trong cộng đồng mạng với những bức tranh minh họa  
By: Truyền Tranh | Yên Hội, Hà Nội, Việt Nam  
0% đã ứng hộ 0đ đã gây quỹ
- Bộ móng Người Sói Wolverine...**  
Sửa Xóa Đang chờ kiểm duyệt  
Bộ móng hoạt động dựa trên cảm biến phản ứng cơ bắp linh hoạt, bao chí cảm gõng hoặc thả lỏng tay để điều khiển búa múa.Norwin Sái  
By: Công Nghệ | Hà Nội, Hà Nội, Việt Nam  
0% đã ứng hộ 0đ đã gây quỹ
- DIY Apple Watch thành phiê...**  
Sửa Xóa Đang chờ kiểm duyệt  
24K ma vàng thật của thép thông minh các thiết bị, đồng hồ, đồ trang sức. Để dàng chuyển đổi đồng hồ của bạn với kit ma vàng  
By: Công Nghệ | Giảng Võ, Hà Nội, Hà Nội, Việt Nam  
0% đã ứng hộ 0đ đã gây quỹ
- Bàn cầu thông minh tích hợp ...**  
Sửa Xóa Đang chờ kiểm duyệt  
Ghế nhà vệ sinh với một fan hâm mộ được xây dựng trong đó ngay lập tức bắt đầu loại bỏ mùi hôi thông qua một khay thô nra  
By: Công Nghệ | Ô Chợ Đồn, Hà Nội, Hà Nội, Việt N...  
0% đã ứng hộ 0đ đã gây quỹ

Dự án đang hoạt động:

**Figure 4-88:** Created project screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List draft project	Danh sách các dự án tạo nháp	List				Display list draft project
2	List running project	Danh sách các dự án tạo đang chạy	List				Display list running project
3	List end date project	Danh sách các dự án tạo đã kết thúc	List				Display list end date project

4	Edit	Sửa	Button				Edit project
5	Delete	Xóa	Button				Delete project
6	List backer	Danh sách ủng hộ	Button				List backer

**Table 4-76: Created project****4.5.1.12.2 List backing of a project screen****Figure 4-89: List backing screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Chart backer	Biểu đồ ủng hộ	Chart		Y		Chart backer
2	Table list backer	Bảng danh sách người ủng hộ	Table		Y		Table list backer

**Table 4-77: List backing**

#### 4.5.1.12.3 Backed project screen

**Dự án đã ủng hộ:**

[Dự án đã ủng hộ](#) [Lịch sử ủng hộ](#)

Sản phẩm	Mô tả	Tình trạng	Đã ủng hộ (%)	Ngày còn lại	Giá quy
Sphericam	Sphericam 2 chắc chắn sẽ là 1 công cụ lưu giữ hình ảnh rất tuyệt vời bởi khả năng quay phim 360° với chất lượng video lên tới 4K	Đang chạy	24%	41 ngày còn lại	59,010,000đ đã giao quỹ
Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	Đang chạy	237%	0 ngày còn lại	71,200,000đ đã giao quỹ
Cloud Think	Cloud Think	Đang chạy	0%	42 ngày còn lại	100,000đ đã giao quỹ

Figure 4-90: Backed project screen

STT	Tên dự án	Số tiền	Thời gian	Tình trạng dự án	Xem
1	Sphericam	59,010,000đ	29/10/2015	Đang chạy	
2	Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	71,200,000đ	18/11/2015	Đang chạy	
3	Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	71,200,000đ	18/11/2015	Đang chạy	
4	Cloud Think	100,000đ	18/11/2015	Đang chạy	

Figure 4-91: Backed project history screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Backed project	Dự án ủng hộ	Hyper link		Y		Display list backed project
2	History backed project	Lịch sử ủng hộ	Hyper link		Y		Display list history backed

3	List backed project	Danh sách các dự án đã ủng hộ	List		Y		Display list backed project
4	Table backed project	Bảng lịch sử ủng hộ	Table		Y		Display list of project backed in table
5	View icon	Xem icon	Button		Y		Display info backer and project is backed

Table 4-78: Backed project

#### 4.5.1.12.4 Starred project

The screenshot shows the 'Dự án theo dõi' (Starred projects) section of the Dandelion application. It lists four projects:

- EEG Cloud**: Located in Hanoi, Vietnam. 84% funded, 1 day left. Total raised: \$4,000,000. Includes a 'Delete' button.
- Sphericam**: Located in Hanoi, Vietnam. 22% funded, 43 days left. Total raised: \$5,010,000. Includes a 'Delete' button.
- BETTER RE**: Located in Hanoi, Vietnam. 10% funded, 6 days left. Total raised: \$2,000,000. Includes a 'Delete' button.
- Gây quỹ xuất bản cuốn sách...**: Located in Da Nang, Vietnam. 4% funded, 2 days left. Total raised: \$1,200,000. Includes a 'Delete' button.

Figure 4-92: Starred project screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List starred project	Dự án đang theo dõi	List		Y		Display project is following
2	Delete	Bỏ theo dõi	Button		Y		Delete project is following

Table 4-79: Starred project

#### 4.5.1.13 Account management screen

##### 4.5.1.13.1 Account screen

The screenshot shows the 'Account' screen with the following interface elements:

- Header:** Dandelion, Danh mục, Tạo mới, Tim kiếm..., Account Test 1
- Title:** Thiết lập
- Email:** chinhthinhanh@gmail.com
- Mật khẩu:** (Current password field) with a 'Thay đổi mật khẩu' button.
- Mật khẩu hiện tại:** (Current password placeholder)
- Mật khẩu mới:** (New password placeholder)
- Xác nhận mật khẩu:** (Confirm new password placeholder)
- Buttons:** Lưu thiết lập (Save settings), Thay đổi mật khẩu (Change password).

Figure 4-93: Account screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Account	Tài khoản	Hyper link		Y		Display information account
2	Edit profile	Sửa thông tin tài khoản	Hyper link		Y		Display edit profile
3	Public profile	Thông tin công khai	Hyper link		Y		Display public profile
4	Email	Email	Text		Y		Email of user
5	Change Password	Thay đổi mật khẩu	Button		Y		Toogle hide/show the change password
6	Current password	Mật khẩu hiện tại	Text		Y		Current password
7	New password	Mật khẩu mới	Text		Y		New password
8	Confirm New	Xác nhận mật khẩu	Text		Y		Confirm New password

	password						
9	Save changes	Lưu thiết lập	Button		Y		Save changes password

**Table 4-80: Account****4.5.1.13.2 Edit profile screen**
**Figure 4-94: Edit profile screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Account	Tài khoản	Hyperlink		Y		Display information account
2	Edit profile	Sửa thông tin tài khoản	Hyperlink		Y		Display edit profile
3	Public profile	Thông tin công khai	Hyperlink		Y		Display public profile
4	Full name	Tên đầy đủ	Text		Y		Full name of user
5	Email	Email	Text		Y		Email of user
6	Avatar	Ảnh đại diện	Image				Avatar of user
7	Introduce	Giới thiệu	Textarea				Introduce about user
8	Gender	Giới tính	Dropdowntlist		Y		Gender: Male/Female
9	Date of birth	Ngày sinh	Date				Date of birth of user
10	Address	Địa chỉ	Text				Address of user
11	Phone number	Số điện thoại	Text				Phone number of user

12	Website	Website	Textarea				Website of user, company,....
13	Save changes	Lưu thiết lập	Button		Y		Save changes

Table 4-81: Edit profile

#### 4.5.1.13.3 Public profile screen

The screenshot shows a user profile page. At the top, there's a navigation bar with 'Dandelion', 'Danh mục', 'Tạo mới', and a search bar. On the right, there's a user icon and 'Account Test 1'. The main content area displays a user profile card for 'Account Test 1'. It includes a placeholder profile picture, a summary section with 'Đã tạo: 0 dự án', 'Đã ủng hộ: 1 lượt', and 'Tham gia vào: 29/10/2015', and a links section with 'www.abc' and 'Account Test 1'. Below this, there's a 'Mô tả thêm:' field and a green button labeled 'Gửi tin nhắn'.

Figure 4-95: Public profile screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Avatar	Ảnh đại diện	Image		Y		Avatar user
2	Full name	Tên đầy đủ	Label		Y		Full name of user
3	Created	Đã tạo	Label		Y		Number of project is created
4	Backed	Đã ủng hộ	Label		Y		Number of project is backed
5	Website	Website	Hyperlink		Y		Website of user, company
6	Facebook	Facebook	Hyperlink		Y		Facebook of user
7	Sent message	Gửi tin nhắn	Button		Y		Sent message to user

Table 4-82: Public profile screen

#### 4.5.2 Admin

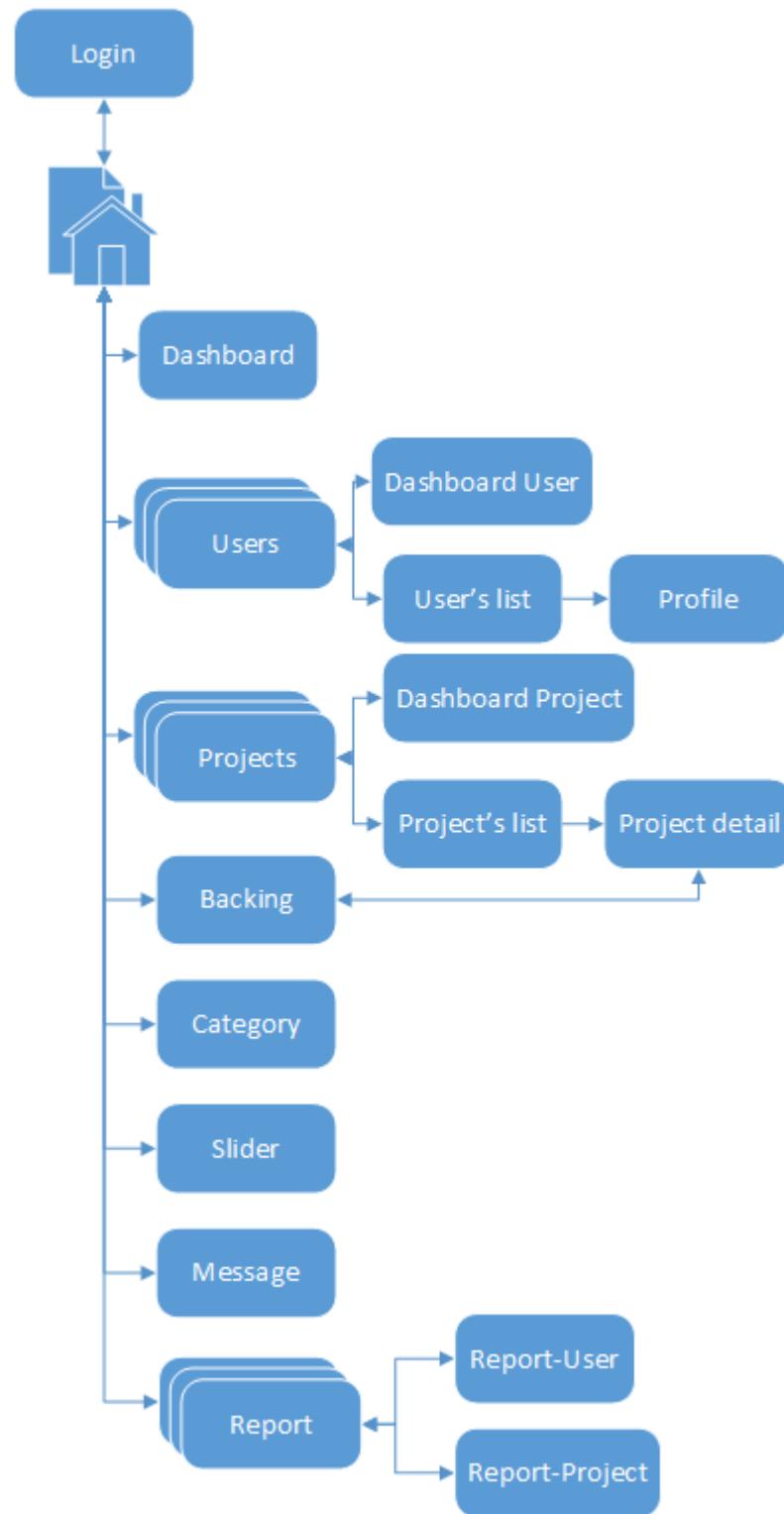
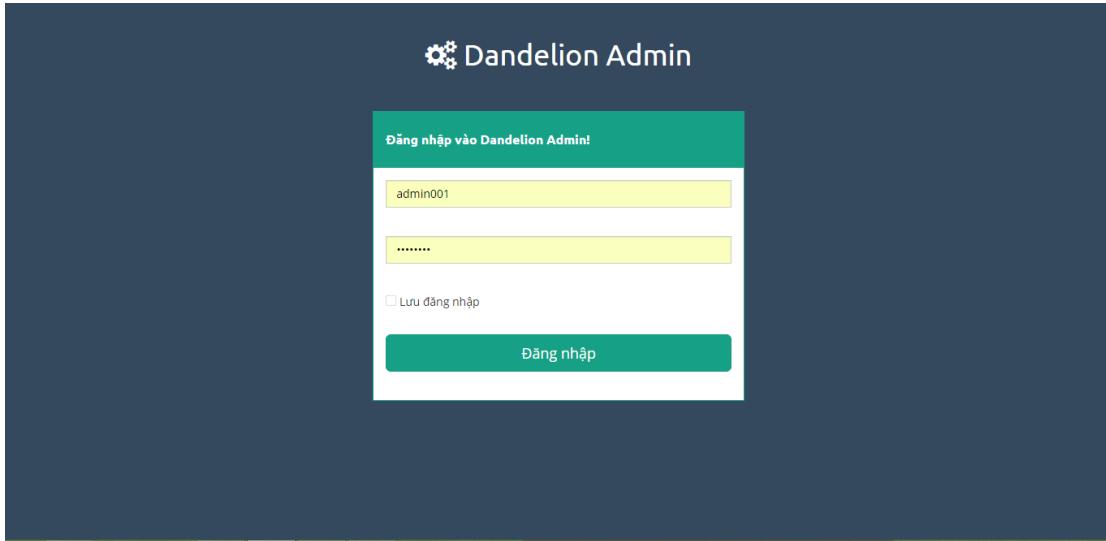


Figure 4-96: Admin screen flow

#### 4.5.2.1 Login screen



**Figure 4-97:** Login screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Username or email	Tên tài khoản hoặc email	Text box	Type	Y		Username or Email of Admin
2	Password	Mật khẩu	Text box	Type	Y		Password of registered Admin
3	Remember Me	Lưu đăng nhập	Check box	Click	Y		Save login information
4	Login	Đăng nhập	Button	Click	Y		Login

**Table 4-83:** Login

#### 4.5.2.2 Dashboard screen

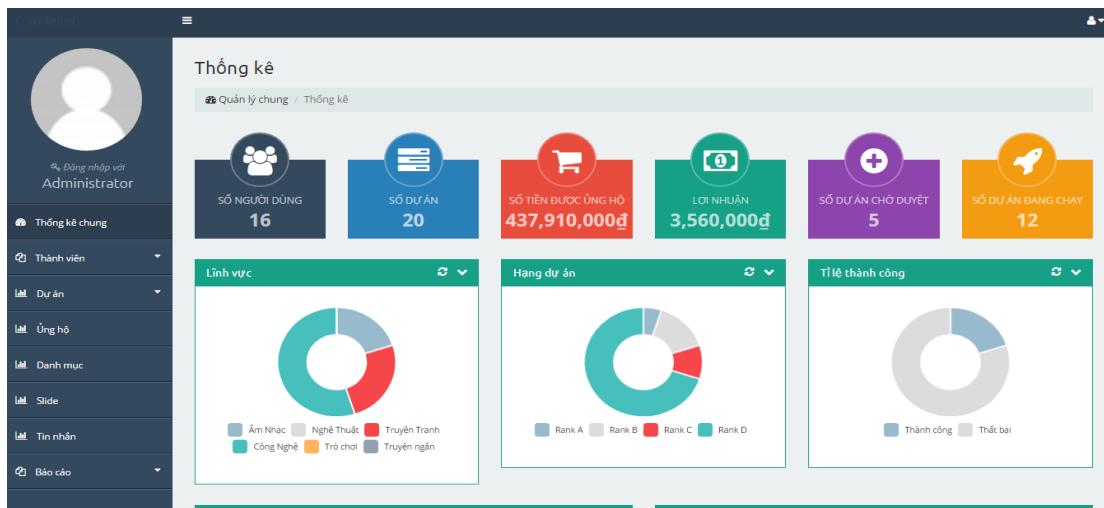


Figure 4-98: Dashboard screen part 1

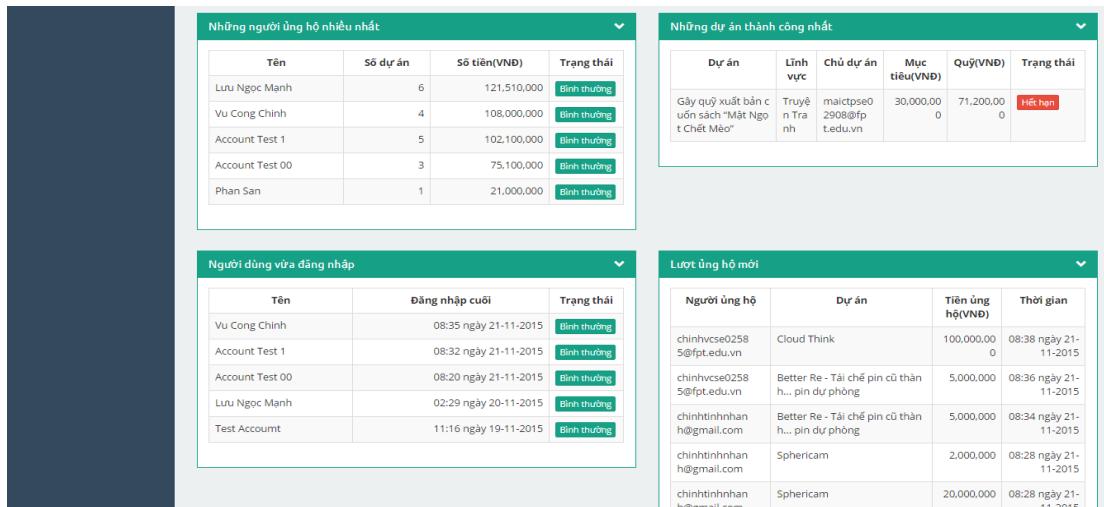


Figure 4-99: Dashboard screen part 2

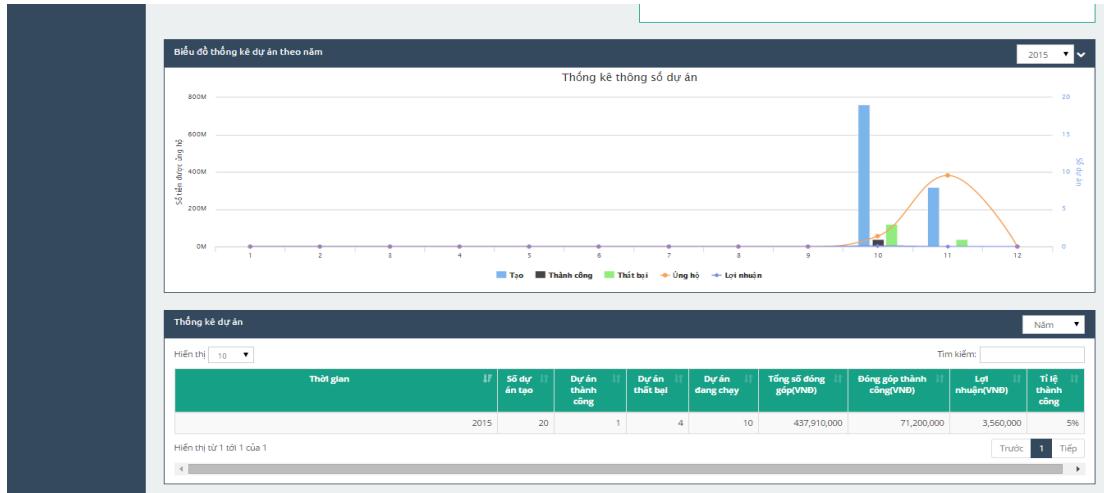


Figure 4-100: Dashboard screen part 3

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Total users	Số người dùng	Label		Y		Total users registered
2	Total project	Số dự án	Label		Y		Total project is created
3	Total back	Số tiền được ủng hộ	Label		Y		Total money is back
4	Profit	Lợi nhuận	Label		Y		Profit from projects success
5	Total project pending	Số dự án chờ duyệt	Label		Y		Total project is pending
6	Total project running	Số dự án đang chạy	Label		Y		Total project is running
7	Category pie chart	Lĩnh vực	Chart		Y		Chart display percent of item in categories
8	Rank pie chart	Hạng dự án	Chart		Y		Chart display percent, number project in Rank (A,B,C,D)
9	Rate success/fail pie chart	Tỷ lệ thành công	Chart		Y		Chart display percent, number project success, fail
10	Top backer	Người ủng hộ nhiều nhất	Table		Y		Table top 5 backer
11	Top success project	Dự án thành công nhất	Table		Y		Table top 5 project success
12	Top recent login	Người dùng vừa đăng nhập	Table		Y		Table top 5 user recent login
13	Top recent back	Lượt ủng hộ mới	Table		Y		Table top 5 recent back
14	Statistic project by Year	Biểu đồ thống kê dự án theo năm	Chart		Y		Chart Statistic project by Year
15	Statistic project	Biểu đồ thống kê dự án	Table		Y		Table Statistic project by Year

Table 4-84: Dashboard

#### 4.5.2.3 Users management screen

##### 4.5.2.3.1 Users dashboard screen

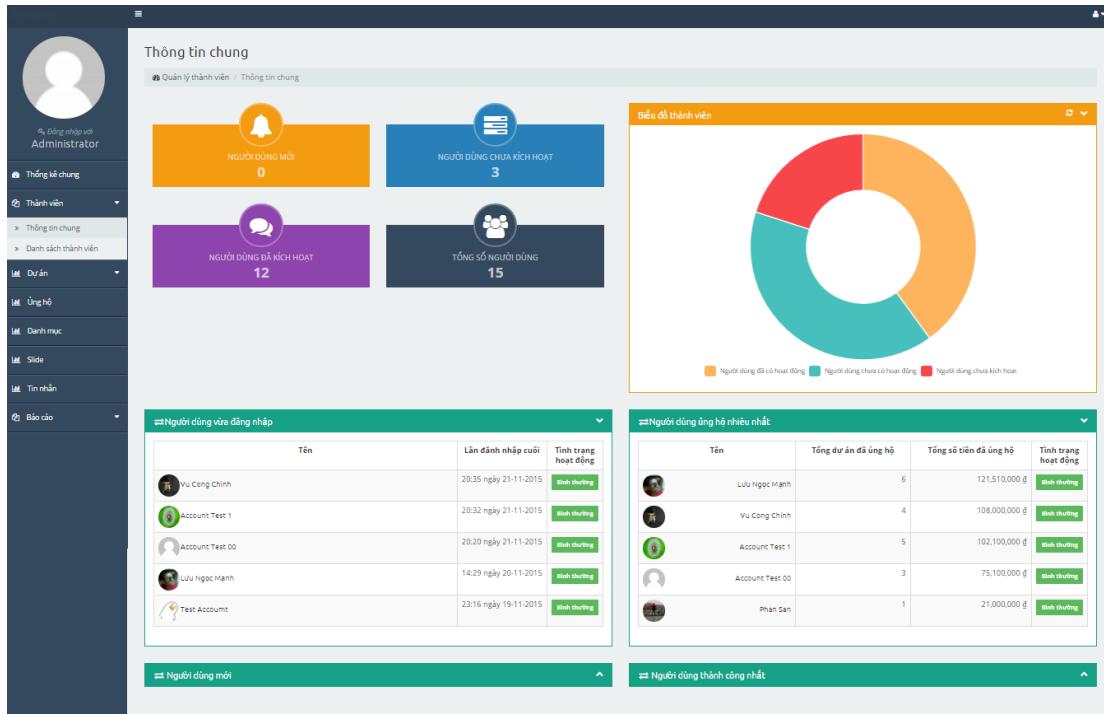


Figure 4-101: Users dashboard screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Total users	Tổng số người dùng	Label		Y		Total users registed
2	Total new users	Người dùng mới	Label		Y		Total new users
3	Total users deactive	Người dùng chưa kích hoạt	Label		Y		Total user NOT Active
4	Total users active	Người dùng đã kích hoạt	Label		Y		Total user Active
5	Member chart	Biểu đồ thành viên	Chart		Y		Member chart
6	User recent loged in	Người dùng vừa đăng nhập	Table		Y		User recent loged in

7	Most user backed	Người dùng ủng hộ nhiều nhất	Table		Y		Most user backed
8	New users	Người dùng mới	Table		Y		New users
9	Most success users	Người dùng thành công nhất	Table		Y		Most success users
10	Avatar	Ảnh đại diện	Button	Click	Y		Click on avatar of user in table will link to profile of user

**Table 4-85:Users dashboard****4.5.2.3.2 Users list screen**

The screenshot shows a user management interface. On the left is a sidebar with navigation links: Đăng nhập với Administrator, Thống kê chung, Thành viên (Thông tin chung, Danh sách thành viên), Dự án, Ứng hò, Danh mục, Slide, Tin nhắn, and Báo cáo. The main area has a header 'Quản lý thành viên / Danh sách thành viên'. Below the header are four summary boxes: 'NGƯỜI DÙNG MỚI TẠO' (0), 'NGƯỜI DÙNG CHƯA KÍCH HOẠT' (3), 'NGƯỜI DÙNG ĐÃ KÍCH HOẠT' (12), and 'TỔNG SỐ NGƯỜI DÙNG' (15). A large table titled 'Danh sách Người dùng' displays 10 rows of user data. The columns are: STT, Tên đầy đủ, Email, Số điện thoại, Loại tài khoản, Trạng thái kích hoạt, and Hành động. The table includes a search bar and a page navigation bar at the bottom.

