

MINISTRY OF EDUCATION AND TRAINING



Capstone Project Document

Jobs Finding System

Project Code: JFS

JFS Team		
Group members	Tuong Thi Thuy	SE02678
	Ly Hai Nam	SE03045
	Duong Dang Duy	SE03254
	Tran Ha Nam	SE03311
	Pham Tuan Son	SE03230
Supervisor	Mr. Tran Dinh Tri	
Capstone Project Code	JFS	

Hanoi, 5th September, 2016

Acknowledgements

We wish to express our deepest gratitude to our supervisor: Mr. Tran Dinh Tri for his tutorial, sharing and motivating throughout the project. Under the instructions of Mr. Tran Dinh Tri, the project team has always completed all assigned tasks and solved all problems. Apart from all the technologies and methodologies, Mr. Tran Dinh Tri shared with us the philosophy of keeping the right attitude towards problems. We believe that was the most important factor that led to the success of this project.

We would also like to thank the instructors at FPT University for all the classes. We hope you will find this project as a reflection of the knowledge and experiences you have given us.

Finally, we truly appreciate all of our friends who have discussed with us. With your help, we have been recognized and solved every arising problem.

TABLE OF CONTENTS 目次

1	INTRODUCTION はじめに	8
1.1	Purpose 目的	8
1.2	Project information プロジェクトの情報	8
1.3	Project members プロジェクトメンバー	8
1.4	Our proposal 提案	9
1.4.1	Problem 問題	9
1.4.2	Solution 解決	9
1.4.3	Idea アイディア	9
1.5	Product 商品	9
1.5.1	Basic Functions 基本的な機能	9
1.6	Benefit from project プロジェクトからベネフィット	11
1.6.1	For our group グループのために	11
1.6.2	For user ユーザーのために	11
2	SOFTWARE PROJECT MANAGEMENT ソフトウェアプロジェクト管理	12
2.1	2.1 Software development process ソフトウェア開発のプロセス	12
2.2	Project Organization プロジェクト組織	13
2.2.1	Organizational Structure 組織構造	13
2.2.1	2.2.2 Project Team プロジェクトチーム	14
2.3	Project schedule プロジェクトスケジュール	15
2.2	Meeting minutes 打ち合わせ議事録	17
2.3	Project Milestones プロジェクトマイルストーン	18
2.4	Project Deliverables プロジェクトの成果物	19
2.5	Resource リソース	20
2.5.1	Human resource	20
2.5.2	Non-human resource 機器やツール	20
2.6	Risk Management リスク管理	20
2.7	Communication Management コミュニケーション管理	22
2.7.1	Communication between Team Members チームメンバーの間のコミュニケーション	22
2.7.2	Communication with Supervisor 指導教員とコミュニケーション	22

2.8 Configuration Management Process コンフィグレーション管理.....	22
2.8.1 CI Identification and Naming Convention 構成アイテムと命名規則	22
2.8.2 Project Infrastructure プロジェクトインフラストラクチャ.....	23
2.8.3 Directory structure ディレクトリ構造	24
2.8.4 Coding convention コーディング規約.....	26
3 SOFTWARE REQUIREMENTS SPECIFICATION はじめに	29
3.1 Introduction はじめに	29
3.1.1 Purpose 目的	29
3.1.2 Scope 範囲.....	29
3.1.3 References 参照資料	29
3.1.4 Overview 概要	29
3.2 Overall Description 概要説明.....	30
3.2.1 Use Case Diagram ユースケース図	30
3.3 Product Features.....	32
3.3.1 Use Characteristics	35
3.3.2 User Documentation	36
3.3.3 Assumption	36
3.4 Entity Relation Model	37
3.4.1 Entity-Relationship Diagram.....	37
3.4.2 Entity description.....	42
3.4.3 Relationship	42
3.5 Specific Requirements 要求仕様.....	44
3.5.1 Business Rules ビジネスルール	44
3.5.2 Functionality 機能要求	45
3.6 Usability ユーザビリティ	132
3.7 Reliability 信頼性.....	132
3.8 Performance 性能	133
3.9 Maintainability 保全性	133
3.10 Supporting Information サポート情報.....	133
4 ARCHITECTURE DESIGN アーキテクチャ設計書	134
4.1 Introduction はじめに	134
4.1.1 Purpose 目的	134

4.1.2	References 参照資料	134
4.2	Architectural Overview アーキテクチャ概要	134
4.2.1	System Architecture システムアーキテクチャ	134
4.3	System Architecture Explanation システムアーキテクチャについての説明	135
4.4	Design of Jobs Finding System Jobs finding system の設計書	136
4.4.1	Database Design データベース設計	136
4.4.2	Database detail データベースの詳細	136
4.5	CRC Cards CRC カード	143
4.5.1	Model モデル	Error! Bookmark not defined.
4.5.2	View ビュー	Error! Bookmark not defined.
4.5.3	Controller コントローラー	Error! Bookmark not defined.
4.6	Detail Design 詳細設計	143
4.6.1	Guest ゲスト	154
4.6.2	Member メンバー	164
4.6.3	Admin 管理	211
4.6.4	System システム	255
5	SOFWARE TESTING ソフトウェアテスト	256
5.1	Introduction はじめに	256
5.1.1	Purpose 目的	256
5.1.2	References 参照資料	256
5.1.3	Background information 背景情報	256
5.1.4	Scope of testing テスト範囲	257
5.2	Constraints 制約事項	257
5.2.1	Risk list リスク一覧	257
5.3	Requirements for Test テスト要求	258
5.3.1	Test items テスト項目	258
5.3.2	Acceptance Test Criteria テスト受入基準	261
5.3.3	Feature not to be tested	262
5.4	Test strategy テスト戦略	262
5.4.1	Test Model テストモデル	262
5.4.2	Test types テストタイプ	264
5.4.3	Function Testing 機能テスト	264

5.4.4	User Interface Testing ユーザーインターフェーステスト	265
5.4.5	Data and Database Integrity Testing データおよびデータベース整合性テスト	266
5.4.6	Test stages テストステージ	266
5.5	Resource リソース	266
5.5.1	Human Resource 人的リソース	267
5.5.2	Test management テスト管理	267
5.5.3	Defect management 欠陥管理	268
5.6	Test environment テスト環境	269
5.6.1	Hardware ハードウェア	269
5.6.2	Software ソフトウェア	269
6	Test Milestones テストマイルストーン	270
6.1	Deliverables 納品物	271
6.2	Test Case テストケース	271
6.3	Test Report テストレポート	272
6.3.1	Common Test Report	273
6.3.2	Integration Test Report 統合テストレポート	273
6.3.3	System Test Report システムテストレポート	273
6.3.4	Test Report テスト	274
7	USER MANUAL ユーザーマニュアル	274
7.1	Install Guide インストールガイド	274
7.1.1	Introduction はじめに	274
7.1.2	Installation Guideline インストールのガイドライン	275
7.1.3	Other considerations	279
7.2	User Guide ユーザーガイド	279
7.2.1	Guest ゲスト	279
7.2.2	Member メンバー	284
7.2.3	Admin 管理	314
7.2.4	System システム	338

Definition and acronyms

Acronym	Definition	Note
JFS	Jobs Finding System	
SRS	Software Requirement Specifications	
CI	Configuration Item	
CM	Configuration Management	
Admin	Administrator	
Member	Employer and Job Seeker	
PM	Project Manager	
PTL	Project Technical Leader	
CRC	Class-Responsibility-Collaborators	
AT	Acceptance Test	
IT	Integration Test	
KLOC	1000 lines of code	
PM	Project Manager	
PTL	Project Technical Leader	
PT/TT	Program Test/ Total Test	
QA	Quality Assurance	
UT	Unit Test	
PC	Personal Computer	
UI	User Interface	

1 INTRODUCTION はじめに

1.1 PURPOSE 目的

This chapter provides an overview of Jobs Finding System, it includes background information, literature review of existing system and raising a proposal for ideas of improvement.

1.2 PROJECT INFORMATION プロジェクトの情報

Project name:	Jobs Finding System
Project code:	JFS
Project type:	Website
Project category:	New development
Business type:	Website Service
Project manager:	Tuong Thi Thuy
Timeline:	5 th September, 2016 to 14 th December, 2016

Table 1.1 - Project Information

1.3 PROJECT MEMBERS プロジェクトメンバー

❖ Supervisor:

Full name	E-mail	Title
Tran Dinh Tri	Tritd@fpt.edu.vn	Lecturer

❖ Team member:

Full name	Roll number	E-mail	Phone	Position
Tuong Thi Thuy	SE02678	Thuyttse02678@fpt.edu.vn	01213038796	Leader
Ly Hai Nam	SE03045	Namlhse03045@fpt.edu.vn	01674277154	Member
Duong Dang Duy	SE03254	Duyddse03254@fpt.edu.vn	0986676226	Member
Pham Tuan Son	SE03230	Sonptse03230@fpt.edu.vn	0943977613	Member
Tran Ha Nam	SE03311	Namthse03311@fpt.edu.vn	01634177717	Member

1.4 OUR PROPOSAL 提案

1.4.1 Problem 問題

The rapid change in technology, knowledge explosion, the increased and complex jobs-finding demands or employees- finding for work challenges so much. People who have free times want to find part-time jobs to earn more income. Moreover, for part-time jobs, Job Seekers also find it difficult to find Employers because of the increase and problems of fees. Therefore, the posed problems are making an interaction between Job Seekers and Employers, and help them to search for other sides as quick as possible.

1.4.2 Solution 解決

Create connection between Employers and Job Seekers through our system.

Provide employment information to Job Seeker, and jobseeker's information to Employers.

By the automatic searching function, cost and time are minimized.

1.4.3 Idea アイディア

So our idea is creating a website not only to link Employers and Job -Seekers but also to manage information about job needs quickly and efficiently.

1.5 PRODUCT 商品

After researching carefully about the market needs of job finding, we have chosen main functions which are necessary and important to our system. They will help the manager easy to use and operate. Besides, we also compare to other website and research about user's habits to design good and intuitive interface. We want to reduce all limitation and improve our own advantages.

The objective user of our Website is Vietnamese people who living in Ha Noi and Ho Chi Minh city. Therefore, our website using Vietnamese as primary language, in order to meet the criteria of user-friendly.

1.5.1 Basic Functions 基本的な機能

Guest

- View and search all Employers and Job Seeker's posts.
- Register (Create a profile to become a member).

Member: includes of Job Seeker and Employer

User can only post as Job Seeker or Employer if they are already logged in by a valid account.

❖ Job Seeker

- View own personal profile.
- Edit User Account's profile.
- Create and post Job Finding Form (fields). This post will be auto matched with suitable posts from Employer.
- Edit own post: only edit before post is selected by any Employer.
- Delete own post.
- Search and view other Job Seeker's posts.

- Search and view Employer's posts.
- View Employer's profile.
- Apply suitable job.
- Online chat with admin (Converse with admin directly, faster than sending email).
- Comment and report an issue.
- Report other member's posts.

❖ **Employer**

- View own personal profile.
- Edit User Account's profile.
- Create and post Job Form (fields). This post will be auto matched with suitable Job Finding Form from Job Seeker.
- Edit own post: only edit before post is selected by any Job Seeker.
- Delete own post: only edit before post is selected by any Job Seeker.
- Search and view other Employer's posts.
- Search and view Job Seeker's posts.
- View Job Seeker's Page (all posts in history).
- View Job Seeker's profile.
- Get suitable Job Seeker. (deduct money on Employer's Financial Account, View Job Seeker Profile)
- Online chat with admin (Converse with admin directly, faster than sending email).
- Comment and report an issue.
- Report other member's posts.

Admin

❖ **Manage Member**

- Search and view Member's profile.
- Ban and unban Member.
- Delete Member.
- Add an admin.

❖ **Manage Employer's Financial Account**

- Add money into Employer's Financial Account.

❖ **Manage Post**

- Search and view Member's post.
- Delete Member's post.
- View report from Member and Guest.

❖ **Manage Job Category**

- Add category.
- Edit category.
- View category.
- Delete category.

❖ **Manage comment and reported issues about performance and interface of system.**

- Admin manage comment and reported issues about performance and interface of system.

❖ **Statistics**

- Statistic website's earned money from Employers.
- View chart to compare between job types.

❖ **Online chat with Member.**

System

- **Send hidden profile of a Job Seeker to Employer:** After System deduct money in Employer's Account successful, system display a pop up contain Profile of Job Seeker. System also send email to Employer, this Employer be able to access to pick Job Seeker's Profile Page.
- Send notification to Member's email in the following cases:
 - Delete Member banned over 3 times.
 - Ban and unban Member based on System's Rule: 1 member will be banned if got 3 deleted posts by admin. Unban after 1 week.
 - Deduct Employer's money when they are received Job Seeker's hidden profile.
- Delete user when that user is banned 3 times.
- Ban user based in website's rule.
System active event "Delete User" and "Ban User" after event "Delete Post" occurs (delete by admin).
- Deduct Employer's money when they are received Job Seeker's hidden information. (Event "Deduct Money" occurs when an Employer accept a transaction".
- Create chart to compare between job types. (Event "Create Chart" occurs when Admin click on link "View Chart".
- List matching posts

Out of scope

❖ **Manage Website's payment activities**

By using website, there are some services that require member's fee. Therefore, website will use online payment service

1.6 BENEFIT FROM PROJECT プロジェクトからベネフィット

1.6.1 For our group グループのために

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 PHP, Laravel, and HTML, Bootstrap.
- Know how to communicate with team members and how to make teamwork more effective.

1.6.2 For user ユーザーのために

- Improving chances to finding part-time jobs for Job Seeker.

- Easy to find another, saving time and costs.
- Job Seeker don't need to paid money to use service.

2 SOFTWARE PROJECT MANAGEMENT ソフトウェアプロジェクト管理

In this section, we will describe our organization and plan of project. Each member of team must use this section as a guideline tracking assigned tasks and deadlines. In addition, we introduce our team and overview about project. Meeting minute also present in this section. Moreover, we will describe risks of project and the way communication of our team.

2.1 SOFTWARE DEVELOPMENT PROCESS ソフトウェア開発のプロセス

This figure below will describe the information and products flow lifecycle process model.

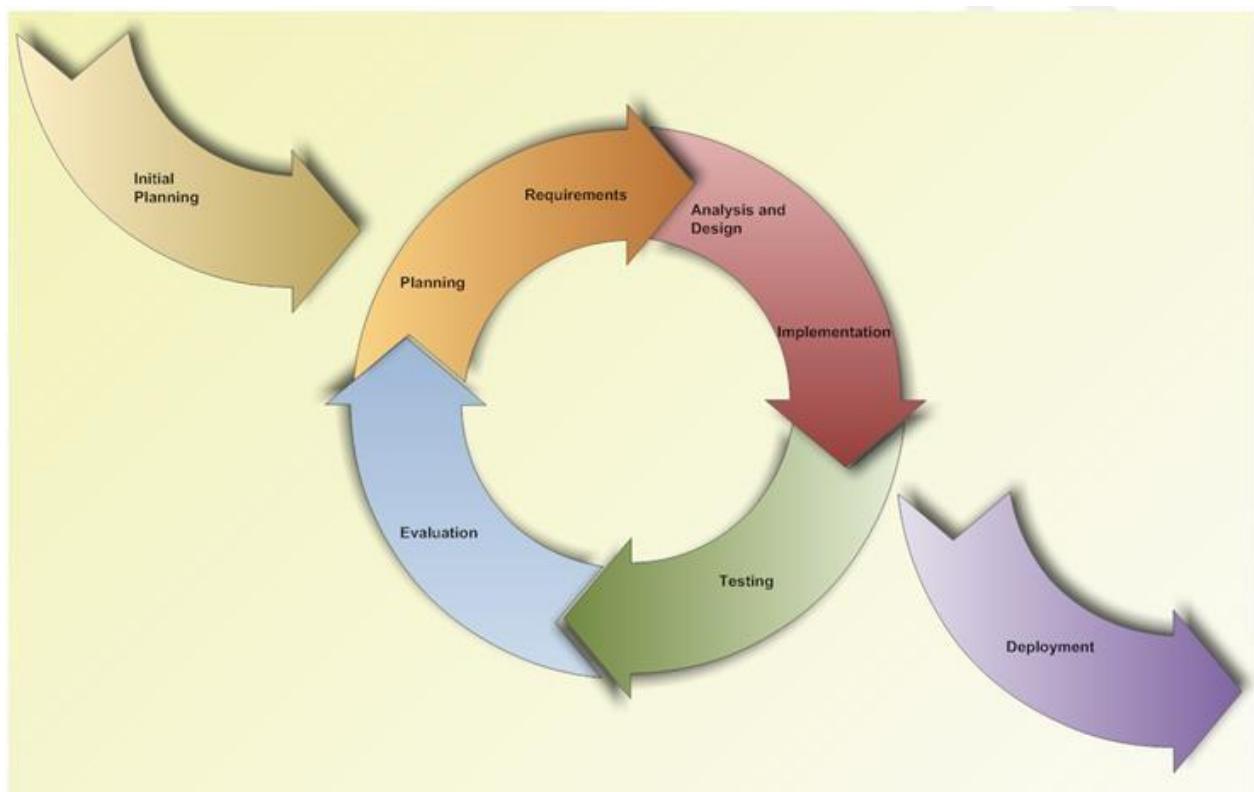


Figure 2.1 Iterative software process model

JFS project uses the Iterative Software Process Model. We have chosen the Iterative Software Process Model because the scope of our project is big, the major requirements were defined clearly, some more detail will be added in time, and for the newbie group in software development. By using this software process model, we generates working software quickly and early during the software life cycle. This model is more flexible – less costly to change scope and requirements. We easier to manage risk because risky pieces are identified and handled. Besides, the iterative model is easier than other models when the issues are discovered. All issue will be raised to the team. They will be solved while the project is still developed.

In our project, we have two iterations:

Iteration 1: Start at Monday 05st – September – 2016 to Thursday 17st – November – 2016. In this iteration we will analyze, design, code and test the main features of project.

Iteration 2: Start at Friday 18th – November – 2016 to Thursday 17th – December – 2016. In this iteration we will fix bugs in iteration 1, complete other features, test again and packing product.

2.2 PROJECT ORGANIZATION プロジェクト組織

2.2.1 Organizational Structure 組織構造

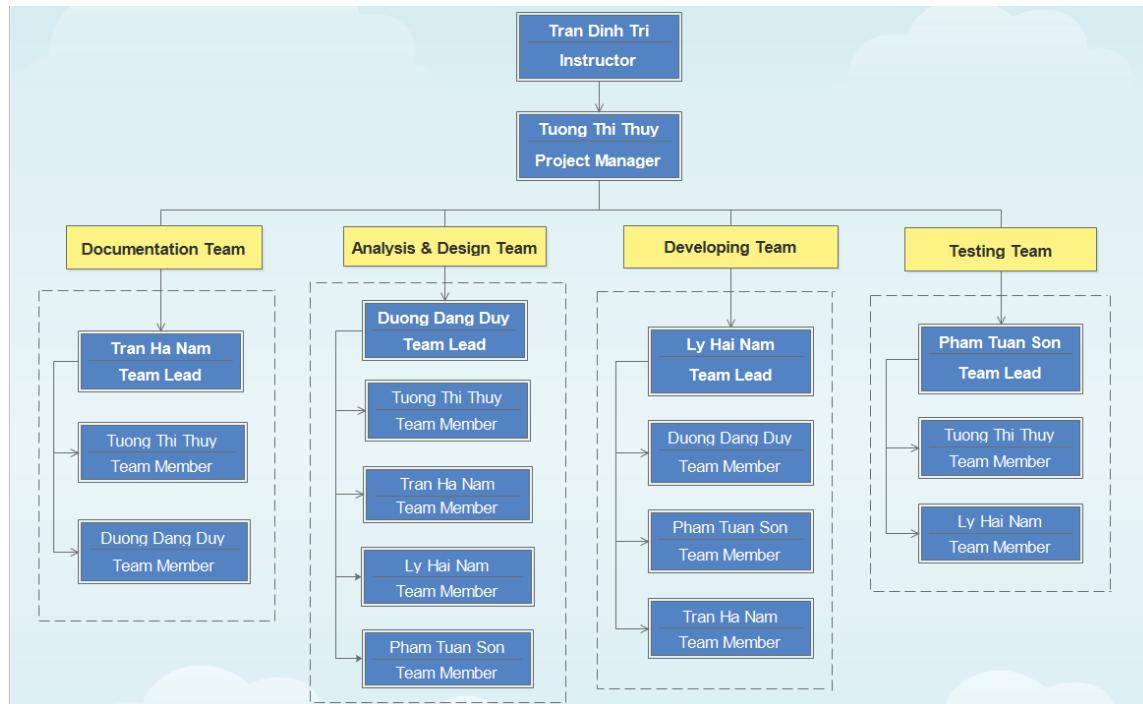


Figure 2.2.1 Organizational structure chart

2.2.2 Project Team プロジェクトチーム

Role	Responsibility	Full name	% Effort	Start date	End date
Instructor	<ul style="list-style-type: none"> - Provide templates & tools - Review deliverables - Review project status - Resolve escalated issues - Consultant 	Trần Đình Trí	10%	September 05 th , 2016	December 16 th , 2016
PM	<ul style="list-style-type: none"> - Planning and defining scope - Developing schedule - Reviewing and Tracking document - Identify resources and assign individual responsibilities - Assign task to team members - Coordinating communication 	Tường Thị Thúy	100%	September 05 th , 2016	December 16 th , 2016
Analysis and Design Team					
Team Leader	<ul style="list-style-type: none"> - Reviewing and Tracking design - Identify main UI - Draw prototype, Architecture Design - Database Design 	Dương Đăng Duy	100%	September 23 th , 2016	November 30 th , 2016
Team member #1	<ul style="list-style-type: none"> - Database Design - Draw prototype, Architecture Design 	Lý Hải Nam	100%	September 23 th , 2016	November 30 th , 2016
Team member #2	<ul style="list-style-type: none"> - Database Design - Draw prototype, Architecture Design 	Trần Hà Nam	100%	September 23 th , 2016	November 30 th , 2016
Team member #3	<ul style="list-style-type: none"> - Screen Design 	Tường Thị Thúy	100%	September 23 th , 2016	November 30 th , 2016
Team member #4	<ul style="list-style-type: none"> - Screen Design 	Phạm Tuấn Sơn	100%	September 23 th , 2016	November 30 th , 2016
Documentation Team					

Team Leader	- Reviewing and Tracking Requirement	Trần Hà Nam	100%	September 05 th , 2016	December 15 th , 2016
Team member #1	- Create SRS document	Dương Đăng Duy	100%	September 05 th , 2016	December 15 th , 2016
Team member #2	- Create SRS document	Tường Thị Thúy	100%	September 05 th , 2016	December 15 th , 2016
Development Team					
PTL	- Developing - Solve issue of project	Lý Hải Nam	100%	October 12 th , 2016	December 08 th , 2016
Developer #1	- Developing	Trần Hà Nam	100%	October 12 th , 2016	December 08 th , 2016
Developer #2	- Developing	Dương Đăng Duy	100%	October 12 th , 2016	December 08 th , 2016
Developer #3	- Developing	Phạm Tuấn Sơn	100%	October 12 th , 2016	December 08 th , 2016
QA and Test Team					
Test Leader	- Responsible for test execution - Test results recording - Evaluation of test run and error recovery	Phạm Tuấn Sơn	100%	November 08 th , 2016	December 13 th , 2016
Tester #1	- Testing	Tường Thị Thúy	100%	November 08 th , 2016	December 13 th , 2016
Tester #2	- Testing	Lý Hải Nam	100%	November 08 th , 2016	December 13 th , 2016

Table 2.2.2 Project Team

2.3 PROJECT SCHEDULE プロジェクトスケジュール

Below are the image of task list used to assign and tracking tasks. Refer to JFS_Schedule_v1.0.mpp to see more detail.

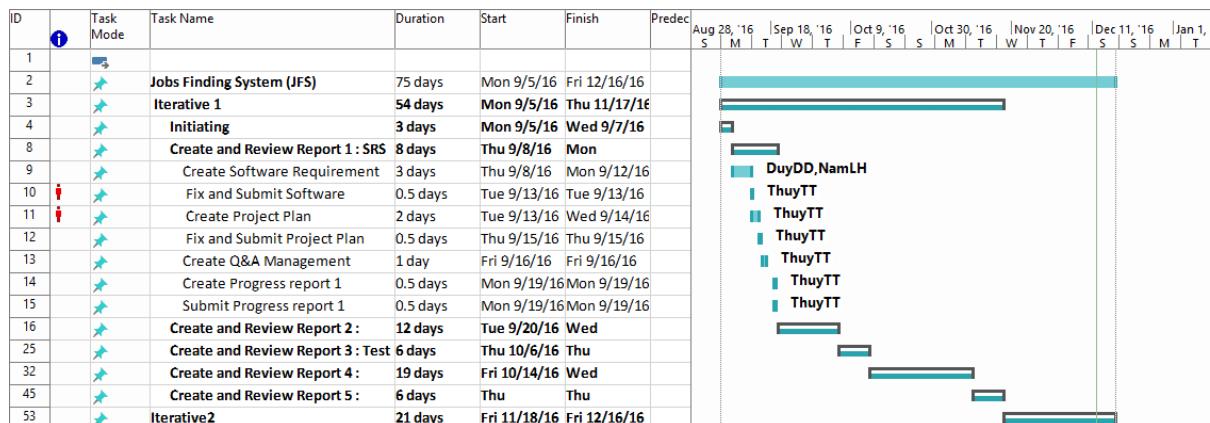


Figure 2.3: Organizational structure chart

2.4 MEETING MINUTES 打ち合わせ議事録

All meeting minutes will be written follow this template:

Meeting Minutes 1			
Subject	Comment on Report 1	Date	2016/09/08
Conductor	Prof. TriTD	Time	13 P.M
Location	P310 ALFA	Secretary	DuyDD
Attendees	ThuyTT, NamLH, NamTH, SonPT		
Absent	No		

Key Points Discussed		
NO.	TOPIC	HIGHLIGHTS
1	Common work progress.	Done writing report 1, report 2 in progress
2	Assigning someone to write meeting minutes.	Assign DuyDD
3	Review report 1	Something need to be fixed: - Incorrect grammar, vocabulary
4	Setup working environment : Tools to manage source code, documents, database...	Done
5	Learn the business. Learn about web advertising business. Learn about online payment.	In progress
6	Submit report 2	In progress

Action Plan			
NO.	ACTION ITEM(S)	OWNER	TARGET DATE
1	Define the roles & responsibilities of each team member	ThuyTT	2016/09/08
2	Analyze requirement, make project plan to be written in template format	ThuyTT, DuyDD, NamTH	2016/09/10
3	Design: ERD, System architecture, Class diagram, Sequence diagram, Use case diagram...	DuyDD, NamLH, NamTH, SonPT	2016/09/10
4	Estimate number of use cases, the number of 50 use cases is appropriate	NamLH ,NamTH, SonPT	2016/09/11
5	Write the report 2, report 3	DuyDD, ThuyTT	2016/09/11

2.5 PROJECT MILLSTONES プロジェクトマイルストーン

No	Milestone	Completion Date	Verification
1	Project Start	September 5 th , 2016	Instructor approval
2	Submit report 1	September 8 th , 2016	Instructor approval
3	Submit report 2	September 13 th , 2016	Instructor approval
4	Submit report 3	September 20 th , 2016	Instructor approval
5	Submit report 4	October 05 th , 2016	Instructor approval
6	Submit report 5	November 17 th , 2016	Instructor approval
7	Submit report 6	December 15 th , 2016	Instructor approval
9	Defend project	December 31 th , 2016	Instructor approval

Table 2.5 Project Milestones

2.6 PROJECT DELIVERABLES プロジェクトの成果物

No	Deliverable	Committed date	Delivery	Description of Deliverable	Delivery media
1	Capstone project register	September 5 th , 2016			Hard copy
Iteration 1					
2	Software Requirement Specifications - SRS	September 16 th , 2016	Document		Commit to Github
3	Q&A Management	September 26 th , 2016	Document		Commit to Github
4	User requirements	September 16 th , 2016	Document		Commit to Github
5	Project plan	September 09 th , 2016	Document		Commit to Github
6	Progress report 1	September 08 th , 2016	Document		Commit to Github
7	Architecture design	October 03 th , 2016	Document		Commit to Github
8	User interface design	October 11 th , 2016	Document		Commit to Github
9	Progress report 2	September 13 th , 2016	Document		Commit to Github
10	Database design	October 07 th , 2016	Document		Commit to Github
11	Test plan	November 08 th , 2016	Document		Commit to Github
12	Test case	November 14 th , 2016	Document		Commit to Github
13	Progress report 3	September 20 th , 2016	Document		Commit to Github
14	Progress report 4	October 05 th , 2016	Document		Commit to Github
15	Test case - execute	November 14 th , 2016	Document		Commit to Github
16	Test report	November 15 st , 2016	Document		Commit to Github
17	Progress report 5	November 17 th , 2016	Document		Commit to Github
Iteration 2					
18	User manual	December 14 th , 2016	Document		Commit to Github
19	Progress report 6	December 14 th , 2016	Document		Commit to Github
20	Final report	December 14 th , 2016	Document		Hard copy

Table 2.6 Project Deliverables

2.7 RESOURCE リソース

2.7.1 Human resource 人的リソース

- Team members
- Supervisor

2.7.2 Non-human resource 機器やツール

- Equipment: Desktop, Laptop, Samsung Galaxy Tablet, Microsoft Windows Phone.
- Building: ALFA & BETA Building - FPT University.

2.8 RISK MANAGEMENT リスク管理

No	Name	Root Cause	Probability	Avoidance plan	Contingency plan	Impact
1	Team don't keep the deadline on time.	Members lack of responsibility, have bad habits. Lack of management.	Medium	Make rules and penalties for people who doesn't meet the deadline.	Working overtime to complete on time Punish member by money...	High
2	Requirement changed while project is processing.	SRS is not provide all information about user requirement such as: lack of customer's requirement, misunderstand requirement of customer.	High	SRS should be review carefully by PM and the Supervisor. Always do brainstorming carefully to design detail system. Hold all meeting and make minimize effort to change and improve design by requirement changes.	Team members will have meetings with supervisor to analyze the requirement change, and make a specific action to resolve that problem.	High
3	Delivery does not meet the deadline.	Team member work late, lack of responsible causing not	High	Assign tasks to suitable team members depending on their	Team leader should regularly care about deadline and	Medium

		meet project schedule.		strengths and weakness.	project schedule to inform members. Break down task to smaller tasks and assign new tasks to team members.	
4	Lack of knowledge about technology needs for project.	With some member, they want to study new technology through this project.	Medium	Each team member must study to understand the technology, framework, which is needed for project.	Technical leader need support carefully for team members in group to develop required skills and knowledge. Review complete task to make comment for team member.	Medium
5	Conflict among team members.	Team member doesn't understand each other.	Medium	Plan some games or team building activities to improve teammate's relationship and to understand each other. Clear role and responsibility between members.	Have a meeting to resolve conflict problems.	High
6	Data or Source Code Lost.	Delete Brand in GitHub. Delete wrong source code.	Medium	Back up source code carefully. Commit all small tasks after complete to GitHub.	Restore data from backed up data. Restore source code from history versions.	High

Table 2.8 Risk Management

2.9 COMMUNICATION MANAGEMENT コミュニケーション管理

2.9.1 Communication between Team Members チームメンバーの間のコミュニケーション

- **Face-to-face meeting:** at least twice a week at Monday and Thursday. This is the fastest way of communication to solve big problems in the project and members can help out each other easily.
- **E-mail and message:** Email, Skype and Facebook are used for members to keep tracking other team member's progress and team members can also help each other online.
- **Mobile Phone:** is using for emergency situation to directly contact to others.
- **Collaboration tool:** GitHub is used for document and source code management.

2.9.2 Communication with Supervisor 指導教員とコミュニケーション

- **Face-to-face meeting:** Weekly on every Thursday afternoons to make sure that supervisor can keep tracking of the team's progress.
- **E-mail:** Gmail is the fastest way to get advice and document checking from supervisor.
- **Mobile phone:** is used to get time and place arranged for the meeting every weeks.

2.10 CONFIGURATION MANAGEMENT PROCESS コンフィグレーション管理

2.10.1 CI Identification and Naming Convention 構成アイテムと命名規則

No	Deliverable	Committed Delivery date	Description of Deliverable	Delivery media
1	Capstone project register	September 5 th , 2016		Hard copy
Iteration 1				
2	Software Requirement Specifications - SRS	September 16 th , 2016	Document	Commit to Github
3	Q&A Management	September 26 th , 2016	Document	Commit to Github
4	User requirements	September 16 th , 2016	Document	Commit to Github
5	Project plan	September 09 th , 2016	Document	Commit to Github
6	Progress report 1	September 08 th , 2016	Document	Commit to Github

7	Architecture design	October 03 th , 2016	Document	Commit to Github
8	User interface design	October 11 th , 2016	Document	Commit to Github
9	Progress report 2	September 13 th , 2016	Document	Commit to Github
10	Database design	October 07 th , 2016	Document	Commit to Github
11	Test plan	November 08 th , 2016	Document	Commit to Github
12	Test case	November 14 th , 2016	Document	Commit to Github
13	Progress report 3	September 20 th , 2016	Document	Commit to Github
14	Progress report 4	October 05 th , 2016	Document	Commit to Github
15	Test case - execute	November 14 th , 2016	Document	Commit to Github
16	Test report	November 15 st , 2016	Document	Commit to Github
17	Progress report 5	November 17 th , 2016	Document	Commit to Github
Iteration 2				
18	User manual	December 14 th , 2016	Document	Commit to Github
19	Progress report 6	December 14 th , 2016	Document	Commit to Github
20	Final report	December 14 th , 2016	Document	Hard copy

Table 2.10.1 Naming Convention

2.10.2 Project Infrastructure プロジェクトインフラストラクチャ

Below is the list of tools and infrastructure requirements needed for development environment.

2.10.2.1 *Software* ソフトウェア

Programming languages	PHP, HTML, CSS, Javascript
Framework	Laravel 5.3, Bootstrap v3.3.7
Version Control	Github
IDEs	PHPStorm v8.0.1, SublimeText 3
DBMS	MySQL v5.7, MySQL Workbench v6.3.7
Deployment tools	Heroku
Communication tools	Trello, Skype, Facebook, Phone, Email.
UML tools	Astah Professional 7.0
Management tools	Microsoft Project Professional 2013
Web server	Xamp v7.0.8
Other	Microsoft Office 2013

Table 2.10.2 Tools

2.10.2.2 *Hardware* ハードウェア

- ❖ Personal computer for developing and testing with the configuration: 4GB RAM, 256GB of hard disk, Intel Core i5 Processor.
- ❖ The smart phone and tablet for testing.
- ❖ Internet network connection with minimum speech 512KByte/s.

2.10.2.3 *Other infrastructure* その他のインフラストラクチャ

- ❖ Rooms for meeting and working.
- ❖ Internet and mobile phone services for communication.

2.10.3 Directory structure ディレクトリ構造

Main folder	Sub-folder	Purpose
Documents	Meeting minutes	Store project meeting minutes
	Q&A	Store QA Management Sheet version English and version Japanese
	Report	Store project reports
Final	Report 1	Store final deliverables of phase 1

deliverable	Report 2	Store final deliverables of phase 2
	Report 3	Store final deliverables of phase 3
	Report 4	Store final deliverables of phase 4
	Report 5	Store final deliverables of phase 5
	Report 6	Store final deliverables of phase 6
Plan		Store project plan, Task list
Resource	Reference materials	Store Reference materials needed in project
	Template	Store template needed in project
	Tool	Store tool needed in project
Working space	Each team members has a folder. (Example: Them)	Team member's working area
Archive	Baseline Name	To released versions of CIs at baselines

Table 2.10.3 Directory structure

2.10.4 Coding convention コーディング規約

The following rules follow the standard rule of developing applications using PHP. It refers to the coding convention on websites:

https://www.codeigniter.com/user_guide/general/styleguide.html

File Format

Files should be saved with Unicode (UTF-8) encoding. The BOM should not be used. Unlike UTF-16 and UTF-32, there's no byte order to indicate in a UTF-8 encoded file, and the BOM can have a negative side effect in PHP of sending output, preventing the application from being able to set its own headers. Unix line endings should be used (LF).

Here is how to apply these settings in some of the more common text editors. Instructions for your text editor may vary; check your text editor's documentation.

PHP Closing Tag

The PHP closing tag ?> is optional to the PHP parser. However, if used, any whitespace following the closing tag, whether introduced by the developer, user, or an FTP application, can cause unwanted output, PHP errors, or if the latter are suppressed, blank pages. For this reason, all PHP files MUST OMIT the PHP closing tag and end with a single empty line instead.

File Naming

Class files must be named in an Ucfirst-like manner, while any other file name (configurations, views, generic scripts, etc.) should be in all lowercase.

INCORRECT:

- somelibrary.php
- applicationConfig.php

CORRECT:

- Somelibary.php
- applicationconfig.php

Furthermore, class file names should match the name of the class itself. For example, if you have a class named Myclass, then its filename must be Myclass.php.

Class and Method Naming

Class names should always start with an uppercase letter. Multiple words should be separated with an underscore, and not CamelCased.

INCORRECT:

- class superclass

- class SuperClass

CORRECT:

- class Super_class {}

Class methods should be entirely lowercased and named to clearly indicate their function, preferably including a verb. Try to avoid overly long and verbose names. Multiple words should be separated with an underscore.

INCORRECT:

- function fileproperties() // not descriptive and needs underscore separator
- function fileProperties() // not descriptive and uses CamelCase
- function getfileproperties() // Better! But still missing underscore separator
- function getFileProperties() // uses CamelCase

CORRECT:

- function get_file_properties() // descriptive, underscore separator, and all lowercase letters

Variable Names

The guidelines for variable naming are very similar to those used for class methods. Variables should contain only lowercase letters, use underscore separators, and be reasonably named to indicate their purpose and contents. Very short, non-word variables should only be used as iterators in for() loops.

INCORRECT:

- \$j = 'foo'; // single letter variables should only be used in for() loops
- \$Str // contains uppercase letters
- \$bufferedText // uses CamelCasing, and could be shortened without losing semantic meaning
- \$groupid // multiple words, needs underscore separator
- \$name_of_last_city_used // too long

CORRECT:

- for (\$j = 0; \$j < 10; \$j++)
- \$str
- \$buffer
- \$group_id
- \$last_city

Commenting

In general, code should be commented prolifically. It not only helps describe the flow and intent of the code for less experienced programmers, but can prove invaluable when returning to your own

code months down the line. There is not a required format for comments, but the following are recommended.

Comment code block:

```
/*
 * @author Author Name
 * @link http://example.com
 */
```

Use single line comments within code, leaving a blank line between large comment blocks and code:

- // break up the string by newlines
- // \$parts = explode("\n", \$str);

Constants

Constants follow the same guidelines as do variables, except constants should always be fully uppercase. Always use CodeIgniter constants when appropriate, i.e. SLASH, LD, RD, PATH_CACHE, etc.

INCORRECT:

- myConstant // missing underscore separator and not fully uppercase
- N // no single-letter constants
- S_C_VER // not descriptive
- \$str = str_replace('{foo}', 'bar', \$str); // should use LD and RD constants

CORRECT:

- MY_CONSTANT
- NEWLINE
- SUPER_CLASS_VERSION
- \$str = str_replace(LD.'foo'.RD, 'bar', \$str);

3 SOFTWARE REQUIREMENTS SPECIFICATION ソフトウェア 要件仕様

3.1 INTRODUCTION はじめに

3.1.1 Purpose 目的

This document is created as the introduction for project JFS – our Capstone Project at FPT University. In this document, 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 Scope 範囲

JFS is a system that build on web platform to create the connect between Employers and Job Seekers by provide all necessary information that all Employers and Job Seekers need, and help them to find each other in a fastest way. To do this, we build our system a special feature to match between Employers and Job Seekers based on their provided needing information, and allow them to connect to others directly.

3.1.3 References 参照資料

No.	Document	Source/Web Address
1	System Requirements Specification Template	FPT Software Cop.

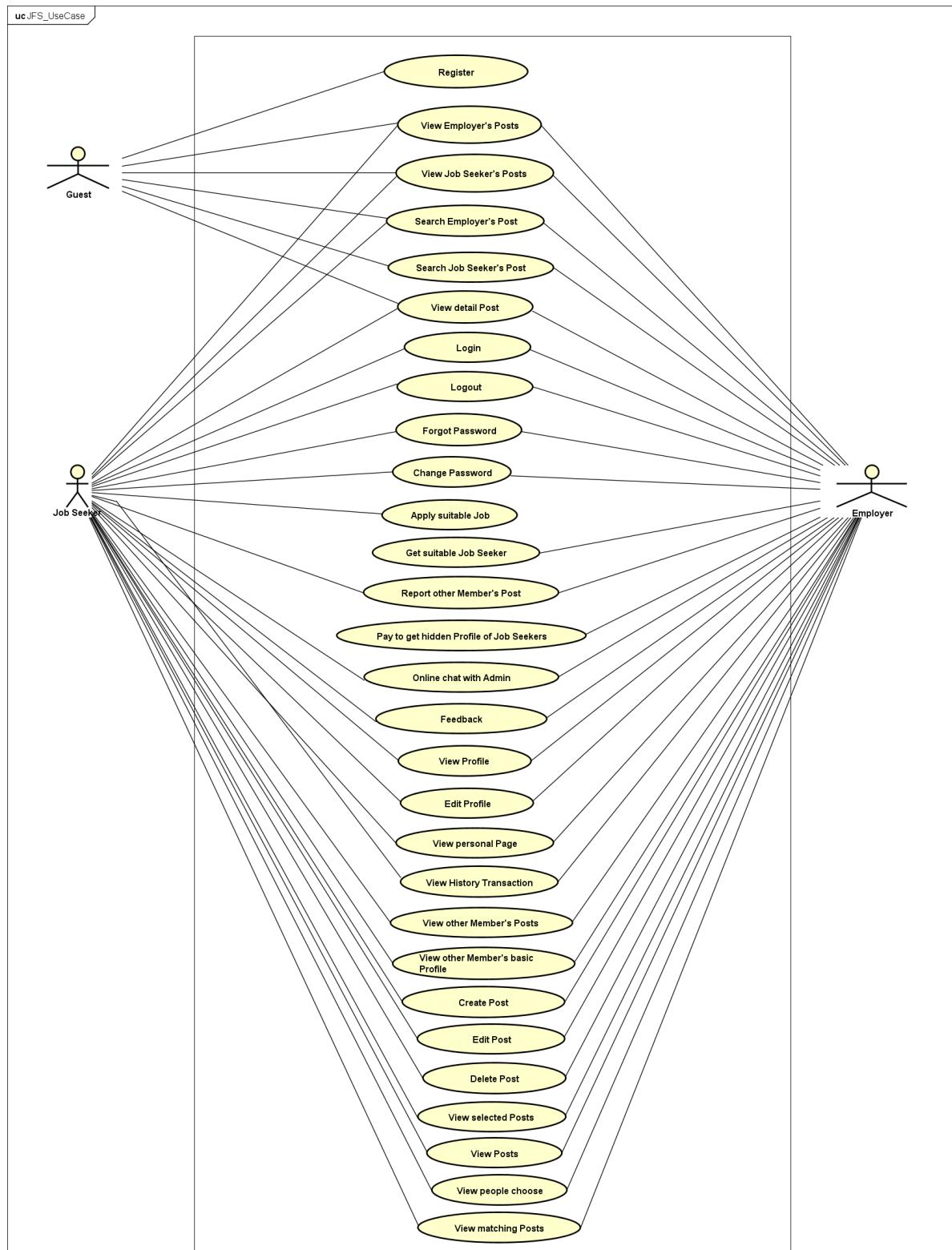
3.1.4 Overview 概要

The rest of this document contains 4 sections:

- **Introduction:** provides an overview of this document. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview.
- **Overall Description:** describes the general factors that affect to the product and their requirements.
- **Specific Requirement:** Contains all requirements to a level of detail sufficient to help designers and testers satisfy those requirements.
- **Supporting Information:** Provide some other reference information that can help people understand easier.

3.2 OVERALL DESCRIPTION 概要説明

3.2.1 Use Case Diagram ユースケース図



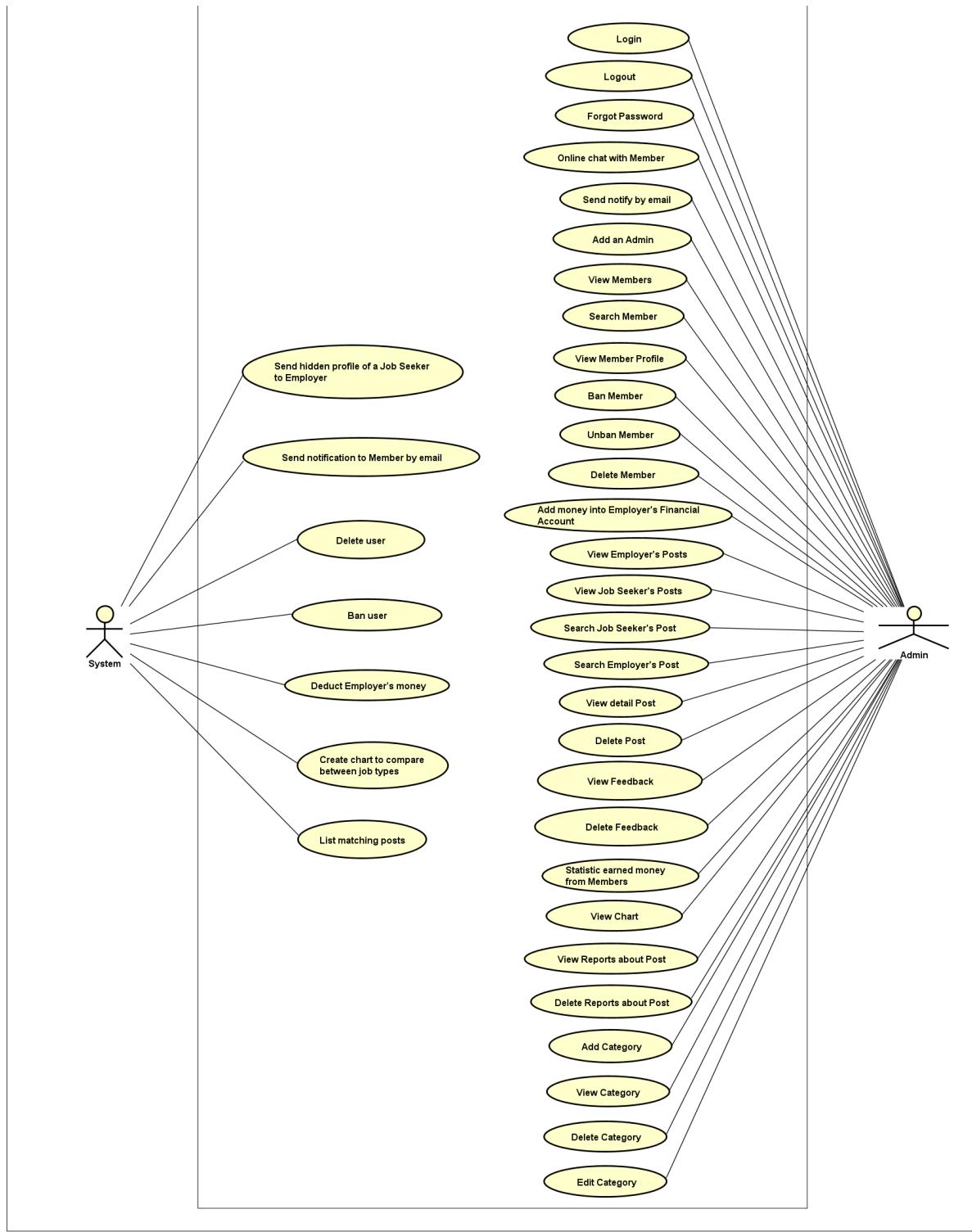


Figure 3.2 List Use Case of JFS

3.3 PRODUCT FEATURES 製品の特徴

UC No.	Group Of Functions	Function	Glossary
Guest			
UC001	Guest	Register	
UC002		View Employer's Posts	
UC003		View Job Seeker's Posts	
UC004		Search Employer's Posts	
UC005		Search Job Seeker's Posts	
UC006		View detail Post	
Member			
UC007	Common	Login	
UC008		Logout	
UC009		Forgot password	
UC0010		Change password	
UC0011		Search Employer's Posts	
UC0012		Search Job Seeker's Posts	
UC0013		View Employer's Posts	
UC0014		View Job Seeker's Posts	
UC0015		Apply suitable Job	
UC0016		Get suitable Job Seeker	
UC0017		Report other Member's Post	

UC0018		Payment to get hidden Profile of Job Seekers	
UC0019		Online chat with Admin	
UC0020		Feedback	
UC0021	Manage Personal Page	View Profile	
UC0022		Edit Profile	
UC0023		View personal Page	
UC0024		View History Transaction	
UC0025	View other Member's Personal Page	View other Member's Posts	
UC0026		View other Member's basic Profile	
UC0027	Manage Post	Create Post	
UC0028		Edit Post	
UC0029		Delete Post	
UC0030		View selected Posts	
UC0031		View Posts	
UC0032		View detail Post	
UC0033		View people choose	
UC0034		View matching Posts	
Administrator			
UC0035	Common	Admin Login	
UC0036		Admin Logout	

UC0037		Admin Forgot password	
UC0038		Online chat with Member	
UC0039		Send notify by email	
UC0040		Add a Admin	
UC0041	Manage Members	View Members	
UC0042		Search Members	
UC0043		View Member Profile	
UC0044		Ban Member	
UC0045		Unban Member	
UC0046		Delete Member	
UC0047		Add money into Employer's Account	
UC0048	Manage Posts	View Employer's Posts	
UC0049		View Job Seeker's Posts	
UC0050		Search Employer's Posts	
UC0051		Search Job Seeker's Posts	
UC0052		Delete Post	
UC0053		View detail Post	
UC0054	Manage feedback of member about website	View feedback	
UC0055		Delete feedback	
UC0056	Statistic	Statistic earned money from	

		Members	
UC0057		View Chart	
UC0058	Manage Reports	View Reports about Post	
UC0059		Delete Reports about Post	
UC0060	Manage Category	Add Category	
UC0061		View Category	
UC0062		Delete Category	
UC0063		Edit Category	
System			
UC0064		Send hidden profile of a Job Seeker to Employer	
UC0065		Send notification to Member's email	
UC0066		Delete Member	
UC0067		Ban Member	
UC0068		Deduct Employer's money	
UC0069		Create chart to compare between job types	
UC0070		List matching posts	

Table 3.3: List of use case diagram

3.3.1 Use Characteristics

Job Finding System towards to user who is interested in the information of part time job, it includes of finding jobs and finding people who can apply any job. By using Job Finding System, everyone will accept system's operation and rules of content and financial. Our system has 3 groups of objective user:

- **Guest:** User who first time access to website, or who don't have account. Guest can use some specific functions: Register, Search Posts, View Posts (Guest can only view some limited content of posts).
- **Member:** Member are all user that already had an account. There are 2 type of Member: Job Seeker and Employer. This 2 types of user are primary types of user system. Both Employer and Job Seeker can use almost main functions of system. Some of that main functions are: View Post, Create Post, Edit Post, View Profile, Edit Profile. Besides, each type of user can use difference specific functions:
 - + **Job Seeker:** Apply Job.
 - + **Employer:** Get hidden Profile of Job Seeker, Get Job Seeker for doing Job.
- **Administrator:** Manage Website's operation (Posts, Users) by use some specific functions: Manage comments and reports, Manage User, Manage Posts.

3.3.2 User Documentation ユーザードキュメンテーション

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

Table 3.3.2: Use Documentation

3.3.3 Assumption 仮定

- Receive the guidance and good support of teacher.
- All member attend in the software development process.
- In the development process no device got fail.
- All task meet the deadline.
- Every one self-study new technology.
- Team member support each other in the work process.

3.4 ENTITY RELATION MODEL エンティティ関係モデル

3.4.1 Entity-Relationship Diagram エンティティ-関係図

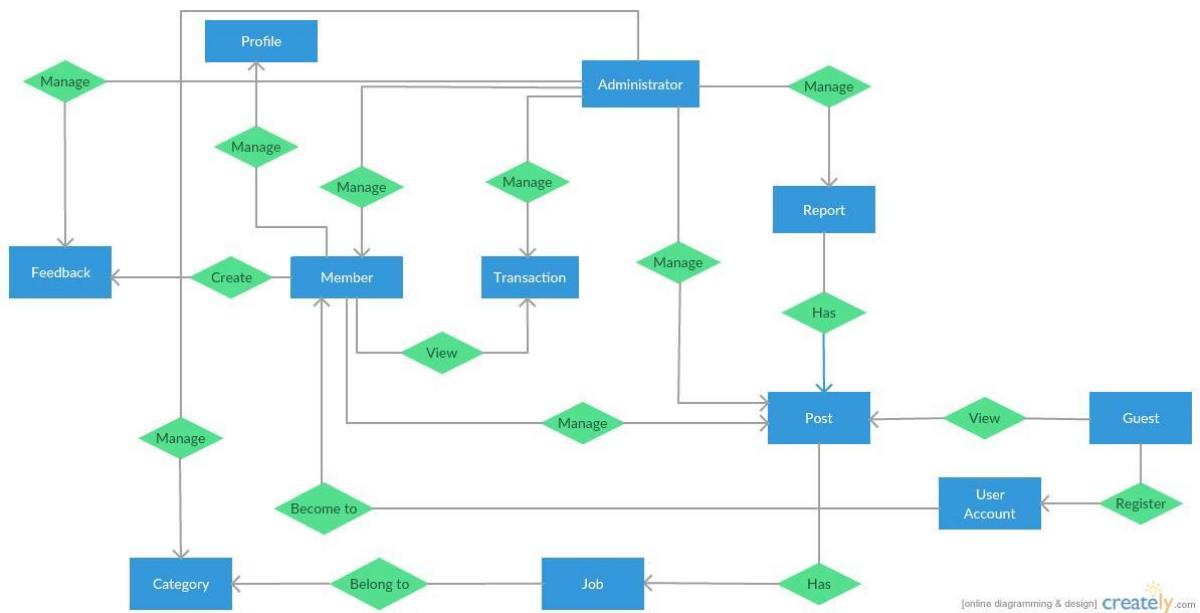


Figure 3.4.1: Entity Relationship Diagram

3.4.1.1 *Admin*

No	Attribute	Type	Mandatory	Description
1	AdminID	Int	Y	Admin's ID
2	Username	String	Y	Admin's name
3	Password	String	Y	Admin's password
4	Email	String	Y	Admin's email
5	Address	String	Y	Admin's address
6	PhoneNumber	Number	Y	Admin's phone number
7	DateOfBirth	Date	Y	Admin's date of birth
8	Gender	String		Admin's gender
9	Picture	Mediumblob	Y	Admin's profile picture
10	RememberToken	Boolean	Y	0: Don't remember account 1: Remember account

Table 3.4.1.1: Admin

3.4.1.2 *Profile*

No	Attribute	Type	Mandatory	Description
1	MemberID	Int	Y	Member's ID
2	FullName	String	Y	Member's full name.
3	Email	String	Y	Member's email
4	Address	String	Y	Member's address
5	Phone_Number	Number	Y	Member's phone number
6	DateOfBirth	Date	Y	Member's date of birth
7	Gender	String		Member's gender
8	Picture	Mediumblob	Y	Member's profile picture
9	Description	Text	Y	Member's personal description

Table 3.4.1.2: Member

3.4.1.3 *Report*

No	Attribute	Type	Mandatory	Description
1	ReportID	Int	Y	Report's ID
2	MemberID	Int	Y	Member's ID who create report
3	PostID	int	Y	Post's ID that post is reported
4	Time	Timestamp	Y	Real time that member create report
5	Content_Report	Text	Y	Report's content

Table 3.4.1.3: Report

3.4.1.4 *Post*

No	Attribute	Type	Mandatory	Description
1	PostID	Int	Y	Post's ID
2	MemberID	Int	Y	Member's ID who create post
3	Type	Boolean	Y	0: Employer's post 1: Job Seeker's post
4	Time_Creat	Datetime	Y	Time post was created
5	Subject	String	Y	Subject of a post
6	Post_Status	Boolean	Y	Status of a post 0: On ; 1: Off

Table 3.4.1.4: Post

3.4.1.5 *Category*

No	Attribute	Type	Mandatory	Description
1	CategoryID	Int	Y	Category's ID
2	CategoryName	String	Y	Category's name
3	Times	Int	Y	Number that a category is selected

Table 3.4.1.5: Category

3.4.1.6 *Job*

No	Attribute	Type	Mandatory	Description
1	JobID	Int	Y	Job's ID
2	PostID	Int	Y	Post's ID post has job
3	Job_Description	Text	Y	Description of a job

Table 3.4.1.6: Job

3.4.1.7 *Feedback*

No	Attribute	Type	Mandatory	Description
1	FeedbackID	Int	Y	Feedback's ID
2	MemberID	Int	Y	Member's ID who create feedback
3	Created_at	Timestamp	Y	Real time that member create feedback
4	Content_feedback	Text	Y	Feedback's content

Table 3.4.1.7: Feedback

3.4.1.8 *Transaction*

No	Attribute	Type	Mandatory	Description
1	TransactionID	Int	Y	Transaction's ID
2	PostID	Int	Y	Post's ID
3	EmployerID	Int	Y	Member's ID who is employer executed transaction
4	JobSeekerID	Int	Y	Member's ID who is jobseeker
5	Time	Timestamp	Y	Time that transaction was implemented
6	Money	Float	Y	Money that Employer have to paid

Table 3.4.1.8: Transaction

3.4.1.9 *Member*

No	Attribute	Type	Mandatory	Description
1	MemberID	Int	Y	Transaction's ID
2	MemberName	String	Y	Member's name
3	Password	String	Y	Member's password
4	Money	Float	Y	Member's money

5	Remember_Token	Boolean	Y	false: Don't remember account true: Remember account
6	IsActive	Int	Y	Member's active status (0: inactive,1: active)

Table 3.4.1.9: Member

3.4.2 Entity description エンティティの説明

No	Table Name	Description
1	Administrator	Manage JFS.
2	Profile	The representation of personal information of a Member.
3	Report	The message which Member send to system when they want to notify of Post's content.
4	Post	The information entries about Job which Member post on system.
5	Category	Category of Job.
6	Job	A job that job seeker want to work.
7	Feedback	A feedback which Member send to system when they want to comment or notify of Website's activity and performance.
8	Transaction	A transaction between employer and job seeker after employer pay money to get job seeker hidden profile
9	Member	User who have account.
10	Member account	Account of Member include Username and Password to login into system
11	Guest	User who access website without log in.

Table 3.4.2 Entity Description

3.4.3 Relationship 関係

No	Relation	Properties	From	To	Description
1	MANAGE	CategoryID	Administrator	Category	Administrator manages

					category.
2	MANAGE	FeedbackID	Administrator	Feedback	Administrator manages all feedback.
3	MANAGE	MemberID	Administrator	Member	Administrator manages all members.
4	MANAGE	TransactionID	Administrator	Transaction	Administrator manages all transactions.
5	MANAGE	ReportID	Administrator	Report	Administrator manages all reports.
6	MANAGE	PostID	Administrator	Post	Administrator manages all posts.
7	MANAGE	ProfileID MemberID	Member	Profile	A member manages his own profile.
8	MANAGE	PostID MemberID	Member	Post	A member manages his own posts.
9	CREATE	MemberID	Member	Feedback	A member creates many feedback.
10	HAS	None	Post	Job	A post has a job.
11	HAS	None	Report	Post	A post has 1 or many posts.
12	BECOME TO	MemberID	Member Account	Member	Member account become to member.
13	BELONG TO	JobID CategoryID	Job	Category	A job belong to a category.
14	VIEW	None	Guest	Post	A guest can view many posts.
15	VIEW	None	Member	Transaction	A member can view transaction.
16	REGISTER	None	Guest	Member Account	Guest register to member account.

Table 3.4.3 Relationship

3.5 SPECIFIC REQUIREMENTS 要求仕様

3.5.1 Business Rules ビジネスルール

No	Description
B1	Search textbox cannot be empty
B2	Text field cannot be empty
B3	Content of post is limited to 1000 characters
B4	Chat box cannot be empty
B5	If user accesses to nonexistent page, JFS displays error message "Page not found"
B6	Email must characters '@' and '.'content
B7	Date of birth must be chosen
B8	Displayed date is mm/dd/yyyy format
B9	Date time input format is date time
B10	Number is rounded to 2 decimal number
B11	Unit of money is VND
B12	Displayed money is rounded to 2 decimal number
B13	Allowed file extension: .jpg, .png, .bmp
B14	File size is limited to 100 MB
B15	Message displayed like a Gmail format

Table 3.5: Business Rules

3.5.2 Functionality 機能要求

3.5.2.1 Guest ゲスト

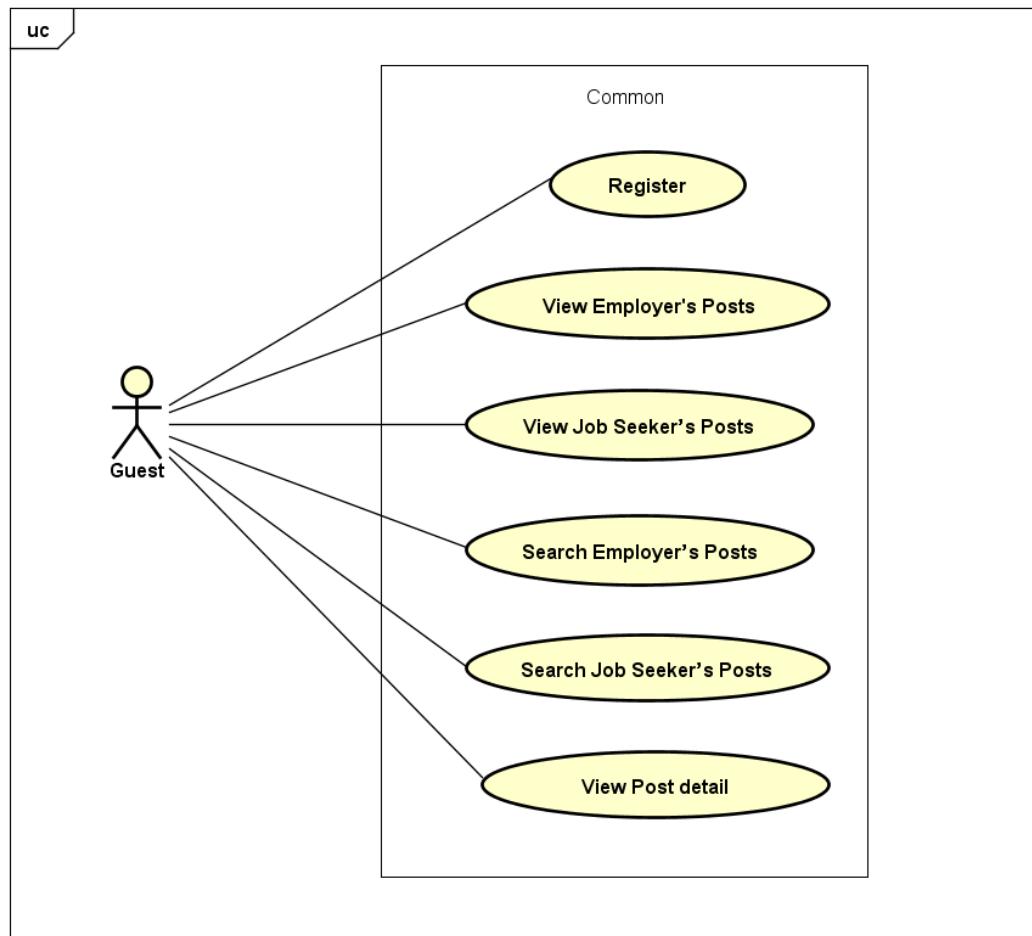


Figure 3.5.2 Guest of common use case

3.5.2.1.1 UC001 – Register

Use Case ID	UC001		Version	1.0
Use Case Name	Register			
Created by:	ThuyTT	Date Created:	13/09/2016	
Primary Actor	Guest	Secondary Actor		
Description:	Guest can create new account to use more function as member			
Preconditions:	1. JFS must be connected to the Internet 2. User browsed JFS website			
Trigger:	Click on “Đăng kí” button on the website			

Post conditions:		<ol style="list-style-type: none"> 1. Add account information to database 2. Send confirm email 3. Logs user into system 4. Redirect user to Homepage and display as Member
Normal Flow		
Step	Actor	Action
1	Guest	Click “Đăng kí” button in header
2	JFS	<p>Display Register page includes:</p> <ul style="list-style-type: none"> - Header - “Tên Đăng Nhập” text box - “Họ và Tên” text box - “Mật khẩu” text box - “Xác nhận mật khẩu” text box - “Ngày sinh” date picker - “Giới tính” checkbox - “Địa chỉ” textbox - Email text box - “Số Điện thoại” text box - “Đăng kí” button - “Hủy” button
3	Guest	Enters data in fields
4	Guest	Click “Đăng Kí” button
5	JFS	Send confirm email
6	JFS	Logs user into system
7	JFS	Display Homepage as Member
Alternative Flows		
AT1	At step 4, user click “Hủy” button	
Step	Actor	Action
4.1	JFS	Redirects user to previous page
AT2	At step 4 in the main flows, if user click other hyperlinks or buttons	

Step	Actor	Action
4.1	JFS	Redirects user to chosen hyperlink or button
Exceptions:		
EC1 At step 4, required fields are not entered		
Step	Actor	Action
4.1	JFS	Display Register page with message
4.2	JFS	Mark error fields
EC2	At step 4, fields are entered with wrong type of data	
Step	Actor	Action
4.1	JFS	Display Register page with message
4.2	JFS	Mark error fields
Priority		High
Frequency of Use:		High
Business Rules:		B2, B6, B7
Other Information:		N/A

Table 3.5.2.1.1: Register Use case

3.5.2.1.2 UC002 – View Employer's Posts

Use Case ID	UC002	Version	1.0
Use Case Name	View Employer's Posts		
Created by:	ThuyTT	Date Created:	13/09/2016
Primary Actor	Guest	Secondary Actor	
Description:	This function allows Guest view Employer's Posts		

Preconditions:	1. JFS website is available 2. User browsed JFS website	
Trigger:	Guest browsed JFS website	
Post conditions:	Employer's Posts is displayed	
Normal Flow		
Step	Actor	Action
1	Guest	Guest browsed JFS website
2	JFS	Display Employer's posts in home page includes: - Home - Sign in & Sign up - List of Employer's Posts
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	B3	
Other Information:	N/A	

Table 3.5.2.1.2: Guest view Employer's Posts Use case

3.5.2.1.3 UC003 – View Job Seeker's Posts

Use Case ID	UC003	Version	1.0
Use Case Name	View Job Seeker's Posts		
Created by:	ThuyTT	Date Created	13/09/2016
Primary Actor	Guest	Secondary Actor	
Description:	This function allows Guest view Job Seeker's Posts		
Preconditions:	1. JFS website is available 2. User browsed JFS website		
Trigger:	Guest browsed JFS website		
Post conditions:	Job Seeker's Posts is displayed		

Normal Flow		
Step	Actor	Action
1	Guest	Guest browsed JFS website
2	JFS	Display Job Seeker's posts in home page includes: - Home - Sign in & Sign up - List of Job Seeker's Posts
Alternative Flows: N/A		
Exceptions: N/A		
Priority		High
Frequency of Use:		High
Business Rules:		B3
Other Information:		N/A

Table 3.5.2.1.3: Guest view Job Seeker's Posts Use case

3.5.2.1.4 UC004 - Search Employer's Posts

Use Case ID	UC004	Version	1.0
Use Case Name	Search Employer's Posts		
Created by:	ThuyTT	Date Created:	13/09/2016
Primary Actor	Guest	Secondary Actor	
Description:	This function allows Guest to search information about Employer's Posts on JFS by category, location or any keyword		
Preconditions:	1.JFS website is available 2.User browsed JFS website		
Trigger:	Guest enters search phrase or choose value in dropdown list.		
Post conditions:	List all results that matched with the keyword Guest entered will be displayed in the Search Details screen		
Normal Flow			

Step	Actor	Action
1	Guest	Enter search phrase to text box or choose value in dropdown
2	JFS	List all results that matched with the keyword Guest entered
Alternative Flows:		
AT1	At step 1 in the main flows, if user click other hyperlinks or buttons	
Step	Actor	Action
2.1	JFS	Redirects user to chosen hyperlink or button
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	B1	
Other Information:	N/A	

Table 3.5.2.1.4: Guest search post Use case

3.5.2.1.5 UC005 – Search Job Seeker’s Posts

Use Case ID	UC005	Version	1.0
Use Case Name	Search Job Seeker’s Posts		
Created by:	ThuyTT	Date Created:	13/09/2016
Primary Actor	Guest	Secondary Actor	
Description:	This function allows Guest to search information about Job Seeker’s Posts on JFS by category, location or any keyword		
Preconditions:	1. JFS website is available 2. User browsed JFS website		
Trigger:	Guest enters search phrase or choose value in dropdown list		
Post conditions:	List all results that matched with the keyword Guest entered will be displayed in the Search Details screen		
Normal Flow			

Step	Actor	Action
1	Guest	Enter search phrase to text box or choose value in dropdown
2	JFS	List all results that matched with the keyword Guest entered
Alternative Flows:		
AT1	At step 1 in the main flows, if user click other hyperlinks or buttons	
Step	Actor	Action
2.1	JFS	Redirects user to chosen hyperlink or button
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	B1	
Other Information:	N/A	

Table 3.5.2.1.5: Guest view Job Seeker's Posts Use case

3.5.2.1.6 UC006 – View detail post

Use Case ID	UC006		Version	1.0		
Use Case Name	View detail post					
Created by:	ThuyTT	Date Created:	13/09/2016			
Primary Actor	Guest	Secondary Actor				
Description:	This function allows Guest to view detail information of post					
Preconditions:	1. JFS website is available 2. Guest browsed JFS website					
Trigger:	Guest clicks on post					
Post conditions:	Common information of this post is displayed					
Normal Flow						

Step	Actor	Action
1	Guest	Click on post on home page
2	JFS	Display detail post page includes: - Content of post
Alternative Flows: N/A		
Exceptions: N/A		
Priority		High
Frequency of Use:		High
Business Rules:		B3, B8
Other Information:		N/A

Table 3.5.2.1.6: Guest view detail post Use case

3.5.2.2 *Member メンバー*

3.5.2.2.1 Common

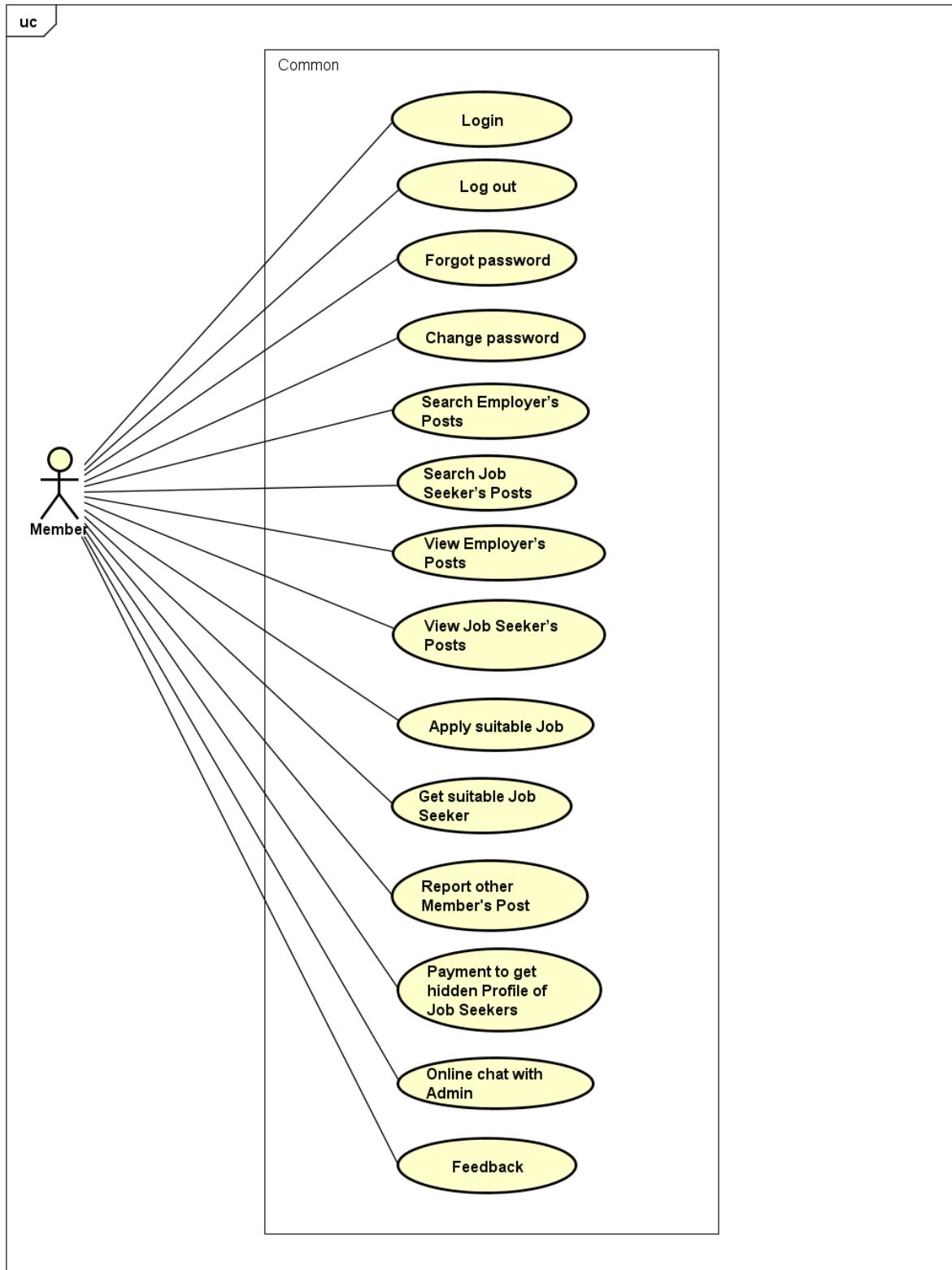


Figure 3.5.2.2.1 Member's common use case

3.5.2.2.1 UC007 – Member login

Use Case ID		UC007	Version	1.0
Use Case Name		Login		
Created by:		ThuyTT	Date Created:	13/09/2016
Primary Actor		Member	Secondary Actor	
Description:		When Member want to login by registered account to use more functions in website		
Preconditions:		1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Account register must be exist and correctly 4. Guest is viewing Login page		
Trigger:		Member click “Đăng nhập” button		
Post conditions:		1. Log Member into system 2. Redirect user to previous page and display as Member		
Normal Flow				
Step	Actor	Action		
1	User	Click “Đăng nhập” in Homepage		
2	JFS	Display Login popup includes: - “Tên tài khoản” text box - “Mật khẩu” text box - “Duy trì đăng nhập” checkbox - “Đăng nhập” button - “Hủy” button - “Quên mật khẩu” hyperlink		
3	User	Enters User name and Password		
4	User	Click “Đăng nhập” button		
5	JFS	Logs user into system		
6	JFS	Close popup		
7	JFS	Display home page as Member		
Alternative Flows				

AT1	At step 4, user click “Hủy” button	
Step	Actor	Action
4.1	JFS	Closes popup.
Exceptions:		
EC1	At step 3 in main flow, if user entered wrong User name or Password	
Step	Actor	Action
3.1	JFS	Display Login popup error message
3.2	JFS	Mark error fields
Priority	High	
Frequency of Use:	High	
Business Rules:	B2	
Other Information:	N/A	

Table 3.5.2.2.1.1 Member Login

3.5.2.2.1.2 UC008 – Logout

Use Case ID	UC008	Version	1.0
Use Case Name	Logout		
Created by:	ThuyTT	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	This function allows a member to logout.		
Preconditions:	1. JFS must be connected to the Internet 2. User browsed JFS website 3. User logged in to system		
Trigger:	User click “Đăng xuất” button in header		
Post conditions:	1. Log Member out from system. 2. Redirect user to home page.		

Normal Flow		
Step	Actor	Action
1	Member	Click “Đăng xuất” button
2	JFS	Display Homepage as Guest
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.2.1.2 Member Logout use case

3.5.2.2.1.3 UC009 – Forgot password

Use Case ID	UC009	Version	1.0
Use Case Name	Forgot password		
Created by:	ThuyTT	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	When Member forgot their password to login the system, this function will receiving their password to email		
Preconditions:	1. JFS must be connected to the Internet 2. User browsed JFS website 3. Member’s account is already created.		
Trigger:	Click “Quên mật khẩu” link in login popup		
Post conditions:	1. Send password to confirm email.		
Normal Flow			
Step	Actor	Action	
1	Member	Click “Quên mật khẩu” in login popup	
2	JFS	Display forgot password popup includes:	

		<ul style="list-style-type: none"> - Header - Email text box - “Xác nhận” button - “Hủy” button
3	Member	Enter email to email text box
4	Member	Click “Xác nhận” button
5	JFS	Send password to member's email
6	JFS	Close popup
Alternative Flows		
AT1	At step 4 in the main flow, Member clicks on “Hủy” button	
Step	Actor	Action
4.1	JFS	Return to homepage screen
Exceptions:		
EC1	At step 4, If member enter email that not exist in system's database.	
Step	Actor	Action
4.1	JFS	Displays text with error message
EC2	At step 4, Member enter wrong type email	
Step	Actor	Action
4.2	JFS	Displays text with error message
Priority	High	
Frequency of Use:	Medium	
Business Rules:	B2, B5	
Other Information:	N/A	

Table 3.5.2.2.1.3: Member forgot password Use case

3.5.2.2.1.4 UC010 – Change password

Use Case ID	UC010	Version	1.0
-------------	--------------	---------	-----

Use Case Name		Change password					
Created by:		DuyDD	Date Created:	13/09/2016			
Primary Actor		Member	Secondary Actor				
Description:		Member can change their account's password					
Preconditions:		1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system					
Trigger:		Click on “Thay đổi mật khẩu” link on the website					
Post conditions:		1. Change password successfully 2. Notify change password successfully					
Normal Flow							
Step	Actor	Action					
1	Member	Click on “Thay đổi mật khẩu” link in header					
2	JFS	Display profile popup includes: - “Mật khẩu cũ” text box - “Mật khẩu mới” text box - “Xác nhận mật khẩu” text box - “Xác nhận” button - “Hủy” button					
3	Member	Type data in fields					
4	Member	Click “Xác nhận” button					
5	JFS	Display notify change password successfully.					
Alternative Flows							
AT1	At step 4, user click “Hủy” button						
Step	Actor	Action					
4.1	JFS	Redirects member to previous page					
Exceptions:							

EC1	At step 4, required fields are not entered	
Step	Actor	Action
4.1	JFS	Display Profile page with message: required fields are not entered
4.2	JFS	Mark error fields
EC2	At step 4, fields are entered with wrong type of data	
Step	Actor	Action
4.1	JFS	Display Profile page with message: fields are entered with wrong type of data
4.2	JFS	Mark error fields
Priority	High	
Frequency of Use:	Medium	
Business Rules:	B2	
Other Information:	N/A	

Table 3.5.2.2.1.4 Member change password use case

3.5.2.2.1.5 UC011 – View Employer's Posts

Use Case ID	UC011	Version	1.0
Use Case Name	View Employer's Posts		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	This function allows Member view Employer's Posts		
Preconditions:	1. JFS website is available 2. User browsed JFS website		
Trigger:	Member browsed JFS website		
Post conditions:	Employer's Posts is displayed		
Normal Flow			

Step	Actor	Action
1	Guest	Guest browsed JFS website
2	JFS	Display Employer's posts in home page includes: - Home - Sign in & Sign up - List of Employer's Posts
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	B3	
Other Information:	N/A	

Table 3.5.2.2.1.5 View Employer's Posts Use case

3.5.2.2.1.6 UC012 – View Job Seeker's Posts

Use Case ID	UC012		Version	1.0		
Use Case Name	View Job Seeker's Posts					
Created by:	DuyDD	Date Created: 13/09/2016				
Primary Actor	Member	Secondary Actor				
Description:	This function allows Member view Job Seeker's Posts					
Preconditions:	1. JFS website is available 2. User browsed JFS website					
Trigger:	Member browsed JFS website					
Post conditions:	Job Seeker's Posts is displayed					
Normal Flow						
Step	Actor	Action				
1	Guest	Member browsed JFS website				
2	JFS	Display Job Seeker's posts in home page includes: - Home - Sign in & Sign up				

	- List of Job Seeker's Posts
Alternative Flows: N/A	
Exceptions: N/A	
Priority	High
Frequency of Use:	High
Business Rules:	B3
Other Information:	N/A

Table 3.5.2.2.1.6 View Job Seeker's Posts Use case

3.5.2.2.1.7 UC0013 – Search Employer's Posts

Use Case ID	UC013	Version	1.0		
Use Case Name	Search Employer's Posts				
Created by:	ThuyTT	Date Created:	13/09/2016		
Primary Actor	Member	Secondary Actor			
Description:	This function allows Member to search information about Employer's Posts on JFS by category, location or any keyword				
Preconditions:	1. JFS website is available 2. User browsed JFS website				
Trigger:	Member enters search phrase or choose value in dropdown list				
Post conditions:	List all results that matched with the keyword Member entered will be displayed in the Search Details screen				
Normal Flow					
Step	Actor	Action			
1	Guest	Enter search phrase to text box or choose value in dropdown			
2	JFS	List all results that matched with the keyword Member entered			
Alternative Flows:					
AT1	At step 1 in the main flows, if user click other hyperlinks or buttons				
Step	Actor	Action			
2.1	JFS	Redirects user to chosen hyperlink or button			

Exceptions: N/A	
Priority	High
Frequency of Use:	High
Business Rules:	B1
Other Information:	N/A

Table 3.5.2.2.1.7: Search Employer's Post

3.5.2.2.1.8 UC014 – Search Job Seeker's Posts

Use Case ID	UC014	Version	1.0
Use Case Name	Search Job Seeker's Posts		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	This function allows Member to search information about Job Seeker's Posts on JFS by category, location or any keyword		
Preconditions:	1. JFS website is available 2. User browsed JFS website		
Trigger:	Member enters search phrase or choose value in dropdown list		
Post conditions:	List all results that matched with the keyword Member entered will be displayed in the Search Details screen		

Normal Flow

Step	Actor	Action
1	Guest	Enter search phrase to text box or choose value in dropdown
2	JFS	List all results that matched with the keyword Member entered

Alternative Flows:

AT1	At step 1 in the main flows, if user click other hyperlinks or buttons	
Step	Actor	Action
2.1	JFS	Redirects user to chosen hyperlink or button

Exceptions: N/A	
Priority	High
Frequency of Use:	High
Business Rules:	B1
Other Information:	N/A

Table 3.5.2.2.1.8 Search Job Seeker's Posts

3.5.2.2.1.9 UC015 – Apply suitable Job

Use Case ID	UC015	Version	1.0
Use Case Name	Apply suitable Job		
Created by:	ThuyTT	Date Created:	15/09/2016
Primary Actor	Member	Secondary Actor	
Description:	This function allows Member can apply a suitable job		
Preconditions:	1. JFS website is available 2. Member browsed JFS website 3. Member logged in to system		
Trigger:	Job Seeker views detail Employer's post then click "Chọn công việc này" button on that post		
Post conditions:	Job Seeker choose employer's post		

Normal Flow

Step	Actor	Action
1	Member	Click on Post
2	JFS	Show detail post includes: - Content of the Post - "Chọn công việc này" button
3	Member	Click "Chọn công việc này" button.
4	JFS	Display popup includes: - Confirm text

		- “Đồng ý” button - “Hủy” button
4	Member	Click Ok button
Alternative Flows : N/A		
Exceptions: N/A		
Priority		High
Frequency of Use:		High
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.2.1.9 Apply suitable Job use case

3.5.2.2.1.10 UC016 – Get suitable Job Seeker

Use Case ID	UC016	Version	1.0
Use Case Name	Get suitable Job Seeker		
Created by:	ThuyTT	Date Created:	15/09/2016
Primary Actor	Member	Secondary Actor	
Description:	This function allows Member can Get suitable Job Seeker		
Preconditions:	1. JFS website is available 2. Member browsed JFS website 3. Member logged in to system		
Trigger:	Employer views detail Job Seeker's post then click “Chọn người này” button on that post		
Post conditions:	Member have an information of member is chosen.		
Normal Flow			
Step	Actor	Action	
1	Member	Click on Job Seeker's Post	
1	JFS	Show detail post includes: - Content of the Post	

		- “Chọn người này” button
2	Member	Click “Chọn người này” button.
3	JFS	Display modal confirm and notify deduct money includes: - confirm text - “Đồng ý” button - “Hủy” button
4	Member	Click “Đồng ý” button
5	JFS	Send information of 2 members to each other
6	JFS	Close modal
Alternative Flows : N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.1.2.2.1.10 Member get suitable Job Seeker Use case

3.5.2.2.1.11 UC017 – Report other member’s post

Use Case ID	UC017	Version	1.0
Use Case Name	Report other member’s post		
Created by:	DuyDD	Date Created:	12/09/2016
Primary Actor	Member	Secondary Actor	
Description:	This function allows member report other member’s post.		
Preconditions:	1. Member must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system 4. Member is on post detail page		
Trigger:	Click on “Báo cáo bài đăng” button on the post detail page		

Post conditions:		1. Send report successful to admin 2. Receive notify report successfully
Normal Flow		
Step	Actor	Action
8	Member	Click “Báo cáo bài đăng” button on the post detail page
9	JFS	Display report modal includes: <ul style="list-style-type: none">- “Nội dung báo cáo” check box- Report content text box- “Gửi” button- “Hủy” button
10	Member	Select or enter report content
11	Member	Click “Gửi” button
12	Member	Recieve notify successfully
Alternative Flows		
AT1	At step 4, member click “Hủy”	
Step	Actor	Action
4.1	JFS	Close popup
Exceptions:		
EC1	At step 4, required fields are not selected	
Step	Actor	Action
4.1	JFS	Display required fields are not selected
4.2	JFS	Mark error fields
Priority		
Priority		High
Frequency of Use:		Low
Business Rules:		N/A

Other Information:	N/A
--------------------	-----

Table 3.5.2.2.1.11 Report other member's post

3.5.2.2.1.12 UC018 – Payment to get hidden Profile of Job Seekers

Use Case ID	UC018		Version	1.0			
Use Case Name	Payment to get hidden Profile of Job Seekers						
Created by:	ThuyTT	Date Created:		13/09/2016			
Primary Actor	Member	Secondary Actor					
Description:	This function allows Member payment to get hidden Profile of Job Seekers						
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system						
Trigger:	Member chooses and views post then click “Chọn” button in the list of other members that have chosen this member’s post.						
Post conditions:	Member payment successfully						
Normal Flow							
Step	Actor	Action					
1	Member	Click on one of their posts on Personal Page.					
2	JFS	Show list of other members that have chosen this member’s post.					
3	Member	Click “Chọn người này” button.					
4	JFS	Display modal include informations: - Confirm text - “Đồng ý” button - “Hủy” button					
5	JFS	Click “Đồng ý” button					
Alternative Flows:							
AL1	At step 4, member click “Hủy” button						
Step	Actor	Action					

4.1	JFS	Close popup
Exceptions:		
EC1	At step 4, member's account is not enough money to pay	
Step	Actor	Action
4.1	JFS	Display popup with message: “Tài khoản của bạn không đủ tiền vui lòng nạp thêm để thực hiện giao dịch này”
Priority		High
Frequency of Use:		Medium
Business Rules:		B11
Other Information:		N/A

Table 3.5.2.2.1.12 Payment to get hidden Profile of Job Seekers use case

3.5.2.2.1.13 UC019 – Online chat with admin

Use Case ID	UC019	Version	1.0
Use Case Name	Online chat with admin		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	Member online chat with admin		
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system		
Trigger:	Click on “Chat online” icon on the website		
Post conditions:	Can send message and receive reply from admin.		
Normal Flow			
Step	Actor	Action	
1	Member	Click on “Chat online” icon on website	

2	JFS	Display small chat screen.
3	Member	Type data in fields
4	Member	Press enter in keyboard
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	Medium	
Business Rules:	B4	
Other Information:	N/A	

Table 3.5.2.2.1.13 Online chat with admin Use Case

3.5.2.2.1.14 UC020 – Feedback

Use Case ID	UC020	Version	1.0
Use Case Name	Feedback		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	Member can send feedback to admin		
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system		
Trigger:	Click on “Góp ý” icon on the website		
Post conditions:	Send feedback successfully		
Normal Flow			
Step	Actor	Action	
1	Member	Click on “Góp ý” icon on website	
2	JFS	Display “Góp ý” page includes: - Feedback description text box - “Hủy” button - “Gửi nhận xét” button	

3	Member	Type feedback content
4	Member	Click “Gửi nhận xét” button
5	JFS	Display notify send successfully
Alternative Flows		
AT1	At step 4, user click “Hủy” button	
Step	Actor	Action
4.1	JFS	Redirects member to previous page
Exceptions:		
EC1	At step 4, required fields are not entered	
Step	Actor	Action
4.1	JFS	Display Profile page with message: required fields are not entered
4.2	JFS	Mark error fields
Priority		High
Frequency of Use:		Medium
Business Rules:		B4
Other Information:		N/A

3.5.2.2.2 Manage Personal Page

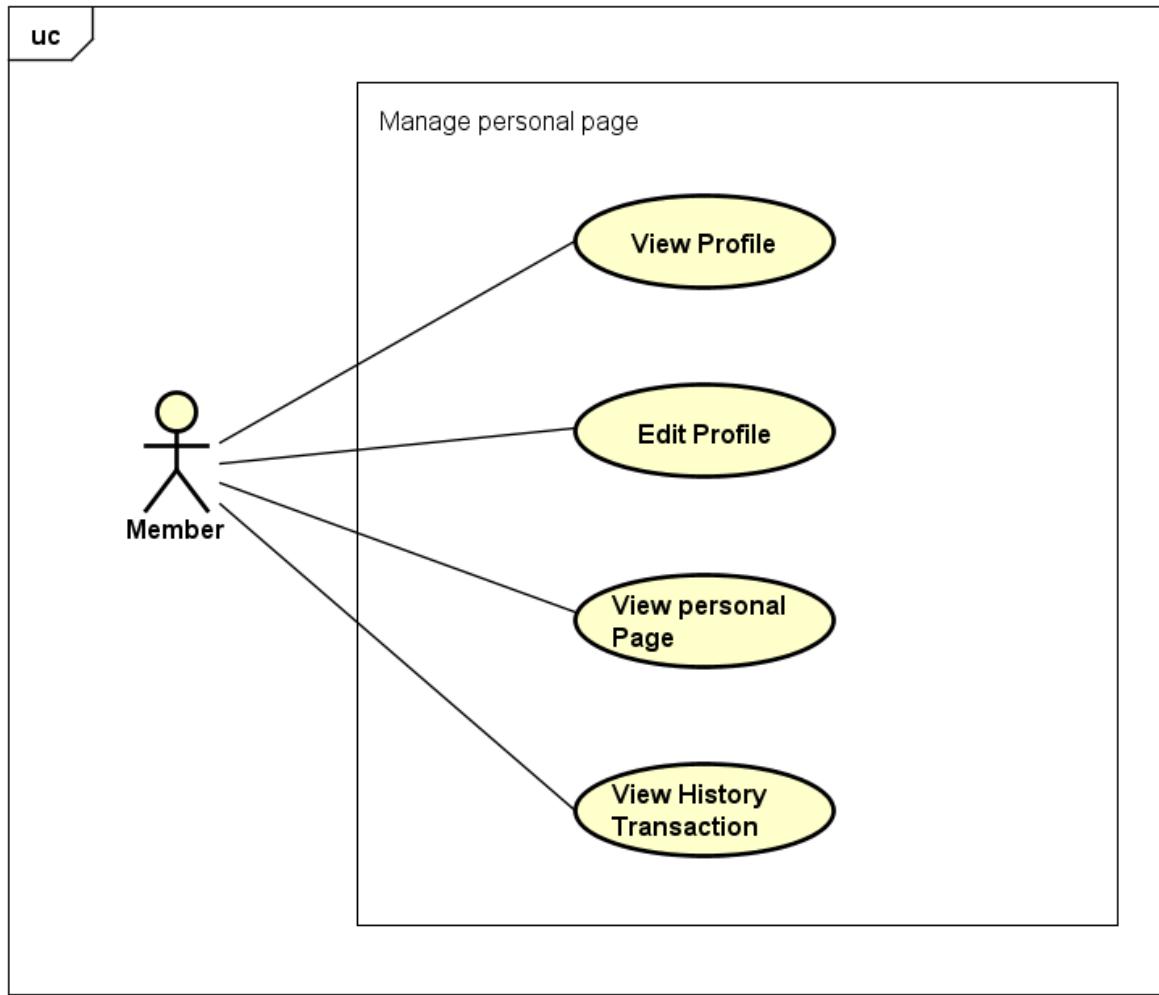


Figure 3.5.2.2.2 Member personal page use case

3.5.2.2.2.1 UC021 – View profile

Use Case ID	UC022		Version	1.0		
Use Case Name	View profile					
Created by:	DuyDD	Date Created:	13/09/2016			
Primary Actor	Member	Secondary Actor				
Description:	Member can view their profile.					
Preconditions:	4. JFS must be connected to the Internet 5. Member browsed JFS website					

		6. Member logged in to system
Trigger:		Click on “Thông tin cá nhân” link in personal page
Post conditions:		Display profile successfully
Normal Flow		
Step	Actor	Action
1	Member	Click on “Thông tin cá nhân” in personal page
2	JFS	<p>Display profile page includes:</p> <ul style="list-style-type: none"> - Header - “Họ và Tên” text box - Email text box - “Ngày tháng năm sinh” date picker - “Chuyên môn” box - “Giới tính” radio button - “Số điện thoại” text box - “Địa chỉ” text box - “Cập nhật” button - “Hủy” button
Alternative Flows: N/A		
Exceptions: N/A		
Priority		High
Frequency of Use:		Medium
Business Rules:		B2
Other Information:		N/A

Table 3.5.2.2.2.1 View profile use case

3.5.2.2.2.2 UC022 – Edit profile

Use Case ID	UC022	Version	1.0
Use Case Name	Edit profile		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	

Description:		Member can edit their profile.
Preconditions:		1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system
Trigger:		Click on “Thông tin cá nhân” link in personal page
Post conditions:		1. Edit information successfully 2. Notify edit successfully
Normal Flow		
Step	Actor	Action
1	Member	Click on “Thông tin cá nhân” link in personal page
2	JFS	Display profile page includes: <ul style="list-style-type: none"> - Header - “Họ và Tên” text box - Email text box - “Ngày tháng năm sinh” date picker - “Giới tính” radio button - “Số điện thoại” text box - “Địa chỉ” text box - “Cập nhật” button - “Hủy” button
3	Member	Edit data in fields
4	Member	Click “Cập nhật” button
5	JFS	Display “Thông tin cá nhân” page.
Alternative Flows:		
AL1	At step 4, member click “Hủy” button	
Step	Actor	Action
4.1	JFS	Redirect to profile page
Exceptions:		
EC1	At step 4, required fields are not entered	

Step	Actor	Action
4.1	JFS	Display Profile page with message: required fields are not selected
4.2	JFS	Mark error fields
EC2	At step 4, fields are entered with wrong type of data	
Step	Actor	Action
4.1	JFS	Display Profile page with message: fields are entered with wrong type of data
4.2	JFS	Mark error fields
Priority	High	
Frequency of Use:	Medium	
Business Rules:	B2, B6	
Other Information:	N/A	

Table 3.5.2.2.2.2 Edit profile use case

3.5.2.2.3 UC023 – View personal Page

Use Case ID	UC023	Version	1.0
Use Case Name	View personal Page		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	Member can view personal Page		
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system		
Trigger:	Click on “Trang cá nhân” link on the website		
Post conditions:	Member can view all personal information on personal Page		
Normal Flow			

Step	Actor	Action
13	Member	Click on “Trang cá nhân” link on the website
14	JFS	JFS will display all personal information on personal Page includes: <ul style="list-style-type: none"> - “Quản lý bài đăng” link - “Thông tin cá nhân” link - “Thay đổi mật khẩu” link - List posts - “Góp ý và báo cáo sự cố” icon - “Online chat”
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.2.2.3 View personal Page use case

3.5.2.2.2.4 UC024 – View History Transaction

Use Case ID	UC024	Version	1.0
Use Case Name	View History Transaction		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	Member view all history transaction		
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system		
Trigger:	Click on “Lịch sử giao dịch” link on the personal page		
Post conditions:	Member can view all history transaction on personal Page		
Normal Flow			
Step	Actor	Action	

15	Member	Click “Lịch sử giao dịch” link on the personal page
16	JFS	JFS will display all history transaction on personal Page.
Alternative Flows: N/A		
Exceptions: N/A		
Priority		High
Frequency of Use:		High
Business Rules:		N/A
Other Information:		N/A

Table 3.5.2.2.2.4 View History Transaction

3.5.2.2.3 View other Member's Personal Page

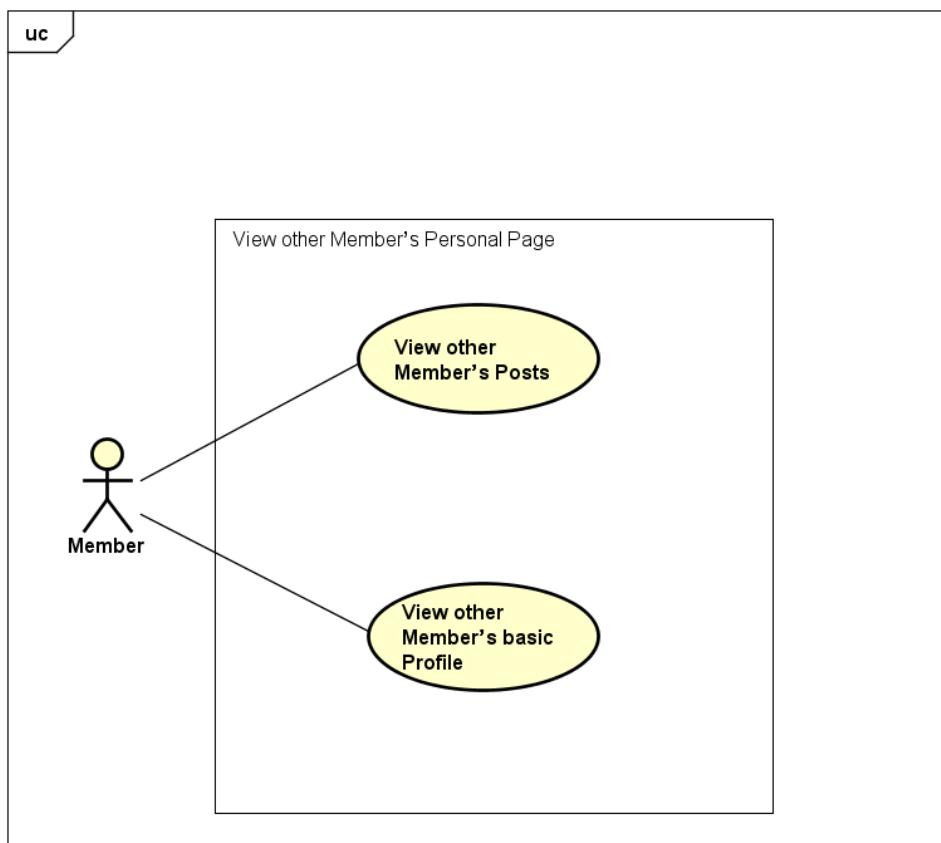


Figure 3.5.2.2.3 View other Member's Personal Page use case

3.5.2.2.3.1 UC025 – View other Member’s Post

Use Case ID		UC025	Version	1.0			
Use Case Name		View other Member’s Post					
Created by:	DuyDD	Date Created:	13/09/2016				
Primary Actor	Member	Secondary Actor					
Description:	Member can view list post of other Member in system						
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member accessed to other Member’s personal Page						
Trigger:	Click on “Quản lý bài đăng” link on the personal Page of other Member						
Post conditions:	1. System display all post was created by other Member						
Normal Flow							
Step	Actor	Action					
1	Member	Click on personal Page link of other member on website					
2	JFS	Click on personal Page link of other member on website Display personal page includes: - Header - List posts - “Thông tin cá nhân” link - “Quản lý bài đăng” link					
3	Member	Click on “Quản lý bài đăng” link					
4	JFS	Display all post was created by this member					
Alternative Flows : N/A							
Exceptions: N/A							
Priority	High						
Frequency of Use:	High						
Business Rules:	B3						
Other Information:	N/A						

Table 3.5.2.2.3.1 View other member posts use case

3.5.2.2.3.2 UC026 - View other member's basic Profile

Use Case ID		UC026		Version	1.0				
Use Case Name		View other member's basic Profile							
Created by:		ThuyTT		Date Created:	20/09/2016				
Primary Actor		Member		Secondary Actor					
Description:		This function allow Job Seeker can view other Member's profile							
Preconditions:		1. JFS must be connected to the Internet 2. Job Seeker logged in to system							
Trigger:		Member click "Thông tin cá nhân" link on personal Page of other member							
Post conditions:		Basic profile of other member will display on profile Page.							
Normal Flow									
Step	Actor	Action							
1	Member	Click on personal Page link of other member on website							
2	JFS	Click on personal Page link of other member on website Display personal page includes: - Header - List posts - "Thông tin cá nhân" link - "Quản lý bài đăng" link							
3	Member	Click on "Quản lý bài đăng" link							
4	JFS	Display basic porfile of member							
Alternative Flows: N/A									
Exceptions: N/A									
Priority		High							
Frequency of Use:		High							

Business Rules:	B8
Other Information:	N/A

3.5.2.2.4 Manage post

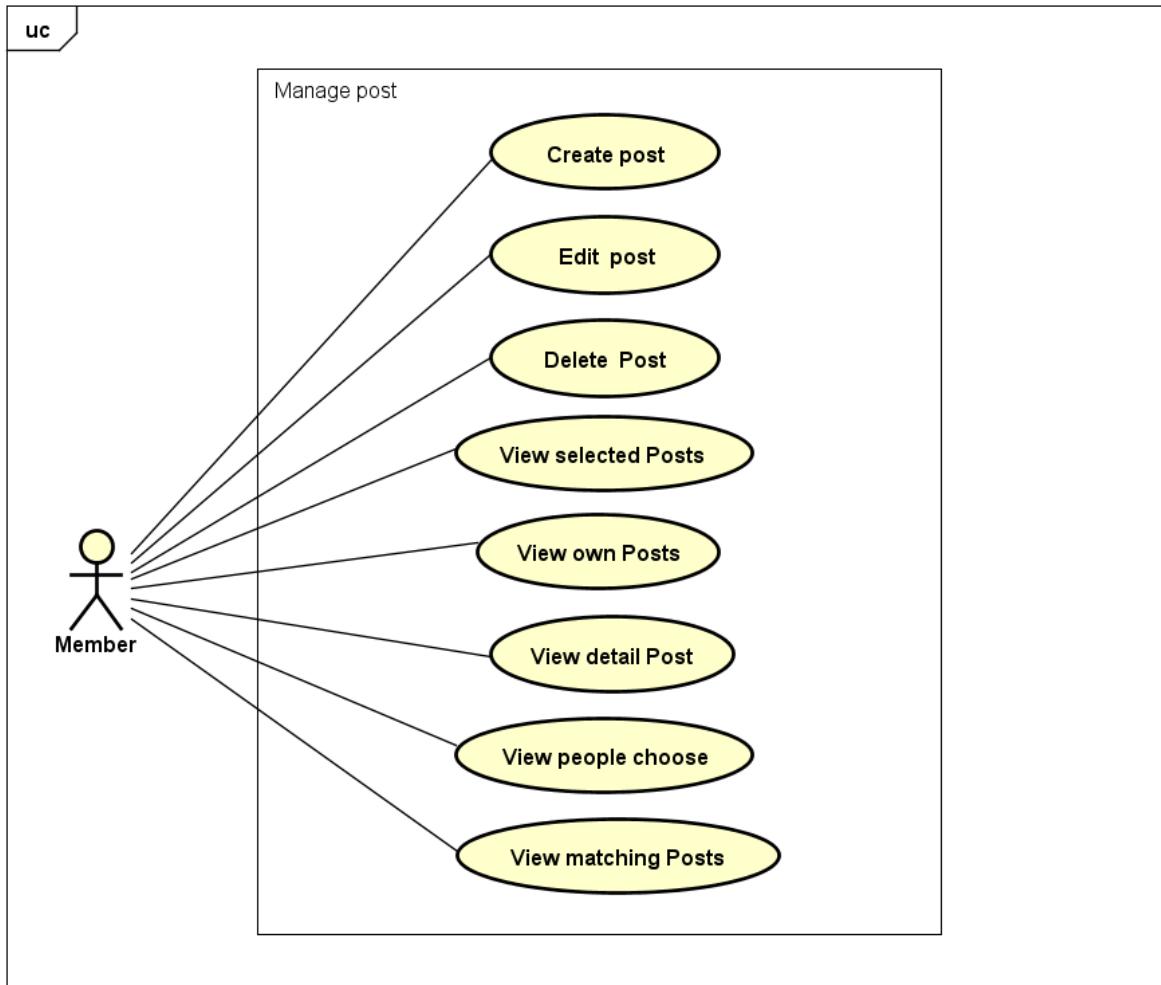


Figure 3.5.2.2.4 Manage posts use case

3.5.2.2.4.1 UC027 – Create post

Use Case ID	UC027	Version	1.0
Use Case Name	Create post		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	

Description:		Member can create a new post
Preconditions:		1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system
Trigger:		Click on “Tạo bài đăng tìm người” or “Tạo bài đăng tìm việc” link on Personal Page
Post conditions:		1. Post will display on Personal Page and homepage
Normal Flow		
Step	Actor	Action
1	Member	Click on “Tạo bài đăng tìm người” or “Tạo bài đăng tìm việc” link on Personal Page
2	JFS	Display “Tạo bài đăng” page includes: - Header - “Chủ đề công việc” text box - “Loại công việc” check box - “Tên công việc” text box - “Thời gian bắt đầu” date - “Địa điểm” text box - “Mô tả tóm tắt công việc” text box - “Mức lương mong muốn” text box - “Tạo” button - “Hủy” button
3	Member	Type data in fields
4	Member	Click “Tạo” button
5	JFS	Display notify create successfully
Alternative Flows		
AT1	At step 4, user click “Hủy” button	
Step	Actor	Action
4.1	JFS	Redirects member to previous page
Exceptions:		

EC1	At step 4, required fields are not entered	
Step	Actor	Action
4.1	JFS	Display Profile page with message: required fields are not entered
4.2	JFS	Mark error fields
EC2	At step 4, fields are entered with wrong type of data	
Step	Actor	Action
4.1	JFS	Display Profile page with message: fields are entered with wrong type of data
4.2	JFS	Mark error fields
Priority	High	
Frequency of Use:	High	
Business Rules:	B2	
Other Information:	N/A	

Table 3.5.2.2.4.1 Member create post use case

3.5.2.2.4.2 UC028 – Edit post

Use Case ID	UC028	Version	1.0
Use Case Name	Edit post		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	Member can edit Post's information		
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system		
Trigger:	Click on "Sửa" button on post detail		

Post conditions:		1. Change password successfully 2. Notify change password successfully
Normal Flow		
Step	Actor	Action
1	Member	Click on “Sửa” link on post detail
2	JFS	<p>Display edit modal on post detail:</p> <ul style="list-style-type: none"> - “Chủ đề công việc” text box - “Loại công việc” check box - “Tên công việc” text box - “Thời gian bắt đầu” - “Địa điểm” text box - “Mô tả tóm tắt công việc” text box - “Mức lương mong muốn” text box - “Cập nhật” button - “Hủy” button
3	Member	Type data in fields
4	Member	Click “Cập nhật” button
5	JFS	Display notify update information successfully
Alternative Flows		
AT1	At step 4, user click “Hủy” button	
Step	Actor	Action
4.1	JFS	Redirects member to previous page
Exceptions:		
EC1	At step 4, required fields are not entered	
Step	Actor	Action
4.1	JFS	Display Profile page with message: required fields are not entered
4.2	JFS	Mark error fields

EC2	At step 4, fields are entered with wrong type of data	
Step	Actor	Action
4.1	JFS	Display Profile page with message: fields are entered with wrong type of data
4.2	JFS	Mark error fields
Priority	High	
Frequency of Use:	Medium	
Business Rules:	B2	
Other Information:	N/A	

Table 3.5.2.2.4.2 Member edit post Use case

3.5.2.2.4.3 UC029 – Delete post

Use Case ID	UC029	Version	1.0
Use Case Name	Delete Post		
Created by:	SonPT	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	This function allows a member to delete his own post.		
Preconditions:	JFS must be connected to the Internet. Member must be logged in.		
Trigger:	Member click “Xóa” button in post detail.		
Post conditions:	Post removed.		
Normal Flow			
Step	Actor	Action	
1	Member	- Member click “Xóa” button in post detail.	
2	JFS	- Display pop-up include: Text “Bạn có chắc mình muốn xóa bài đăng này?”. Button “Đồng ý”. Button “Hủy”.	

4	Member	- Click button “Đồng ý”.
5	JFS	- Delete post.
Alternative Flows:		
AT1	At step 4 in the main flows, if member click button “Hủy”.	
Step	Actor	Action
4.1	Member	- Click button “Hủy”.
5.1	JFS	- Pop up closed. System make no action.
Exceptions: N/A		
Priority	High	
Frequency of Use:	Medium	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.2.4.3 Delete Post Use Case

3.5.2.2.4.4 UC030 – View selected Posts

Use Case ID	UC030	Version	1.0
Use Case Name	View selected Posts		
Created by:	DuyDD	Date Created:	13/09/2016
Primary Actor	Member	Secondary Actor	
Description:	Member can view posts that they are selected.		
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system		
Trigger:	Click on “Bài đăng tìm việc” link on the personal Page		
Post conditions:	Display selected Posts successfully		
Normal Flow			
Step	Actor	Action	
1	Member	Click on “Bài đăng tìm việc” link or “Bài đăng tìm người” link on the personal Page	

2	JFS	Display all posts that they are selected.
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	Medium	
Business Rules:	B3	
Other Information:	N/A	

Table 3.5.2.2.4.4 View selected Posts Use case

3.5.2.2.4.5 UC031 – View Posts

Use Case ID		UC031	Version	1.0			
Use Case Name		View Posts					
Created by:	DuyDD	Date Created:		13/09/2016			
Primary Actor	Member	Secondary Actor					
Description:	Member can view posts that they are selected.						
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system						
Trigger:	Click on “Quản lý bài đăng” link on the personal Page						
Post conditions:	Display selected Posts successfully						
Normal Flow							
Step	Actor	Action					
1	Member	Click on “Các bài đã đăng” link on the personal Page					
2	JFS	Display all posts that they are selected.					
Alternative Flows: N/A							
Exceptions: N/A							

Priority	High
Frequency of Use:	Medium
Business Rules:	B1
Other Information:	N/A

Table 3.5.2.2.4.5 View Posts Use case

3.5.2.2.4.6 UC032 – View detail post

Use Case ID	UC032		Version	1.0		
Use Case Name	View detail post					
Created by:	DuyDD	Date Created:	13/09/2016			
Primary Actor	Member	Secondary Actor				
Description:	Member can view detail a post of in system					
Preconditions:	1. JFS must be connected to the Internet 2. Member browsed JFS website					
Trigger:	Click on a post on website					
Post conditions:	Display detail of post					
Normal Flow						
Step	Actor	Action				
1	Member	Click on a post on website				
2	JFS	Display detail post includes: - Header - “Chủ đề công việc” dropdown list - “Tên công việc” text box - “Thời gian bắt đầu” date picker - “Địa điểm” text box - “Mô tả tóm tắt công việc” text box - “Mức lương mong muốn” text box				
Alternative Flows : N/A						

Exceptions: N/A	
Priority	High
Frequency of Use:	High
Business Rules:	B3
Other Information:	N/A

Table 3.5.2.2.4.6 View detail post

3.5.2.2.4.7 UC033 – View people choose

Use Case ID	UC033		Version	1.0					
Use Case Name	View people choose								
Created by:	SonPT	Date Created:	13/09/2016						
Primary Actor	Member	Secondary Actor							
Description:	This function allows a member to view all people that is selected their post in detail Post page.								
Preconditions:	<ol style="list-style-type: none"> 1. JFS must be connected to the Internet 2. Member browsed JFS website 3. Member logged in to system 								
Trigger:	Member click on one of their posts on Personal Page.								
Post conditions:	Display all people that selected their post in detail Post page.								
Normal Flow									
Step	Actor	Action							
1	Member	- Click on one of their posts on Personal Page.							
3	JFS	- Display all people that selected their post in detail Post page.							
Alternative Flows: N/A									
Exceptions: N/A									
Priority		High							
Frequency of Use:		Medium							
Business Rules:		N/A							
Other		N/A							

Information:	
---------------------	--

Table 3.5.2.2.4.7 View people choose Use Case

3.5.2.2.4.8 UC034 – View matching Post

Use Case ID		UC034	Version	1.0			
Use Case Name		View matching Posts					
Created by:	SonPT	Date Created:	13/09/2016				
Primary Actor	Member	Secondary Actor					
Description:	This function allows a member to view matching Posts in detail Post page.						
Preconditions:	4. JFS must be connected to the Internet 5. Member browsed JFS website 6. Member logged in to system						
Trigger:	Member click on one of their posts on Personal Page.						
Post conditions:	Display matching Posts in detail Post page.						
Normal Flow							
Step	Actor	Action					
1	Member	- Click on one of their posts on Personal Page.					
3	JFS	- Display matching Posts page.					
Alternative Flows: N/A							
Exceptions: N/A							
Priority		High					
Frequency of Use:		Medium					
Business Rules:		N/A					
Other Information:		N/A					

Table 3.5.2.2.4.8 View matching Posts Use Case

3.5.2.3 Administrator 管理

3.5.2.3.1 Common

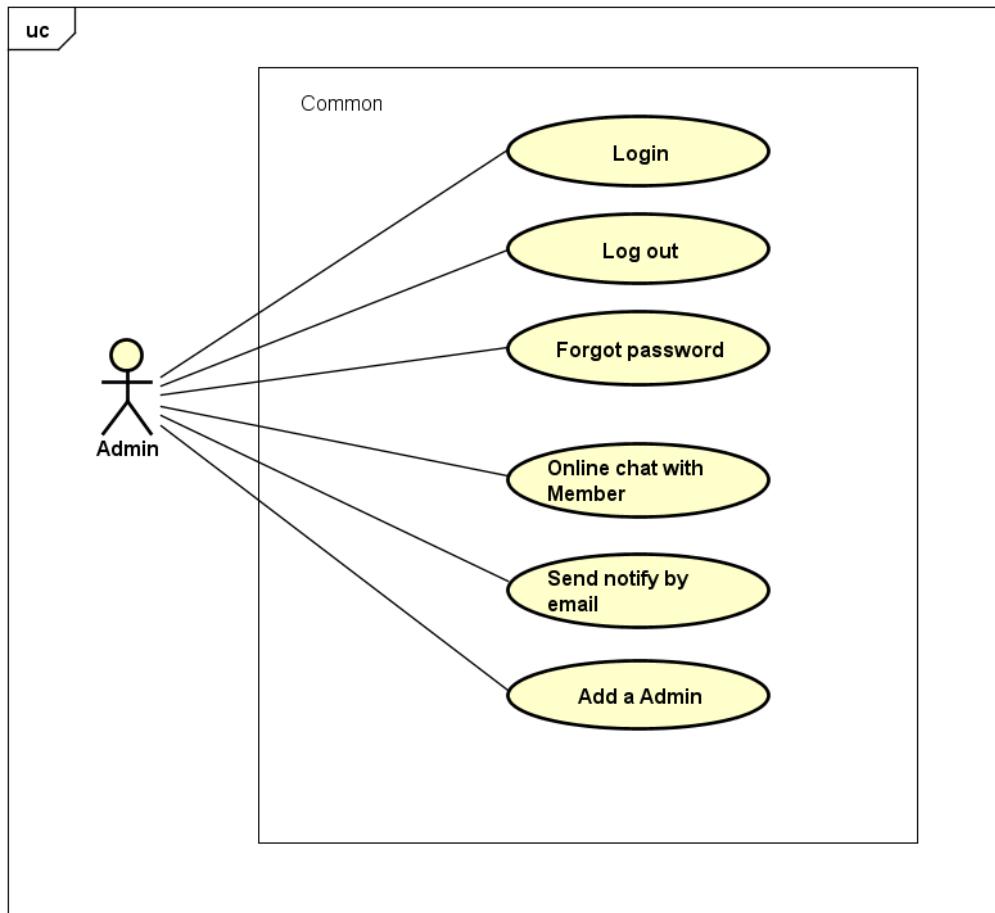


Figure 3.5.2.3.1 Administrator common Use Case

3.5.2.3.1.1 UC035 – Admin login

Use Case ID	UC035		Version	1.0		
Use Case Name	Admin Login					
Created by:	SonPT	Date Created:	13/09/2016			
Primary Actor	Admin	Secondary Actor				
Description:	This function allows an admin to login with an administrator account.					
Preconditions:	JFS must be connected to the Internet. Admin isn't login to website by any account.					
Trigger:	Admin click on button “Đăng nhập” on header of any screen.					
Post conditions:	Admin logged in by an administrator account.					

Normal Flow		
Step	Actor	Action
1	Admin	- Click “Đăng nhập” button on the header of any screen.
2	JFS	- Show the Login Pop up. The Login Pop up includes: Text Field to enter account’s name with textbox “Tên tài khoản”. Text Field to enter password with textbox “Mật khẩu”. Button “Đăng nhập”. Button “Hủy”. Checkbox “Duy trì đăng nhập”. Hyperlink “Quên mật khẩu” with link to access the Forgot Password Page.
3	Admin	- Enter account’s name and password.
4	Admin	- Admin check at “Duy trì đăng nhập” checkbox.
5	Admin	- Admin click button “Đăng nhập”.
6	JFS	- Pop up is closed.
7	JFS	- Move to Administrator’s Page.
Alternative Flows:		
AT1	Admin enter correct account and password, but does not check at “Lưu mật khẩu” checkbox, then click button “Đăng nhập”.	
Step	Actor	Action
4.1	Admin	- Admin does not check at “Duy trì đăng nhập” checkbox.
5.1	Admin	- Admin click button “Đăng nhập”
6.1	JFS	- Pop up is closed.
7.1	JFS	- Move to Administrator’s Page.
Exceptions:		
EC1	At step 3, Admin enter invalid account or password.	
Step	Actor	Action
3.1	Admin	Admin enter incorrect username and password.
4.1	Admin	Admin click button “Đăng nhập”.
5.1	JFS	Pop up show text “Bạn đã nhập tên tài khoản hoặc mật khẩu không đúng.”

		Xin hãy kiểm tra và đăng nhập lại.”.
EC2		At step 3, Admin don't enter data in all field.
Step	Actor	Action
3.2	Admin	Admin don't enter data in all field.
4.2	Admin	Admin click button “Đăng nhập”.
5.2	JFS	Show text “Bạn phải nhập vào các trường được đánh dấu *”.
Priority		High
Frequency of Use:		High
Business Rules:		B2
Other Information:		N/A

Table 3.5.2.3.1.1 Admin login Use Case

3.5.2.3.1.2 UC036 – Admin logout

Use Case ID	UC036		Version	1.0		
Use Case Name	Admin Logout					
Created by:	SonPT	Date Created:	13/09/2016			
Primary Actor	Admin	Secondary Actor				
Description:	This function allows an admin to logout.					
Preconditions:	JFS must be connected to the Internet. Admin is already loggin by an administrator account.					
Trigger:	Admin click button “Đăng xuất” on header of any screen.					
Post conditions:	<ul style="list-style-type: none"> - Admin logged out. - JFS redirect to home page. 					
Normal Flow						
Step	Actor	Action				
1	Admin	- Admin move cursor to Administrator Icon in header of anyscreen.				
2	JFS	- Display a list includes button “Đăng xuất”.				
3	Admin	- Click button “Đăng xuất”.				

4	JFS	- Move to Website's Homepage as guest's view.
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	Low	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.3.1.2 Admin logout Use Case

3.5.2.3.1.3 UC037 - Admin forgot password

Use Case ID	UC037		Version	1.0		
Use Case Name	Admin Forgot Password					
Created by:	SonPT	Date Created:	13/09/2016			
Primary Actor	Admin	Secondary Actor				
Description:	This function allows an admin to receive password send to email from system when forgot password.					
Preconditions:	JFS must be connected to the Internet. Admin's account is already created.					
Trigger:	At the Login Pop up, Admin click the link "Quên mật khẩu".					
Post conditions:	Admin receive password send to email from system.					
Normal Flow						
Step	Actor	Action				
1	Admin	- Click "Quên mật khẩu" link on Login Pop up.				
2	JFS	- Move to Forgot Password Page. This page includes: Text field to enter email address. Button "Gửi mật khẩu".				
3	Admin	- Admin enter account's address				
4	Admin	- Admin click button "Gửi mật khẩu".				
5	JFS	- Send password to confirmation email of admin.				
Alternative Flows:						

Exceptions:		
EC1	At step3, If admin enter email that not exist in system's database.	
Step	Actor	Action
3.1	Admin	Admin enter email address that not exist in system's database and then click button "Gửi".
4.1	JFS	Display text: "Email này chưa được đăng ký."
EC2	At step3, If admin doesn't enter data to field email address.	
Step	Actor	Action
3.2	Admin	If admin doesn't enter data to field email address.
4.2	Admin	Click button "Gửi".
4.1	JFS	Display text: "Bạn vui lòng nhập địa chỉ email."
Priority	High	
Frequency of Use:	Low	
Business Rules:	B6	
Other Information:	N/A	

Table 3.5.2.3.1.3 Admin forgot password Use Case

3.5.2.3.1.4 UC038 - Admin Chat Online with Member

Use Case ID	UC038		Version	1.0		
Use Case Name	Admin Chat Online with Member					
Created by:	SonPT	Date Created:	13/09/2016			
Primary Actor	Admin	Secondary Actor				
Description:	This function allows admin to chat online with member.					
Preconditions:	JFS must be connected to the Internet. Admin is already logged in.					
Trigger:	Admin received message from member. Chat box is displayed.					
Post conditions:	Admin can read message and reply to member					
Normal Flow						
Step	Actor	Action				

1	Member	- Member send message to admin by chat box.
2	JFS	- Display chat box for admin includes: Message content. Text field for admin to type reply message Button “Gửi”. Button “x” to close the chat box.
3	Admin	- Click on text field of chat box.
4	Admin	- Type reply content.
5	Admin	- Click “Gửi” button, or press “Enter” on keyboard.
6	JFS	- Send reply message from admin to member.
Alternative Flows:		
AT1	At step 5, if admin click button “x”.	
Step	Actor	Action
5.1	Admin	- Admin click button “x”.
6.1	JFS	- Close the chat box.
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	B4	
Other Information:	N/A	

Table 3.5.2.3.1.4 Admin chat online with Member Use Case

3.5.2.3.1.5 UC039 – Send notify by mail

Use Case ID	UC039	Version	1.0
Use Case Name	Admin Send Notification to Members by Email		
Created by:	SonPT	Date Created:	13/09/2016
Primary Actor	Admin	Secondary Actor	
Description:	This function allows admin to send notification to users's email.		
Preconditions:	JFS must be connected to the Internet.		

Admin access to Administrator's Page.		
Trigger:		Admin click button “Gửi thông báo” trong “Quản lý thành viên” page
Post conditions:		Admin send notification to member.
Normal Flow		
Step	Actor	Action
1	JFS	- Display Administrator Screen
2	Administrator	- Click on “Quản lý thành viên”
3	JFS	<p>Display list account of member include</p> <ul style="list-style-type: none"> - numbers post had been delete - numbers that member had been ban - “Ban” or “Unban”button - “Gửi thông báo” <p>“Xóa” button</p>
4	Admin	- Admin click button “Gửi thông báo”.
5	JFS	<ul style="list-style-type: none"> - Display pop up includes: Name of members that been selected before. Textfield to type notification content. Button “Gửi”.
6	Admin	Type notification content.
7	Admin	- Click button “Đồng ý”.
8	JFS	- Send notification to member’s email.
Alternative Flows: N/A		
Exceptions:		
EC1	At step 7, Admin does not type anything in notification content.	
Step	Actor	Action
7.1	Admin	- Admin click button “Gửi”.
8.1	JFS	<ul style="list-style-type: none"> - Display pop up includes: Text “The notification’s content is empty!”. Button “OK”.
9.1	Admin	- Click button “OK”.
10.1	JFS	- Pop up closed.

Priority	High
Frequency of Use:	Medium
Business Rules:	B15
Other Information:	N/A

Table 3.5.2.3.1.5 Admin Send Notification to Members by Email Use Case

3.5.2.3.1.6 UC040 – Add Admin

Use Case ID	UC040		Version	1.0		
Use Case Name	Add an admin					
Created by:	ThuyTT	Date Created:	20/09/2016			
Primary Actor	Admin	Secondary Actor				
Description:	This function allow Admin can add an admin					
Preconditions:	1. JFS must be connected to the Internet 2. Admin browsed JFS website 3. Admin is already logged in.					
Trigger:	Click on “Thêm quản trị viên” link on Admin personal page					
Post conditions:	1. Add Admin’s account information to database 2. Send confirm email					
Normal Flow						
Step	Actor	Action				
1	Admin	Click on “Thêm quản trị viên” button on website				
2	JFS	Display Register page includes: - Header - “Họ và Tên” text box - “Mật khẩu” text box - “Xác nhận mật khẩu” text box - “Ngày sinh” date picker - “Giới tính” checkbox - “Địa chỉ” textbox - “Email” text box - “Số Điện thoại” text box - “Thêm” button				

		- “Hủy” button
3	Admin	Enters data in fields
4	Admin	Click “Hoàn Tất” button
5	JFS	Send confirm email
6	JFS	Back to Admin personal page
Alternative Flows		
AT1	At step 4, user click “Hủy bỏ” button	
Step	Actor	Step
4.1	JFS	4.1
Exceptions:		
EC1	At step 4, required fields are not entered	
Step	Actor	Step
4.1	JFS	4.1
4.2	JFS	4.2
EC2	At step 4, fields are entered with wrong type of data	
Step	Actor	Step
4.1	JFS	4.1
4.2	JFS	4.2
Priority		High
Frequency of Use:		Low
Business Rules:		B2, B6, B7,
Other Information:		N/A

Table 3.5.2.3.1.6 Add Admin Use Case

3.5.2.3.2 Manage Member

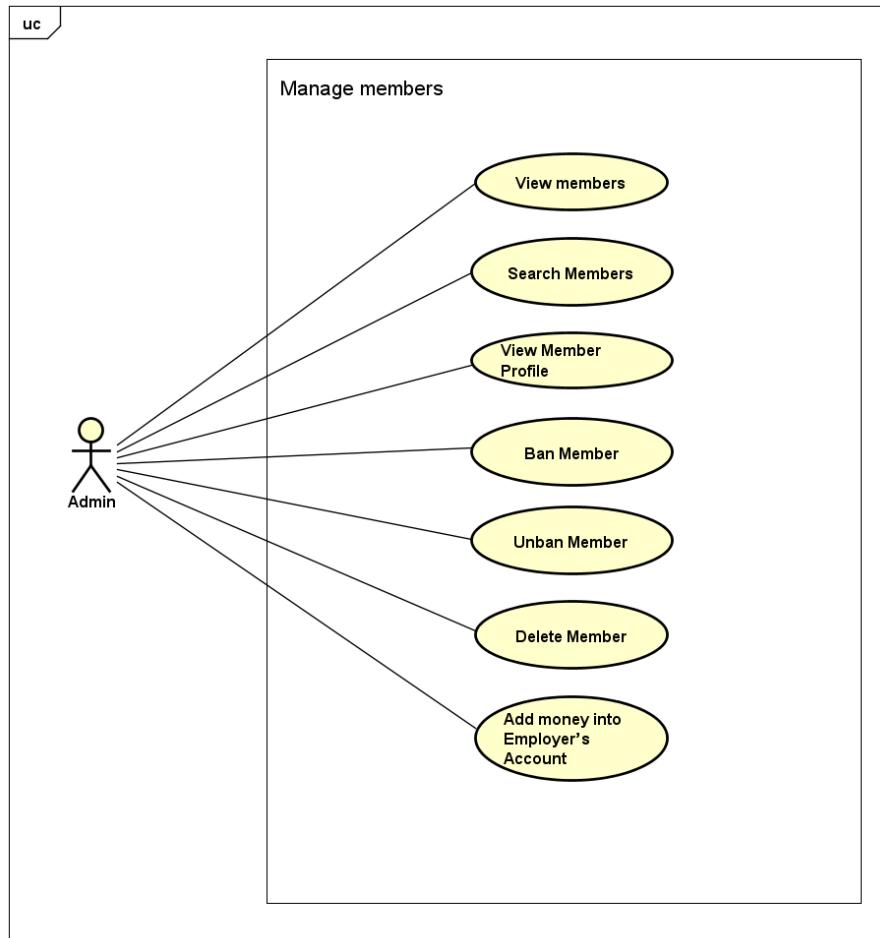


Figure 3.1.2.3.2 Manage Member Use Case

3.5.2.3.2.1 UC041 – View Members

Use Case ID	UC041		Version	1.0		
Use Case Name	Admin View List of Members					
Created by:	SonPT	Date Created:	13/09/2016			
Primary Actor	Admin	Secondary Actor				
Description:	This function allows admin to view list of all member's accounts of system.					
Preconditions:	JFS must be connected to the Internet.					

		Admin is already logged in.
Trigger:		Click on button “Quản lý thành viên” at Administrator’s Page.
Post conditions:		Admin view list of all members of system.
Normal Flow		
Step	Actor	Action
1	Admin	- Access to Administrator’s Page.
2	JFS	- Display Administrator’s Page. This page includes: + Navigation bar. This navigation bar includes: “Thông tin cá nhân” tab. “Quản lý thành viên” tab. “Quản lý bài đăng” tab. “Quản lý góp ý, báo cáo sự cố” tab. “Thống kê” tab. + Admin’s icon with account name.
3	Admin	- Move the cursor to tab “Quản lý thành viên”.
4	Admin	- Click on button “Xem thành viên”.
5	JFS	- Display the View All Members Page. This page includes: + List all member’s account name. Each member’s name is a link to access to that member’s Profile Page.
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.3.2.1 Admin View List of Members Use

3.5.2.3.2.2 UC042 - Search Members

Use Case ID	UC042	Version	1.0
Use Case Name	Search members		
Created by:	NamTH	Date Created:	12/09/2016
Primary Actor	Admin	Secondary Actor	

Description:	This function allows Admin to search a member	
Preconditions:	Admin is viewing Member List at “Quản lý thành viên” page.	
Trigger:	Admin types name or id of member at Search textbox .	
Post conditions:	Display list of searched member	
Normal Flow		
Step	Actor	Action
1	JFS	- Display Administrator Screen
2	Administrator	- Click on “Quản lý thành viên”
3	JFS	Display list account of member include +numbers post had been delete + numbers that member had been ban +”Ban” or “Unban”button +”Gửi thông báo” +”Xóa” button
4	Admin	- Type name or id of member at Search textbox
5	JFS	- Display list of members who have the name or id is typed
Alternative Flows: N/A		
Exceptions:		
EC1	At step 4 in the main flows, if the typed name is not exist	
Step	Actor	Action
4.1	JFS	Do not display any record
Priority		High
Frequency of Use:		High
Business Rules:		B1
Other Information:		N/A

Table 3.5.2.3.2.2 Search members

3.5.2.3.2.3 UC043 - View member profile

Use Case ID		UC043	Version	1.0		
Use Case Name	View member profile					
Created by:	NamTH	Date Created:	12/09/2016			
Primary Actor	Admin	Secondary Actor				
Description:	This function allows Admin to view profile of a Member					
Preconditions:	Admin viewing personal information page					
Trigger:	Admin clicks on “Thông tin cá nhân” link on the website					
Post conditions:	Profile of this Member is displayed					
Normal Flow						
Step	Actor	Action				
1	Admin	Click on name of a member link				
2	JFS	Display corresponding member page				
3	Admin	Click on “Thông tin cá nhân” link on member personal page				
4	JFS	Display all detail information of this Member				
Alternative Flows: N/A						
Exceptions: N/A						
Priority	High					
Frequency of Use:	High					
Business Rules:	N/A					
Other Information:	N/A					

Table 3.5.2.3.2.3: View member profile Use Case

3.5.2.3.2.4 UC044 - Ban Member

Use Case ID		UC044	Version	1.0
Use Case Name	Ban Member			
Created by:	NamTH	Date Created:	20/09/2016	
Primary Actor	Admin	Secondary Actor		
Description:	This function allows Administrator to ban a Member			

Preconditions:		1. JFS must be connected to the Internet 2. Admin is already logged in. 3. Admin viewing Member List
Trigger:		Admin clicks on “Cấm” button
Post conditions:		Member is banned
Normal Flow		
Step	Actor	Action
1	JFS	Display Administrator Screen
2	Administrator	Click on “Quản lý thành viên”
3	JFS	Display list account of member include + numbers post had been delete + numbers that member had been ban + “Ban” or “Unban” button + “Gửi thông báo” + “Xóa” button
4	Admin	Click on “Ban” button at action column of the Member whom want to ban
5	JFS	Display Ban confirm popup includes : - “Bạn có muốn ban thành viên này?” label. - “Đồng ý” button - “Hủy” button
6	Admin	Click on “Đồng ý” button
7	JFS	Send email to notify for Member that “Bài đăng của bạn đã vi phạm luật của eJob . Tài khoản của bạn đã bị khóa trong vòng 1 tuần”.
Alternative Flows		
AT1	At step 6 in the main flows, if Admin click on “Hủy” button	
Step	Actor	Step
6.1	JFS	- Close Ban confirm popup

Exceptions: N/A	
Priority	High
Frequency of Use:	Low
Business Rules:	N/A
Other Information:	N/A

Table 3.5.2.3.2.4: Ban member's account Use Case

3.5.2.3.2.5 UC045 - Unban member r's user account

Use Case ID	UC045	Version	1.0
Use Case Name	UnBan member's user account		
Created by:	NamTH	Date Created:	20/09/2016
Primary Actor	Admin	Secondary Actor	
Description:	This function allows Administrator to unban a Member		
Preconditions:	Admin viewing Member List		
Trigger:	Admin clicks on “Unban” button		
Post conditions:	All Member's action get back to normal.		
Normal Flow			
Step	Actor	Action	
1	JFS	Display Administrator Screen	
2	Administrator	Click on “Quản lý thành viên”	
3	JFS	Display list account of member include +numbers post had been delete + numbers that member had been ban +“Ban” or “Unban”button +“Gửi thông báo” +“Xóa” button	
4	Admin	Click on “Unban” button at Action column of the Member whom want to see profile	
5	JFS	Display Unan confirm popup includes :	

		<ul style="list-style-type: none"> - “Bạn có chắc chắn muốn unban thành viên này?” label - “Đồng ý” button - “Hủy” button
6	Admin	Click on “Đồng ý” button
7	JFS	All Member's action get back to normal.
Alternative Flows		
AT1	At step 6 in the main flows, if Admin click on “Hủy” button	
Step	Actor	Step
6.1	JFS	<ul style="list-style-type: none"> - Turn off Unban confirm popup - Keep of status ban of this member that can't create Post and follow any post.
Exceptions: N/A		
Priority	High	
Frequency of Use:	Low	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.3.2.5: UnBan member's account Use Case

3.5.2.3.2.6 UC046 - Delete member

Use Case ID	UC046	Version	1.0
Use Case Name	Delete Member		
Created by:	NamTH	Date Created:	20/09/2016
Primary Actor	Admin	Secondary Actor	
Description:	This function allows Administrator to delete a member		
Preconditions:	Admin viewing Member List		
Trigger:	Admin clicks on “Xóa” button		
Post conditions:	Member deleted from system.		

Normal Flow		
Step	Actor	Action
1	JFS	Display Administrator Screen
2	Administrator	Click on “Quản lý thành viên”
3	JFS	Display list account of member include + numbers post had been delete + numbers that member had been ban + “Ban” button + “Gửi thông báo” + “Xóa” button
4	Admin	Click on “Xóa” button at Action column of the member whom want to delete
5	JFS	Display Deleted confirm popup includes : - “Bạn có chắc chắn muốn xóa thành viên này?” label - “Đồng ý” button - “Hủy” button
6	Admin	Click on “Đồng ý” button
7	JFS	Member deleted
Alternative Flows		
AT1	At step 6 in the main flows, if Admin click on “Hủy” button	
Step	Actor	Step
6.1	JFS	- Close Delete confirm popup
Exceptions: N/A		
Priority		High
Frequency of Use:		Low
Business Rules:		N/A
Other Information:		N/A

Delete Member Use Case

3.5.2.3.2.7 UC047 - Add money into Employer's financial account

Use Case ID		UC047		Version	1.0			
Use Case Name		Add money into Employer's financial account						
Created by:	NamTH	Date Created:	12/09/2016					
Primary Actor	Admin	Secondary Actor						
Description:	This function allows Administrator to Add money into Employer's account							
Preconditions:	<ul style="list-style-type: none"> - Admin is already logged in - Member 's account is already created - Member paid money to account - Admin is viewing Member List at manage account page. 							
Trigger:	Admin click button "Thêm tiền" and Set Money Popup							
Post conditions:	Money of member's account will be added with value is typed.							
Normal Flow								
Step	Actor	Action						
1	JFS	Display Administrator Screen						
2	Administrator	Click on "Quản lý thành viên"						
3	JFS	Display list account of member include +numbers post had been delete + numbers that member had been ban						
4	Administrator	Click on "Quản lý tài khoản" button						
5	JFS	Display list account of member include +money amount of member						
6	Administrator	Click "Thêm tiền" button						
7	JFS	Show popup "Thêm tiền vào tài khoản"						
8	Administrator	Admin input number of money on textbox "value" on Popup.						
9	JFS	Display set money confirm popup includes : <ul style="list-style-type: none"> - "Bạn có muốn thay đổi số tiền của người dùng này không?" label . 						

		- “Đồng ý” button - “Hủy” button
10	Administrator	Click on “Đồng ý” button
11	JFS	The money of user will be added with value is typed
Alternative Flows		
AT1	At step 10 in the main flows, if Admin click on “Hủy” button	
Step	Actor	Action
10.1	JFS	- Close Delete confirm popup
Exceptions: N/A		
EC1	At Step 8 admin don't enter data to field	
Step	Actor	Action
8.1	JFS	Display message “Vui long điền số tiền muốn nạp”
Priority		High
Frequency of Use:		High
Business Rules:		B11
Other Information:		N/A

Table 3.5.2.3.2.7 Add money into Employer's account Use case

3.5.2.3.3 Manage Member's Posts

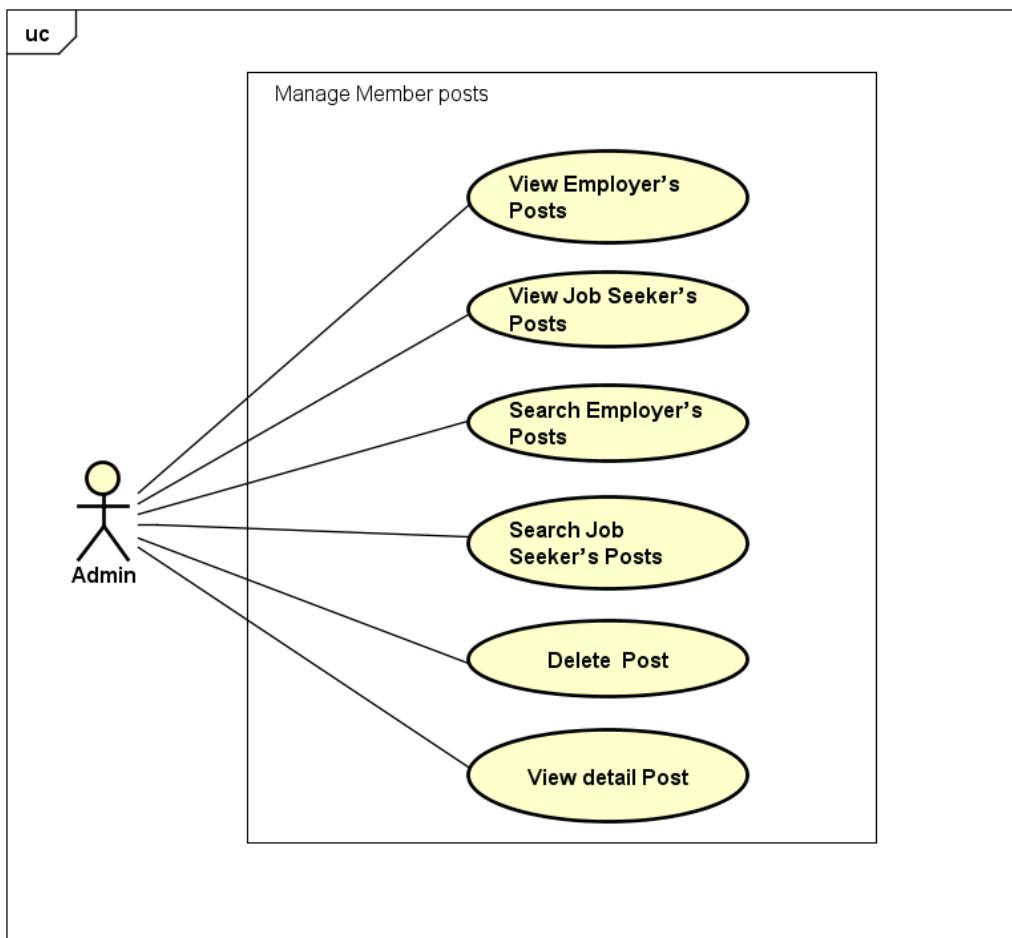


Figure 3.5.2.3.3 Manage Member's Post

3.5.2.3.3.1 UC048 – Admin view Employer's Posts

Use Case ID	UC048		Version	1.0
Use Case Name	Admin view Employer's Posts			
Created by:	NamTH	Date Created:	12/09/2016	
Primary Actor	Admin	Secondary Actor		
Description:	This function allows Administrator to view a list of all Posts of Employers.			
Preconditions:	Admin is already logged in.			
Trigger:	Admin click on “Xem tất cả bài đăng tìm người”.			
Post conditions:	All Post of all Employers are displayed.			
Normal Flow				

Step	Actor	Action
1	Admin	Access to Administrator Page.
2	Admin	Move the cursor to “Quản lý bài đăng” tab.
3	Admin	Click “Bài đăng tìm người”.
4	JFS	Display all Posts of All Employers.
Alternative Flows: N/A		
Exceptions:		
EC1	If there are no Post to show	
Step	Actor	Action
4	JFS	Display page with no Post.
Priority		High
Frequency of Use:		High
Business Rules:		B3
Other Information:		N/A

Table 3.5.2.3.3.1 Admin view Employer's Posts Use Case

3.5.2.3.3.2 UC049 – Admin view Job Seeker's Posts

Use Case ID	UC049		Version	1.0		
Use Case Name	Admin view Job Seeker's Posts					
Created by:	NamTH	Date Created:	12/09/2016			
Primary Actor	Admin	Secondary Actor				
Description:	This function allows Administrator to view a list of all Posts of Job Seekers.					
Preconditions:	Admin is already logged in.					
Trigger:	Admin click on “Bài đăng tìm việc”.					
Post conditions:	All Post of all Job Seekers are displayed.					
Normal Flow						
Step	Actor	Action				

1	Admin	Access to Administrator Page.
2	Admin	Move the cursor to “Quản lý bài đăng” tab.
3	Admin	Click “Bài đăng tìm việc”.
4	JFS	Display all Posts of All Job Seekers.
Alternative Flows: N/A		
Exceptions:		
EC1	If there are no Post to show	
Step	Actor	Action
4	JFS	Display page with no Post.
Priority		High
Frequency of Use:		High
Business Rules:		B3
Other Information:		N/A

Table 3.5.2.3.3.2 Admin view Job Seeker's Posts Use Case

3.5.2.3.3.3 UC050– Admin search JobSeeker Posts

Use Case ID	UC050-51	Version	1.0
Use Case Name	Admin search JobSeeker Posts		
Created by:	NamTH	Date Created:	12/09/2016
Primary Actor	Admin	Secondary Actor	
Description:	This function allows Administrator to search Posts. - Search Job Seeker's Posts		
Preconditions:	Admin is already logged in.		
Trigger:	Admin types keyword in Search textbox, or select category, city and district at drop-down list .		
Post conditions:	Display list of searched post that has same subject or list post that has same category at city (and district).		

		Search Posts are displayed.
Normal Flow		
Step	Actor	Action
1	Admin	Access to Administrator Page.
2	Admin	Move the cursor to “Quản lý bài đăng” tab.
3	Admin	Click “Bài đăng tìm việc” button.
4	JFS	Access to Search Post Page.
5	Admin	Enter needed information to search. + input key work +select category +select city +select district
6	JFS	Display list of post that has the id is typed or same category , city, district what selected.
Alternative Flows: N/A		
Exceptions:		
EC1	If there are no Post found.	
Step	Actor	Action
6	JFS	Display message “Không tìm thấy bài đăng nào”.
Priority		High
Frequency of Use:		High
Business Rules:		B1
Other Information:		N/A

Table 3.5.2.3.3.3 Admin Search Job Seeker's Posts Use Case

3.5.2.3.3.4 UC051– Admin search Employer Posts

Use Case ID		UC050-51		Version	1.0				
Use Case Name		Admin search Employer Posts							
Created by:		NamTH		Date Created:	12/09/2016				
Primary Actor		Admin		Secondary Actor					
Description:		This function allows Administrator to search Posts. - Search Employer's Posts							
Preconditions:		Admin is already logged in.							
Trigger:		Admin types keyword in Search textbox, or select category, city and district at drop-down list .							
Post conditions:		Display list of searched post that has same subject or list post that has same category at city (and district). Search Posts are displayed.							
Normal Flow									
Step	Actor	Action							
1	Admin	Access to Administrator Page.							
2	Admin	Move the cursor to “Quản lý bài đăng” tab.							
3	Admin	Click “Bài đăng tìm người” button.							
4	JFS	Access to Search Post Page.							
5	Admin	Enter needed information to search. + input key work +select category +select city +select district							
6	JFS	Display list of post that has the id is typed or same category , city, district what selected.							
Alternative Flows: N/A									
Exceptions:									
EC1	If there are no Post found.								
Step	Actor	Action							

6	JFS	Display message “Không tìm thấy bài đăng nào”.
Priority	High	
Frequency of Use:	High	
Business Rules:	B1	
Other Information:	N/A	

Table 3.5.2.3.3.4 Admin Search Employer's Posts Use Case

3.5.2.3.3.5 UC052– Delete Post

Use Case ID		UC052	Version	1.0
Use Case Name		Admin delete Post		
Created by:		NamLH	Created by:	NamLH
Primary Actor		Admin	Primary Actor	Admin
Description:		This function allows Administrator to delete Member's Post		
Preconditions:		1. JFS website is available 2. Administrator logged in to the system.		
Trigger:		Administrator click “Xóa” icon		
Post conditions:		Delete this post successfully (delete in database)		
Normal Flow				
Step	Actor	Action		
1	User	Access to Administrator Page of JFS		
2	JFS	Display Administrator Screen		
3	Administrator	Hover “Quản lý bài đăng” tab		
4	JFS	Display a list include : 1. “Bài đăng tìm việc” link 3. “Bài đăng tìm người” link		
5	Administrator	Click “Bài đăng tìm việc” or “Bài đăng tìm người” link		

6	JFS	Display list of member's post
7	Administrator	Click a post
8	JFS	Display content of post
9	Administrator	Click icon "Xóa"
10	JFS	Display alert to confirm “Bạn có muốn chắc chắn xóa bài bài đăng tìm việc này ?” / “Bạn có muốn chắc chắn xóa bài bài đăng tìm người này ?”
11	Administrator	Click button “Đồng ý”
12	JFS	Display list post updated
Alternative Flows:		
AT1	At step 11 in the main flows, Admin click “Không” button	
Step	Actor	Action
11.1	JFS	Back to display content of post
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.3.3.5: Admin Delete Post Use Case

3.5.2.3.3.6 UC053 - View detail Post

Use Case ID	UC053	Version	1.0
Use Case Name	Admin view detail Post		
Created by:	NamTH	Date Created:	12/09/2016
Primary Actor	Admin	Secondary Actor	
Description:	This function allows Administrator to view and check detail member's post		

Preconditions:	Admin is already logged in.	
Trigger:	Click on post.	
Post conditions:	Information of this Post is displayed.	
Normal Flow		
Step	Actor	Action
1	Admin	Click on post.
2	JFS	Information of this Post is displayed
3	Admin	View information of this post.
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	B3	
Other Information:	N/A	

Table 3.5.2.3.3.6: Admin view detail Post Use Case

3.5.2.3.4 Manage feedback of website

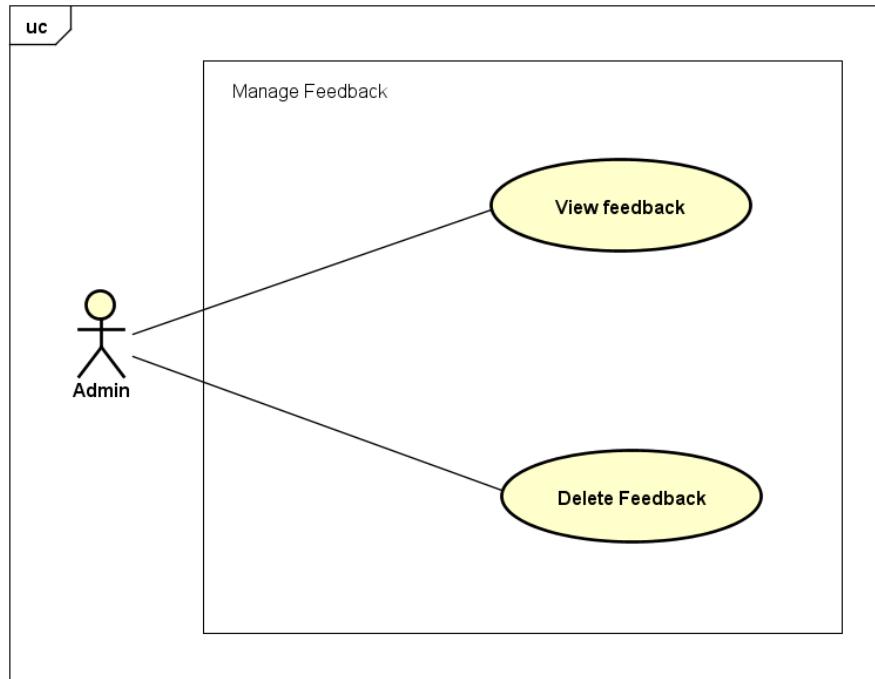


Figure 3.5.2.3.4 Manage feedback of website Use Case

3.5.2.3.4.1 UC054 - View feedback of website

Use Case ID	UC054		Version	1.0
Use Case Name	Admin view feedback of website			
Created by:	NamLH	Date Created:	12/09/2016	
Primary Actor	Administrator	Secondary Actor		
Description:	This function allows user to view all feedback from user (guess, member)			
Preconditions:	1. JFS website is available 2. Admin logged in to the system.			
Trigger:	Administrator click "Góp ý" tab			
Post conditions:	Administrator can view all feedbacks from user			
Normal Flow				
Step	Actor	Action		
1	User	Access to Administrator Page of JFS		
2	JFS	Display Administrator Screen		

3	Administrator	Click “Góp ý” tab
4	JFS	Display list feedbacks from user
Alternative Flows: N/A		
Exceptions:		
EC1		At step 4, if don't have feedbacks
Step	Actor	Action
4.1	JFS	Display message: “Không có góp ý hay báo cáo sự cố nào”
Priority		High
Frequency of Use:		High
Business Rules:		B4
Other Information:		N/A

Table 3.5.2.3.5.1: View feedback of website Use Case

3.5.2.3.4.2 UC055 - Delete feedback of website

Use Case ID	UC055	Version	1.0
Use Case Name	Admin delete feedback of website		
Created by:	NamLH	Date Created:	12/09/2016
Primary Actor	Administrator	Secondary Actor	
Description:	This function allows administrator to delete feedback of website from user		
Preconditions:	1. JFS website is available 2. User sent comment or reported issue 3. Administrator logged in website		
Trigger:	Administrator clicks on button “Xóa”		
Post conditions:	Delete feedback successfully		
Normal Flow			
Step	Actor	Action	
1	User	Access to Administrator Page of JFS	

2	JFS	Display Administrator Screen
3	Administrator	Click “Góp ý” tab
4	JFS	Display list feedback from user
5	Administrator	Click on Name of the feedback
6	JFS	Display the content of feedback that is selected
7	Administrator	Click button “Xóa”
8	JFS	Display alert to confirm “Bạn chắc chắn xóa góp ý này ?”
9	Administrator	Click button “Đồng ý”
10	JFS	Display list updated feedback
Alternative Flows:		
AT1	At step 9 in the main flows, Administrator click “Không” button	
Step	Actor	Action
9.1	System	Back to display content of feedback
Exceptions:		
EC1	At step 4, if don't have feedback	
Step	Actor	Action
4.1	JFS	Display message: “Không có góp ý hay báo cáo sự cố nào”
Priority	High	
Frequency of Use:	Low	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5..2.3.4.2 Admin delete feedback of website Use Case

3.5.2.3.5 Statistic

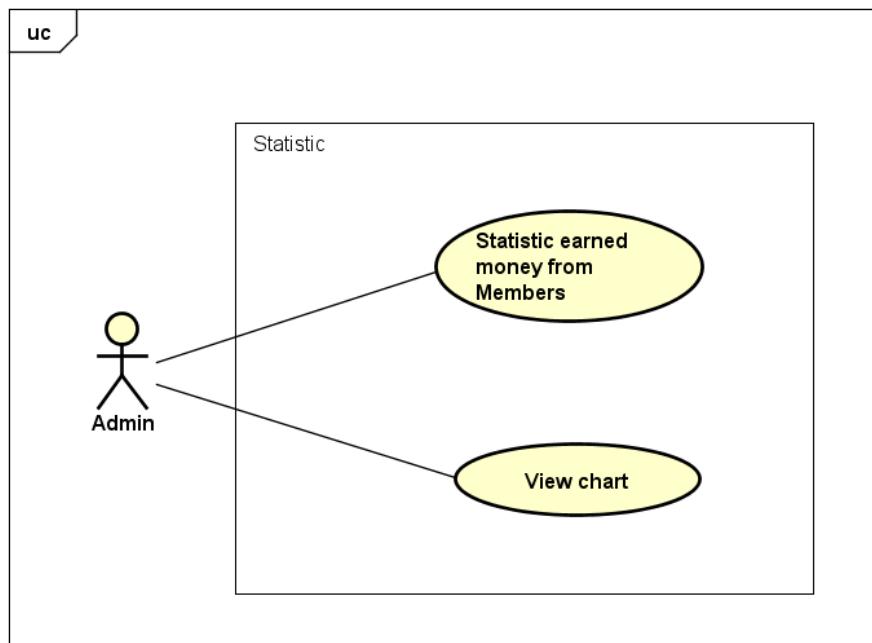


Figure 3.5.2.3.5 Statistic Use Case

3.5.2.3.5.1 UC056- Statistic earned money from Members

Use Case ID	UC056		Version	1.0
Use Case Name	Admin statistic earned money from Members			
Created by:	NamLH	Date Created:	12/09/2016	
Primary Actor	Administrator	Secondary Actor		
Description:	Administrator can statistic money that website earned			
Preconditions:	1. JFS website is available 2. Member have completed transaction 3. Administrator logged in to system			
Trigger:	Click on “Thống kê” Tab			
Post conditions:	Display statistic table about money that website have earned			
Normal Flow				
Step	Actor	Action		
1	Administrator	Access to Administrator Page of JFS		
2	JFS	Display Administrator Screen		

3	Administrator	Hover “Thống kê” tab
4	JFS	Display “Thống kê theo loại công việc” and “Thống kê doanh thu” button
5	Administrator	Click “Thống kê doanh thu” button
6	JFS	Display Money Statistic that website have earned from member
Alternative Flows: N/A		
Exceptions:		
EC1	At step 6, if don't have transaction (system don't get money from member)	
Step	Actor	Action
6.1	JFS	Display message: “Không có giao dịch nào”
Priority	High	
Frequency of Use:	High	
Business Rules:	B11	
Other Information:	N/A	

Table 3.5.2.3.5.1: Admin statistic earned money from Members Use Case

3.5.2.3.5.2 UC057 – View Chart

Use Case ID	UC057		Version	1.0		
Use Case Name	Admin View Chart					
Created by:	NamLH	Date Created:	12/09/2016			
Primary Actor	Administrator	Secondary Actor				
Description:	Administrator view a chart which show the percentage of category was selected in one months					
Preconditions:	1 JFS website is available 2 Member have completed transaction 3 Administrator logged in to system					
Trigger:	Click on “Thống kê theo loại công việc” button					

Post conditions:		Display chart which show the percentage of category with numbers post of the category .
Normal Flow		
Step	Actor	Action
1	Administrator	Access to Administrator Page of JFS
2	JFS	Display Administrator Screen
3	Administrator	Hover “Thống kê” tab
4	JFS	Display “Thống kê theo loại công việc” and “Thống kê doanh thu” button
5	Administrator	Click “Thống kê theo loại công việc” button
6	JFS	Display chart which show the percentage of category with numbers post of the category .
Alternative Flows: N/A		
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.3.5.2: Admin view Chart Use Case

3.5.2.3.6 Manage Report

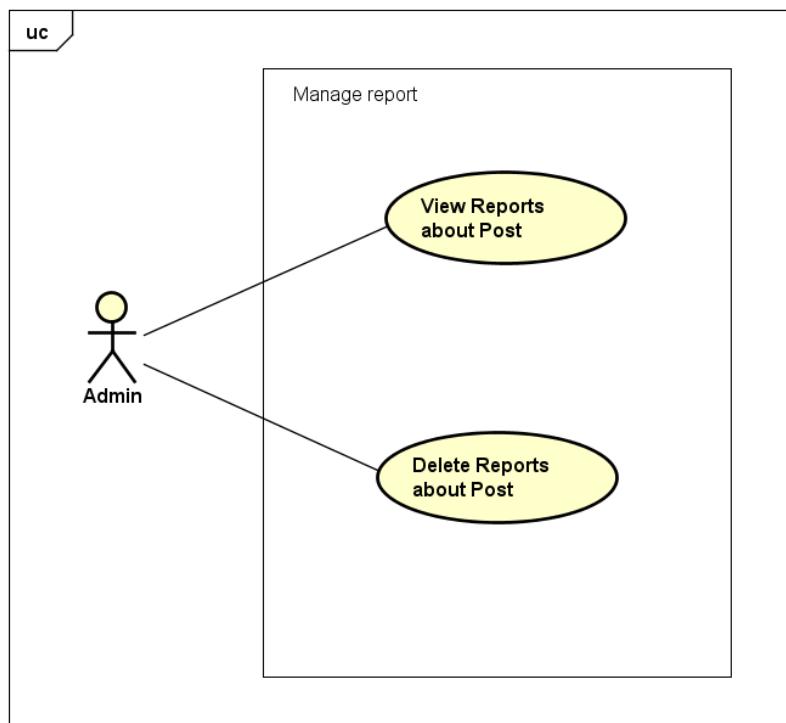


Figure 3.5.2.3.6 Manage report

3.5.2.3.6.1 UC058 - View Reports about Post

Use Case ID		UC058	Version	1.0	
Use Case Name		Admin view Reports about Post			
Created by:	NamLH	Date Created:	12/09/2016		
Primary Actor	Administrator	Secondary Actor			
Description:	Administrator can view list report about some post from member				
Preconditions:	1. JFS website is available 2. Member sent report about other member's post 3. Administrator logged in website				
Trigger:	Click on “Quản lý bài đăng bị report” link				
Post conditions:	Display list report about some post from member				
Normal Flow					
Step	Actor	Action			
1	Administrator	Access to Administrator Page of JFS			