STT	Tên đầy đủ	Email	Số điện thoại	Loại tài khoản	Trạng thái kích hoạt	Hành động
1	Lưu Ngọc Mạnh	ngocmanh1712@gmail.com	0973232734	Facebook	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
2	Test Account	test001@gmail.com	0973838383	Bình thường	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
3	Vũ Công Chính	chinhvncse02585@fpt.edu.vn		Facebook	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
4	Account Test 1	chinhvnhnhanh@gmail.com	01648214714	Bình thường	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
5	Nguyễn Huy	getbarkers@yahoo.com		Facebook	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
6	Account Test 00	maiictps02908@fpt.edu.vn		Bình thường	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
7	Mạnh Nguyễn	cropy_201@yahoo.com		Facebook	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
8	Phan San	san93s@gmail.com		Facebook	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
9	VŨ Ngọc Trung	vungoctrung93@gmail.com		Facebook	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>
10	Mai Thị Phương Cao	10204969183621586@facebook.com		Facebook	<span>Đã kích hoạt</span>	<span>Chỉnh sửa</span> <span>Xoá</span>

**Figure 4-102: Users list screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Total users	Tổng số người dùng	Label		Y		Total users registered
2	Total new users	Người dùng mới	Label		Y		Total new users
3	Total users deactive	Người dùng chưa kích hoạt	Label		Y		Total user NOT Active

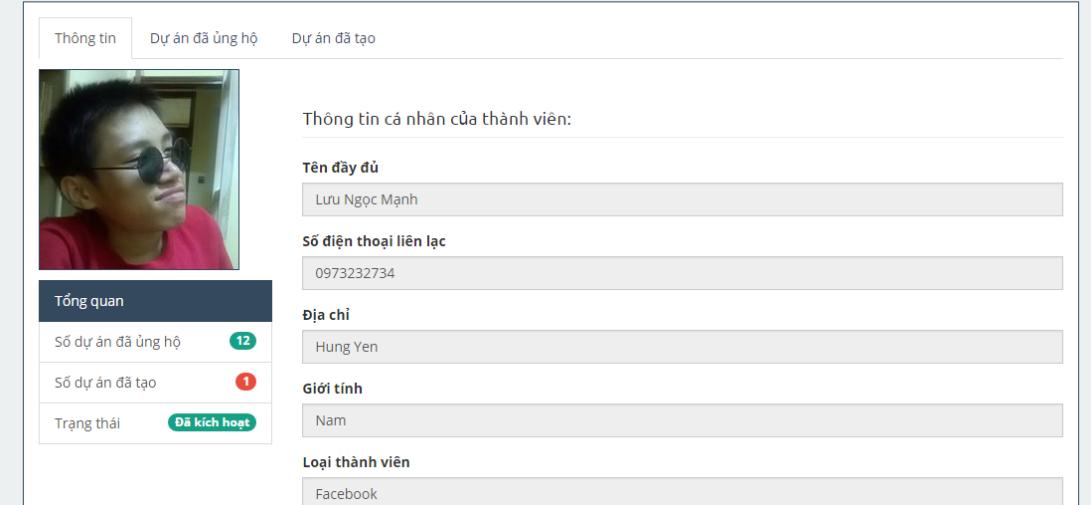
4	Total users active	Người dùng đã kích hoạt	Label		Y		Total user Active
5	List users	Danh sách người dùng	Table		Y		List users
6	View (icon)	Xem (biểu tượng)	Button	Click	Y		Link to profile of user
7	Active (icon)	Mở khóa (biểu tượng)	Button	Click	Y		Unlock/Active user
8	Deactive (icon)	Khóa (biểu tượng)	Button	Click	Y		Lock/Deactive user

**Table 4-86: Users list**

#### 4.5.2.3.3 User's profile screen

Thông tin thành viên

Quản lý thành viên / Thông tin thành viên



The screenshot shows a user profile page. At the top, there are tabs for 'Thông tin' (selected), 'Dự án đã ủng hộ' (12 projects), and 'Dự án đã tạo' (1 project). Below this is a large photo of a person wearing sunglasses. To the right, there are several input fields: 'Thông tin cá nhân của thành viên:' (Personal information of member:), 'Tên đầy đủ' (Full name: Lưu Ngọc Mạnh), 'Số điện thoại liên lạc' (Phone number: 0973232734), 'Địa chỉ' (Address: Hưng Yên), 'Giới tính' (Gender: Nam), and 'Loại thành viên' (Member type: Facebook).

**Figure 4-103: User'profile screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Infomation	Thông tin	Tab		Y		Profile
2	Backed project	Dự án đã ủng hộ	Tab		Y		Backed project
3	Created project	Dự án đã tạo	Tab		Y		Created project

**Table 4-87: User'profile**

#### 4.5.2.4 Projects management screen

##### 4.5.2.4.1 Projects dashboard screen

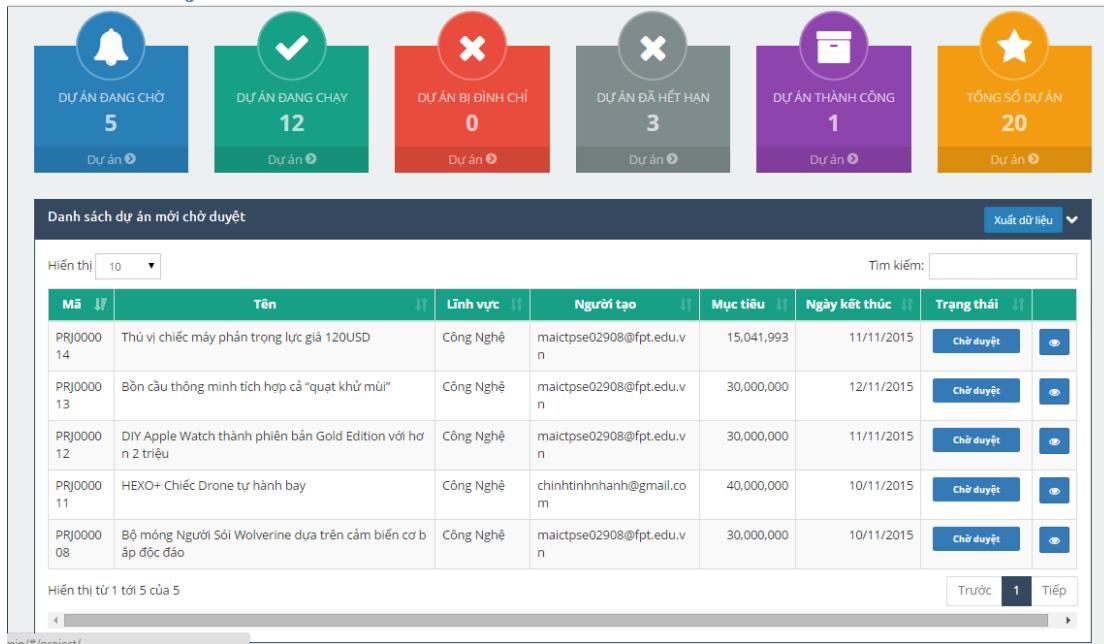


Figure 4-104: Project dashboard screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Total project pending	Số dự án đang chờ	Label		Y		Total project pending
2	Total project running	Số dự án đang chạy	Label		Y		Total project running
3	Total project postpone	Số dự án đình chỉ	Label		Y		Total project postpone
4	Total project endate	Số dự án hết hạn	Label		Y		Total project endate
5	Total project success	Số dự án thành công	Label		Y		Total project success
6	Total project	Tổng số dự án	Label		Y		Total project
7	List projects	Danh sách dự án mới chờ duyệt	Table		Y		List projects
8	View (icon)	Xem (biểu tượng)	Button	Click	Y		Link to project details

Table 4-88: Project dashboard

#### 4.5.2.4.2 Projects list screen

Thông tin chung

The screenshot shows a web-based application interface for managing projects. At the top, there's a header bar with a back arrow, the title 'Danh sách dự án / Danh sách dự án', and a 'Xuất dữ liệu' button. Below the header is a search bar labeled 'Tim kiếm:' and a dropdown for 'Hiển thị' set to 10. The main area is a table titled 'Danh sách dự án' with the following columns: Mã, Tên, Linh vực, Người tạo, Mục tiêu, Ngày kết thúc, Lượt ủng hộ, Tiền được ủng hộ, and Trạng thái. The table contains six rows of project data, each with a status indicator (e.g., 'Hết hạn', 'Đang chạy', 'Chờ duyệt') and a view icon.

Mã	Tên	Linh vực	Người tạo	Mục tiêu	Ngày kết thúc	Lượt ủng hộ	Tiền được ủng hộ	Trạng thái
PRJ00027	234 234 24	Âm Nhạc	getbarkers@yahoo.com	23,444.44 4,444	20/11/2015	0	0	Hết hạn
PRJ00005	Better Re - Tái chế pin cũ thành... pin dự phòng	Công Nghệ	maictpse02908@fpt.edu.vn	20,000.00 0	23/11/2015	3	12,000,000	Đang chạy
PRJ000013	Bồn cầu thông minh tích hợp cá “quạt khử mùi”	Công Nghệ	maictpse02908@fpt.edu.vn	30,000.00 0	12/11/2015	0	0	Chờ duyệt
PRJ00008	Bộ móng Người Sói Wolverine dự a trên cát biển cơ bắp độc đáo	Công Nghệ	maictpse02908@fpt.edu.vn	30,000.00 0	10/11/2015	0	0	Chờ duyệt
PRJ000021	Cloud Think	Công Nghệ	cropx_201@yahoo.com	500,000.00	31/12/2015	5	110,300,000	Đang chạy
PRJ0000011	Divinity: Original Sin 2	Âm Nhạc	ngocmanh1712@gmail.com	100,000.00	05/11/2015	1	200,000	Đang chạy

Figure 4-105: Projects list screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List projects	Danh sách dự án mới chờ duyệt	Table		Y		List projects
2	View (icon)	Xem (biểu tượng)	Button	Click	Y		Link to project details

Table 4-89: Project list

#### 4.5.2.4.3 Project detail screen

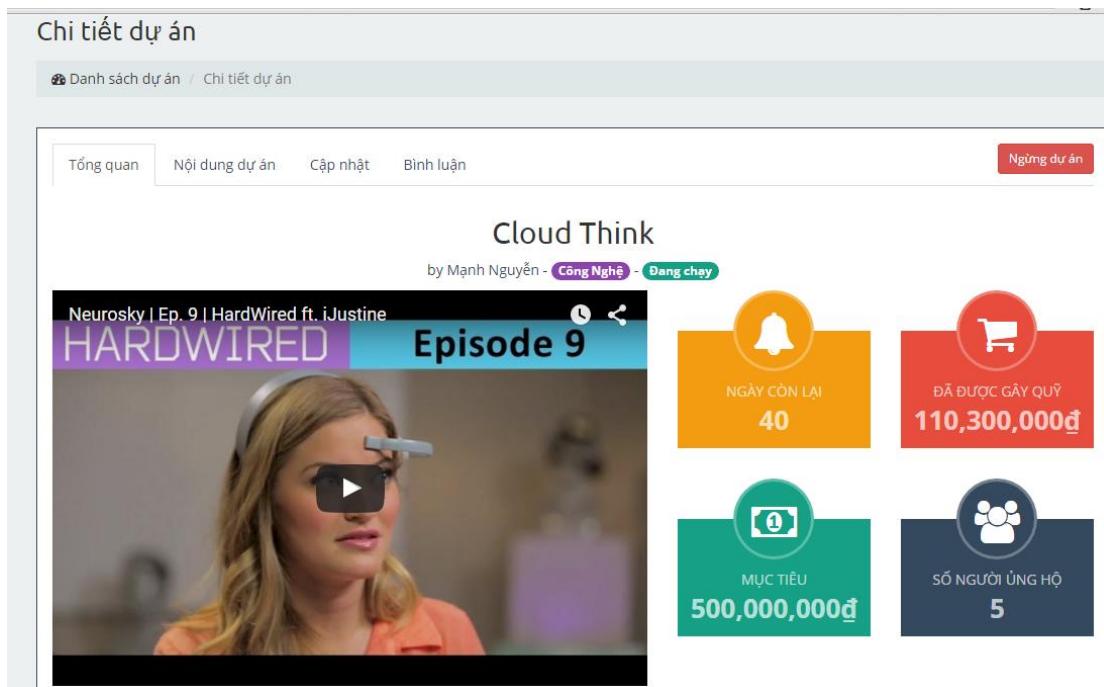
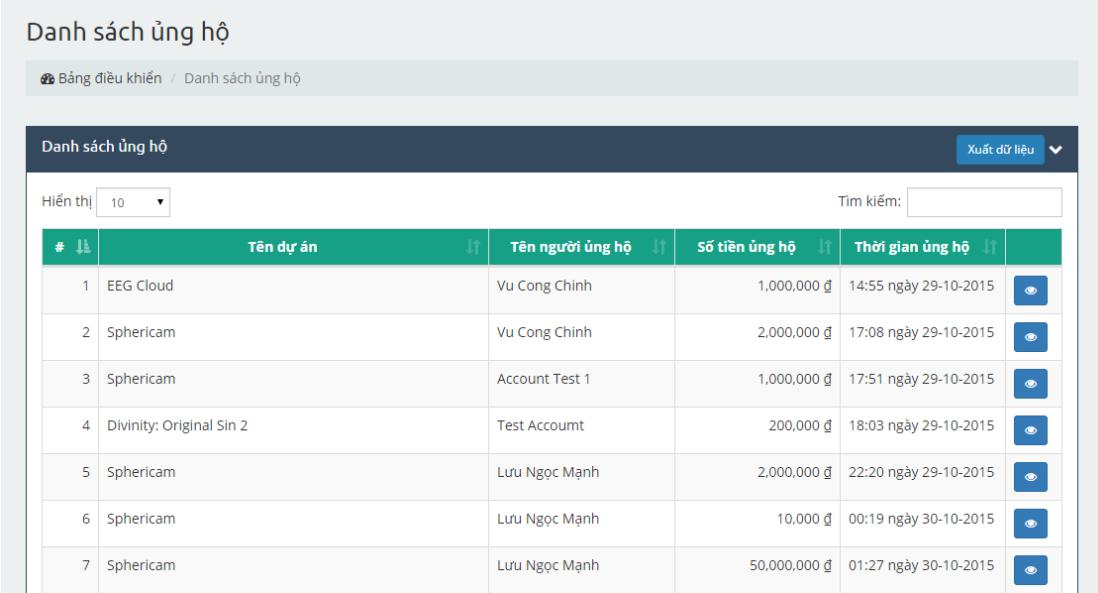


Figure 4-106: Project detail screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Infomation	Tổng quan	Tab		Y		Profile
2	Backed project	Nội dung dự án	Tab		Y		Backed project
3	Created project	Cập nhật	Tab		Y		Created project
4	Comment	Bình luận	Tab		Y		Comment
5	Accept	Chấp nhận	Button		Y		Accept to project running
6	Reject	Từ chối	Button		Y		Reject project
7	Postpone	Ngừng dự án	Button		Y		Postpone project

Table 4-90: Project detail

#### 4.5.2.5 Backing management screen



The screenshot shows a web-based application interface for managing backings. At the top, there's a header bar with the title "Danh sách ủng hộ". Below it is a breadcrumb navigation: "Bảng điều khiển / Danh sách ủng hộ". The main area is a table titled "Danh sách ủng hộ" with columns: #, Tên dự án, Tên người ủng hộ, Số tiền ủng hộ, Thời gian ủng hộ, and actions (represented by blue icons). The table contains 7 rows of data.

#	Tên dự án	Tên người ủng hộ	Số tiền ủng hộ	Thời gian ủng hộ	
1	EEG Cloud	Vu Cong Chinh	1,000,000 ₫	14:55 ngày 29-10-2015	
2	Sphericam	Vu Cong Chinh	2,000,000 ₫	17:08 ngày 29-10-2015	
3	Sphericam	Account Test 1	1,000,000 ₫	17:51 ngày 29-10-2015	
4	Divinity: Original Sin 2	Test Accountt	200,000 ₫	18:03 ngày 29-10-2015	
5	Sphericam	Lưu Ngọc Mạnh	2,000,000 ₫	22:20 ngày 29-10-2015	
6	Sphericam	Lưu Ngọc Mạnh	10,000 ₫	00:19 ngày 30-10-2015	
7	Sphericam	Lưu Ngọc Mạnh	50,000,000 ₫	01:27 ngày 30-10-2015	

Figure 4-107: Backing list screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List backing	Danh sách ủng hộ	Table		Y		List backing
2	View (icon)	Xem (biểu tượng)	Button	Click	Y		Link to backing details

Table 4-91: Backing list

#### 4.5.2.6 Category management screen



The screenshot shows a web-based application interface for managing categories. At the top, there's a header bar with the title "Quản lý danh mục". Below it is a breadcrumb navigation: "Quản lý danh mục / Danh sách danh mục". The main area is a table titled "Danh sách danh mục" with columns: ID, Tên danh mục, Mô tả, Số dự án, and Trạng thái. The table contains 7 rows of data. A footer at the bottom shows a page number "Trước 1 Tiếp" and a note "Hiển thị từ 1 đến 6 của 6".

ID	Tên danh mục	Mô tả	Số dự án	Trạng thái
1	Âm Nhạc	abncbnbc cbcn nbcnbc	4	Mở
2	Nghệ Thuật	abncbnbc cbcn nbcnbc	0	Mở
3	Truyện Tranh	abncbnbc cbcn nbcnbc	5	Mở
5	Công Nghệ	abncbnbc cbcn nbcnbc	11	Mở
6	Trò chơi	trò chơi, ứng dụng điện thoại, video game	0	Mở
7	Truyện ngắn	không có j cả chỉ là test thôi	0	Khóa

Figure 4-108: Category management screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List categories	Danh sách danh mục	Table		Y		List categories
2	Open (icon)	Mở (biểu tượng)	Button	Click	Y		Unlock a category
3	Close (icon)	Khóa (biểu tượng)	Button	Click	Y		Lock a category
4	Edit (icon)	Sửa (biểu tượng)	Button	Click	Y		Edit or update a category
5	Add new category	Thêm danh mục mới	Button	Click	Y		Display popup to create new category

Table 4-92: Category management

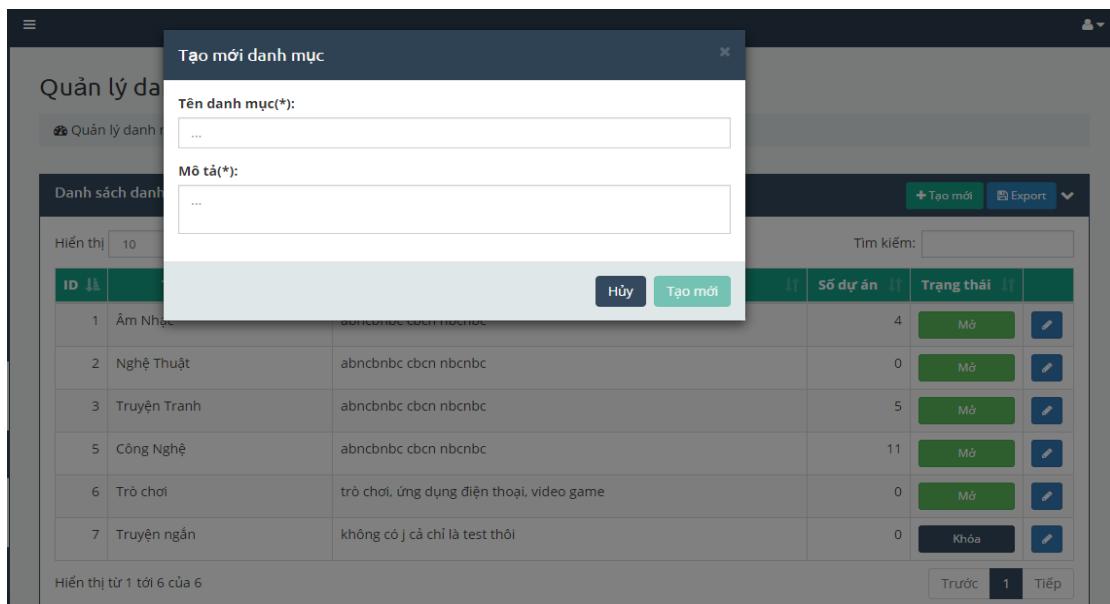


Figure 4-109: New category dialog

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Category name	Tên danh mục	Text		Y		Name of new category

2	Description	Mô tả	Text		Y		Description of new category
3	Create	Tạo	Button	Click	Y		Category is created
4	Cancel	Hủy	Button	Click	Y		Cancel the create category

Table 4-93: New category dialog

#### 4.5.2.7 Slide management screen

The screenshot shows a table titled "Danh sách Slide" (Slide list) with the following columns:

- #
- Tiêu đề (Title)
- Mô tả (Description)
- Ảnh (Image)
- Trạng thái (Status)
- Thứ tự (Order)

There are four rows of data:

#	Tiêu đề	Mô tả	Ảnh	Trạng thái	Thứ tự
1	The World's Cleanest Power Plant	Support a team of architects who are working to reduce carb on emissions by making art in the sky.		Mở (Open)	
2	Eco - Global Survival Game	Collaborate to build civilization in a simulated ecosystem, creating laws to make group decisions.		Mở (Open)	
3	Emmett Louis Till, 1941-1955	is murder catalyzed the civil rights movement. Help make the film that will tell his story.		Mở (Open)	
4	khu vườn trên mây	slide của khu vườn trên mây		Mở (Open)	

At the bottom, there are navigation buttons: Trước (Previous), Tiếp (Next), and a search bar labeled "Tim kiếm:".

Figure 2-4: Slide management screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List slide	Danh sách slide	Table		Y		List slide
2	Open (icon)	Mở (biểu tượng)	Button	Click	Y		Unlock a slide
3	Close (icon)	Khóa (biểu tượng)	Button	Click	Y		Lock a slide
4	Up (icon)	Lên trên (biểu tượng)	Button	Click	Y		Set order of slide
5	Down (icon)	Xuống dưới (biểu tượng)	Button	Click	Y		Set order of slide
6	Edit (icon)	Sửa (biểu tượng)	Button	Click	Y		Edit or update a slide

7	Delete (icon)	Xóa (biểu tượng)	Button	Click	Y		Delete a slide
8	Add new slide	Thêm danh mục mới	Button	Click	Y		Display popup to create new slide

**Table 4-94: Slide management**

The screenshot shows a dialog box titled "Tạo Slide mới". It contains several input fields and dropdown menus:

- Tiêu đề(\*):** Nhập tiêu đề...
- Mô tả(\*):** Nhập mô tả...
- Đường dẫn của Slide(\*):** http://...
- Màu chữ(\*):** Sáng (dropdown menu)
- Ảnh bìa(\*):** Chọn tệp (button), Không có tệp nào được chọn (text)
- Video (nếu có):** youtube...
- Chữ trong nút bấm(\*):** Chữ gì đó...
- Màu nút(\*):** Xanh lục (dropdown menu)

At the bottom right are two buttons: **Hủy** (Cancel) and **Tạo mới** (Create New).

**Figure 4-110: New slide dialog screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	Slider name	Tiêu đề	Text		Y		Name of new Slider
2	Description	Mô tả	Text		Y		Description of new category
3	Path Url	Đường dẫn	Text		Y		Link of slides can link to project

4	Text color	Màu chữ	Drop down	Click	Y		Select “sáng/tối” category
5	Image	Ảnh bìa	Button	Click	Y		Background of slide
6	Video	Video	Text				Video of slide
7	Name in button	Chữ trong nút bấm	Text		Y		Name in button
8	Color button	Màu nút	Drop down	Click	Y		Color button
9	Create	Tạo mới	Button	Click	Y		A new slide is created
10	Cancel	Hủy	Button	Click	Y		Cancel the create slide

**Table 4-95: New slide dialog screen****4.5.2.8 Message (same as message of user)**

Tin nhắn

[Tin nhắn](#) / Danh sách tin nhắn

Tin nhắn		<input type="checkbox"/> Soạn tin nhắn	<input type="checkbox"/> Xóa tin nhắn	Tim kiếm:
<b>Tất cả thư</b>		<input type="checkbox"/> Tới: Account Test 1 Dandelion - duyệt thành công dự án Thiết bị đeo Vufine: Biển kinh thường t hành màn hình! 21/11/15 <input type="checkbox"/> Tới: Account Test 00 Dandelion - duyệt thành công dự án Gây quỹ Xuất bản sách "Truyện Cực Ng ăn" của Đào Quang Huy! 21/11/15 <input type="checkbox"/> Tới: Account Test 1 Dandelion - duyệt thành công dự án Máy tính siêu tí hon chơi game lướt we b có giá chỉ... 9 USD! 21/11/15 <input type="checkbox"/> Tới: Account Test 1 Dandelion - duyệt thành công dự án Silk Light - Bóng đèn thông minh "đọc v i" đồng hồ sinh học! 21/11/15 <input type="checkbox"/> Từ: Test Accountt Test gửi tin nhắn 19/11/15 <input type="checkbox"/> Tới: Account Test 00 Dandelion - duyệt thành công dự án Gây quỹ ủng hộ dự án xuất bản "Quan t rọng là phải Đẹp Trai"! 19/11/15		
Hộp thư đến	Hộp thư đi			

**Figure 4-111: Message screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	New message	Soạn tin nhắn	Button	Click	Y		Display new message form
2	All message	Tất cả thư	Button	Click	Y		Filter table as all message
3	Inbox	Hộp thư đến	Button	Click	Y		Filter table as inbox message

4	Sent	Hộp thư đi	Button	Click	Y		Filter table as sent message
6	Delete message	Xóa tin nhắn	Button	Click	Y		Delete message selected
7	Table message	Danh sách tin nhắn	Table		Y		Table contain messages

**Table 4-96: Message****4.5.2.9 Report management****4.5.2.9.1 Report user management**

Báo xấu người dùng

Quản lý chung / Báo xấu người dùng

Danh sách người dùng bị báo xấu

#	Người báo xấu	Người bị báo xấu	Ngày báo	Trạng thái
Không có dữ liệu				

Hiển thị 10 ▾

Tìm kiếm:

Không có dữ liệu

Hiển thị từ 0 đến 0 của 0

Trước Tiếp

**Figure 4-112: Report user management screen**

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List report user	Danh sách người dùng bị báo xấu	Table		Y		List report user
2	View (icon)	Xem (biểu tượng)	Button	Click	Y		View content of the report
3	Accept (icon)	Đồng ý (biểu tượng)	Button	Click	Y		Accept with the report
4	Cancel (icon)	Hủy (biểu tượng)	Button	Click	Y		Cancel with the report

**Table 4-97: Report user management**

#### 4.5.2.9.2 Report project management screen

Figure 4-113: Report project management screen

No	Field name	Field name in Vietnamese	Type	Require	Mandatory	Max-Length	Description
1	List report project	Danh sách người dùng bị báo cáo	Table		Y		List report project
2	View (icon)	Xem (biểu tượng)	Button	Click	Y		View content of the report
3	Accept (icon)	Đồng ý (biểu tượng)	Button	Click	Y		Accept with the report
4	Cancel (icon)	Hủy (biểu tượng)	Button	Click	Y		Cancel with the report

Table 4-98: Report project management

## 5 SOFTWARE TESTING

### 5.1 Introduction

#### 5.1.1 Purpose

This is the comprehensive test plan of the DDL project. The purpose of this chapter describes scopes of test and activities which need to be taken during test process of project. It addresses the following items: Scopes of Testing, Requirements for Testing, Test Strategy, Test Resources, Test Environment, Test Milestones and Deliverables.

#### 5.1.2 Definitions and Acronyms

This section describes the definitions, terms, and acronyms that are used in software requirements specification.

Acronym	Definition	Note
DDL	Dandelion Project	
GUI	Graphic User Interface	
IT	Integration Test	
KLOC	1000 line of code	
PM	Project Manager	
QA	Quality Assurance	
SRS	Software Requirement Specification	
ST	System Test	
TC	Test Case	
TL	Test Leader	
TP	Test Plan	
TR	Test Report	
UT	Unit Test	

**Table 5-1: Definitions and Acronyms**

### 5.1.3 Scope of testing

DDL will be tested by 4 phases:

#### Phase 1: Unit testing

- Unit testing will be done by developers
- Developers user White Box Testing technique to do
- When executing unit testing, if any bugs are found, developers have to log bug on “Defect Log Management” file and fix it until it is correct.

*Rule for filling test result:*

Test result pass	Pass
Test result fail	Fail
Do not test	Untested
Cannot test	N/A (Not available)

**Table 5-2: Rule unit test**

#### Phase 2: Integration testing

- After finishing component testing, integration testing will be performed by testers.
- Material are integration test cases, high- level design and test tools.

- Integration test focuses on specific areas of use cases when all requirements are completed.
- Integration test should be performed to ensure all components incorporate well.
- When executing integration testing, if any bugs are found, testers have to log on “Defect Log Management” file and assign to developer fix it and redo this process until it is correct.

*Rule for filling test result:*

Test result pass	Pass
Test result fail	Fail
Do not test	Untested
Cannot test	N/A (Not available)

**Table 5-3: Rule integration test**

### **Phase 3: System testing**

- After finishing integration testing and developers collect all functions and items, testers will be performed system testing, it means doing test whole system.
- Material area system test case, SRS
- If any bugs are found, developers have to fix and testers will verify them. System test is ended only when test cases are passed and no bug is found.

*Rule for filling test result:*

Test result pass	Pass
Test result fail	Fail
Do not test	Untested
Cannot test	N/A (Not available)

**Table 5-4: Rule system test**

### **Phase 4: Acceptance testing**

- Base on customer/user requirement specification, system is tested again, for ensure there is not lacking or mistake any requirement.
- If there is any problem, developers have to fix/update and tester will verify them.
- Acceptance testing is ended only when whole system met requirement specification.

#### **5.1.4 Constraints**

- Deadline for testing only can be met if development progress is on time.

- Test execution can be performed when system passes Unit Test Inspection.
- At least one round of testing must be performed for requirements.
- Have more environments should be tested: Window 10, Window 8... and more browsers: Firefox 30 and Google Chrome 40...

## 5.2 Test plan

### 5.2.1 Test type

#### 5.2.1.1 Function Testing

- Functional testing is a type of software testing whereby the system is tested against the functional requirements/specifications.
- Functions are tested by feeding them input and examining the output. Functional testing ensure that the requirements are properly satisfied by the website. This type of testing is not concerned with how processing occurs, but rather, with the results of processing.
- During functional testing, Black Box Testing technique is used in which the internal logic of the system being tested is not known to the tester.

<b>Test Objective:</b>	The type of this test is to ensure proper target-of-test functionality, including user interaction, all function defined in specification document implemented correctly.
<b>Technique:</b>	Executing each use case, use-case flow, or function, using valid and invalid data, to verify the following: <ul style="list-style-type: none"> <li>- The expected results occur when valid data is used.</li> <li>- The appropriate error or warning messages are displayed when invalid data is used.</li> <li>- Each business rule is properly applied.</li> </ul>
<b>Completion Criteria:</b>	<ul style="list-style-type: none"> <li>- All planned tests have been executed.</li> <li>- All identified defects have been addressed and closed.</li> </ul>
<b>Special Considerations:</b>	Testing may be stopped when Time runs out A certain number of defects found Test coverage > 97% Stop when testing becomes unproductive

**Table 5-5: Function Testing**

### 5.2.1.2 User Interface Testing

GUI testing is the process of ensuring proper functionality of the GUI for a given web and making sure it conforms to its written specifications.

GUI testing evaluates design elements such as layout, colors, fonts, font sizes, labels, text boxes, text formatting, captions, buttons, lists, icons, links, content and more.

<b>Test Objective:</b>	Verify the following: <ul style="list-style-type: none"> <li>- Navigation through the target-of-test properly reflects business functions and requirements, including window-to-window, field-to-field, and use of access methods (tab keys, mouse movements, accelerator keys)</li> <li>- Window objects and characteristics, such as menus, size, position, state, and focus conform to standards.</li> </ul>
<b>Technique:</b>	Create or modify tests for each window to verify proper navigation and object states for each application window and objects.
<b>Completion Criteria:</b>	Each window successfully verified to remain consistent with benchmark version or within acceptable standard
<b>Special Considerations:</b>	Not all properties for custom and third party objects can be accessed.

**Table 5-6: GUI Testing**

### 5.2.1.2.1 Data and Database Integrity Testing

The databases and the database processes should be tested as a subsystem within the Project. These subsystems should be tested without the target-of-test's User Interface as the interface to the data. Additional research into the Database Management System (DBMS) needs to be performed to identify the tools and techniques that may exist to support the testing identified below.

<b>Test Objective:</b>	Ensure database access methods and processes function properly and without data corruption.
<b>Technique:</b>	<ul style="list-style-type: none"> <li>- Invoke each database access method and process, seeding each with valid and invalid data or requests for data.</li> <li>- Inspect the database to ensure the data has been populated as intended, all database events occurred properly, or review the returned data to ensure that the correct data was retrieved for the correct reasons.</li> </ul>
<b>Completion Criteria:</b>	All database access methods and processes function as designed and without any data corruption.

<b>Special Considerations:</b>	<ul style="list-style-type: none"> <li>- Testing may require a DBMS development environment or drivers to enter or modify data directly in the databases.</li> <li>- Processes should be invoked manually.</li> <li>- Small or minimally sized databases (limited number of records) should be used to increase the visibility of any non-acceptable events.</li> </ul>
--------------------------------	---

**Table 5-7: Data and Data Integrity Testing**

### 5.2.2 Test stages

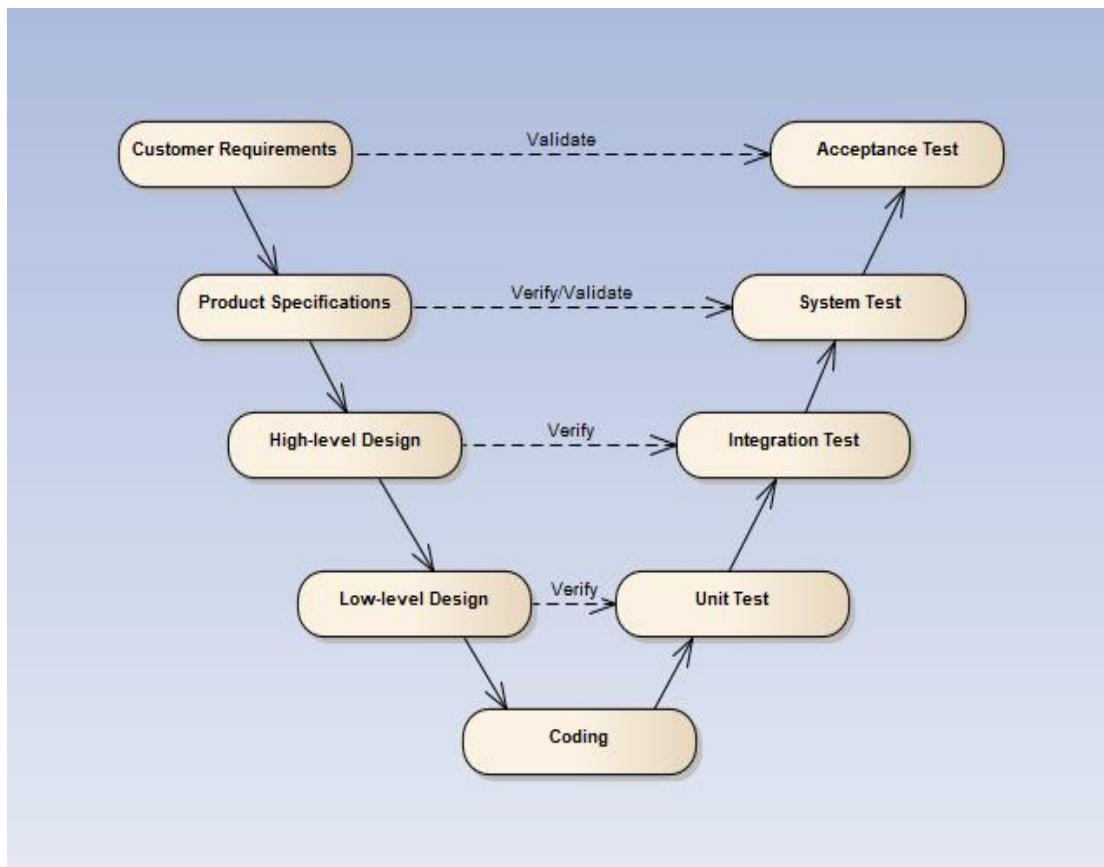
Clearly state the stage in which the test will be executed. Identified below are the stages in which common test are executed

<b>Type of Tests</b>	<b>Stage of Test</b>			
	<b>Unit</b>	<b>Integration</b>	<b>System</b>	<b>Acceptance</b>
Function Testing	X	X	X	X
User Interface Testing	X		X	X

**Table 5-8: Test stages**

### 5.2.3 Test model

DDL follows V-Model process:

**Table 5-9: V-model process**

Testing progress is divided to 5 phases include: Unit test, Component test, Integration test, System test and Acceptance test

- Unit test:
  - Unit testing is used to verify a single minimal unit of source code. The purpose of unit testing is to isolate the smallest testable parts of DDL and verify that they function properly in isolation.
  - Unit testing is the first level of testing and is performed prior to component testing
  - Unit testing will be done by developer.
- Component test:
  - Component testing is used to validate a combined many minimal units of source code.
  - Component testing is performed after unit testing and before integration testing
  - Component testing will be done by tester
- Integration test:
  - Integration testing is a level of the software testing process where individual units or components are combined and tested as a group.
  - The purpose is to expose faults in the interaction between integrated units.
  - Integration testing is performed after component testing
  - Integration testing will be done by tester
  - There are two methods of doing integration testing: Bottom-up Integration testing and Top Down Integration testing:

No	Integration Testing Method
1	Bottom-up integration This testing begins with unit testing, followed by tests of progressively higher-level combinations of units called modules.
2	Top- Down integration This testing, the highest-level modules are tested first and progressively lower-level modules are tested after that

**Figure 5-1: Integration test**

- System test:
  - System Testing is a level of the software testing process where a complete, integrated system is tested
  - The purpose is to evaluate the system's compliance with the specified requirements

- System testing is performed after integration testing
- System testing will be done by tester
- Acceptance test:
  - Acceptance testing is performed after system testing
  - Acceptance testing will be performed by the test leader and team leader.
  - The acceptance test will be done for a period of 1 weeks after completion of the system test process

#### 5.2.4 Acceptance test criteria.

- Criteria for Unit test of Development team, for Test team accepts to start testing:
  - Number of TC/KLOC: 40TC/KLOC
  - Number defects/KLOC: 3-4 defects/KLOC
  - Statement coverage: 100%
  - Branch coverage: 100%
  - Path coverage: 100%
- Criteria for Integration test:
  - Number of TC/KLOC: 30 TC/KLOC
  - Number defects/KLOC: 2-3 defects/KLOC
- Criteria for System test:
  - Number of TC/KLOC: 20 TC/KLOC
  - Number defects/KLOC: 4-6 defects/KLOC
- Criteria for Acceptance test:
  - Number defects/KLOC: 1-2 defects/KLOC

#### 5.2.5 Futures to be tested

Refer [3.3.1.2 Product features](#) for more detail.

#### 5.2.6 Feature not to be tested

- The stable of website when do not connect internet.
- Over than 5000 users connect to system at the same time.

#### 5.2.7 Resources

##### 5.2.7.1 Human resource

Worker/Doer	Role	Specific Responsibilities/Comments
MaiCTP	Test Leader	Manage Test Resource and assign test tasks. Create and review Test Plan. Create and review Test Case. Execute test.

		Create and review Test Report
ChinhVC	Tester	Create and review Test Case. Execute test. Create Test view points Create and review Test Report

**Table 5-10: Human resource****5.2.7.2 Test Environment****5.2.7.2.1 Hardware**

Name	Purpose	Detail
Laptop HP	Device for create and execute test	Window 8.1 Pro Core i3
Laptop Asus	Device for create and execute test	Window 8.1 Pro Core i5

**Table 5-11: Hardware****5.2.7.2.2 Software**

Name	Purpose	Detail
Test Plan	Managing test	Microsoft Word 2010, 2013
Test case	Executing test	Microsoft Excel 2010, 2013
Test report, Test checklist	Tracking test	Microsoft Excel 2010, 2013
Chrome	Executing test	Chrome 40.0
Firefox	Executing test	Firefox

**Table 5-12: Software****5.2.7.2.3 Infrastructure**

Name	Purpose	Detail	Vendor/In-house	Version
DDL_Defect Log Management_v1.0_EN	Tracking bug during testing time	Microsoft Excel 2013, 2010	FPT-University	v1.0
Test Effort	Effort execute test	Microsoft Excel 2013, 2010	FPT-University	v1.0

**Table 5-13:Infrastructure****5.2.8 Test milestones**

Milestone Task	Effort (pd)	Start Date	End Date
----------------	-------------	------------	----------

Create Test Plan	6		
Review & Update Test Plan	2		
Create Component Test case	12		
Review & Update Component Test Case	2		
Create Integration Test case	2		
Review & Update Integration Test Case	2		
Create System Test case	6		
Review & Update System Test Case	2		
Execute Component test phase 1	1		
Execute Component test phase 2	1		
Execute Integration test phase 1	2		
Execute Integration test phase 2	2		
Execute System test phase 1	4		
Execute System test phase 2	4		

**Table 5-14: Test milestones**

### 5.2.9 Deliverables

No	Deliverables	Responsibilities	Delivered Date
1	Test Plan	Tester	
2	Component Test case	Tester	
3	Integration Test case	Tester	
4	System Test case	Tester	
5	Defect Log Management	All members	
6	Test report	Test Leader	

**Table 5-15:Deliverables**

## 5.3 Test case

Refer to:

- DDL\_Integration Test Case\_v1.0\_EN.xlsx
- DDL\_System Test Case\_v1.0\_EN.xlsx
- DDL\_Common Test Case\_v1.0\_EN.xlsx
- DDL\_UnitTestCase\_CategoryRepository\_v1.0.xlsx
- DDL\_UnitTestCase\_MessageRepository\_v1.0.xlsx
- DDL\_UnitTestCase\_ProjectRepository\_v1.0.xlsx
- DDL\_UnitTestCase\_SlideRepository\_v1.0.xlsx

- DDL\_UnitTestCase\_UserRepository\_v1.0.xls

## 5.4 Test Report

### 5.4.1 System test report

No	Module Code	Pass	Fail	N/A	Number of Test Case
1	Common	20	0	0	20
2	Display Personal Page	30	0	0	30
3	Account management module	122	0	0	122
4	Create Edit Project	228	0	0	228
5	Project detail	88	0	0	88
6	Back Project	46	0	0	46
7	Project management	62	0	0	62
8	Discover	18	0	0	18
9	Statistic	24	0	0	24
10	Message	52	0	0	52
11	Admin Module	108	0	0	108
	<b>Total</b>	<b>812</b>	<b>0</b>	<b>0</b>	<b>812</b>

Table 5-16: System test report

### 5.4.2 Integration test report

No	Module Code	Pass	Fail	N/A	Number of Test Case
1	Registered_User_function	188	0	0	118
2	Admin_Function	110	2	0	112
	<b>Total</b>	<b>298</b>	<b>2</b>	<b>0</b>	<b>300</b>

Table 5-17: Integration test report

### 5.4.3 Test report

No	Module Code	Round 1		Round 2		Round 3		Final
		Pass	Fail	Pass	Fail	Pass	Fail	
1	Unit Test	356	0	356	0	356	0	356
2	Integration Test	289	2	300	0	300	0	300
3	System Test	692	64	714	60	812	0	812
	<b>Total</b>	<b>1337</b>	<b>66</b>	<b>1370</b>	<b>60</b>	<b>1468</b>	<b>0</b>	<b>1468</b>

Table 5-18: Test Report

### 5.4.4 Defect report

Defects	Logged	%Closed	%Invalid	%Open
<b>Critical</b>	0	0%	0%	0%
<b>High</b>	4	0%	0%	0%
<b>Medium</b>	24	100%	0%	0%
<b>Low</b>	38	100%	0%	0%
<b>Total</b>	66	100%	0%	0%

## 6 USER MANUAL

### 6.1 Introduction

#### 6.1.1 Purpose

This document contains guide-lines step by step to deploy DDL App to Azure website and This user's manual has been developed to help users better understand the requirements that may apply to use this website by providing them simple step by step tutorials. This manual is intended to help make website's functions more accessible and their associated requirements more understandable to users.