2	JFS	Display Administrator Screen
3	Administrator	Click “Quản lý bài đăng” tab
4	JFS	Display a list include : - “Bài đăng của member” link 4. Content reports
5	JFS	Display list report about some post from member
Alternative Flows: N/A		
Exceptions:		
EC1	At step 6, if don't have comment or report issues	
Step	Actor	Action
6.1	JFS	Display message: “Không có bài đăng nào bị report”
Priority	High	
Frequency of Use:	High	
Business Rules:	B3	
Other Information:	N/A	

Table 3.5.2.3.6.1: Admin view Report about Post

3.5.2.3.6.2 UC059- Delete Reports about Post

Use Case ID	UC059	Version	1.0
Use Case Name	Admin delete Reports about Post		
Created by:	NamLH	Date Created:	12/09/2016
Primary Actor	Administrator	Secondary Actor	
Description:	Administrator can delete a report from table list of report		
Preconditions:	1. JFS website is available 2. Member sent report about other member's post 3. Administrator logged in website		
Trigger:	Click on button “Xóa”		

Post conditions:		Delete successfully a report from table list of report
Normal Flow		
Step	Actor	Action
1	Administrator	Access to Administrator Page of JFS
2	JFS	Display Administrator menu tab
3	Administrator	Click “Quản lý bài đăng” tab
4	JFS	Display a list include : - “Bài đăng của member” link Content reports
5	Administrator	Click on “Quản lý báo cáo” link
6	JFS	Display list report about some post from member
7	Administrator	Select a report from list
8	Administrator	Click button “Xóa”
9	JFS	Display alert to confirm “Bạn chắc chắn muốn xóa report này ?”
10	Administrator	Click button “Đồng ý”
11	JFS	Display list reports updated
Alternative Flows		
AT1	At step 8, user click “Hủy” button	
Step	Actor	Action
8.1	JFS	Back to display list report about post from user
Exceptions: N/A		
Priority		High
Frequency of Use:		High
Business Rules:		N/A
Other Information:		N/A

Table 3.5.2.3.6.2 : Admin delete Reports about Post Use Case

3.5.2.3.7 Manage Category

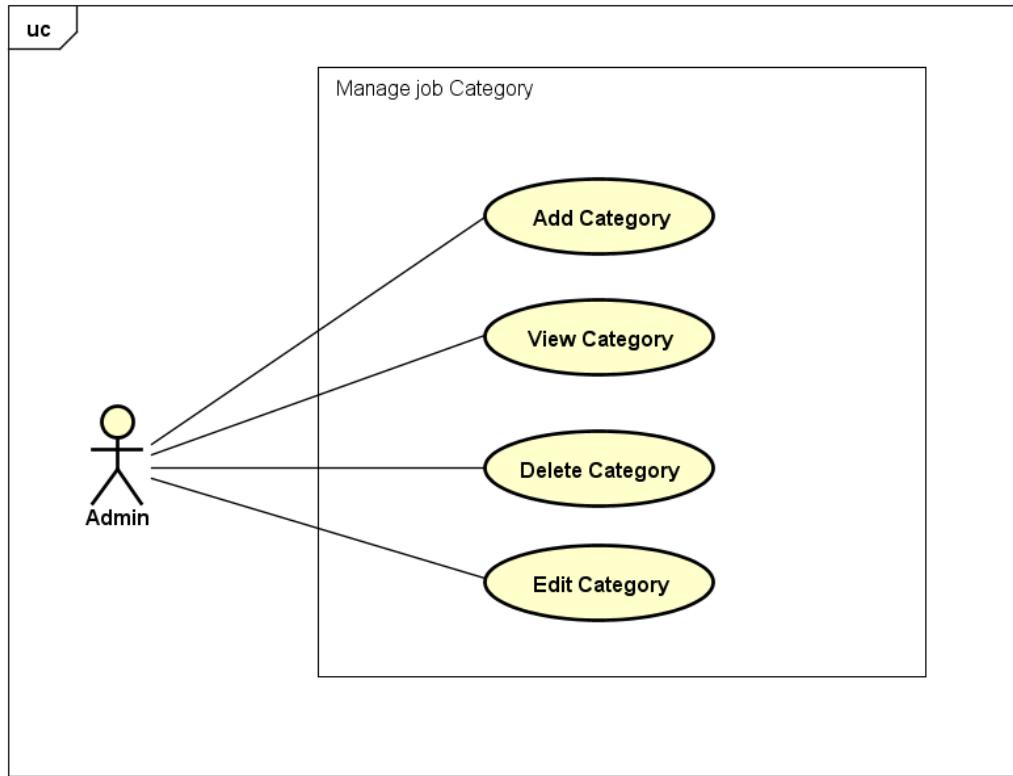


Figure 3.5.2.3.7 Manage job Category

3.5.2.3.7.1 UC060 – Add category

Use Case ID	UC060		Version	1.0		
Use Case Name	Admin add Job Category					
Created by:	NamLH	Date Created:	12/09/2016			
Primary Actor	Administrator	Secondary Actor				
Description:	Add new job's category to list of category					
Preconditions:	1. JFS website is available 2. Administrator logged in website					
Trigger:	Click on “Nhóm công việc” tab					
Post conditions:	Add new job's category to database					
Normal Flow						

Step	Actor	Action
1	Administrator	Access to Administrator Page of JFS
2	JFS	Display Administrator Screen
3	Administrator	Click on “Nhóm công việc” tab
4	JFS	Display list of category
5	Administrator	Click “Thêm Nhóm công việc” button
6	JFS	Display popup to fill name and picture of category
7	Administrator	Fill name to Name's field
8	Administrator	Choose picture's category
9	Administrator	Click button “Thêm”
10	JFS	Display update list of category
Alternative Flows		
AT1	At step 8, user click “Hủy”	
Step	Actor	Action
8.1	System	Back to list of category (not change)
Exceptions:		
EC1	At step 9, required fields are not entered	
Step	Actor	Action
9.1	JFS	Display message: “Vui lòng điền tên loại công việc” below name field
EC2	At step 9, admin did not choose picture	
Step	Actor	Action
9.2	JFS	Display message: “Xin vui long chọn ảnh” below picture field
EC3	At step 9, admin chosen file that not picture(ex: file *.doc , *.txt...)	
Step	Actor	Action

9.2	JFS	Display message: “Ảnh sai định dạng” below picture field
Priority	High	
Frequency of Use:	High	
Business Rules:	B2	
Other Information:	N/A	

Table 3.5.2.3.7.1: Admin add Category Use Case

3.5.2.3.7.2 UC061 – View list category

Use Case ID		UC061	Version	1.0					
Use Case Name		Admin view Job Category							
Created by:	NamLH	Date Created:	12/09/2016						
Primary Actor	Admin	Secondary Actor							
Description:	This function allow administrator can view list job's category								
Preconditions:	1. JFS website is available 2. Administrator logged in website								
Trigger:	Click on “Quản lý loại công việc” button								
Post conditions:	JFS show list job's category								
Normal Flow									
Step	Actor	Action							
1	Administrator	Access to Administrator Page of JFS							
2	JFS	Display Administrator Screen							
3	Administrator	Click on “Quản lý loại công việc” tab							
4	JFS	Display list of category							
Alternative Flows									
AT1	At step 3, admin click on other tab or button or hyperlink								
Step	Actor	Action							

7.1	JFS	Redirects user to chosen hyperlink or button or tab
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.3.7.2 Admin view Job Category

3.5.2.3.7.3 UC062 – Delete category

Use Case ID	UC062		Version	1.0
Use Case Name	Delete job's category			
Created by:	NamLH	Date Created:	12/09/2016	
Primary Actor	Admin	Secondary Actor		
Description:	Admin can delete job's category from list of category			
Preconditions:	3. JFS website is available 4. Administrator logged in website			
Trigger:	Click on “Xóa” button			
Post conditions:	Delete job's category successfully			
Normal Flow				
Step	Actor	Action		
1	Administrator	Access to Admin Page of JFS		
2	JFS	Display Admin Screen		
3	Administrator	Click on “Quản lý loại công việc” tab		
4	JFS	Display list of category		
5	Administrator	Select a job's category		
6	Administrator	Click “Xóa” button		

7	JFS	Display alert to confirm “ Bạn có muốn chắc chắn muốn xóa loại công việc này ?”
8	Administrator	Click button “Đồng ý”
9	JFS	Display updated list of job's category
Alternative Flows		
AT1	At step 8, user click “Huy”	
Step	Actor	Action
8.1	JFS	Back to display list of job's category
Exceptions: N/A		
Priority	High	
Frequency of Use:	High	
Business Rules:	N/A	
Other Information:	N/A	

Table 3.5.2.3.7.3: Delete job's Category Use Case

3.5.2.3.7.4 UC063 – Edit category

Use Case ID	UC063	Version	1.0
Use Case Name	Edit job's category		
Created by:	NamLH	Date Created:	12/09/2016
Primary Actor	Administrator	Secondary Actor	
Description:	Administrator job's category		
Preconditions:	1. JFS website is available 2. Administrator logged in website		
Trigger:	Click on “Sửa” button		
Post conditions:	Update job's category to database		
Normal Flow			
Step	Actor	Action	

1	Administrator	Access to Administrator Page of JFS
2	JFS	Display Administrator Screen
3	Administrator	Click on “Quản lý công việc” tab
4	JFS	Display list of category
5	Administrator	Select a job's category
6	Administrator	Click “Sửa” button
7	JFS	Display popup to fill name of category
8	Administrator	Fill other name for job's category
9	Administrator	Click “Đồng ý” button
10	JFS	Display updated list of job's category
Alternative Flows		
AT1	At step 9, user click “Hủy” button	
Step	Actor	Action
8.1	JFS	Back to the list of job's category
Exceptions:		
EC1	At step 8, Name fields are not entered	
Step	Actor	Action
8.1	JFS	Display message “Điền tên loại công việc”
Priority		
Priority		High
Frequency of Use:		
Frequency of Use:		Medium
Business Rules:		
Business Rules:		B1
Other Information:		
Other Information:		N/A

Table 3.5.2.3.7.4: Admin edit Job Category

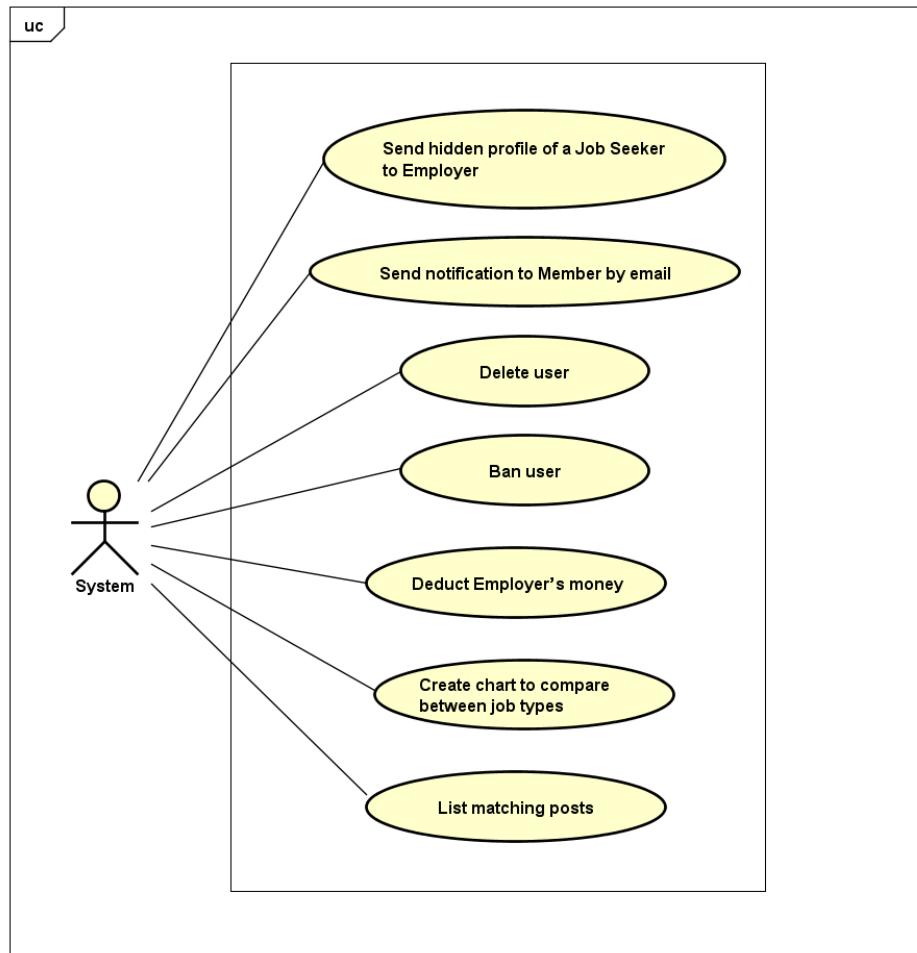
3.5.2.4 *System* システム

Figure 3.5.2.4 System use case diagram

3.6 USABILITY ユーザビリティ

- Interface should be beauty, simple, easy-to-use, clear.
- Provide an introduction to support users.

3.7 RELIABILITY 信頼性

- Search result return must be correctly.
- Use's information must be reliable.
- *The database must be backed up daily and can be recovered if necessary.*

3.8 PERFORMANCE 性能

- Maximum response time for display information must be smaller than 1 second.
- Waiting time of processing any functions will be 2 to 5 seconds.

3.9 MAINTAINABILITY 保全性

- The output of the project must include naming conventions and coding standards documentations. Code implementation must be easy to maintain.
- All code must be comment clearly, including class and method documentations.
- The design of the system must be loosely coupled that changes on some module will not affect others.

3.10 SUPPORTING INFORMATION サポート情報

N/A

4 ARCHITECTURE DESIGN アーキテクチャ設計書

4.1 INTRODUCTION はじめに

4.1.1 Purpose 目的

The purpose of this chapter is to give developers the overview and the detailed technical design of Jobs Finding System. This chapter consist of:

- Architecture overview
- Component diagram
- Detailed design
- Detailed description of components
- Database design

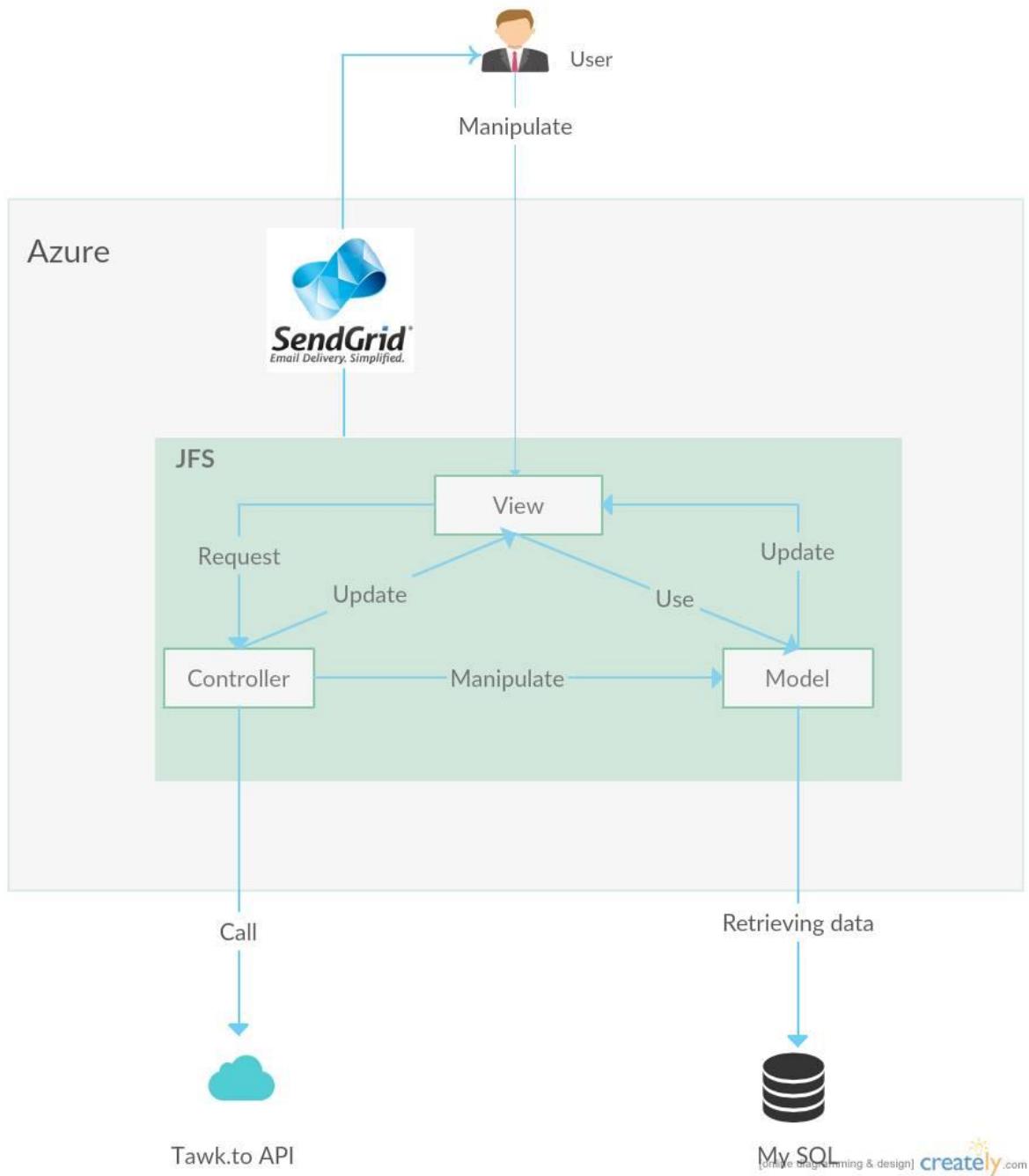
4.1.2 References 参照資料

No.	Document	Source/Web Address
1	Architecture Design Template	FPT Software Cop.
2	JFS_Software_Requirements_Specification_v1.0	Internal Resource

4.2 ARCHITECTURAL OVERVIEW アーキテクチャ概要

4.2.1 System Architecture システムアーキテクチャ

In our project we use MVC architectural model.

**Figure 4.1:** System Overview

4.3 SYSTEM ARCHITECTURE EXPLANATION システムアーキテクチャについての説明

The following is the explanation of the function and mechanism of each unit in the System Design.

Azure Cloud Service: The entire system will be deployed to Azure Cloud. Our target is developing a highly available and responsive system.

Azure SendGrid: We use SendGrid to send information to Users by email. SendGrid is a cloud-based email service that provides reliable transactional email delivery and real time analytics.

Tawk.to API: This is a free live chat API which we use to chat with all visitor on our website.

MySQL: MySQL is an open-source relational database management system. We use MySQL for adding, accessing and managing content of our system's database.

4.4 DESIGN OF JOBS FINDING SYSTEM JOBS FINDING SYSTEM の設計書

4.4.1 Database Design

データベース設計

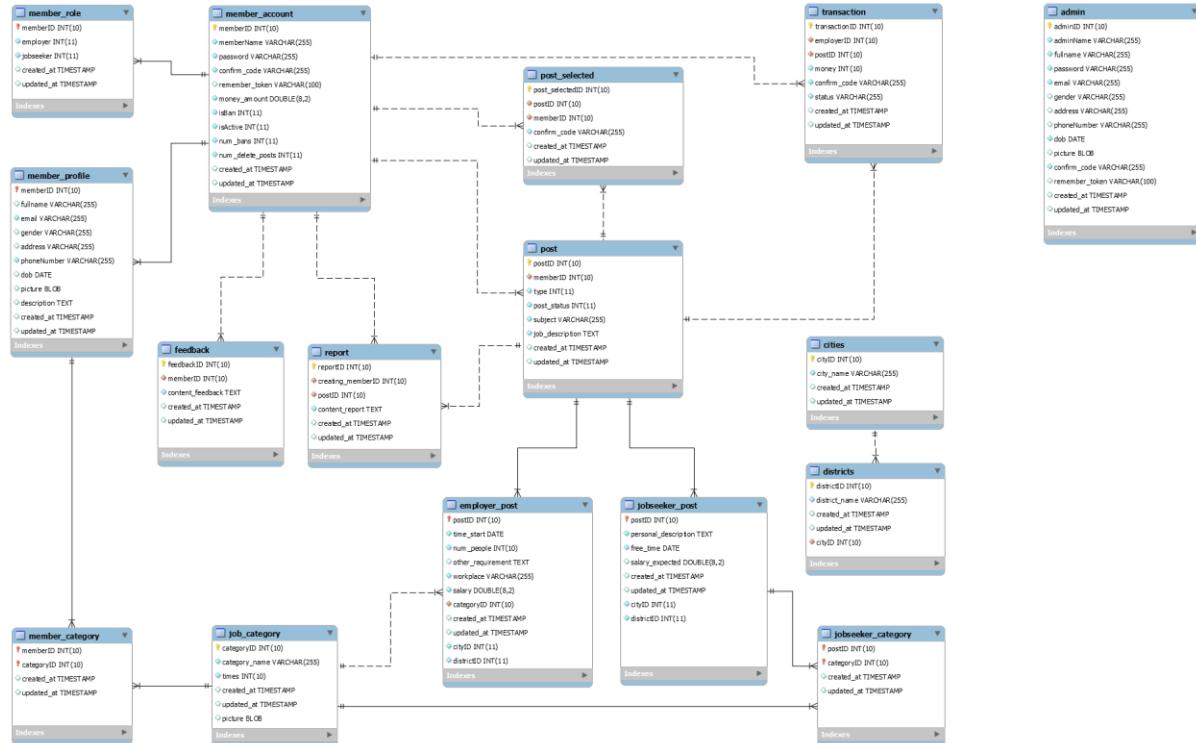


Figure 4.4.1: Table Diagram

4.4.2 Database detail データベースの詳細

4.4.2.1 Member_account table

No	Field Name	Type	Allow Null	Description
1	memberID	Int	No	Member's ID
2	memberName	String	No	Member's name
3	password	String	No	Member's password
4	money_amount	Float	No	Member's money
5	remember_token	Bool	No	0: Don't remember account 1: Remember account
6	confirm_code	String		For member to verify account.

7	isBan	Int	No	Member's ban status (0: unban,1: ban)
8	isActive	Int	No	Member's active status (0: inactive,1: active)
9	num_bans	Int	No	Number of times that member has been ban.
10	num_delete_posts	Int	No	Numbers of the posts that is deleted

Table 4.4.2.1: Member_account table

4.4.2.2 Member_profile table

No	Field Name	Type	Allow Null	Description
1	memberID	Int	No	Member's ID
2	fullname	String	No	Member's full name
3	picture	Binary		Member's profile image
4	email	String	No	Member's email
5	address	String		Member's address
6	gender	String		Member's gender
7	phoneNumber	String	No	Member's phone number
8	dob	Datetime		Member's date of birth
9	description	String		Member's description

Table 4.4.2.2: Member_profile table

4.4.2.3 Member_role table

No	Field Name	Type	Allow Null	Description
1	memberID	Int	No	Member's ID
2	jobseeker	Int	No	Default is 1
3	employer	Int	No	Default is 1

Table 4.4.2.3: Member_role table

4.4.2.4 Admin table

No	Field Name	Type	Allow Null	Description
1	adminID	Int	No	Admin's ID
2	adminName	String	No	Admin's name
3	password	String	No	Admin's password
4	fullname	String	No	Admin's fullname
5	email	String	No	Admin's email
6	picture	String		Admin's profile image
7	address	String		Admin's address
8	gender	String		Admin's gender
9	phoneNumber	String	No	Admin's phone number
10	dob	Datetime		Admin's date of birth
11	remember_token	Bool	No	0: Don't remember account 1: Remember account
12	confirm_code	String		For admin verify account

Table 4.4.2.4: Admin table

4.4.2.5 Post_selected table

No	Field Name	Type	Allow Null	Description
1	post_seletedID	Int	No	Auto increasing the ID
2	postID	Int	No	Post's ID that the post is seleted
3	memberID	Int	No	Member's ID who selected this post
4	confirm_code	String	No	Code send for jobseeker by mail to accept or reject an employer

Table 4.4.2.5: Post_selected table

4.4.2.6 Feedback table

No	Field Name	Type	Allow Null	Description
1	feedbackID	Int	No	Feedback's ID
2	memberID	Int	No	Member's ID who created feedback

3	created_at	Timestamp	No	Real time that member create report
4	content_feedback	String	No	Content of feedback

Table 4.4.2.6: Feedback table

4.4.2.7 Job_Category table

No	Field Name	Type	Allow Null	Description
1	categoryID	Int	No	Category's ID
2	Category_name	Int	No	Category's name
3	times	Int	No	Number that a category is selected
4	picture	Binary	No	Picture of each category to display in view

Table 4.4.2.7: Job_category table

4.4.2.8 JobSeeker_Category table

No	Field Name	Type	Allow Null	Description
1	categoryID	Int	No	category's ID
2	postID	Int	No	jobseekerPost's ID

Table 4.4.2.8: JobSeeker_Category table

4.4.2.9 Report table

No	Field Name	Type	Allow Null	Description
1	reportID	Int	No	Report's ID
2	creating_memberID	Int	No	Member who create a report
3	postID	Int	No	Post that was reported
4	created_at	Timestamp	No	Time that member create a report
5	content_report	Text	No	Content of report

Table 4.4.2.9: Report table

4.4.2.10 Transaction table

No	Field Name	Type	Allow Null	Description
1	transactionID	Int	No	Transaction's ID
2	postID	Int	No	Post's ID
3	employerID	Int	No	Employer's ID
4	jobSeekerID	Int	No	JobSeeker's ID
5	money	Float	No	Money paid for each transaction
6	confirm_code	String	No	Code send for jobseeker by mail to accept or reject an employer
7	status	String	No	Status of a transaction
8	updated_at	Timestamps	No	Time that a transaction was completed

Table 4.4.2.10: Transaction table

4.4.2.11 Post table

No	Field Name	Type	Allow Null	Description
1	postID	Int	No	Post's ID
2	memberID	Int	No	Member who create this post
3	type	Int	No	Employer's post or JobSeeker's post
4	created_at	Timestamp	No	Time that a post was created
5	post_status	String	No	Status of post (0: Off, 1: On)
6	subject	String	No	Subject of a post
7	job_description	Text	No	Description of a job

Table 4.4.2.11: Post table

4.4.2.12 Employer_post table

No	Field Name	Type	Allow Null	Description
1	postID	Int	No	ID of employer's post
2	time_start	Date	No	Time that a job will start.
3	num_people	Int		Number of people need to complete job
4	other_requirement	Text		Other_requirement of employer
5	workplace	String		Workplace that a job take place
6	salary	Float		Salary that Employer pay for JobSeeker
7	categoryID	Int	No	Category's ID
8	cityID	Int	No	City's ID
9	districtID	Int	No	District's ID

Table 4.4.2.12: Employer_post table

4.4.2.13 JobSeeker_post table

No	Field Name	Type	Allow Null	Description
1	postID	Int	No	ID of jobseeker's post
2	personal_description	Text	No	Describe personal of job seeker.
3	free_time	Date	No	Day that jobseeker can work
4	salary_expected	Float		Salary that jobseeker want to be paid
5	cityID	Int	No	City's ID
6	districtID	Int	No	District's ID

Table 4.4.2.13: Job Seeker table

4.4.2.14 Member_category table

No	Field Name	Type	Allow Null	Description
1	memberID	Int	No	member's ID
2	categoryID	Int	No	Category's ID

Table 4.4.2.14: Member_category table

4.4.2.15 Cities table

No	Field Name	Type	Allow Null	Description
1	cityID	Int	No	City's ID
2	city_name	String	No	Name of city

Table 4.4.2.15: Cities table

4.4.2.16 Districts table

No	Field Name	Type	Allow Null	Description
1	districtID	Int	No	District's ID
2	district_name	String	No	Name of district
3	cityID	Int	No	City's ID

Table 4.4.2.16: Districts table

4.5 COMPONENT DIAGRAM コンポーネント図

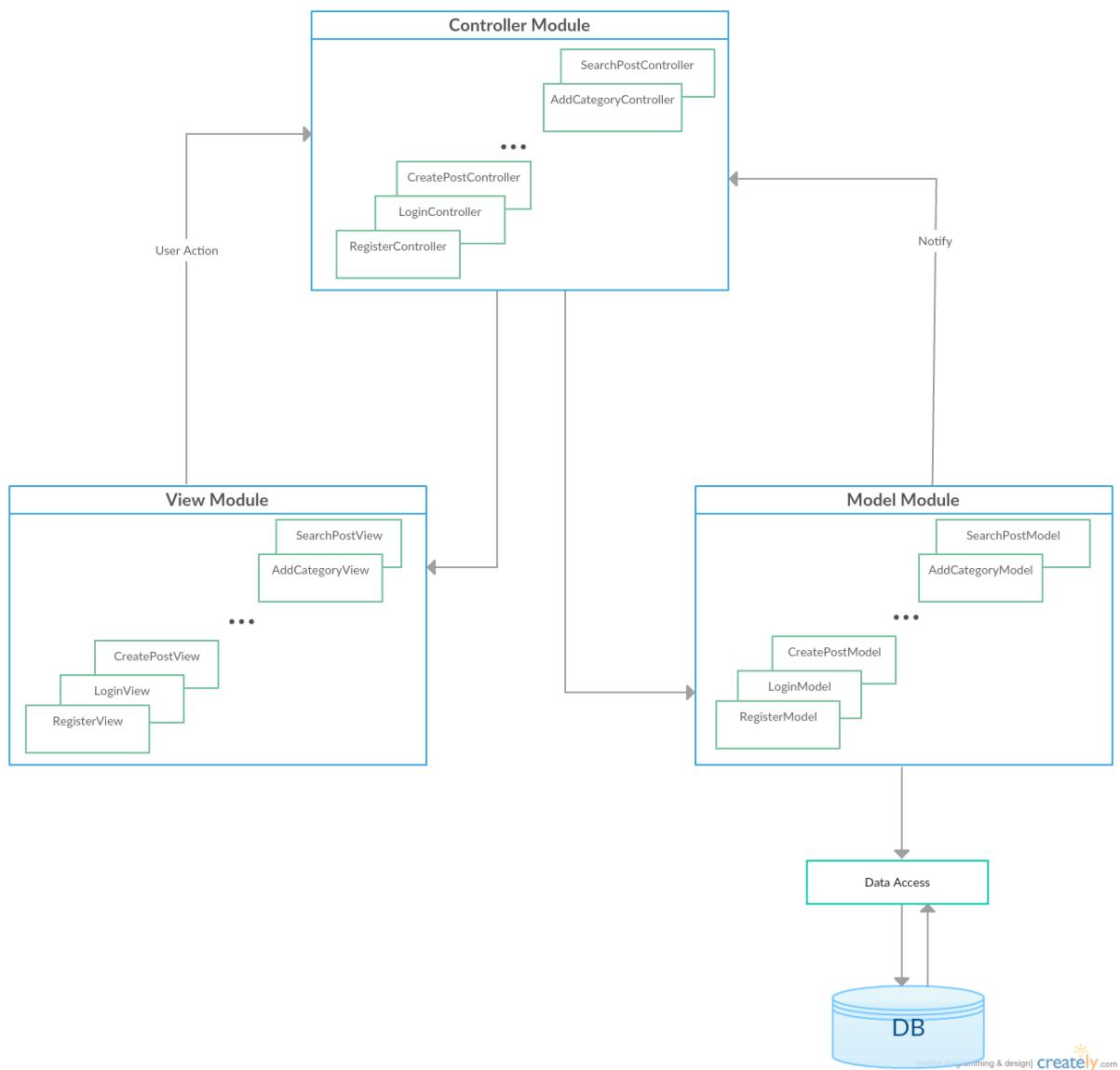


Figure 4.5 Component Diagram

The **Model** manages behavior of application in terms of its problem domain including data, logic and rules of application. It takes request from controller layer and makes changes to database of the system.

The **View** provides the user interface element of the application. It takes information from model and display in a form that suitable for users.

The **Controller** receives user action and makes calls to model object.

4.6 CRC CARDS CRC カード

CRC for Model - View – Controller

4.6.1 Model モデル

4.6.1.1 Admin

Admin	
Create new admin Implement admin login to system, logout from system Allow admin update admin's profile	

4.6.1.2 Member Account

MemberAccount	
Create new member's account Implement login, logout, change password Allow admin implement ban, unban member Allow admin implement add money to member's financial account Allow admin send notify to member.	MemberProfile MemberRole Post Feedback Report PostSelected Transaction

4.6.1.3 MemberProfile

MemberProfile	
Get information of member Update member's profile	MemberAccount

4.6.1.4 MemberRole

MemberRole	
Create two type of member : Employer and JobSeeker	MemberAccount

4.6.1.5 Post

Post	
Create a post Edit information of a post Get information of a post Get a list posts	MemberAccount Employer_post JobSeeker_post PostSelected Transaction Report

4.6.1.6 Employer_post

Employer_post	
Get information of an employer_post by ID	Post
Get list of employer post.	JobCategory
Add information to an employe_post	
Search employer post	
Matching for an employer_post	

4.6.1.7 JobSeeker_post

JobSeeker_post	
Get information of an jobSeeker_post by ID	Post
Get list of jobSeeker post.	JobCategory
Add information to an jobSeeker_post	
Search jobSeeker post	
Matching for an jobSeeker_post	

4.6.1.8 JobCategory

JobCategory	
Get information of a job category	Post
Get all category	Employer_post
Add new category	JobSeeker_post
Edit category	
Delete category	

4.6.1.9 Feedback

Feedback	
Create a feedback	MemberAccount
Get list feedback	Post
Admin delete feedback	

4.6.1.10 Report

Report	
Create a report	MemberAccount
Get list report	Post
Admin delete report	

4.6.1.11 PostSelected

PostSelected	
Add postSelected when jobSeeker select a post of employer	MemberAccount
JobSeeker choose employer	Post
Get list following post of a member	

4.6.1.12 Transaction

Transaction	
Create pending transaction when employer select a post of jobSeeker Create transaction when employer choose a jobSeeker Get all transaction's information	MemberAccount Post

4.6.1.13 MemberCategory

MemberCategory	
Display specialize of member	MemberProfile JobCategory

4.6.1.14 JobSeekerCategory

JobSeekerCategory	
Get list category of a jobSeeker_post	JobSeeker_Post JobCategory

4.6.1.15 City

City	
Get list city	Post

4.6.1.16 District

District	
Get list district	Post

4.6.2 View ビュー

4.6.2.1 HomePageView

HomePageView

<p>HomePageView screen allow Guest and member to do functions:</p> <ul style="list-style-type: none"> • Button “Đăng kí” for Guest sign up become to member of system • Button “Đăng nhập” for Member login into system • Button “Đăng xuất” for Member logout from system • Search bar for Guest and Member search Posts follow keyword, location or category. • List Posts latest for Guest and Member can view • Service package of system for Guest and Member can view 	PostController Post JobSeekerPost EmployerPost
---	---

4.6.2.2 ListPostView

ListPostView	
ListPost view allow Guest and Member to do functions: <ul style="list-style-type: none"> • Search bar to search Posts • View the Posts sorted by time 	PostController Post JobSeekerPost EmployerPost

4.6.2.3 PostDetailOfJobSeekerView

PostDetailOfJobSeekerView	
Display all information of a Post includes: <ul style="list-style-type: none"> • Subject text • Job category text • Job description text • Salary text • Time to do job date • Address to do job text • Display matching posts • Display people choose that post 	PostController Post JobSeekerPost EmployerPost

4.6.2.4 PostDetailOfEmployerView

PostDetailOfEmployerView	
Display all information of a Post includes: <ul style="list-style-type: none"> • Subject text • Job category text • Job description text • Salary text 	PostController Post JobSeekerPost EmployerPost

<ul style="list-style-type: none"> • Time to do job date • Address to do job text • Display matching posts • Display people choose that post • Select button to choose people 	
--	--

4.6.2.5 ProfileView

ProfileView	
<p>Profile view to member see and update personal information The screen display personal information of member includes:</p> <ul style="list-style-type: none"> • FullName text box • Email text box • Address text box • Phone number text box • Description myself text • Gender radio button • Date of birth Date 	MemberController MemberProfile MemberAccount

4.6.2.6 LoginView

LoginView	
<p>Login view to Member login into system The screen includes:</p> <ul style="list-style-type: none"> • Username text box • Password text box • Remember me check box • Forgot password link • Login button • Cancel button • Register link 	MemberLoginController MemberAccount

4.6.2.7 RegisterView

RegisterView

<p>Register view to Guest login into system</p> <p>The screen includes:</p> <ul style="list-style-type: none"> • Username text box • Password text box • Confirm text box • Email text box • Phone number text box • Address text box • Date of birth Date • Gender radio button • Register button • Cancel button 	MemberRegisterController MemberProfile MemberAccount
--	--

4.6.2.8 ChangePasswordView

ChangePasswordView Change password view to Member change password of account The screen includes: <ul style="list-style-type: none"> • Username text box • Password text box • Confirm password • Ok button • Cancel button 	MemberController MemberAccount
---	-----------------------------------

4.6.2.9 PersonalPageView

PersonalPageView PersonalPageView screen allow member to do functions: <ul style="list-style-type: none"> • Manage Post link to manage personal post • Profile link to manage profile • Transaction link to see transaction history • Create post link to create a new posts 	PostController MemberController MemberAccount
--	---

4.6.2.10 AdminManageAllGUI

AdminManageAllGUI <ul style="list-style-type: none"> • Display Admin's name and profile icon. • Display button “Đăng nhập”. • Display button “Đăng xuất”. • Display link “Quên mật khẩu”. • Display button “Gửi thông báo”. • Display button “Thêm quản trị viên”. • Display button “Quản lý thành viên”. • Display button “Quản lý bài đăng”. 	AdminModel AdminAuthenController
---	-------------------------------------

<ul style="list-style-type: none"> Display button “Quản lý góp ý”. Display button “Quản lý doanh thu”. Display button “Quản lý thành viên”. Display button “Quản lý báo cáo”. Display button “Quản lý loại công việc”. 	
---	--

4.6.2.11 AdminManagePostGUI

AdminManagePostGUI	
<ul style="list-style-type: none"> Display all Posts of members. Click on each post's link to go to Post's detail. Display button “Bài đăng tìm người”. Click this button to go to “View and Search Employer's Posts” page. Display button “Bài đăng tìm việc”. Click this button to go to “View and Search Job Seeker's Posts” page. Display button “Xóa” next to each post. Click this button to delete post. 	PostModel PostController

4.6.2.12 AdminManageCategoryGUI

AdminManageCategoryGUI	
<ul style="list-style-type: none"> Display all Job Category. Display button “Thêm loại công việc”. Display button “Sửa”. Display button “Xóa”. 	CategoryModel CategoryController

4.6.2.13 AdminManageMemberGUI

AdminManageMemberGUI	
<ul style="list-style-type: none"> Display all Member's account name. Click on each account to go to profile. Display button “Ban”, “Unban”, “Thêm tiền”, “Xóa”, “Gửi thông báo” next to each account. Button “Tìm kiếm” to search for account. 	MemberAccountModel MemberProfileModel ManageMemberController

4.6.2.14 AdminManageStatisticGUI

AdminManageStatisticGUI	
<ul style="list-style-type: none"> Display chart. 	StatisticController

4.6.2.15 AdminManageFeedbackGUI

AdminManageFeedbackGUI	
<ul style="list-style-type: none"> Display all feedback about system. Click on each feedback to go to detail. Display button “Xóa” next to each feedback to delete. 	FeedbackModel FeedbackController

4.6.3 Controller コントローラ

4.6.3.1 HomeController

HomeController	
Allow the user to view all post Allow the user to search post Allow the user to perform Register Use Case Allow the user to perform Login Use Case Allow the user to perform Logout Use Case	Post EmployerPost JobSeekerPost

4.6.3.2 AdminAuthenController

AdminAuthenController	
Allow the admin to perform Login Use Case Allow the admin to perform Logout Use Case Allow the admin to perform addAdmin Use Case	Admin

4.6.3.3 AdminController

AdminController	
Allow the admin to perform update admin profile Use Case Allow the admin to show admin's profile Use Case	Admin

4.6.3.4 ManageMemberController

ManageMemberController	
Allow the admin to perform search member Use Case Allow the admin to perform view memberProfile Use Case Allow the admin to perform delete member Use Case Allow the admin to perform banMember Use Case Allow the admin to perform unbanMember Use Case Allow the admin to perform addMoney Use Case Allow the admin to perform sendNotify Use Case	MemberAccount MemberProfile MemberRole

--	--

4.6.3.5 ManagePostController

ManagePostController	
Allow admin view all post Allow the admin to perform delete posts Use Case Allow admin to search post	Post Employer_post JobSeeker_post

4.6.3.6 CategoryController

CategoryController	
Allow the admin to perform view category Use Case Allow the admin to perform add category Use Case Allow the admin to perform edit category Use Case Allow the admin to perform delete category Use Case	JobCategory

4.6.3.7 StatisticController

StatisticController	
Allow the admin to perform view Statistic Use Case Allow the admin to perform view chart s Use Case	Transaction JobCategory

4.6.3.8 MemberLoginController

MemberLoginController	
Allow the member to perform login Use Case Allow the admin to perform forget password Use Case Allow the admin to perform logout Use Case	MemberAccount MemberProfile

4.6.3.9 MemberRegisterController

MemberRegisterController	
Allow the guest to perform register Use Case Allow system confirm guess clicked verify link in mail	MemberAccount MemberProfile MemerRole

4.6.3.10 MemberPostController

MemberPostController	
Allow the member to perform create post Use Case Allow the member to perform edit post Use Case Allow the member to perform delete post Use Case Allow the member to select suitable jobSeeker Use Case Allow the member to apply suitable job Use Case	Post Employer_post JobSeeker_post

4.6.3.11 MemberController

MemberController	
Allow the member to perform show personal profile post Use Case Allow the member to perform edit profile post Use Case Allow the member to perform show other member's profile Use Case	MemberProfile

4.6.3.12 PostController

PostController	
Allow the user to perform search post Use Case Allow the user to perform display list posts Use Case Allow the member to perform display other post detail Use Case Allow the member to perform display own post detail Use Case..	Post JobSeeker_Category Employer_post JobSeeker_Post PostSelected

4.6.3.13 FeedbackController

FeedbackController

Allow the admin to perform view feedback Use Case Allow the member to perform create feedback Use Case Allow the admin to perform delete feedback Use Case	Feedback MemberAccount
--	---------------------------

4.6.3.14 ReportController

ReportController

Allow the admin to perform view report Use Case Allow the member to perform report post Use Case	Report MemberAccount Post
---	---------------------------------

4.7 DETAIL DESIGN 詳細設計

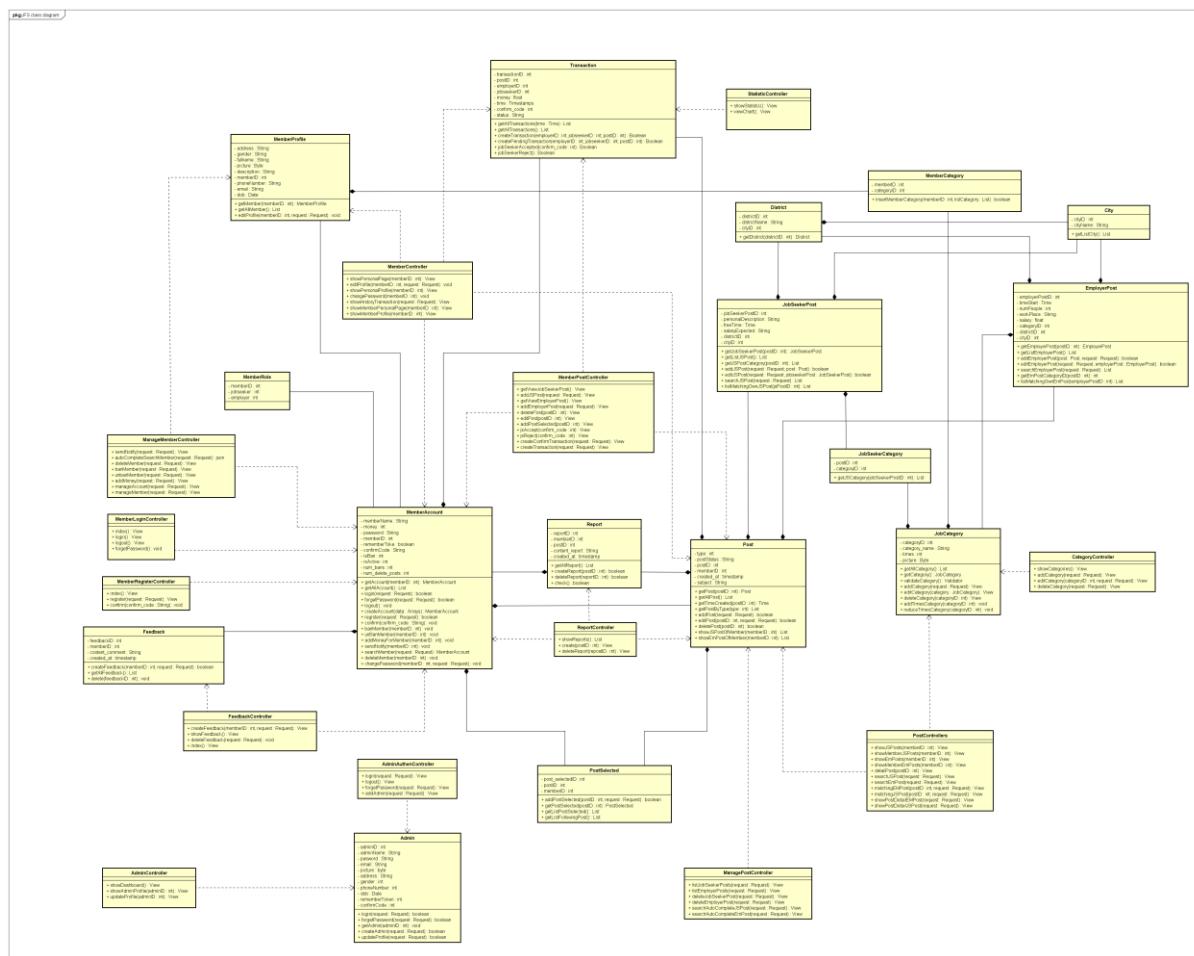


Table 4.7 Class Diagram

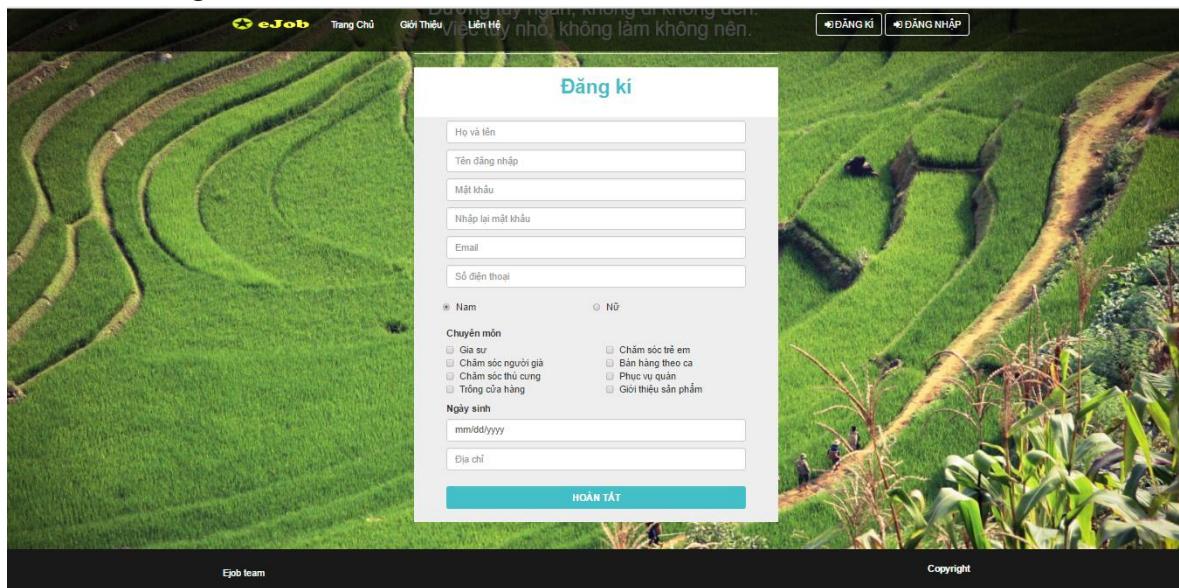
4.7.1 Guest ゲスト

4.7.1.1 Register

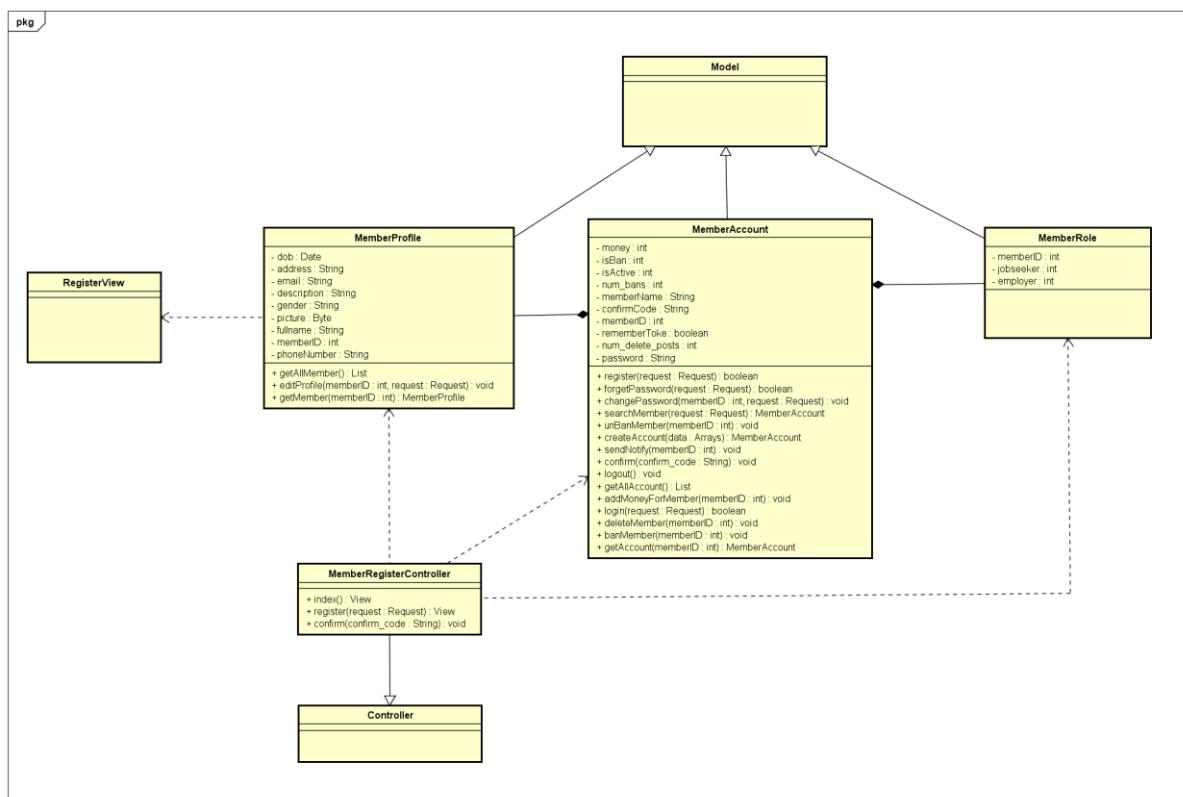
4.7.1.1.1 Description

Guest can create new account to use more function as member

4.7.1.1.2 Design



4.7.1.1.3 Class Diagram

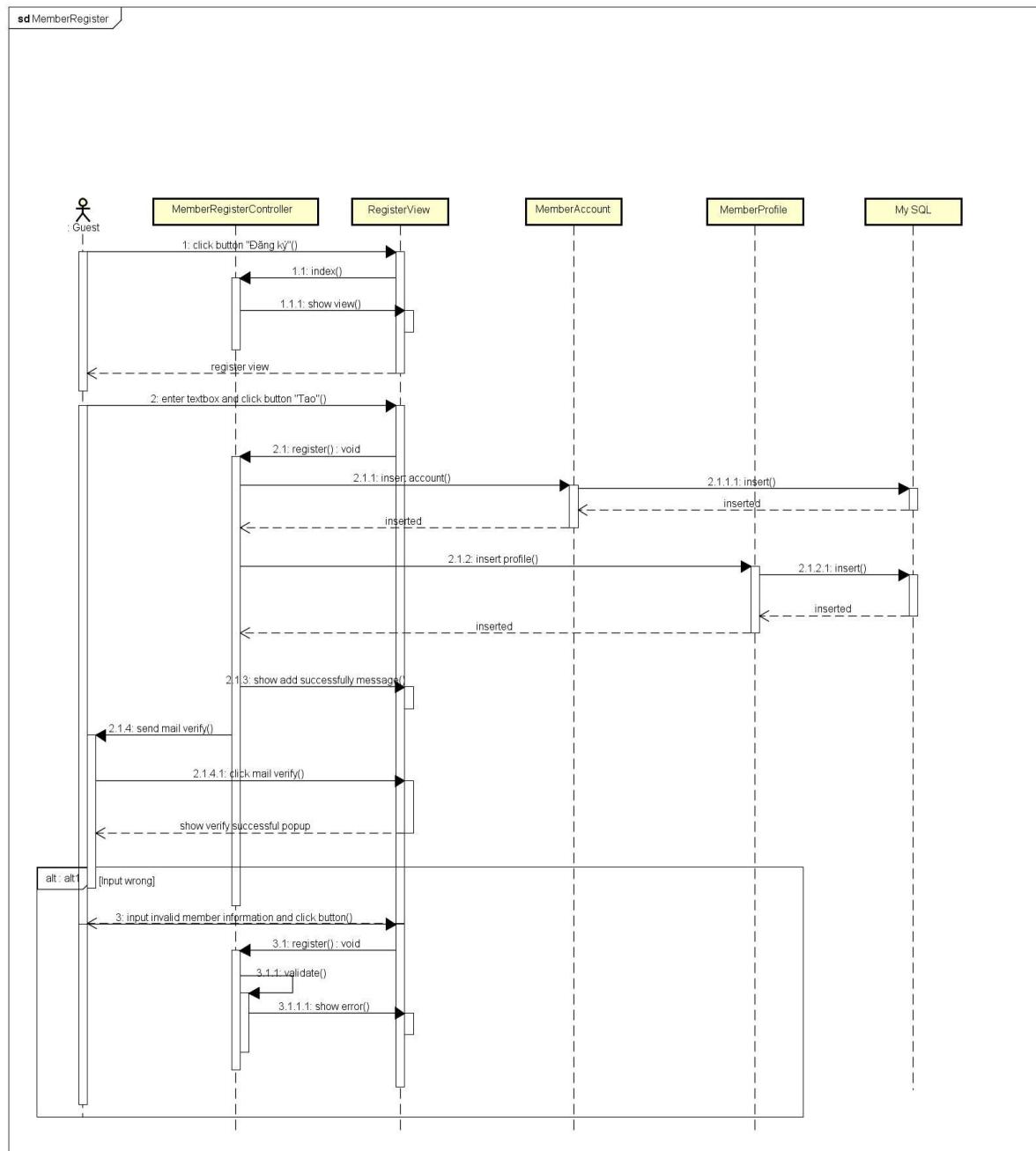


4.7.1.1.4 Class Specifications

Class	MemberRegisterController
Description	Control Guest register action
Base Class	Controller
Source File	MemberRegisterController.php
Namespace	app/https/controllers/

Attributes	N/A			
Methods	Name	Input	Output	Description
	register	request	View	Create and validate account, send verify mail to registered account

4.7.1.1.5 Sequence Diagram

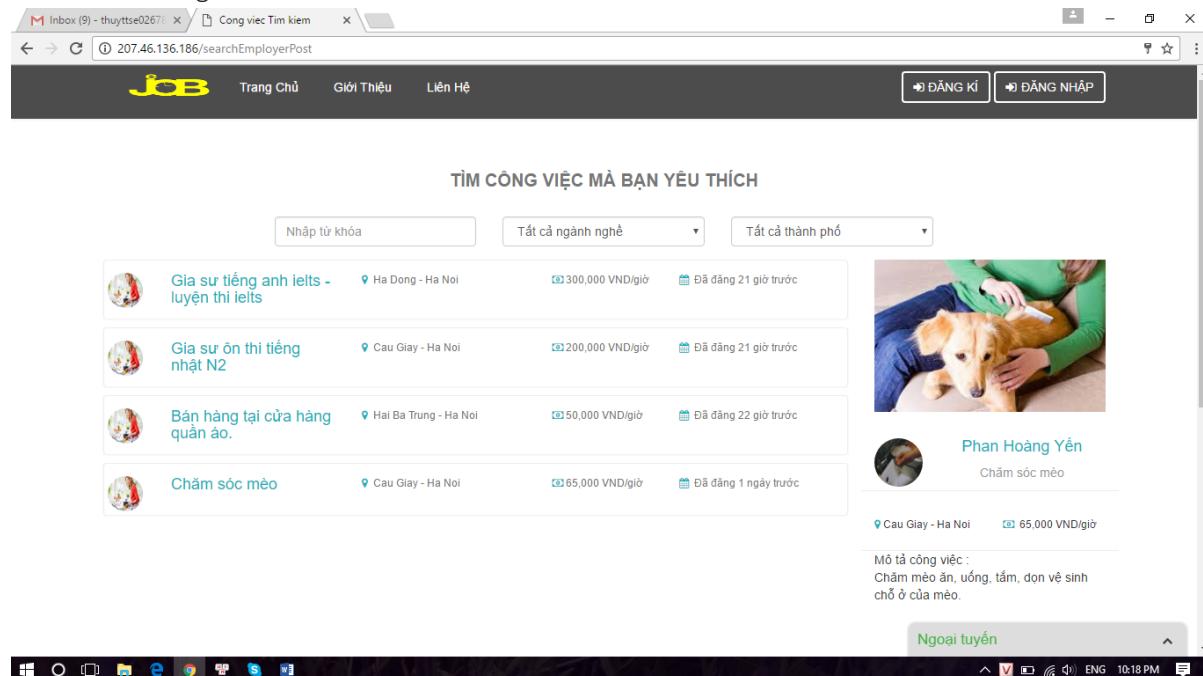


4.7.1.2 View Employer's Posts

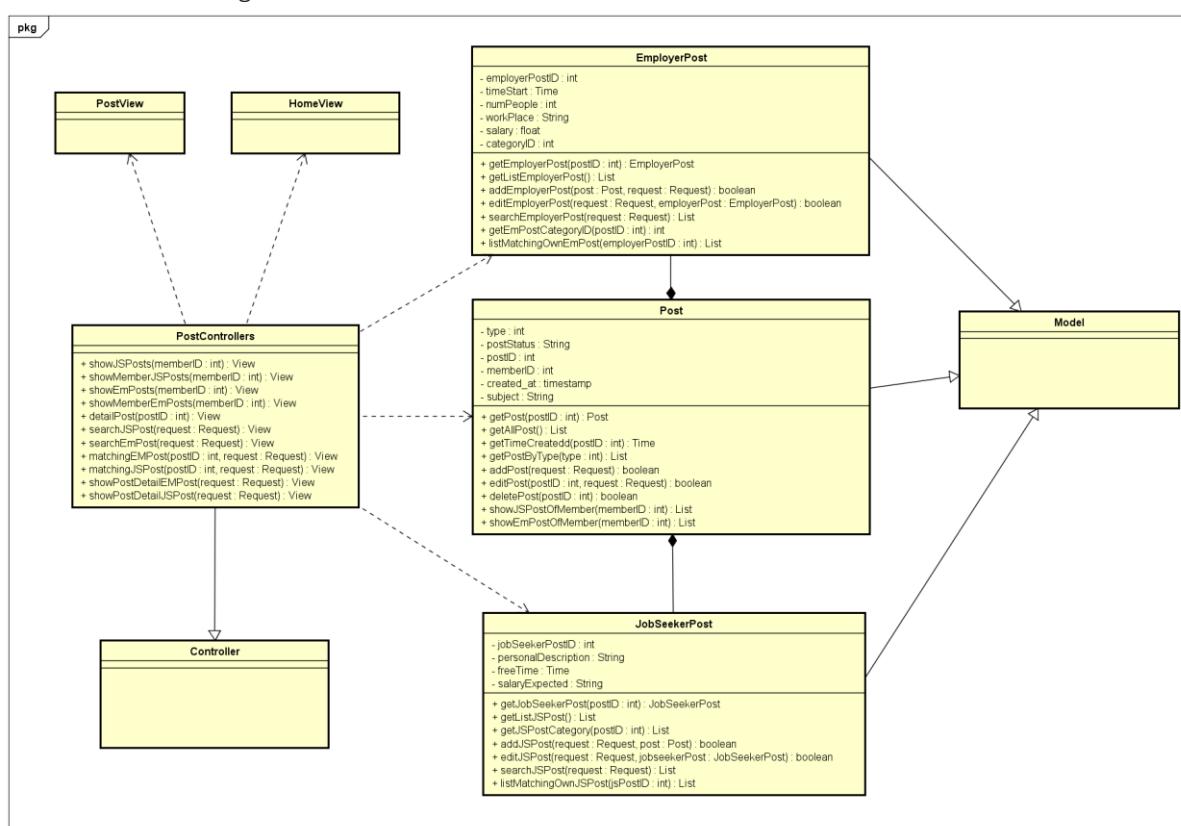
4.7.1.2.1 Description

This function allows Guest view Employer's Posts

4.7.1.2.2 Design



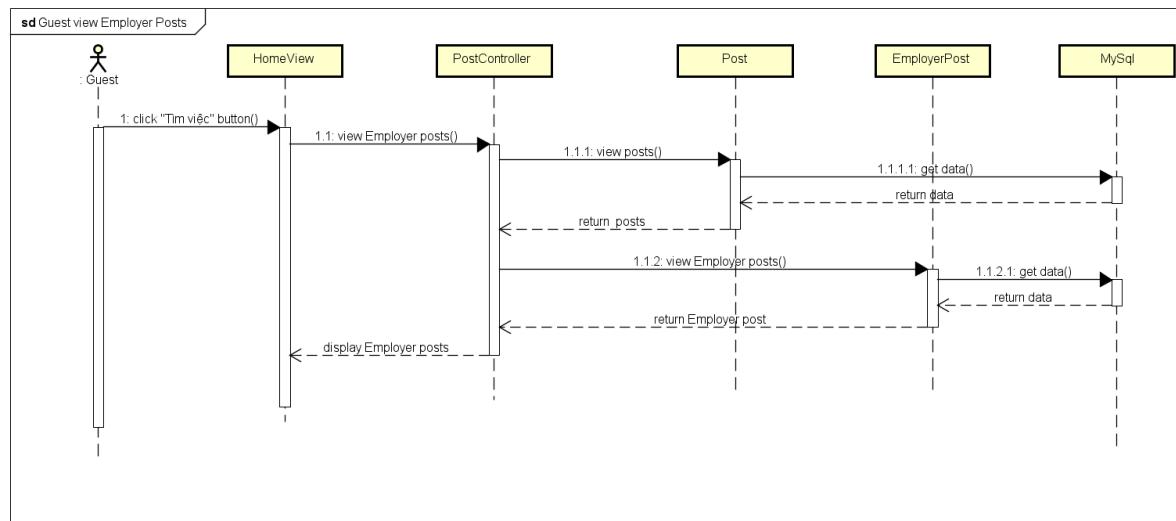
4.7.1.2.3 Class Diagram



4.7.1.2.4 Class Specifications

Class	PostController			
Description	Control Guest view Employer's Post action			
Base Class	Controller			
Source File	PostController.php			
Namespace	app/http/controllers/post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchEmPost	request	View list employer's post	Display all Post of Employer (when guest click button "tìm việc" on Homepage)

4.7.1.2.5 Sequence Diagram



4.7.1.3 View Job Seeker's Posts

4.7.1.3.1 Description

This function allows Guest to view Job Seeker's Posts

4.7.1.3.2 Design

The screenshot shows a web application interface for searching job posts. The main heading is "TÌM NGƯỜI LÀM MONG MUỐN". Below it are five search results:

- Chăm sóc thực cung (Cầu Giấy - Hà Nội) - 222.222VND/giờ (Đã đăng 4 giờ trước)
- Chăm sóc trẻ em (Ba Đình - Hà Nội) - 150.000VND/giờ (Đã đăng 2 ngày trước)
- Tôi cần một công việc như nội trợ hoặc chăm sóc trẻ em (Quận 1 - TP Hồ Chí Minh) - 150.000VND/giờ (Đã đăng 2 ngày trước)
- Tôi cần một công việc như nội trợ hoặc chăm sóc trẻ em (Hà Đông - Hà Nội) - 200.000VND/giờ (Đã đăng 2 ngày trước)
- Tôi cần một công việc như nội trợ hoặc chăm sóc trẻ em (Đống Đa - Hà Nội) - 150.000VND/giờ (Đã đăng 2 ngày trước)

On the right side, there is a sidebar with a profile picture of a person named "Dương Đặng Duy" and the text "Chuyên đồ Cận thận".

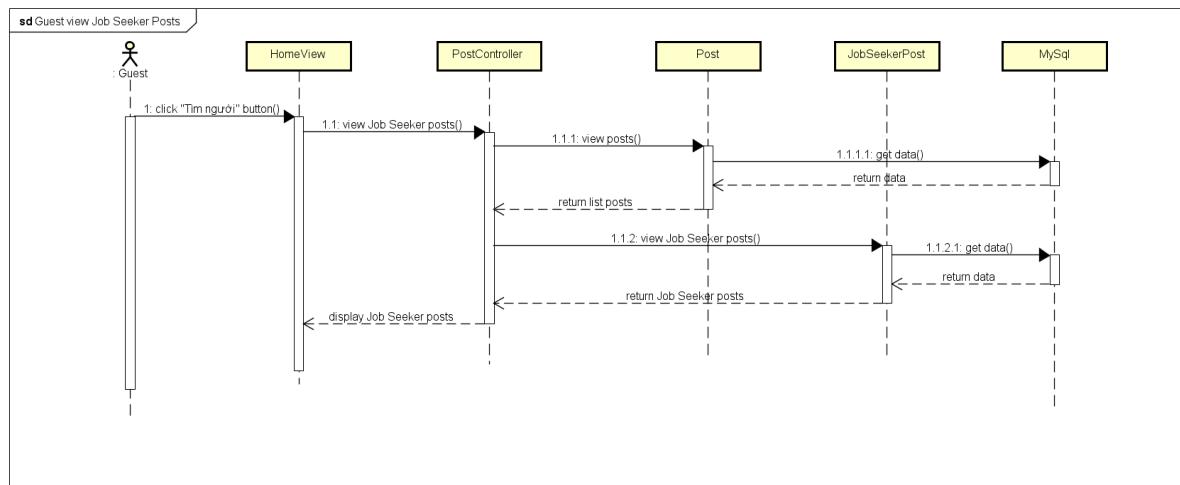
4.7.1.3.3 Class Diagram

Refer to class diagram of view Employer's Posts in 4.7.1.2 View Employer's Posts

4.7.1.3.4 Class Specifications

Class	PostController			
Description	Control Guest view JobSeeker's Post action			
Base Class	Controller			
Source File	PostController.php			
Namespace	app/http/controllers/post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchJSPost	request	View list JobSeeker's Post	Display all Post of JobSeeker (when guess click button "tìm người" on Homepage)

4.7.1.3.5 Sequence Diagram

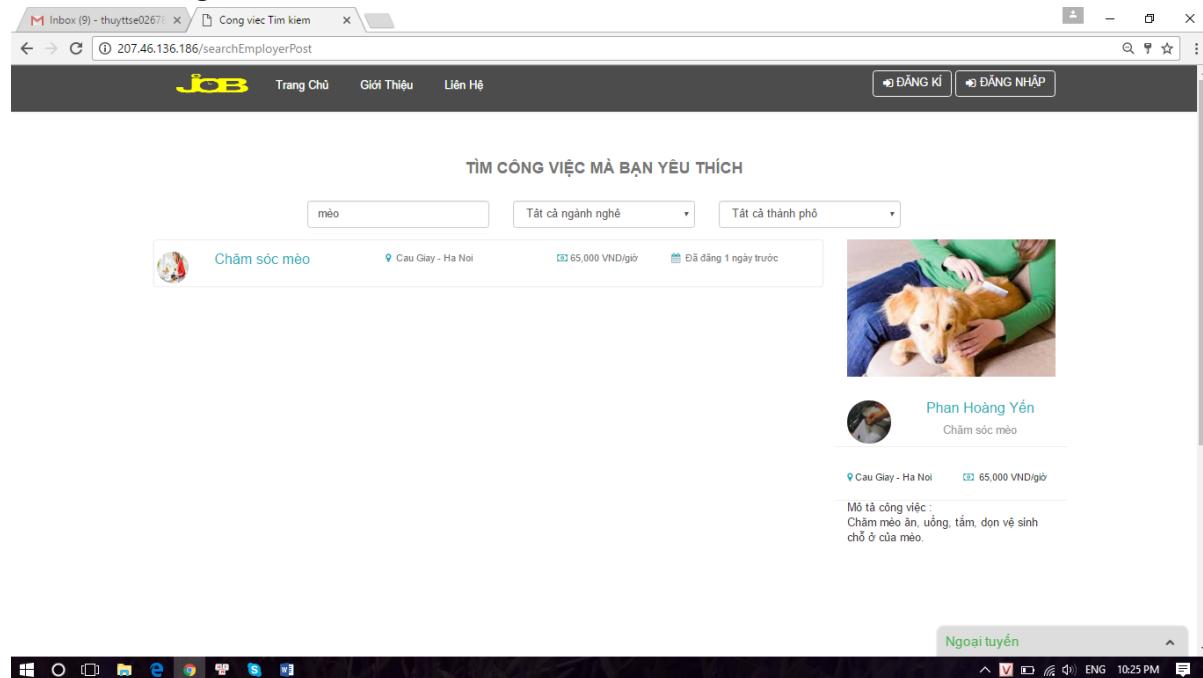


4.7.1.4 Search Employer's Posts

4.7.1.4.1 Description

This function allows Guest to searching for specific Employer's Posts on JFS by the information of category, location or any keyword.

4.7.1.4.2 Design



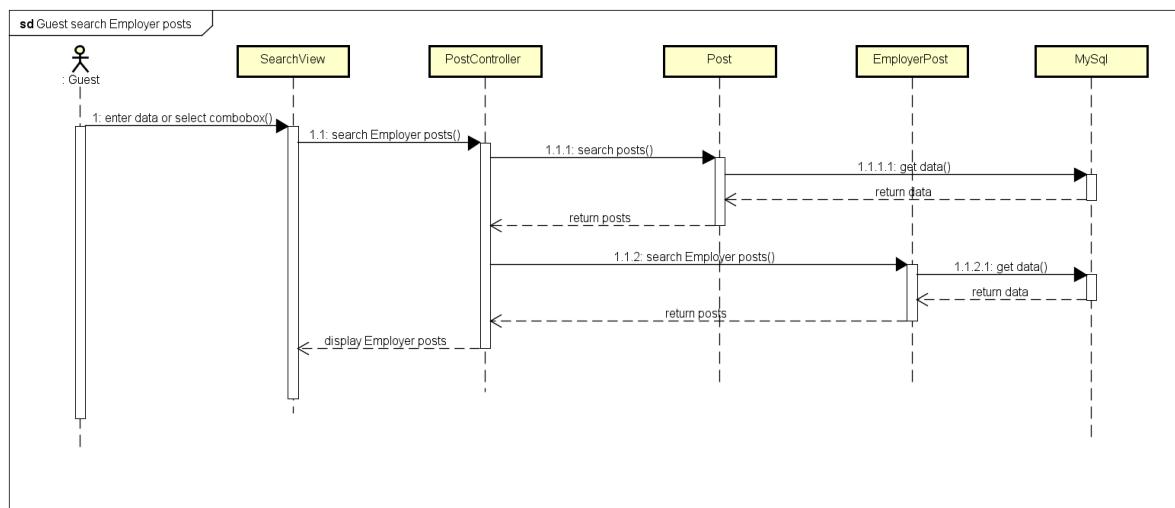
4.7.1.4.3 Class Diagram

Refer to class diagram of view Employer's Posts in 4.7.1.2 View Employer's Posts

4.7.1.4.4 Class Specifications

Class	PostController			
Description	Control Guest search Employer's Post action			
Base Class	Controller			
Source File	PostController.php			
Namespace	app/http/controllers/post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchEmPost	request	View list employer's post	Guest type keyword or click select search Post of Employer by category, location

4.7.1.4.5 Sequence Diagram



4.7.1.5 Search Job Seeker's Posts

4.7.1.5.1 Description

This function allows Guest to searching for specific Job Seeker's Posts on JFS by information of category, location or keyword.

4.7.1.5.2 Design

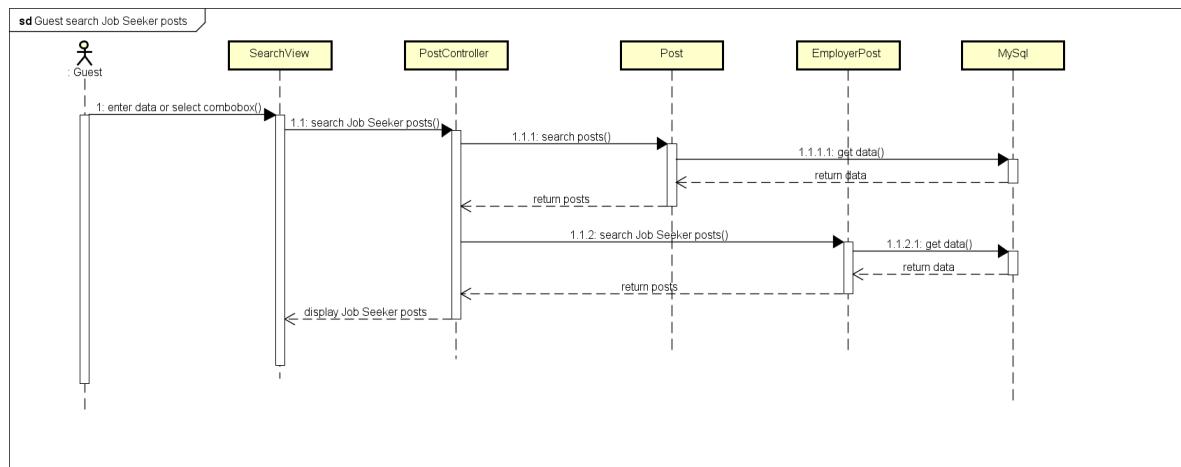
4.7.1.5.3 Class Diagram

Refer to class diagram of view Employer's Posts in 4.7.1.2 View Employer's Posts

4.7.1.5.4 Class Specifications

Class	PostController			
Description	Control Guest seach JobSeeker's Post action			
Base Class	Controller			
Source File	PostController.php			
Namespace	app/http/controllers/post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchJSPost	request	View list jobSeeker's post	Guest type keyword or click select search Post of JobSeeker by category, location or keyword

4.7.1.5.5 Sequence Diagram

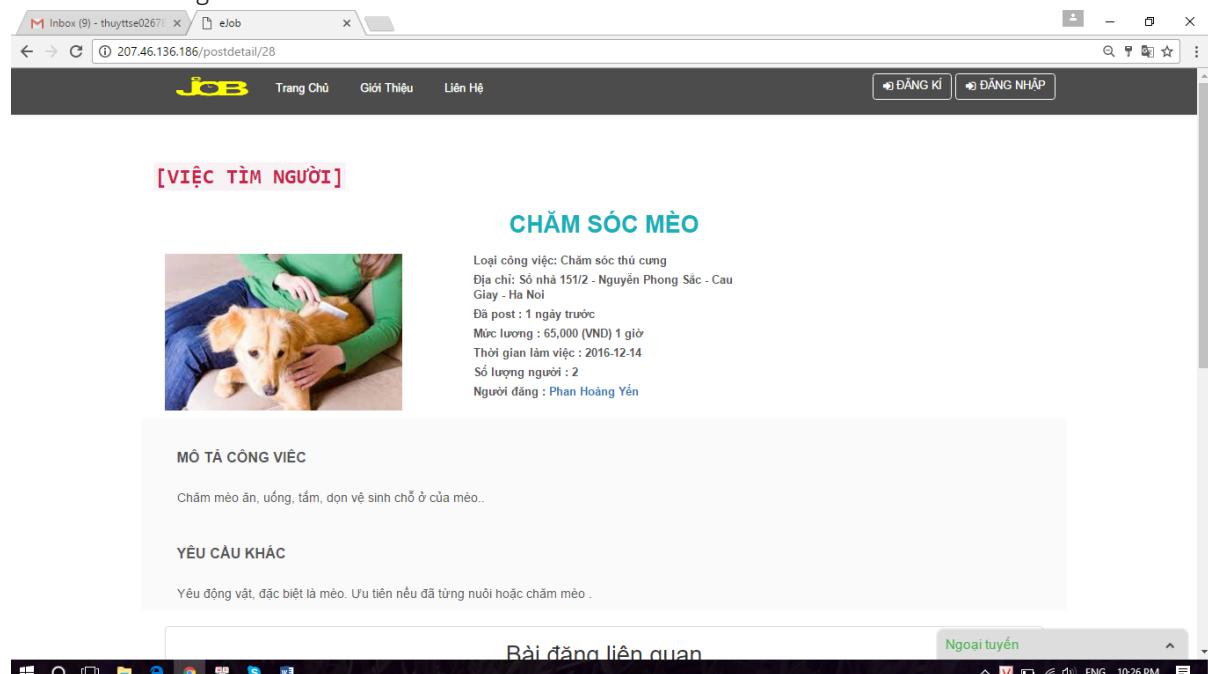


4.7.1.6 View detail Post

4.7.1.6.1 Description

This function allows Guest to view detail information of a post.