#### 6.1.2 Environment

Following are the software required to start the DDL system:

- Operating System: Windows 7, Windows 8, Window 10
- Browsers: Firefox 40, Chrome 44 or higher.
- Database: SQL 2010 or higher
- .NET Framework 4.5

### 6.2 Installation Guideline

#### 6.2.1 Connecting an ASP.NET MVC Web App with SQL Azure

##### 6.2.1.1 Creating the SQL Azure Database

The first thing to do is creat your sql database on azure:

- Step 1: Go to your azure portal, and click the new button in the bottom left corner

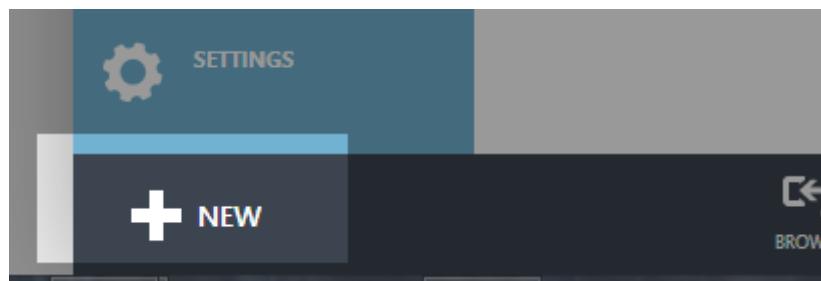
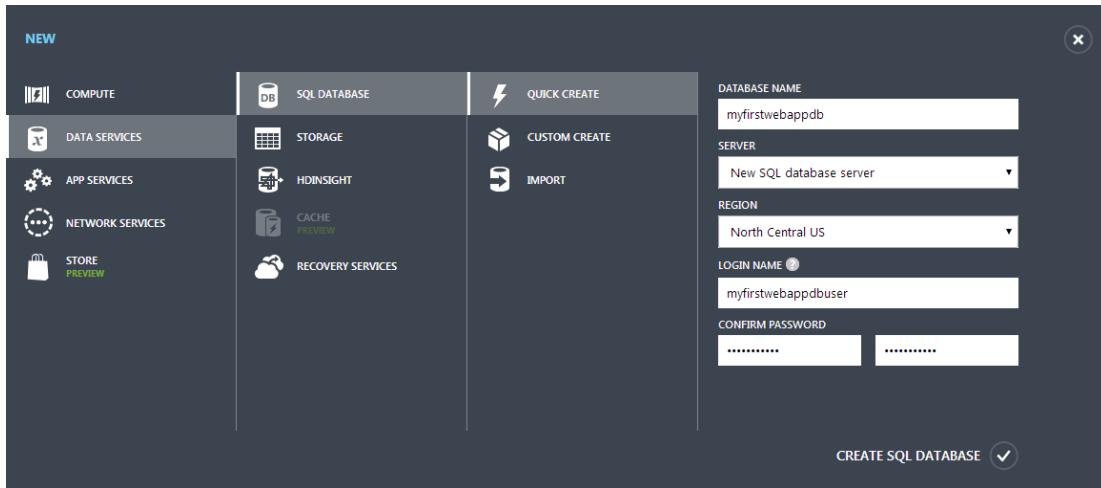


Figure 6-1: New button

- Step 2: Then choose Data Services >> SQL Database >> Quick Create and fill in the form. My settings were:
  - Database Name: **DDL\_CapstoneProject**
  - Server: **New SQL database server**
  - Region: **South East Asia**
  - **Make sure you remember the username and password you choose**

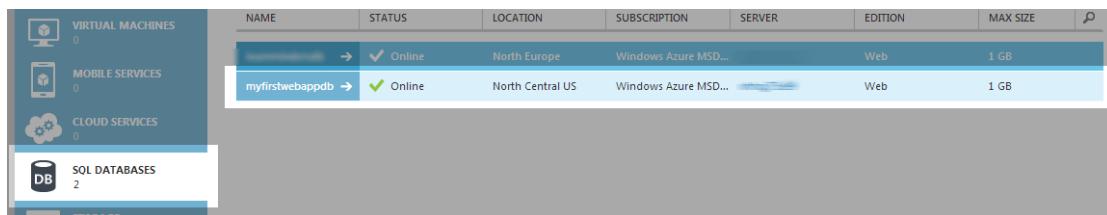
- Step 3: Press [Create SQL Database] button



**Figure 6-2: Create SQL Database**

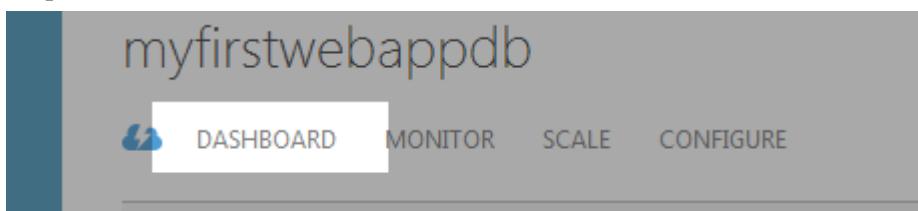
After waiting a few seconds your database should be created.

- Step 4: Click SQL Databases on the left column, then select your database in the main pane



**Figure 6-3: List SQL Database**

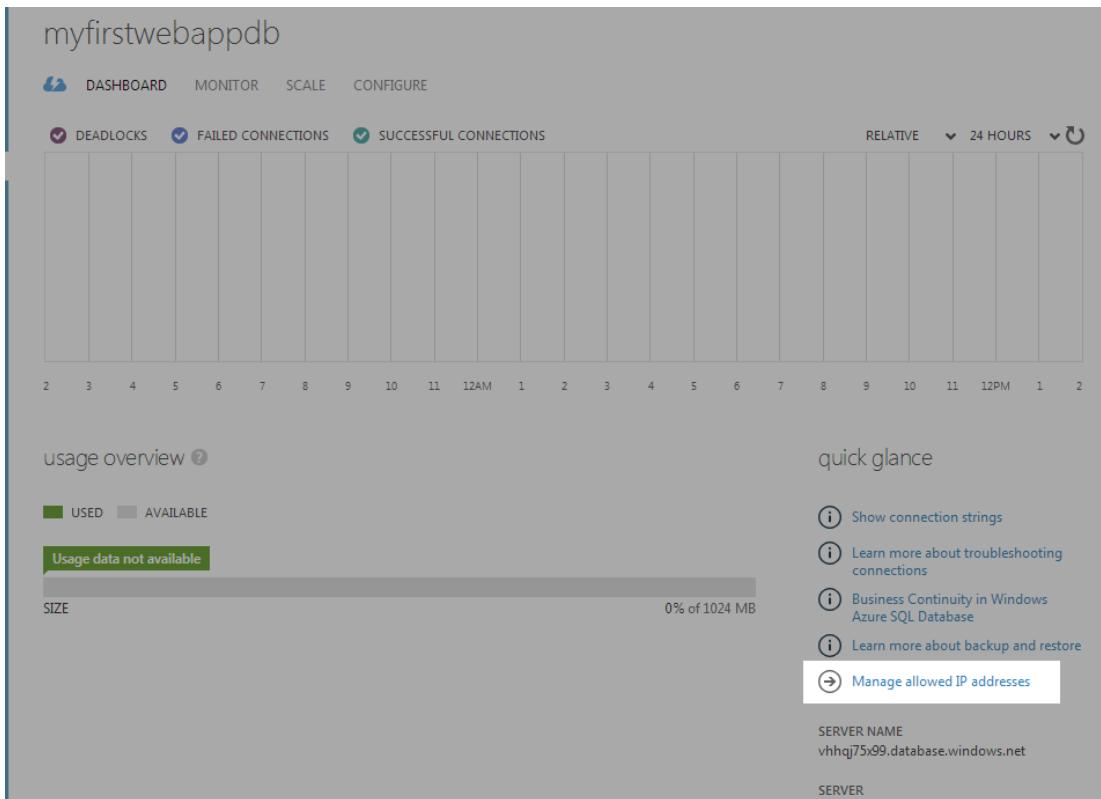
- Step 5: Go to the database dashboard



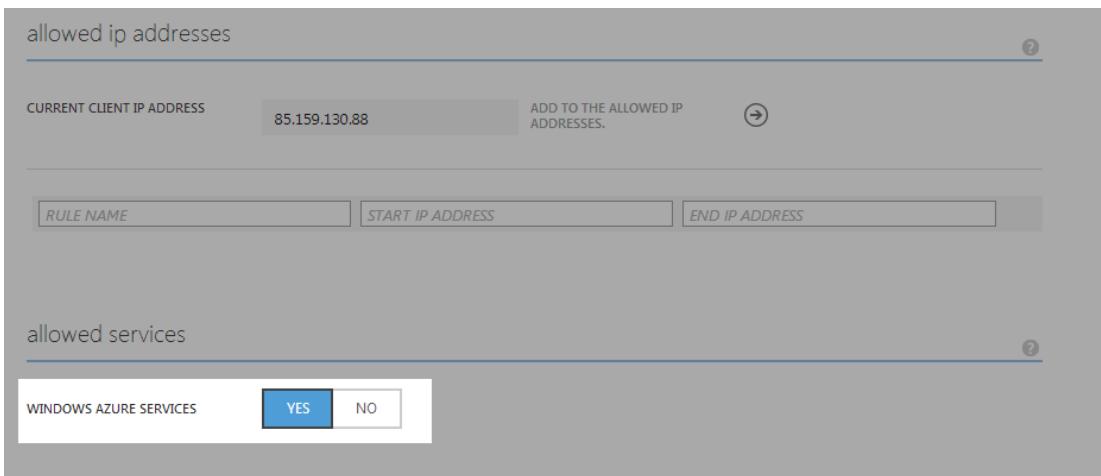
**Figure 6-4: Menu**

Once on the dashboard you need to ensure that other azure services are allowed to communicate with your database (this should be set by default). If this option is not selected the firewall on the sql azure database will disallow connections from your website:

- Step 6: Select 'Manage allowed IP addresses'

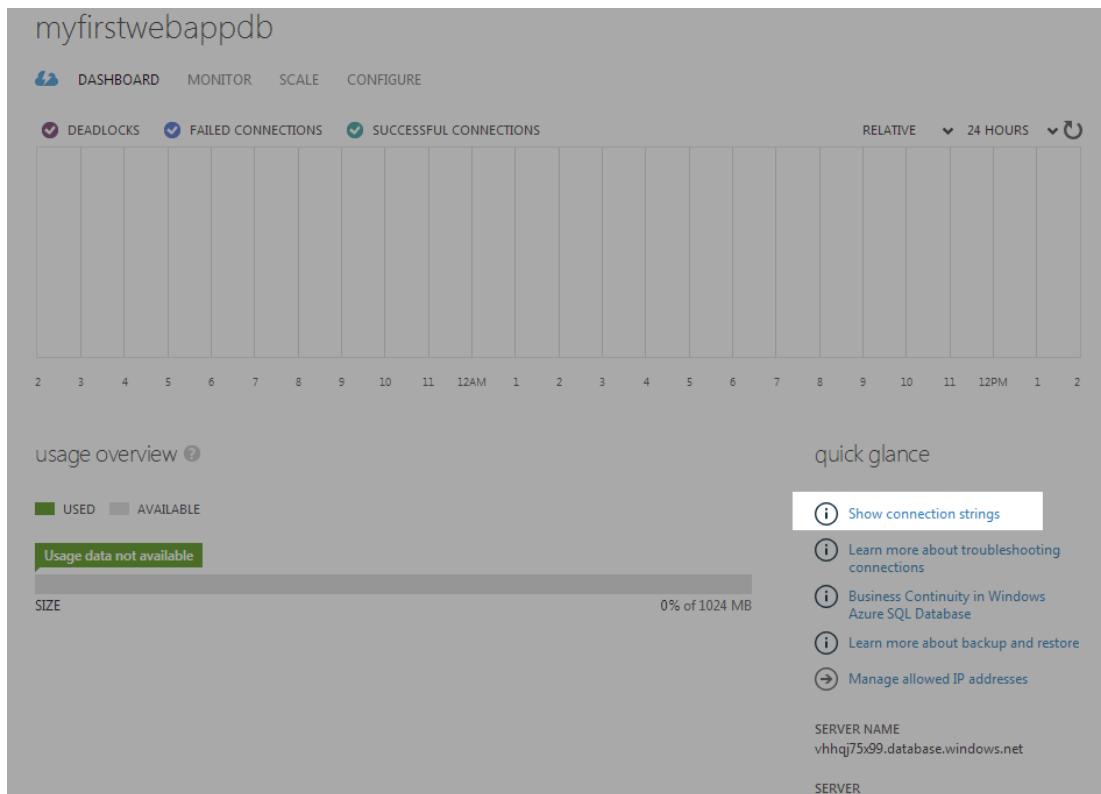
**Figure 6-5: Dashboard**

- Step 7: Ensure that ‘Windows Azure Services’ under ‘allowed services’ is set to Yes

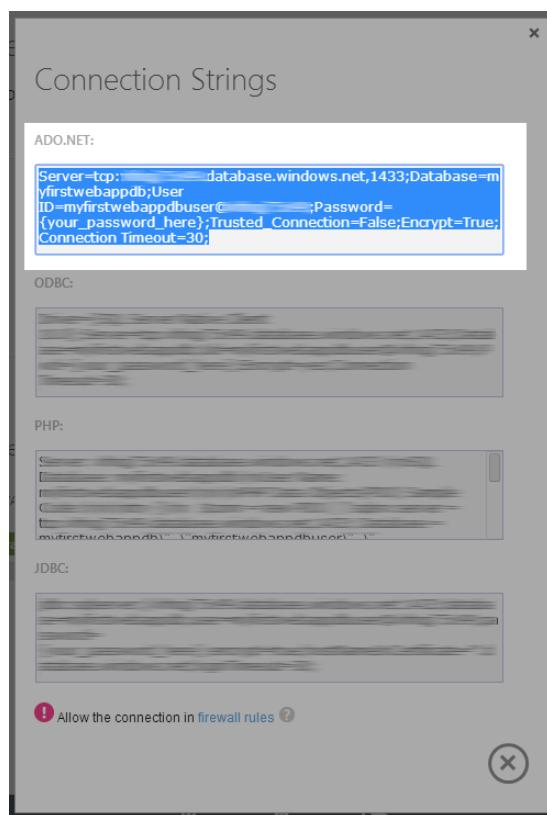
**Figure 6-6: Managed allowed IP address**

Finally you need to get the connection string for the database:

- Step 8: From the database dashboard (see Figure 6-4), select “Show connection strings”

**Figure 6-7: Dashboard**

- Step 9: On the window that pops up copy the connection string labelled ‘ADO.NET’ for later use:



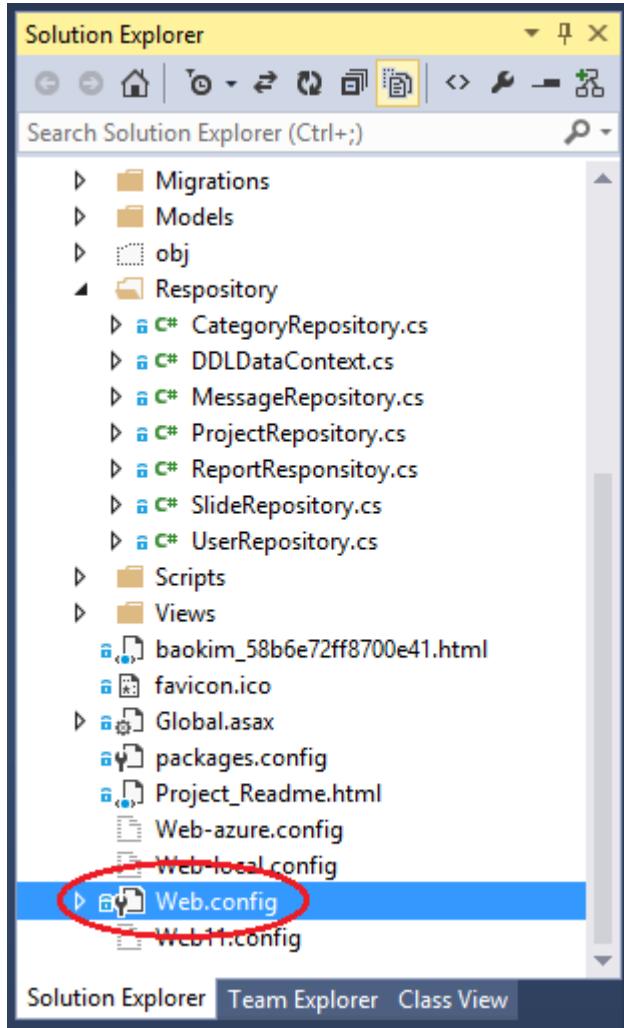
**Figure 6-8: Connection Strings**

Your database is now set-up and ready for you to configure in your web app.

### 6.2.1.2 Updating Connection Details in Your Web App

The next step is to update your mvc application to point at the newly created sql azure database.

- Step 1: Within your visual studio project, find the file called Web.config and open it

**Figure 6-9: Project Solution Explorer**

- Step 2: Add the following code into the file directly.

```
<connectionStrings>
  <add name="DDLDataContext"
    connectionString="*enter your connection string here*"
    providerName="System.Data.SqlClient" />
</connectionStrings>
```

- Step 3: Ensure that you replace 'enter your connection string here' with the one copied from sql azure earlier

- Step 4: There should also be a placeholder inside the connection string that read '{your\_password\_here}' – this should be replaced with the password you set earlier.
- Step 5: Save the file

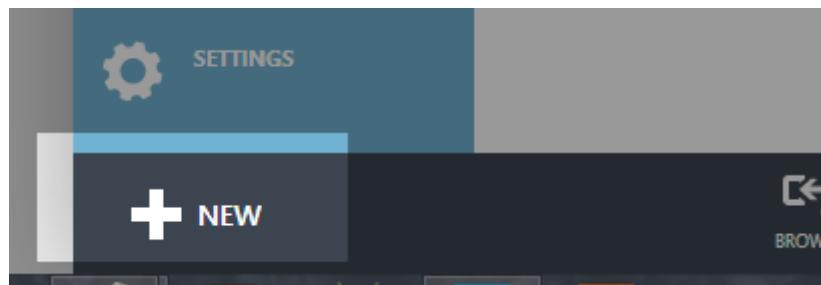
## 6.2.2 Create & Deploying the Azure Website

### 6.2.2.1 Create Azure Website

In order to deploy your website to windows azure you're going to need to set-up an account. At [Windows Azure website](#) you can create an account for free and have a free trial. Furthermore, at this point in time you can set-up up to 10 azure websites for free – you cannot use custom domain names, ssl or several other features, but they are perfect for experimentation.

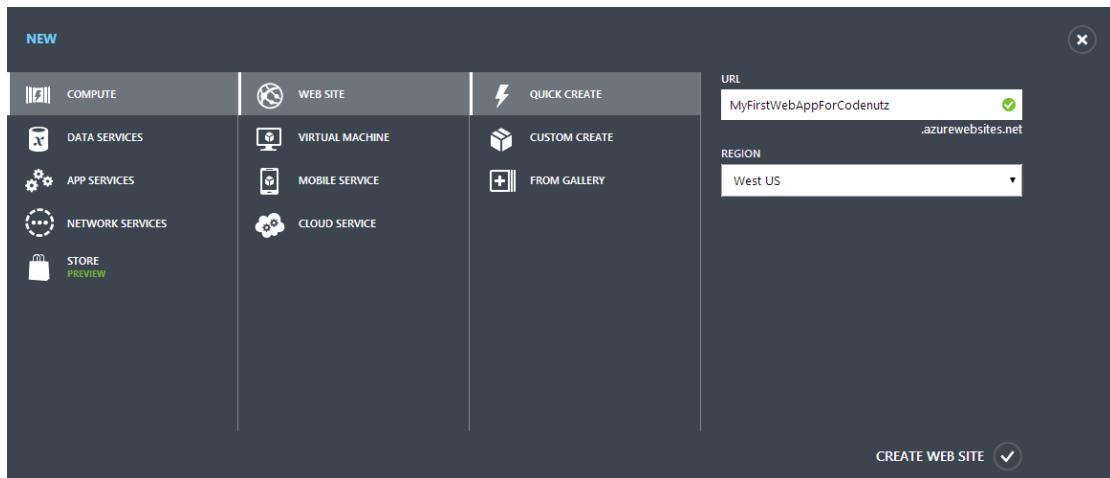
Once you have created your account head to <http://manage.windowsazure.com> where you can set-up your first website:

- Step 1: From the main screen click the new button in the bottom left corner



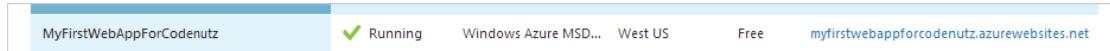
**Figure 6-10:** New Button

- Step 2: Select Compute → Web Site → Quick Create.
- Step 3: Enter the name for your website, I used “[dandeliondemo](#)” (note that you have to choose a name that's available).
- Step 4: Select a region where you want your site to be hosted, I suggest choose South East Asia
- Step 5: Press create website



**Figure 6-11:** Create Web Site

You'll have to wait a few seconds for azure to spin up the website, but you will be able to see in the interface a green tick denoting that it is ready to use:



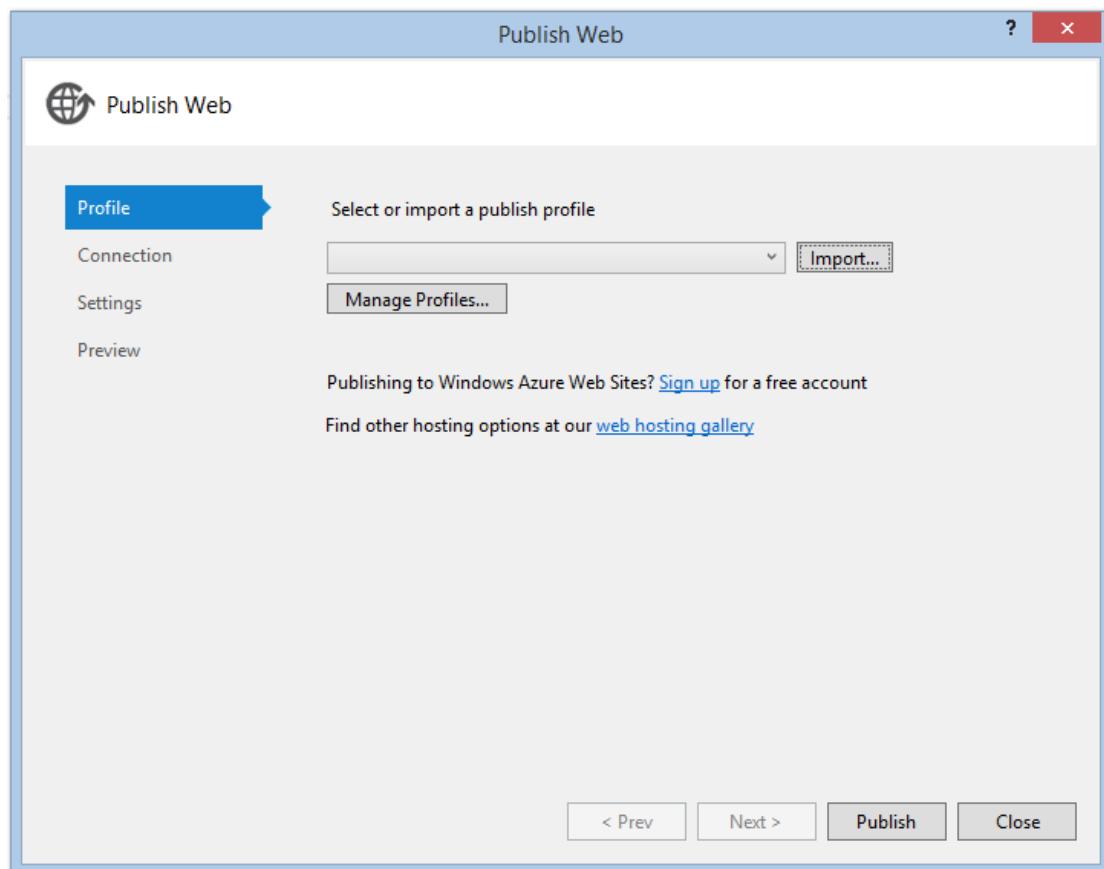
**Figure 6-12:** List Website

- Step 6: Once it is ready click on it, and select ‘Download publish profile’ – this should download a file to your computer called something like [Dandeliondemo.azurewebsites.net.PublishSetting](#)

Now you've got an azure website ready, we can get to publishing your mvc site to the azure service.