4.7.1.6.2 Design



4.7.1.6.3 Class Diagram

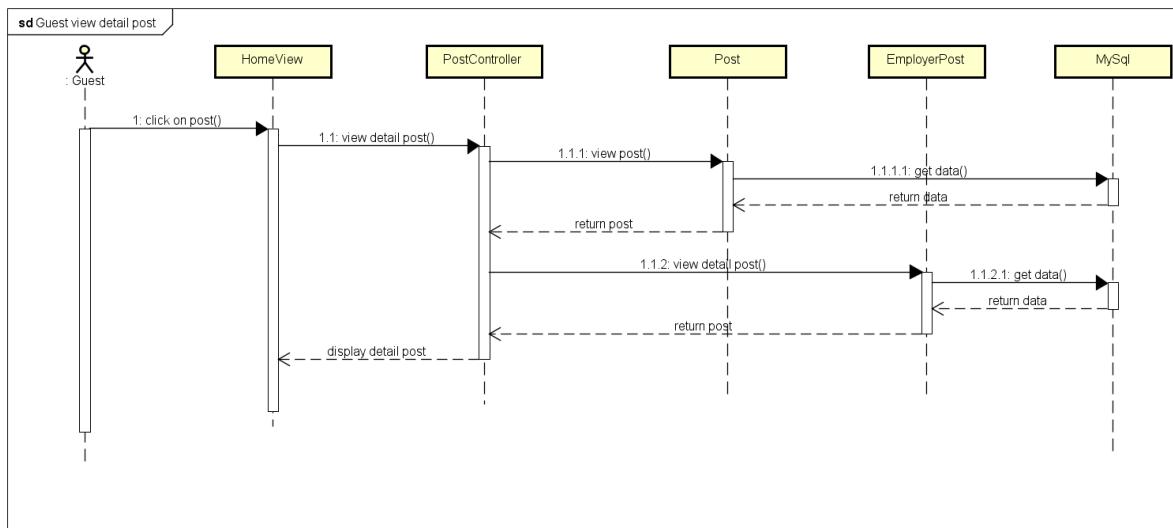
Refer to class diagram of view Employer's Posts in 4.7.1.2 View Employer's Posts

4.7.1.6.4 Class Specifications

Class	PostController
Description	Control Guest view detail Post action
Base Class	Controller

Source File	PostController.php		
Namespace	app/http/controllers/post		
Attributes	N/A		
Methods	Name	Input	Output
	detailPost	postID	View detail of post Display post detail of Employer or Job Seeker

4.7.1.6.5 Sequence Diagram



4.7.2 Member メンバー

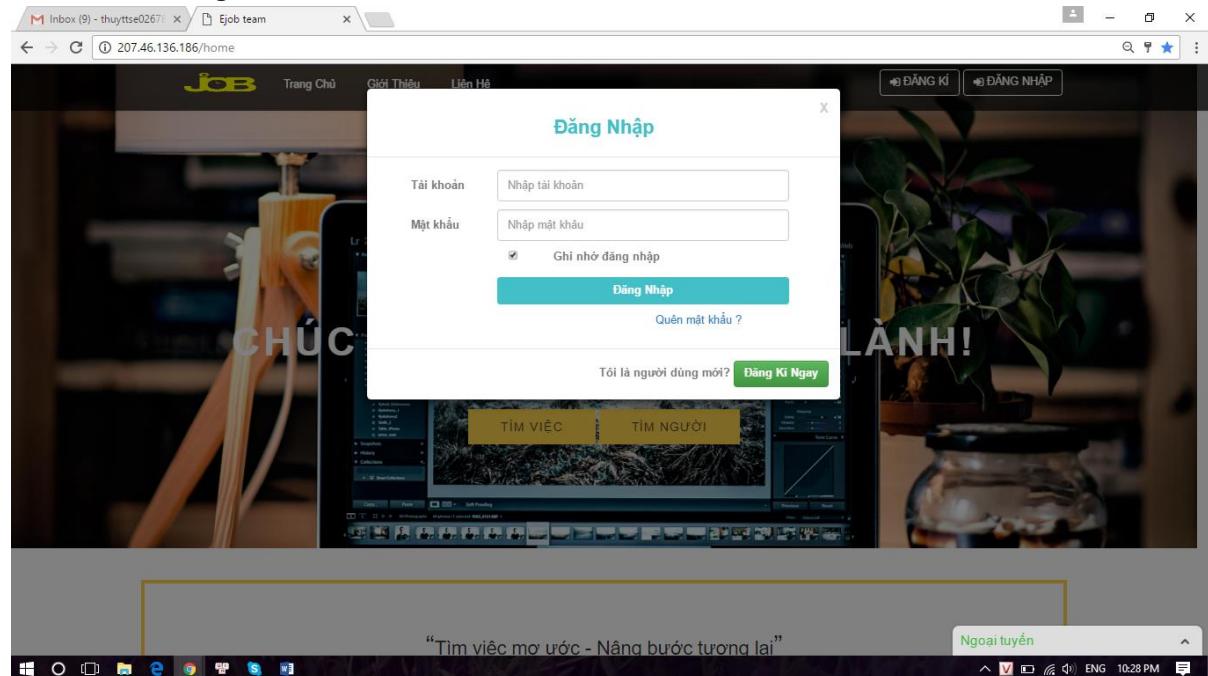
4.7.2.1 Common

4.7.2.1.1 Login

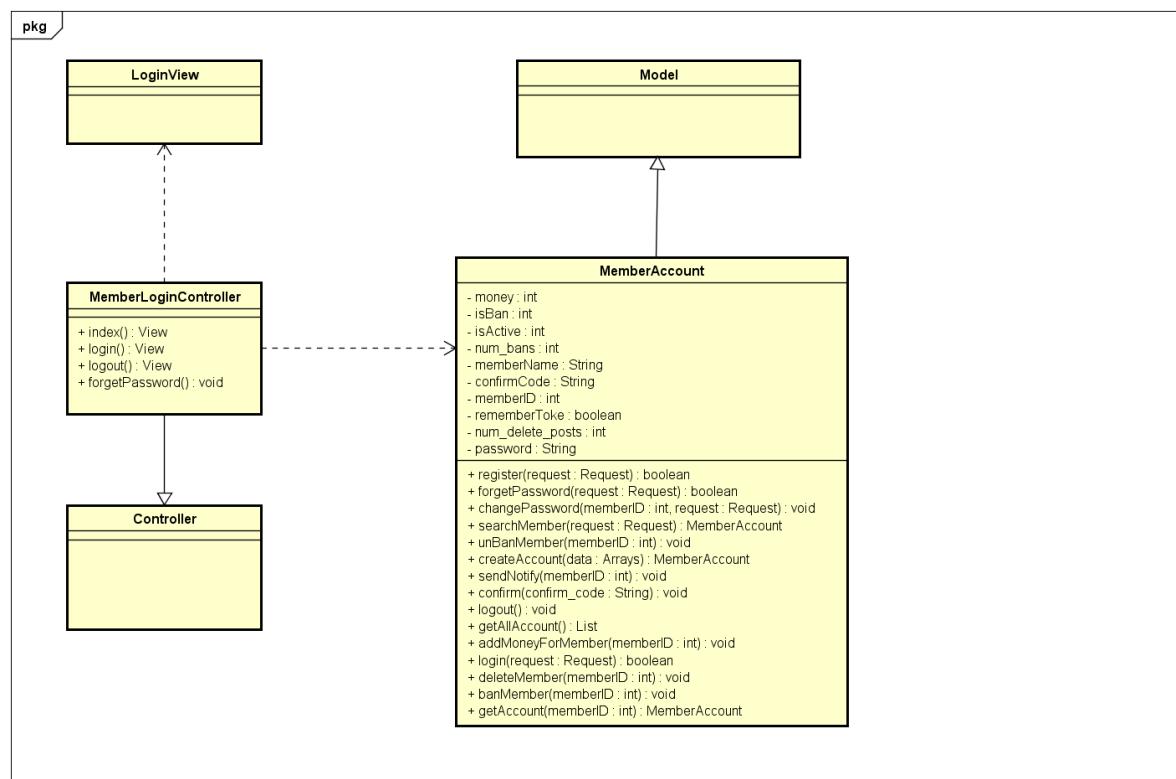
4.7.2.1.1.1 Description

Member have to login by a registered account to use specific functions of JFS.

4.7.2.1.1.2 Design



4.7.2.1.1.3 Class Diagram

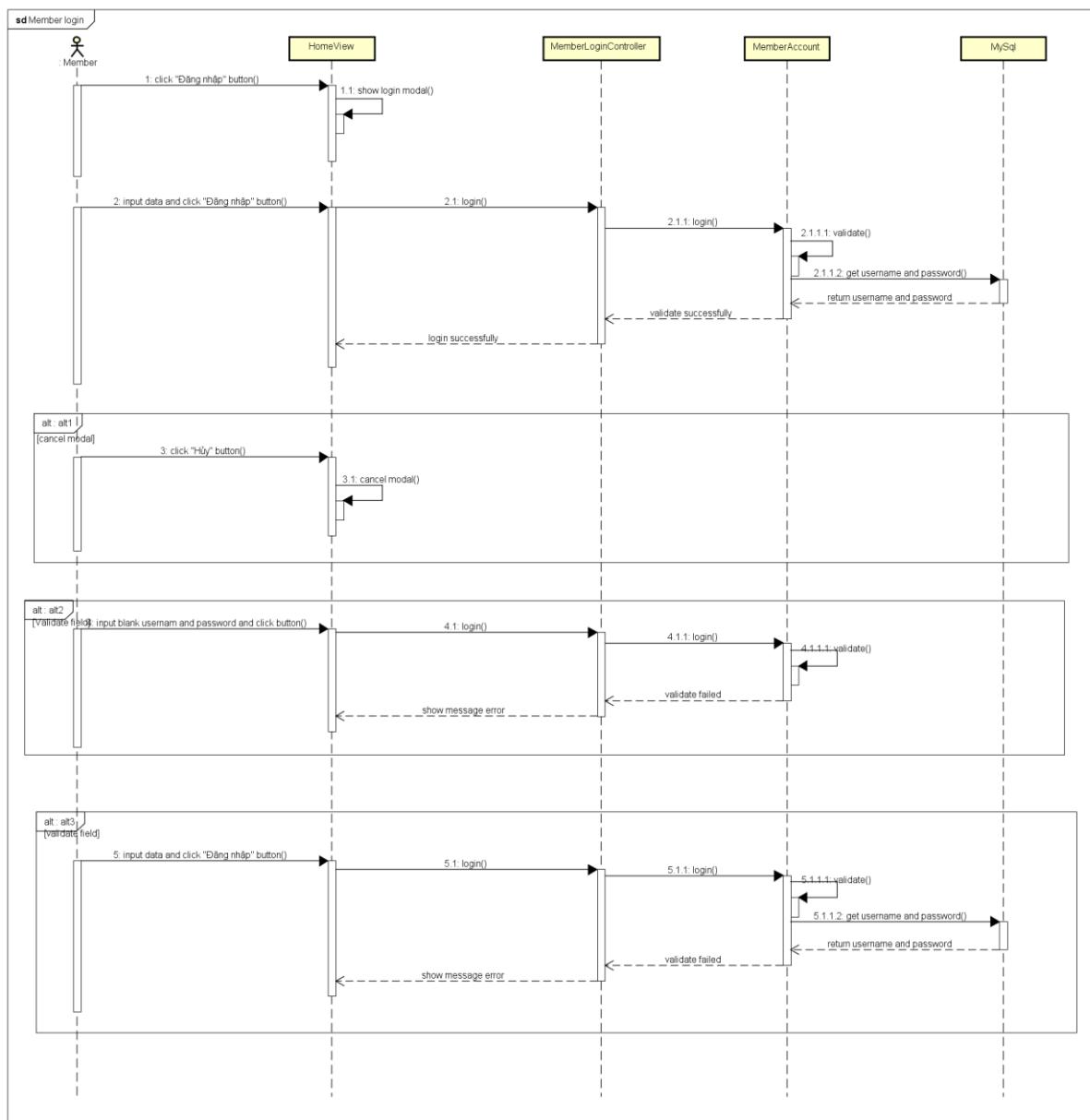


4.7.2.1.1.4 Class Specifications

Class	MemberLoginController
Description	Control member login action
Base Class	Controller

Source File	MemberLoginController.php		
Namespace	app/http/controllers/member		
Attributes	N/A		
Methods	Name	Input	Output
	login	N/A	View

4.7.2.1.1.5 Sequence Diagram



4.7.2.1.2 Logout

4.7.2.1.2.1 Description

This function allows a member to logout.

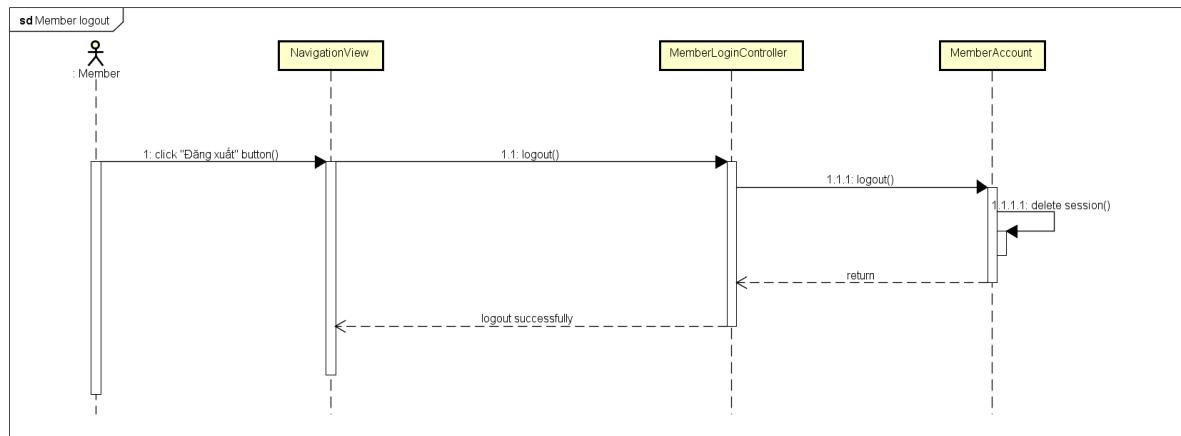
4.7.2.1.2.2 Class Diagram

Refer to class diagram of Member Login in 4.7.2.1.1Login

4.7.2.1.2.3 Class Specifications

Class	MemberLoginController			
Description	Control member logout action			
Base Class	Controller			
Source File	MemberLoginController.php			
Namespace	app/http/controllers/member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	logout	N/A	void	Logout member from system

4.7.2.1.2.4 Sequence Diagram

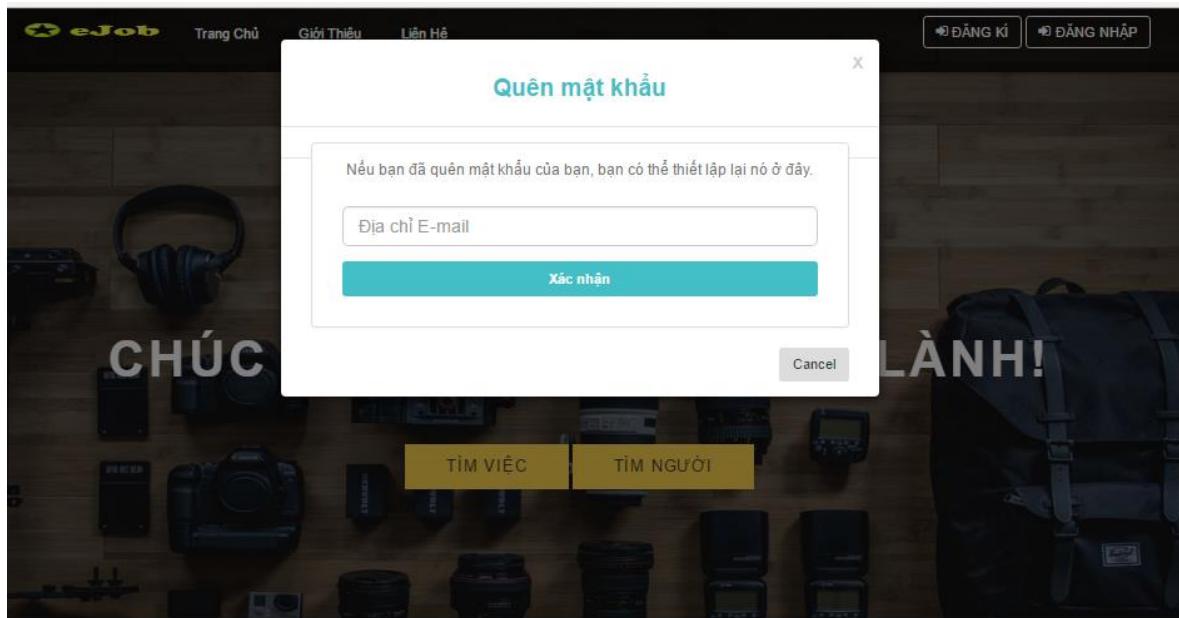


4.7.2.1.3 Forgot password

4.7.2.1.3.1 Description

When Member forgot password that used to login to system, use this function to receive new password by email.

4.7.2.1.3.2 Design



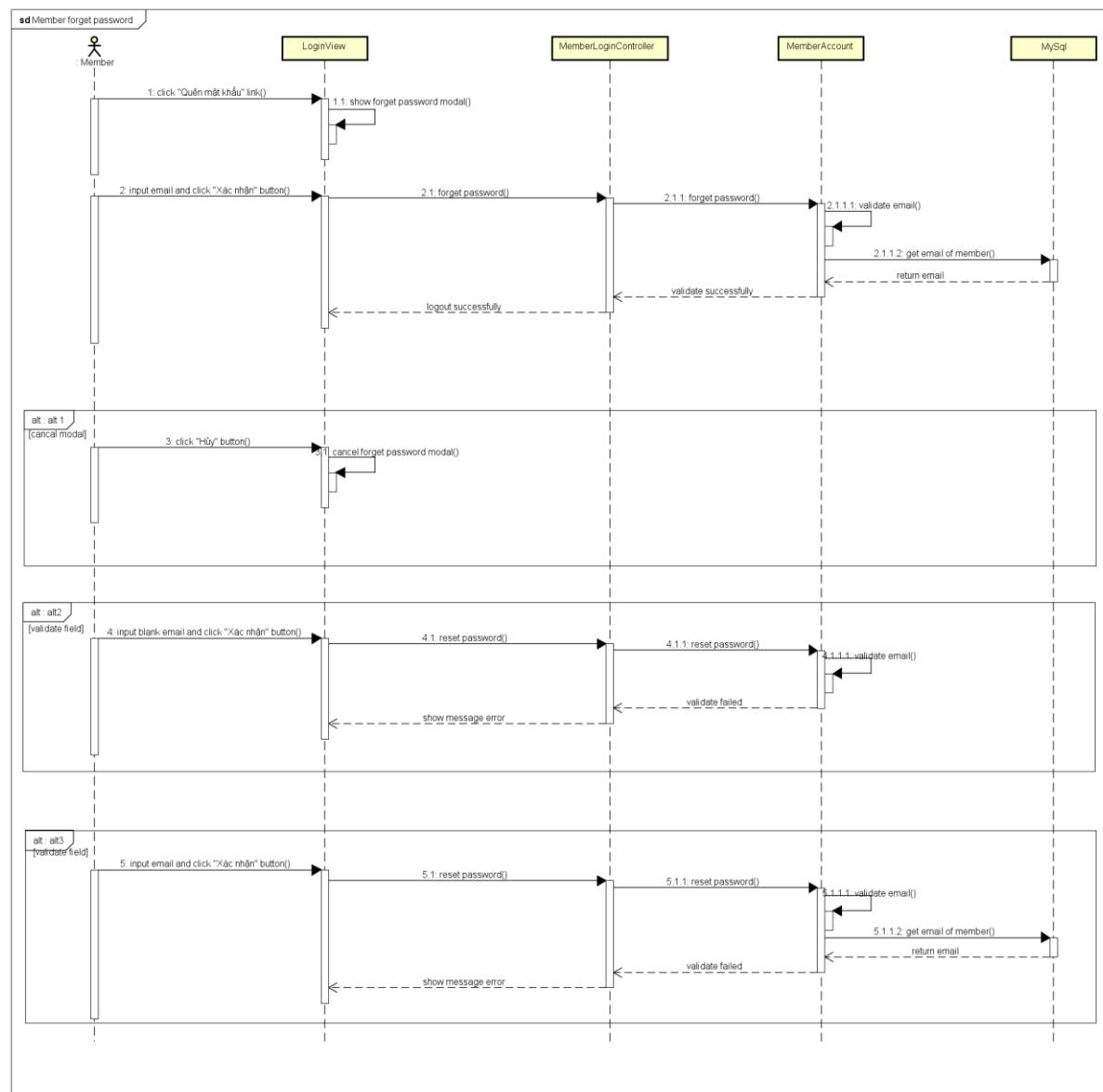
4.7.2.1.3.3 Class Diagram

Refer to class diagram of Member Login in 4.7.2.1.1Login

4.7.2.1.3.4 Class Specifications

Class	MemberLoginController			
Description	Control member forgot password action			
Base Class	Controller			
Source File	MemberLoginController.php			
Namespace	app/http/controllers/member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	forgetPassword	N/A	void	Get email from member and send new password to this email

4.7.2.1.3.5 Sequence Diagram

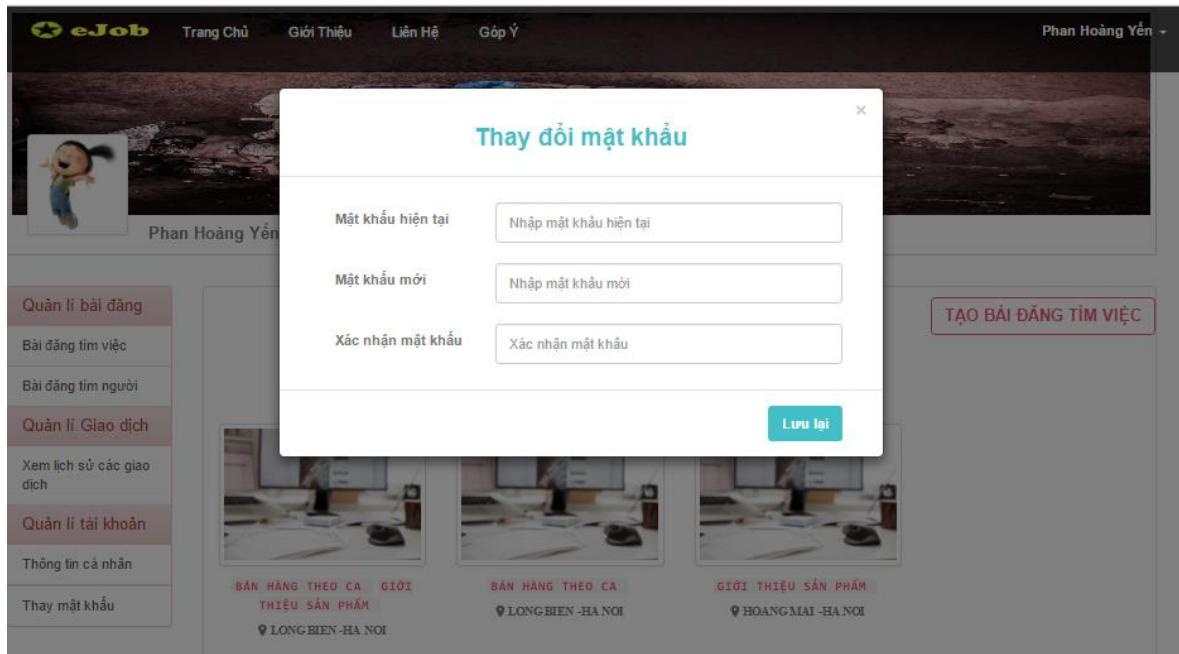


4.7.2.1.4 Change password

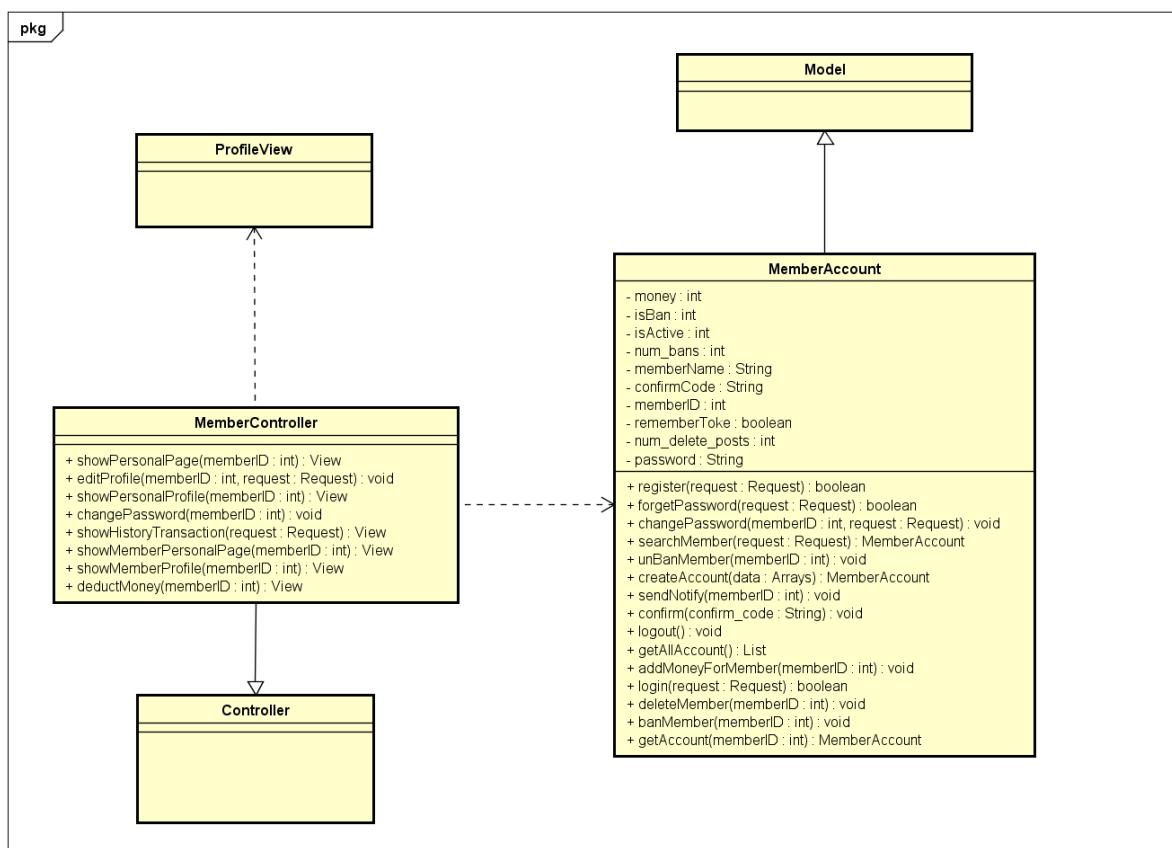
4.7.2.1.4.1 Description

Member can change their account's password.

4.7.2.1.4.2 Design



4.7.2.1.4.3 Class Diagram

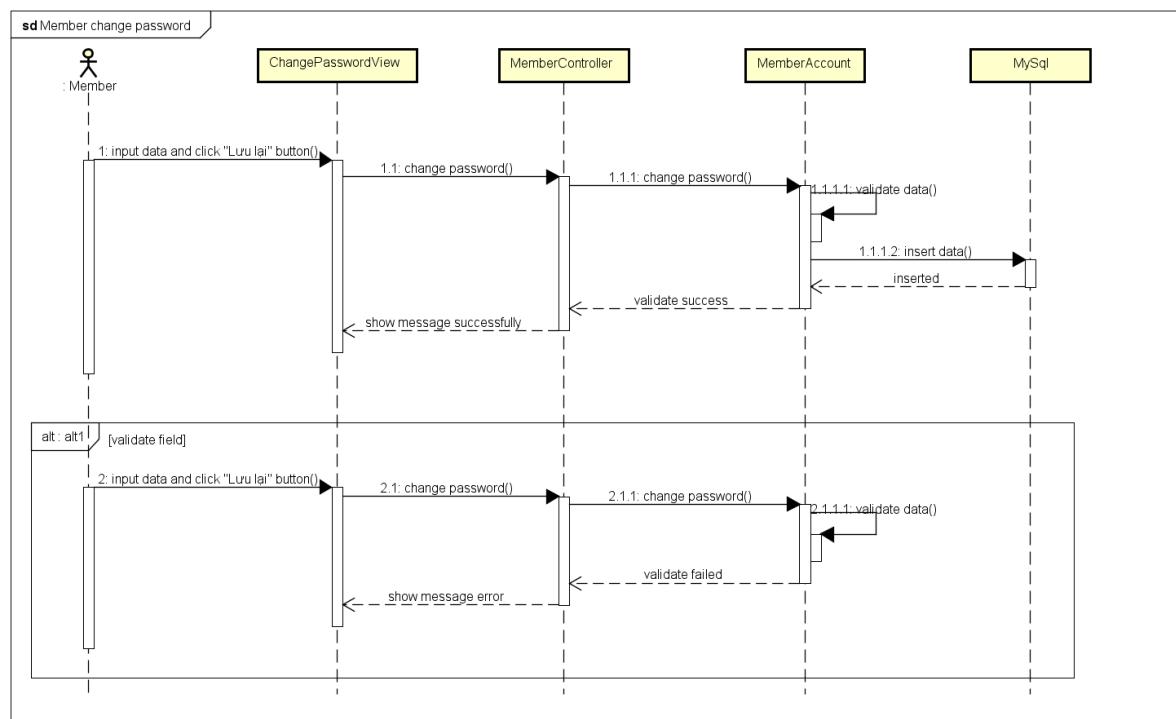


4.7.2.1.4.4 Class Specifications

Class	MemberController
Description	Control member change password action

Base Class	Controller			
Source File	MemberController.php			
Namespace	app/Http/Controllers/Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	changePassword	memberID	View personal page with successful message	Allow member change their account's password

4.7.2.1.4.5 Sequence Diagram

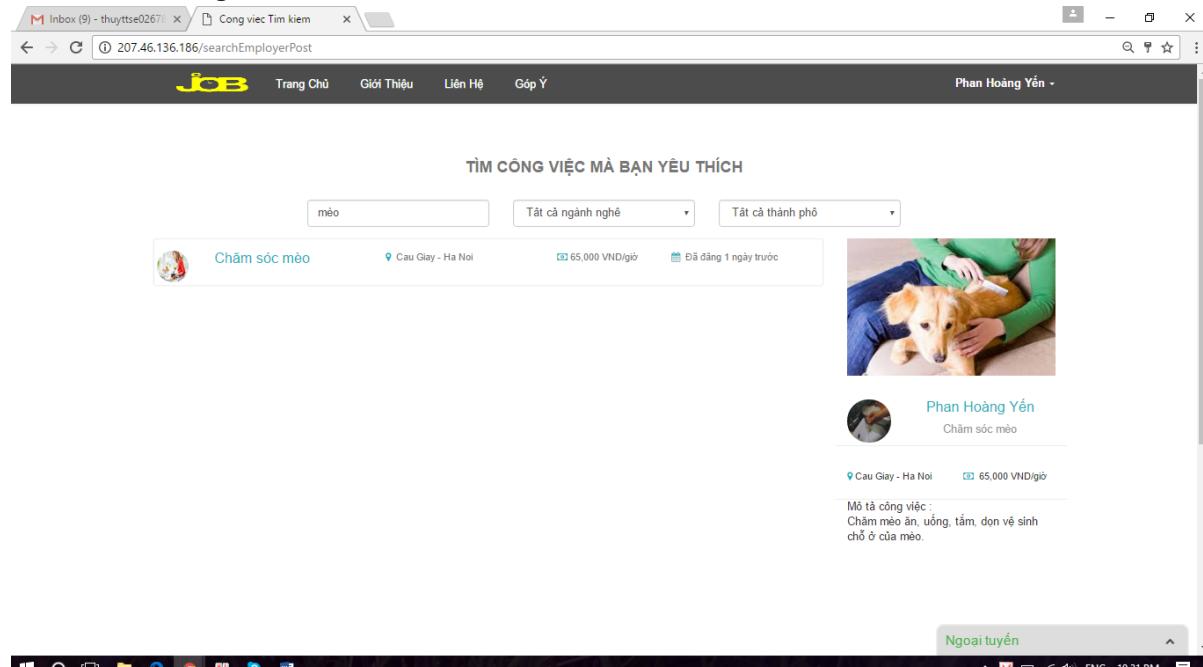


4.7.2.1.5 Search Employer's Posts

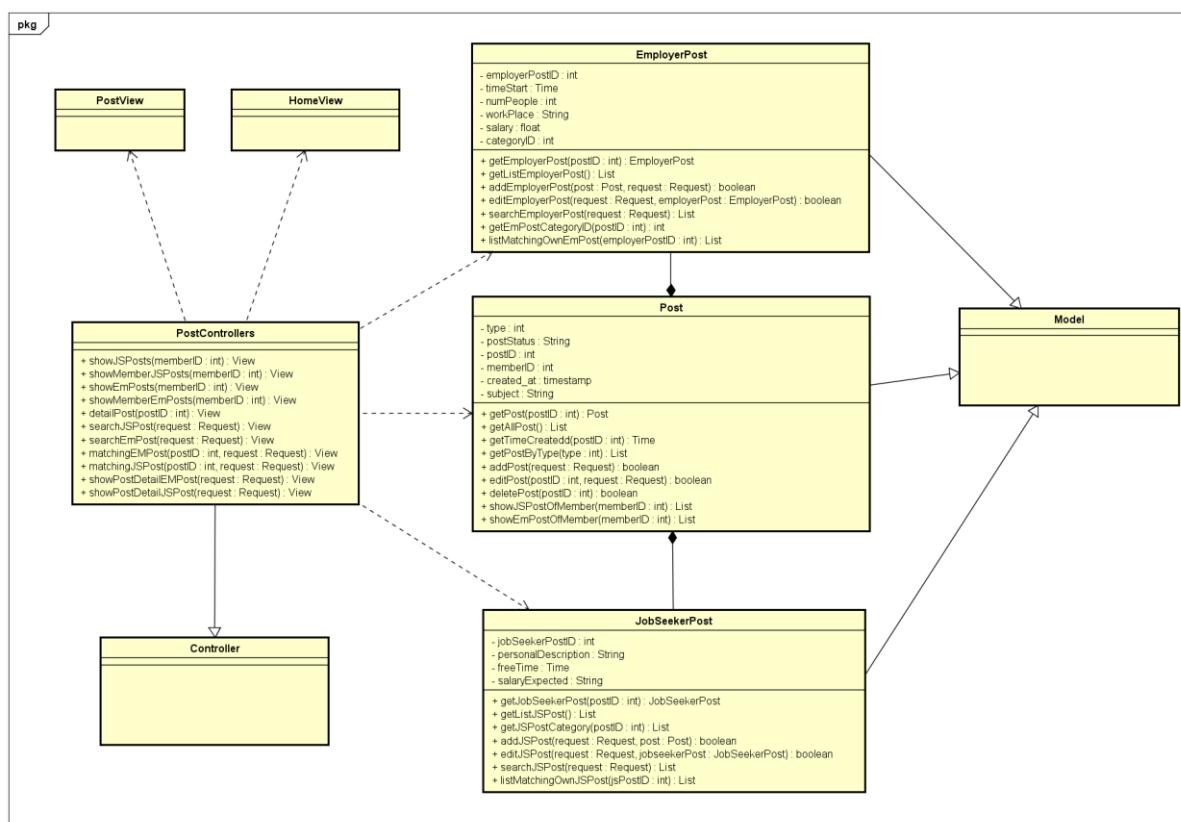
4.7.2.1.5.1 Description

This function allows Member to searching for specific Employer's Posts.

4.7.2.1.5.2 Design



4.7.2.1.5.3 Class Diagram

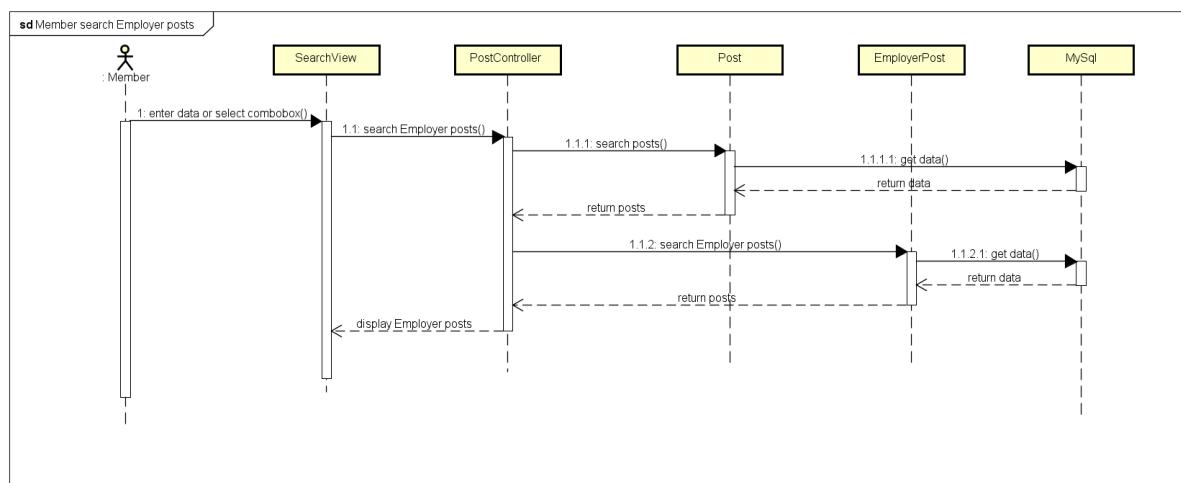


4.7.2.1.5.4 Class Specifications

Class	PostController
Description	Control member search for Employer's posts action

Base Class	Controller			
Source File	PostController.php			
Namespace	app/Http/Controllers/post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchEmPost	request	View list employer's post	Member type keyword or click select search Post of Employer by category, location

4.7.2.1.5.5 Sequence Diagram



4.7.2.1.6 Search Job Seeker's Posts

4.7.2.1.6.1 Description

This function allows Member to searching for specific Job Seeker's Posts

4.7.2.1.6.2 Design

The screenshot shows a web application interface for job seekers. At the top, there are three search filters: 'chuyên' (specialty), 'Tất cả ngành nghề' (All industries), and 'Tất cả thành phố' (All cities). Below these filters, there are two job listings:

- Chuyển đồ**: Located at Hai Ba Trung - Ha Noi, salary 50.000VND/giờ, posted 2 days ago. Description: Cận thận.
- Dương Đặng Duy**: Located at Hai Ba Trung - Ha Noi, salary 50.000VND/giờ. Description: Chuyển đồ.

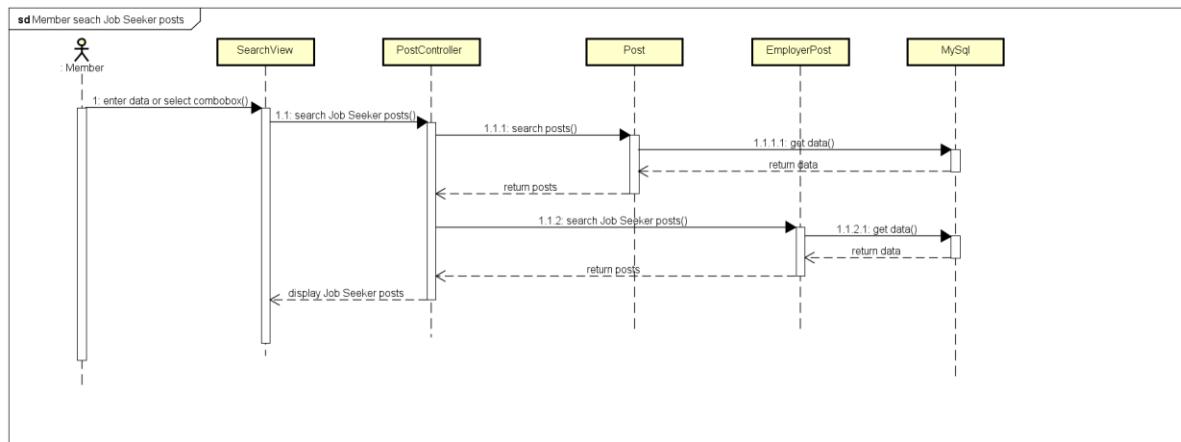
4.7.2.1.6.3 Class Diagram

Refer to class diagram of view Search Employer's Posts in 4.7.2.1.5 Search Employer's Posts

4.7.2.1.6.4 Class Specifications

Class	PostController			
Description	Control member search for Job Seeker's posts action			
Base Class	PostController			
Source File	Controller.php			
Namespace	app/Http/Controllers/post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchJSPost	request	View list of jobseeker's post	Member type keyword or click select search Post of JobSeeker by category, location

4.7.2.1.6.5 Sequence Diagram

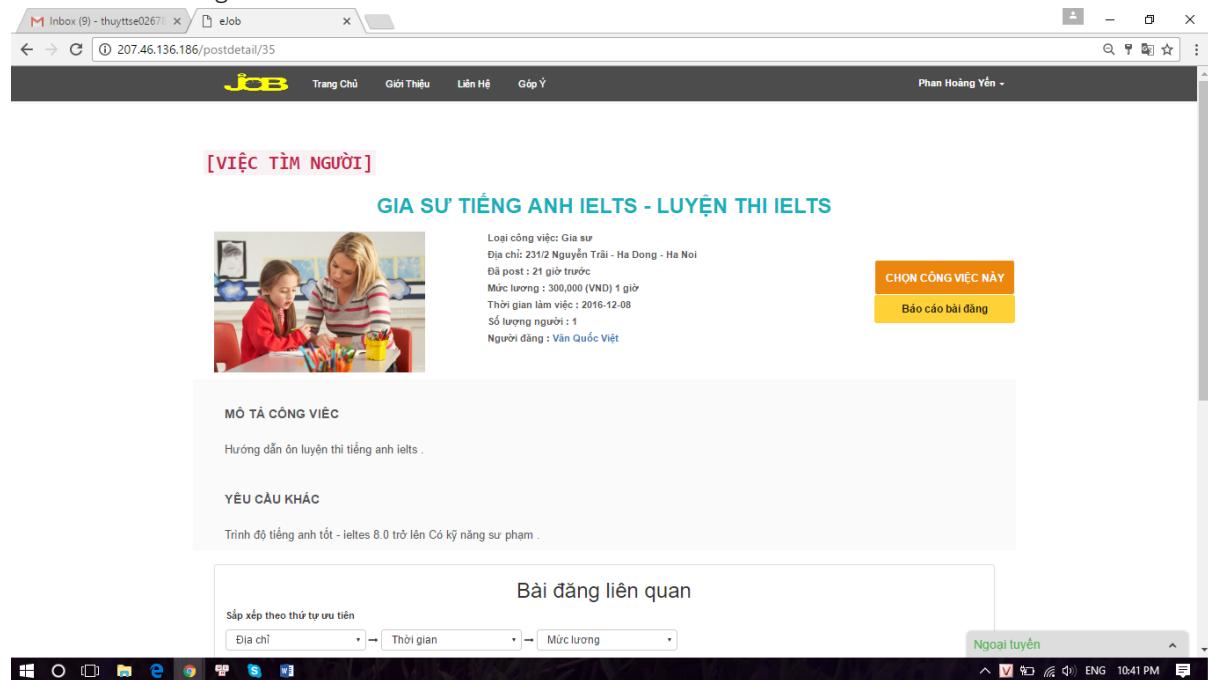


4.7.2.1.7 View Employer's Posts

4.7.2.1.7.1 Description

This function allows Member to view Employer's Posts

4.7.2.1.7.2 Design



4.7.2.1.7.3 Class Diagram

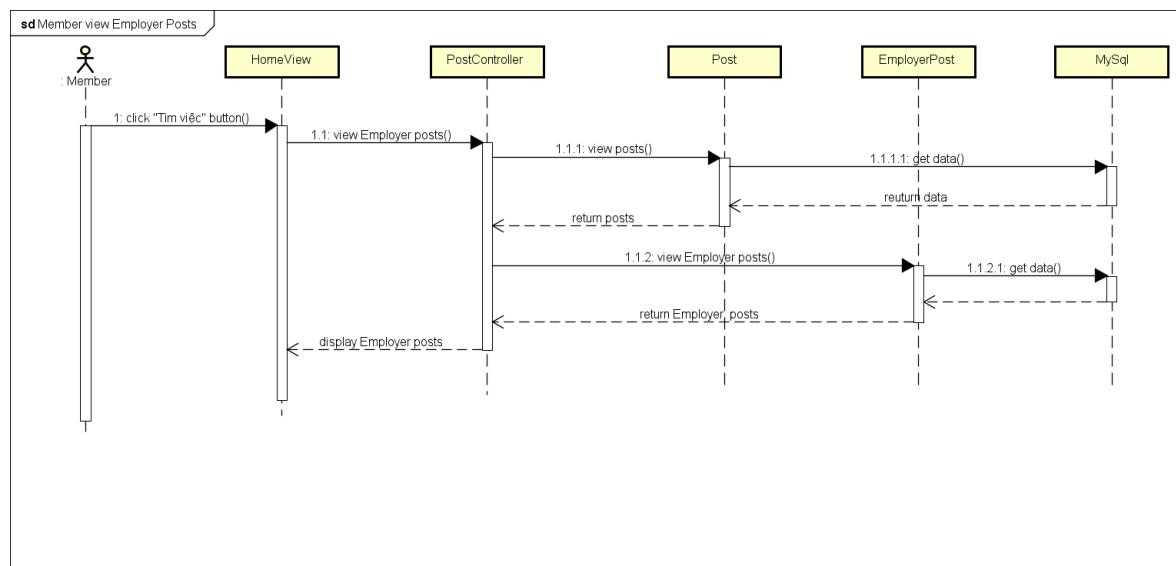
Refer to class diagram of view Search Employer's Posts in 4.7.2.1.5 Search Employer's Posts

4.7.2.1.7.4 Class Specifications

Class	PostController
Description	Control member view Employer's Post action

Base Class	Controller			
Source File	PostController.php			
Namespace	app/Http/Controllers/post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchEmPost	request	View list employer's post	Display all Post of Employer (after implemented search function)

4.7.2.1.7.5 Sequence Diagram



4.7.2.1.8 View Job Seeker's Posts

4.7.2.1.8.1 Description

This function allows Member to view Job Seeker's Posts

4.7.2.1.8.2 Design

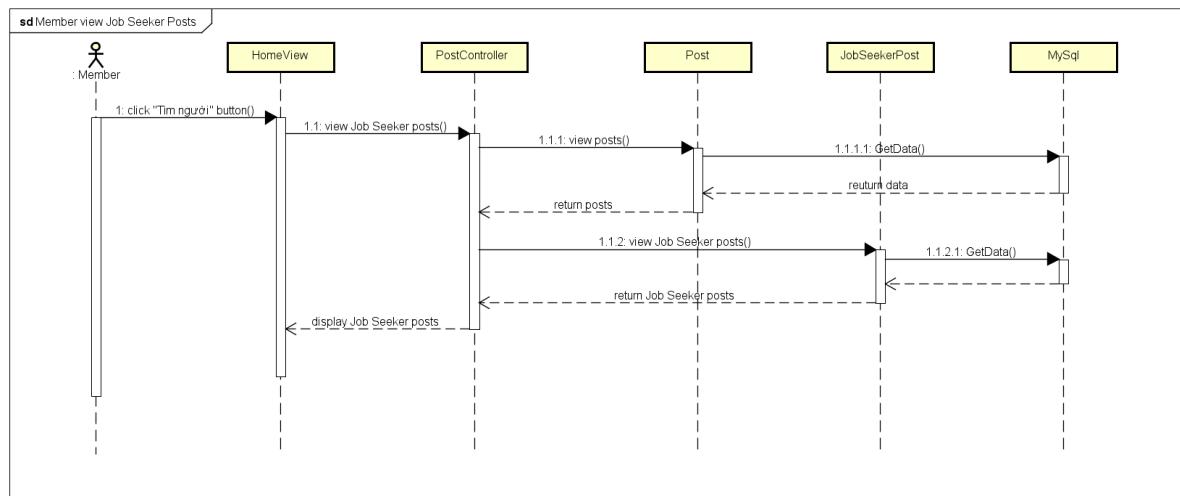
4.7.2.1.8.3 Class Diagram

Refer to class diagram of view Search Employer's Posts in 4.7.2.1.5 Search Employer's Posts

4.7.2.1.8.4 Class Specifications

Class	PostController			
Description	Control member view Job Seeker's Post action			
Base Class	Controller			
Source File	PostController.php			
Namespace	app/Http/Controllers/post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchEmPost	request	View list jobseeker's post	Display all Post of jobseeker (after implemented search function)

4.7.2.1.8.5 Sequence Diagram



4.7.2.1.9 Apply suitable Job

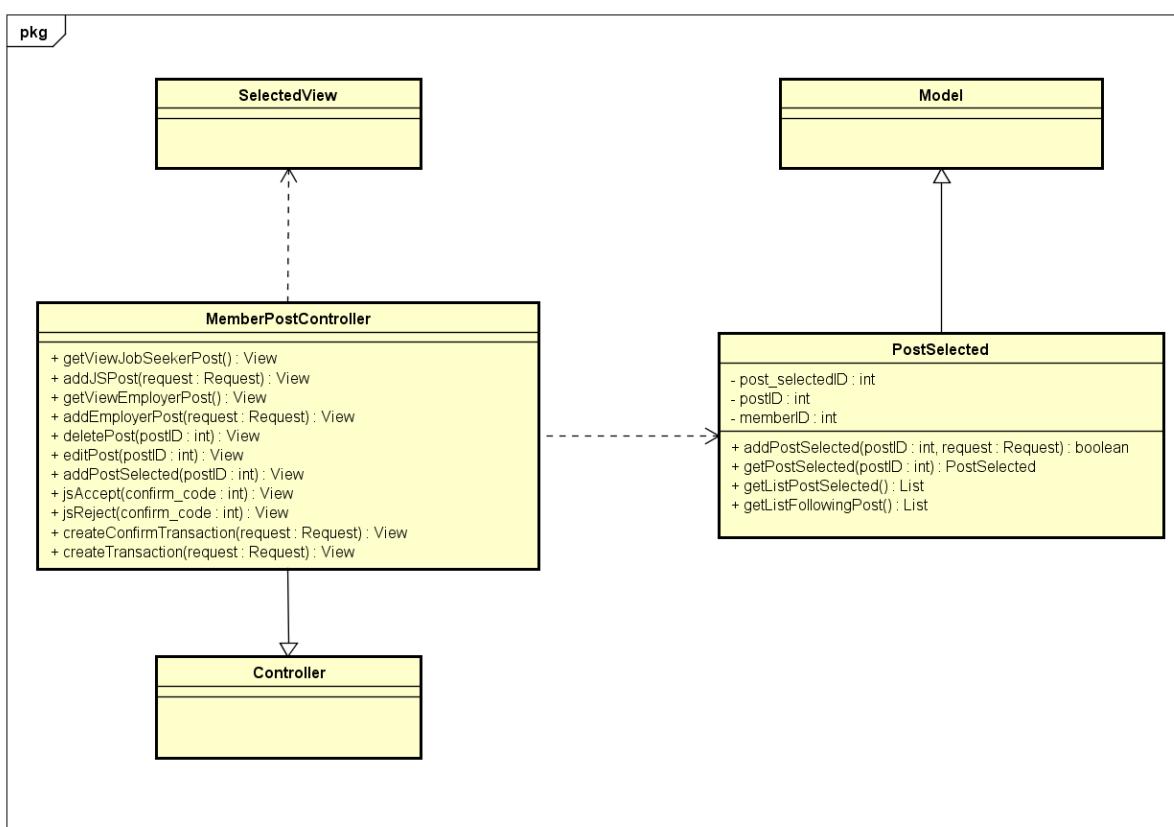
4.7.2.1.9.1 Description

This function allows Job Seeker to apply a job.

4.7.2.1.9.2 Design

Refer Design of View Employer's Post 4.7.2.1.7.2 View Employer's Post

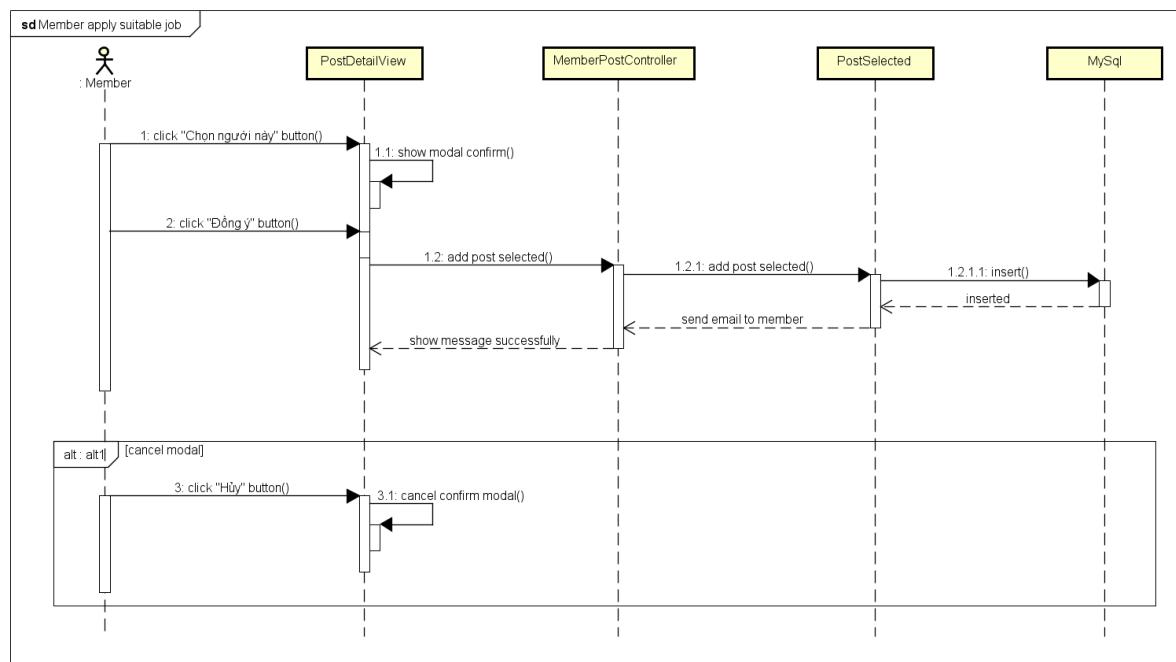
4.7.2.1.9.3 Class Diagram



4.7.2.1.9.4 Class Specifications

Class	MemberPostController			
Description	Control Member apply a job action			
Base Class	Controller			
Source File	MemberPostController.php			
Namespace	app/Http/Controllers/ Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	addPostSelected	postID	View postdetail display success	Member select an Employer and apply job.
	jsAccept	confirm_code	View postdetail display success	jobSeeker accept a job after employer select yourself

4.7.2.1.9.5 Sequence Diagram



4.7.2.1.10 Get suitable Job Seeker

4.7.2.1.10.1 Description

This function allows Member to get a Job Seeker.

4.7.2.1.10.2 Design

Refer to Design of View Job Seeker's Post 4.7.2.1.8.2 View Job Seeker's Post

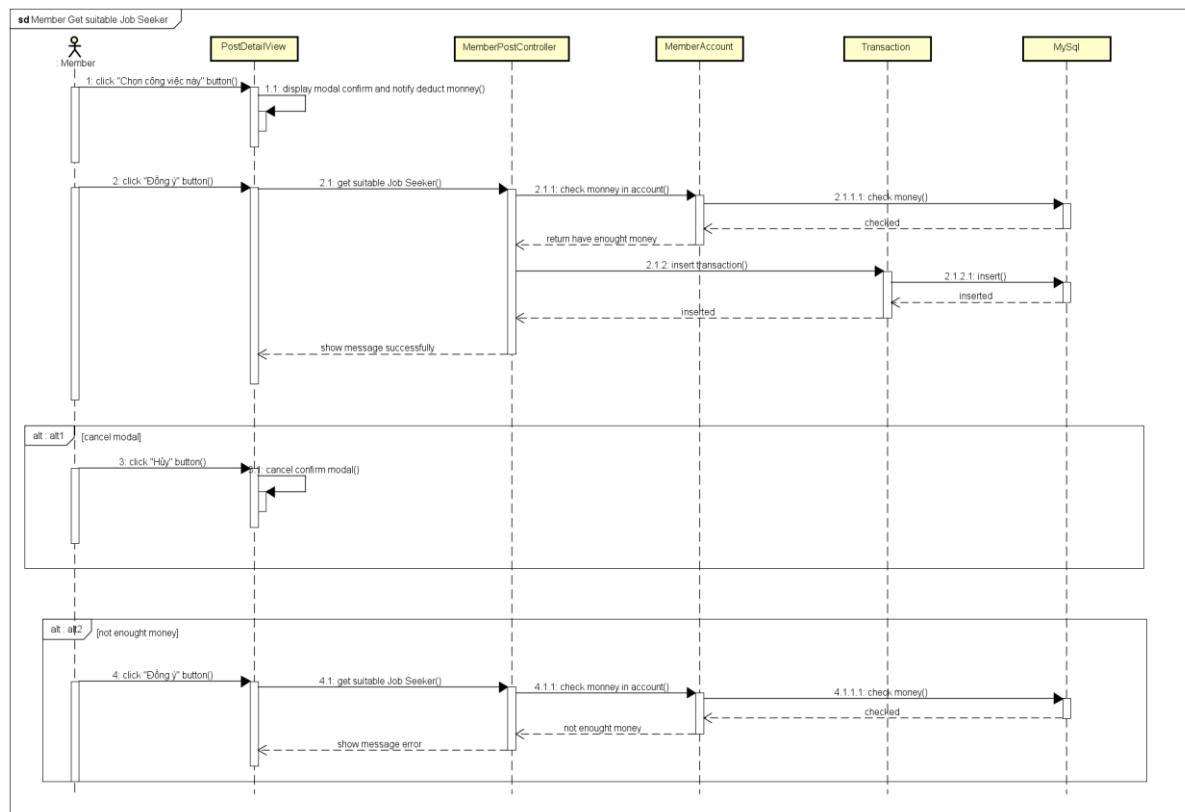
4.7.2.1.10.3 Class Diagram

Refer to class diagram of Apply suitable Job in 4.7.2.1.9 Apply suitable Job

4.7.2.1.10.4 Class Specifications

Class	MemberPostController			
Description	Control member selected a job seeker action			
Base Class	Controller			
Source File	MemberPostController.php			
Namespace	app/Http/Controllers/Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	createConfirmTransaction	request	View jobSeekerPostDetail	Member select a job seeker
	createTransaction	request	View employerPostDetail	Member accept a job seeker

4.7.2.1.10.5 Sequence Diagram



4.7.2.1.11 Report other Member's Post

4.7.2.1.11.1 Description

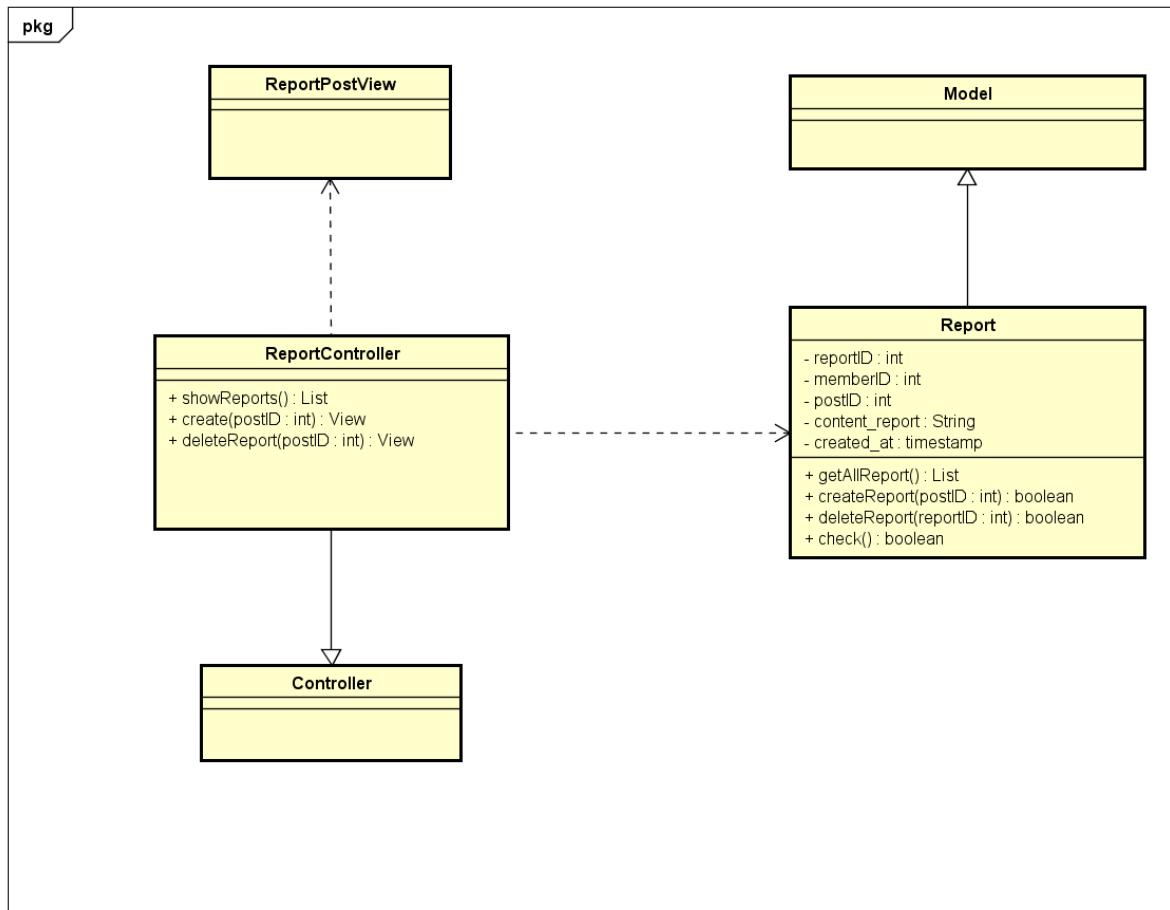
Report other member's post.

4.7.2.1.11.2 Design

Refer to Design of View Employer's Post 4.7.2.1.7.2 View Employer's Post

Refer to Design of View Job Seeker's Post 4.7.2.1.8.2 View Job Seeker's Post

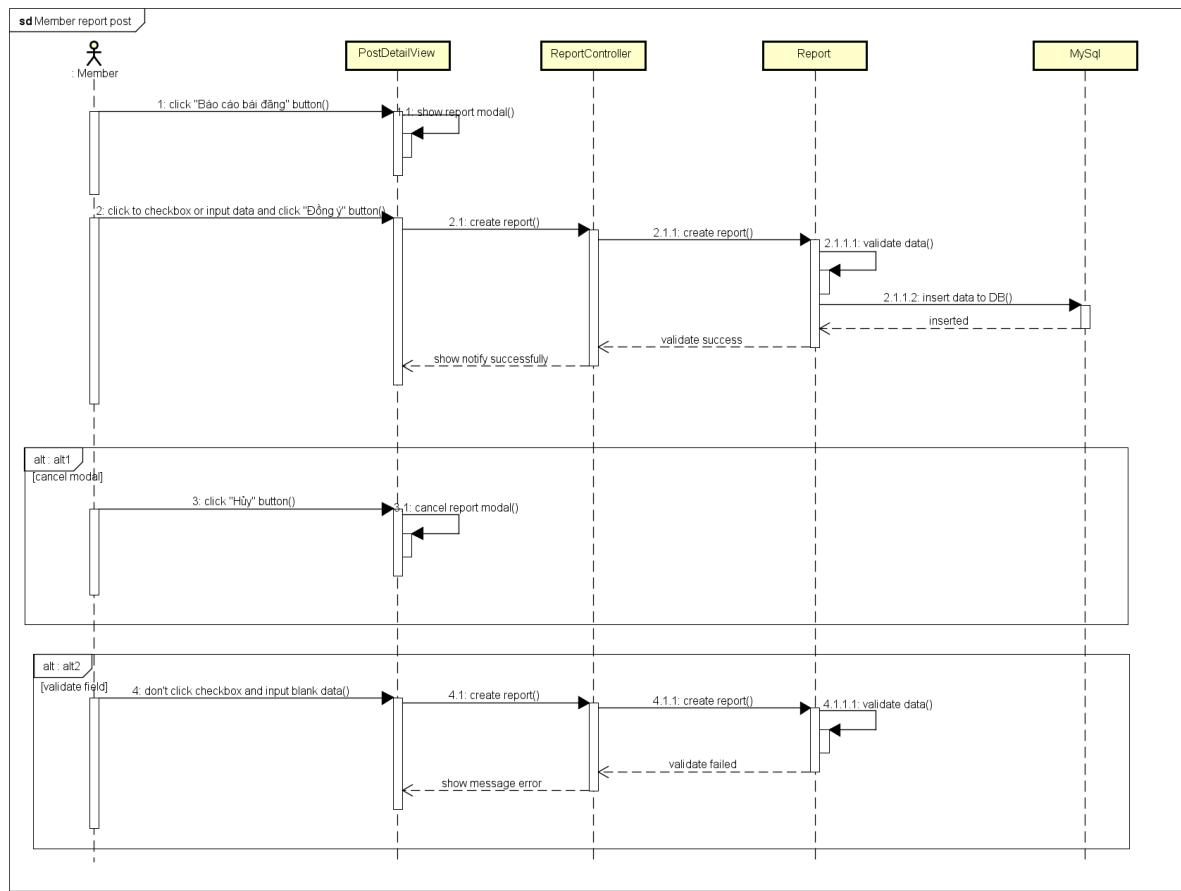
4.7.2.1.11.3 Class Diagram



4.7.2.1.11.4 Class Specifications

Class	ReportController			
Description	Control Member report a post action			
Base Class	ReportController			
Source File	MemberPostController.php			
Namespace	app/Http/Controllers			
Attributes	N/A			
Methods	Name	Input	Output	Description
	create	postID	View post detail and message report success	Report a post of a member

4.7.2.1.11.5 Sequence Diagram

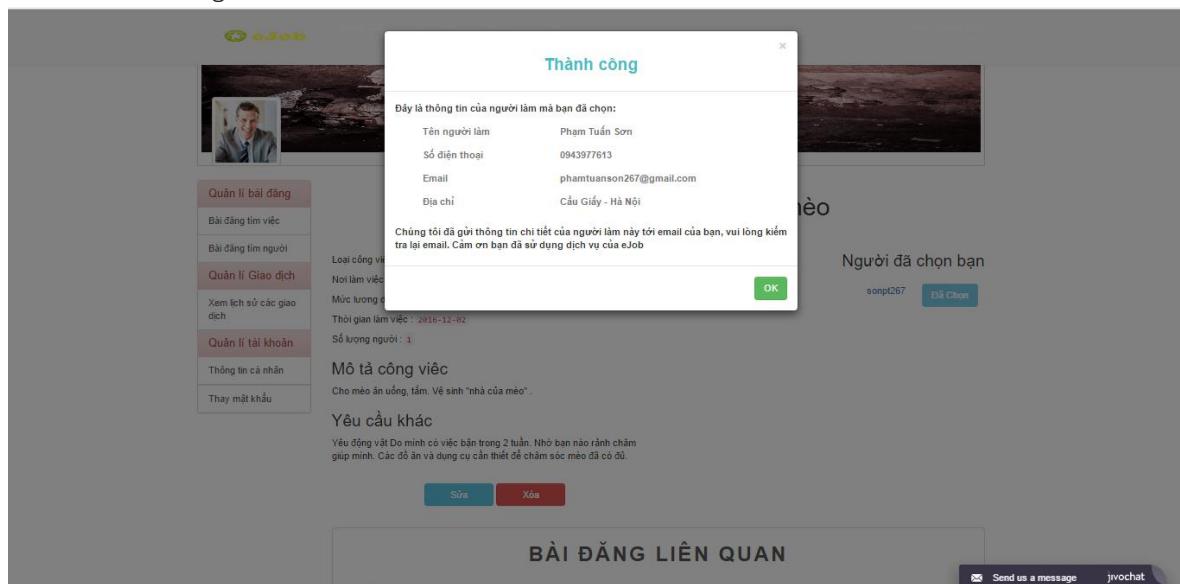


4.7.2.1.12 Payment to get hidden Profile of Job Seekers

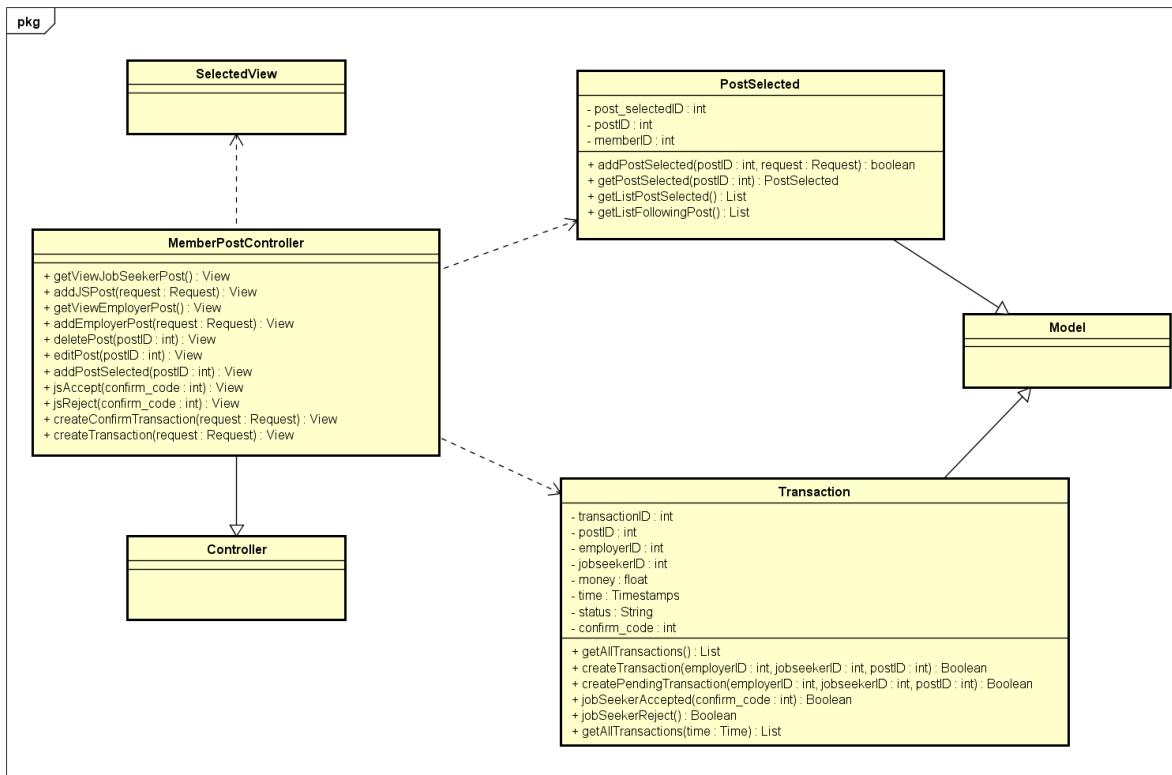
4.7.2.1.12.1 Description

This function allows Member pay to get hidden Profile information of Job Seekers.

4.7.2.1.12.2 Design



4.7.2.1.12.3 Class Diagram



4.7.2.1.12.4 Class Specifications

Class	MemberPostController			
Description	Control Member select Job Seeker's post and payment to get hidden profile action			
Base Class	Controller			
Source File	MemberPostController.php			
Namespace	app/Http/Controllers/ Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	createTransaction	request	JobSeeker's hidden information	Select Job Seeker and pay to get hidden profile
	jsAccept	request	JobSeeker's hidden information	After job seeker accept employer, deduct employer money and send job seeker's hidden profile to employer

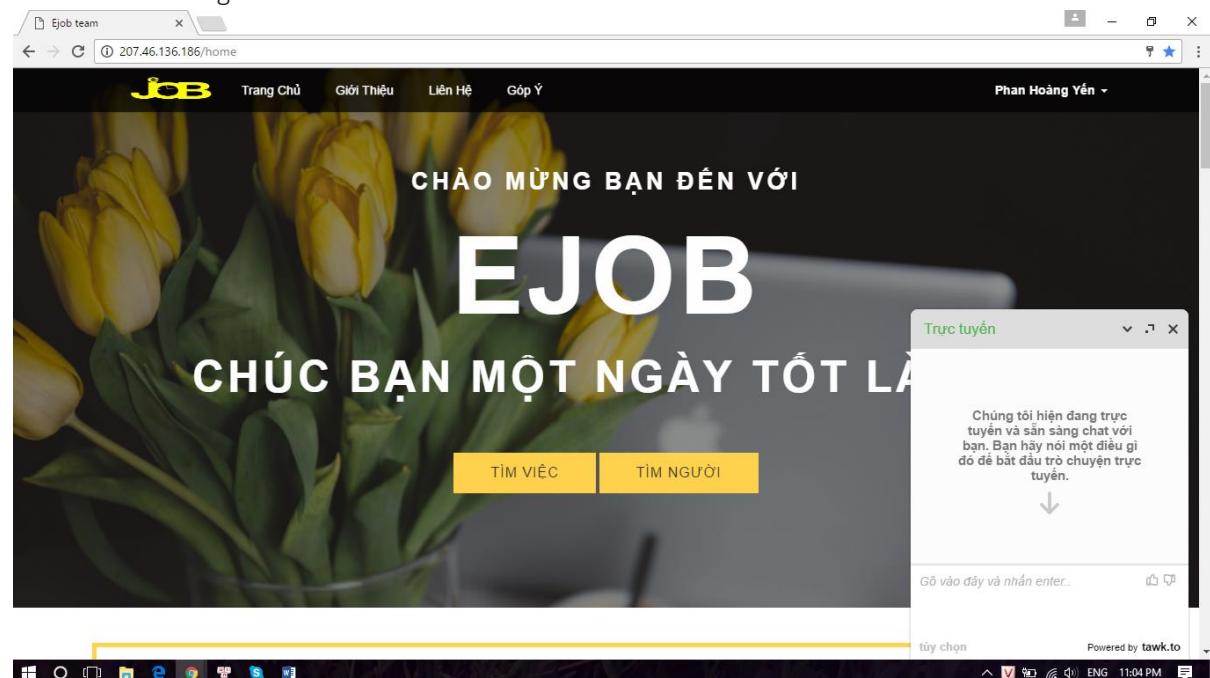
4.7.2.1.12.5 Sequence Diagram

4.7.2.1.13 Online chat with Admin

4.7.2.1.13.1 Description

Member chat online with admin. When Admin online (in session), Member create message and this message is sent to Admin and displayed on time by Tawk.to. Admin also can reply Member's message, and this message also sent to Member and displayed on time. When a message is sent during the time Admin or Member offline (out of session), message is stored and directly displayed when Admin or Member online. All message that be seen will not be stored.

4.7.2.1.13.2 Design



4.7.2.1.13.3 Class Diagram

N/A

4.7.2.1.13.4 Class Specifications

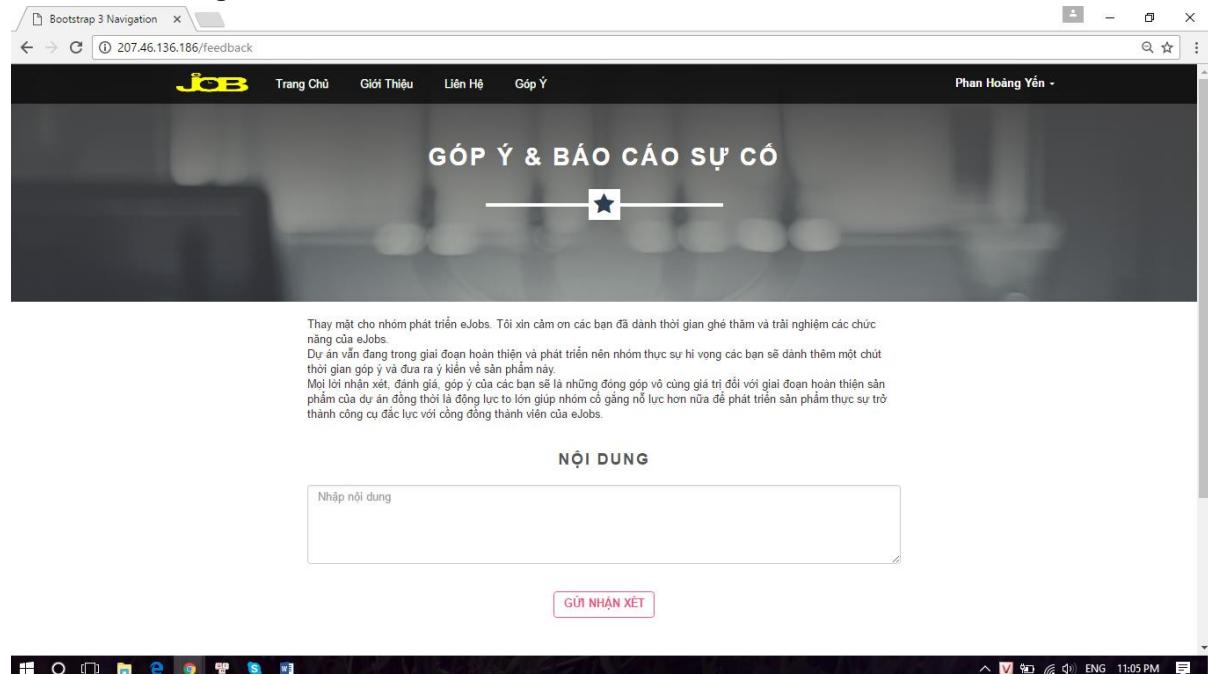
4.7.2.1.13.5 Sequence Diagram

4.7.2.1.14 Feedback issue

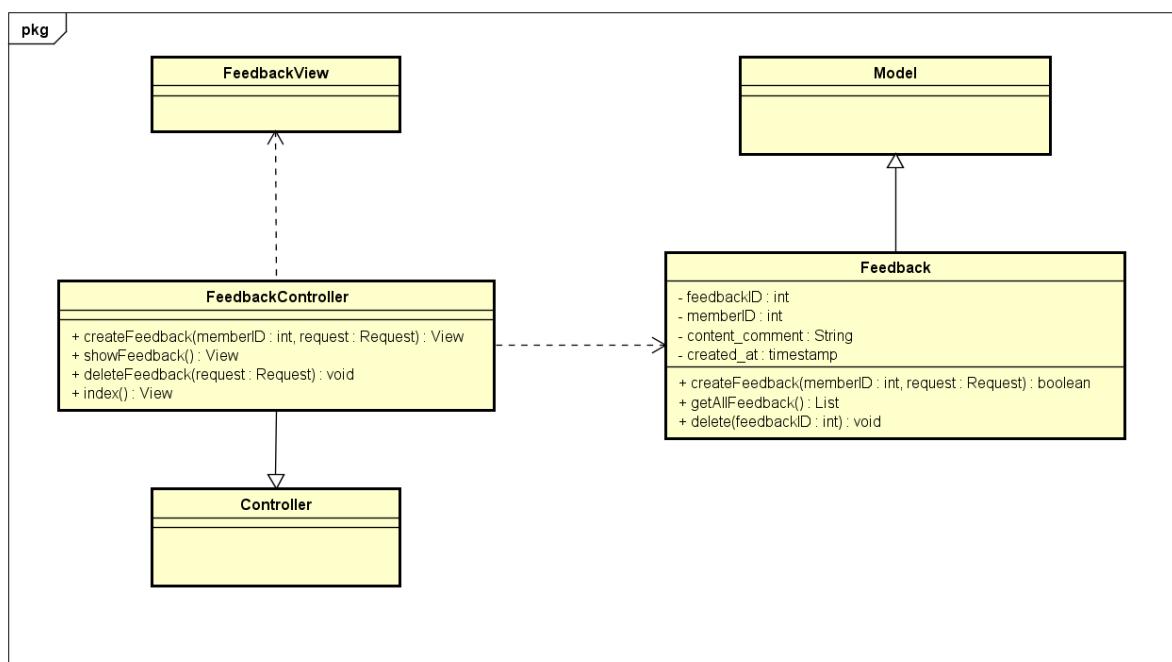
4.7.2.1.14.1 Description

Member can send feedback issue to admin

4.7.2.1.14.2 Design



4.7.2.1.14.3 Class Diagram

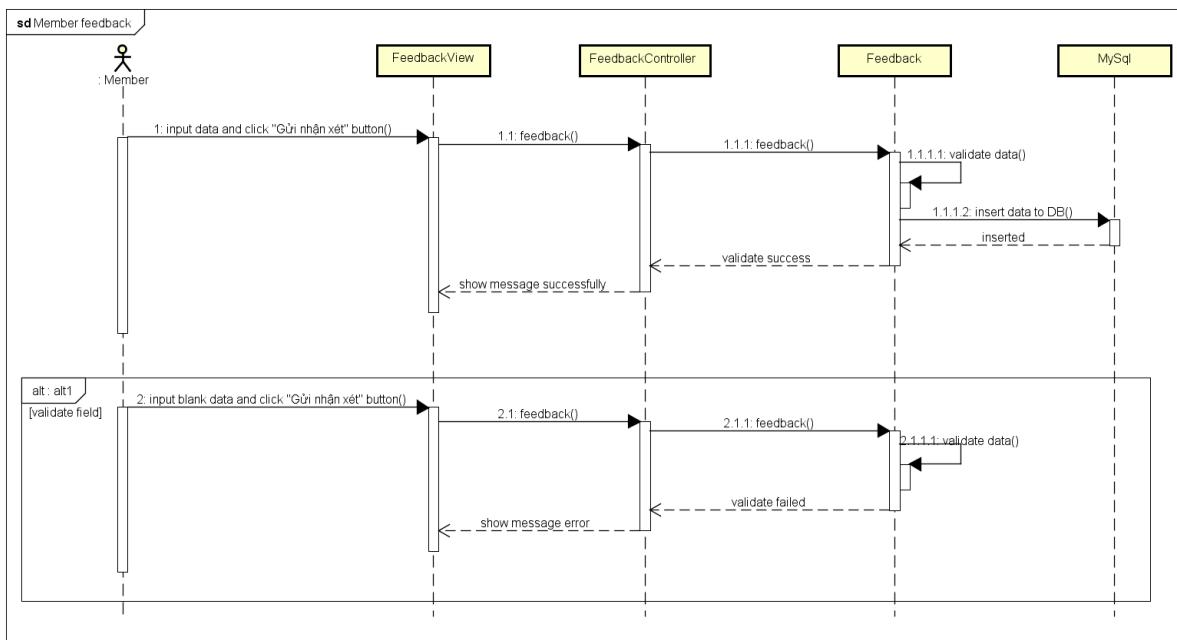


4.7.2.1.14.4 Class Specifications

Class	FeedbackController
Description	Control Member create and send feedback action
Base Class	Controller
Source File	FeedbackController.php
Namespace	app/controllers/
Attributes	N/A

Methods	Name	Input	Output	Description
	createFeedback	memberID, request	Feedback View and message send feedback success	Send created feedback to admin.