### 6.2.2.2 Deploying to the Azure Website

- Step 1: From visual studio, right click the website project and choose the ‘Publish...’ option. You should be presented with a publish dialog like this:

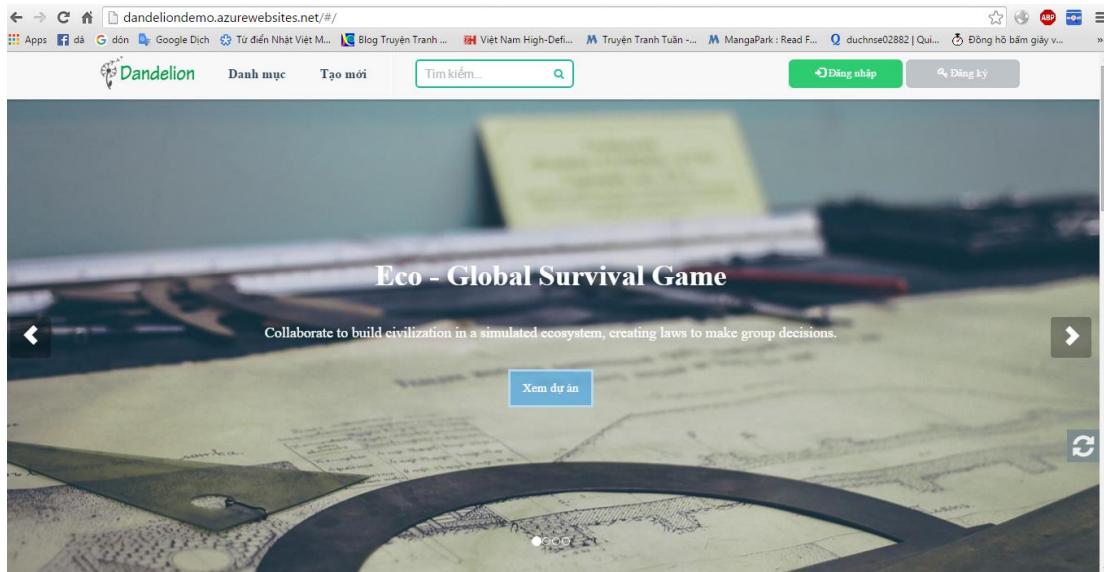


**Figure 6-13:** Publish Web dialog

- Step 2: Choose ‘Import...’ – on this screen you can choose to import the profile directly from azure, or from a profile file. Either is very simple, but as we downloaded this file we’ll choose ‘Import from a publish profile file’. Click the browse button, select the file you just downloaded and click Ok. ***This will prepopulate a number of fields in the publish dialog which for now we dont need to worry about.***

- Step 3: Finally just click the ‘Publish’ button, and your website will be deployed to windows azure. You should see the progress reported from within the visual studio output window with various bits of information like ‘Adding file...’ and ‘Adding ACLs’ etc. The first time you deploy tends to be the longest as all of the files have to be uploaded – subsequent deployments are much quicker as only the changed files are transferred.

Once your website has finished deploying, visual studio should automatically open it in your browser for you, which should look something like this:



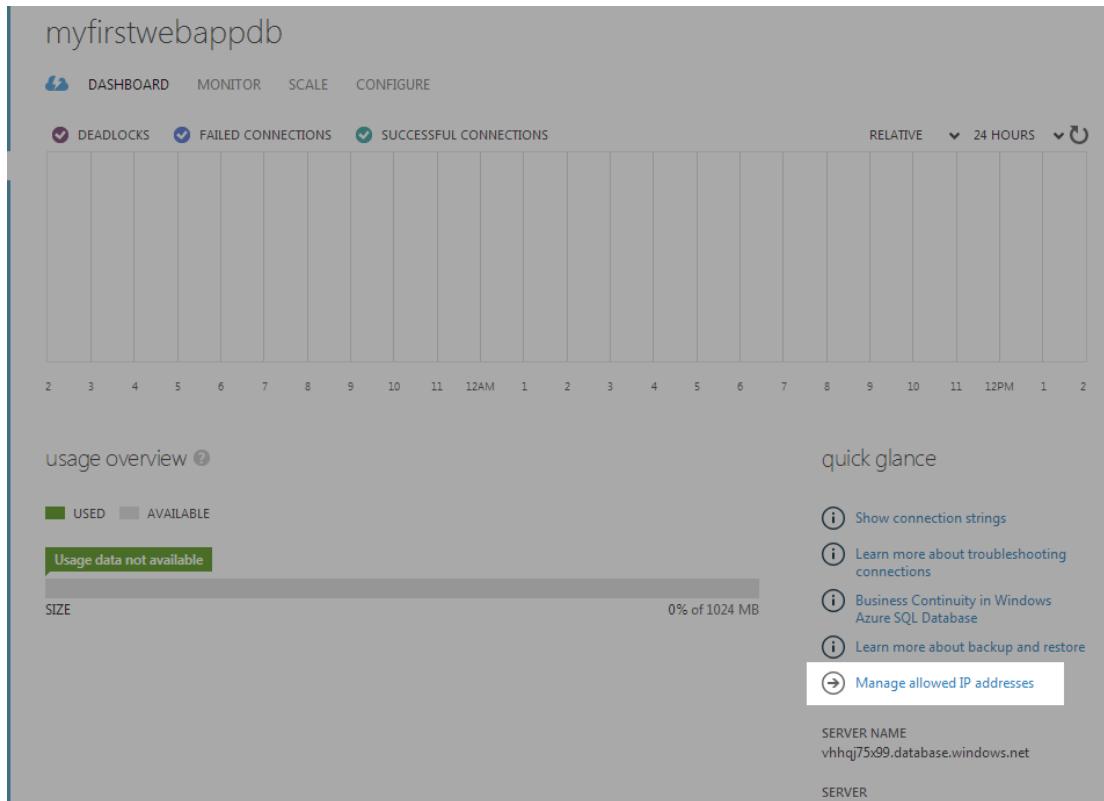
**Figure 6-14:** DDL screen after deploy successfully

### 6.2.3 Connecting SQL Management Studio to SQL Azure

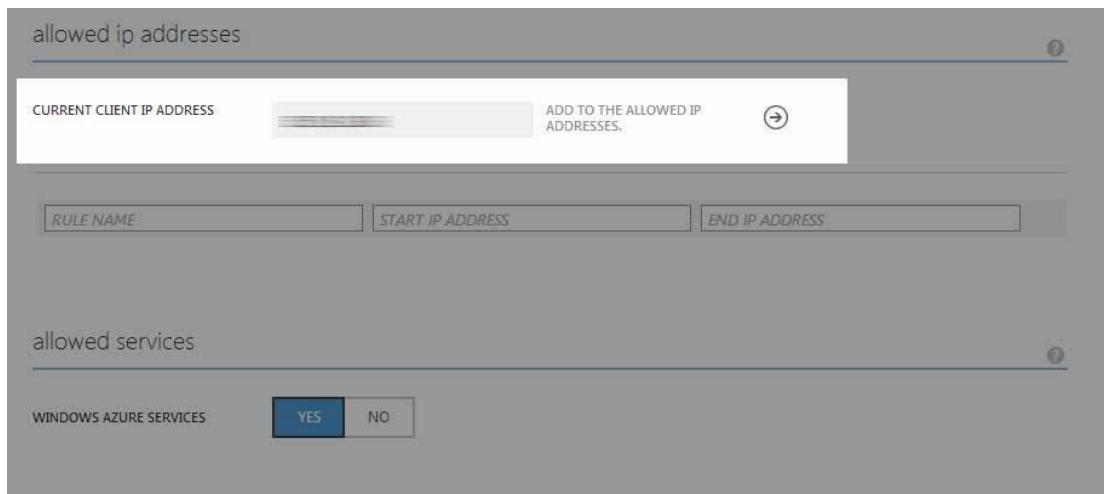
#### 6.2.3.1 Allowing access to SQL Azure

Here we’re going to set the SQL Azure firewall to allow connections from our local IP.

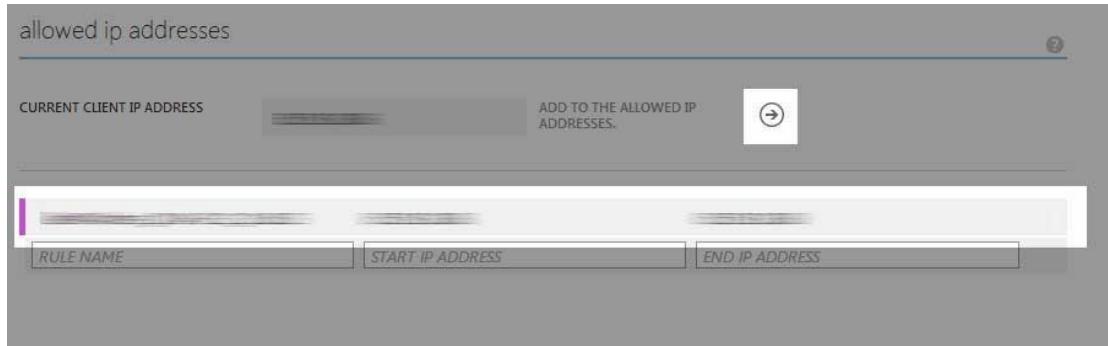
- Step 1: Navigate to your SQL Azure database dashboard (more detail instructions can be found in [Creating the SQL Azure Database](#)).
- Step 2: Next go to ‘Manage allowed IP addresses’

**Figure 6-15:** Dashboard

- Step 3: You should see on this screen that your IP address is listed as ‘Current client IP Address’. Here Azure has recognised your ip address and listed it for you so that you can easily add it to the firewall rules:

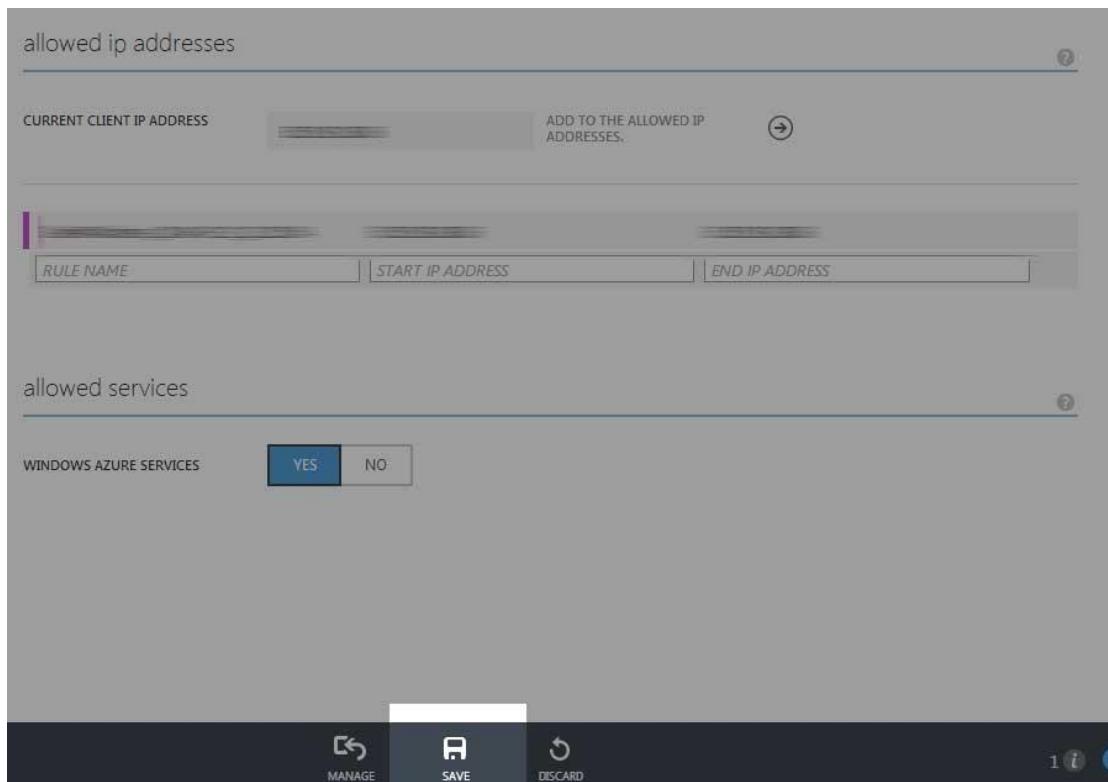
**Figure 6-16:** Manage allowed IP address

- Step 4: To add the firewall rule simple click right arrow and you should see the IP address move into the list of rules:



**Figure 6-17:** Manage allowed IP address

- Step 5: Finally click the Save button at the bottom of the page

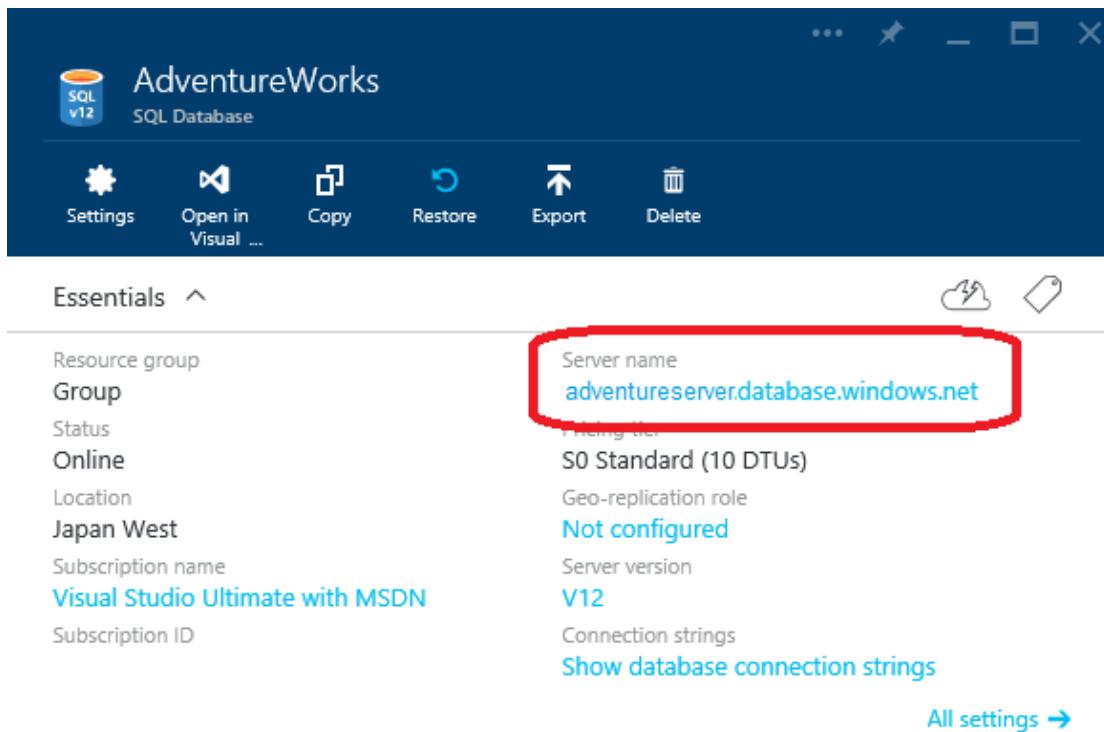


**Figure 6-18:** Save button

#### 6.2.3.2 Connecting with SQL Server Management Studio

To connect to your database you need the full name of the server (**servername.database.windows.net**) that contains the database you want to connect to:

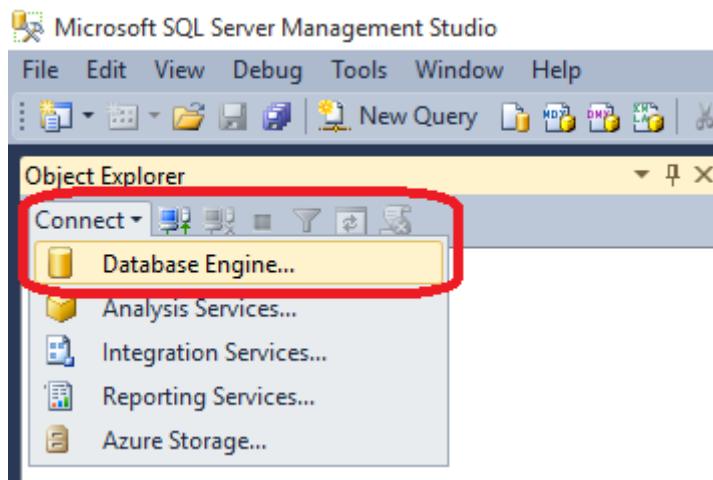
- Step 1: Go to the [Azure Portal](#)
- Step 2: Browse to the database you want to connect to.
- Step 3: Locate the full server name:



**Figure 6-19:** SQL database server name

Use the fully qualified server name in step 3 below to connect to SQL database.

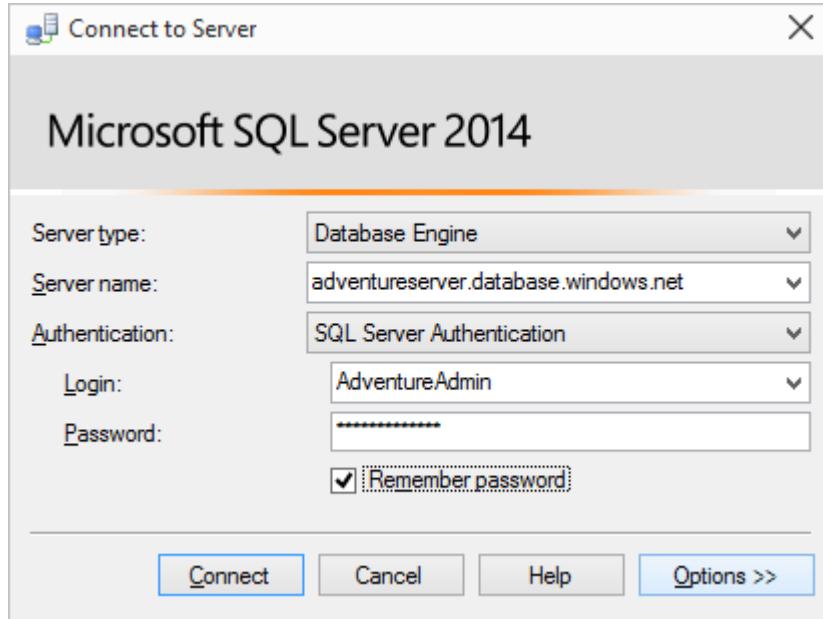
- Step 1: Open SSMS.
- Step 2: Click **Connect > Database Engine...**



**Figure 6-20:** SQL Server Management Studio

- Step 3: In the **Connect to Server** dialog box, in the **Server name** box, type the server name in the format <servername>.database.windows.net.
- Step 4: In the Authentication list, select SQL Server Authentication

- Step 5: Enter the **Login** and **Password** you specified when you created your SQL Database server, and click **Connect**.



**Figure 6-21:**      **Connect dialog**

If the connection fails: Make sure that the firewall of the logical server you have created allows connections from your local computer. For more information, see [How to: Configure Firewall Settings on SQL Database](#).

## 6.3 User Guideline

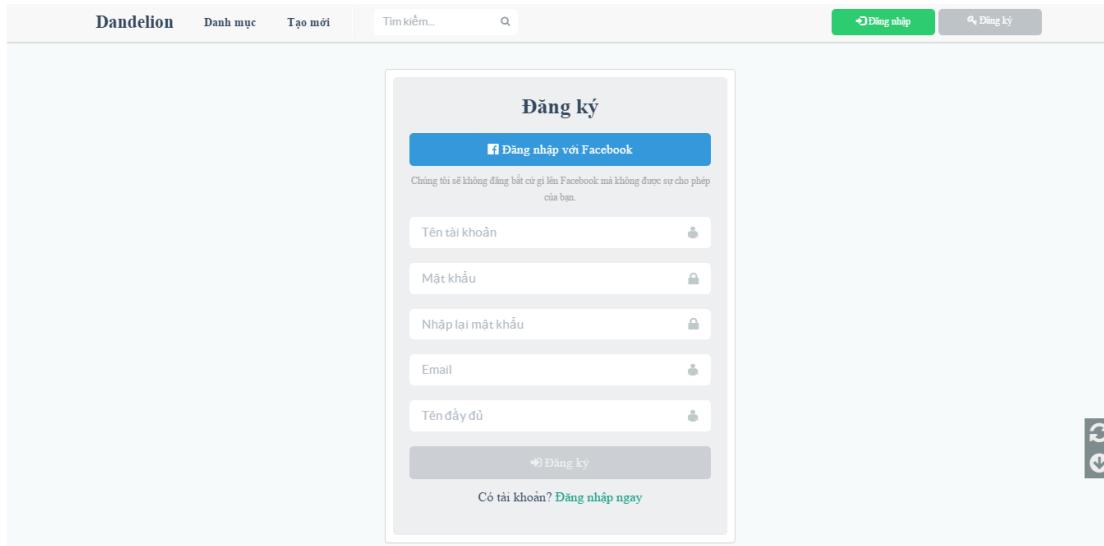
### 6.3.1 User Client

#### 6.3.1.1 Register

This is the first page which user will see. From this page the user can go to every page they want.

To register account on website, the users do the steps below:

- Step 1: Open browser and enter to the address bar: <http://dandelionvn.com/>. Homepage is displayed.
- Step 2: Click on [Đăng ký] button in header. Users will be gone to register page.
- Step 3: Enter information and click on [Đăng ký] button or User can register by choose Login with Facebook.