4.7.2.1.14.5 Sequence Diagram



4.7.2.2 Manage Personal Page

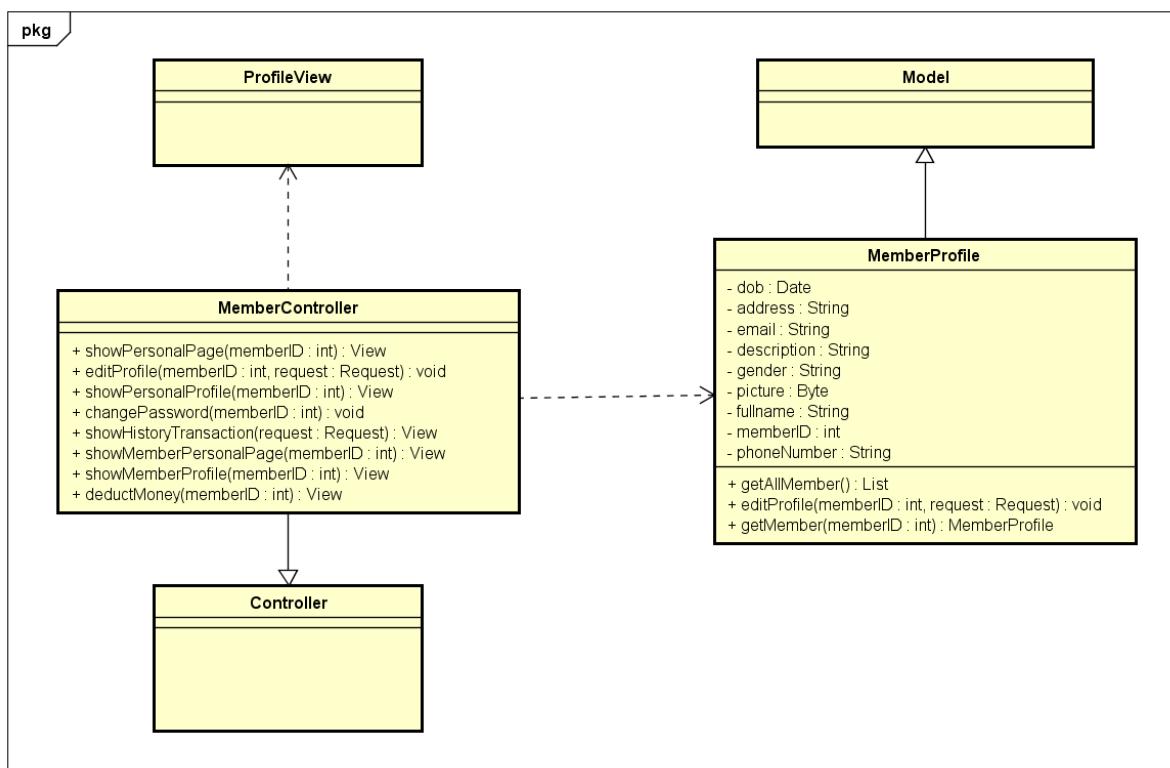
4.7.2.2.1 View Profile

4.7.2.2.1.1 Description

Member can view own profile.

4.7.2.2.1.2 Design

4.7.2.2.1.3 Class Diagram

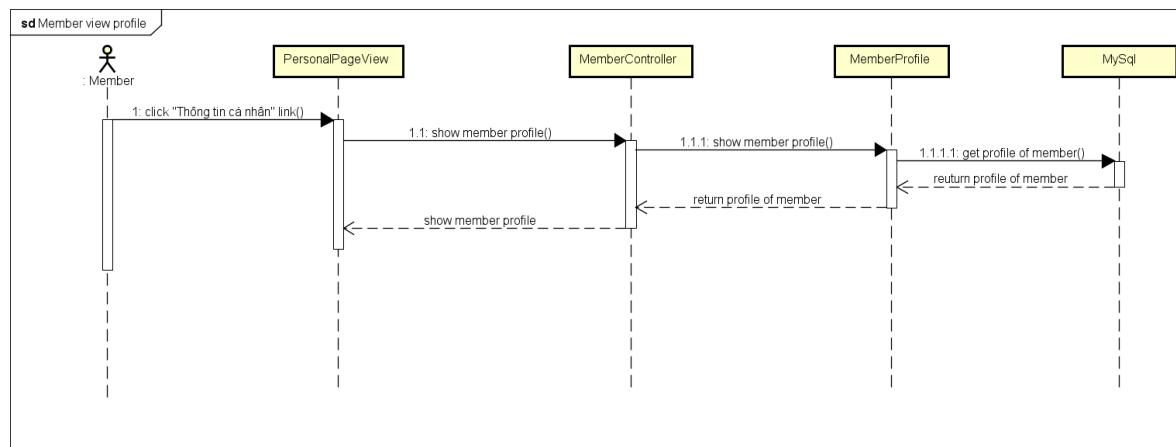


4.7.2.2.1.4 Class Specifications

Class	MemberController
Description	Control member view personal profile action
Base Class	Controller
Source File	MemberController.php

Namespace	app/Http/Controllers/Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	showPersonalProfile	memberID	View that display personal profile	Display member's personal profile

4.7.2.2.1.5 Sequence Diagram



4.7.2.2 Edit Profile

4.7.2.2.1 Description

Member can edit own profile.

4.7.2.2.2 Design

Refer to Design of View Profile 4.7.2.2.1.2 View Profile

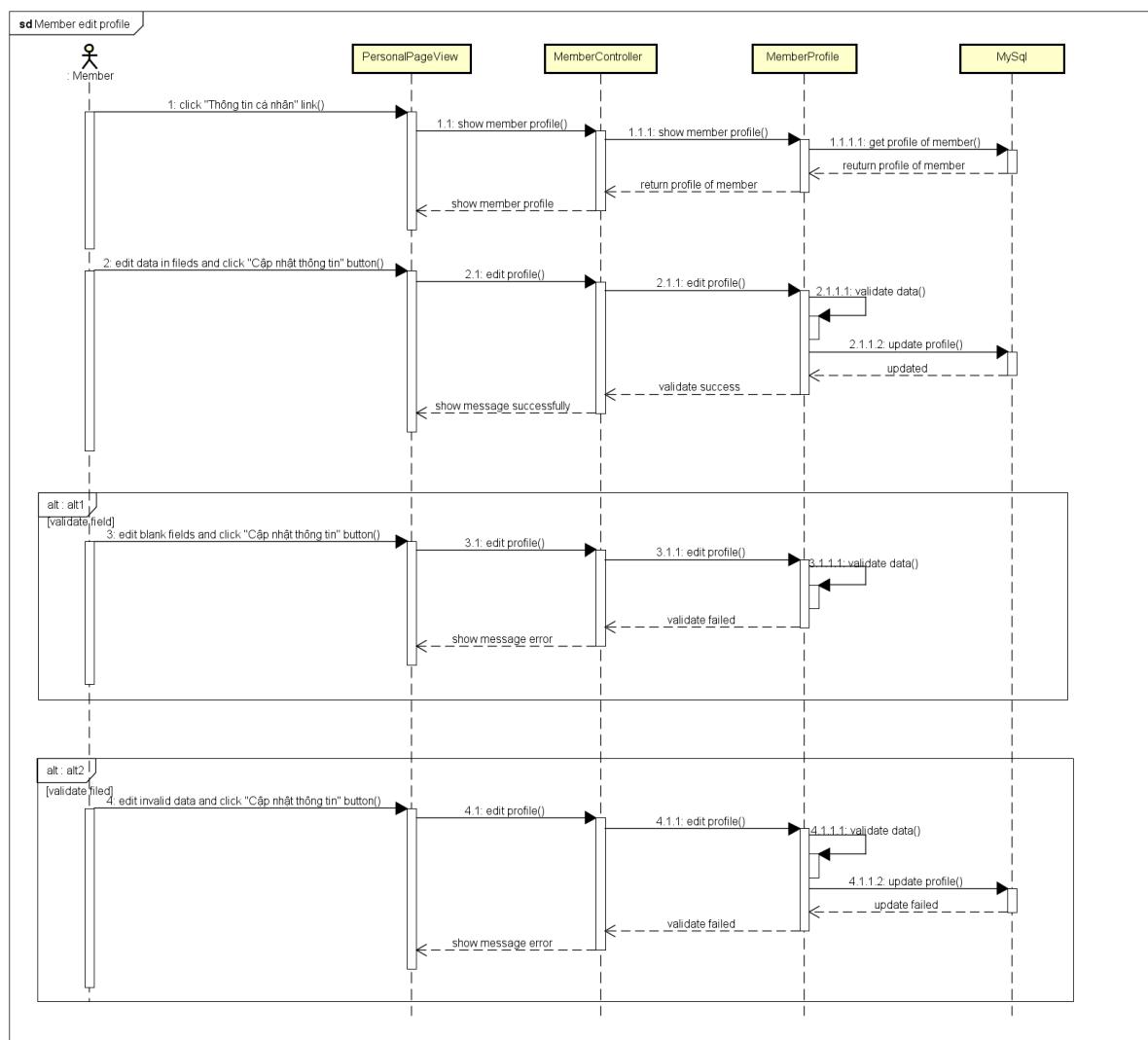
4.7.2.2.2.3 Class Diagram

Refer to class diagram of View Profile in 4.7.2.2.1 View Profile

4.7.2.2.2.4 Class Specifications

Class	MemberController			
Description	Control member edit personal profile action			
Base Class	Controller			
Source File	MemberController.php			
Namespace	app/Http/Controllers/Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	editProfile	memberID	View pesonal profile	Update member's profile information

4.7.2.2.2.5 Sequence Diagram

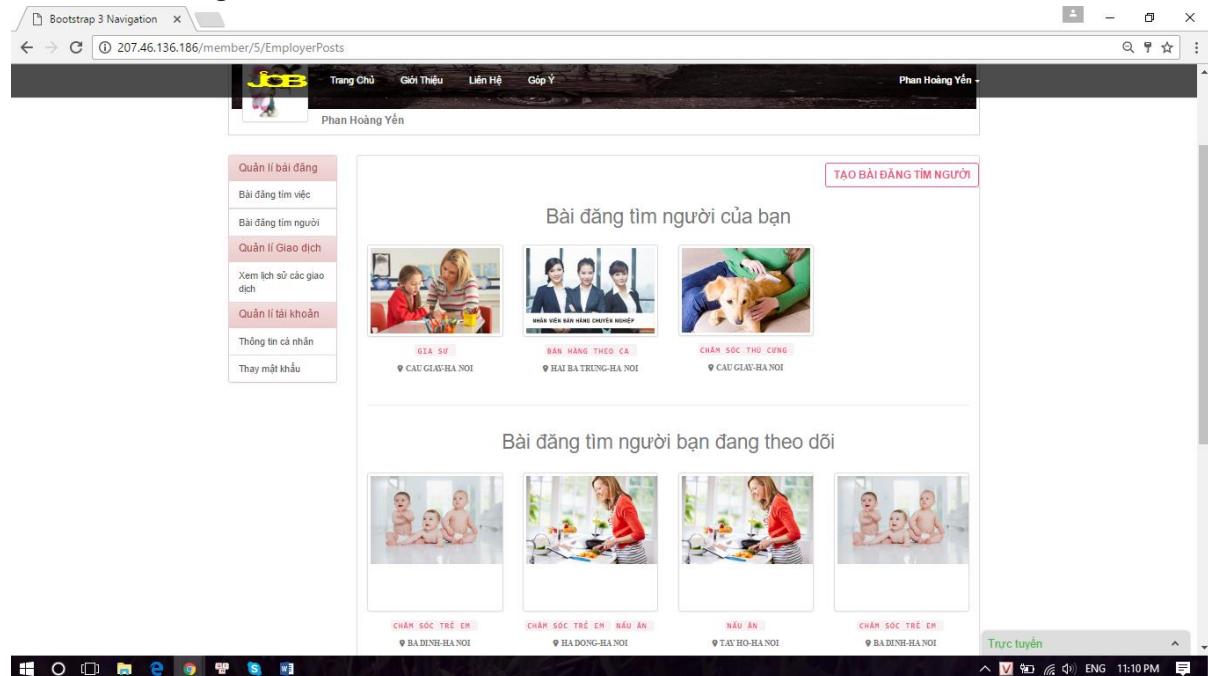


4.7.2.2.3 View personal Page

4.7.2.2.3.1 Description

Member can view personal page. From this page, member can go to other pages that used to view all personal Job Seeker posts, Employer Post and Profile information.

4.7.2.2.3.2 Design



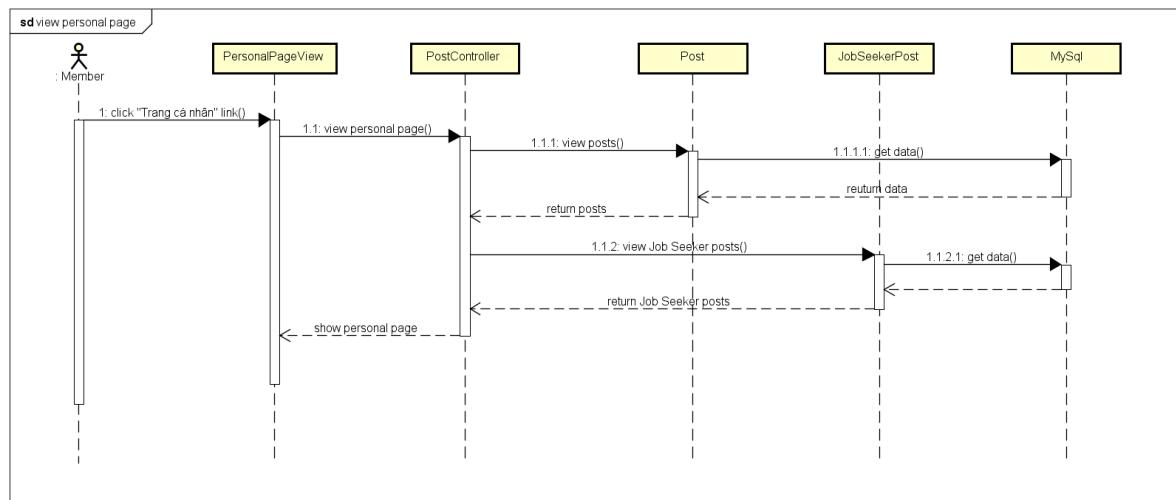
4.7.2.2.3.3 Class Diagram

Refer to class diagram of View Profile in 4.7.2.2.1 View Profile

4.7.2.2.3.4 Class Specifications

Class	MemberController			
Description	Control member personal page action			
Base Class	Controller			
Source File	MemberController.php			
Namespace	app/Http/Controllers/Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	showPersonalPage	memberID	View that show member personal page	Show member personal page.

4.7.2.2.3.5 Sequence Diagram

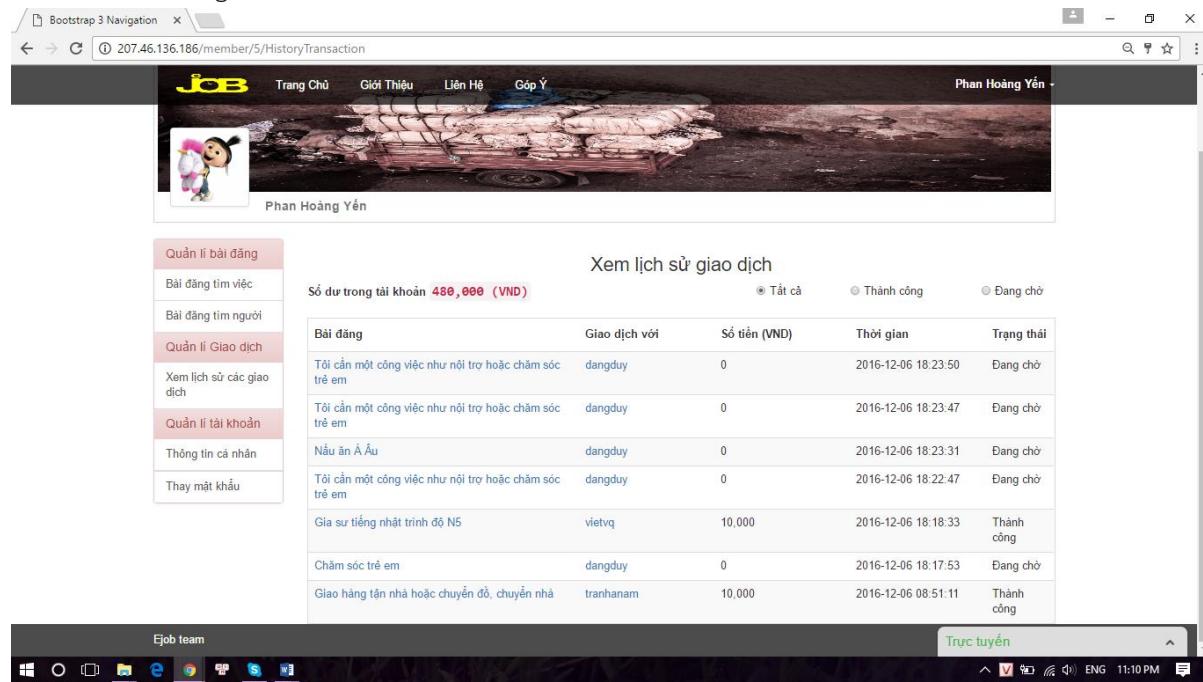


4.7.2.2.4 View History Transaction

4.7.2.2.4.1 Description

Member can view all transaction has completed . They can know how much money paid for the system.

4.7.2.2.4.2 Design



4.7.2.2.4.3 Class Diagram

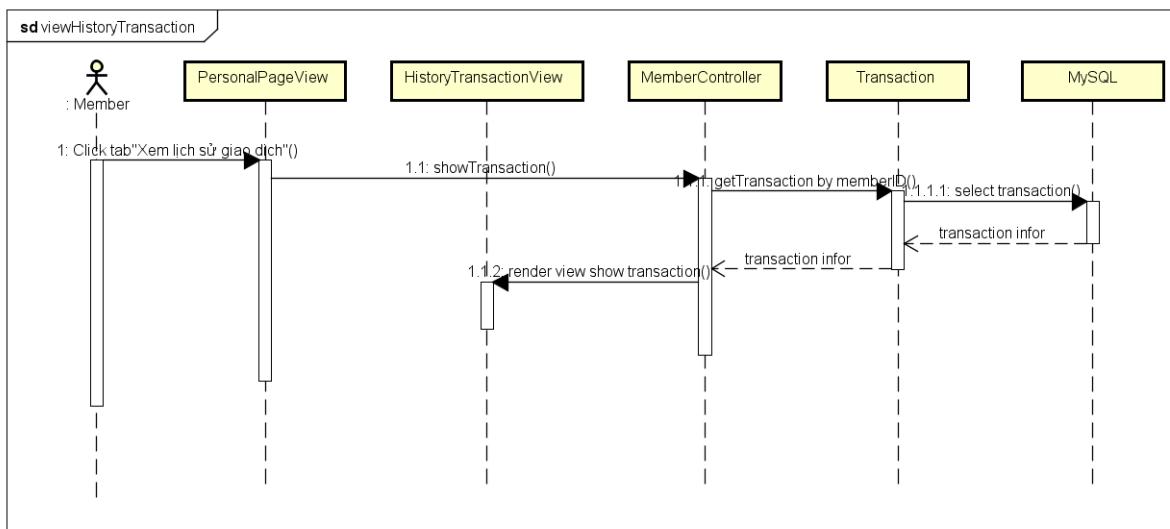
Refer to class diagram of View Profile in 4.7.2.2.1 View Profile

4.7.2.2.4.4 Class Specifications

Class	MemberController
Description	Control member personal page action

Base Class	Controller			
Source File	MemberController.php			
Namespace	app/Http/Controllers/Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	viewHistoryTransaction	request	View show transaction has been completed	Show transaction that member perform

4.7.2.2.4.5 Sequence Diagram



4.7.2.3 View other Member's Personal Page

4.7.2.3.1 View other Member's Posts

4.7.2.3.1.1 Description

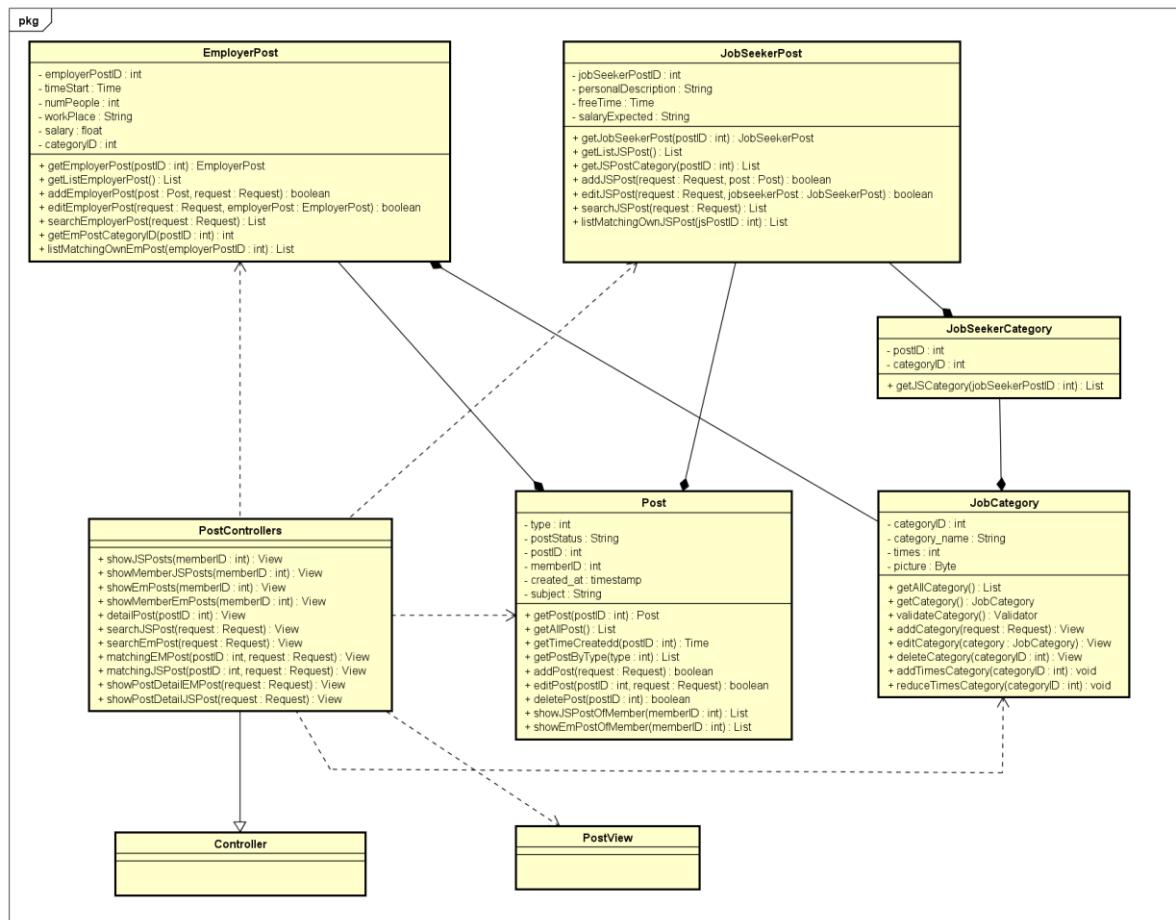
Member can view posts of other Members in system.

4.7.2.3.1.2 Design

Refer to Design of View Employer's Post 4.7.2.1.7.2 View Employer's Post

Refer to Design of View Job Seeker's Post 4.7.2.1.8.2 View Job Seeker's Post

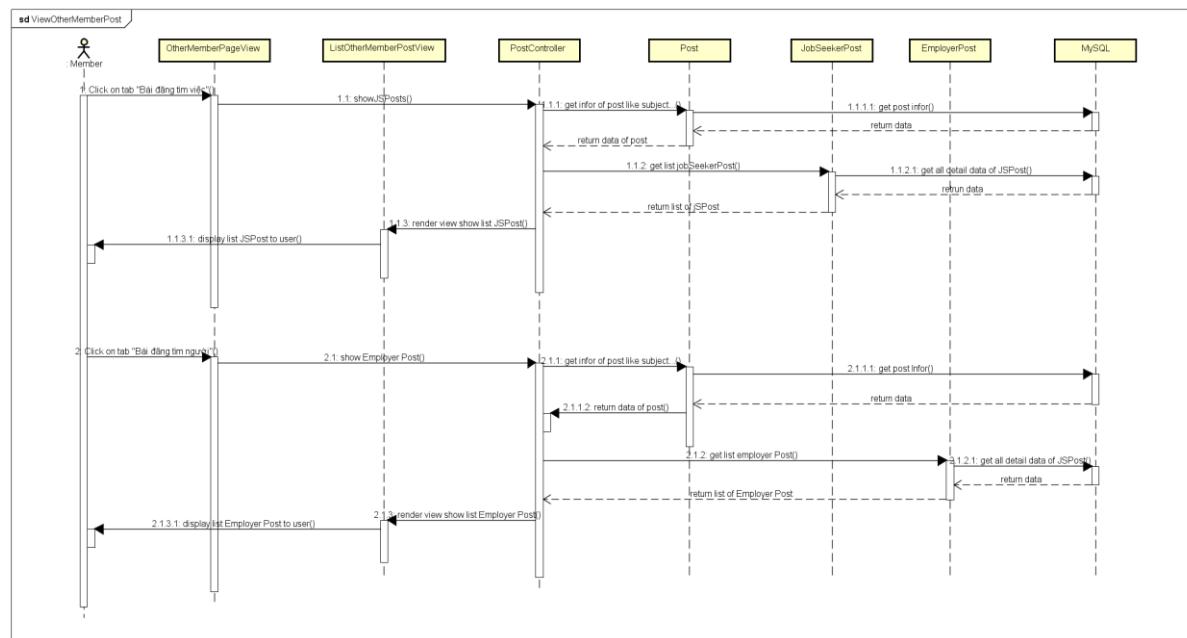
4.7.2.3.1.3 Class Diagram



4.7.2.3.1.4 Class Specifications

Class	PostController			
Description	Control and do business relate to Post			
Base Class	Controller			
Source File	PostController.php			
Namespace	App\Http\Controllers\Post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	showJSPosts	memberID	View display list jobSeeker post	Get all jobSeeker post of this member and render view
	showEmPosts	memberID	View display list employer post	Get all employer of this member post and render view

4.7.2.3.1.5 Sequence Diagram

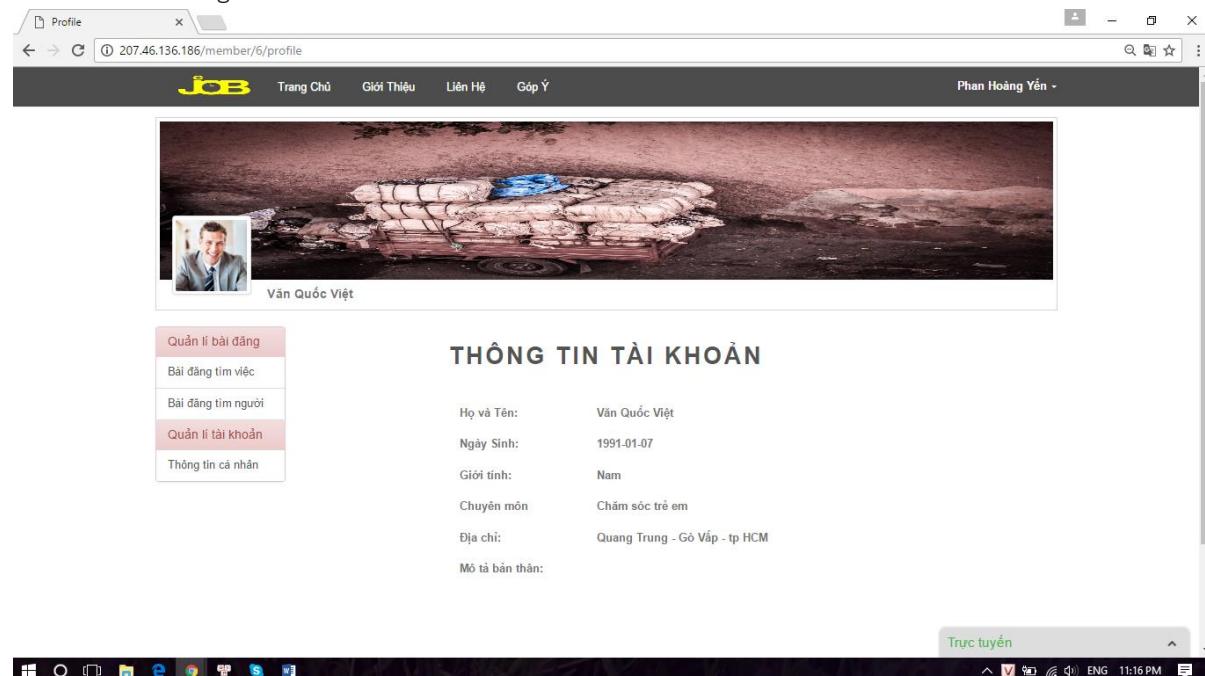


4.7.2.3.2 View other Member's basic Profile

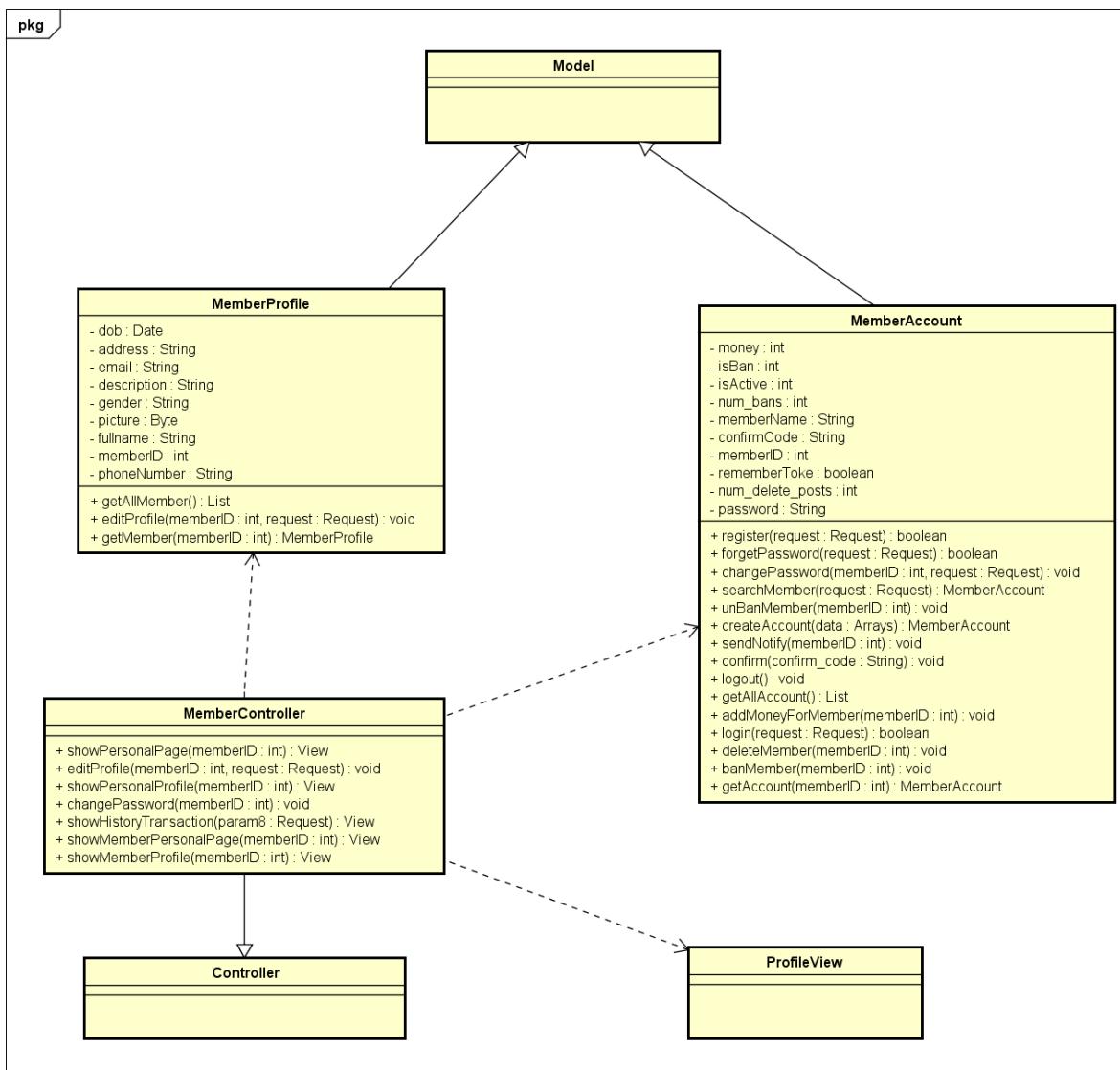
4.7.2.3.2.1 Description

This function allow member to view other Member's basic profile.

4.7.2.3.2.2 Design



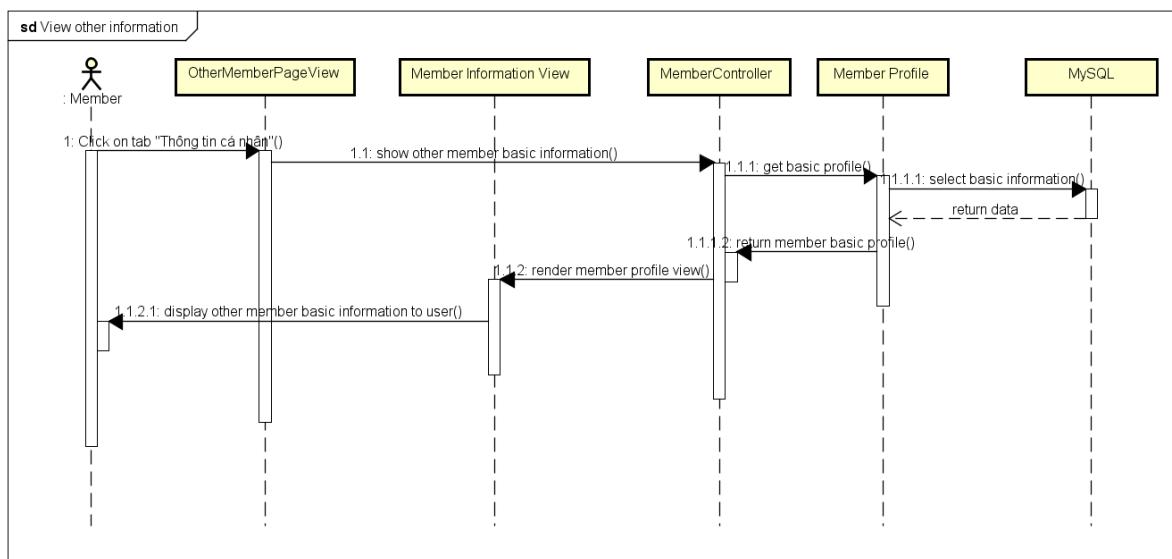
4.7.2.3.2.3 Class Diagram



4.7.2.3.2.4 Class Specifications

Class	MemberController			
Description	Get some basic profile of member			
Base Class	Controller			
Source File	MemberController.php			
Namespace	App\Http\Controllers\Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	showMemberProfile	memberID	View show basic other's member profile information	Get some basic profile information basic and render view

4.7.2.3.2.5 Sequence Diagram



4.7.2.4 Manage Post

4.7.2.4.1 Create Post

4.7.2.4.1.1 Description

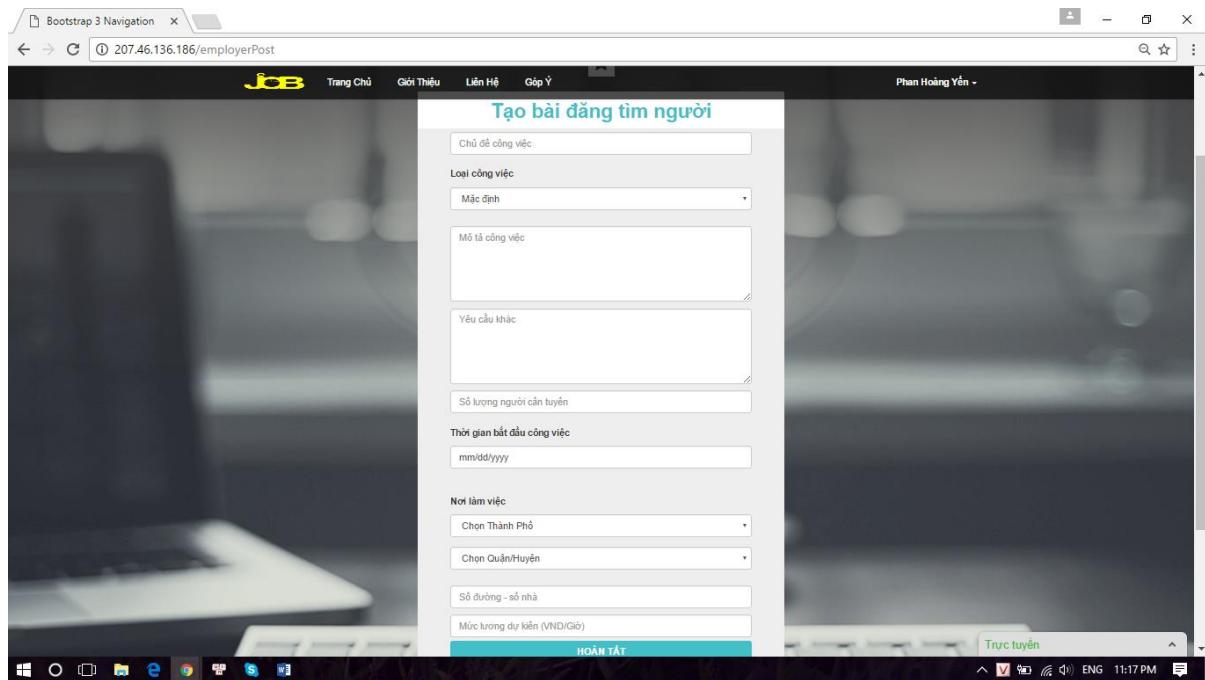
Member can create post.

4.7.2.4.1.2 Design

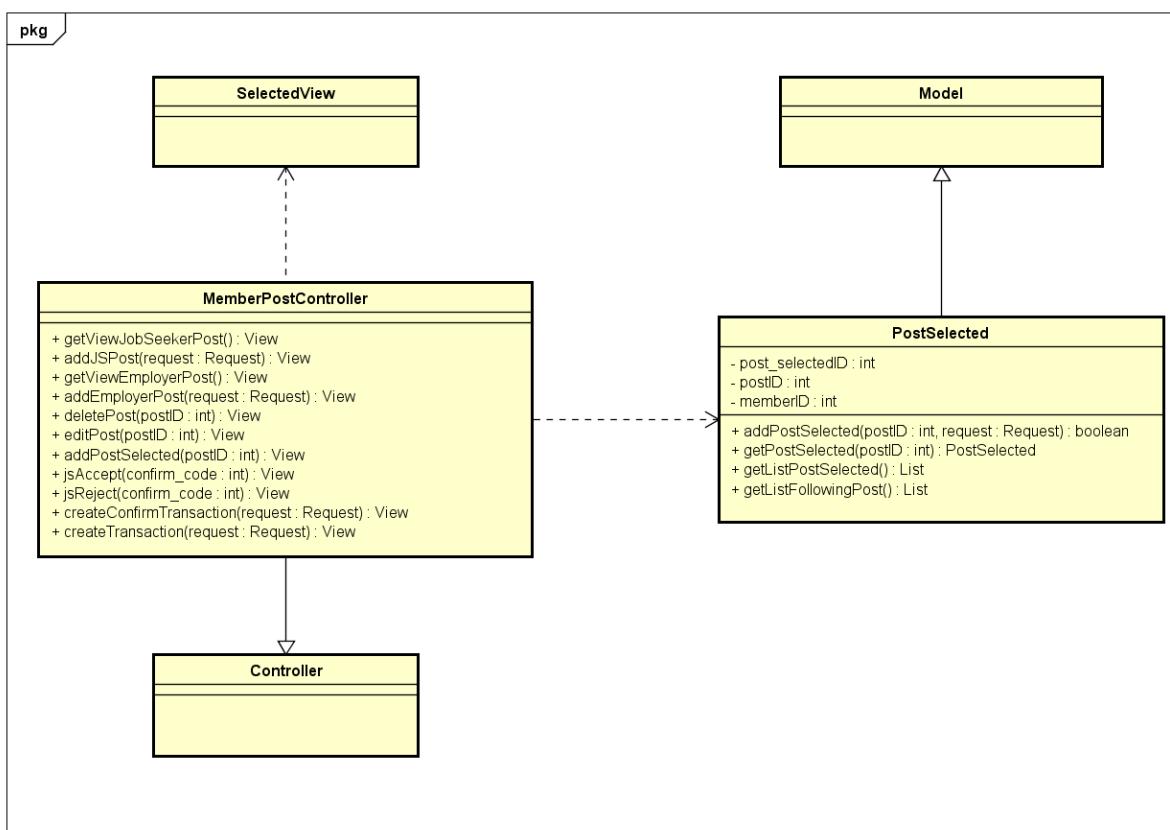
The screenshot shows a web application interface for creating a job post. The page title is "Tạo bài đăng tìm việc". The form fields include:

- Chủ đề công việc:** A text input field.
- Loại công việc:** A dropdown menu with options: Mặc định, Chăm sóc thú cưng, Phục vụ quán ăn, Chuyển đồ, Nấu ăn, Chăm sóc trẻ em, Gia sư, Bán hàng theo ca, Nội trợ.
- Mô tả công việc:** A text area for describing the job.
- Thời gian bắt đầu công việc:** A date input field in the format mm/dd/yyyy.
- Nơi làm việc:** A dropdown menu for selecting the city and district.
- Mức lương mong muốn (VND/Giờ):** A text input field for salary requirements.
- Mô tả bắn thắn:** A text area for additional notes.

At the bottom right of the form is a blue button labeled "HOÀN TẤT" (Finish).



4.7.2.4.1.3 Class Diagram

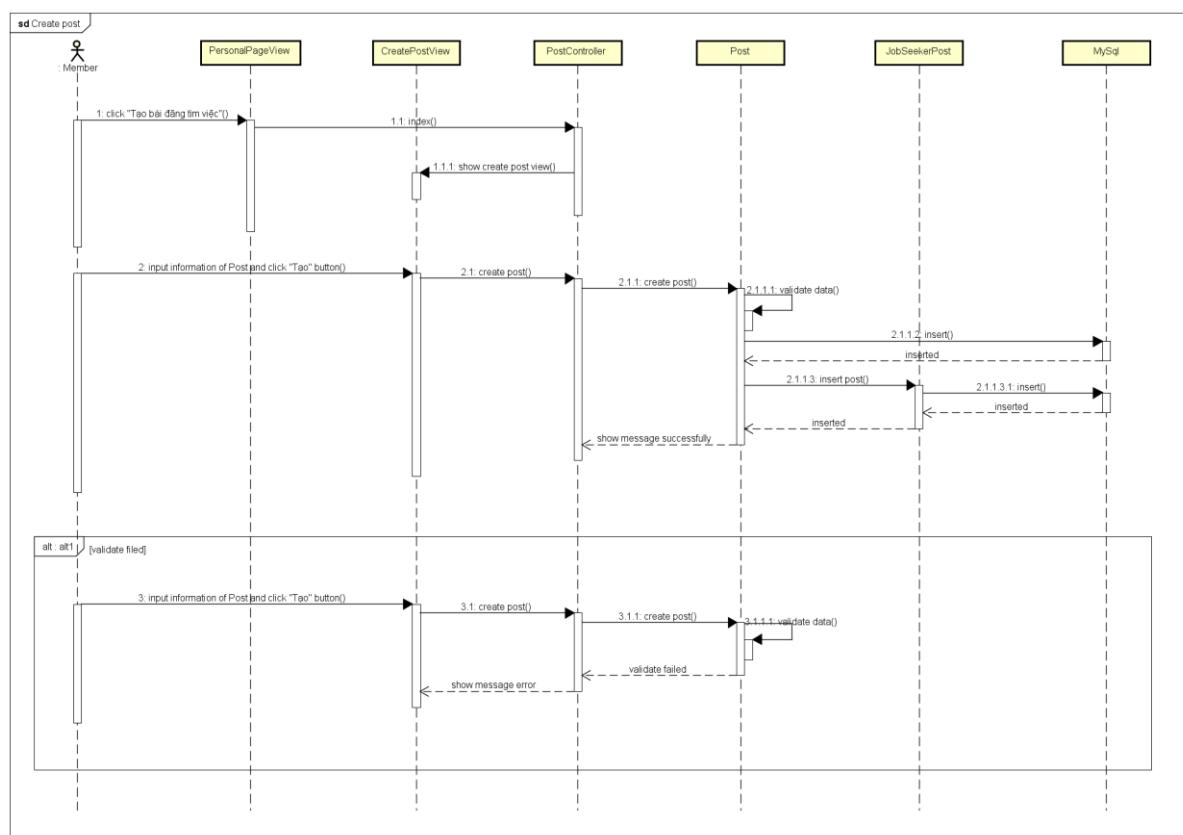


4.7.2.4.1.4 Class Specifications

Class	MemberPostController
Description	Control and do business relate to post
Base Class	Controller

Source File	MemberPostController.php		
Namespace	App\Http\Controllers\Member		
Attributes	N/A		
Methods	Name	Input	Output
	addJSPost	request	View list JobSeeker's post
	addEmployerPost	request	View list Employer's post
	getViewJobSeekerPost		View that create jobSeeker post
	getViewEmployerPost		View that create employer post

4.7.2.4.1.5 Sequence Diagram



4.7.2.4.2 Edit Post

4.7.2.4.2.1 Description

Member can edit posted Post's information.

4.7.2.4.2.2 Design

The screenshot shows a web-based application interface for managing posts. A modal window titled "Sửa bài đăng" (Edit Post) is open. Inside the modal, there are several input fields and dropdown menus. The "Chủ đề công việc" field contains "Dọn dẹp nhà cửa". The "Loại công việc" field has a checked checkbox for "Nâng đỡ". The "Mô tả công việc" field contains "Lau dọn nhà chuẩn bị cho Tết. Trang trí nhà theo yêu cầu của chủ nhà.". The "Mô tả bản thân" field contains "Nhanh nhẹn, chăm chỉ. Trung thực và có trách nhiệm với công việc.". The "Thời gian bắt đầu làm việc" field shows "12/31/2018". The "Nơi làm việc" field shows "Hà Nội". The "Quận/Huyện" field is a dropdown menu set to "Chọn Quận/Huyện". The "Mức lương mong muốn (VND/Giờ)" field shows "71000". At the bottom of the modal are two buttons: "Hoàn tất" (Finish) and "Hủy" (Cancel).

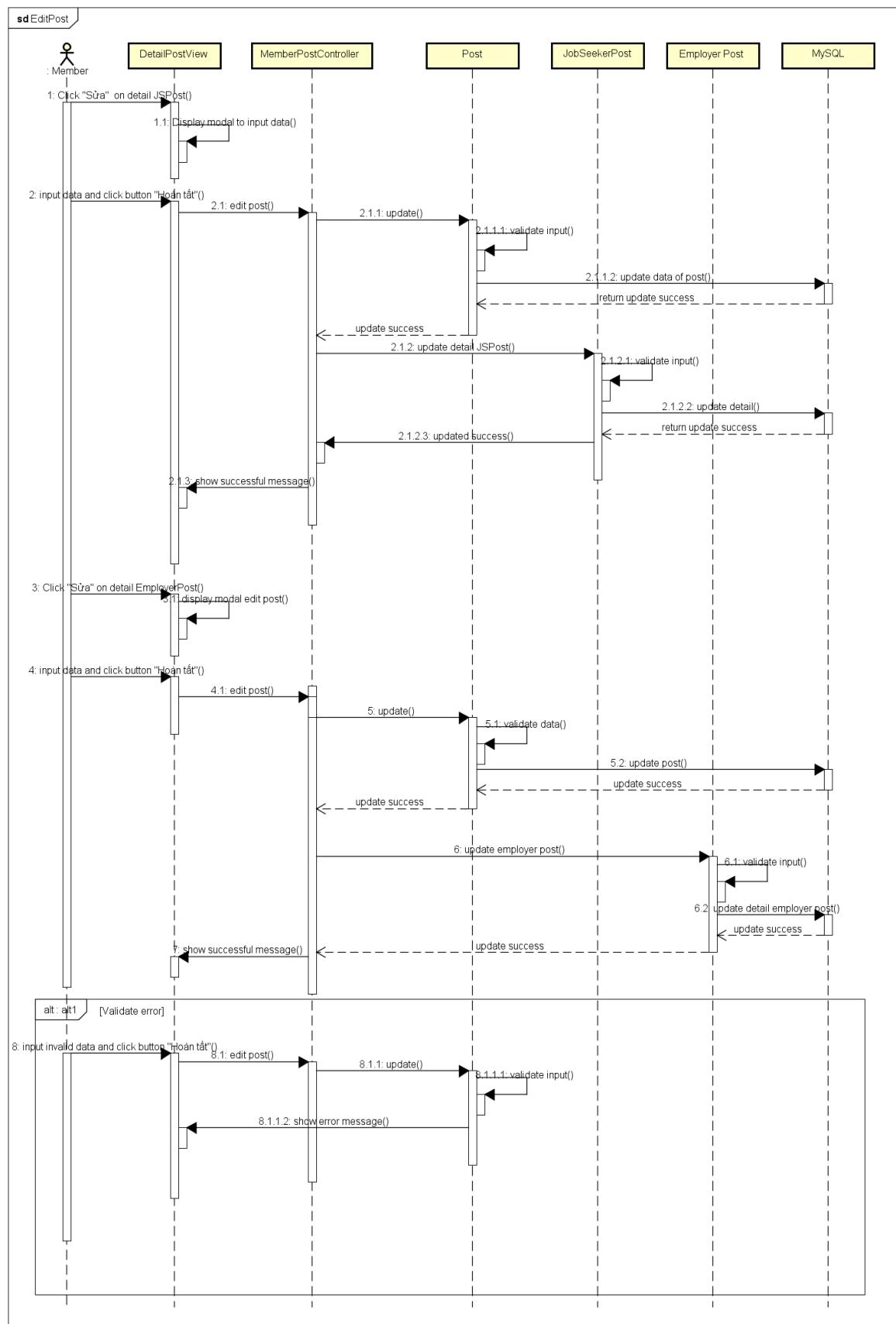
4.7.2.4.2.3 Class Diagram

Refer to class diagram of Create Post in 4.7.2.4.1 Create Post

4.7.2.4.2.4 Class Specifications

Class	MemberPostController			
Description	Control actions relate to post			
Base Class	Controller			
Source File	MemberPostController.php			
Namespace	App\Http\Controllers\Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	editPost	postID	View display new detail post	Update post information

4.7.2.4.2.5 Sequence Diagram



4.7.2.4.3 Delete Post

4.7.2.4.3.1 Description

This function allows a member to delete own post.

4.7.2.4.3.2 Design

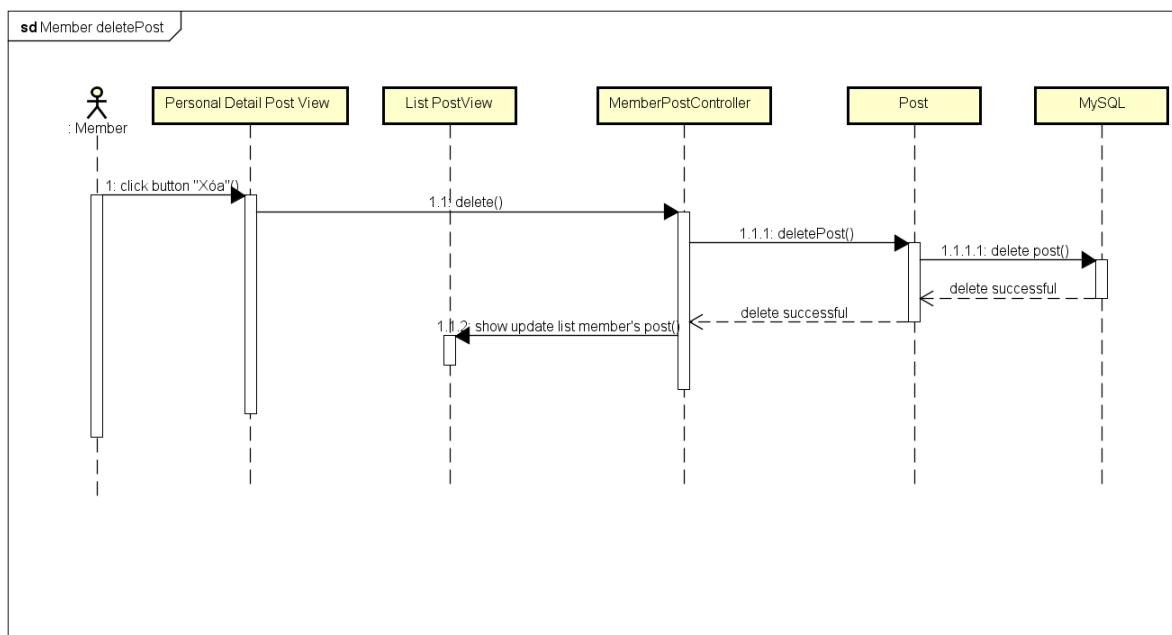
4.7.2.4.3.3 Class Diagram

Refer to class diagram of Create Post in 4.7.2.4.1 Create Post

4.7.2.4.3.4 Class Specifications

Class	MemberPostController			
Description	Control and do business relate to post			
Base Class	Controller			
Source File	MemberPostController.php			
Namespace	App\Http\Controllers\Member			
Attributes	N/A			
Methods	Name	Input	Output	Description
	deletePost	postID	View list jobseeker post or list employer post	Delete post from database

4.7.2.4.3.5 Sequence Diagram

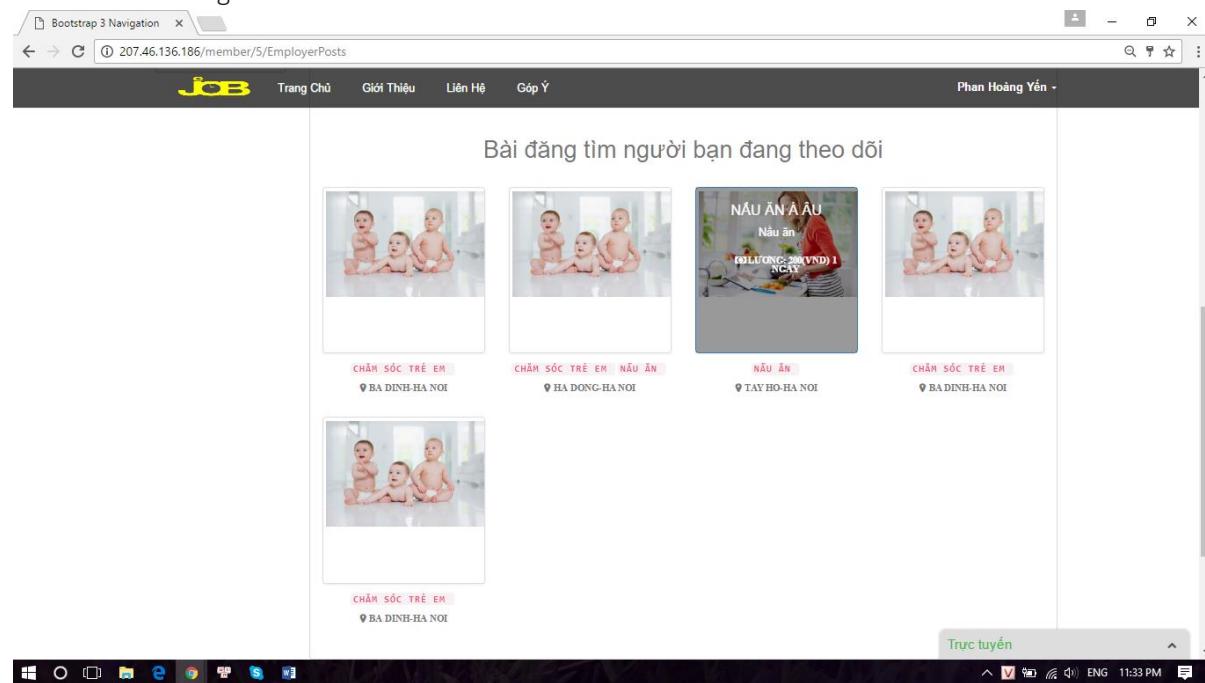


4.7.2.4.4 View selected Posts

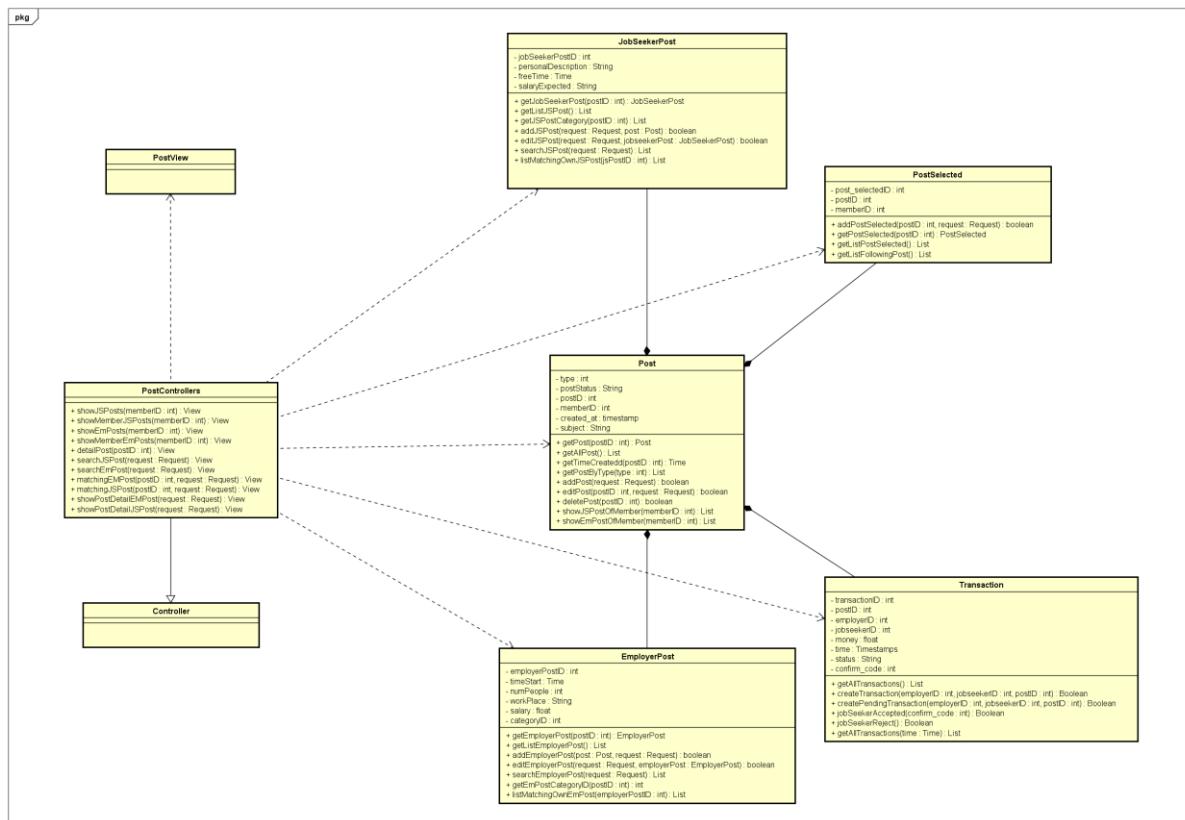
4.7.2.4.4.1 Description

Member can view posts that he was selected.

4.7.2.4.4.2 Design



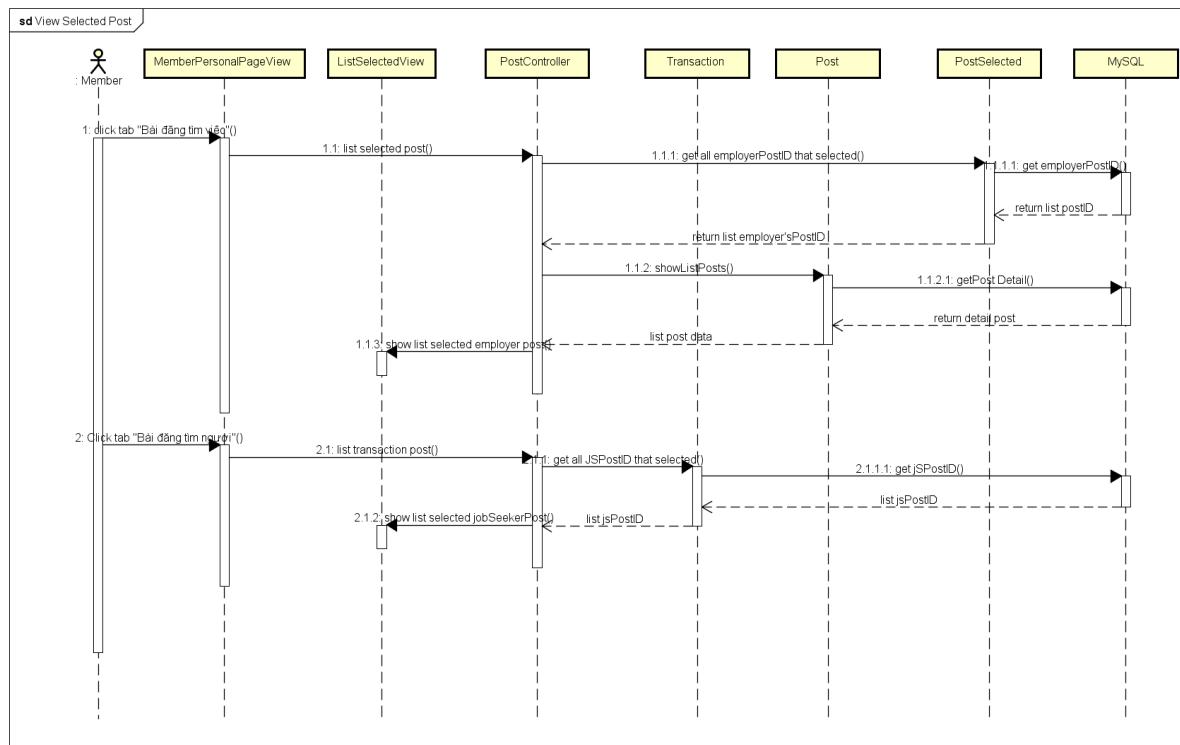
4.7.2.4.4.3 Class Diagram



4.7.2.4.4.4 Class Specifications

Class	PostController			
Description	Display all post that selected			
Base Class	Controller			
Source File	PostController.php			
Namespace	App\Http\Controllers\post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	showJSPosts	memberID	display list following post	Get all information posts that selected and render view
	showEmPosts	memberID	display list following post	Get all information posts that selected and render view

4.7.2.4.4.5 Sequence Diagram



4.7.2.4.5 View Posts

4.7.2.4.5.1 Description

Member can view posts that he was created.

4.7.2.4.5.2 Design

Refer to Design View personal page 4.7.2.4.4.2 View selected Post

4.7.2.4.5.3 Class Diagram

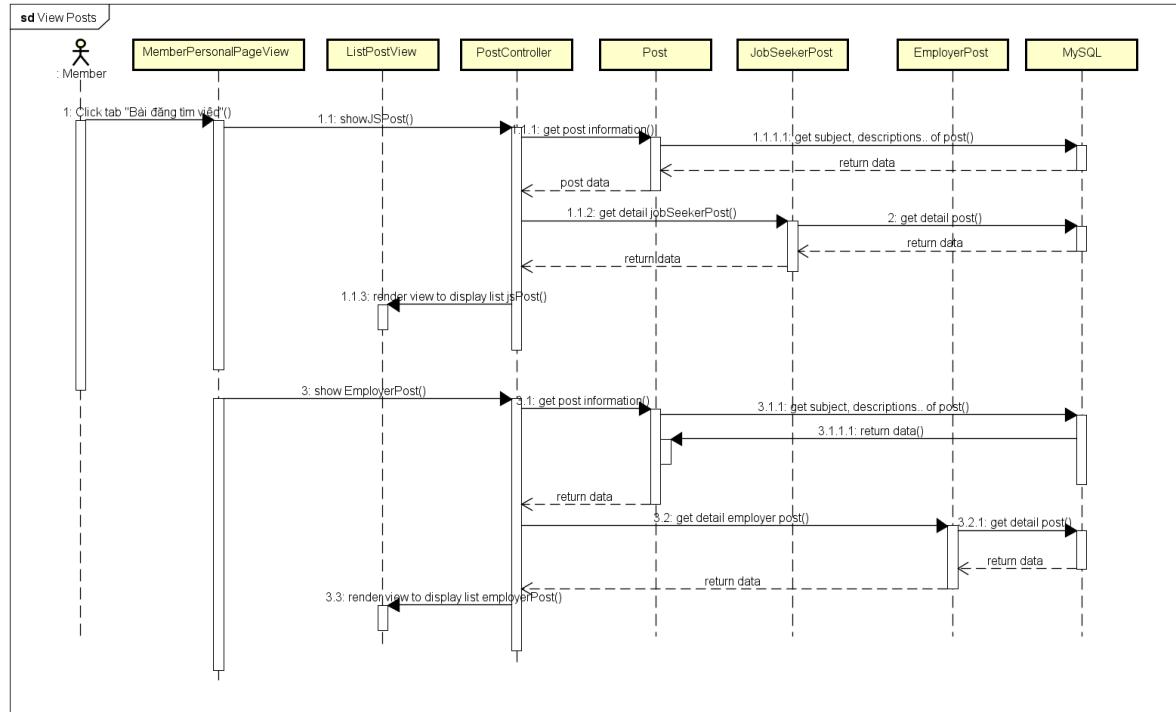
Refer to class diagram of Create Post in 4.7.2.4.1 Create Post

4.7.2.4.5.4 Class Specifications

Class	PostController		
Description	Control view post action		
Base Class	Controller		
Source File	PostController.php		
Namespace	App\Http\Controllers\Post		
Attributes	N/A		
Methods	Name	Input	Description
	showJSPosts	memberID	display list personal jobseeker post Get all information job seeker posts that personal created

	showEmPosts	memberID	display list following post	Get all information employer posts that personal created
--	-------------	----------	-----------------------------	--

4.7.2.4.5.5 Sequence Diagram



4.7.2.4.6 View detail Post

4.7.2.4.6.1 Description

Member can view detail of any post in system.

4.7.2.4.6.2 Design

The screenshot displays two adjacent browser windows. The left window shows a post detail page for a house cleaning service. The right window shows a related job listing for selling goods at a clothing store.

Left Window (Post Detail Page):

- Header:** eJob | 207.46.136.186/postdetail/43 | Phan Hoàng Yên
- Menu (Left):** Quản lý bài đăng, Bài đăng tìm việc, Bài đăng tìm người, Quản lý Giao dịch, Xem lịch sử các giao dịch, Quản lý tài khoản, Thông tin cá nhân, Thay mật khẩu.
- Content:**
 - [Người tìm việc] Dọn dẹp nhà cửa
 - Loại công việc: **Nhặt rác**
 - Nơi làm việc: **Hai Bà Trưng - Hà Nội | 3 phút trước**
 - Mức lương mong muốn: **71,000 (VND) 1 giờ**
 - Văn Quốc Việt
 - Thời gian bắt đầu làm việc: **2016-12-31**
 - Người đã chọn bạn**: Chấp nhận, Từ chối
 - Mô tả công việc**: Lau dọn nhà chuẩn bị cho Tết. Trang trí nhà theo yêu cầu của chủ nhà.
 - Mô tả bản thân**: Nhanh nhẹn, chăm chỉ. Trung thực và có trách nhiệm với công việc.
 - Sửa, Xóa**

Right Window (Related Job Listing):

- Header:** BÀI ĐĂNG LIÊN QUAN | Trực tuyến | 207.46.136.186/postdetail/32 | Phan Hoàng Yên
- Menu (Left):** Quản lý bài đăng, Bài đăng tìm việc, Bài đăng tìm người, Quản lý Giao dịch, Xem lịch sử các giao dịch, Quản lý tài khoản, Thông tin cá nhân, Thay mật khẩu.
- Content:**
 - [Việc tìm người] Bán hàng tại cửa hàng quần áo.
 - Loại công việc: **Bán hàng theo ca**
 - Nơi làm việc: **May - 43 Lê Duẩn - Hai Bà Trưng | 23 giờ trước**
 - Mức lương dự kiến: **50,000(VND) 1 giờ**
 - Thời gian làm việc: **2016-12-10**
 - Người đã chọn bạn**: tranhanam, Chọn người này
 - Văn Quốc Việt**: Chọn người này
 - Số lượng người**: 3
 - Mô tả công việc**: Trông coi shop. Cập nhật các mẫu mới của shop trên kén online. Tư vấn giúp khách hàng có những lựa chọn phù hợp với bản thân..
 - Yêu cầu khác**: Năng động, giao tiếp tốt..
 - Sửa, Xóa**

4.7.2.4.6.3 Class Diagram

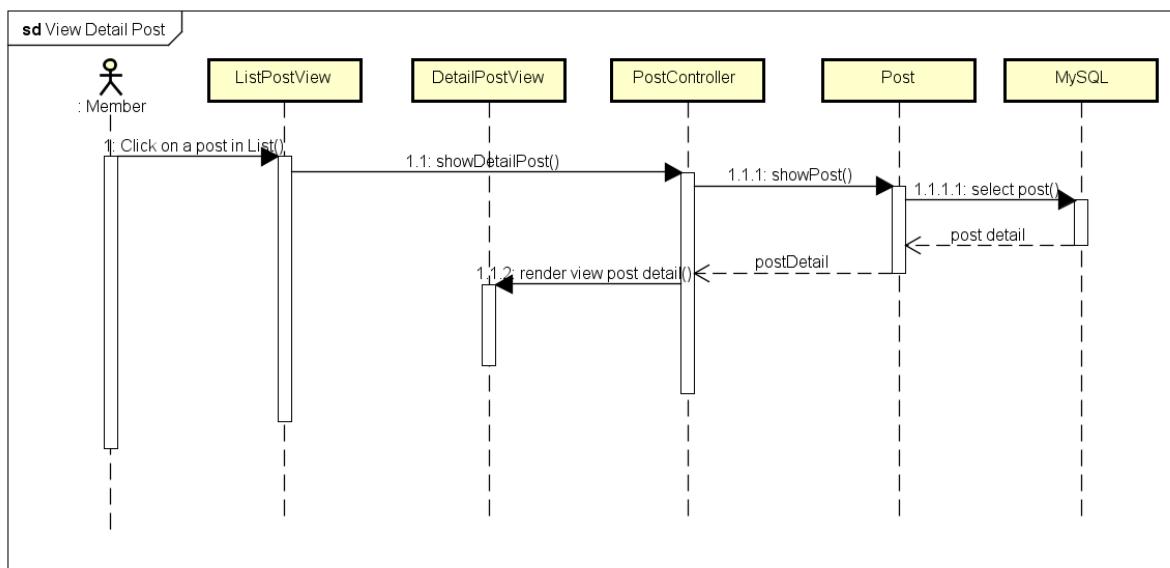
Refer to class diagram of Create Post in 4.7.2.4.1 Create Post

4.7.2.4.6.4 Class Specifications

Class	PostController
Description	Control all information of a post
Base Class	Controller
Source File	PostController.php
Namespace	App\Http\Controllers\Post

Attributes	N/A			
Methods	Name	Input	Output	Description
	detailPost	postID	View that display all information of a post	Show detail of a post to member
	showPostDetailEmPost	request	View that display all information of a employer post	Show detail of a employer post to member
	showPostDetailJSPost	request	View that display all information of a job seeker post	Show detail of a jobseeker post to member

4.7.2.4.6.5 Sequence Diagram



4.7.2.4.7 View people choose

4.7.2.4.7.1 Description

This function allows a member to view all people that was selected his post in Detail Post page

4.7.2.4.7.2 Design

Quản lý bài đăng

Bài đăng tìm việc

Bài đăng tìm người

Quản lý Giao dịch

Xem lịch sử các giao dịch

Quản lý tài khoản

Thông tin cá nhân

Thay mật khẩu

[Việc tìm người] Bán hàng tại cửa hàng quần áo.

Loại công việc: Bán hàng theo ca

Nơi làm việc: May - 43 Lê Duẩn - Hải Ba Trung | 23 giờ trước

Mức lương dự kiến: 50,000(VND) 1 giờ

Thời gian làm việc: 2016-12-10

Số lượng người: 3

Người đã chọn bạn

tranhanam Chọn người này

Văn Quốc Việt Chọn người này

Mô tả công việc

Trống coi shop. Cập nhật các mẫu mới của shop trên kenh online. Tư vấn giúp khách hàng có những lựa chọn phù hợp với bản thân..

Yêu cầu khác

Năng động, giao tiếp tốt..

Sửa Xóa

BÀI ĐĂNG LIÊN QUAN

Sắp xếp theo thứ tự ưu tiên: [Thời gian: Trong 5 ngày] - [Địa chỉ: Cùng quận huyện] - [Mức lương: Tăng dần]

Thời gian Địa chỉ Mức lương

Trực tuyến

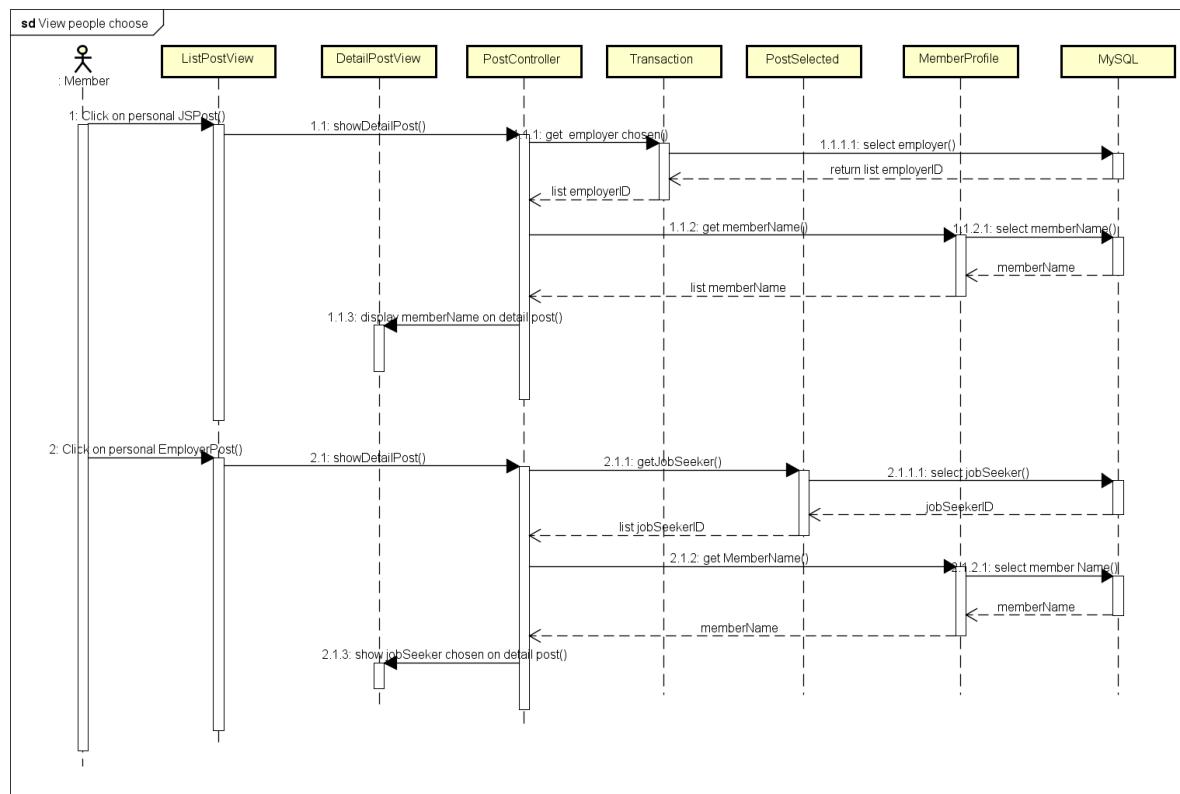
4.7.2.4.7.3 Class Diagram

Refer to class diagram of Create Post in 4.7.2.4.1 Create Post

4.7.2.4.7.4 Class Specifications

Class	PostController			
Description	Control showing member who choose the post			
Base Class	Controller			
Source File	PostController.php			
Namespace	App\Http\Controllers\Post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	detailPost	postID	View that display member who chosen this post	Show list member who chosen this post

4.7.2.4.7.5 Sequence Diagram

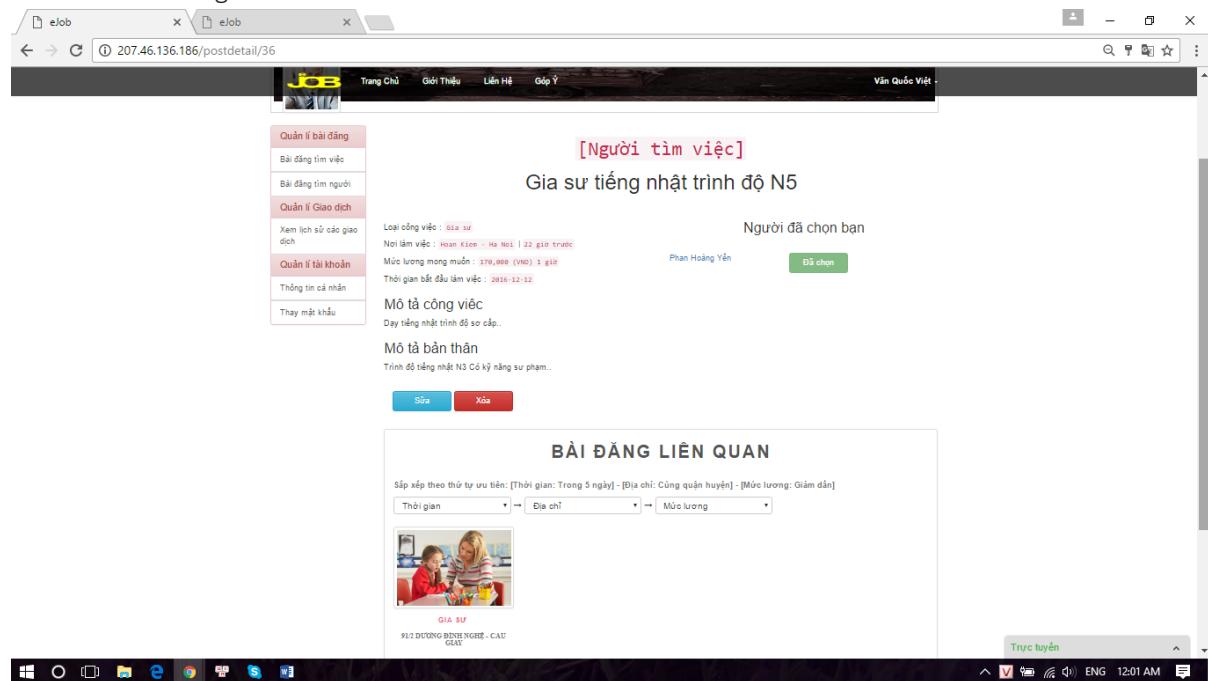


4.7.2.4.8 View matching Posts

4.7.2.4.8.1 Description

This function allows a member to view matching Posts in detail post page.

4.7.2.4.8.2 Design



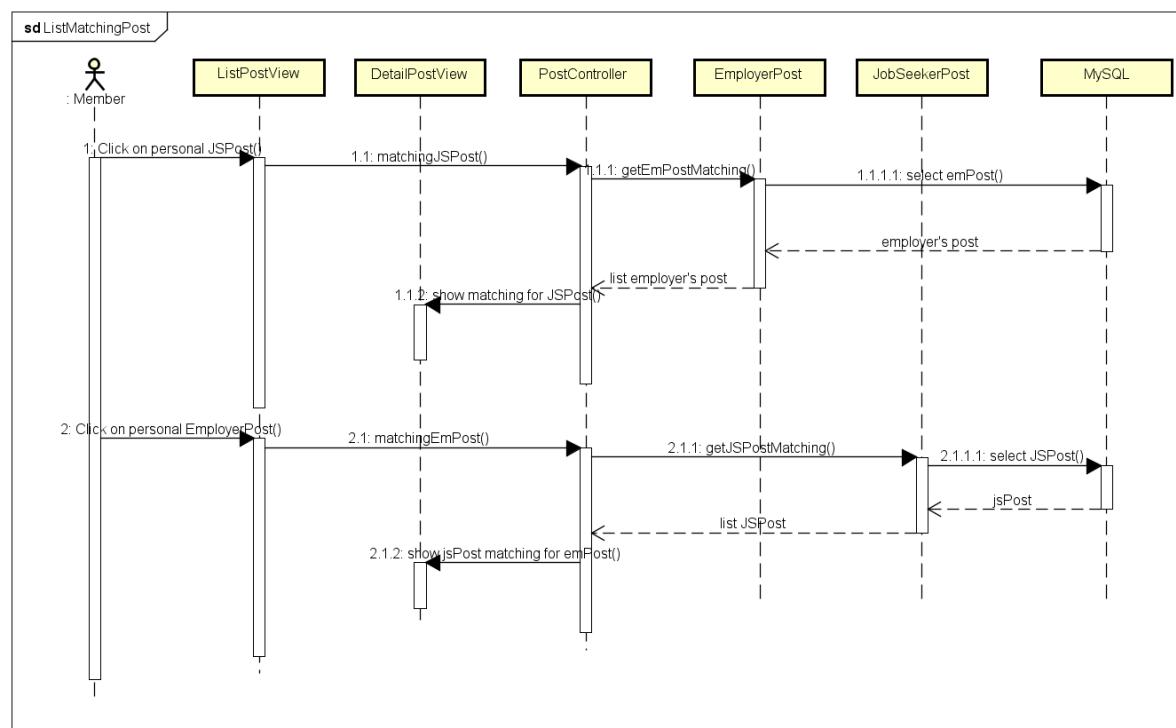
4.7.2.4.8.3 Class Diagram

Refer to class diagram of Create Post in 4.7.2.4.1 Create Post

4.7.2.4.8.4 Class Specifications

Class	PostController			
Description	Control display matching posts with detail of a post			
Base Class	Controller			
Source File	PostController.php			
Namespace	App\Http\Controllers\ PostController			
Attributes	N/A			
Methods	Name	Input	Output	Description
	matchingEmPost	postID,request	View that display list jobseeker post that matching with this post	Select and display list jobseeker post that matching
	matchingJSPost	postID,request	View that display list employer post that matching with this post	Select and display list employer post that matching

4.7.2.4.8.5 Sequence Diagram



4.7.3 Admin 管理

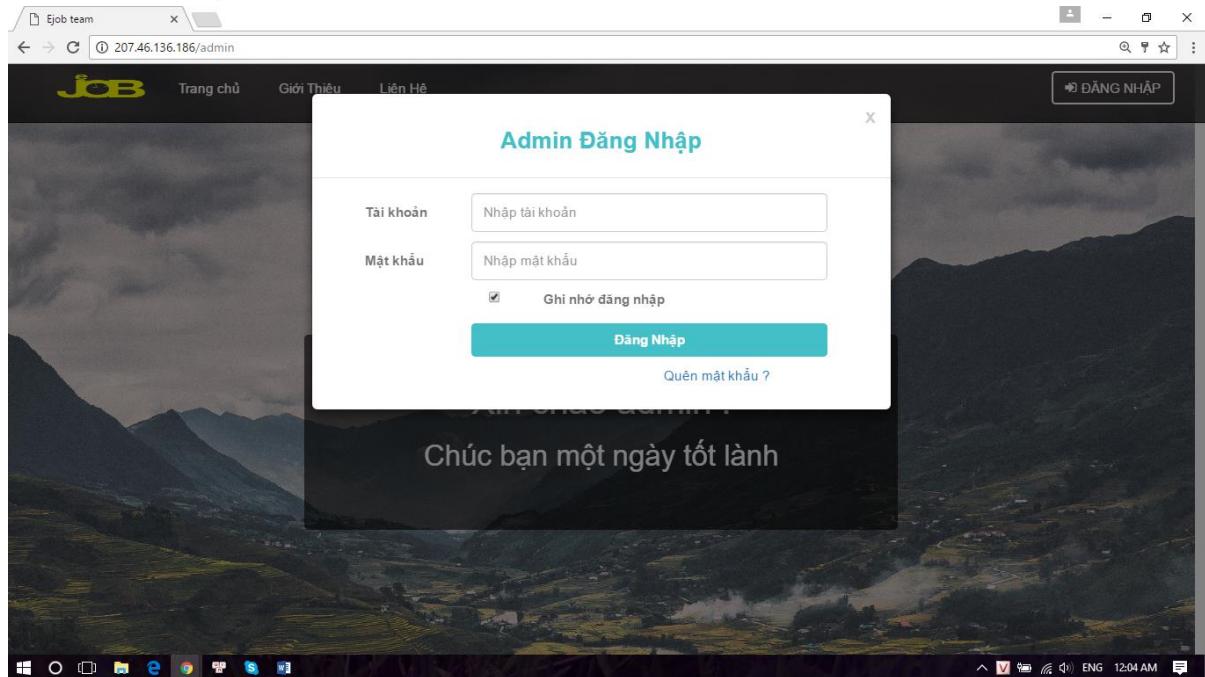
4.7.3.1 Common

4.7.3.1.1 Admin Login

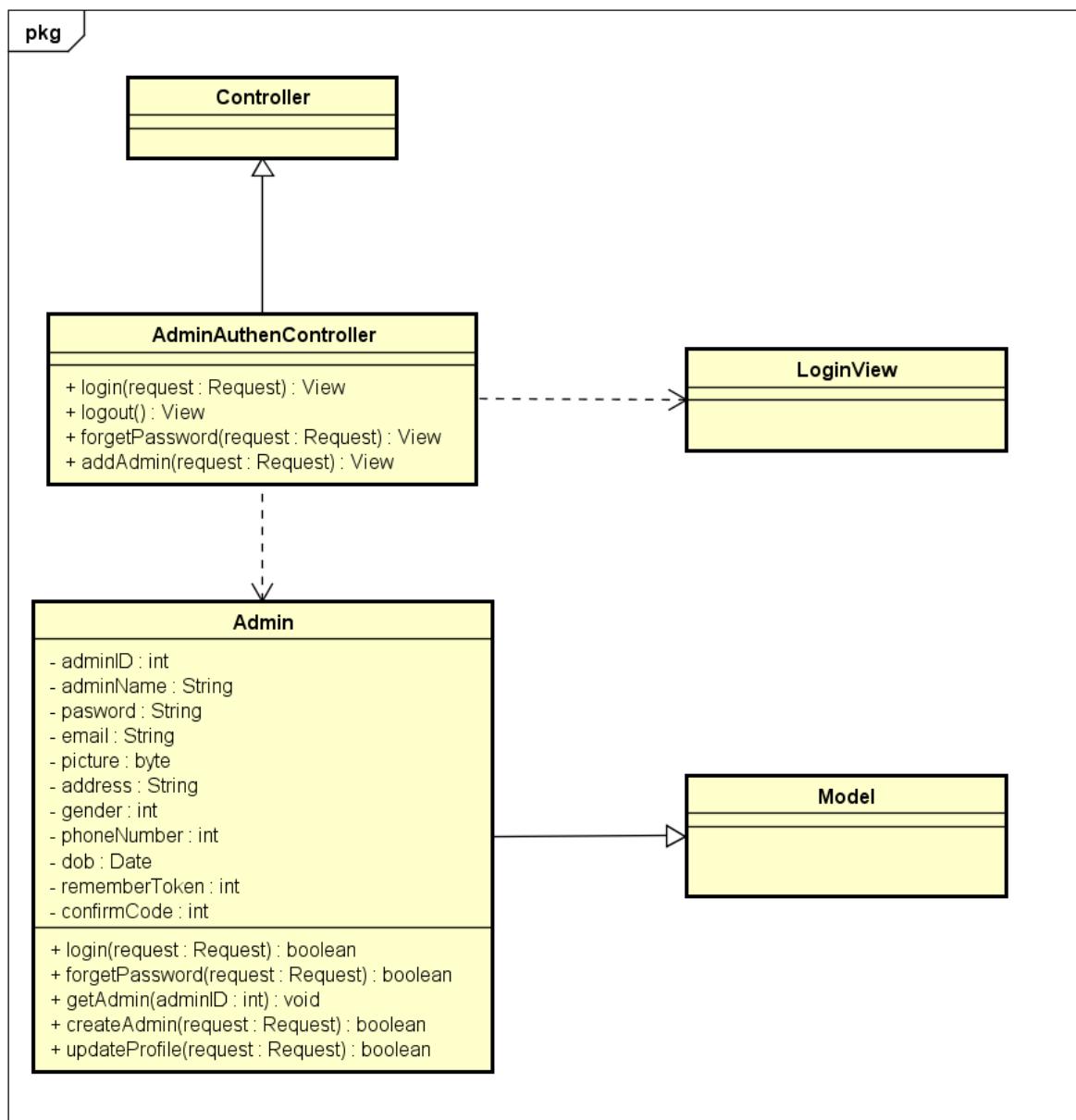
4.7.3.1.1.1 Description

This function allows an admin to login with an administrator account.

4.7.3.1.1.2 Design



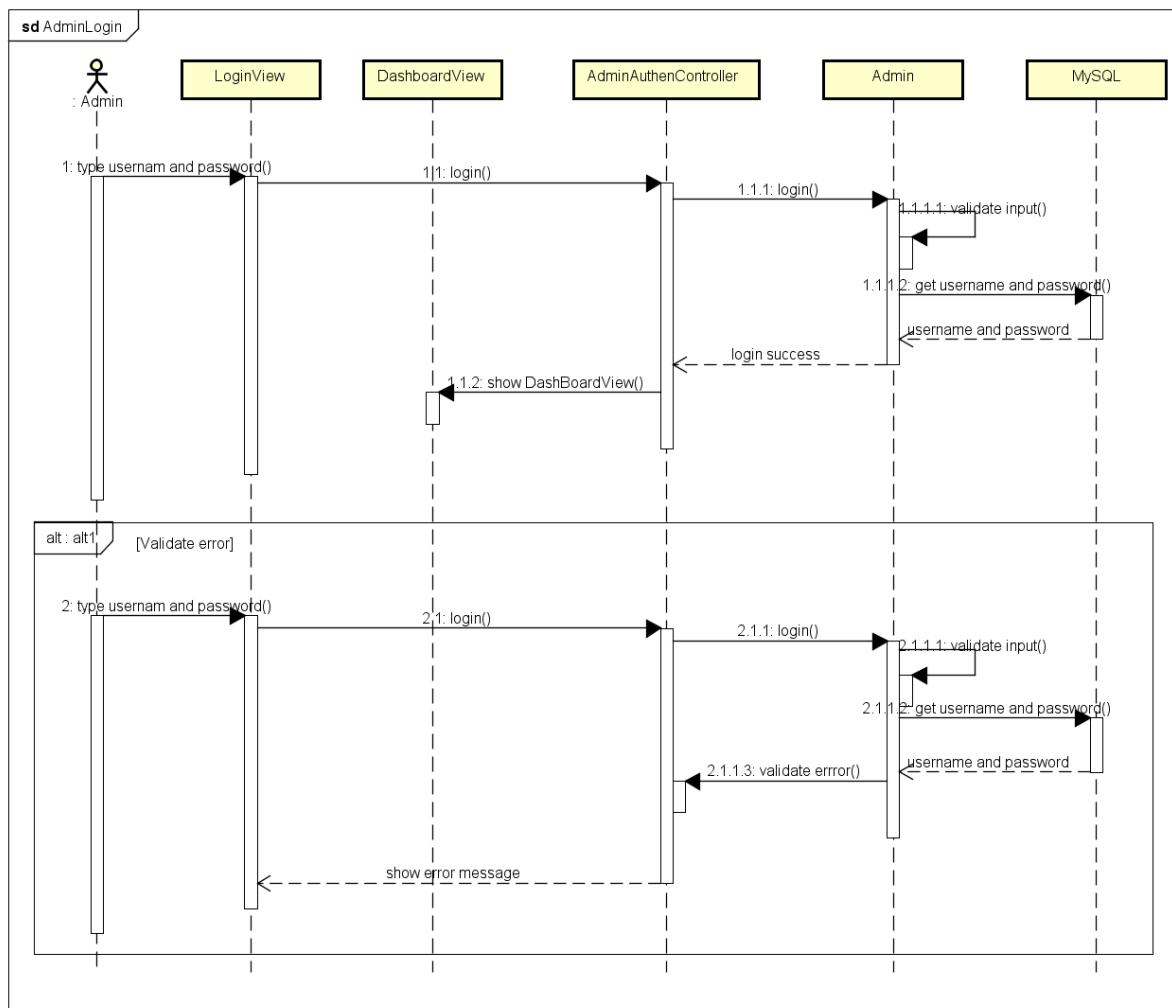
4.7.3.1.1.3 Class Diagram



4.7.3.1.1.4 Class Specifications

Class	AdminAuthenController			
Description	Control admin login to system			
Base Class	Controller			
Source File	AdminAuthenController.php			
Namespace	App\Http\Controllers\Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description
	login	request	Dashboard View	Admin login to the system

4.7.3.1.1.5 Sequence Diagram



4.7.3.1.2 Admin Logout

4.7.3.1.2.1 Description

This function allows a member to logout.

4.7.3.1.2.2 Class Diagram

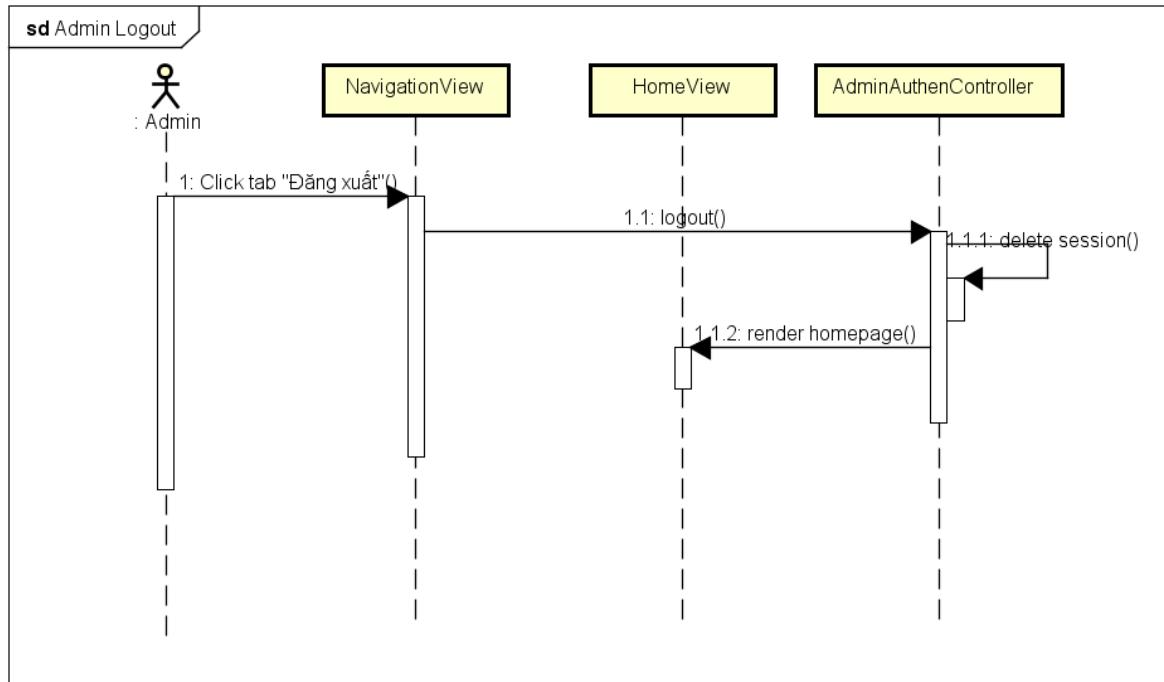
Refer to class diagram of Admin Login in 4.7.3.1.1 Admin Login

4.7.3.1.2.3 Class Specifications

Class	AdminAuthenController			
Description	Control admin logout action			
Base Class	Controller			
Source File	AdminAuthenController.php			
Namespace	App\Http\Controllers\Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description

	logout	N/A	Logout and back to homepage	Logout from system
--	--------	-----	-----------------------------	--------------------

4.7.3.1.2.4 Sequence Diagram

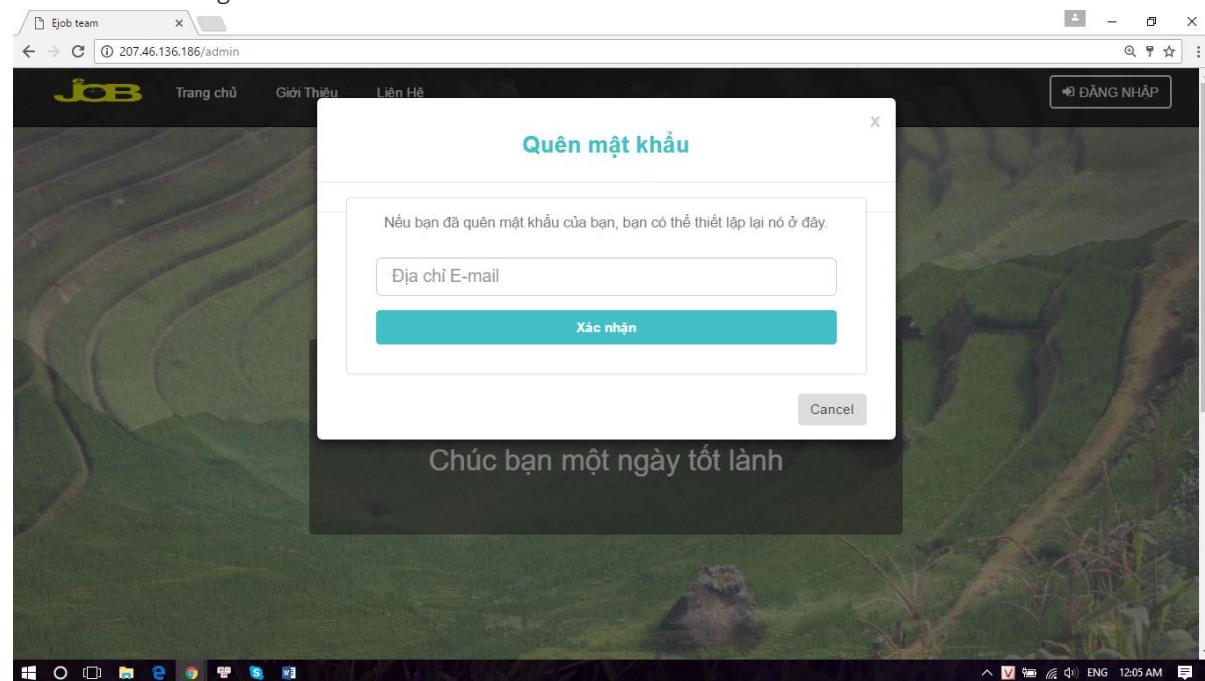


4.7.3.1.3 Admin Forgot password

4.7.3.1.3.1 Description

When Admin forgot password that used to login to system, use this function to receive new password by email.

4.7.3.1.3.2 Design



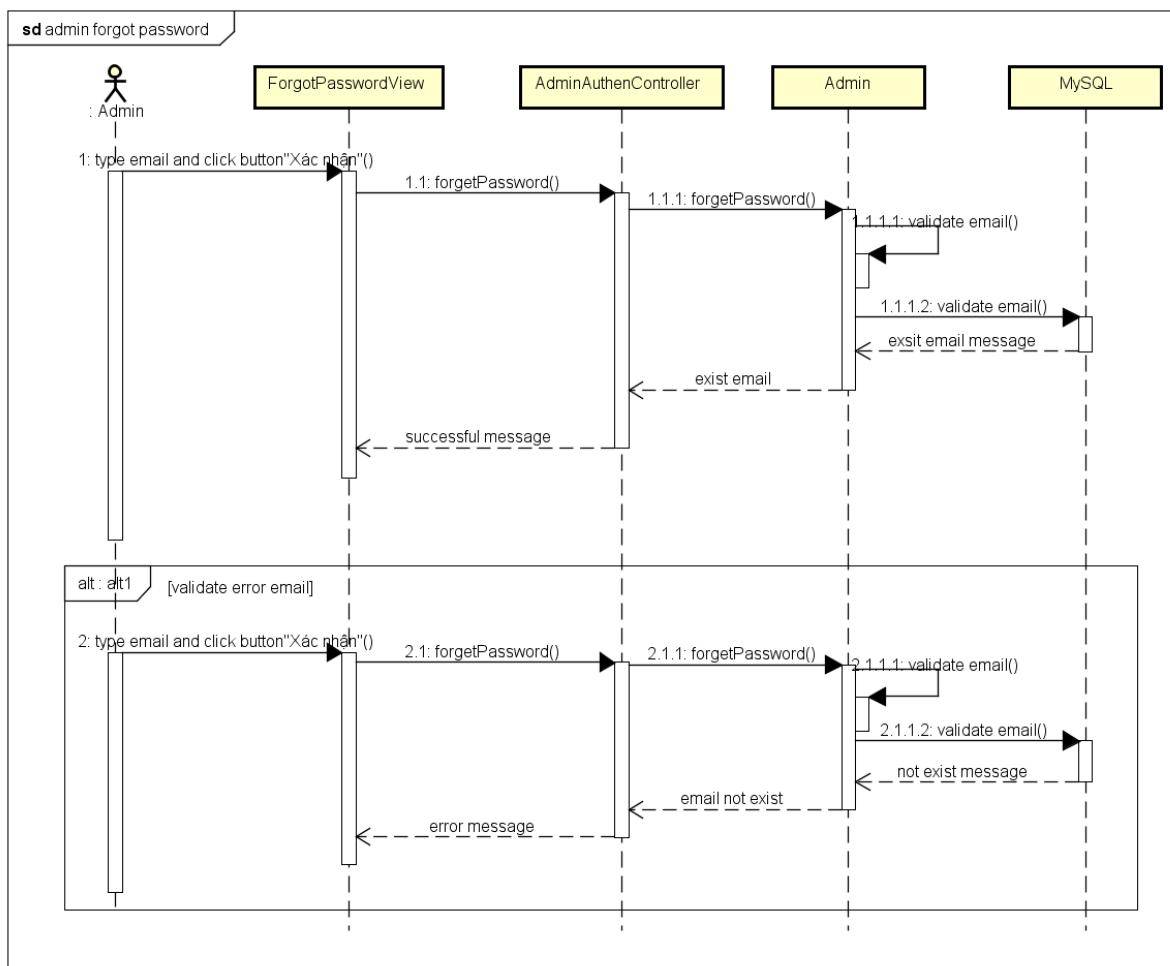
4.7.3.1.3.3 Class Diagram

Refer to class diagram of Admin Login in 4.7.3.1.1 Admin Login

4.7.3.1.3.4 Class Specifications

Class	AdminAuthenController			
Description	Control admin action			
Base Class	Controller			
Source File	AdminAuthenController.php			
Namespace	App\Http\Controllers\Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description
	forgetPassword	request	New password in mail	After input email address, admin will received new password in mail

4.7.3.1.3.5 Sequence Diagram



4.7.3.1.4 Online chat with Member

4.7.3.1.4.1 Description

Member chat online with admin. When Admin online (in session), Member create message and this message is sent to Admin and displayed on time by Tawk.to. Admin also can reply Member's message, and this message also sent to Member and displayed on time. When a message is sent during the time Admin or Member offline (out of session), message is stored and directly displayed when Admin or Member online. All message that be seen will not be stored.

4.7.3.1.4.2 Design

4.7.3.1.4.3 Class Diagram

4.7.3.1.4.4 Class Specifications

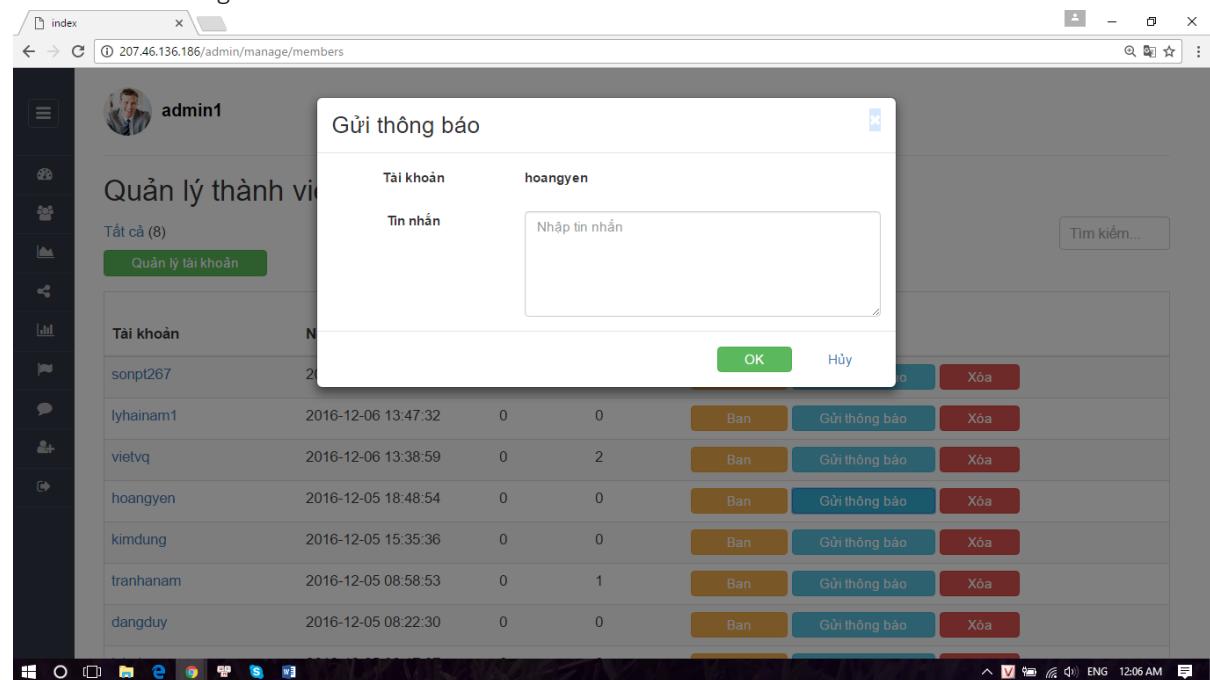
4.7.3.1.4.5 Sequence Diagram

4.7.3.1.5 Send notify by email

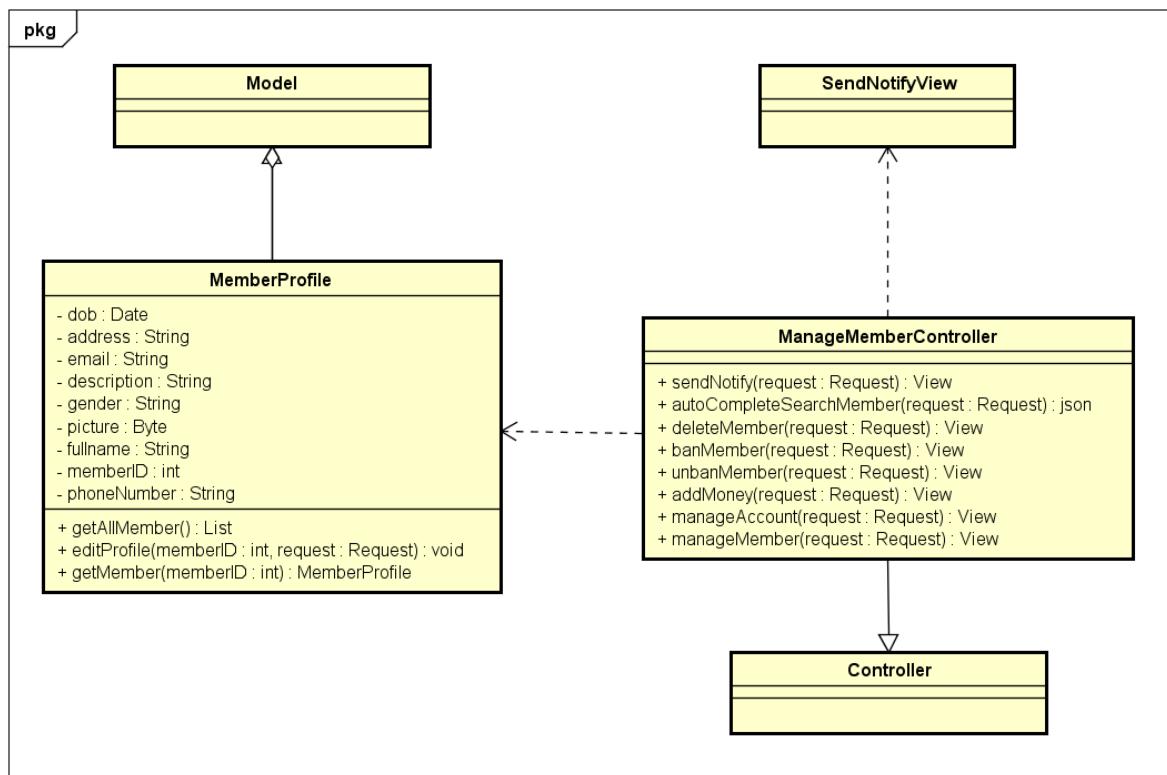
4.7.3.1.5.1 Description

Admin can send email to Member's email.

4.7.3.1.5.2 Design



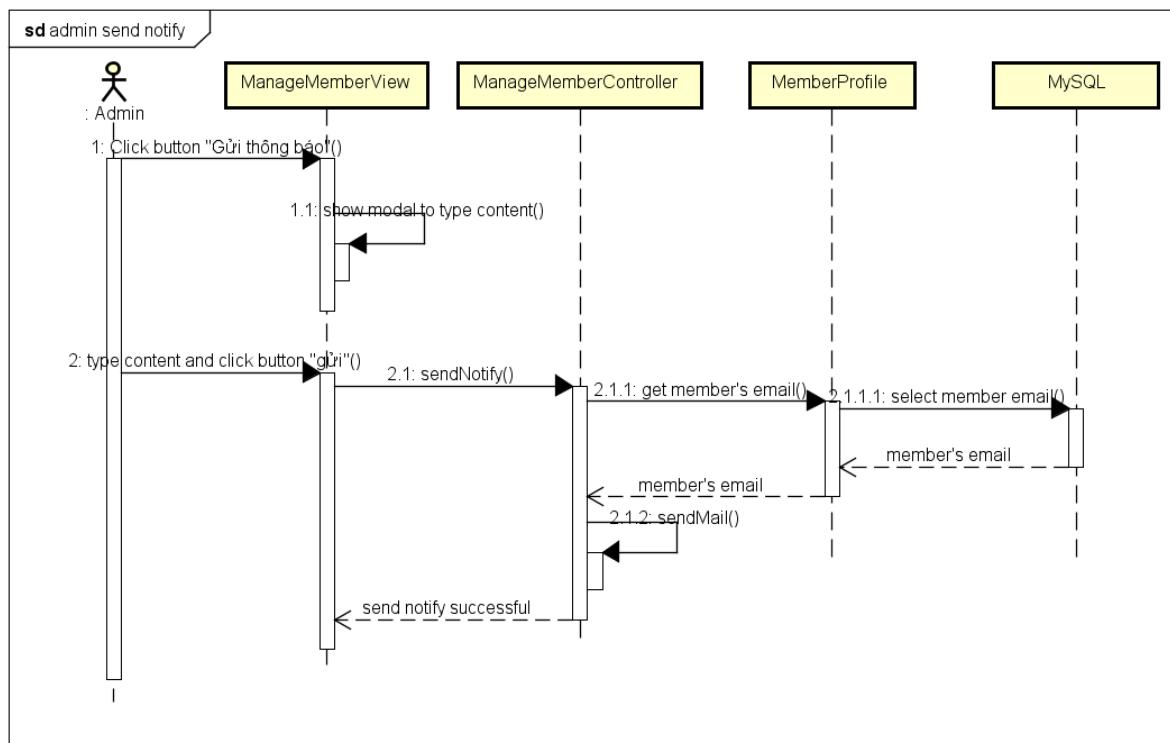
4.7.3.1.5.3 Class Diagram



4.7.3.1.5.4 Class Specifications

Class	ManageMemberController			
Description	Control send notify to member by email			
Base Class	Controller			
Source File	ManageMemberController.php			
Namespace	App\Http\Controllers\Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description
	sendNotify	request	View manage member and successful message	Create new notify and send by email from admin to members.

4.7.3.1.5.5 Sequence Diagram



4.7.3.1.6 Add a Admin

4.7.3.1.6.1 Description

Admin can add new admin.

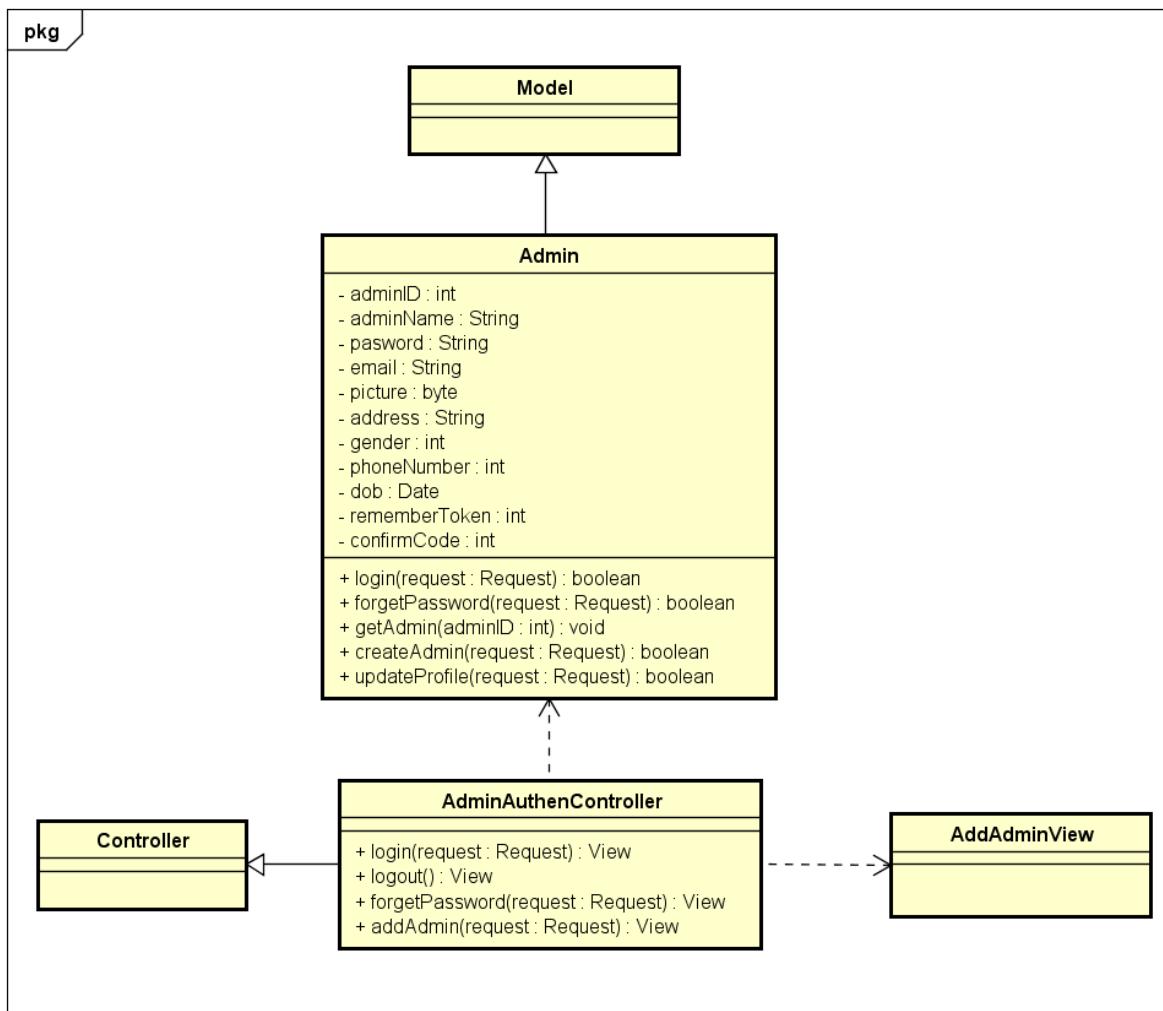
4.7.3.1.6.2 Design

The screenshot shows a web application interface for adding a new administrator. The title bar indicates the URL is 207.46.136.186/admin/add. The form is titled 'Tạo thêm quản trị viên'. It contains fields for:

- Tên đăng nhập *: Nhập tên đăng nhập
- Mật khẩu *: Nhập Mật khẩu
- Xác nhận mật khẩu *: Nhập lại mật khẩu
- Email *: Nhập Email
- Số điện thoại *: Nhập số điện thoại
- Họ và Tên *: Nhập Họ và tên
- Giới tính: Radio buttons for Nam and Nữ
- Ngày sinh: mm/dd/yyyy
- Địa chỉ: Nhập Địa chỉ

A red error message at the bottom left reads: "Vui lòng nhập vào các trường có dấu (*)". At the bottom right are 'HỦY BỎ' and 'HOÀN TẤT' buttons. The status bar at the bottom shows: Waiting for demo.neontheme.com... and ENG 12:07 AM.

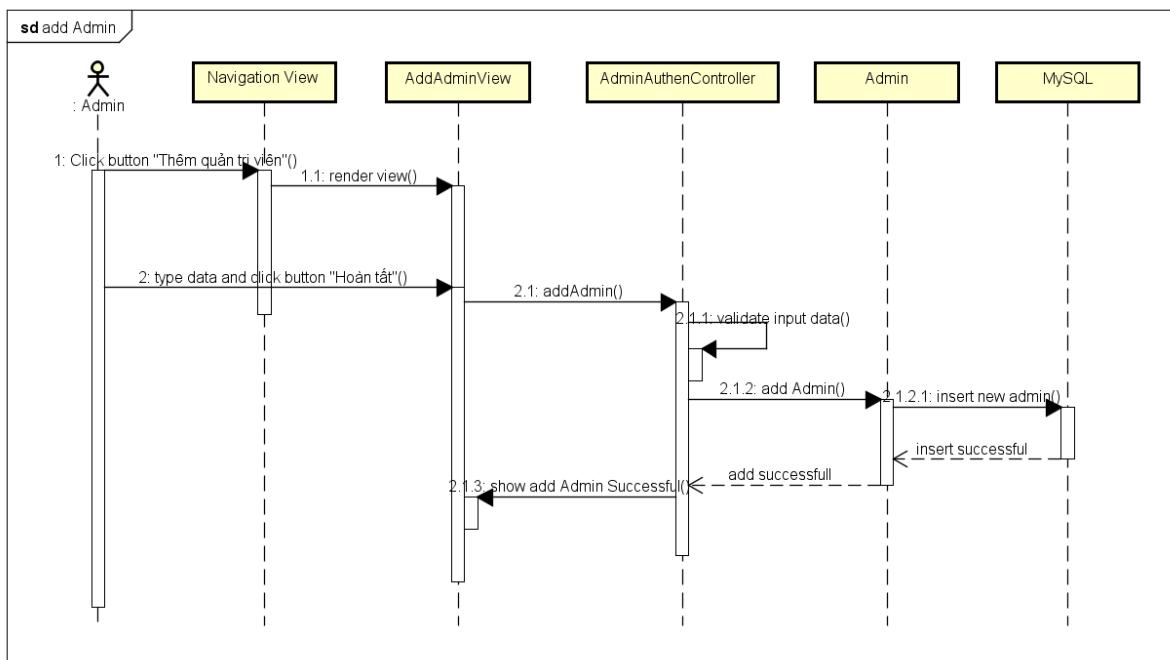
4.7.3.1.6.3 Class Diagram



4.7.3.1.6.4 Class Specifications

Class	AdminAuthenController			
Description	Setting for Admin can create another admin			
Base Class	Controller			
Source File	AdminAuthenController.php			
Namespace	App\Http\Controllers\Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description
	addAdmin	request	Create new admin successful	Create new admin account to control system

4.7.3.1.6.5 Sequence Diagram



4.7.3.2 Manage Members

4.7.3.2.1 View Members

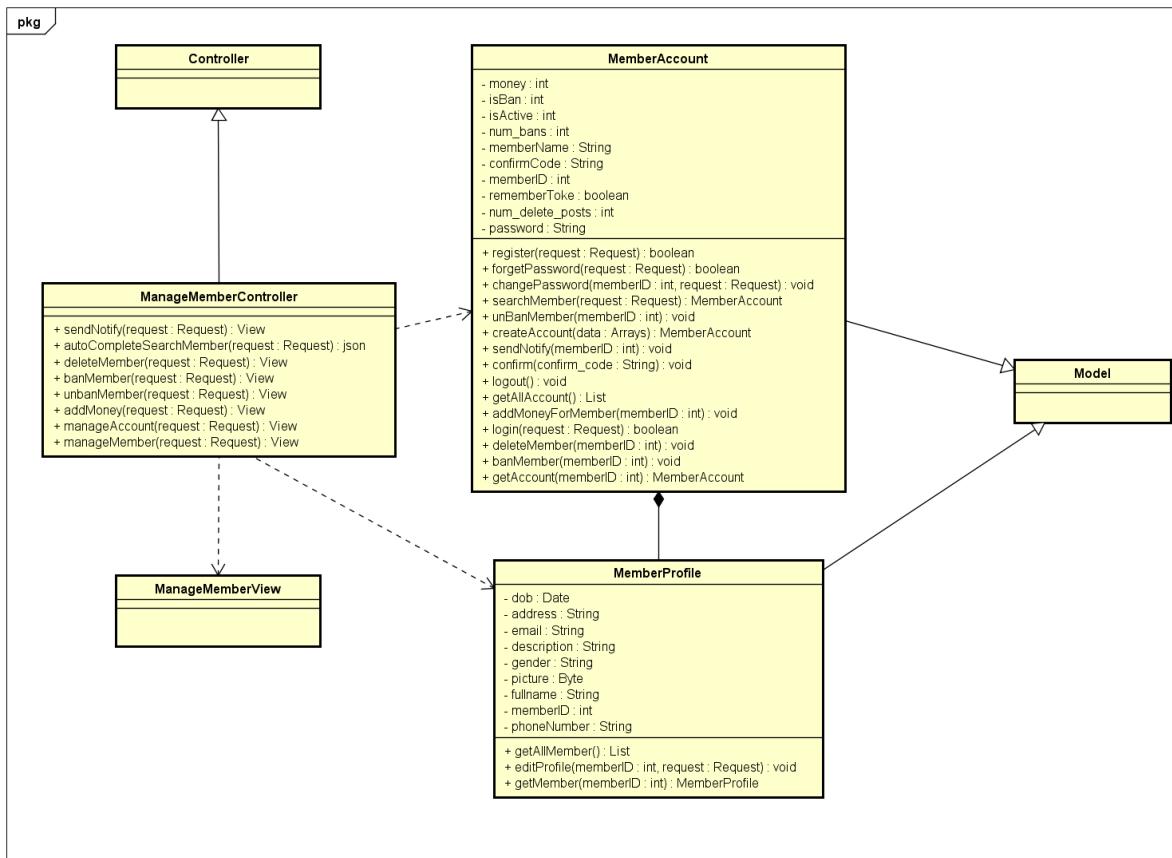
4.7.3.2.1.1 Description

Admin can view list of all Member of system.

4.7.3.2.1.2 Design

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
sonpl267	2016-12-07 01:57:52	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
vietvq	2016-12-06 13:38:59	0	2	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
kimdung	2016-12-05 15:35:36	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
dangduy	2016-12-05 08:22:30	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

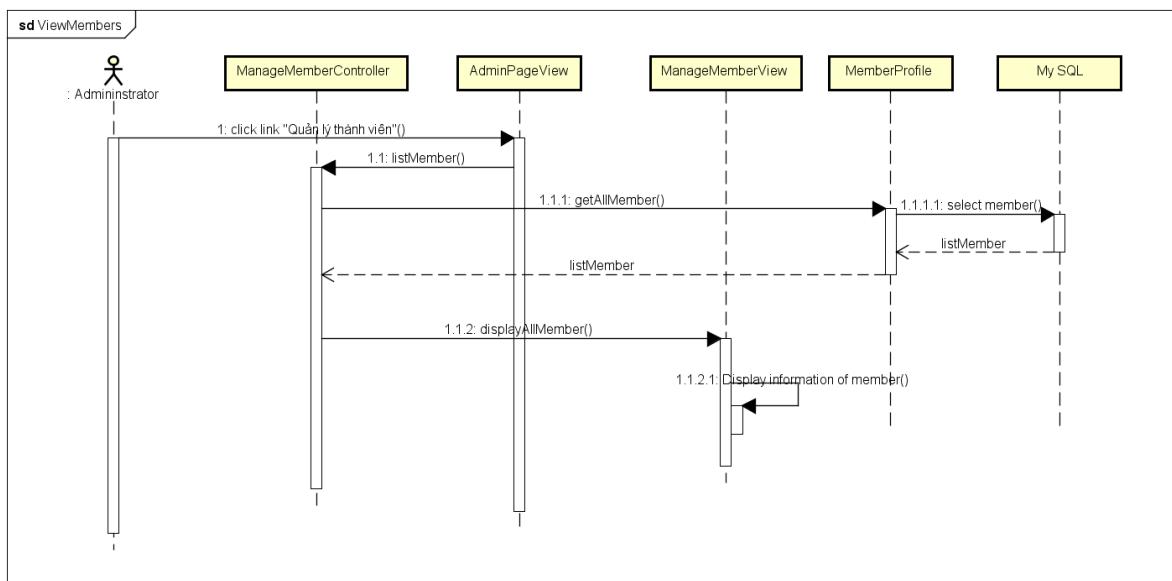
4.7.3.2.1.3 Class Diagram



4.7.3.2.1.4 Class Specifications

Class	ManageMemberController			
Description	Allows admin to view list of all member's accounts of system.			
Base Class	Controller			
Source File	ManageMemberController.php			
Namespace	App /Http /Controllers/Admin/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	manageMember	request	View that display all of member	Select all member and display to manage view

4.7.3.2.1.5 Sequence Diagram



4.7.3.2.2 Search Members

4.7.3.2.2.1 Description

Admin can search for Members that have specific information.

4.7.3.2.2.2 Design

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
dangduy	2016-12-05 08:22:30	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam	2016-12-05 08:17:07	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

4.7.3.2.2.3 Class Diagram

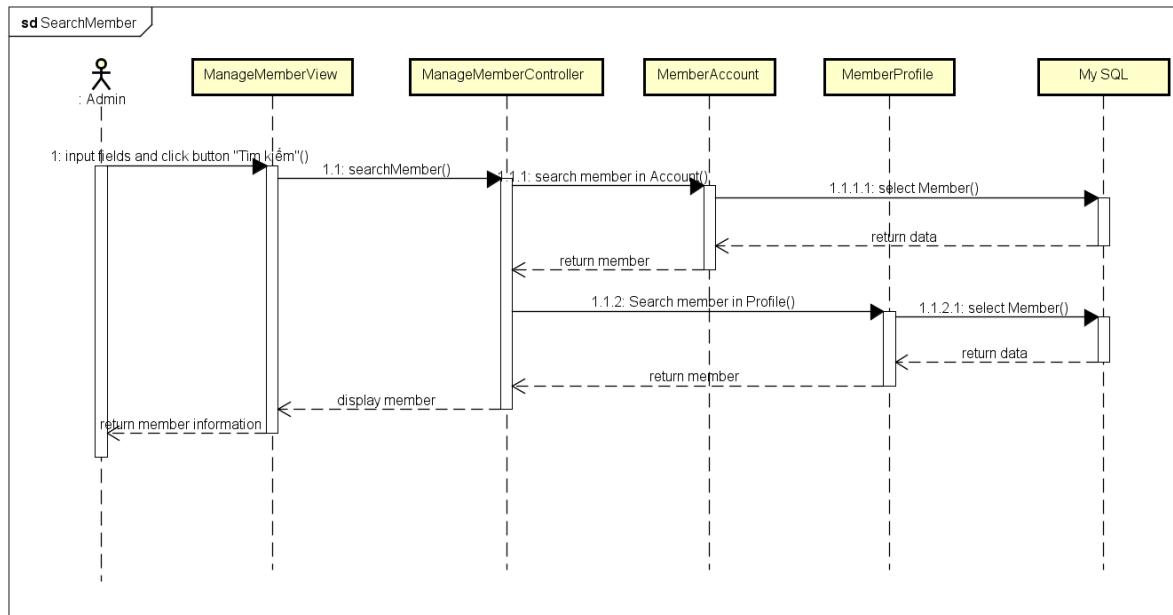
Refer to class diagram of View Member in 4.7.3.2.1 View Member

4.7.3.2.2.4 Class Specifications

Class	ManageMemberController
Description	Search member by ID, memberName

Base Class	Controller			
Source File	ManageMemberController.php			
Namespace	App /Http /Controllers/Admin/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	autoCompleteSearchMember	request	Member object	Get member and some information of member

4.7.3.2.2.5 Sequence Diagram

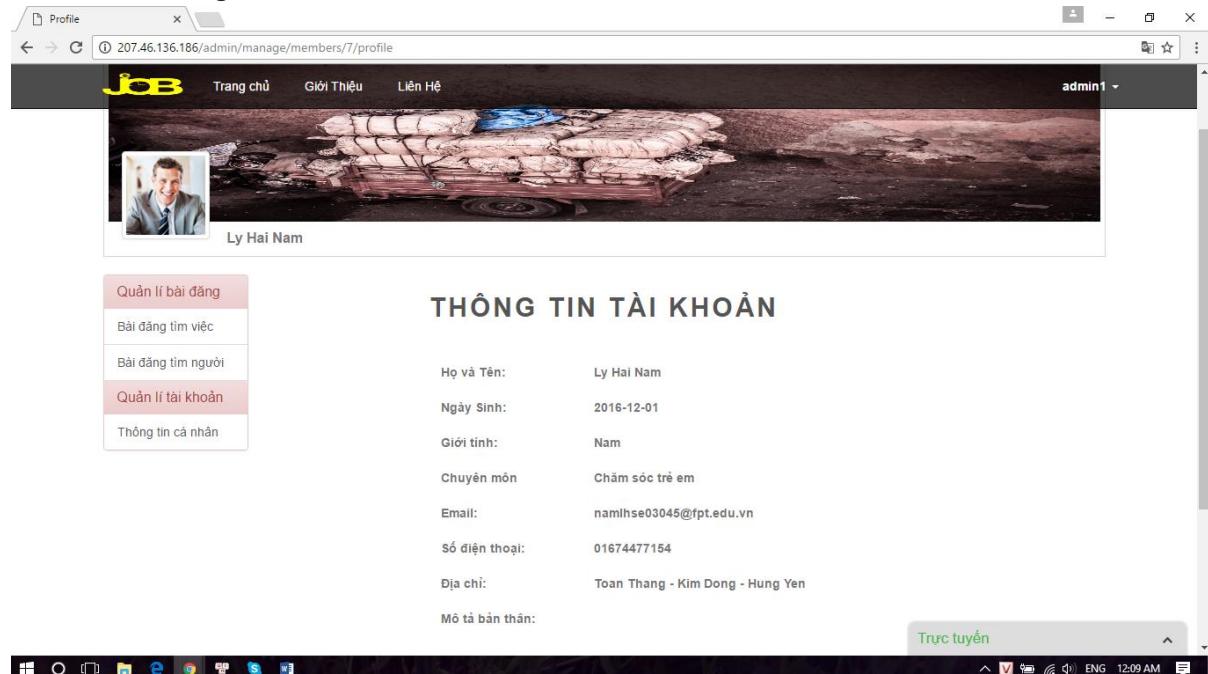


4.7.3.2.3 View Member Profile

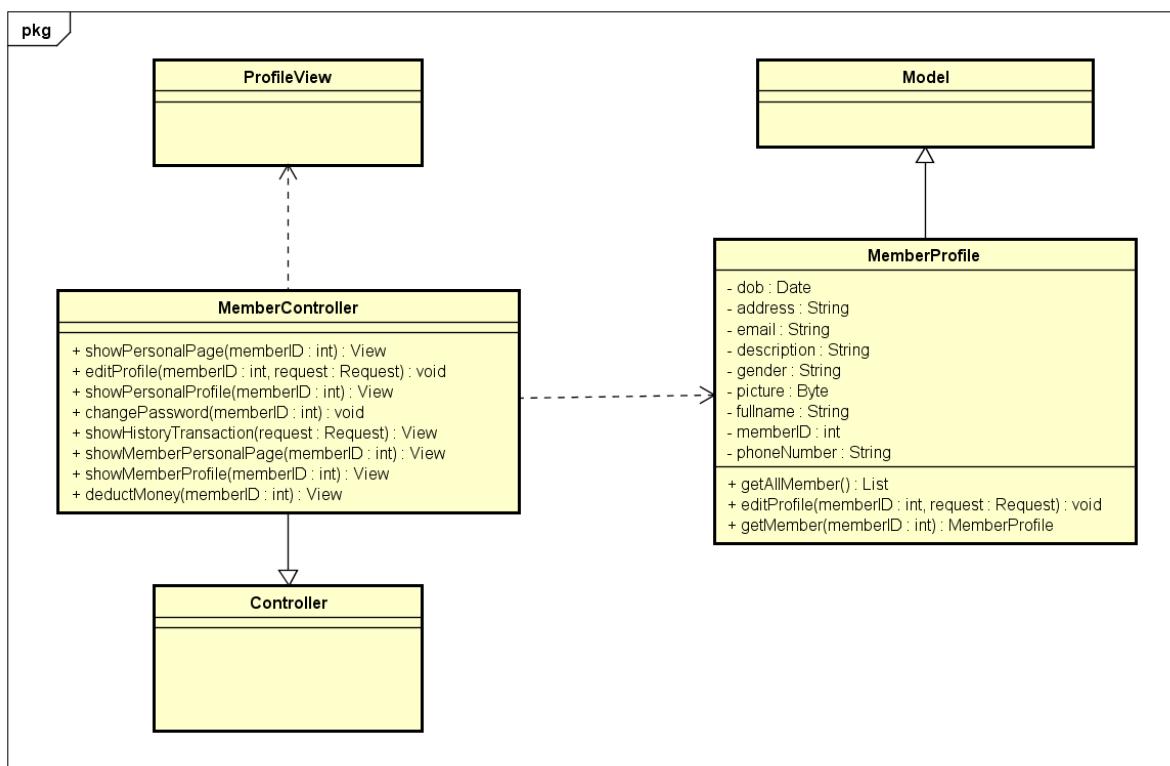
4.7.3.2.3.1 Description

Admin can view Member's Profile information.

4.7.3.2.3.2 Design



4.7.3.2.3.3 Class Diagram

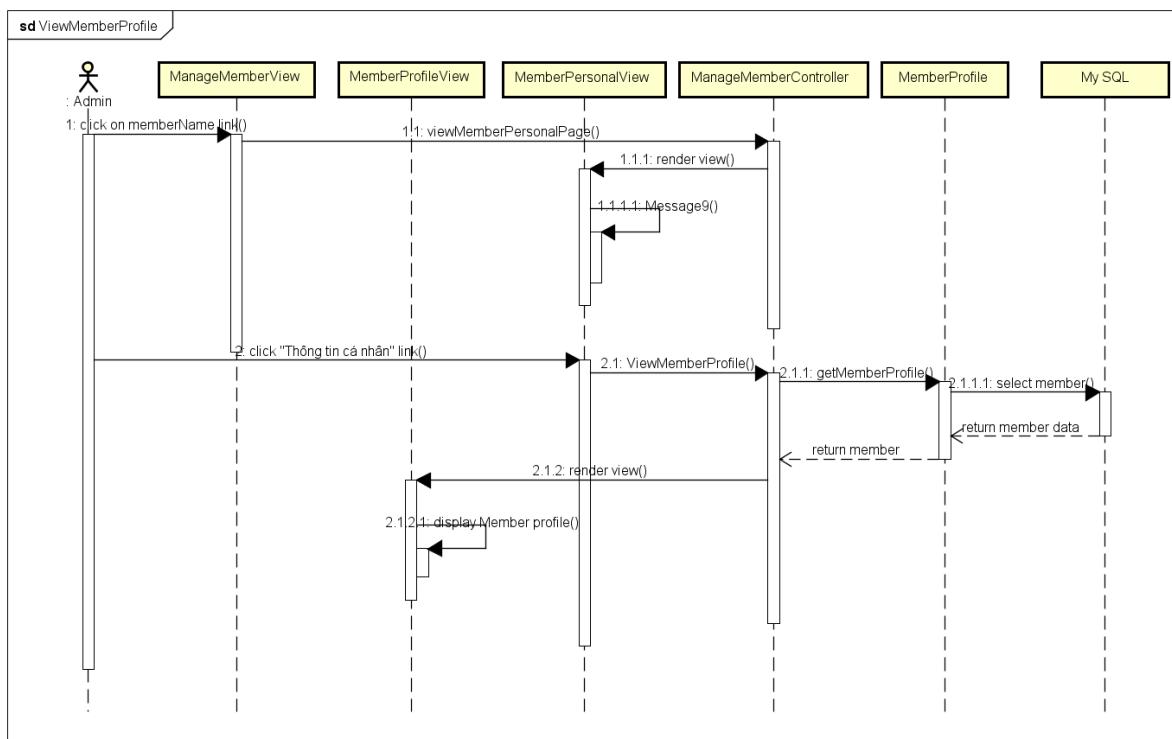


4.7.3.2.3.4 Class Specifications

Class	MemberController
Description	This function allows Admin to view profile of a Member
Base Class	Controller
Source File	MemberController.php

Namespace	App /Http /Controllers/Member/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	showMemberProfile	memberID	View display all member's profile	display profile of members to view

4.7.3.2.3.5 Sequence Diagram



4.7.3.2.4 Ban Member

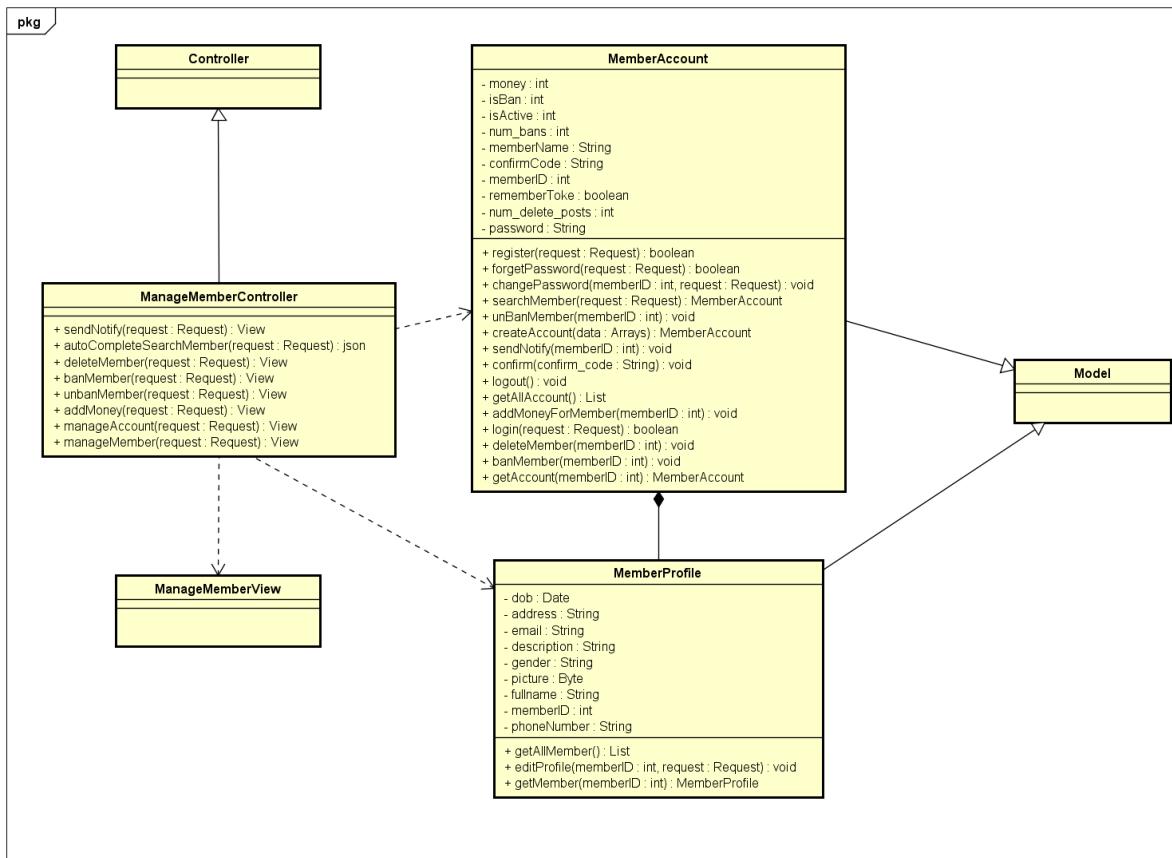
4.7.3.2.4.1 Description

Admin can ban member.

4.7.3.2.4.2 Design

Refer to Design of View Members 4.7.3.2.1.2 View Member

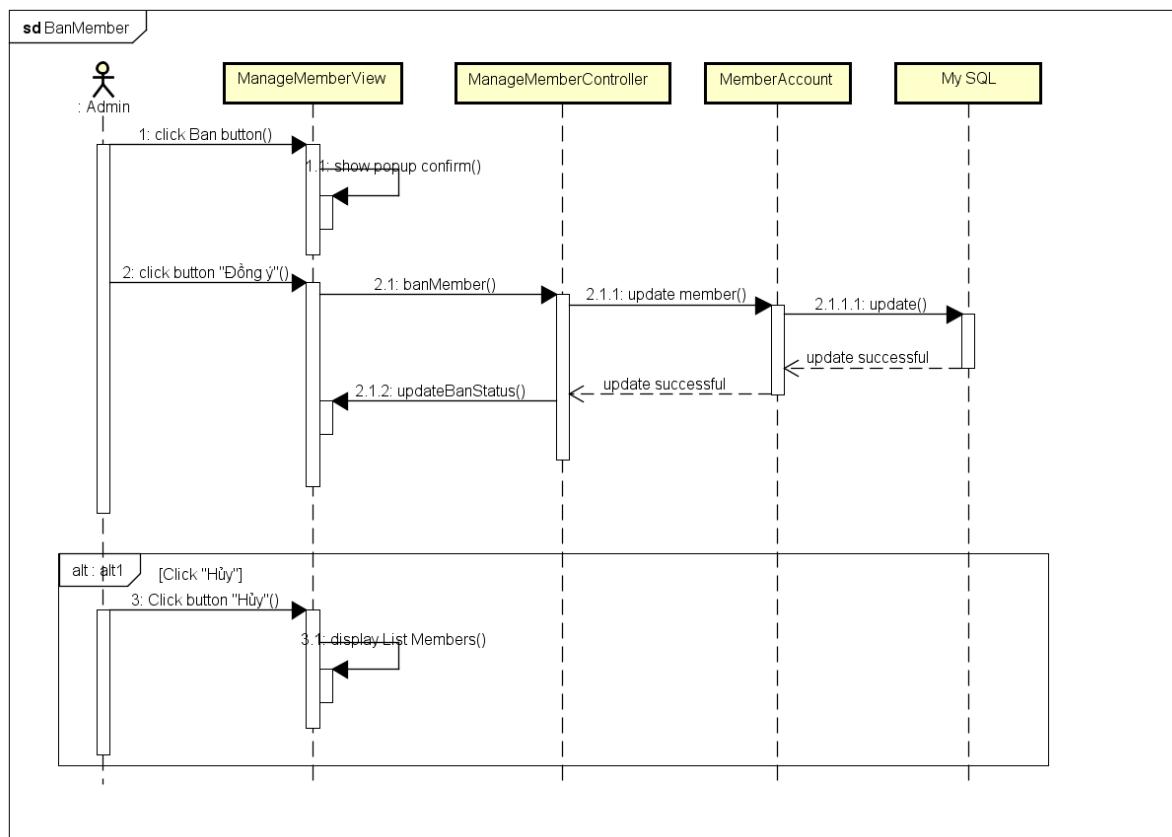
4.7.3.2.4.3 Class Diagram



4.7.3.2.4.4 Class Specifications

Class	ManageMemberController			
Description	Allows Administrator to ban a Member			
Base Class	Controller			
Source File	ManageMemberController.php			
Namespace	App /Http /Controllers/Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description
	banMember	request	Display ban successful in manage member view	Get memberID from Request and ban that member

4.7.3.2.4.5 Sequence Diagram



4.7.3.2.5 Unban Member

4.7.3.2.5.1 Description

This function allow admin unban a member from ban status

4.7.3.2.5.2 Design

Refer to Design of View Members 4.7.3.2.1.2 View Member

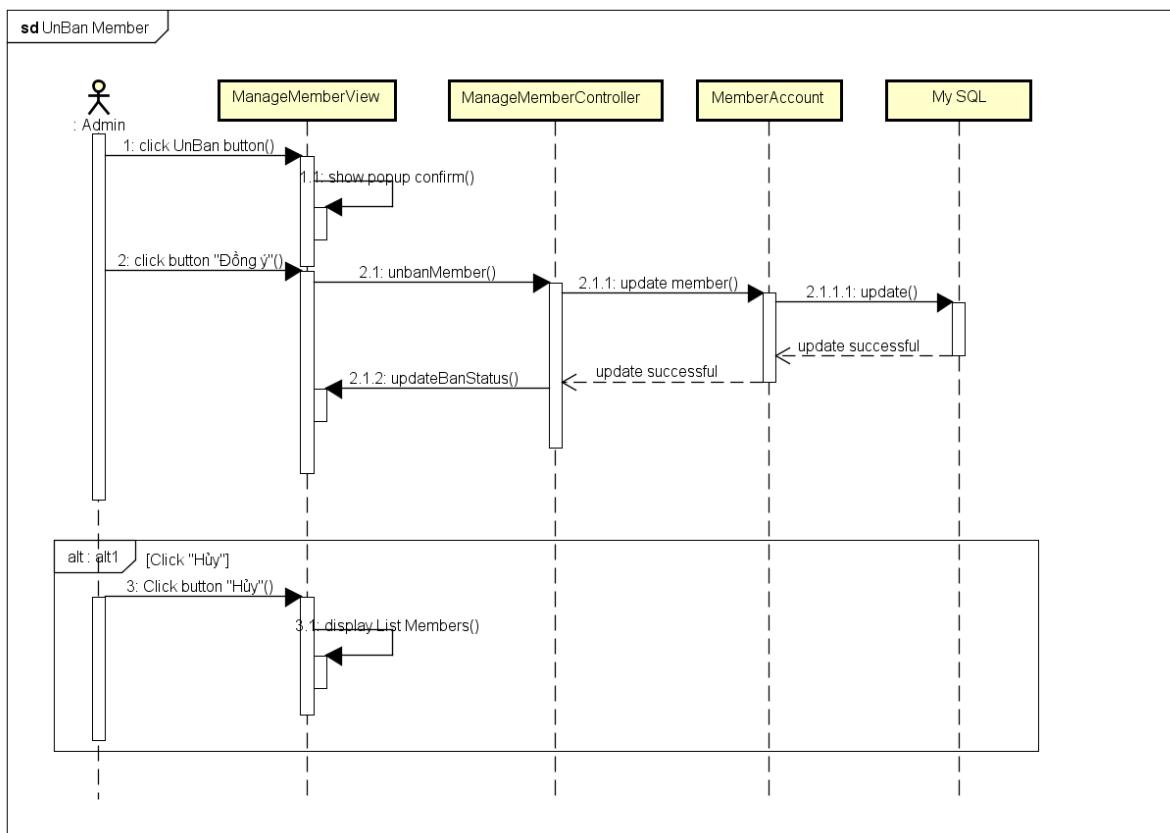
4.7.3.2.5.3 Class Diagram

Refer to class diagram of Ban Member in 4.7.3.2.4 Ban Member

4.7.3.2.5.4 Class Specifications

Class	ManageMemberController			
Description	Allows Administrator to unban a Member			
Base Class	Controller			
Source File	ManageMemberController.php			
Namespace	App\Http\Controllers\Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description
	unbanMember	request	Display unban successful in manage member view	Admin get memberID from Request and unban that member

4.7.3.2.5.5 Sequence Diagram



4.7.3.2.6 Delete Member

4.7.3.2.6.1 Description

Admin can delete Member's account (make account no longer exist on system).

4.7.3.2.6.2 Design

Refer to Design of View Members 4.7.3.2.1.2 View Member

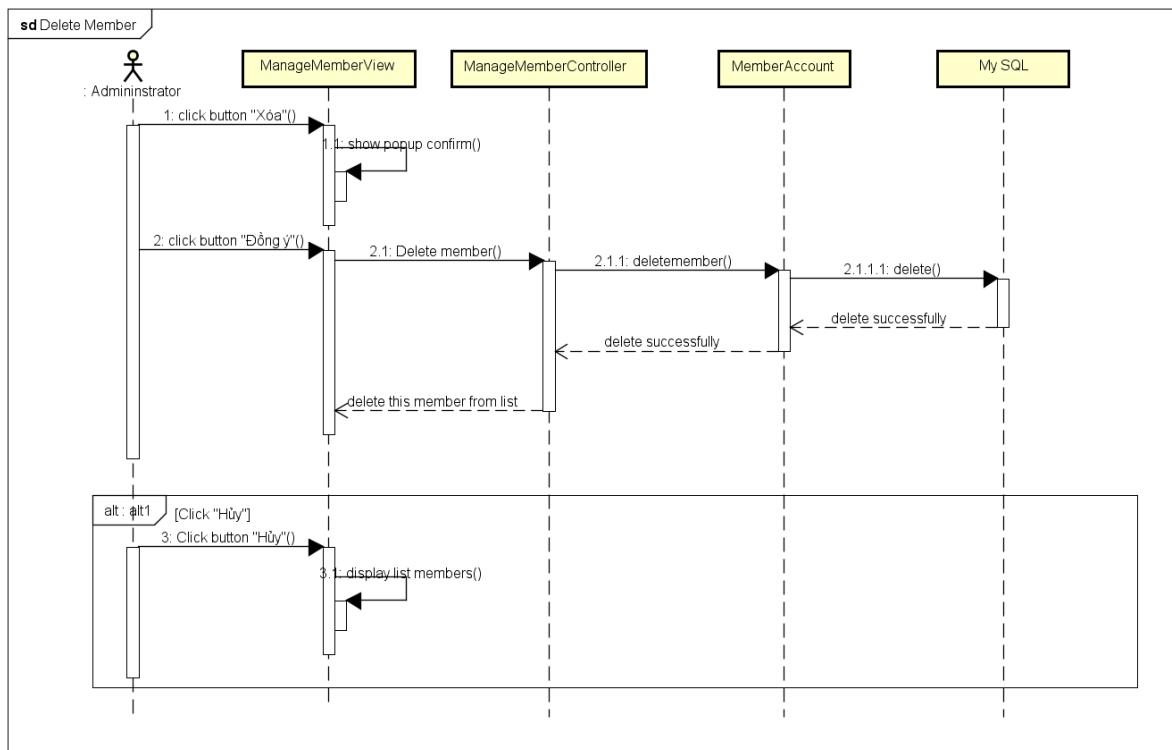
4.7.3.2.6.3 Class Diagram

Refer to class diagram of Ban Member in 4.7.3.2.4 Ban Member

4.7.3.2.6.4 Class Specifications

Class	ManageMemberController			
Description	Allows Administrator to delete a member			
Base Class	Controller			
Source File	ManageMemberController.php			
Namespace	App\Http\controllers\Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description
	deleteMember	request	Delete that member from manage member view	Allow admin get memberID and delete that member

4.7.3.2.6.5 Sequence Diagram

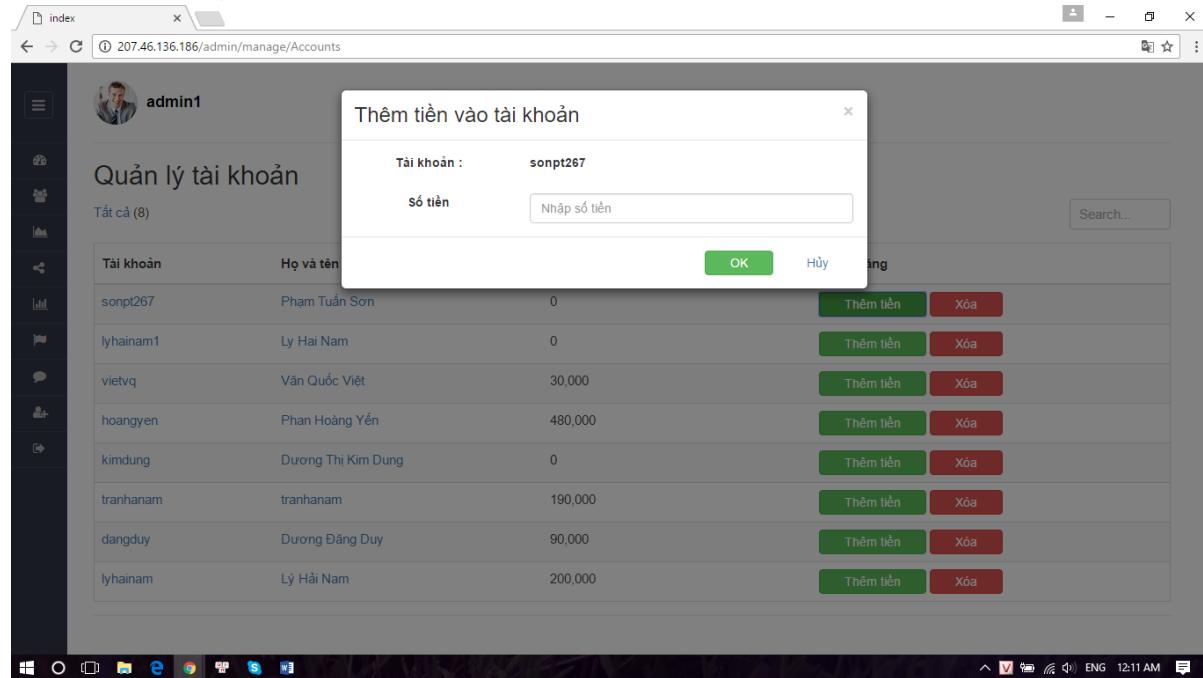


4.7.3.2.7 Add money into Employer's Account

4.7.3.2.7.1 Description

Admin can add money to Member's account.

4.7.3.2.7.2 Design



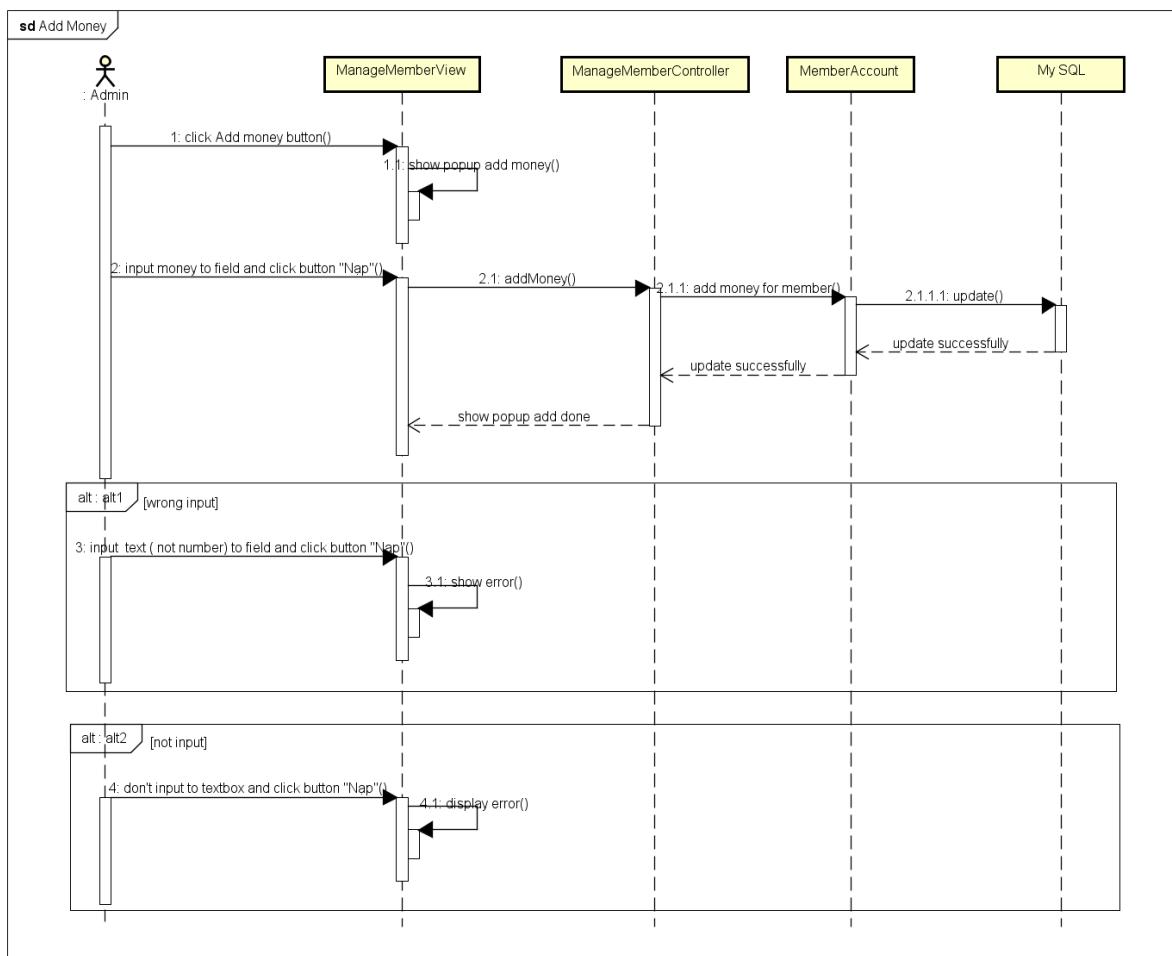
4.7.3.2.7.3 Class Diagram

Refer to class diagram of Ban Member in 4.7.3.2.4 Ban Member

4.7.3.2.7.4 Class Specifications

Class	ManageMemberController		
Description	Allows Administrator to Add money into Employer's financial account		
Base Class	Controller		
Source File	ManageMemberController.php		
Namespace	App\Http\Controllers\Admin		
Attributes	N/A		
Methods	Name	Input	Description
	addMoney	request	Add money to member finalcial account Admin get memberID and add money to that Employer's financial account

4.7.3.2.7.5 Sequence Diagram



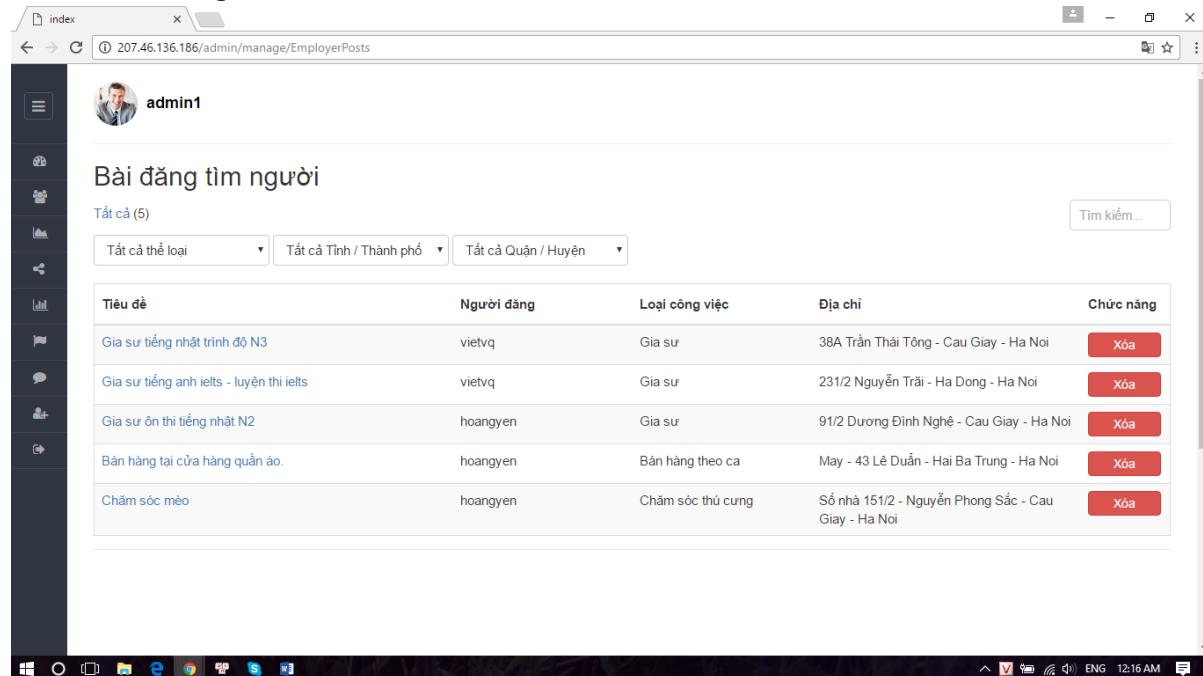
4.7.3.3 Manage Posts

4.7.3.3.1 View Employer's Posts

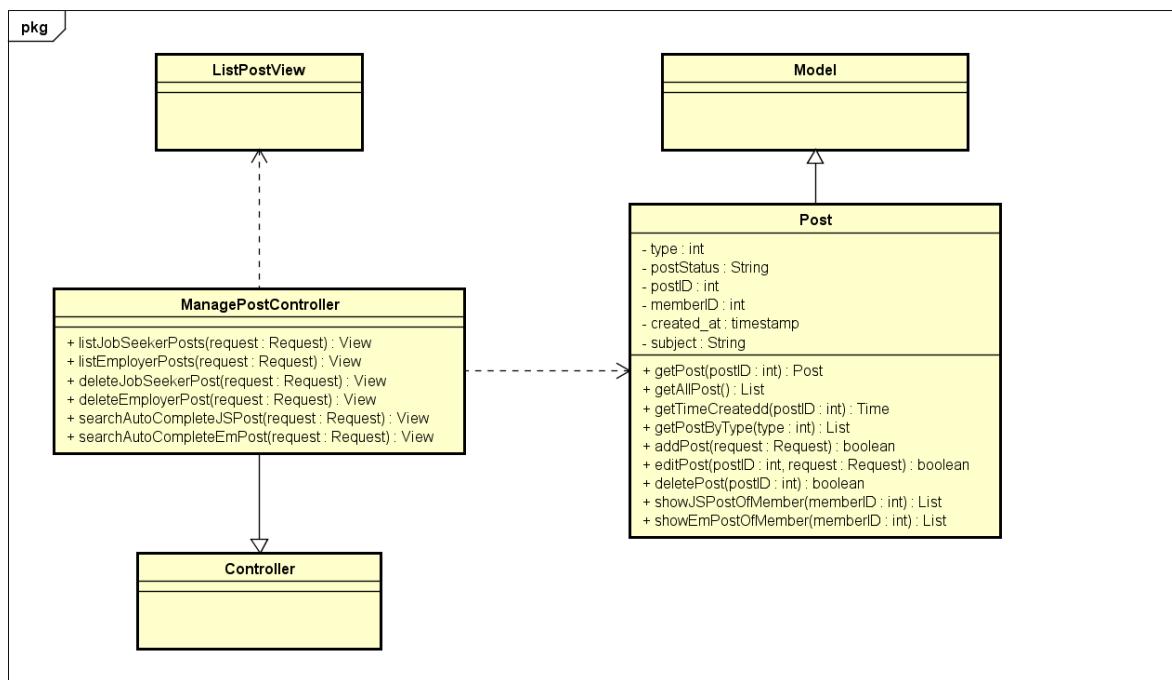
4.7.3.3.1.1 Description

This function allow admin view all Employer's posts of a Member.