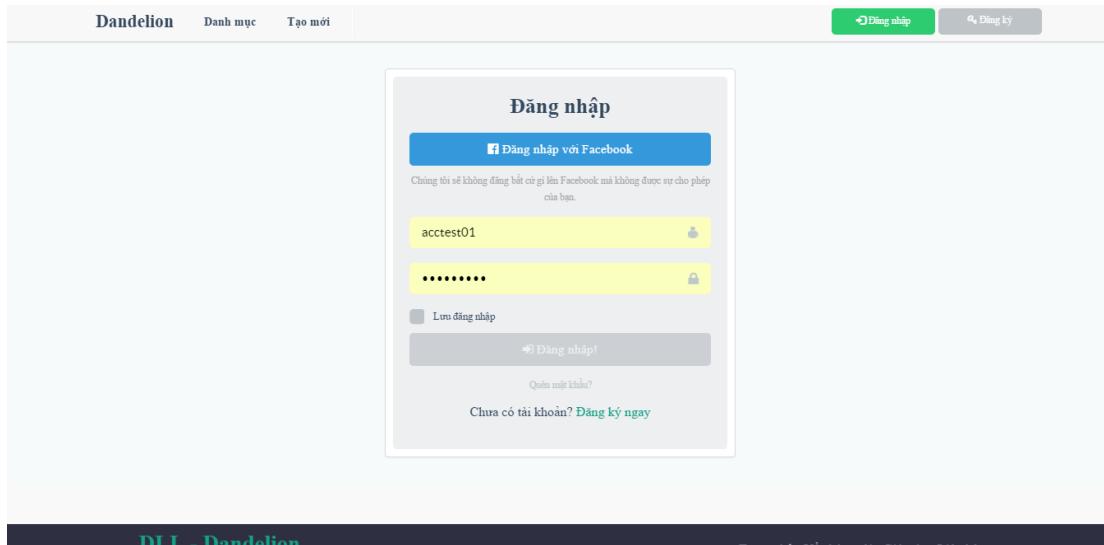


**Figure 6-22:** Register screen

### 6.3.1.2 Login

To login account on website, User do the steps below:

- Step 1: Open browser and enter to the address bar: <http://dandelionvn.com/>. Homepage is displayed.
- Step 2: Click on [Đăng nhập] button in header. User will go to Login page.
- Step 3: User can choose Login with Facebook.
- Step 3: User can choose Login with DDL's account:
  - Step 3.1: User enter username and password
  - Step 3.2: Click on Login button



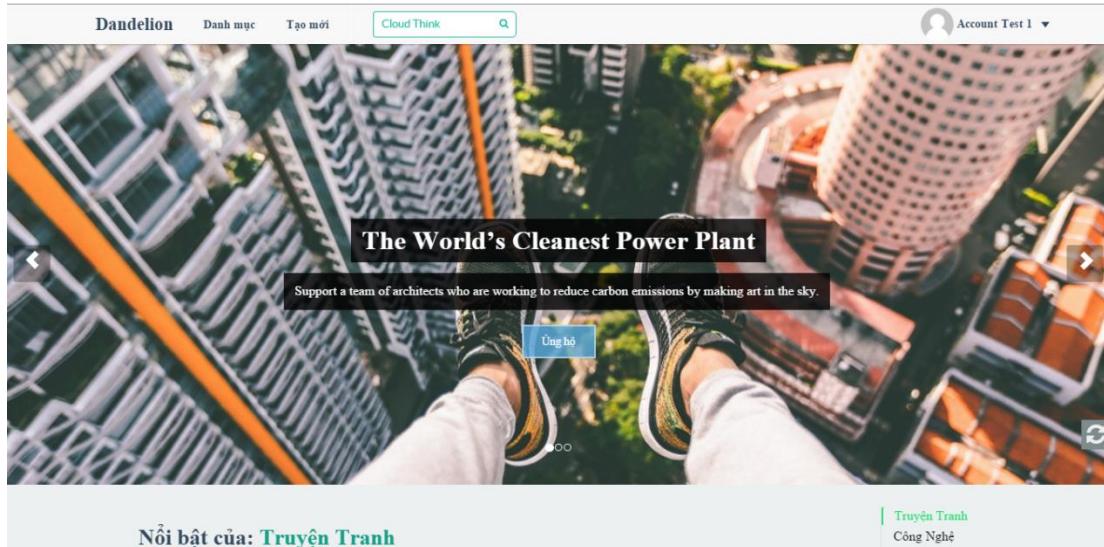
**Figure 6-23:** Login page

### 6.3.1.3 Search project

To use search function on website, the users do the steps below:

- Step 1: In Home page, enter project to search in header.

- Step 2: Click on search icon or press enter.



**Figure 6-24:** Search project

- Step3: Search result page is displayed.

**Figure 6-25:** Search project result

#### 6.3.1.4 Discover

In this page, user can view projects, view status of project and filter project by level, status... Go to Discover page, the user do the steps below:

- Step 1: Go to Homepage.
- Step 2: Click on [Danh Mục] hyperlink in header
- Step 3: Discover page is displayed and allow user can view and filter project.

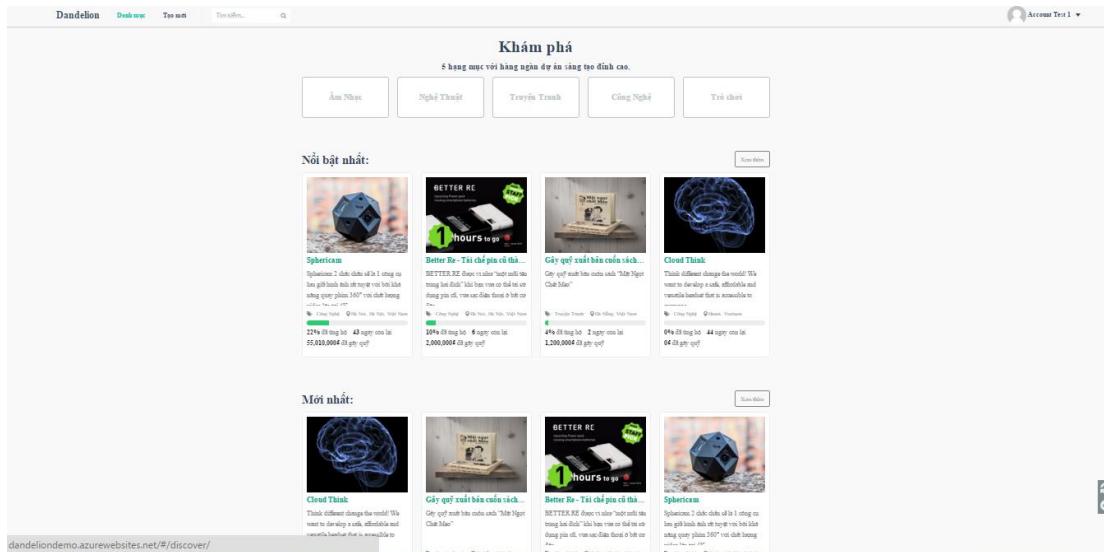


Figure 6-26: Discover

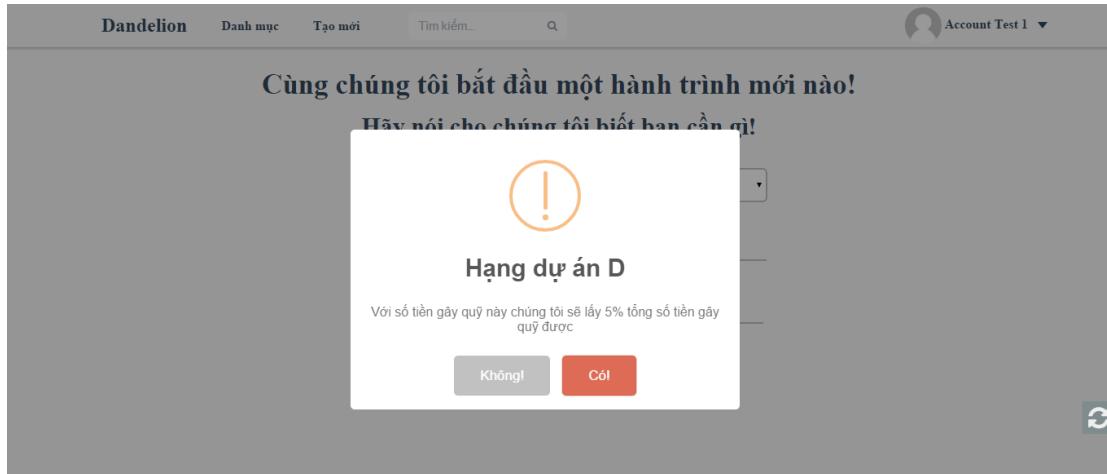
### 6.3.1.5 Create project

In this page, user view projects, view status of project and filter project by level, status... Go to Discover page, the user do the steps below:

- Step 1: Go to Homepage.
- Step 2: Click on [Tạo mới] hyperlnl in header. System will go to [Create Project] page.
- Step 3: User enter information
- Step 4: Click on [Bắt đầu] button.

Figure 6-27: Create project screen

- Step 5: Confirm form is display some of information of your project. Click [Có] button to create project, click [Không] to cancel.



**Figure 6-28:** Popup create project screen

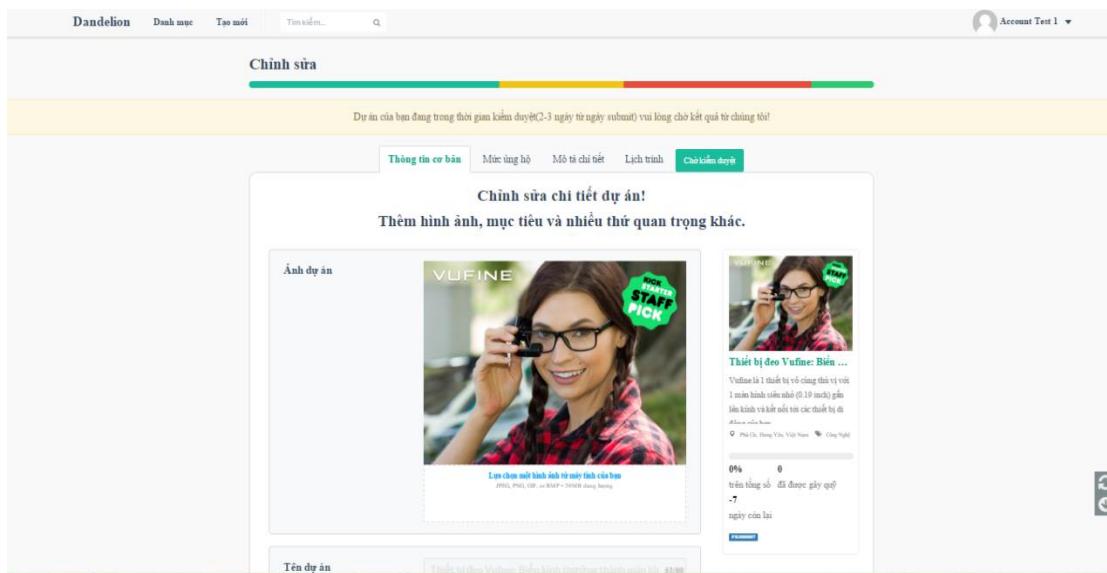
### 6.3.1.6 Edit project

When project is created user can edit project by the way go to edit project page.

- Step 1: Go to edit project page.
- Step 2: In this page, have tabs to user enter information
- Step 3: Click [Gửi dự án] button project to admin review and accept project.

#### 6.3.1.6.1 Basic tab

User choose image, enter name, short description, category, position, goal, time. Them save information.



**Figure 6-29:** Basic tab screen

#### 6.3.1.6.2 Reward tab

User create reward to give people donate project. At a reward user can edit, hide reward.

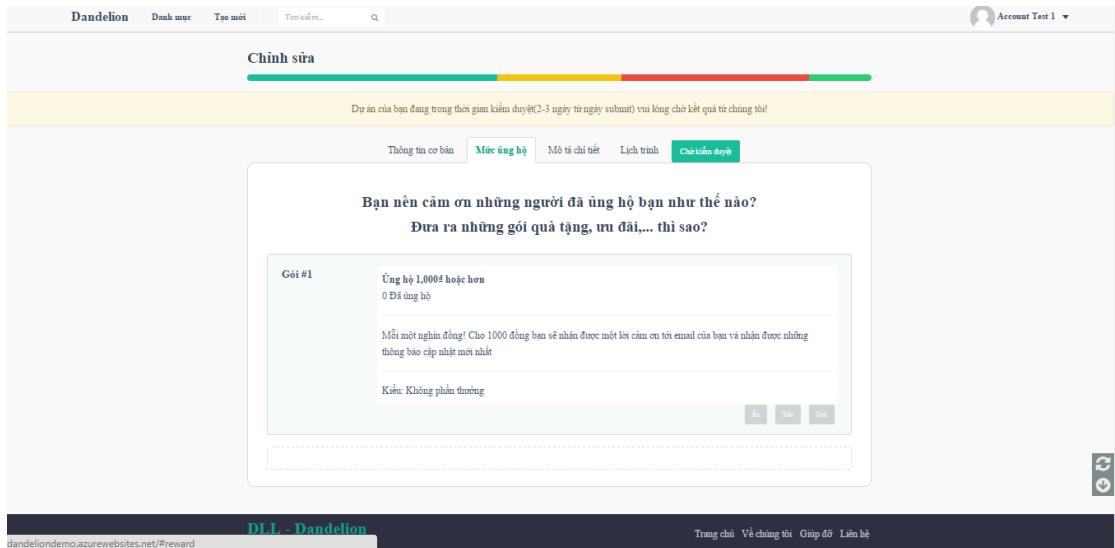


Figure 6-30: Reward tab screen

To create reward user do the step below:

- Step 1: Click on Add new reward.
- Step 2: Enter information reward
- Step 3: Click on Create button

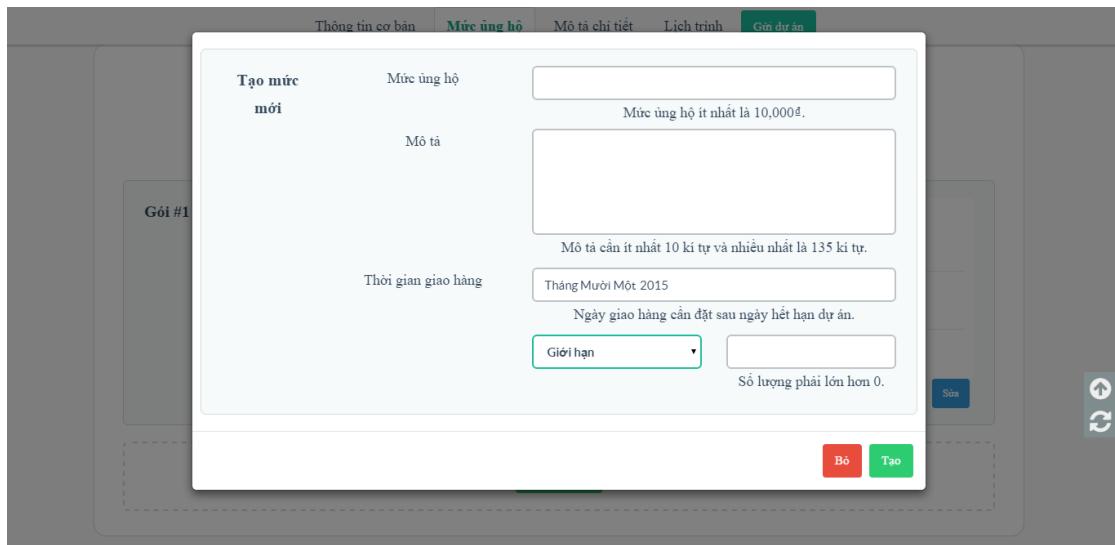


Figure 6-31: Reward tab screen

### 6.3.1.6.3 Story tab

In this tab, user fills information details about project.

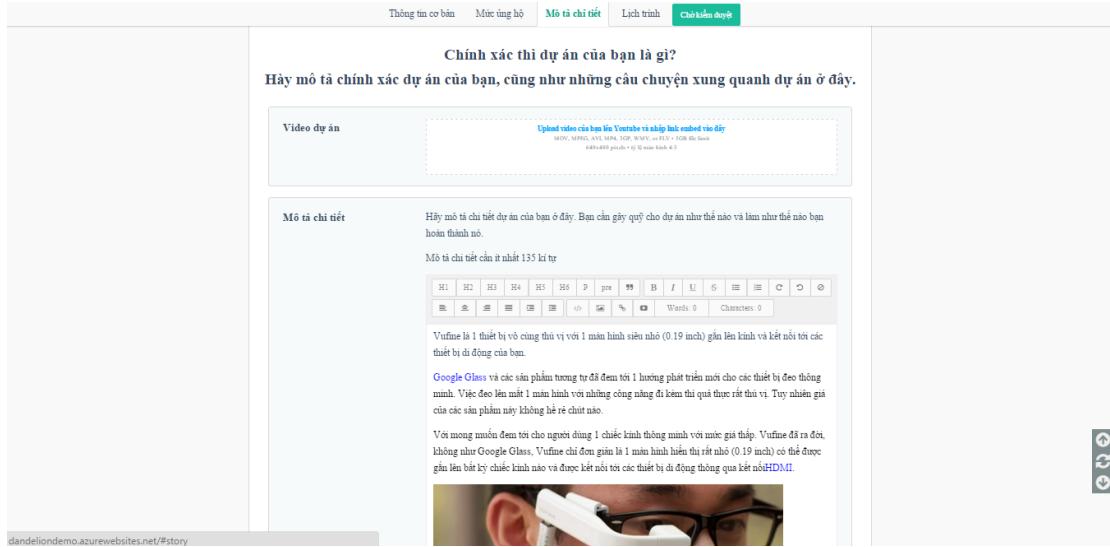


Figure 6-32: Story tab screen

### 6.3.1.6.4 Update tab

User create update version to people can follow status project.

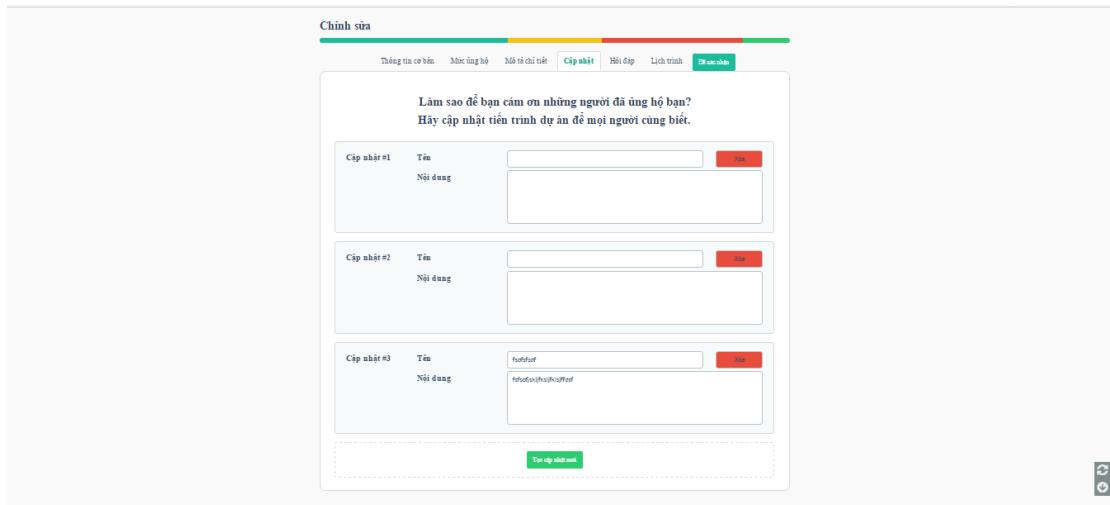


Figure 6-33: Update tab screen

To create update user do the step below:

- Step 1: Click on Add new update.
- Step 2: Enter information update
- Step 3: Click on Create button

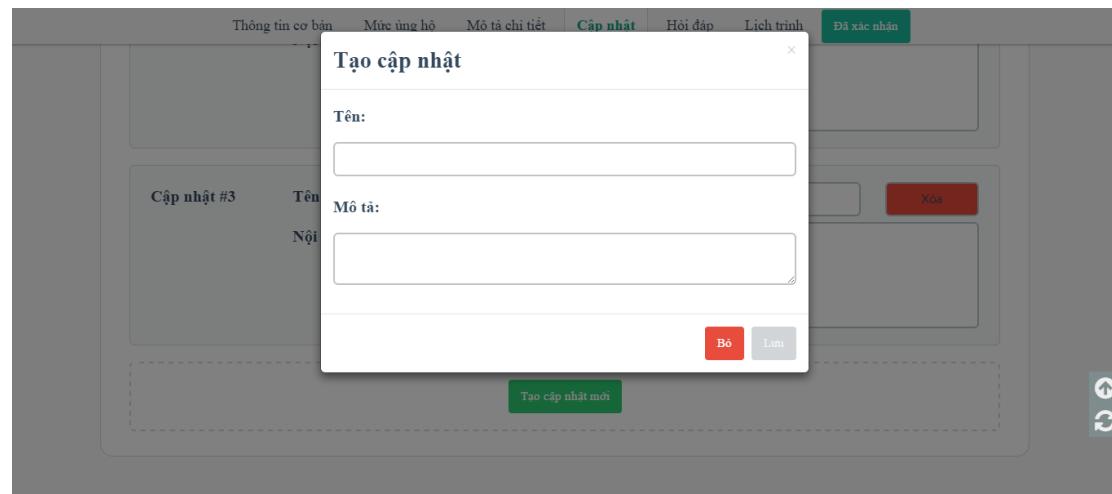


Figure 6-34: Create new update

### 6.3.1.6.5 Question and Answer tab

User create Q&A to answer the question about project.

Đặt những câu hỏi và trả lời  
Giải đáp những thắc mắc và biết thêm nhiều thông tin

#1	Câu hỏi	Đây là câu hỏi
	Trả lời	Đây là câu trả lời
		Xóa

Thêm hỏi đáp

Figure 6-35: Q&A tab screen

To create update user do the step below:

- Step 1: Click on Add new Q&A.
- Step 2: Enter question and answer.
- Step 3: Click on Create button

Thông tin cơ bản Mục ứng hò Mô tả chi tiết Cập nhật **Hỏi đáp** Lịch trình Đã xác nhận

Thêm mới hỏi đáp

#2	Câu hỏi	Đây là câu hỏi 3
	Trả lời	Đây là câu trả lời 3

BỎ LƯU

Thêm hỏi đáp

Figure 6-36: Create new Q&A

### 6.3.1.7 Project detail

In this page user can view details project. And user also can use share, follow, back, comment, report, send question to creator.

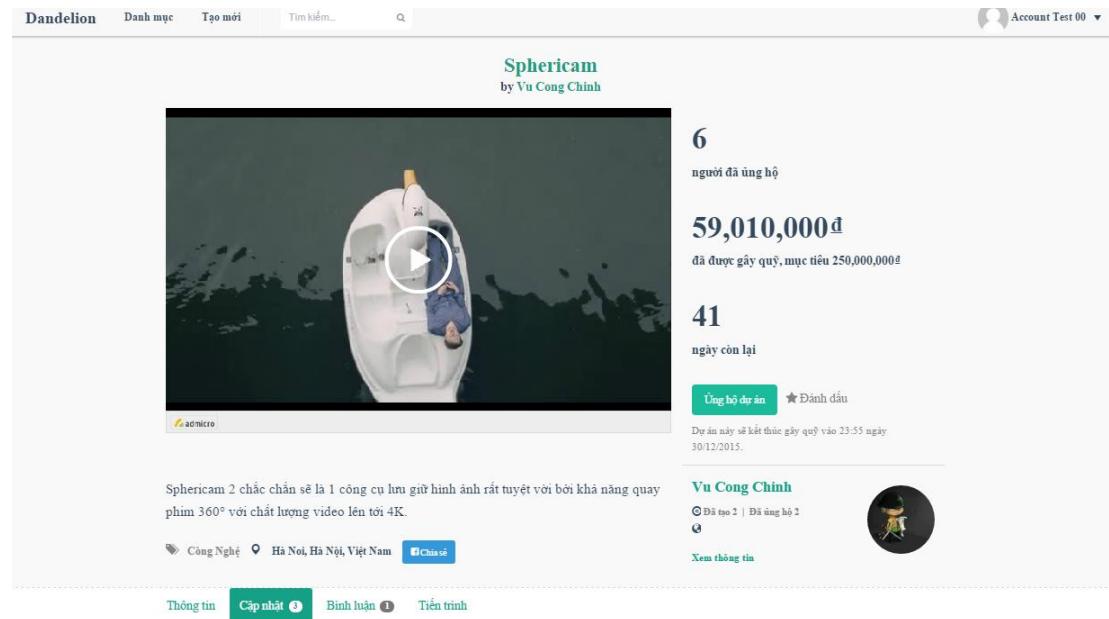


Figure 6-37: Project detail screen

### 6.3.1.7.1 Share project

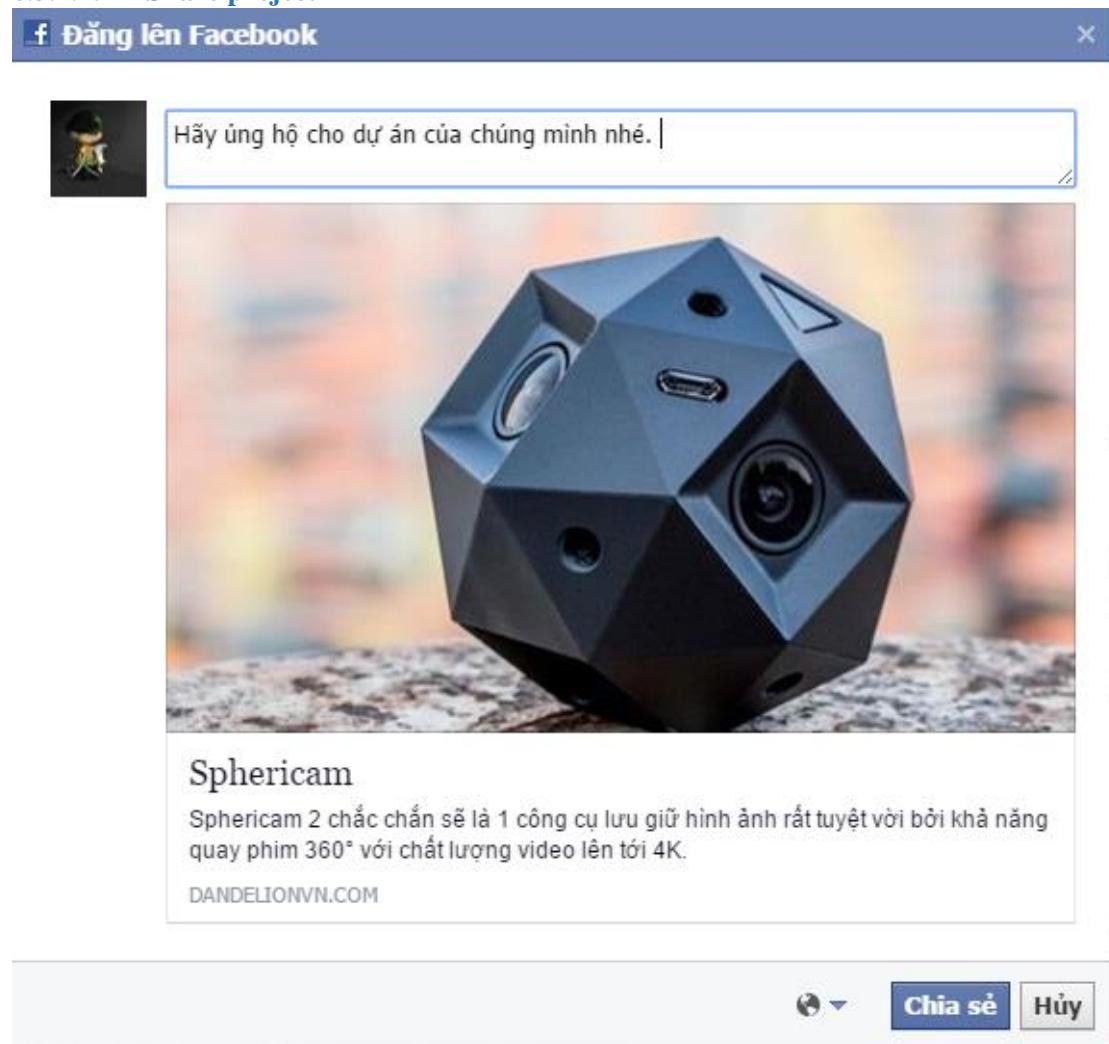
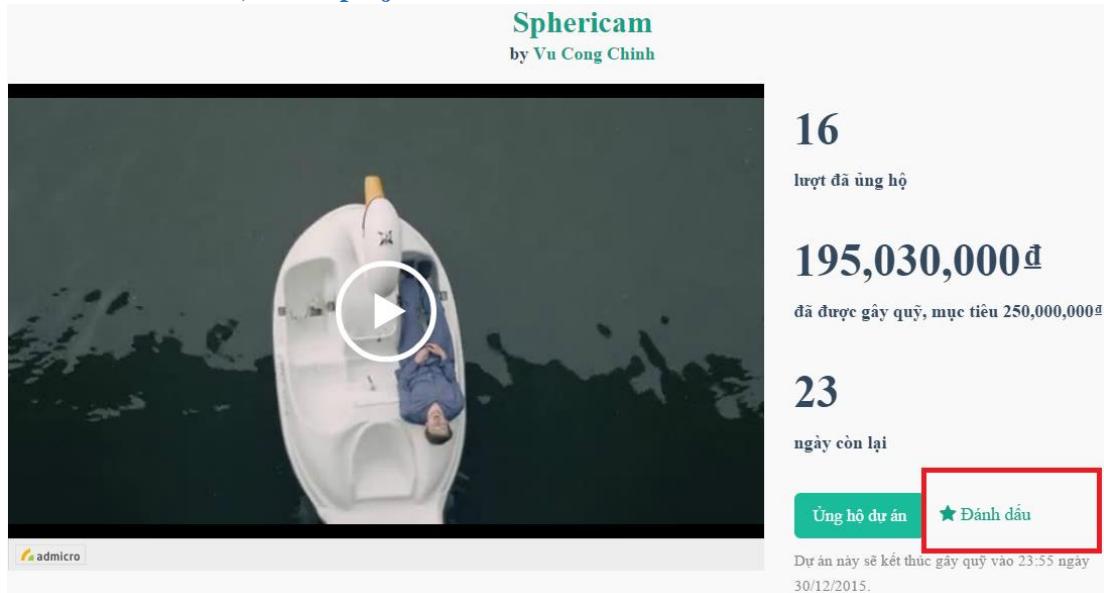


Figure 6-38: Share project

### 6.3.1.7.2 Remind, follow project

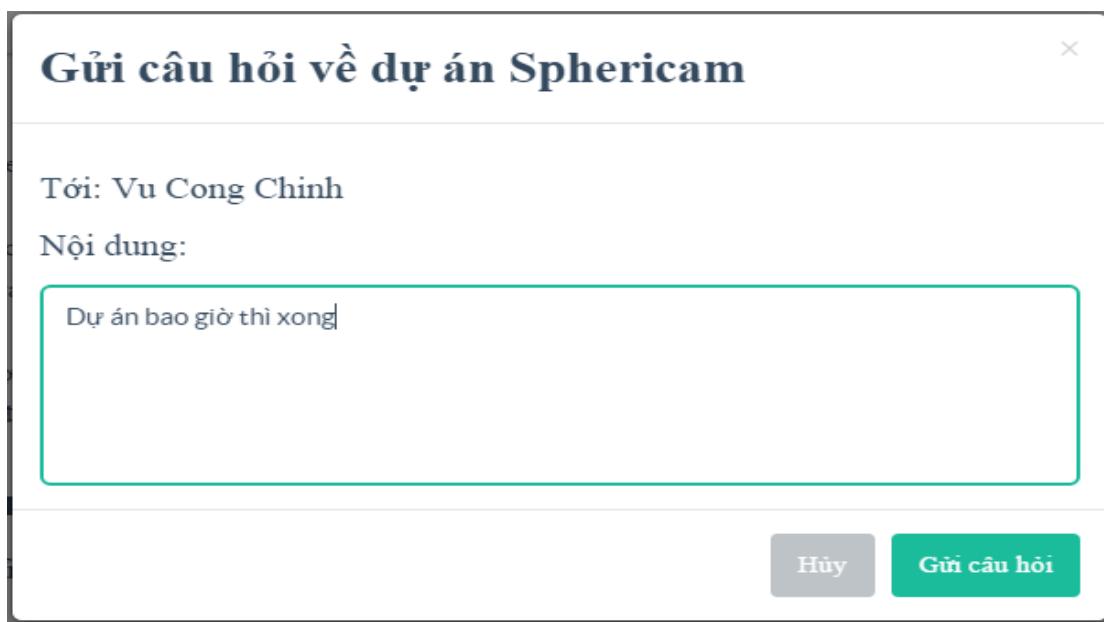


**Figure 6-39:** Follow/remind project

### 6.3.1.7.3 Send question about project

To send question about project user do the step below:

- Step 1: Go to project detail of a project
- Step 2: Go to Thông tin tab and click on Gửi câu hỏi button
- Step 3: Enter question and click on Gửi câu hỏi.



**Figure 6-40:** Send question

### 6.3.1.7.4 Report project

To send report about project user do the step below:

- Step 1: Go to project detail of a project

- Step 2: Go to Thông tin tab and click on Báo cáo sai phạm button
- Step 3: Enter content and click on Gửi báo cáo.



**Figure 6-41:** Sent report

#### 6.3.1.7.5 Comment tab

User can comment all of things relate about project

**Figure 6-42:** Comment tab screen

#### 6.3.1.8 Back project

This is the most importance feature. To use this fearture user do the steps below:

- Step 1: From project details page, click back this project button (“Üng hộ dự án”). System go to back project page.

- Step 2: Choose a reward, then fill pledge and click continue button. System go to payment project.

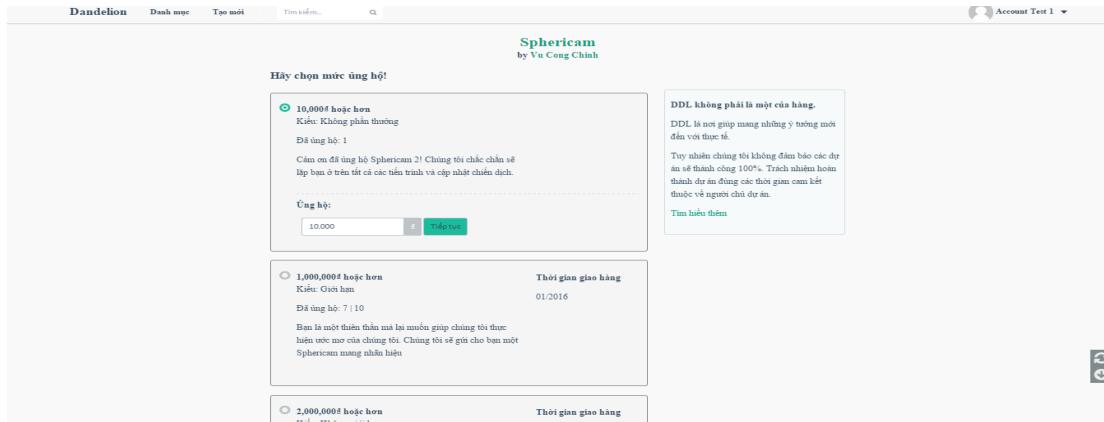


Figure 6-43: Back project screen

- Step 3: Enter information and click on 'Üng hộ' button. System go to the Bao Kim and backer complete some of step in Baokim to back success project.

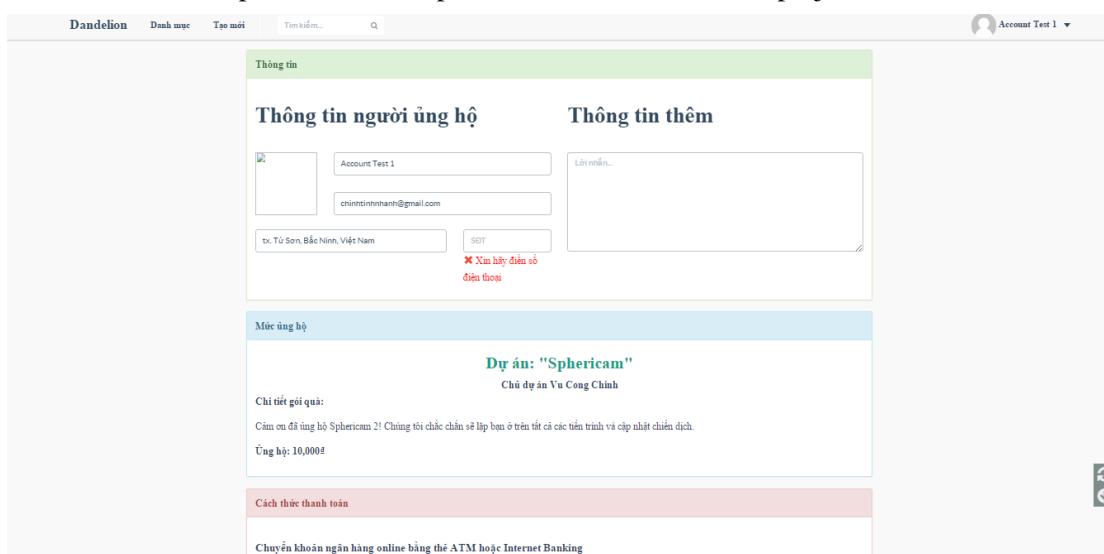


Figure 6-44: Payment project screen

### 6.3.1.9 Message

In this page user can view messages to create new message, delete message, view message conversation.

	Tên	Tiêu đề	Ngày
1	Từ: Account Test 00	test	19/11/15
2	Tới: Account Test 00	Gửi câu hỏi về dự án "Gây quỹ xuất bản cuốn sách "Mặt Ngọt hết Mèo""	18/11/15
3	Tới: Mạnh Nguyễn	Gửi câu hỏi về dự án "Cloud Think"	15/11/15
4	Tới: Vu Cong Chinh	Gửi câu hỏi về dự án "Sphericam"	15/11/15
5	Tới: Mai Thị Phương Cao	test	15/11/15
6	Từ: Vũ Công Văn	test	15/11/15

**Figure 6-45:** Message screen

### 6.3.1.9.1 Send message

To sent message user do the steps below:

- Step 1: Click on avatar in header, then click on [Tin nhắn] hyperlink. System will go to message page.
- Step 2: Click on [Soạn tin nhắn] button.
- Step 3: Enter information and click [Gửi] button.

**Figure 6-46:** Send message

### 6.3.1.9.2 Delete message

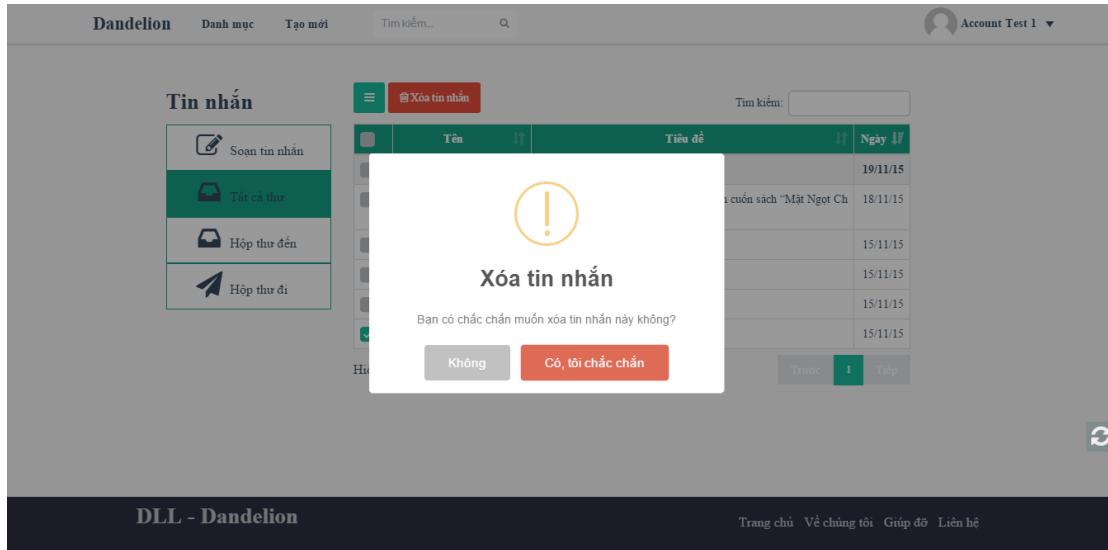
To delete message user do the steps below:

Step 1: Click on avatar in header, then click on [Tin nhắn] hyperlink. System go to message page.

Step 2: Select message want to delete.

Step 3: Click [Xóa tin nhắn] button, confirm delete

Step 4: Click [Có, tôi chắc chắn] to delete messages, [Không] to cancel.

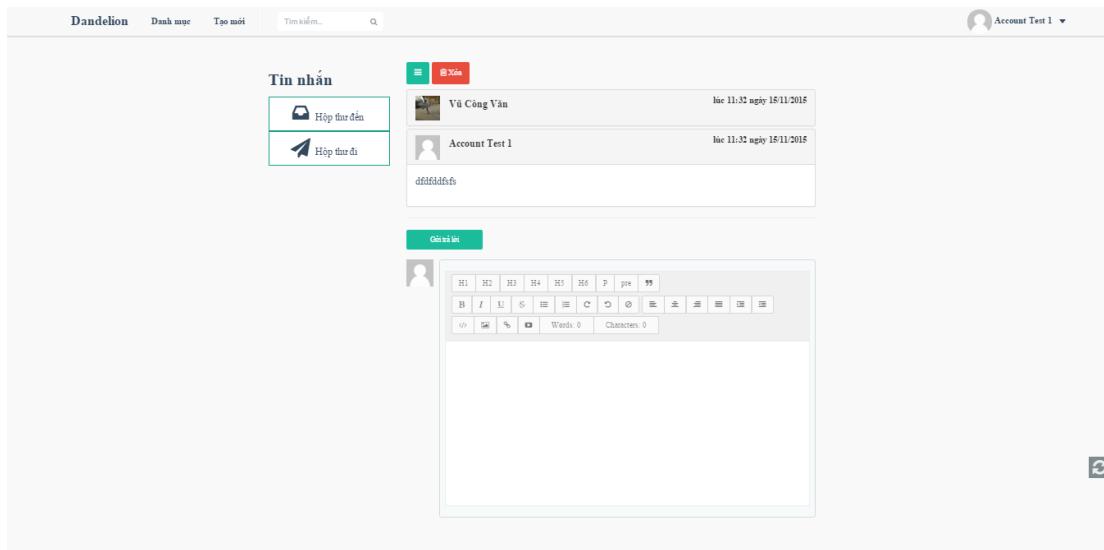


**Figure 6-47: Delete message screen**

### 6.3.1.9.3 Message view

In this page, user can sent and receive message create a conversation. To view conversation user do the steps below:

- Step 1: Click on avatar in header, then click on Message “Tin nhắn” button. System go to message page.
- Step 2: Click on a message to view conversation.



**Figure 6-48: Message view screen**

### 6.3.1.10 Project management

In this page help user manage project created, backed, reminded.

#### 6.3.1.10.1 View created project

To view project created user do the steps below:

- Step 1: Click on avatar in header, then click on [Dự án đã tạo] hyperlink. System will go to created project page.
- Step 2: User can delete draft project.
- Step 2.1: Or User can edit draft project, running project.
- Step 2.2: Or User can view list backer of running project.

**Dự án đã tạo:**

**Dự án nháp:**

- Gây quỹ xuất bản Tuyệt Đinh...  
Sửa Xóa Đang chờ kiểm duyệt  
Tuyệt Đinh Sinh Vật là bút danh của một nữ họa sĩ trẻ của Việt Nam, rất nổi tiếng trong cộng đồng mang với những bức tranh minh họa  
Thúy Linh | Yên Hội, Hà Nội, Việt Nam  
0% đã ủng hộ 0đ đã gây quỹ
- Bộ móng Người Sói Wolverine...  
Sửa Xóa Đang chờ kiểm duyệt  
Bộ móng hoạt động dựa trên cảm biến phản ứng cơ bắp linh hoạt, bạn chỉ cần gõ hoặc thả lồng tay để điều khiển bộ móng Người Sói  
Công Nghệ | Quảng Vũ, Hà Nội, Việt Nam  
0% đã ủng hộ 0đ đã gây quỹ
- DIY Apple Watch thành phi...  
Sửa Xóa Đang chờ kiểm duyệt  
24K mạ vàng thật của thép không gỉ các thiết bị, đồng hồ, đồ trang sức. Để dàng chuyển đổi đồng hồ của bạn với kim mạ vàng  
Công Nghệ | Quảng Vũ, Hà Nội, Việt Nam  
0% đã ủng hộ 0đ đã gây quỹ
- Bàn cầu thông minh tích hợp ...  
Sửa Xóa Đang chờ kiểm duyệt  
Ghế nhà vệ sinh với một fan hâm mộ được xây dựng trong đó ngay lập tức bắt đầu loại bỏ mùi hôi thông qua một Ánh sáng  
Công Nghệ | Ô Chợ Dừa, Hà Nội, Việt N...  
0% đã ủng hộ 0đ đã gây quỹ

**Figure 6-49:****Created project screen**

### 6.3.1.10.2 View list backing

To view list backing do the steps below:

- Step 1: Click on avatar in header, then click on [Dự án đã tạo] button. System will go to created project page.
- Step 2: Click on [Danh sách ủng hộ] button of project. System will go to list backer page
- Step 3: Choose a backer and click view backing details.

**Danh sách người ủng hộ**

Số tiền đã được ủng hộ: 250.000.000đ | Mục tiêu: 250.000.000đ

Danh sách người ủng hộ | Xuất dữ liệu

STT	Tên người ủng hộ	Số tiền	Thời gian	Xem
1	Account Test 1	2.000.000đ	19/11/2015	[View]
2	Account Test 00	60.000.000đ	19/11/2015	[View]
3	Lưu Ngọc Mạnh	4.000.000đ	19/11/2015	[View]
4	Lưu Ngọc Mạnh	50.000.000đ	30/10/2015	[View]
5	Lưu Ngọc Mạnh	10.000đ	30/10/2015	[View]
6	Lưu Ngọc Mạnh	2.000.000đ	29/10/2015	[View]
7	Account Test 1	1.000.000đ	29/10/2015	[View]
8	Vũ Công Chính	2.000.000đ	29/10/2015	[View]

**Figure 6-50:** List backer screen**6.3.1.10.3 View backed project list**

In this page help user manage project backed. To view project backed user do the steps below:

**Dự án đã ủng hộ:**

[Dự án đã ủng hộ](#) [Lịch sử ủng hộ](#)

Project Name	Description	Funding Status	Progress (%)	Days Left	Amount Raised
Sphericam	A camera that captures 360° video in 4K.	Đang chạy	24%	41 days left	59,010,000đ
Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	Financing the publication of the book "Mặt Ngọt Chết Mèo".	Đang chạy	237%	0 days left	71,200,000đ
Cloud Think	A safe, affordable, and versatile headset.	Đang chạy	0%	42 days left	100,000đ

**Figure 6-51:** Backed project screen

- Step 1: Click on avatar in header, then click on [Dự án ủng hộ] hyperlink. System will go to backed project page.
- Step 2: Click on [Lịch sử ủng hộ] hyperlink
- Step 3: Choose a project and click view details.