4.7.3.3.1.2 Design



4.7.3.3.1.3 Class Diagram

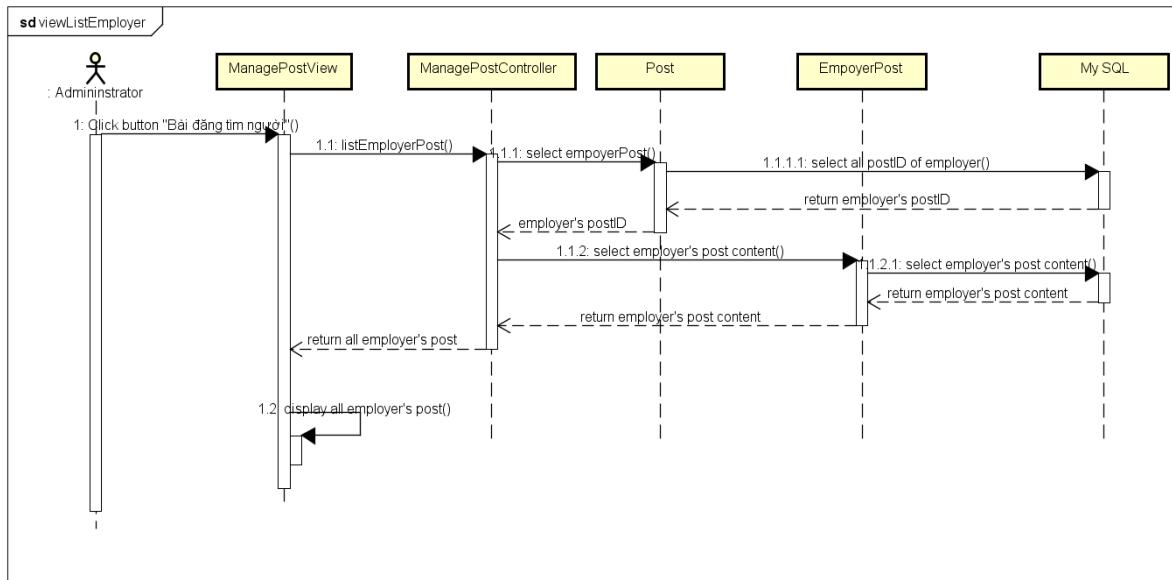


4.7.3.3.1.4 Class Specifications

Class	ManagePostController
Description	Allows Administrator to view a list of all Posts of Employers.
Base Class	Controller
Source File	ManagePostController.php
Namespace	App/Http/controllers/Admin/

Attributes	N/A			
Methods	Name	Input	Output	Description
	listEmployerPosts	request	List post of all employer	Admin view list of all employer post

4.7.3.3.1.5 Sequence Diagram



4.7.3.3.2 View Job Seeker's Posts

4.7.3.3.2.1 Description

This function allow admin to view all Job Seeker's posts of a member.

4.7.3.3.2.2 Design

Tiêu đề	Người đăng	Loại công việc	Địa chỉ	Chức năng
Dọn dẹp nhà cửa	hoangyen	Nội trợ	Hai Ba Trung - Ha Noi	Xóa
Bưng bê/ bồi bàn cho các quán cafe hoặc quán ăn	vietvq	Phục vụ quán ăn	Quan Go Vap - TP Ho Chi Minh	Xóa
Chăm sóc thú cưng	lyhainam	Chăm sóc thú cưng.Bán hàng theo ca	Cau Giay - Ha Noi	Xóa
Chăm sóc trẻ em	dangduy	Chăm sóc trẻ em	Ba Dinh - Ha Noi	Xóa
Tôi cần một công việc như nội trợ hoặc chăm sóc trẻ em	dangduy	Chăm sóc trẻ em	Quan 1 - TP Ho Chi Minh	Xóa
Tôi cần một công việc như nội trợ hoặc chăm sóc trẻ em	dangduy	Chăm sóc trẻ em,Nấu ăn	Ha Dong - Ha Noi	Xóa

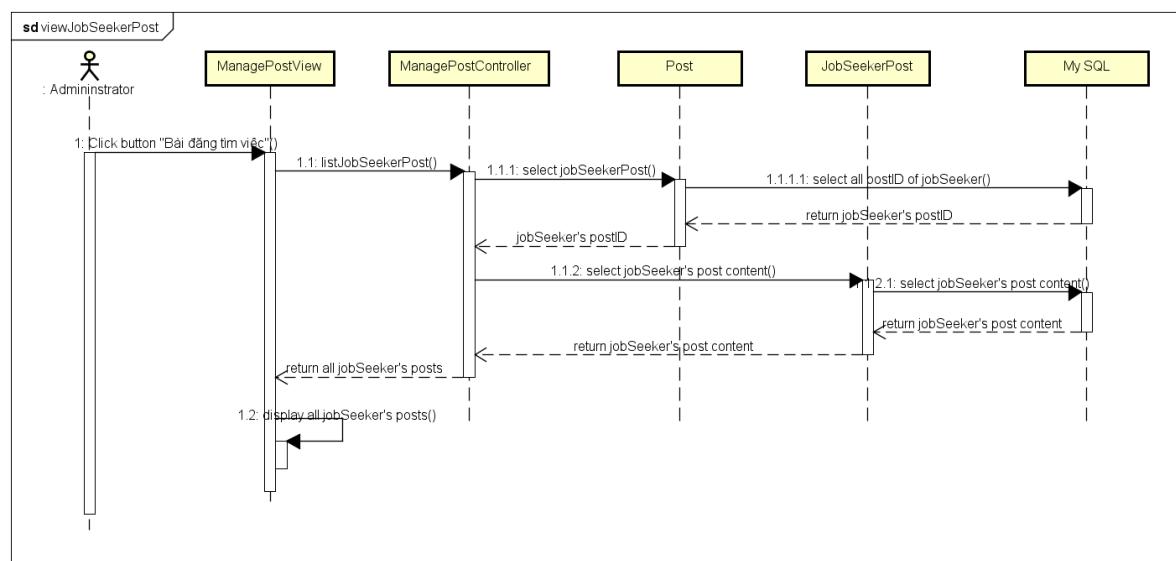
4.7.3.3.2.3 Class Diagram

Refer to class diagram of View Employer's Post in 4.7.3.3.1View Member

4.7.3.3.2.4 Class Specifications

Class	ManagePostController		
Description	Allows Administrator to view a list of all Posts of Employers.		
Base Class	Controller		
Source File	ManagePostController.php		
Namespace	App/Http/controllers/Admin		
Attributes	N/A		
Methods	Name	Input	Output
	listJobSeekerPosts	request	List post of all job seeker
			Admin view list of all job seeker post

4.7.3.3.2.5 Sequence Diagram



4.7.3.3 Search Employer's Posts

4.7.3.3.3.1 Description

Admin can search for specific Employer posts by post's subject or description.

4.7.3.3.3.2 Design

Bài đăng tìm người

Tất cả (5)

Tiêu đề	Người đăng	Loại công việc	Địa chỉ	Chức năng
Gia sư tiếng nhật trình độ N3	vietvq	Gia sư	38A Trần Thái Tông - Cau Giay - Ha Noi	Xóa
Gia sư tiếng anh ielts - luyện thi ielts	vietvq	Gia sư	231/2 Nguyễn Trãi - Ha Dong - Ha Noi	Xóa
Gia sư ôn thi tiếng nhật N2	hoangyen	Gia sư	91/2 Dương Đình Nghệ - Cau Giay - Ha Noi	Xóa

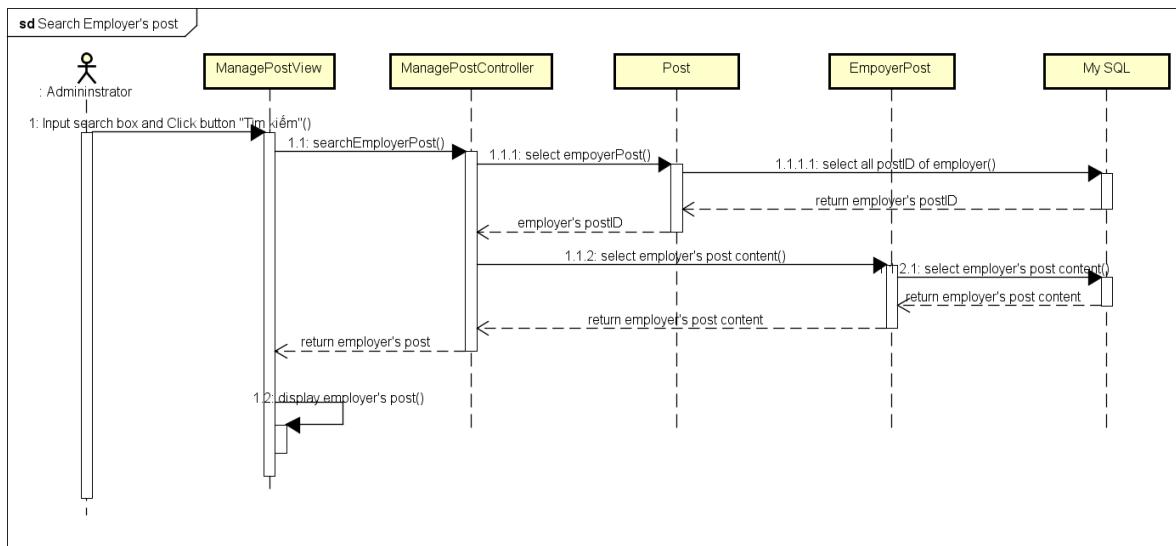
4.7.3.3.3.3 Class Diagram

Refer to class diagram of View Employer's Post in 4.7.3.3.1 View Member

4.7.3.3.3.4 Class Specifications

Class	ManagePostController			
Description	Allows Administrator to search Employer's Post			
Base Class	Controller			
Source File	ManagePostController.php			
Namespace	App\Http\Controllers\Admin			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchAutoCompleteEmPost	request	View display employer's post	Find and return Employer's Post

4.7.3.3.3.5 Sequence Diagram

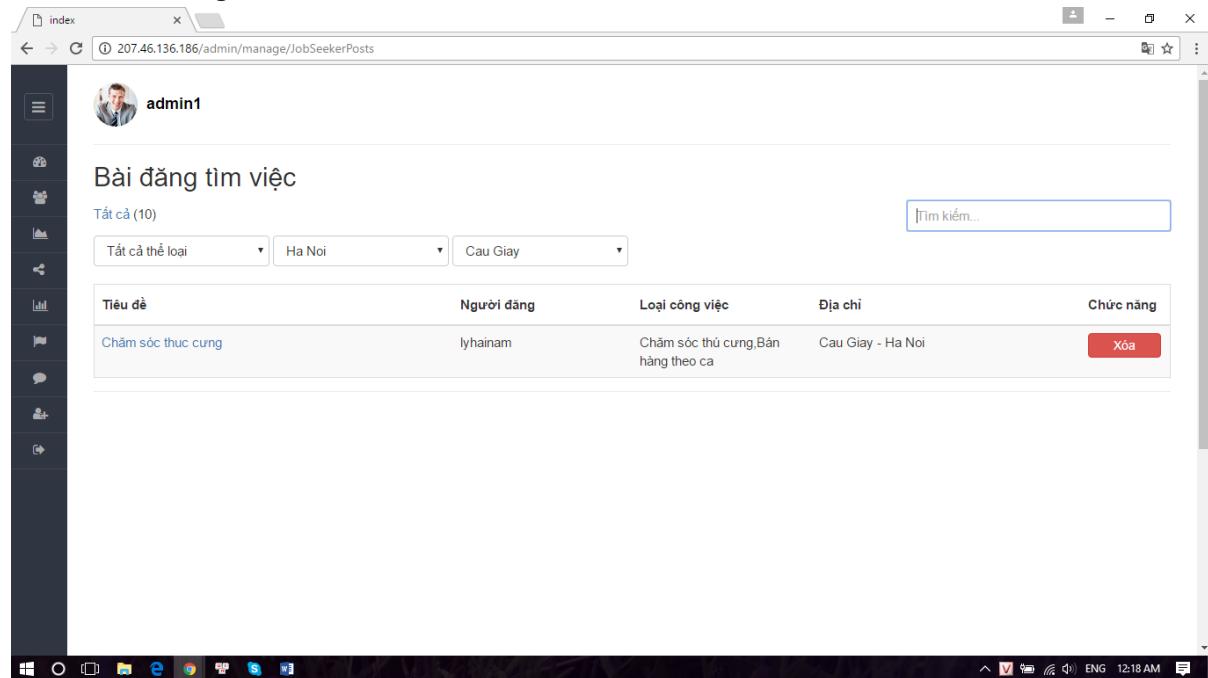


4.7.3.3.4 Search Job Seeker's Posts

4.7.3.3.4.1 Description

Admin can search for specific Job Seeker posts by post's subject or description.

4.7.3.3.4.2 Design



4.7.3.3.4.3 Class Diagram

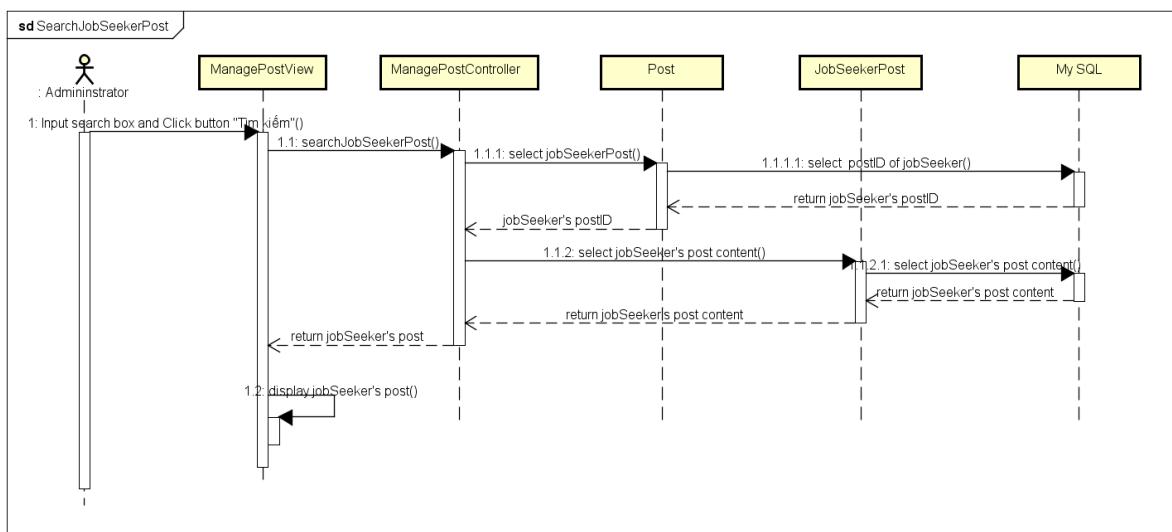
Refer to class diagram of View Employer's Post in 4.7.3.3.1 View Member

4.7.3.3.4.4 Class Specifications

Class	ManagePostController
Description	Allows Administrator to search JobSeeker's Post

Base Class	Controller			
Source File	ManagePostController.php			
Namespace	App/Http/controllers/Admin/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	searchAutoCompleteJSPost	request	View display jobSeeker's post	Find and return JobSeeker's Post

4.7.3.3.4.5 Sequence Diagram



4.7.3.3.5 Delete Posts

4.7.3.3.5.1 Description

Admin delete Job Seeker's post or Employer's post by Post ID.

4.7.3.3.5.2 Design

Refer to Design of View Employer's Post 4.7.3.3.1.2 View Employer's Post

Refer to Design of View Employer's Post 4.7.3.3.2.2 View Employer's Post

4.7.3.3.5.3 Class Diagram

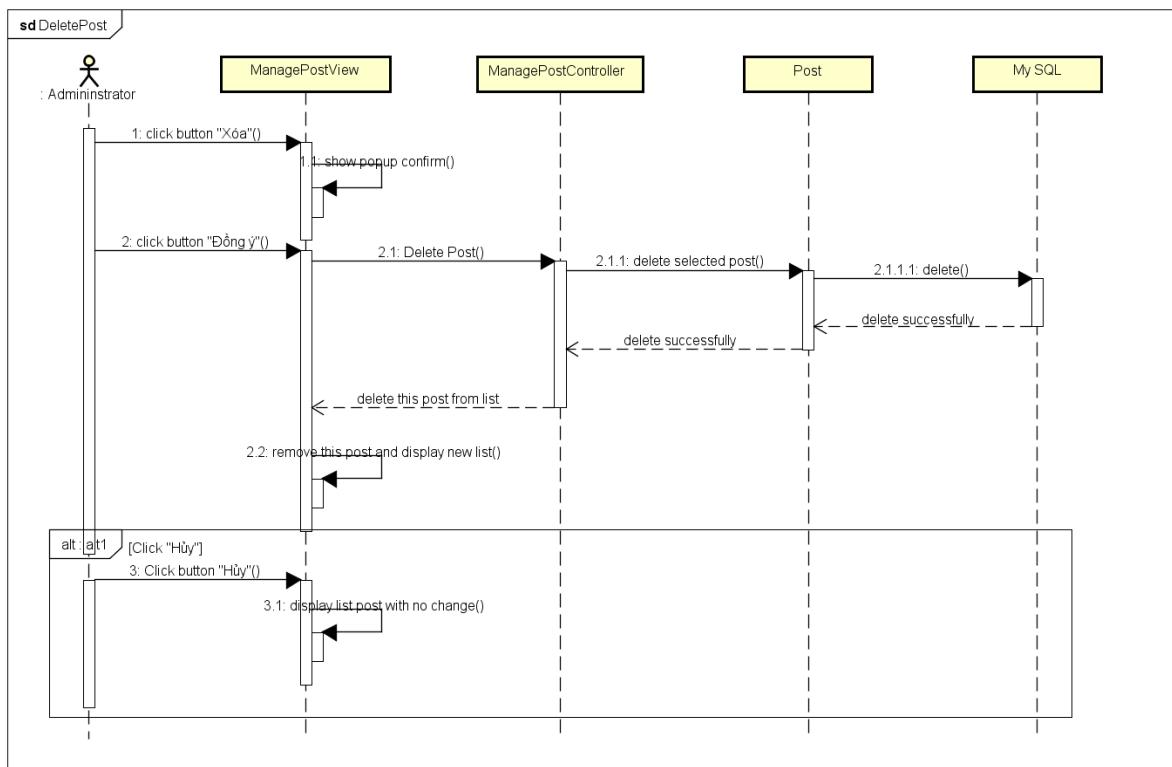
Refer to class diagram of View Employer's Post in 4.7.3.3.1 View Member

4.7.3.3.5.4 Class Specifications

Class	ManagePostController
Description	Allows Administrator to delete Post
Base Class	Controller
Source File	ManagePostController.php
Namespace	App/Http/controllers/Admin
Attributes	N/A

Methods	Name	Input	Output	Description
	deleteJobSeekerPost	request	View update manage jobSeeker's post	Get postID and delete that post
	deleteEmployerPost	request	View update manage employer's post	Get postID and delete that post

4.7.3.3.5.5 Sequence Diagram

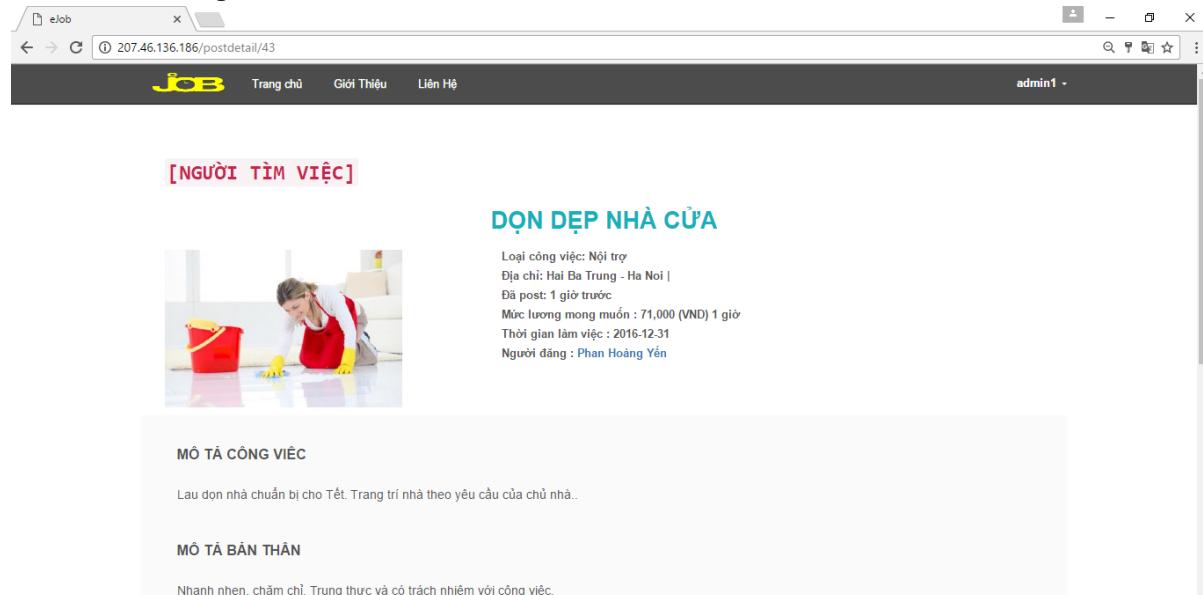


4.7.3.3.6 View detail Posts

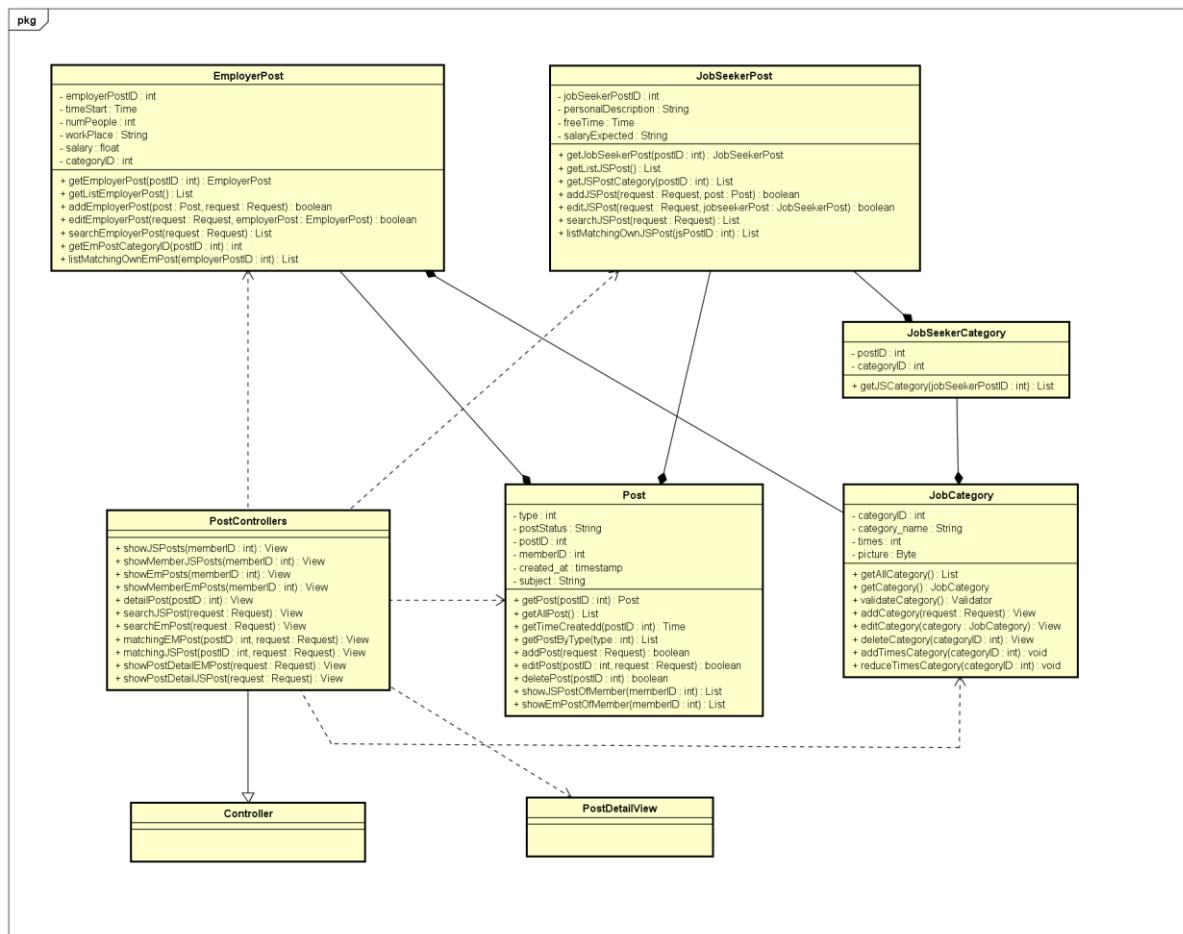
4.7.3.3.6.1 Description

Admin view detail content of a post.

4.7.3.3.6.2 Design



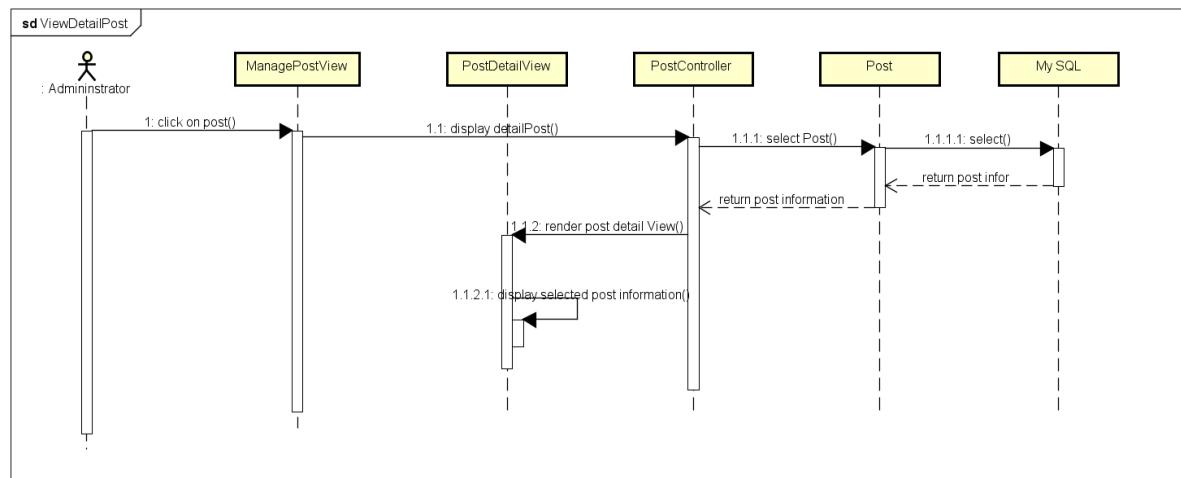
4.7.3.3.6.3 Class Diagram



4.7.3.3.6.4 Class Specifications

Class	PostController			
Description	Allows Administrator to view detail of a Post			
Base Class	Controller			
Source File	PostController.php			
Namespace	App\Http\controllers\post			
Attributes	N/A			
Methods	Name	Input	Output	Description
	detailPost	postID	View display detail content of a post	Information of this Post is displayed.

4.7.3.3.6.5 Sequence Diagram



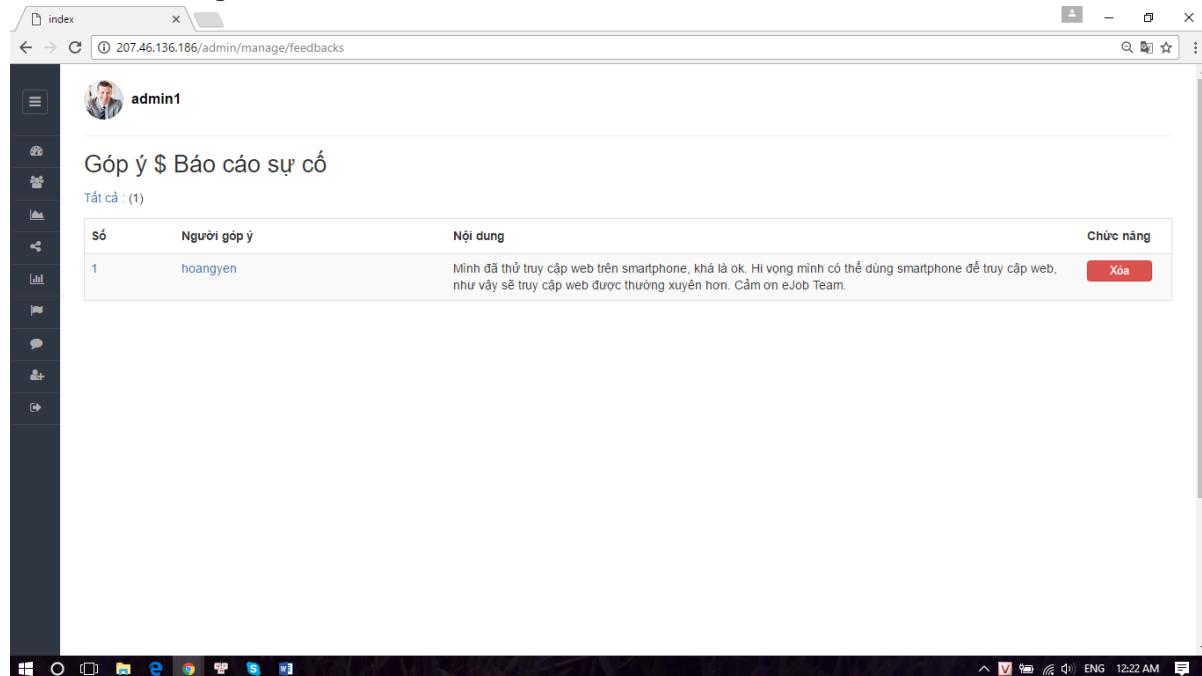
4.7.3.4 Manage Feedback from member

4.7.3.4.1 View feedback about website from member

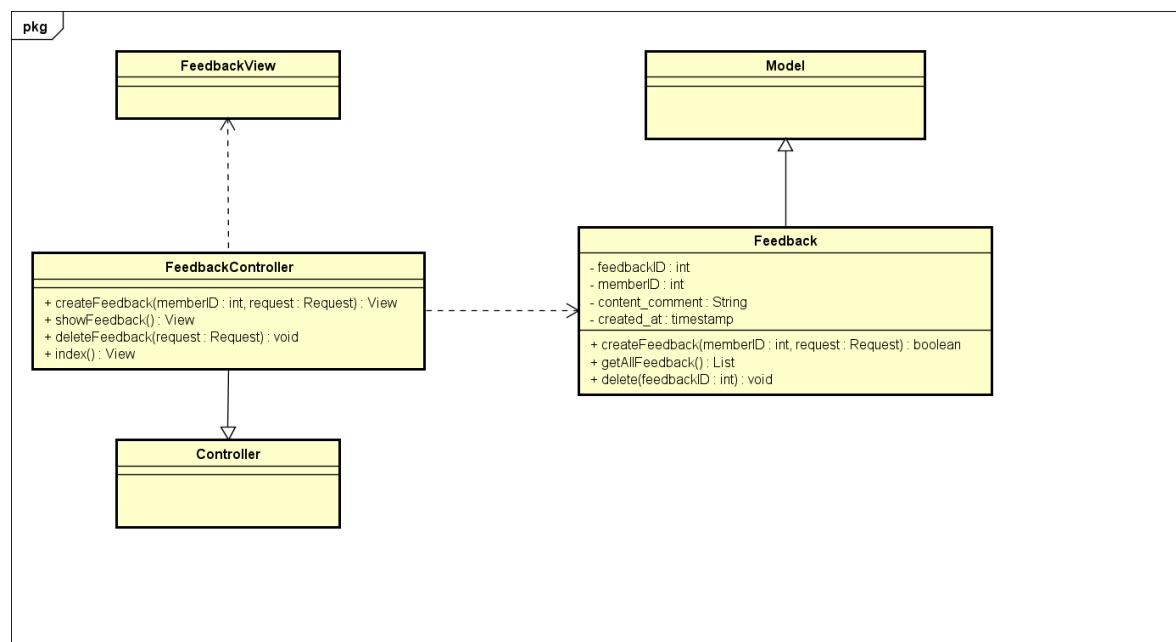
4.7.3.4.1.1 Description

Admin view all feedbacks created by Members.

4.7.3.4.1.2 Design



4.7.3.4.1.3 Class Diagram

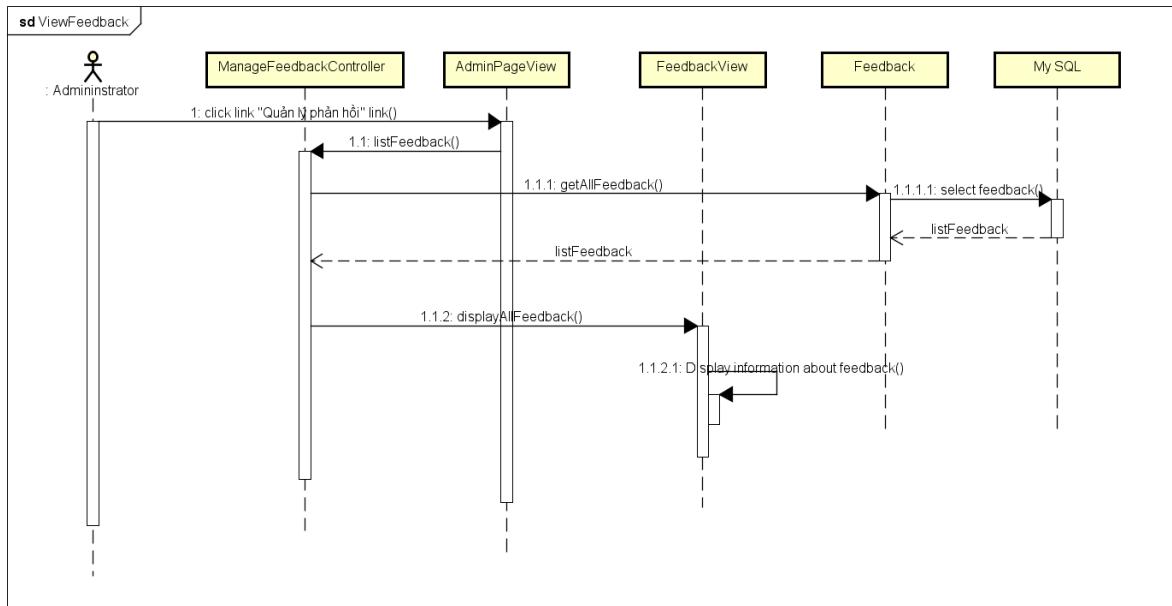


4.7.3.4.1.4 Class Specifications

Class	FeedbackController
Description	Allows Admin view feedback about issues of website
Base Class	Controller
Source File	FeedbackController.php
Namespace	App/Http/controllers/
Attributes	N/A

Methods	Name	Input	Output	Description
	showFeedback	N/A	View all feedback from member	Administrator can view all feedback about issues from user

4.7.3.4.1.5 Sequence Diagram



4.7.3.4.2 Delete feedback about issues of website

4.7.3.4.2.1 Description

Admin delete feedback that created by member.

4.7.3.4.2.2 Design

Refer to design of View feedback about website from member 4.7.3.4.1.2

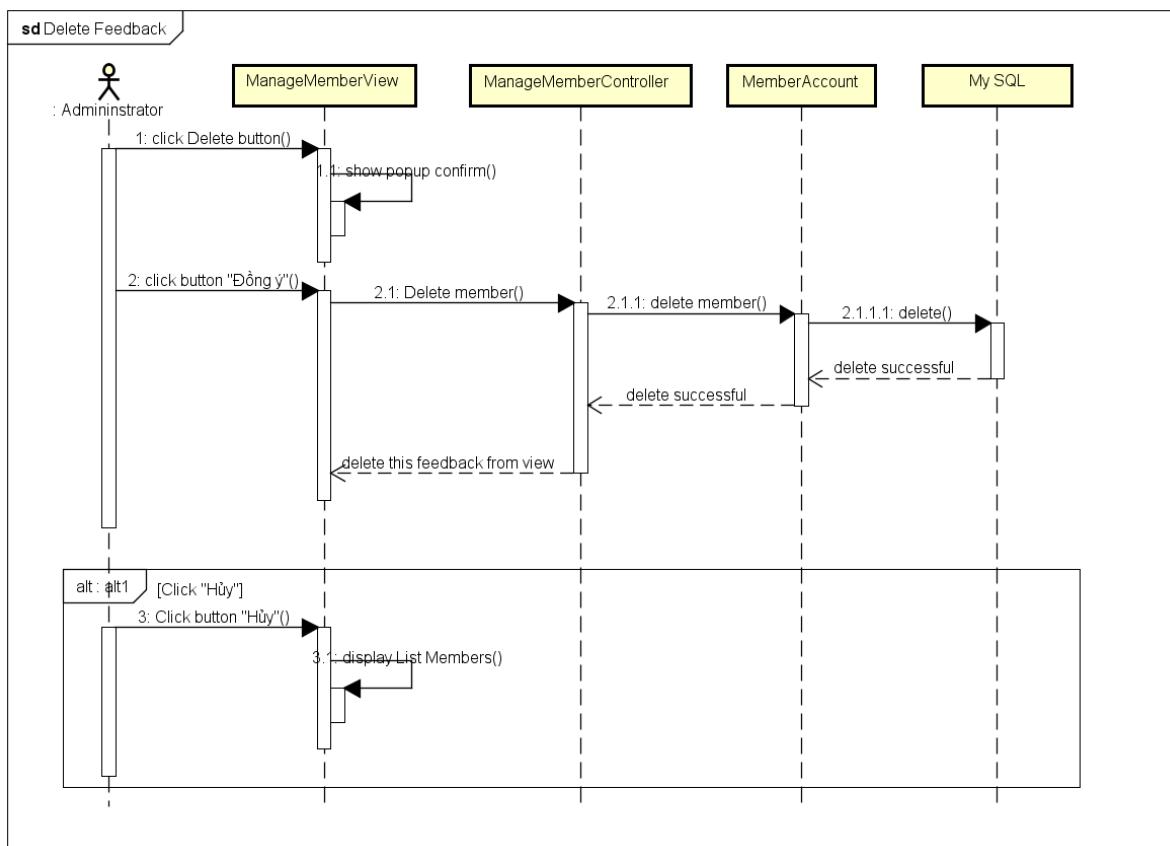
4.7.3.4.2.3 Class Diagram

Refer to class diagram of View Feedback in 4.7.3.4.1 View Feedback

4.7.3.4.2.4 Class Specifications

Class	FeedbackController			
Description	Allows Admin view feedback about issues of website			
Base Class	Controller			
Source File	CommentController.php			
Namespace	App\Http\controllers/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	deleteFeedback	request	Update view manage feedback	Delete feedback that is chosen by admin

4.7.3.4.2.5 Sequence Diagram



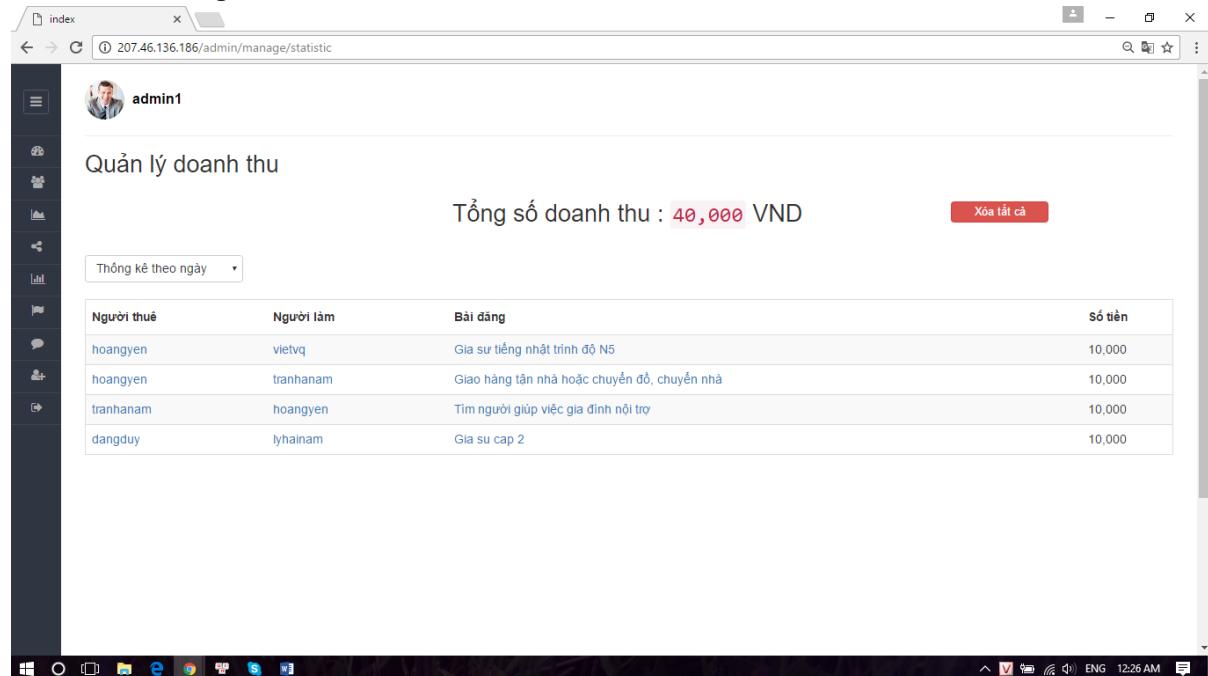
4.7.3.5 Statistic

4.7.3.5.1 Statistic earned money from Members

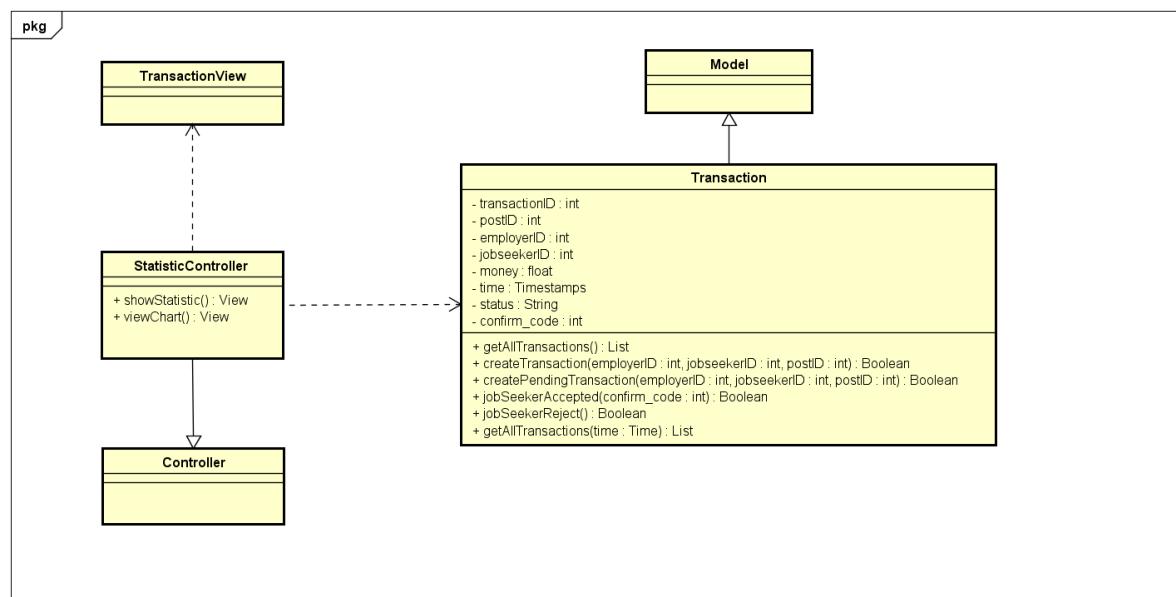
4.7.3.5.1.1 Description

Admin view all transaction and money that website earned (Money that Member must be paid in each transaction).

4.7.3.5.1.2 Design



4.7.3.5.1.3 Class Diagram

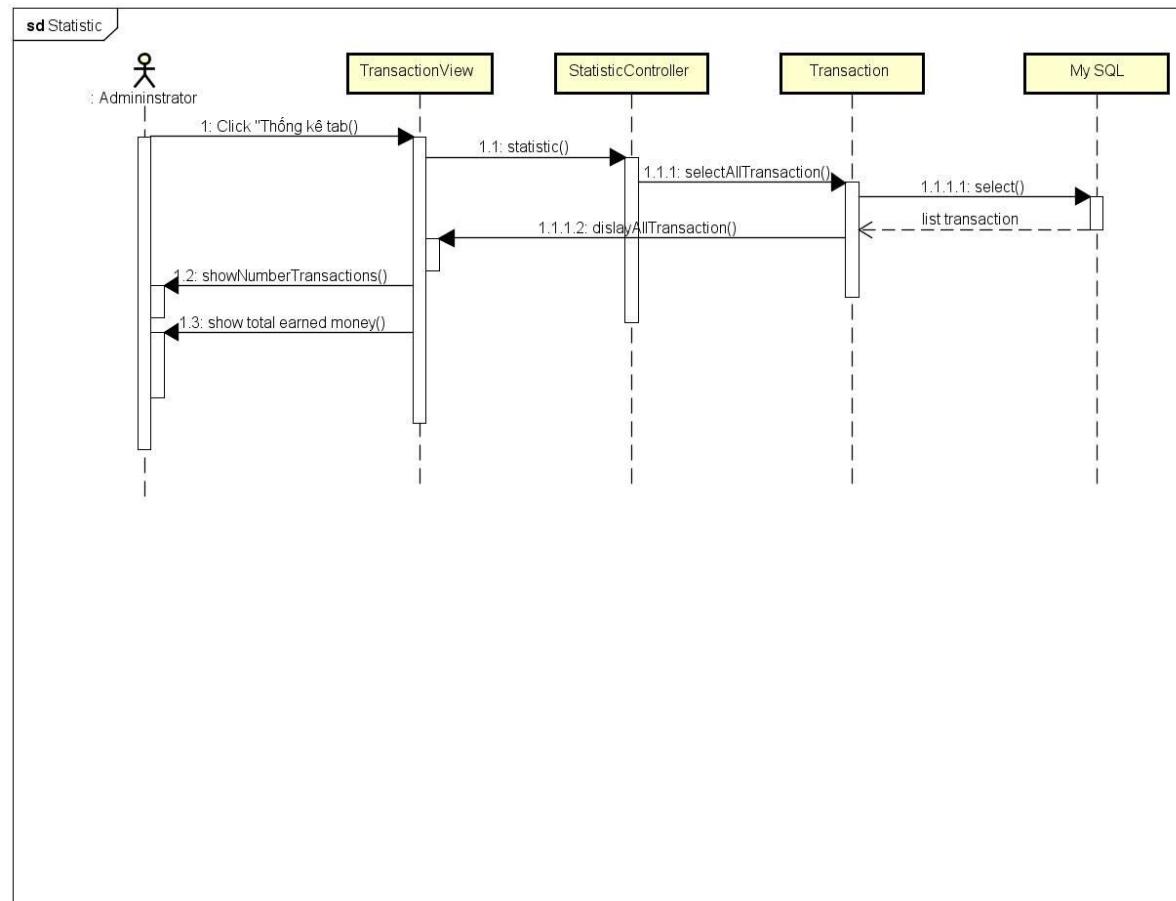


4.7.3.5.1.4 Class Specifications

Class	StatisticController			
Description	Allows Admin statistic earned money from Members			
Base Class	Controller			
Source File	StatisticController.php			
Namespace	App\Http\controllers/			
Attributes	N/A			
Methods	Name	Input	Output	Description

	showStatistic	N/A	View display all transaction	Display all transaction and total amount of money that website earned
--	---------------	-----	------------------------------	---

4.7.3.5.1.5 Sequence Diagram



4.7.3.5.2 View Chart

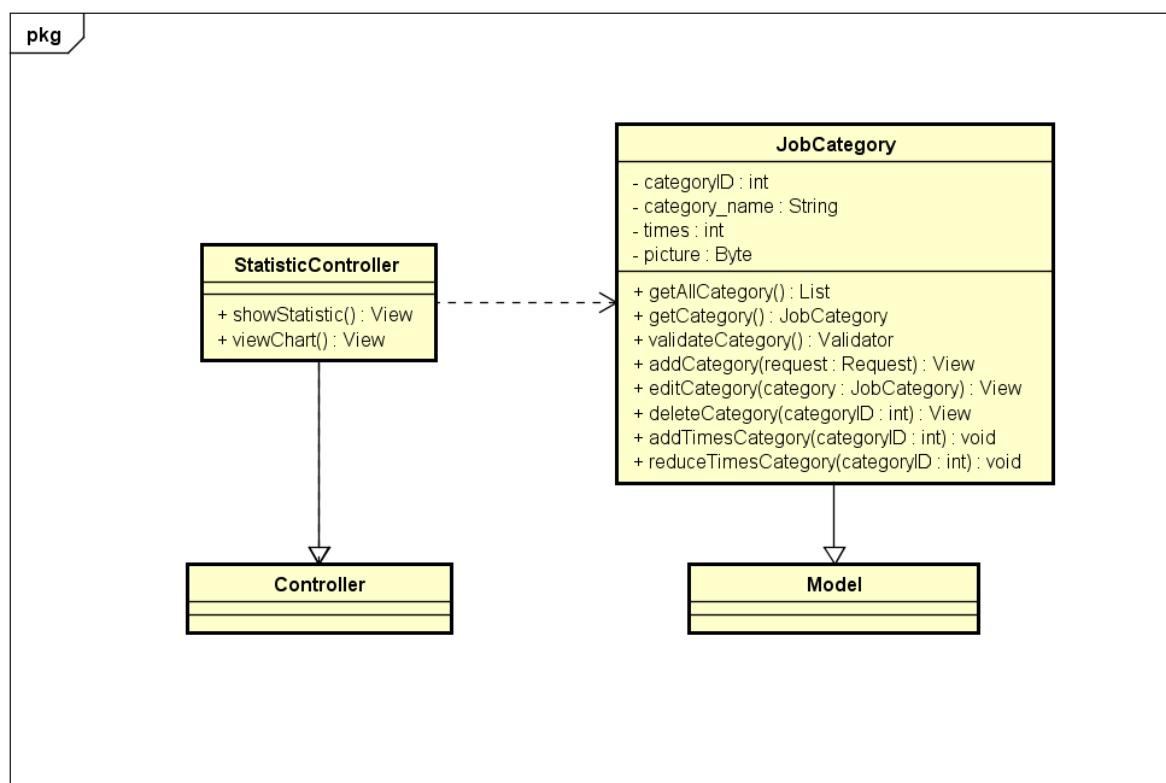
4.7.3.5.2.1 Description

Admin can view chart (that created by system) of number of posts in each job category.

4.7.3.5.2.2 Design



4.7.3.5.2.3 Class Diagram

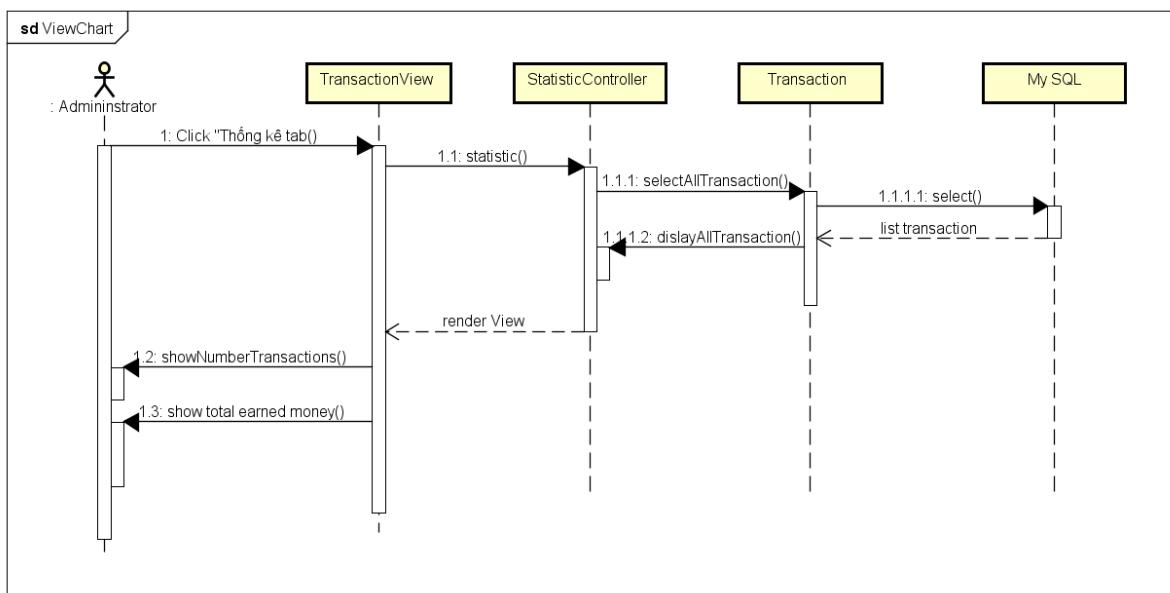


4.7.3.5.2.4 Class Specifications

Class	StatisticController
Description	Allows Administrator view a chart which show the number of post follow category
Base Class	Controller

Source File	StatisticController.php		
Namespace	App\Http\controllers/		
Attributes	N/A		
Methods	Name	Input	Output
	viewChart	N/A	Display chart
Allows Administrator view a chart which show the number of post follow category			

4.7.3.5.2.5 Sequence Diagram



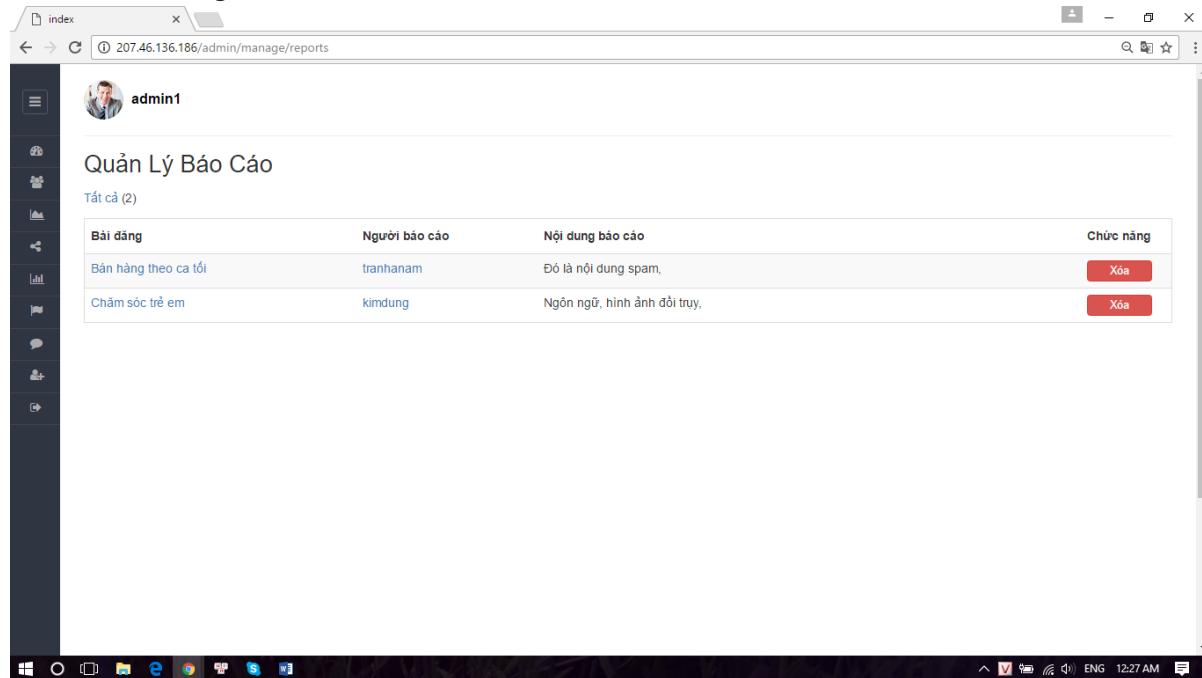
4.7.3.6 Manage Reports

4.7.3.6.1 View Reports about Post

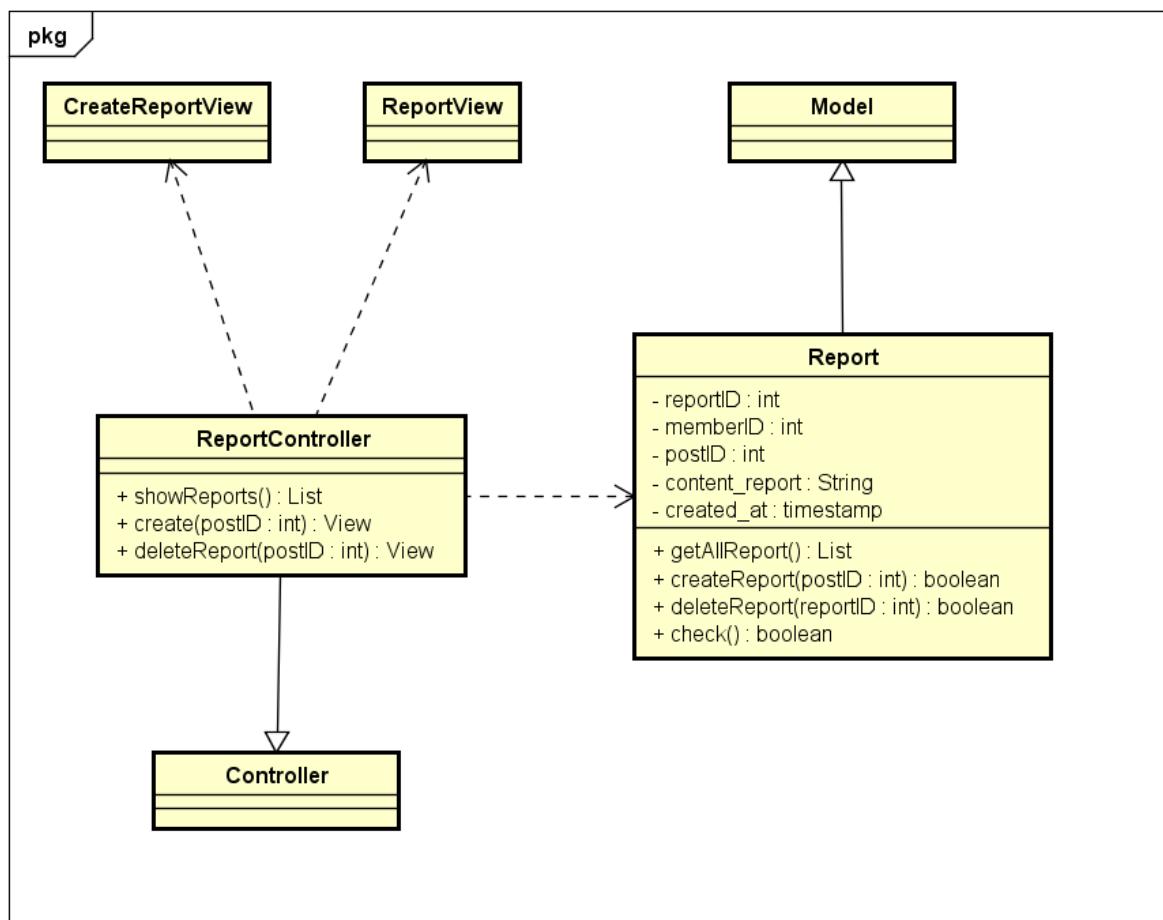
4.7.3.6.1.1 Description

View all reports from members.

4.7.3.6.1.2 Design



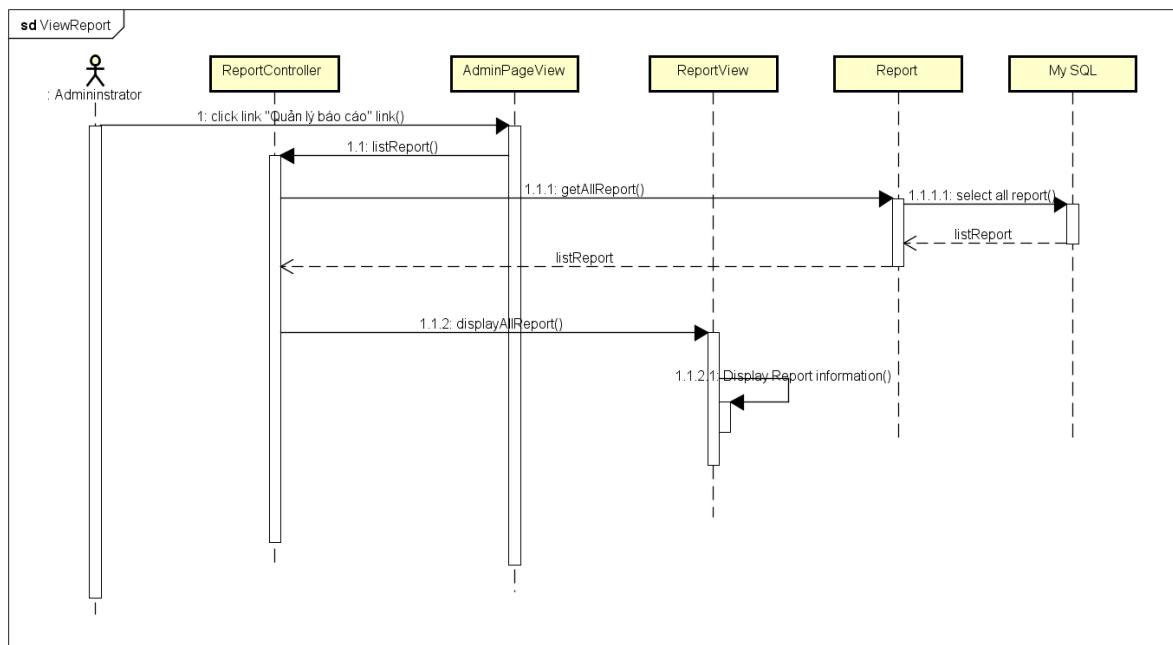
4.7.3.6.1.3 Class Diagram



4.7.3.6.1.4 Class Specifications

Class	ReportController			
Description	Allows Administrator can view list report about some post from member			
Base Class	Controller			
Source File	ReportController.php			
Namespace	App\Http\controllers/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	showReports	N/A	View display list report	Display list report about some post from member

4.7.3.6.1.5 Sequence Diagram



4.7.3.6.2 Delete Reports about Post

4.7.3.6.2.1 Description

Admin delete report.

4.7.3.6.2.2 Design

Refer to design of View Report about Post 4.7.3.6.1.2 View Reports

4.7.3.6.2.3 Class Diagram

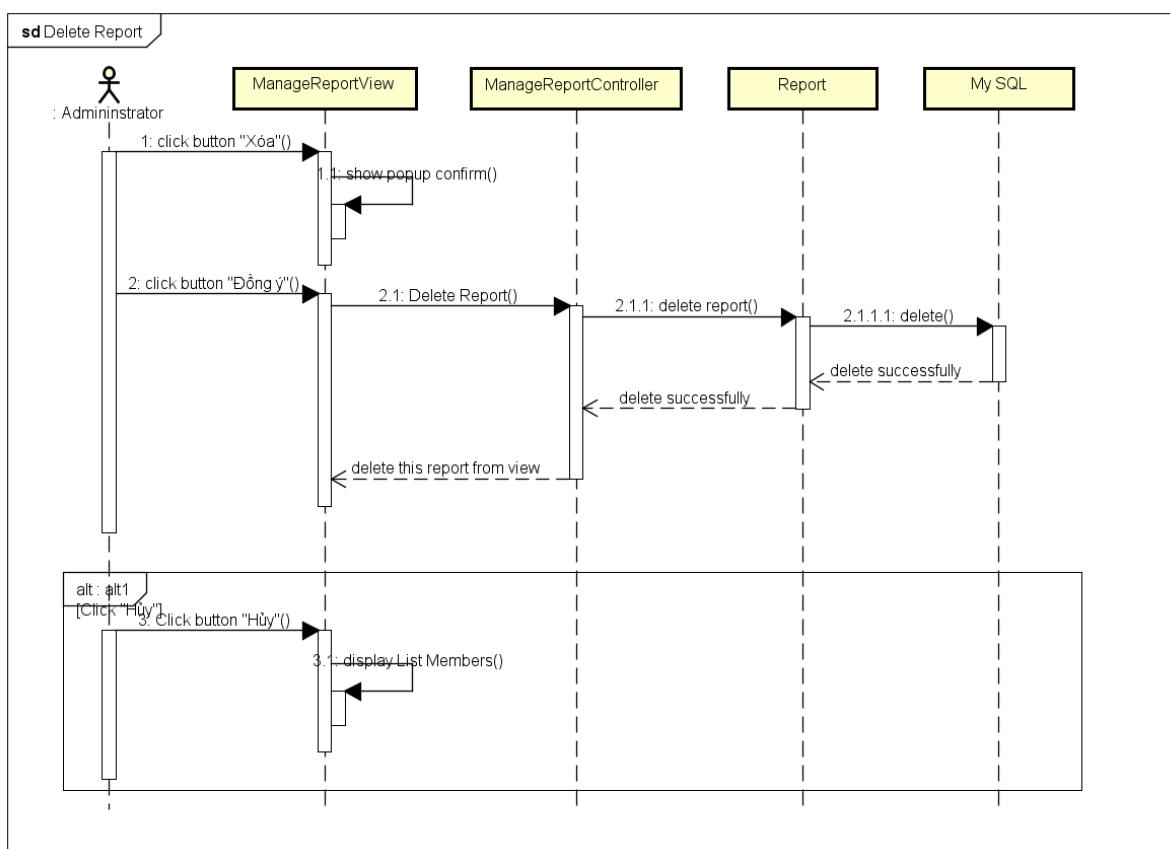
Refer to class diagram of View Reports about Post in 4.7.3.6.1 View Reports about Post

4.7.3.6.2.4 Class Specifications

Class	ReportController
Description	Allows Administrator delete Reports about Post

Base Class	Controller			
Source File	ReportController.php			
Namespace	App/Http/controllers/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	deleteReport	reportID	Delete report successful and update to view	Delete successfully a report from table list of report

4.7.3.6.2.5 Sequence Diagram



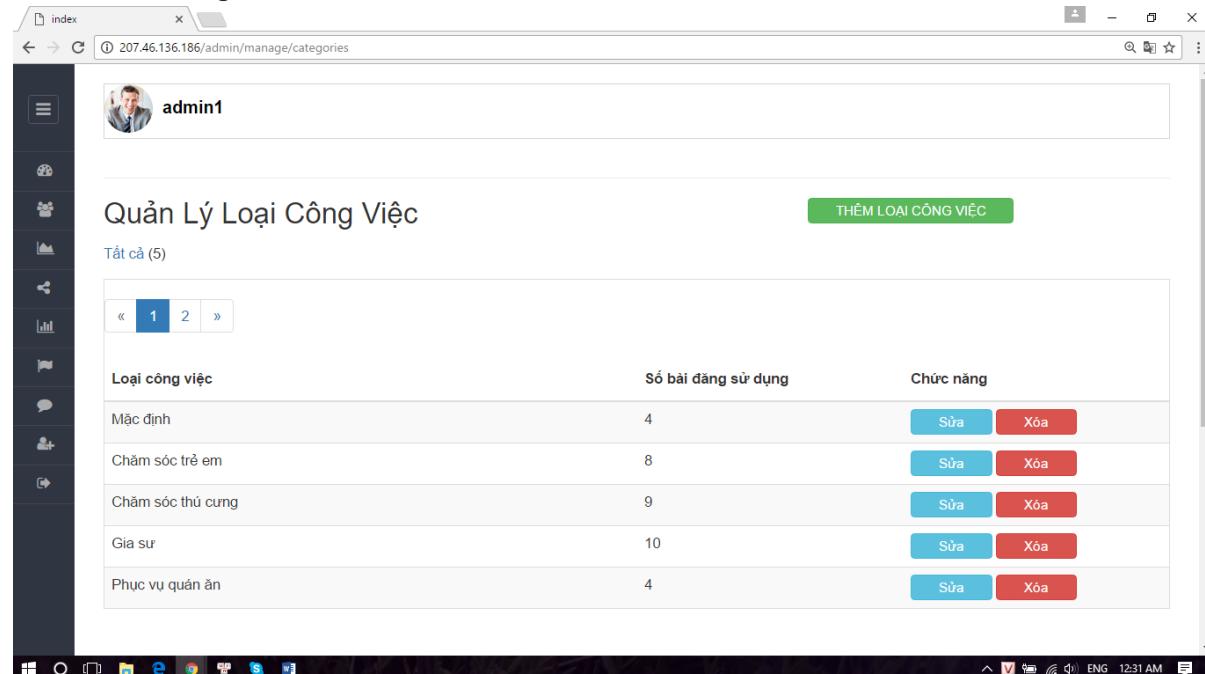
4.7.3.7 Manage Category

4.7.3.7.1 Add Category

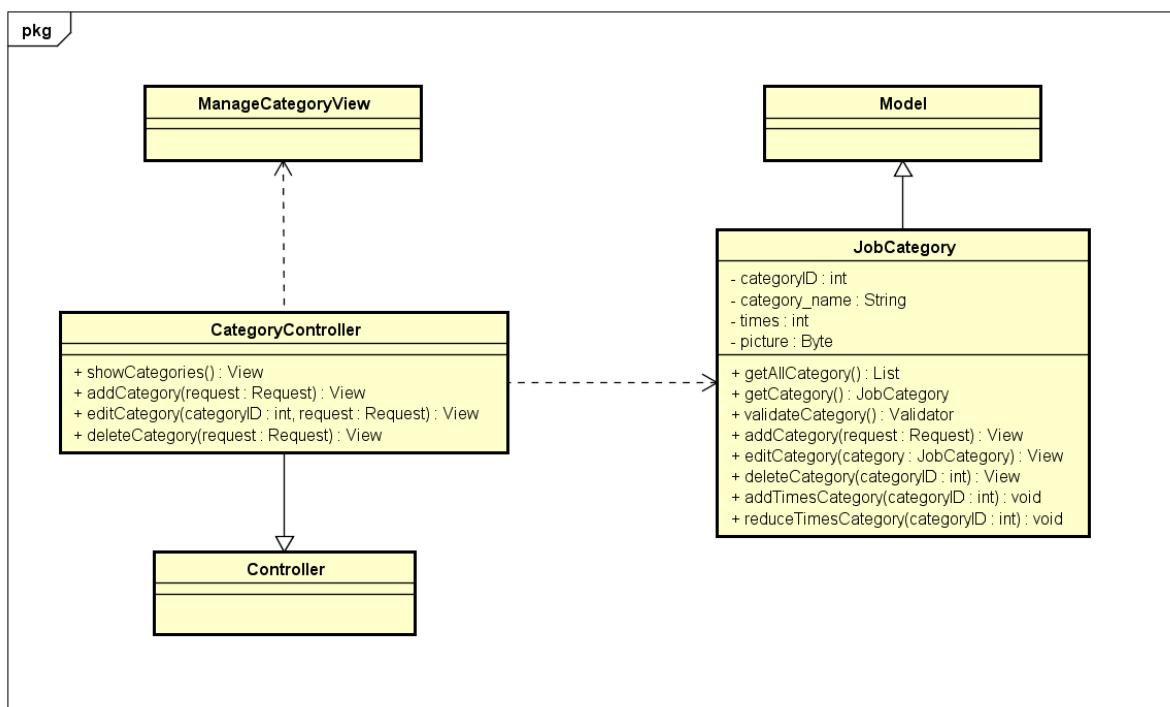
4.7.3.7.1.1 Description

Admin add new job category.

4.7.3.7.1.2 Design



4.7.3.7.1.3 Class Diagram

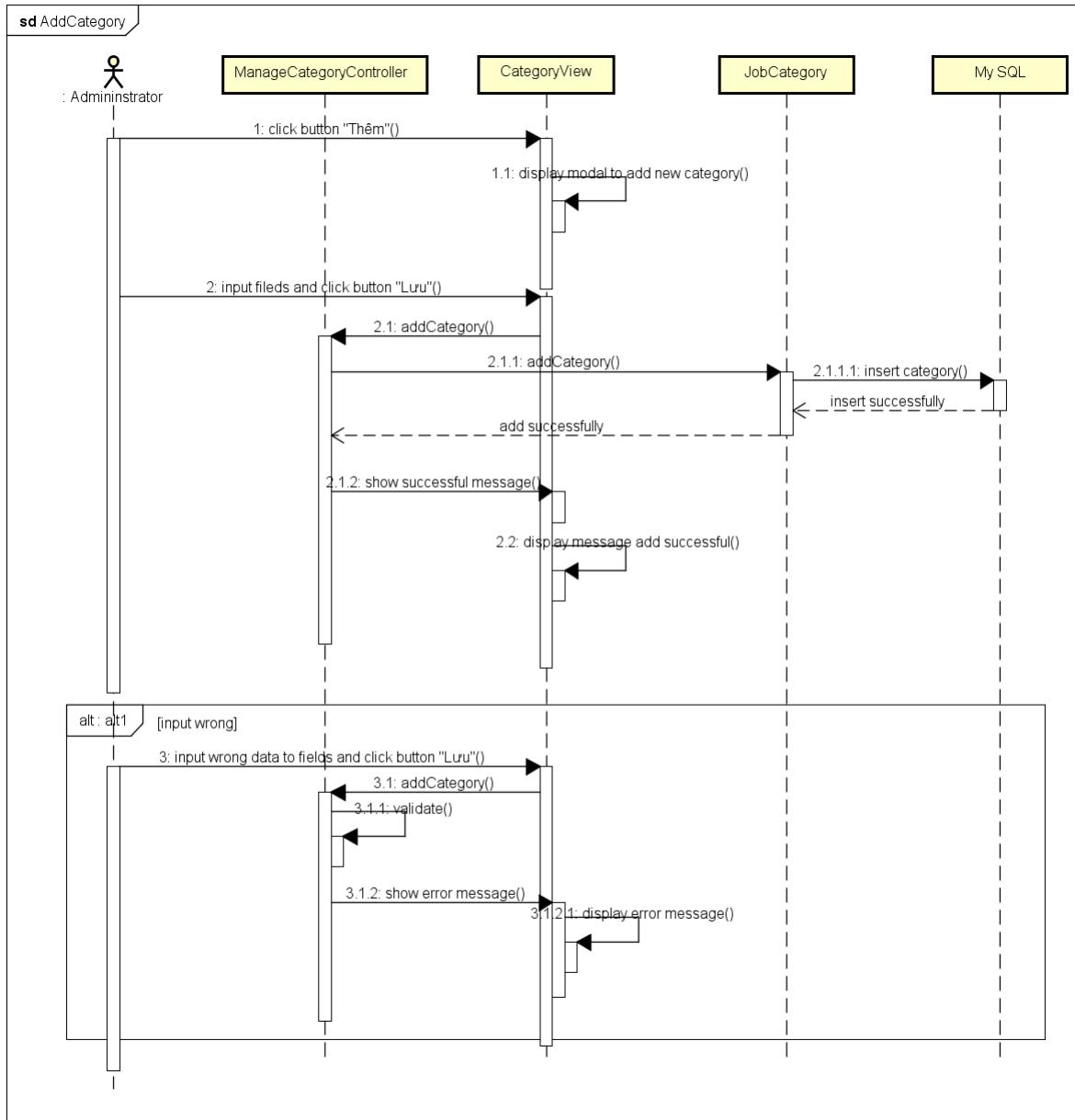


4.7.3.7.1.4 Class Specifications

Class	CategoryController
Description	Allows administrator add new job's category to list of category
Base Class	Controller
Source File	CategoryController.php
Namespace	App\Http\controllers/

Attributes	N/A			
Methods	Name	Input	Output	Description
	addCategory	request	Add new category and display to list view	Add new job's category to database

4.7.3.7.1.5 Sequence Diagram



4.7.3.7.2 View Category

4.7.3.7.2.1 Description

Admin view all job categories.