**Backed projects:**

[Dự án đã ủng hộ](#) [Lịch sử ủng hộ](#)

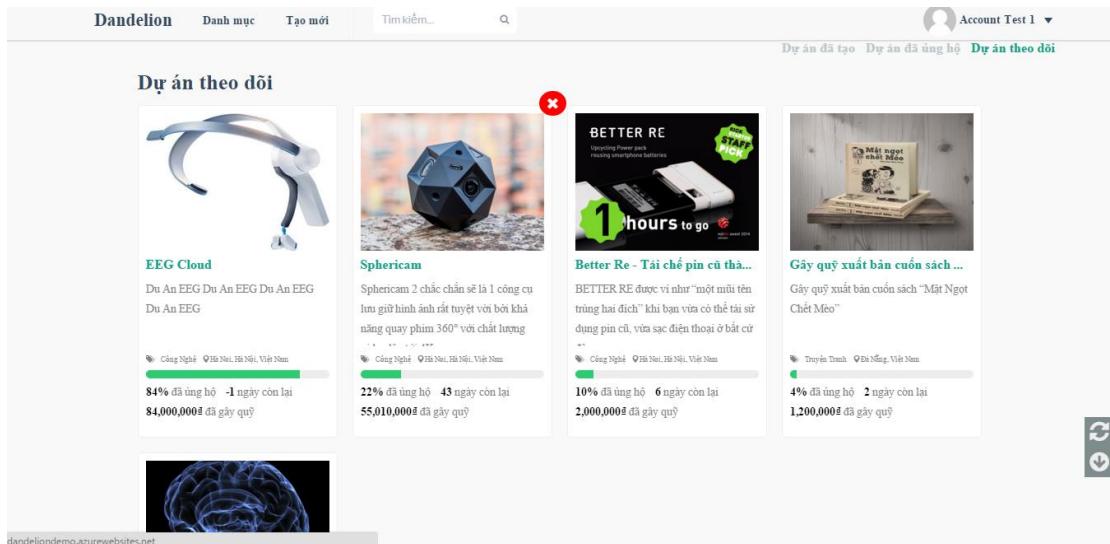
STT	Tên dự án	Số tiền	Thời gian	Tình trạng dự án	Xem
1	Sphericam	59,010,000đ	29/10/2015	Đang chạy	
2	Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	71,200,000đ	18/11/2015	Đang chạy	
3	Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	71,200,000đ	18/11/2015	Đang chạy	
4	Cloud Think	100,000đ	18/11/2015	Đang chạy	

**Figure 6-52:** Backed project history screen

### 6.3.1.10.4 View reminded project

To view project reminded user do the steps below:

- Step 1: Click on avatar in header, then click on [Dự án theo dõi] hyperlink. System will go to backed project page.
- Step 2: Click on “X” icon to cancel remind a project



**Figure 6-53: Starred project screen**

### 6.3.1.11 Account management

#### 6.3.1.11.1 Change password

To change password user do the steps below:

- Step 1: Click on avatar in header, then click on “Tài khoản” button. System will go to account project page.
- Step 2: Click on “Thay đổi mật khẩu” button
- Step 3: Enter current pass, new pass, confirm pass.
- Step 4: Click “Lưu thiết lập” button.

**Figure 6-54:** Account screen

#### 6.3.1.11.2 Edit profile

To change password user do the steps below:

- Step 1: Click on avatar in header, then click on “Tài khoản” button. System will go to account project page.
- Step 2: Click on “Sửa thông tin cá nhân” button
- Step 3: Enter information.
- Step 4: Click “Lưu thiết lập” button.

Thiết lập

Tài khoản [Sửa thông tin cá nhân](#) Thông tin công khai

Tên đầy đủ: Account Test 1

Giới tính: Nam

Email: osintintenmann@gmail.com

Ngày sinh: 10/01/1993

Địa chỉ: tp. Từ Sơn, Bắc Ninh, Việt Nam

Số điện thoại:

Website: www.abc

Lưu thiết lập

Figure 6-55: Edit profile screen

### 6.3.1.11.3 Public profile

User can view public profile.

Account Test 1

Đã tạo: 0 dự án

Đã ứng h借口: 1 lượt

Tham gia vào: 29/10/2015

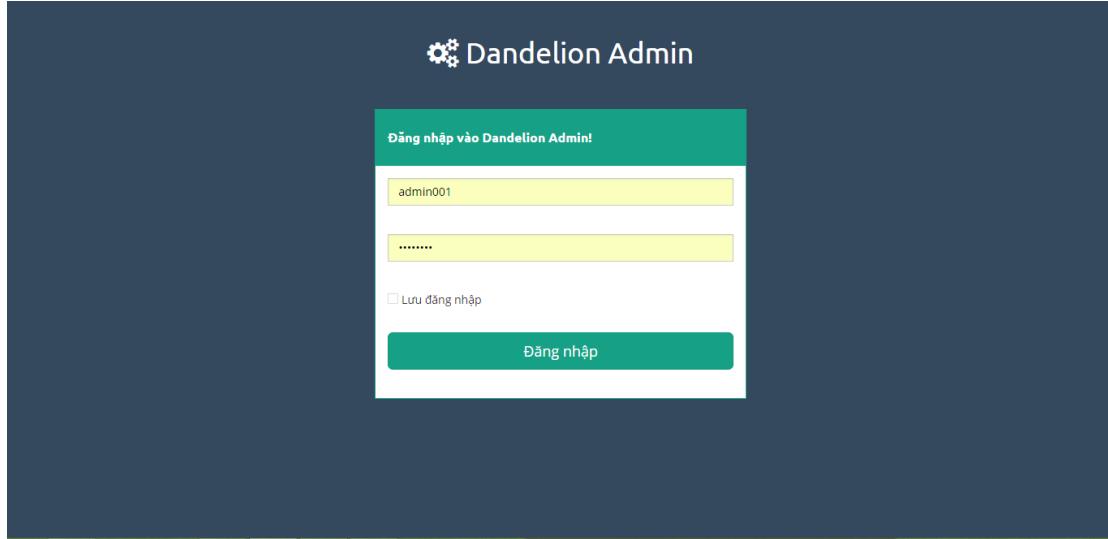
Mô tả thêm:

Gửi tin nhắn

Figure 6-56: Public profile screen

### 6.3.2 Admin

#### 6.3.2.1 Login



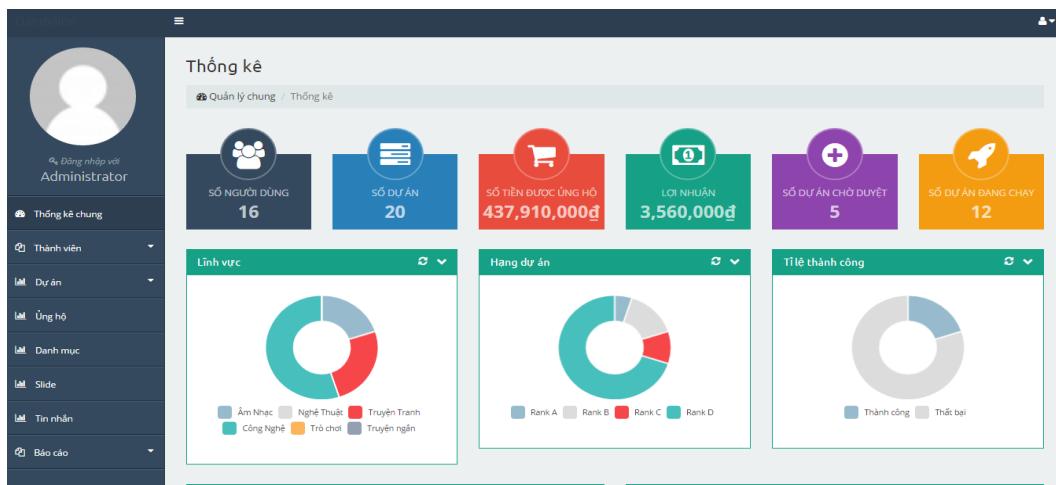
**Figure 6-57:** Login screen

To login admin account, the users do the steps below:

- Step 1: Open browser and enter to the address bar: <http://dandelionvn.com/admin/>.
- Step 2: User enter username and password
- Step 3: Click on Login button

#### 6.3.2.2 Dashboard

At this page, Admin can view overview about project, user, backing, category.



**Figure 6-58:** Dashboard 1

**Những người ủng hộ nhiều nhất:**

Tên	Số dự án	Số tiền(VND)	Trạng thái
Lưu Ngọc Mạnh	6	121,510,000	Bình thường
Vũ Công Chính	4	108,000,000	Bình thường
Account Test 1	5	102,100,000	Bình thường
Account Test 00	3	75,100,000	Bình thường
Phan San	1	21,000,000	Bình thường

**Những dự án thành công nhất:**

Dự án	Lĩnh vực	Chủ dự án	Mức tiêu(VND)	Quỹ(VND)	Trạng thái
Gây quỹ xuất bản cuốn sách "Mặt Nạ t Chết Mèo"	Truyện	maiictse0908@fp.edu.vn	30,000,000	71,200,000	Hết hạn

**Người dùng vừa đăng nhập:**

Tên	Đăng nhập cuối	Trạng thái
Vũ Công Chính	08:35 ngày 21-11-2015	Bình thường
Account Test 1	08:32 ngày 21-11-2015	Bình thường
Account Test 00	08:20 ngày 21-11-2015	Bình thường
Lưu Ngọc Mạnh	02:29 ngày 20-11-2015	Bình thường
Test Accountt	11:16 ngày 19-11-2015	Bình thường

**Lượt ủng hộ mới:**

Người ủng hộ	Dự án	Tiền ủng hộ(VND)	Thời gian
chinhvccse02585@fp.edu.vn	Cloud Think	100,000,000	08:38 ngày 21-11-2015
chinhvccse02585@fp.edu.vn	Better Re - Tái chế pin cũ thành pin dự phòng	5,000,000	08:36 ngày 21-11-2015
chinthinhnhanh@gmail.com	Better Re - Tái chế pin cũ thành pin dự phòng	5,000,000	08:34 ngày 21-11-2015
chinthinhnhanh@gmail.com	Sphericam	2,000,000	08:28 ngày 21-11-2015
chinthinhnhanh@gmail.com	Sphericam	20,000,000	08:28 ngày 21-11-2015

Figure 6-59: Dashboard 2

**Điều đồ thống kê dự án theo năm:**

Thống kê thông số dự án

Số tiền được quyên góp (VND)

Tháng

2015

Tạo Thành công Thất bại Ủng hộ Lợi nhuận

**Thống kê dự án:**

Thời gian	Số dự án tạo	Dự án thành công	Dự án thất bại	Dự án đang cheo	Tổng số đóng góp(VND)	Đóng góp thành công(VND)	Lợi nhuận(VND)	Tỷ lệ thành công
2015	20	1	4	10	437,910,000	71,200,000	3,560,000	5%

Figure 6-60: Dashboard 3

### 6.3.2.3 Manage User

To lock/unlock an account. Admin do the step below:

- Step 1: Go to Admin page, Click on [Danh sách thành viên]. System return list account.
- Step 2: Choose account and click on lock/unlock button to lock/unlock account.

The screenshot shows a user management interface. At the top, there's a dashboard with four colored boxes: orange (0 users), blue (3 users), purple (12 users), and dark blue (15 total users). Below this is a table titled "Danh sách Người dùng" (User List) with columns for STT, Tên đầy đủ (Full Name), Email, Số điện thoại (Phone Number), Loại tài khoản (Account Type), Trạng thái kích hoạt (Activation Status), and Hành động (Actions). The table lists 10 users, each with a unique ID, name, email, phone number, account type (Facebook or Bình thường), activation status (Đã kích hoạt or Chưa kích hoạt), and a set of icons for actions like edit, delete, and deactivate.

Figure 6-61: Users list screen

### 6.3.2.4 Manage Project

To accept/reject project. Admin do the step below:

- Step 1: Go to Admin page, Click on [Danh sách dự án]. System returns list project.
- Step 2: Choose account and click view button. System go to project detail page.
- Step 3: Click [Chấp nhận/Từ chối] button to accept/reject project.

The screenshot shows a project management interface. At the top, there's a header with "Thông tin chung" and a breadcrumb navigation: "Danh sách dự án / Danh sách dự án". Below this is a table titled "Danh sách dự án" with columns for Mã (ID), Tên (Name), Lĩnh vực (Category), Người tạo (Creator), Mục tiêu (Target), Ngày kết thúc (End Date), Lượt ủng hộ (Number of Backers), Tiền được ủng hộ (Total Backed Amount), Trạng thái (Status), and two small icons. The table lists six projects, each with a unique ID, name, category, creator email, target amount, end date, number of backers, total backed amount, status (Hết hạn, Đang chạy, Chờ duyệt), and a set of icons for actions like edit, delete, and preview.

Figure 6-62: Projects list screen

### 6.3.2.5 View Backing

Admin can manage list banker.

Danh sách ứng hò					
Bảng điều khiển / Danh sách ứng hò					
Danh sách ứng hò					
Hiển thị	10	Tim kiếm:			
#	Tên dự án	Tên người ứng hò	Số tiền ứng hò	Thời gian ứng hò	
1	EEG Cloud	Vu Cong Chinh	1,000,000 ₫	14:55 ngày 29-10-2015	
2	Sphericam	Vu Cong Chinh	2,000,000 ₫	17:08 ngày 29-10-2015	
3	Sphericam	Account Test 1	1,000,000 ₫	17:51 ngày 29-10-2015	
4	Divinity: Original Sin 2	Test Accountt	200,000 ₫	18:03 ngày 29-10-2015	
5	Sphericam	Lưu Ngọc Mạnh	2,000,000 ₫	22:20 ngày 29-10-2015	
6	Sphericam	Lưu Ngọc Mạnh	10,000 ₫	00:19 ngày 30-10-2015	
7	Sphericam	Lưu Ngọc Mạnh	50,000,000 ₫	01:27 ngày 30-10-2015	

Figure 6-63: Backing list screen

### 6.3.2.6 Manage Category

Admin can manage categories. (Create new, edit, lock, unlock category)

Quản lý danh mục					
Bảng điều khiển / Danh sách danh mục					
Danh sách danh mục					
Hiển thị	10	Tim kiếm:			
ID	Tên danh mục	Mô tả	Số dự án	Trạng thái	
1	Âm Nhạc	abncbnbc cbcn nbcnbc	4	Mở	
2	Nghệ Thuật	abncbnbc cbcn nbcnbc	0	Mở	
3	Truyện Tranh	abncbnbc cbcn nbcnbc	5	Mở	
5	Công Nghệ	abncbnbc cbcn nbcnbc	11	Mở	
6	Trò chơi	trò chơi, ứng dụng điện thoại, video game	0	Mở	
7	Truyện ngắn	không có j cả chỉ là test thôi	0	Khóa	
Hiển thị từ 1 đến 6 của 6					
Trước 1 Tiếp					

Figure 6-64: Category management screen

### 6.3.2.7 Manage Slide

Admin can manage sliders. (Create new, edit, active/deactive slide)

Danh sách Slide						
#	Tiêu đề	Mô tả	Ảnh	Trạng thái	Thứ tự	
1	The World's Cleanest Power Plant	Support a team of architects who are working to reduce carbon emissions by making art in the sky.		Mở		
2	Eco - Global Survival Game	Collaborate to build civilization in a simulated ecosystem, creating laws to make group decisions.		Mở		
3	Emmett Louis Till, 1941-1955	His murder catalyzed the civil rights movement. Help make the film that will tell his story.		Mở		
4	khu vườn trên mây	slide của khu vườn trên mây		Mở		

**Figure 6-65:** Slide management screen

### 6.3.2.8 Message (same as message of user)

Admin can manage message. (Create sent, receive, delete message)

Tôi: Account Test 1	Dandelion - duyệt thành công dự án Thiết bị đeo Vufine: Biển kính thường t hành màn hình!	21/11/15
Tôi: Account Test 00	Dandelion - duyệt thành công dự án Gây quỹ Xuất bản sách "Truyện Cực Ng ắn" của Đào Quang Huy!	21/11/15
Tôi: Account Test 1	Dandelion - duyệt thành công dự án Máy tính siêu tí hon chơi game lướt we b có giá chỉ... 9 USD!	21/11/15
Tôi: Account Test 1	Dandelion - duyệt thành công dự án Silk Light - Bóng đèn thông minh "đọc v ì" đồng hồ sinh học!	21/11/15
Tử: Test Account	<b>Test gửi tin nhắn</b>	19/11/15
Tôi: Account Test 00	Dandelion - duyệt thành công dự án Gây quỹ ủng hộ dự án xuất bản "Quan t rong là phải Đẹp Trai!"	19/11/15

Figure 6-66: Message screen

### 6.3.2.9 Manage Report

#### 6.3.2.9.1 Manage User Report

- Step 1: Click on [Báo cáo người dùng] hyperlink
- Step 2: Choose user to report
- Step 3: Click on [Đồng ý]/[Hủy]

#	Người báo xấu	Người bị báo xấu	Ngày báo	Trạng thái
Không có dữ liệu				

Hiển thị 10

Trước Tiếp

Hiển thị từ 0 đến 0 của 0

Figure 6-67: User Report management screen

#### 6.3.2.9.2 Manage Project Report

- Step 1: Click on [Báo cáo dự án] hyperlink
- Step 2: Choose project to report
- Step 3: Click on [Đồng ý]/[Hủy]

Báo xấu dự án

Quản lý chung / Báo xấu dự án

Danh sách dự án bị báo xấu

#	Người báo xấu	Dự án bị báo xấu	Người tạo dự án	Ngày báo	Trạng thái
1	Administrator	Divinity: Original Sin 2	Lưu Ngọc Mạnh	5/11/2015	<span>Đã đồng ý</span> <span>(eye)</span>
2	Test Account	Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	Account Test 00	19/11/2015	<span>Đã xem</span> <span>(eye)</span> <span>(checkmark)</span> <span>(cross)</span>
3	Administrator	Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	Account Test 00	19/11/2015	<span>Đang chờ</span> <span>(eye)</span> <span>(checkmark)</span> <span>(cross)</span>
4	Administrator	Gây quỹ xuất bản cuốn sách "Mặt Ngọt Chết Mèo"	Account Test 00	19/11/2015	<span>Đang chờ</span> <span>(eye)</span> <span>(checkmark)</span> <span>(cross)</span>

Hiển thị từ 1 đến 4 của 4

Trước 1 Tiếp

**Figure 6-68:** Project Report management screen

## 7 PROJECT RESULT REPORT

### 7.1 Product Judging

#### 7.1.1 Advantage of Dandelion system

The idea of Dandelion project is to develop a website where anyone can launch their project with the creative ideas to get funding from community and they can make their ideas and products become a reality. We want to create a website that support for Vietnamese people by using the payment method that is widely used by Vietnamese people and support them to create or back projects easily.

As the goals that we have set before started project, the Dandelion system has been developed in right direction. And what it provides to users is listed below:

- Guest: guest can search, view projects which is fundraising on Dandelion website with friendly interface.
- Member: member can create, launch and back the project to get the rewards, deals from creator. They can manage their backed, created and reminded project easily.
- Admin: Admin can manage users, projects in system. Admin can view statistics about projects, backed money and profit.

#### 7.1.2 Current Limitations

- Some screen can't display well on Mobile or screen which has resolution lower than 1024x768.
- Website not support for English language.
- Not support payment methods for people in other countries.
- Not support payment by mobile cards.

#### 7.1.3 Expectation in Future

- Support multi-language and other payment method such as visa and master card to get funding from people in other countries.
- Support payment by mobile cards such as Viettel, Mobi, Vina, ...
- App for Android and IOS which supports members manage their projects and admin can manage system anywhere by smart phone.
- Expand server to increase number of users.

### 7.2 Project Summary

<b>Project Title</b>		<b>Dandelion</b>	
<b>Project Supervisor</b>		Nguyễn Văn Sang	
<b>Committers (6)</b>			
1	<b>Project Manager</b>	Lưu Ngọc Mạnh	<a href="mailto:manhlnse02619@fpt.edu.vn">manhlnse02619@fpt.edu.vn</a>
2	<b>Team member</b>	Vũ Công Chính	<a href="mailto:chinhvcse02585@fpt.edu.vn">chinhvcse02585@fpt.edu.vn</a>
3	<b>Team member</b>	Nguyễn Minh Huy	<a href="mailto:huynmse02723@fpt.edu.vn">huynmse02723@fpt.edu.vn</a>
4	<b>Team member</b>	Vũ Ngọc Trung	<a href="mailto:trungvnse02967@fpt.edu.vn">trungvnse02967@fpt.edu.vn</a>
5	<b>Team member</b>	Đương Đức Anh	<a href="mailto:anhddse02700@fpt.edu.vn">anhddse02700@fpt.edu.vn</a>

<b>6</b>	<b>Team member</b>	Cao Thị Phương Mai	<a href="mailto:maictpse02908@fpt.edu.vn">maictpse02908@fpt.edu.vn</a>
<b>Main objective</b>		Develop a system which will provide to users functionality to collect funding for their project and back to get the amazing products from creative projects.	
<b>Purpose</b>		Provide a Dandelion online on website.	
<b>Source Repository</b>		<a href="https://github.com/sangnvus/2015FALLJS01.git">https://github.com/sangnvus/2015FALLJS01.git</a>	
		<b>Plan</b>	<b>Actual</b>
<b>Start Date</b>		07/09/2015	07/09/2015
<b>End Date</b>		25/12/2014	23/12/2015
<b>Releases version</b>		1.0	1.0
<b>Duration</b>		80	78
<b>Effort (person-day)</b>		480	468
<b>Test cases</b>		1800	1524
<b>Defects</b>		200-300	66
<b>Document pages</b>		~ 3000 pages	~ 3500 pages
<b>Lines of code (C#)</b>		~ 10000	~ 15000
<b>Lines of code (HTML)</b>		~ 5000	~ 10000
<b>Lines of code (AngularJS)</b>		~ 10000	~ 13000
<b>LOC (Total)</b>		~ 25000	~ 38000

**Table 7-1: Project summary****Knowledge:**

- Work process of FPT Software to develop a project.
- Putting the learned knowledge into practice.
- Improve skills: teamwork, leadership, research and self-study, plan and manage time.
- Technical: AngularJS (Single Page Application), Azure Web App, WebApi 2.0, ASP.Net MVC5.
- Tools: Visual Studio, TortoiseGit, Enterprise Architect, Office tools.

### 7.3 Lesson Learnt

- Together with leadership skills, the project manager needs to be aware of the strengths and weaknesses of his/her staff, so that the talents are harnessed and the shortfalls downplayed for the benefit of the project.
- The success of a project is largely dependent on the skills and strengths of the people involved. Therefore, a project needs to have a dedicated, talented set of individuals working towards a common goal.

- If you have the knowledge and experience to make a decision, then you should go ahead and so, without expecting leader to assign you task at every turn.
- Procrastination does not work. After assimilating the relevant information, decisions need to be made. Wrong decisions can be salvaged, if discovered early; but right decisions cannot be postponed.
- Be open to change. Sometimes, you may find that the things you knew along may not be correct at this given time, under these specific conditions.
- Good communication is that which will stop mistakes from becoming failures.
- Please think carefully before making a decision because if the wrong decisions you'll have to redo it from scratch.

## 8 REFERENCES

- [1]. Kickstarter  
<https://www.kickstarter.com/>
- [2]. Comicola  
<https://www.comicola.com/>
- [3]. Indiegogo  
<https://www.indiegogo.com/>
- [4]. Wikipedia, 2012, IBM Rational Unified Process, viewed 30 September 2013  
[http://en.wikipedia.org/wiki/IBM\\_Rational\\_Unified\\_Process](http://en.wikipedia.org/wiki/IBM_Rational_Unified_Process)
- [5]. AngularJS Overview  
[http://www.tutorialspoint.com/angularjs/angularjs\\_overview.html](http://www.tutorialspoint.com/angularjs/angularjs_overview.html)
- [6]. Capstone Project Report Teamplate – FPT University, Ha Noi, Viet Nam.
- [7]. Report document guidelines (2013) – FPT University, Ha Noi, Viet Nam.
- [8]. Template\_Project\_Plan, (2010), FPT University, Hanoi, Vietnam
- [9]. Template\_Software\_Requirements\_Specification, (2010), FPT Software, Hanoi, Vietnam
- [10]. Template\_Software\_Design\_Template, (2010), FPT Software, Hanoi, Vietnam
- [11]. Templates for Test case & Defect log, FPT Software, Hanoi, Vietnam
- [12]. The Unified Modeling Language diagram  
<http://www.uml-diagrams.org/>