4.7.3.7.2.2 Design

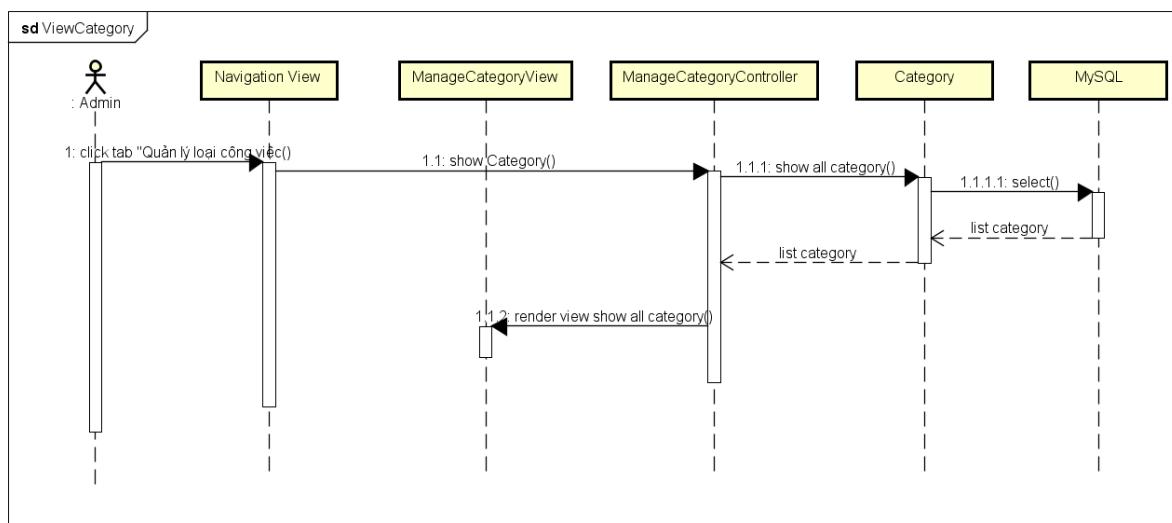
4.7.3.7.2.3 Class Diagram

Refer to class diagram of Add Category in 4.7.3.7.1 Add Category

4.7.3.7.2.4 Class Specifications

Class	CategoryController			
Description	Allows administrator view list job's category			
Base Class	Controller			
Source File	CategoryController.php			
Namespace	App/Http/controllers/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	showCategories	N/A	void	Show list job's category

4.7.3.7.2.5 Sequence Diagram



4.7.3.7.3 Delete Category

4.7.3.7.3.1 Description

Admin delete category.

4.7.3.7.3.2 Design

Refer to class diagram of Add Category in 4.7.3.7.1 Add Category

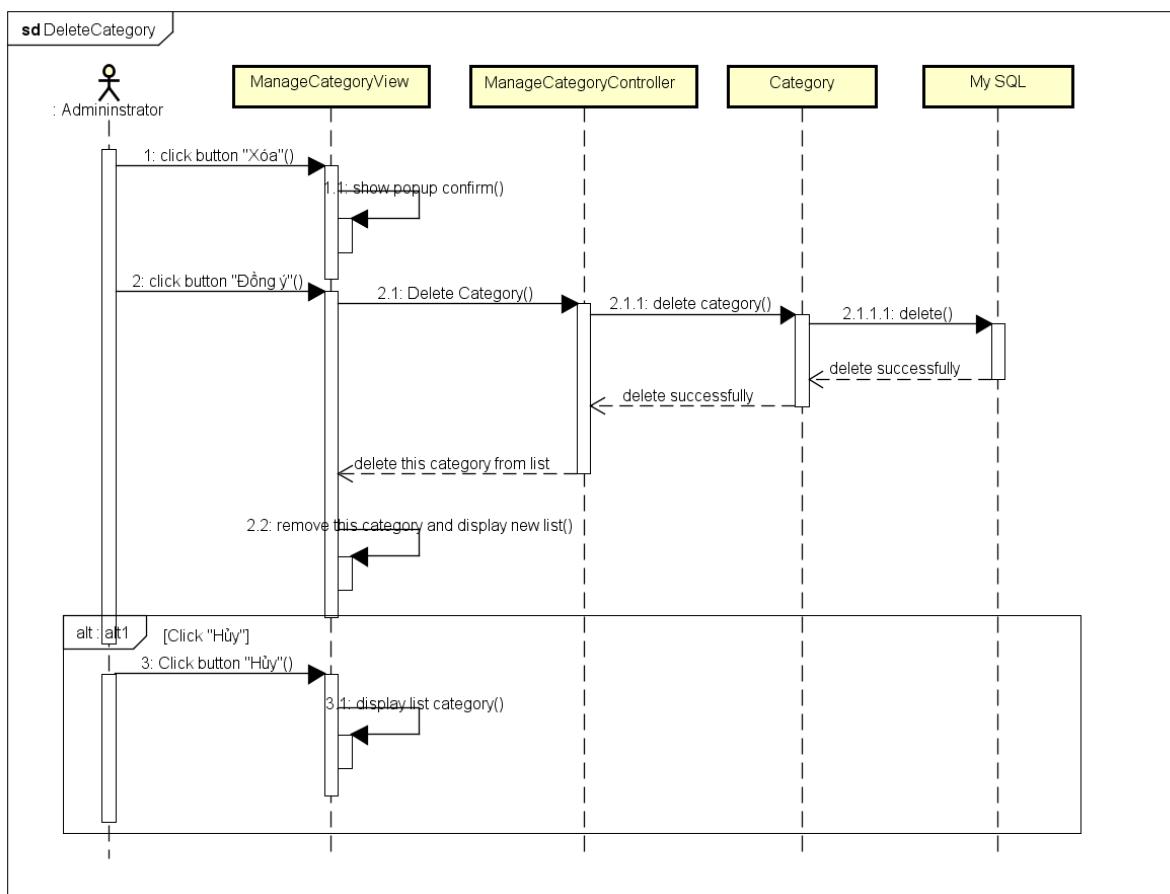
4.7.3.7.3.3 Class Diagram

Refer to class diagram of Add Category in 4.7.3.7.1 Add Category

4.7.3.7.3.4 Class Specifications

Class	CategoryController			
Description	Allows administrator to delete job's category			
Base Class	Controller			
Source File	CategoryController.php			
Namespace	App\Http\controllers/			
Attributes	N/A			
Methods	Name	Input	Output	Description
	deleteCategory	request	Delete successful and update list category view	Admin select a job's category and delete successfully

4.7.3.7.3.5 Sequence Diagram



4.7.3.7.4 Edit Category

4.7.3.7.4.1 Description

Admin edit name of a job category.

4.7.3.7.4.2 Design

Refer to class diagram of Add Category in 4.7.3.7.1 Add Category

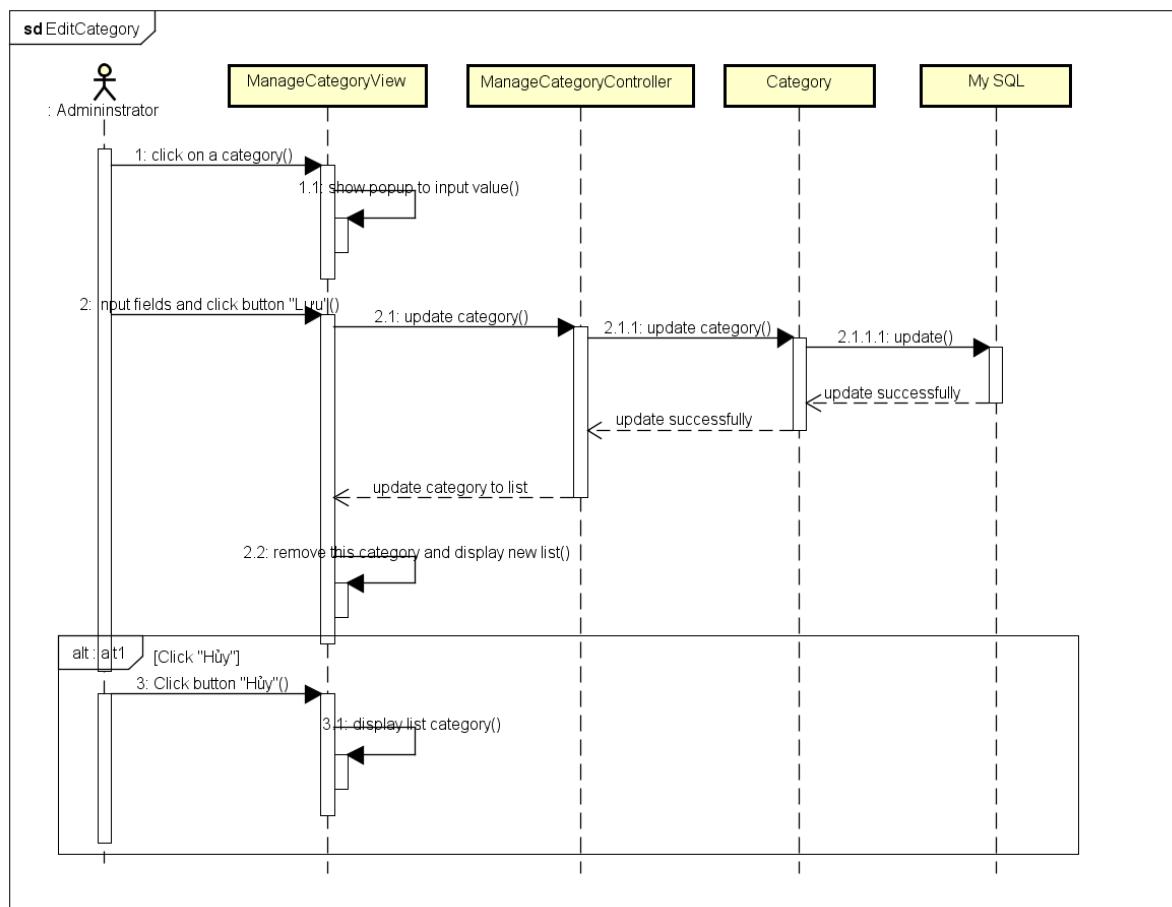
4.7.3.7.4.3 Class Diagram

Refer to class diagram of Add Category in 4.7.3.7.1 Add Category

4.7.3.7.4.4 Class Specifications

Class	CategoryController			
Description	Allows administrator edit job's category			
Base Class	Controller			
Source File	CategoryController.php			
Namespace	App/Http/controllers/			
Attributes	N/A			
	Name	Input	Output	Description
Methods	editCategory	categoryID, request	Update list category to view	Update this category to database

4.7.3.7.4.5 Sequence Diagram



4.7.4 System システム

4.7.4.1 Send hidden profile of a Job Seeker to Employer

4.7.4.1.1 Description

After an Employer choose a Job Seeker and this Employer agree pay to get Job Seeker's profile information, system deducts money from Employer's account, and send Job Seeker's profile information to Employer's email. If Employer is in session, information will be displayed ontime by a pop up.

4.7.4.2 Send notification to Member's email

4.7.4.2.1 Description

At the following cases: when a post of a member is deleted, when member is banned, when member is unbanned.

After a event already happened, system send email to member's email address to notify that the event is happened.

4.7.4.3 Delete Member

4.7.4.3.1 Description

After the number that a member is reached 3 times, system immediately send notify email of account deletion to member's email address, and after that, delete member's account (member's account no longer exist on system).

4.7.4.4 Ban Member

4.7.4.4.1 Description

After the number of deleted posts of a member is reached 3 times, system immediately send notify email to member's email address, and after that, system sets member's account status to banned. Account will be unbanned by system after specific time.

4.7.4.5 Deduct Employer's money

4.7.4.5.1 Description

After an Employer choose a Job Seeker and this Employer agree pay to get Job Seeker's profile information, if Member's account have money value equal or bigger than needed fee for a transaction, system deducts money from Member's account.

4.7.4.6 Create chart to compare between job types

4.7.4.6.1 Description

System provide chart of number of posts in each job type. When number of posts changed (new post created or deleted), system will update chart on time.

4.7.4.7 List matching post

4.7.4.7.1 Description

System matching a post with others by compare post's job category. Posts those are have the same job category, will be matched with others.

5 SOFTWARE TESTING ソフトウェアテスト

5.1 INTRODUCTION はじめに

5.1.1 Purpose 目的

This is the comprehensive test plan of the Jobs Finding System project. The purpose of this chapter describes scopes of test plan and activities which need to be taken during test process of project. It addresses the following items:

- Scope of Testing
- Requirement for Testing
- Test Strategy
- Test Resources
- Test Milestones
- Test Deliverables

5.1.2 References 参照資料

Title/File name	Author	Version
JFS_Software-Requirements-Specification_v1.0	DuyDD	v1.0
JFS_DataDesign_v1.0	NamTH	v1.0
JFS_Architecture-Desgin_v1.0	DuyDD, NamLH	v1.0
JFS_Screen-Desgin_v1.0	DuyDD	v1.0
JFS_TestCase_v1.0	SonPT	v1.0
JFS_UnitTest_v1.0	SonPT	v1.0
JFS_Project-Plan_v1.0	ThuyTT	v1.0

5.1.3 Background information 背景情報

The target of testing is ensured all functions will be run correctly as SRS description. In addition, restrict maximum of defect during the using of Website. To do this target, Website have to:

- Pass all of the following stages of testing: Unit Testing, Integration Testing, System Testing and Acceptance Testing.
- Pass all of the following types of testing: Function Testing, User Interface Testing, Data and Data Integrity Testing
- Run normally in popular browsers and many types of device or display resolution.

5.1.4 Scope of testing テスト範囲

The scope of test will be limited to testing on 3 browser: Google Chrome, Mozilla Firefox, Microsoft Edge.

Functional items and non-functional items will be verified and passed by eJob development team, then be validated and approved by project manager and supervisor of JFS project.

5.2 CONSTRAINTS 制約事項

- Deadline for testing only can be met if development progress is on time.
- Working process must be reported daily to PTL.
- Have more environments should be tested: Window XP, Window 8 ... and more browsers: Chrome 53, Firefox, Microsoft Edge 38 ...

5.2.1 Risk list リスク一覧

No	Risk	Mitigation	Contingencies
1.	PC have problem during test	Fix a problem	Find a other PC
2.	Tester may be sick in test phase or may leave the project before completion	+ Discussing + Committing + Assigning tasks appropriately	+ Persuading + Reviewing and re-planning the whole project
3.	Scope: SRS/SAD may be changed	+ Discussing + Reviewing the whole project frequently	+ Re-planning test phase + Re-creating test case
4.	Time of Coding phase may last too long	+ Committing + Assigning tasks	Working over time

Table 5.2.1 Risk list

5.3 REQUIREMENTS FOR TEST テスト要求

5.3.1 Test items テスト項目

No	Name of features and functions	Outline of features and functions	Note
1.	Guest_Common	Register	
2.		View Employer's Posts	
3.		View Job Seeker's Posts	
4.		Search Employer's Posts	
5.		Search Job Seeker's Posts	
6.		View detail Post	
7.	Member_Common	Login	
8.		Logout	
9.		Forgot password	
10.		Change password	
11.		Search Employer's Posts	
12.		Search Job Seeker's Posts	
13.		View Employer's Posts	
14.		View Job Seeker's Posts	
15.		Apply suitable Job	
16.		Get suitable Job Seeker	

17.		Report other Member's Post	
18.		Payment to get hidden Profile of Job Seekers	
19.		Online chat with Admin	
20.		Comment and report issue	
21.	Member_Manage Personal Page	View Profile	
22.		Edit Profile	
23.		View personal Page	
24.	Member_View other Member's Personal Page	View other Member's Posts	
25.		View other Member's basic Profile	
26.	Member_Manage Post	Create Post	
27.		Edit Post	
28.		Delete Post	
29.		View selected Posts	
30.		View Posts	
31.		View detail Post	
32.		View people choose	
33.		View matching Posts	
34.	Admin_Common	Admin Login	

35.		Admin Logout	
36.		Admin Forgot password	
37.		Online chat with Member	
38.		Send notify by email	
39.		Add a Admin	
40.	Admin_Manage Members	View Members	
41.		Search Members	
42.		View Member Profile	
43.		Ban Member	
44.		Unban Member	
45.		Delete Member	
46.		Add money into Employer's Account	
47.	Admin_Manage Posts	View Employer's Posts	
48.		View Job Seeker's Posts	
49.		Search Employer's Posts	
50.		Search Job Seeker's Posts	
51.		Delete Post	
52.		View detail Post	
53.		View comments and reported issues of	

	Admin_Manage Feedback	website	
54.		Delete comment and reported issues of website	
55.	Admin_Statistic	Statistic earned money from Members	
56.		View Chart	
57.	Admin_Reports	View Reports about Post	
58.		Delete Reports about Post	
59.	Admin_Manage Category	Add Category	
60.		View Category	
61.		Delete Category	
62.		Edit Category	
63.	System	Send hidden profile of a Job Seeker to Employer	
64.		Send notification to Member's email	
65.		Delete Member	
66.		Ban Member	
67.		Deduct Employer's money	
68.		Create chart to compare between job types	
69.		List matching posts	

Table 5.3.1 Test items

5.3.2 Acceptance Test Criteria テスト受入基準

List of criteria to define what levels of test quality are sufficient to move to the next testing phase:

- Test coverage : 100%
- Successful Test coverage : 90%

- Number of defects/Weighted defects : 3~4 defects/KLOC

List of criteria which are based on to accept the products, for Test team to accept source code after Unit test of Development team:

- Number of UTC/KLOC: 40 UTC/KLOC
- Number or Weighted defects/KLOC : 3~4 defects/KLOC
- Statement coverage: 100%
- Branch coverage: 100%
- Path coverage: 100%

List of criteria which are based on to accept the products, for Integration test:

- Number of UTC/KLOC: 30 UTC/KLOC
- Number or Weighted defects/KLOC : 2~3 defects/KLOC
- Statement coverage: 100%
- Branch coverage: 100%
- Path coverage: 100%

List of criteria which are based on to accept the products, for System test:

- Number of UTC/KLOC: 20 UTC/KLOC
- Number or Weighted defects/KLOC : 5~6 defects/KLOC
- Statement coverage: 100%
- Branch coverage: 100%
- Path coverage: 100%

List of criteria which are based on to accept the products, for Acceptance test:

- Number or Weighted defects/KLOC : 1~2 defects/KLOC
- Statement coverage: 100%
- Branch coverage: 100%
- Path coverage: 100%

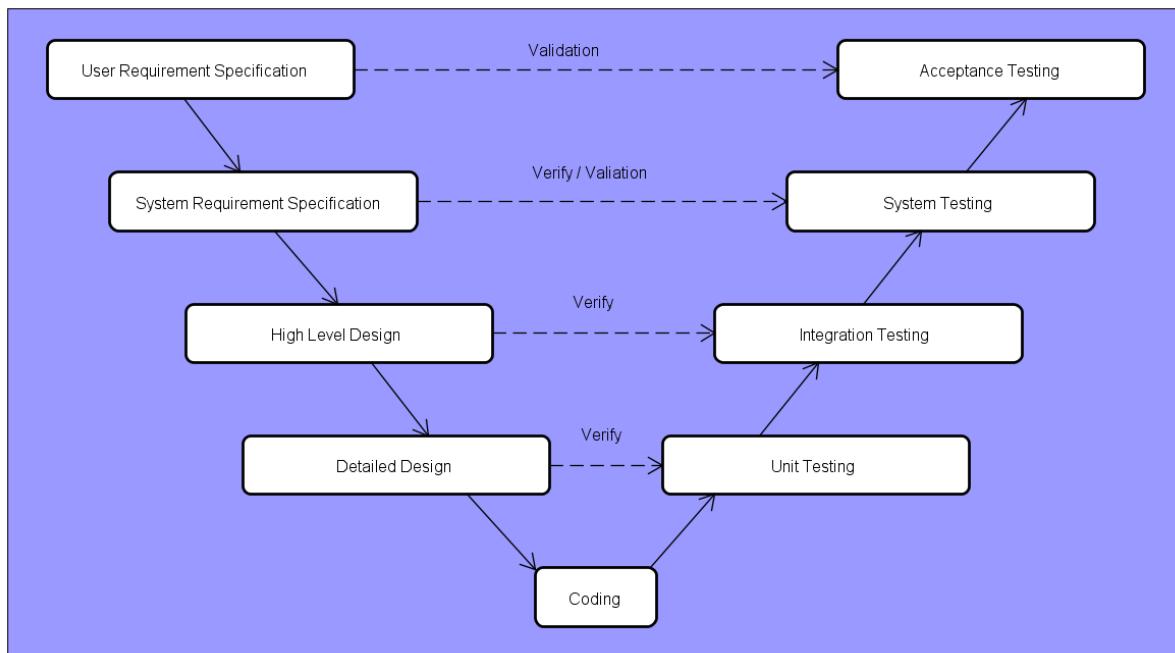
5.3.3 Feature not to be tested テストされない特徴

- The stable of website when do no connect internet.
- Manage Website's payment activities. Manage the activity that user pay for system when they use paid-required service.

5.4 TEST STRATEGY テスト戦略

5.4.1 Test Model テストモデル

JFS follows V-Model process: The entire process of our Project is divided into 2 corresponding stages: Developing and Testing. Each developing process will be performed in parallel with the corresponding testing process. Bugs will be detected early.



JFS will be tested by 4 phases:

Phase 1: Unit testing

- Unit testing will be performed 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)

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)

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)

Phase 4: Acceptance testing

- Acceptance testing is performed after the system testing
- Base on business/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.

5.4.2 Test types テストタイプ

5.4.3 Function Testing 機能テスト

- Function 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 ensures that the requirements are properly satisfied by application. This type of testing is not concerned with how processing occurs, but rather, with 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.

Test Objective:	Ensure proper target-of-test functionality, including navigation, data entry, processing, and retrieval.
Technique:	<p>Execute each use case, use-case flow, or function, using valid and invalid data, to verify the following:</p> <ul style="list-style-type: none"> - The expected results occur when valid data is used. - The appropriate error or warning messages are displayed when invalid data is used. - Each business rule is properly applied. - Use Test tool
Completion Criteria:	<ul style="list-style-type: none"> - All planned tests have been executed. - All identified defects have been addressed and closed
Special Considerations:	Identify or describe those items or issues (internal or external) that impact the implementation and execution of function test

Table 5.4.3 Function Testing

5.4.4 User Interface Testing ユーザーインターフェーステスト

- User interface testing, a testing technique used to identify the presence of defects is a product/software under test by using GUI.
- GUI testing evaluates design elements such as layout, colors, [fonts](#), font sizes, labels, text boxes, text formatting, captions, buttons, lists, icons, links and content.

Test Objective:	<p>Verify the following:</p> <p>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)</p> <p>Window objects and characteristics, such as menus, size, position, state, and focus conform to standards.</p>
Technique:	Create or modify tests for each window to verify proper navigation and object states for each application window and objects.
Completion Criteria:	Each window successfully verified to remain consistent with benchmark version or

	within acceptable standard.
Special Considerations:	Not all properties for custom and third party objects can be accessed.

Table 5.4.4 User Interface Testing

5.4.5 Data and Database Integrity Testing データおよびデータベース整合性テスト

Data integrity corresponds to the quality of data in the databases and to the level by which users examine data quality, integrity and reliability. Data integrity testing verifies that the data in the database is accurate and functions as expected within a given application.

Test Objective:	Ensure database access methods and processes function properly and without data corruption.
Technique:	<ul style="list-style-type: none"> - Invoke each database access method and process, seeding each with valid and invalid data or requests for data. - 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
Completion Criteria:	All database access methods and processes function as designed and without any data corruption.
Special Considerations:	<ul style="list-style-type: none"> - Testing may require a DBMS development environment or drivers to enter or modify data directly in the databases. - Processes should be invoked manually. - Small or minimally sized databases (limited number of records) should be used to increase the visibility of any non-acceptable events.

Table 5.4.5 Data and Database Integrity Testing

5.4.6 Test stages テストステージ

Type of Tests	Stage of Test			
	Unit	Integration	System	Acceptance
Function Test	X	X	X	X
User Interface test	X		X	
Data integrity test		X	X	

Table 5.4.6 Test Stages

5.5 RESOURCE リソース

5.5.1 Human Resource 人的リソース

Worker/Doer	Role	Specific Responsibilities/Comments	Location
SonPT	Test Leader	Manage Test resource and assign test tasks Create TP Review TC Create and review Test Report	FPT University Vietnam
ThuyTT	Tester	Create and review TC for modules Create and review UT Execute test Create Test view points	FPT University Vietnam
NamLH	Tester	Create and review TC for modules Execute test Report test result Create and review Test Report	FPT University Vietnam

Table 5.5.1 Human Resource

5.5.2 Test management テスト管理

5.5.2.1 *Test management* テスト管理

Test management is practice of organizing and controlling the process and artifacts required for testing effort.

The general goal of test management is to allow teams to plan, develop, execute, and assess all testing activities within the overall software development effort. This includes coordinating efforts of all those involved in the testing effort, tracking dependencies and relationships among test assets and, most importantly, defining, measuring, and tracking quality goals.

- **Test planning** is the overall set of tasks that address the questions of why, what, where, and when to test. The reason why a given test is created is called a test motivator (for example, a specific requirement must be validated). What should be tested is broken down into many test cases for a project. Where to test is answered by determining and documenting the needed software and hardware configurations. When to test is resolved by tracking iterations (or cycles, or time period) to the testing.
- **Test authoring** is a process of capturing the specific steps required to complete a given test. This addresses the question of how something will be tested. This is where somewhat abstract test cases are developed into more detailed test steps, which in turn will become test scripts (either manual or automated).

- **Test execution** entails running the tests by assembling sequences of test scripts into a suite of tests. This is a continuation of answering the question of how something will be tested (more specifically, how the testing will be conducted).
- **Test reporting** is how the various results of the testing effort are analyzed and communicated. This is used to determine the current status of project testing, as well as the overall level of quality of the application or system.

5.5.3 Defect management 欠陥管理

Defect management is crucial to closing the loop between requirements, implementation and verification and validation. Traditional defect tracking management, implemented in a standalone fashion, can no longer address the complexity and pace of change in modern software development. Defect management processes must be tightly interlinked with all of the other software development processes. The defect management process contains the following elements:

- **Defect Discovery** – Identification and reporting of potential defects. The defect tracking software must be simple enough so that people will use it, but ensure that the minimum necessary information is captured. The information captured here should be enough to reproduce the defect and allow development to determine root cause and impact.
- **Defect Analysis & Prioritization** – The development team determines if the defect report corresponds to an actual defect, if the defect has already been reported, and what the impact and priority of the defect is. Prioritization and scheduling of the defect resolution is often part of the overall change management process for the software development organization.
- **Defect Resolution** – Here the development team determines the root cause, implements the changes needed to fix the defect, and documents the details of the resolution in the defect management software, including suggestions on how to verify the defect is fixed. In organizations using [Software Product Lines](#) approaches, or other shared component approaches, defect resolution may need to be coordinated across multiple branches of development.
- **Defect Verification** – The build containing the resolution to the defect is identified, and testing of the build is performed to ensure the defect truly has been resolved, and that the resolution has not introduced side effects or regressions. Once all affected branches of development have been verified as resolved, the defect can be closed.
- **Defect Communication** – This encompasses automatic generation of defect metrics for management reporting and process improvement purposes, as well as visibility into the presence and status of defects across all disciplines of the software development team.

5.6 TEST ENVIRONMENT たテス ト環 境

5.6.1 Hardware ハードウェア

Name	Screen Resolution	Browser	Purpose	Detail
Laptop HP 4430s		Google Chrome, Mozilla Firefox, Microsoft Edge	Device for create and execute test	Microsoft Windows 10 Professional
Laptop Dell Vostro 3560	1366x768	Google Chrome, Mozilla Firefox, Internet Explorer	Device for create and execute test	Microsoft Windows 8.1 Professional

Table 5.6.1 Hardware

5.6.2 Software ソフトウェア

Name	Purpose	Detail
Chrome	Executing test	Chrome 53.0
Firefox	Executing test	Firefox
Microsoft Edge	Executing test	Microsoft Edge 38

5.7 TEST MILESTONES テストマイルストーン

Milestone Task	Effort (pd)	Start Date	End Date
Create Test Plan	6	October 10, 2016	October 15, 2016
Review & update TP	2	October 16, 2016	October 16, 2016
Create Unit Test case	2	October 17, 2016	October 18, 2016
Review & update UTC	1	October 19, 2016	October 19, 2016
Create Integration Test case	3	October 20, 2016	October 25, 2016
Review & Update Integration TC	1	October 26, 2016	October 26, 2016
Create System Test case	3	October 27 , 2016	November 01 , 2016
Review & Update System TC	1	November 02, 2016	November 02, 2016
Execute Unit Test Iterative 1	3	November 03, 2016	November 04, 2016
Execute Integration test Iterative 1	3	November 05, 2016	November 09, 2016
Execute System test Iterative 1	3	November 10, 2016	November 15, 2016
Execute Unit Test Iterative 2	3	November 10, 2016	November 15, 2016
Execute Integration test Iterative 2	3	November 10, 2016	November 15, 2016
Execute System test Iterative 2	3	November 10, 2016	November 15, 2016

Table 5.7 Test Milestones

5.8 DELIVERABLES 納品物

No	Deliverables	Responsibilities	Delivered Date
1.	Test Plan	Test Lead	October 20 th , 2016
2.	Unit Test cases	Tester	November 15 th , 2016
3.	Common Test cases	Tester	November 15 th , 2016
4.	Integration Test Cases	Tester	November 15 th , 2016
5.	System Test cases	Tester	November 15 th , 2016
6.	Defect log Management	Tester	December 04 th , 2016
7.	Test reports	PM	December 04 th , 2016

Table 5.8 Deliverables

5.9 TEST CASE テストケース

Refer to:

- JFS_Integration Test Case_v1.0.xlsx
- JFS_System Test Case_v1.0.xlsx
- JFS_Common Test Case_v1.0.xlsx
- JFS_UnitTestCase_GuestCommon_v1.0.xlsx
- JFS_UnitTestCase_MemberCommon_v1.0.xlsx
- JFS_UnitTestCase_MemberManagePersonalPage_v1.0.xlsx
- JFS_UnitTestCase_MemberManagePost_v1.0.xlsx
- JFS_UnitTestCase_AdminStatistic_v1.0.xlsx
- JFS_UnitTestCase_AdminManageReports_v1.0.xlsx
- JFS_UnitTestCase_AdminManagePosts_v1.0.xlsx
- JFS_UnitTestCase_AdminManageMembers_v1.0.xlsx
- JFS_UnitTestCase_AdminManageFeedback_v1.0.xlsx
- JFS_UnitTestCase_AdminManageCategory_v1.0.xlsx

5.10 TEST REPORT テストレポート

5.10.1 Unit Test Report 単体テスト

No	Function code	Passed	Failed	Untested	N	A	B	Total Test Cases
1	Guest Register	15	0	0	15	0	0	15
2	Guest View Employer's Posts	10	0	0	10	0	0	10
3	Guest View Job Seeker's Posts	10	0	0	10	0	0	10
4	Guest Search Employer's Posts	15	0	0	15	0	0	15
5	Guest Search Job Seeker's Posts	12	0	0	12	0	0	12
	Sub total	62	0	0	62	0	0	62

Table 5.10.1 Guest Common

No	Function code	Passed	Failed	Untested	N	A	B	Total Test Cases
1	Member Login	10	0	0	10	0	0	10
2	Member Logout	5	0	0	5	0	0	5
3	Member Change Password	8	0	0	8	0	0	8
4	Member Search Employer's Posts	15	0	0	15	0	0	15
5	Member Search Job Seeker's Posts	12	0	0	12	0	0	12
6	Member Report Other Member's Post	7	0	0	7	0	0	7
7	Member feedback	4	0	0	4	0	0	4
	Sub total	61	0	0	61	0	0	61

Table 5.10.2 Member Common

5.10.2 Common Test Report 一般テストレポート

No	Module code	Pass	Fail	Untested	N/A	Number of test cases
1	Common	12	0	0	0	12
2	Security	60	0	0	0	60
3	UI	26	0	0	0	26
	Sub total	98	0	0	0	98

Table 5.10.3 Common Test report

5.10.3 Integration Test Report 統合テストレポート

No	Module code	Pass	Fail	Untested	N/A	Number of test cases
1	Guest Function	60	0	2	0	62
2	Admin Function	146	0	0	0	146
3	Member Function	154	0	0	0	154
4	System Function and Transaction	68	0	0	0	68
	Sub total	428	0	2	0	430

Table 5.10.4 Integration Report

5.10.4 System Test Report システムテストレポート

No	Module code	Pass	Fail	Untested	N/A	Number of test cases
1	Common	24	2	2	0	28
2	Homepage	22	0	0	0	22
3	Member's Account	68	0	0	0	68
4	Create Employer's Post	66	0	0	0	66
5	Create Job Seeker's Post	62	0	0	0	62
6	Post Detail	46	0	0	0	46
7	Matching Post	48	0	0	0	48
8	Admin Function	150	0	0	0	150
	Sub total	486	2	2	0	490

Table 5.10.5 System Test Report

5.10.5 Test Report テストレポート

Phase	Round 1				Round 2			
	Pass	Fail	Untested	N/A	Pass	Fail	Untested	N/A
Unit Test	450	0	0	0	450	0	0	0
Integration Test	480	46	0	0	526	0	0	0
System Test	508	4	0	0	512	0	0	0
Total	1438	50	0	0	1488	0	0	0

Table 5.10.5 Test Report

Defects	Logged	%Closed	%Invalid	%Open
Critical	0	0%	0%	0%
High	2	0%	0%	0%
Medium	15	100%	0%	0%
Low	33	100%	0%	0%
Total	50	100%	0%	0%

Table 5.10.5.1 Defect report

6 USER MANUAL ユーザーマニュアル

6.1 INSTALL GUIDE インストールガイド

6.1.1 Introduction はじめに

6.1.1.1 Purpose 目的

This document will give general description of how to run our project at local machine and deploy to Azure.

6.1.1.2 Environment 環境

Following are the software required to start the JFS system:

- Operating System: Windows 7, Windows 8, Window 10
- Browsers: Firefox 40, Chrome 44 or higher.

- Database: MySQL
- IDE: PHP Storm
- Azure Cloud, Ubuntu Virtual Machine, Laravel, Nginx server and Php 5.6.

6.1.2 Installation Guideline インストールのガイドライン

6.1.2.1 Run at local.

1. Install Composer, PHP 5.6, Laravel 5.3
2. Open command line and then go to folder of Project's source code.



Run command “composer update”

A screenshot of a Windows Command Prompt window titled "cmd" with the path "C:\Windows\system32\cmd.exe". The window shows the command "composer update" being run. The output indicates that 1 package was updated, and it took 0 seconds. The command prompt then asks if the user wants to run "php artisan migrate".

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

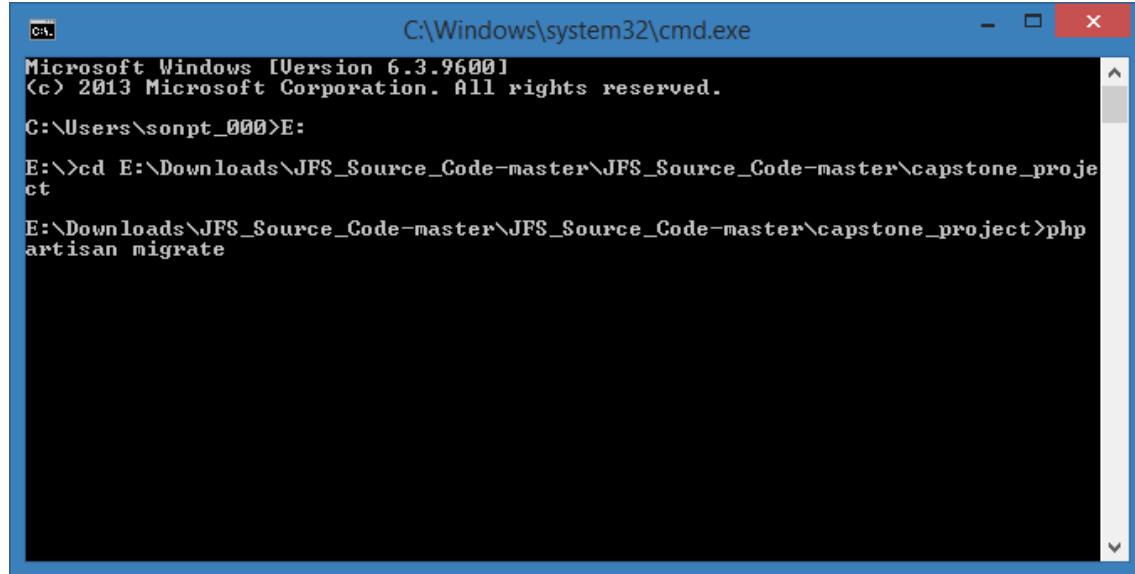
C:\Users\sonpt_000>E:
E:\>cd E:\Downloads\JFS_Source_Code-master\JFS_Source_Code-master\capstone_project
E:\Downloads\JFS_Source_Code-master\JFS_Source_Code-master\capstone_project>composer update

Up-to-date packages (1):
  laravel/framework v5.4.28

Nothing to install or update.
Time: 0 seconds

Would you like to run "php artisan migrate"? [y/N] y
```

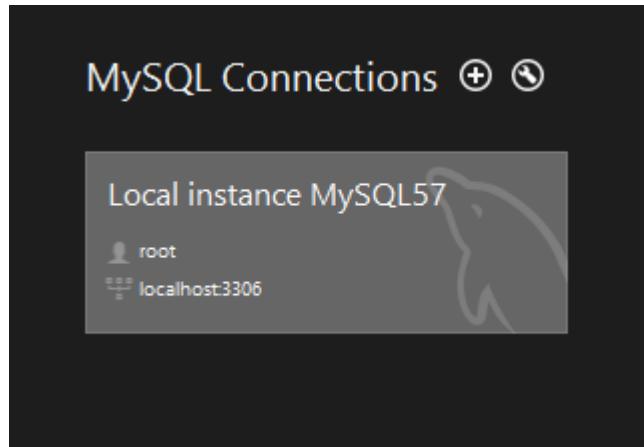
After “composer update” success, run “php artisan migrate”



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

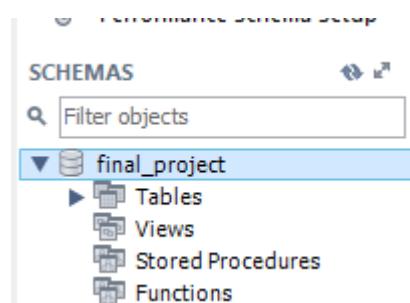
C:\Users\sonpt_000>E:
E:>\cd E:\Downloads\JFS_Source_Code-master\JFS_Source_Code-master\capstone_project
E:\Downloads\JFS_Source_Code-master\JFS_Source_Code-master\capstone_project>php artisan migrate
```

3. Install my SQL Workbench.

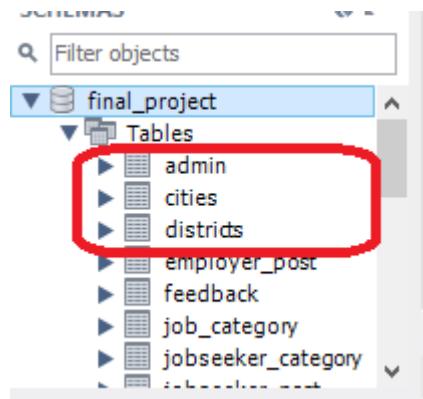


Then connect with the user name "root"

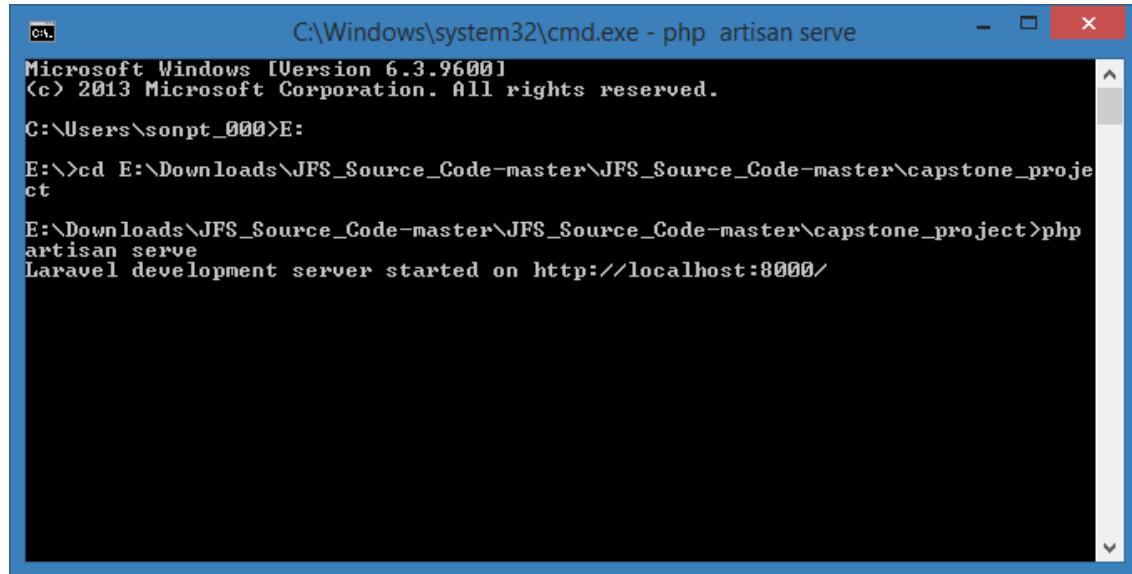
4. After run "php artisan migrate" at step 1, connect, and then see database "final_project"



5. Add data of Admin, City, District, and Category in tables below:



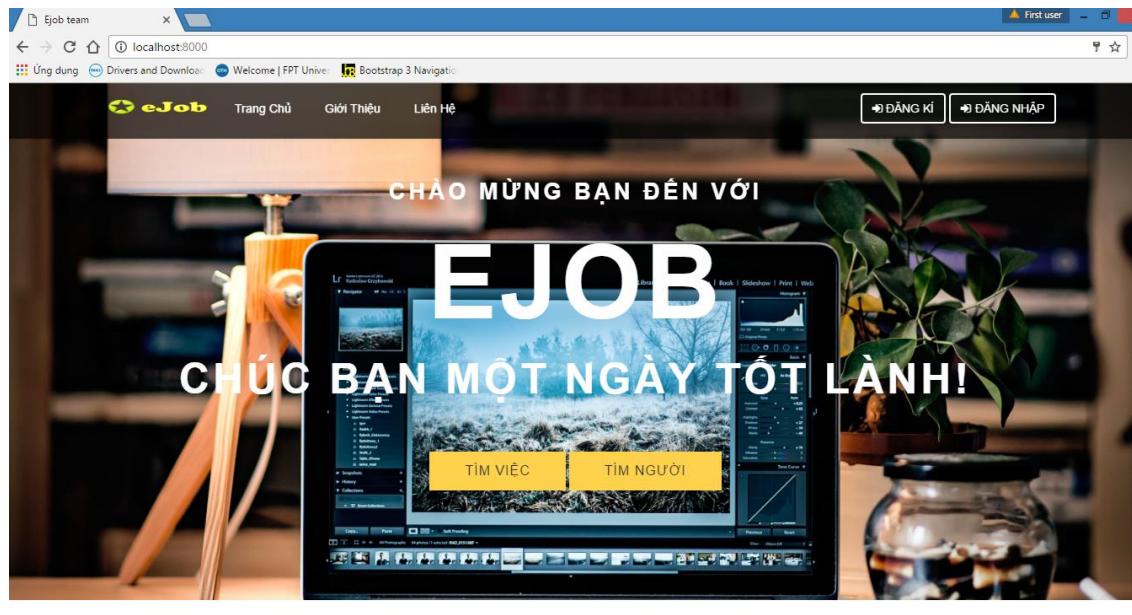
6. Run “php artisan serve”



```
C:\Windows\system32\cmd.exe - php artisan serve
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\sonpt_000>E:
E:>cd E:\Downloads\JFS_Source_Code-master\JFS_Source_Code-master\capstone_project
E:\Downloads\JFS_Source_Code-master\JFS_Source_Code-master\capstone_project>php artisan serve
Laravel development server started on http://localhost:8000/
```

7. Go to “localhost:8000” at browser.



6.1.2.2 Deploy to Azure

1: Upload source code to Github. (<https://github.com>)

2: Create an Azure account.

3: In Azure, create an Ubuntu virtual machine (Standard DS1 (1core, 3.5 GB memory), 7GB, 2 disk data) include an Ip Address.

4: In Ubuntu virtual machine, install nginx server and Php 5.6.

Install Laravel 5.3.

5: Install MySQL in Ubuntu virtual machine.

6: Deploy source code from Github to Ubuntu virtual machine on Azure.

7: In Ubuntu , go to source code folder, run : “sudo composer update”, then run “php artisan migrate”

8: Import Admin, City and District and Category data to MySQL

9: Start Nginx server.

6.1.3 Other considerations 他の検討事項

N/A

6.2 USER GUIDE ユーザーガイド

6.2.1 Guest ゲスト

6.2.1.1 Register 登録

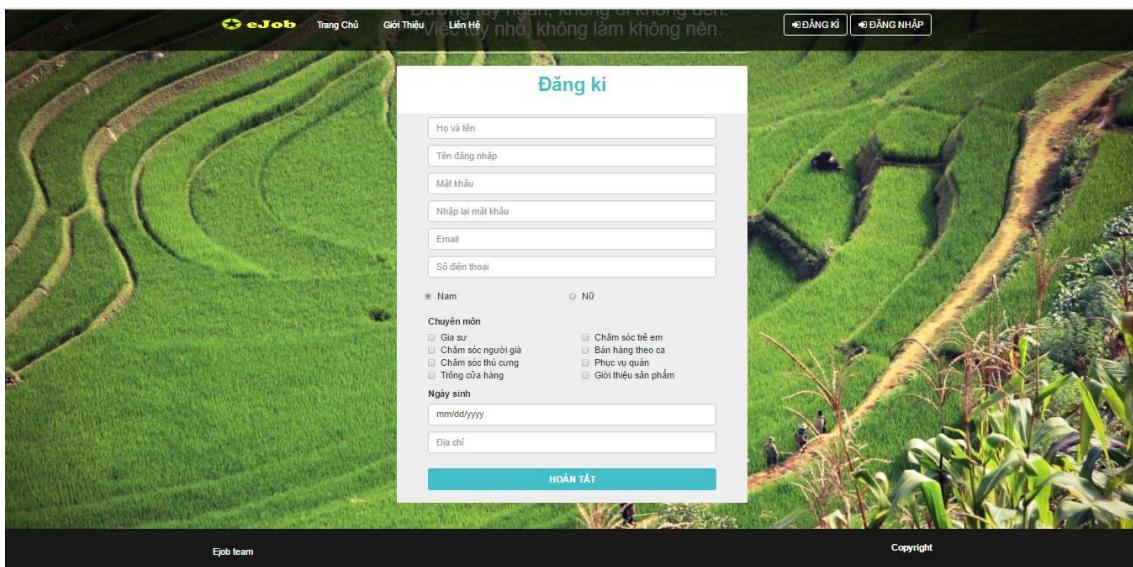
Homepage is the first page displayed when user go to Website. From homepage the user can go to any other page.

To register account of website, User do the steps below:

Step 1: Open browser and enter to the address bar: <http://207.46.136.186/home>, then Website's Homepage is displayed.

Step 2: Click on “Đăng ký” button in the header to go to Register page.

Step 3: Fill all needed information and then click “Hoàn tất” button.

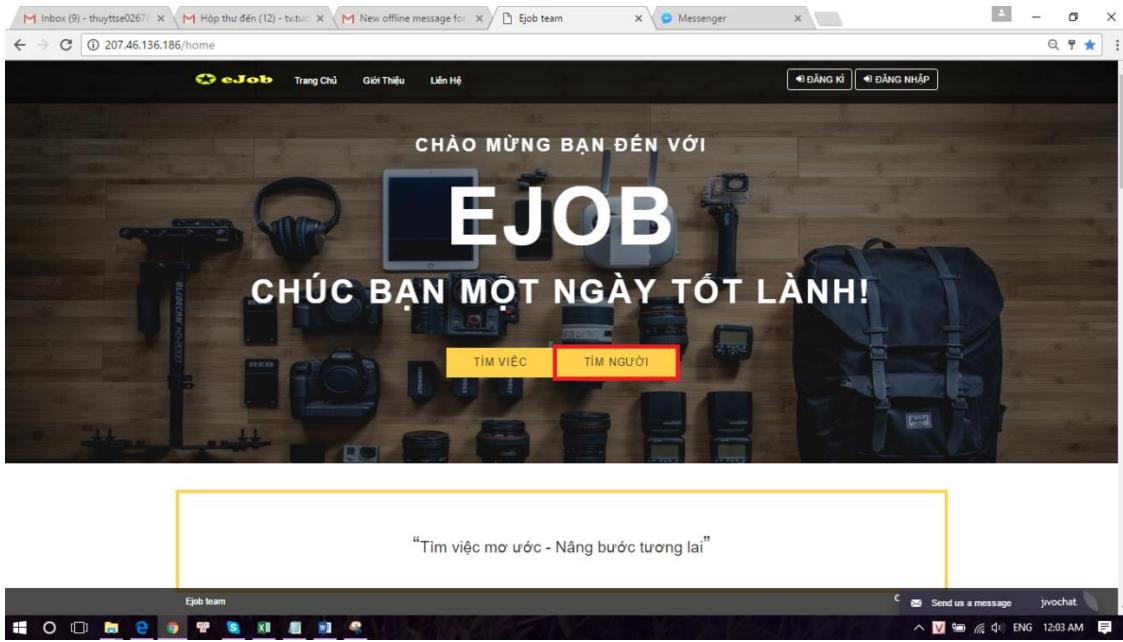


Step 4: Check confirmation mail that be sent to registered email.

Step 5: Click at confirmation link in confirmation mail to verify account registration. Then go to homepage and login.

6.2.1.2 Search and view Employer's Posts

In this page, guest can search and view all Employer's Posts.



Step 1: Open browser and go to <http://207.46.136.186/home>, then Homepage is displayed.

Step 2: Click “Tìm Người” button.

Step 3: View all displayed Employer's Posts.

The screenshot shows a web browser window with multiple tabs open. The active tab is '207.46.136.186/search/jobseekerPost'. The page title is 'TÌM NGƯỜI LÀM MONG MUỐN' (Search for people who want to work). It features a search bar with dropdowns for 'Nhập từ khóa' (Search term), 'Tất cả ngành nghề' (All industries), and 'Tất cả thành phố' (All cities). Below the search bar is a list of job posts:

- Chăm sóc chó bông - Cầu Giấy - Hà Nội - 12121 VND/giờ - Đã đăng 2 ngày trước
- Nhân viên bán hàng cho cửa hàng tiện lợi 24/24 - Long Biên - Hà Nội - 55500 VND/giờ - Đã đăng 4 ngày trước
- Tìm việc bán hàng hoặc giới thiệu sản phẩm - Cầu Giấy - Hà Nội - 40000 VND/giờ - Đã đăng 4 ngày trước
- Day học sinh lớp 1 - Long Biên - Hà Nội - 50000 VND/giờ - Đã đăng 4 ngày trước
- Tôi muốn tìm việc làm gia sư hoặc chăm sóc người lớn tuổi - Cầu Giấy - Hà Nội - 80000 VND/giờ - Đã đăng 4 ngày trước
- Cần làm về giúp việc gia đình chăm sóc thú cưng - Long Biên - Hà Nội - 12000 VND/giờ - Đã đăng 4 ngày trước

On the right side, there is a profile for 'Phạm Tuấn Sơn' (Pham Tuann Son) with the role 'Chăm sóc chó bông' (Dog grooming). Below the profile is a note: 'Mô tả bản thân: Yêu thích động vật' (Self-description: Loves animals).

6.2.1.3 Search and view Job Seeker's Posts

In this page, guest can search and view all Job Seeker's Posts.

The screenshot shows the homepage of the eJob website. The main banner features a wooden background with various electronic equipment like cameras, headphones, and a backpack. The text 'CHÀO MỪNG BẠN ĐẾN VỚI EJOB CHÚC BẠN MỘT NGÀY TỐT LÀNH!' (Welcome to EJOB,祝您有一个美好的一天!) is displayed prominently. Below the banner are two buttons: 'TÌM VIỆC' (Search Job) and 'TÌM NGƯỜI' (Search People). The 'TÌM VIỆC' button is highlighted with a yellow border. A yellow box surrounds the text '“Tim việc mơ ước - Nâng bước tương lai”' (Achieve your dream job - Advance your future steps) located below the buttons.

Step 1: Open browser and go to <http://207.46.136.186/home>, then Homepage is displayed.

Step 2: Click “Tìm Việc” button.

Step 3: View all displayed Job Seeker's Posts.

The screenshot shows the homepage of the Jobs Finding System. At the top, there is a navigation bar with links for 'Trang Chủ', 'Giới Thiệu', 'Liên Hệ', 'ĐĂNG KÝ', and 'ĐĂNG NHẬP'. Below the navigation bar, there is a search bar with the placeholder 'Nhập từ khóa' and dropdown menus for 'Tất cả ngành nghề' and 'Tất cả thành phố'. A main content area displays a list of job posts:

- Chăm sóc chuột hamster - Cầu Giấy - Hà Nội - 12121 VND/giờ - Đã đăng 2 ngày trước
- Trồng cối shop quần áo - Long Biên - Hà Nội - 9 VND/giờ - Đã đăng 4 ngày trước
- Tìm người chăm sóc thú cưng - Hoàn Kiếm - Hà Nội - 10000 VND/giờ - Đã đăng 4 ngày trước
- Chăm sóc cụ già trên 70 tuổi - Cầu Giấy - Hà Nội - 72000 VND/giờ - Đã đăng 4 ngày trước
- Gia sư môn cầu lông đường phố - Cầu Giấy - Hà Nội - 12000 VND/giờ - Đã đăng 5 ngày trước
- Nhân viên bán hàng mang thời trang cao cấp - Cầu Giấy - Hà Nội - 51000 VND/giờ - Đã đăng 1 tuần trước

On the right side of the page, there is a profile section for 'Phan Hoàng Yến' with the title 'Nhân viên bán hàng mang thời trang cao cấp' and a photo of her. Below the profile, there is a detailed description of a job post:

Mô tả công việc:
Bán hàng, thực hiện các chỉ tiêu doanh số được giao. Thực hiện các bước trong quy trình bán hàng giới thiệu, tư vấn và bán các sản phẩm cho khách hàng. Làm việc theo ca. Thường xuyên kiểm tra cửa hàng, quầy hàng và hàng hóa để đảm bảo số lượng hàng hóa đúng, chính xác. Tính toán số tiền cần thanh toán cho khách hàng. Giao tiếp với khách hàng, giải quyết các vấn đề phát sinh. Thực hiện việc mua sắm phục vụ công việc kinh doanh.

At the bottom of the page, there is a footer with links for 'Send us a message' and 'jvochat'.

6.2.1.4 View detail Post

Step 1: Open browser and go to: <http://207.46.136.186/home>, then Homepage is displayed.

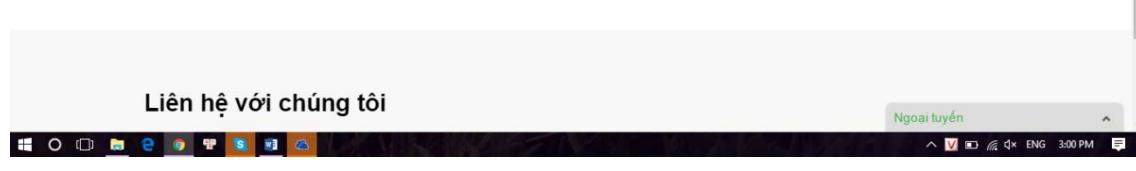
Step 2: Scroll to “Bài đăng tìm việc” or “Bài đăng tìm người”

The screenshot shows the homepage of the Jobs Finding System again. A red box highlights the 'Bài đăng tìm việc' button in the top left corner of the page. Below the button, there is a text box with the message: 'Hãy lựa chọn và tìm cho mình những người làm phù hợp nhất...'.

Below the text box, there are several job post cards:

- Chăm sóc thú cưng - Cầu Giấy-Hà Nội
- Bán hàng theo ca - Giới thiệu sản phẩm - Long Biên-Hà Nội
- Gia sư - Bán hàng theo ca - Giới thiệu sản phẩm - Cầu Giấy-Hà Nội
- Gia sư - Long Biên-Hà Nội
- Gia sư - Cầu Giấy-Hà Nội
- Chăm sóc thú cưng - Long Biên-Hà Nội

At the bottom of the page, there is a footer with links for 'Send us a message' and 'jvochat'.



Step 3: Click any post.

Step 4: Post detail displayed.

The screenshot shows a detailed job listing for a part-time salesperson. The title is 'BÁN HÀNG THEO CA TỐI' (Part-time Sales). The job description includes: Loại công việc: Bán hàng theo ca; Địa chỉ: Hà Đông - Hà Nội; Thời gian: 23 giờ trước; Mức lương mong muốn: 150,000 (VND) 1 giờ; Thời gian làm việc: 2016-12-16; Người đăng: Dương Đăng Duy. There are two buttons: 'CHỌN NGƯỜI NÀY' (Select This Person) and 'Bảo cáo bài đăng' (Report Post). Below the main details, there are sections for 'MÔ TẢ CÔNG VIỆC' (Job Description) and 'MÔ TẢ BẢN THÂN' (About Yourself), both of which are currently empty. The status bar at the bottom indicates 'Bài đăng liên quan' (Related Posts) and 'Ngoại tuyến' (Offline).

The screenshot shows a job listing on the eJob website. The title is "NHÂN VIÊN BÁN HÀNG MÀNG THỜI TRANG CAO CẤP". The job description includes a photo of a person cleaning, details like address (349 Cầu Giấy - Cầu Giấy - Hà Nội), post type (1崙 trước), salary (51000 VND) 1 giờ, work hours (2016-11-28), and number of workers (5). The poster is Phan Hoàng Yên.

MÔ TẢ CÔNG VIỆC

Bán hàng, thực hiện các chỉ tiêu doanh số được giao. Thực hiện các bước trong quy trình bán hàng giới thiệu, tư vấn và bán các sản phẩm cho khách hàng. Làm việc theo ca. Thường xuyên kiểm tra cửa hàng, quản lý số lượng hàng hóa, quản lý tài sản, tiền bán hàng, hóa đơn, chứng từ, trang thiết bị được cung cấp, giữ gìn vệ sinh cửa hàng, vệ sinh sản phẩm. Tham dự các lớp huấn luyện khi được chỉ định. Thực hiện việc mặc đồng phục, bảo quản đồng phục theo quy định. Luôn duy trì và truyền tải hình ảnh đẹp của sản phẩm, của công ty tới khách hàng. Các công việc khác liên quan sẽ được giao trong quá trình làm việc.

YÊU CẦU KHÁC

Ejob team

6.2.1.5 Online chat with Admin

The screenshot shows an online chat window between a user named "lyhainamfpt" and an admin. The admin's message reads: "Chào eJob Team thân yêu". The user's message reads: "Bạn (đổi tên)" and "eJob Team có thể giúp gì cho bạn?". Below the chat, a yellow box highlights the slogan "Tìm việc mơ ước - Nâng bước tương lai".

6.2.2 Member メンバー

6.2.2.1 Common

6.2.2.1.1 Login

To login, the user do the steps below:

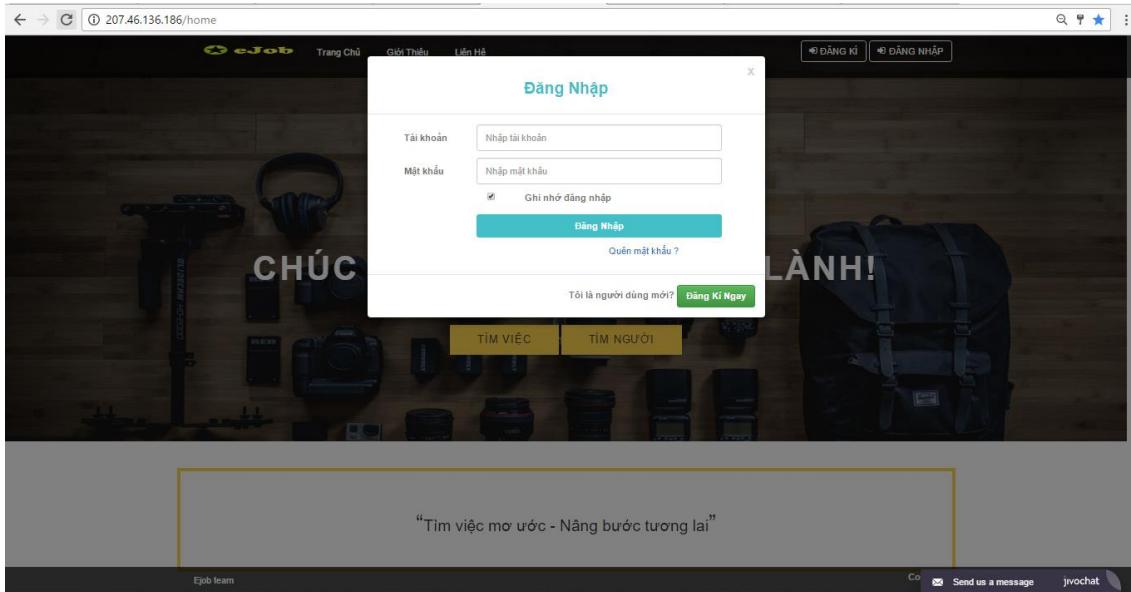
Step 1: Open browser and go to <http://207.46.136.186/home>, then Homepage is displayed.

Step 2: Click "Đăng nhập" button in header, Login popup displayed.

Step 3: User Login with activated JFS' account:

Step 3.1: User enter username and password

Step 3.2: Click on Login button



6.2.2.1.2 Logout

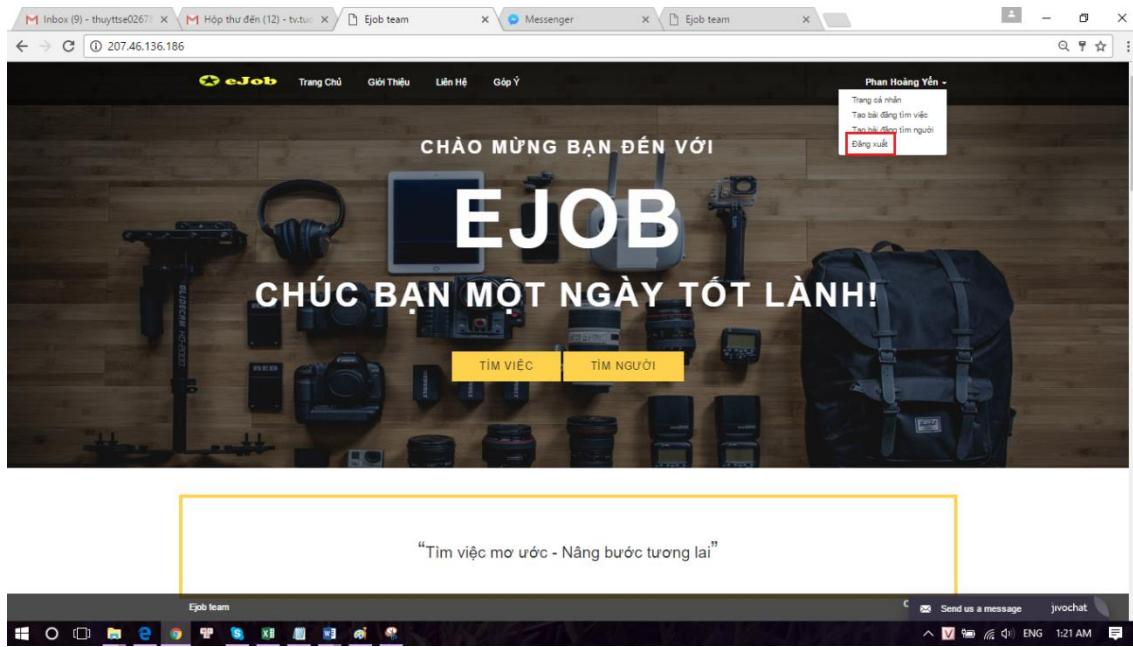
To logout account on website, the users do the steps below:

Step 1: Open browser and go to: <http://207.46.136.186/home>, then Homepage is displayed.

Step 2: Login to website.

Step 3: Move the cursor to Account Name at the top of screen.

Step 4: Click on “Đăng xuất”



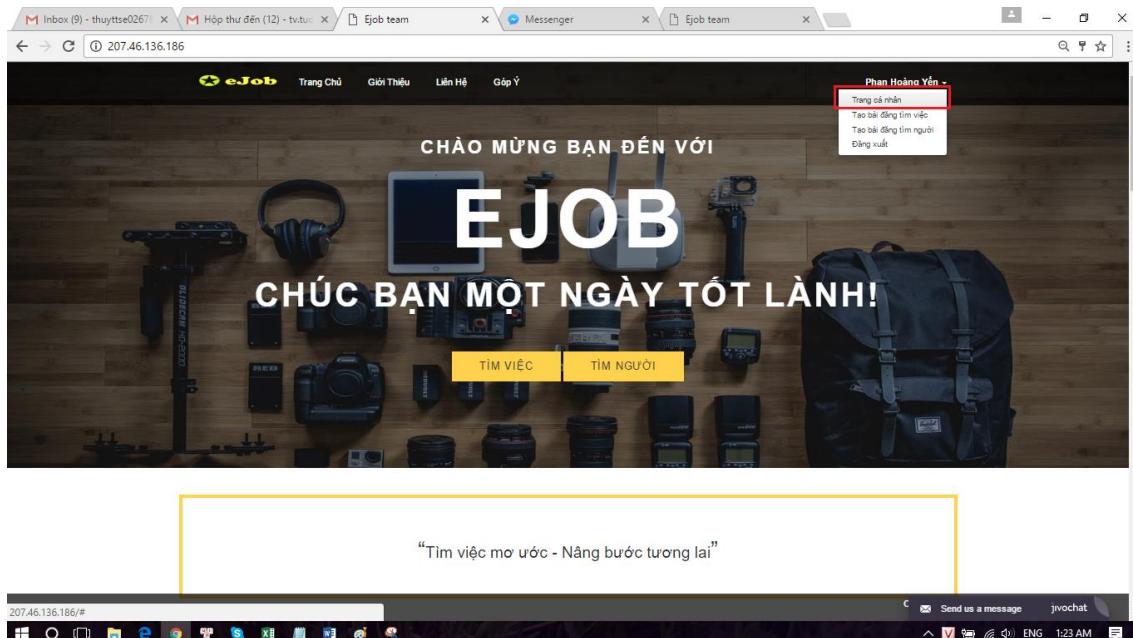
6.2.2.1.3 Change password

To change password, User do the steps below:

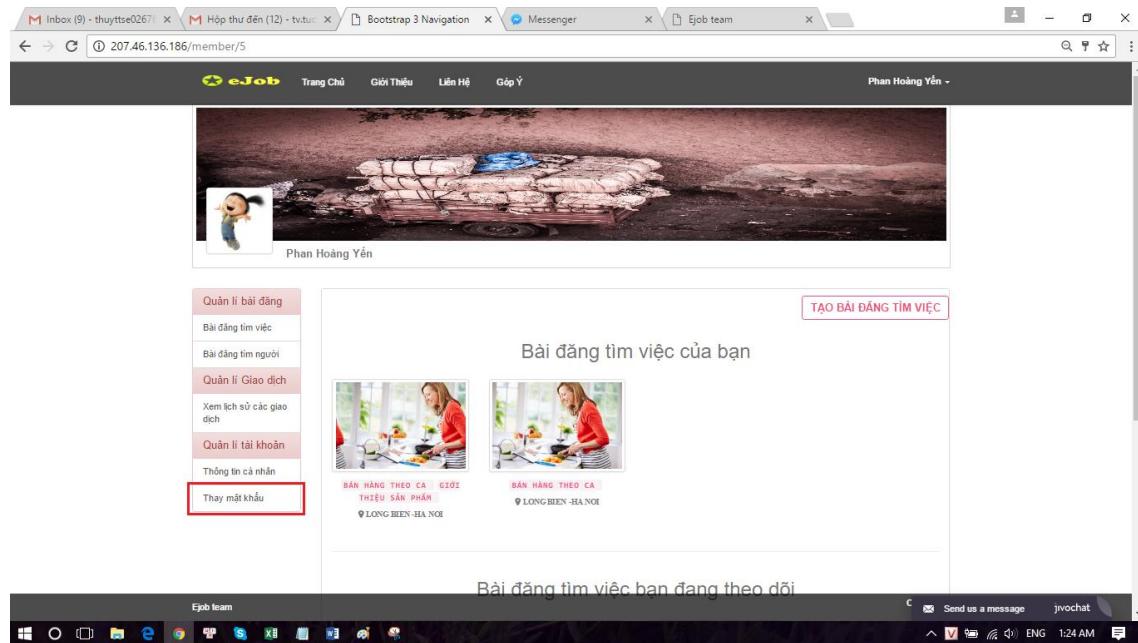
Step 1: Open browser and go to: <http://207.46.136.186/home>, then Homepage is displayed.

Step 2: Login

Step 3: Click on “Trang cá nhân”.



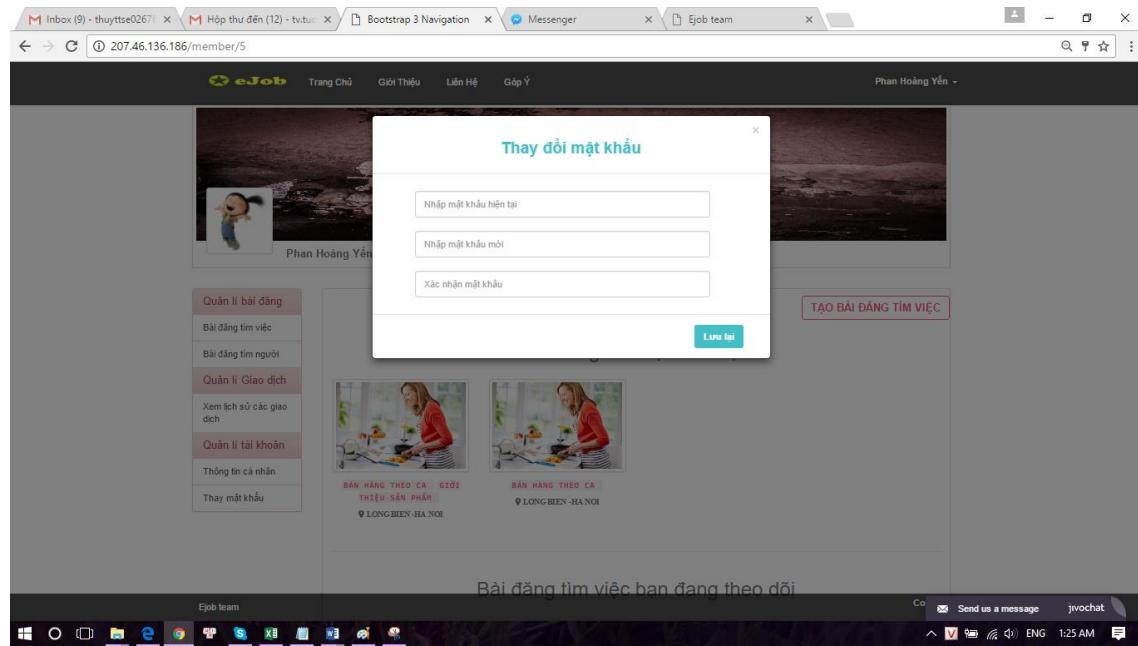
Step 4: Click on “Thay mật khẩu”



Step 5: Change password popup is displayed.

Step 6: Enter current password, new password and confirm password.

Step 7: Click “Lưu lại” button to save new password.



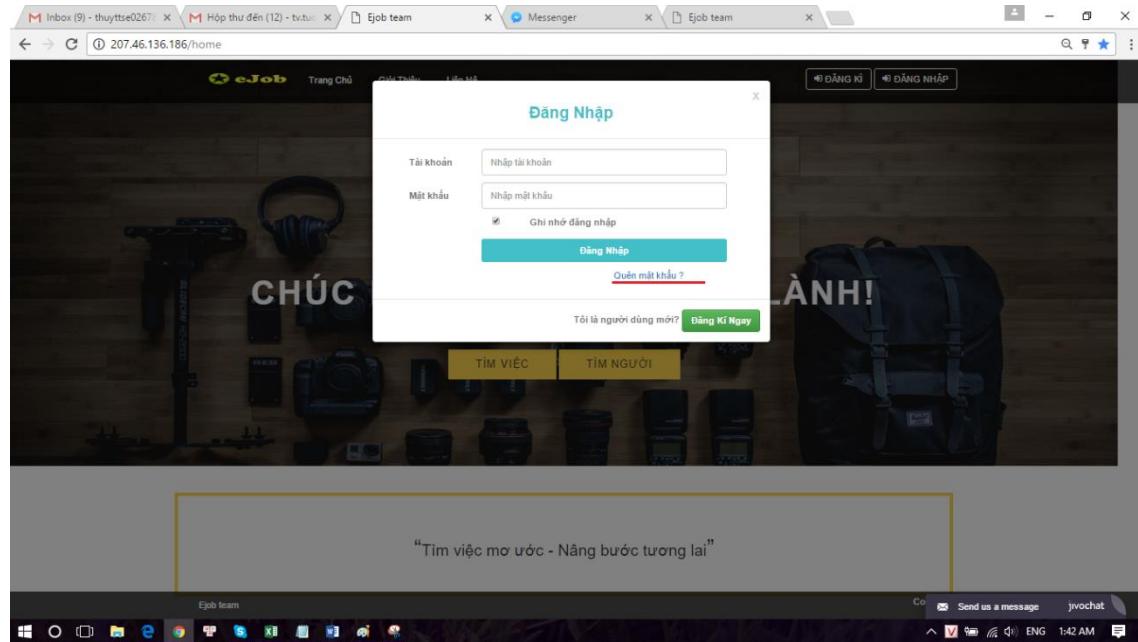
6.2.2.1.4 Forgot password

To receive password when forgot, the User do the steps below:

Step 1: Open browser and go to: <http://207.46.136.186/home>, then Homepage is displayed.

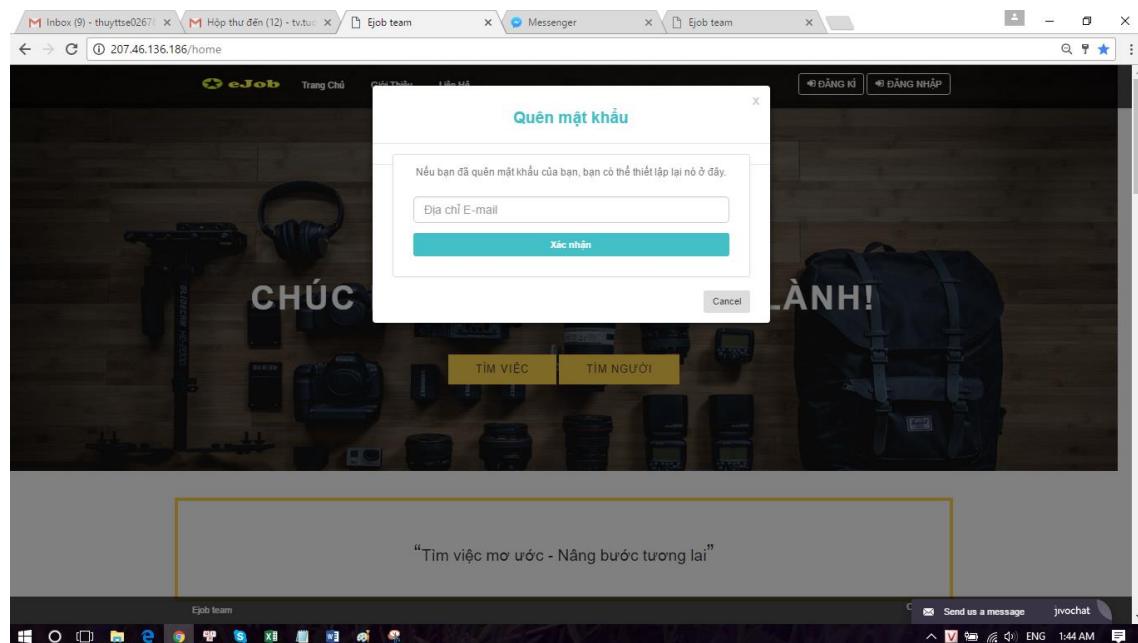
Step 2: Click “Đăng nhập” button.

Step 3: Click “Quên mật khẩu” link.

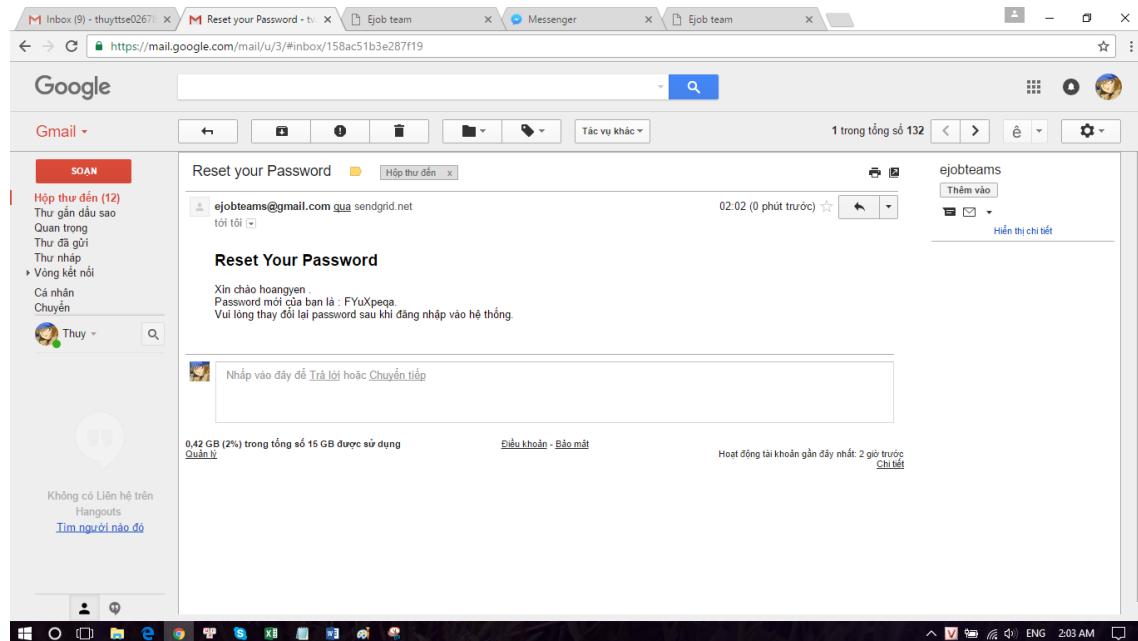


Step 4: Enter registered email in “Địa chỉ E-mail” text box.

Step 5: Click on “Xác nhận” button.



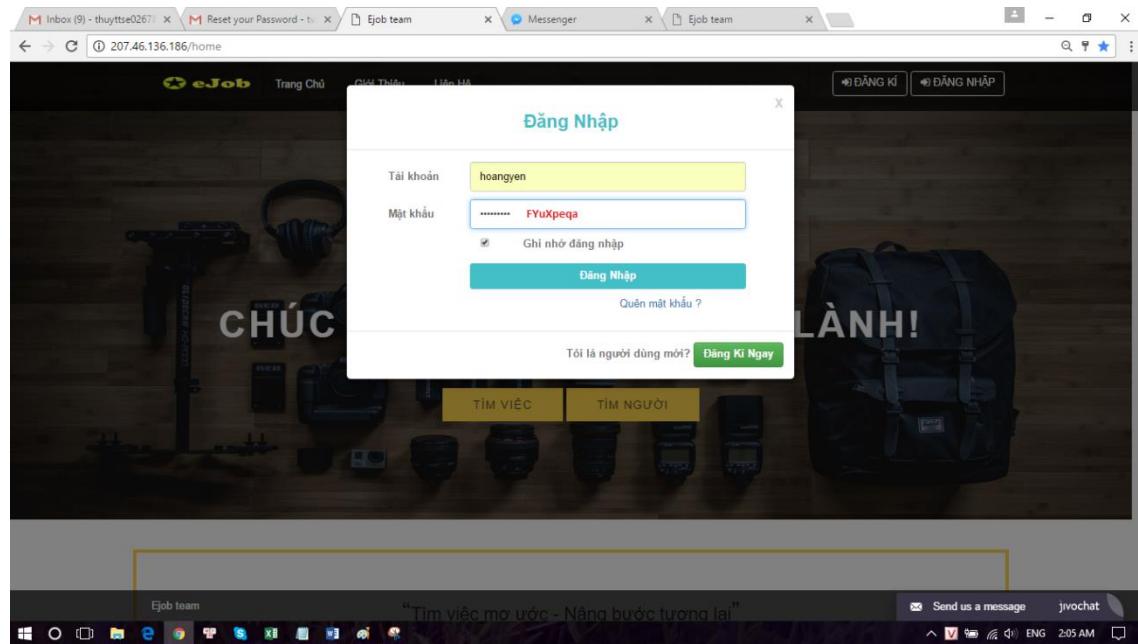
Step 6: Check received mail to see password.



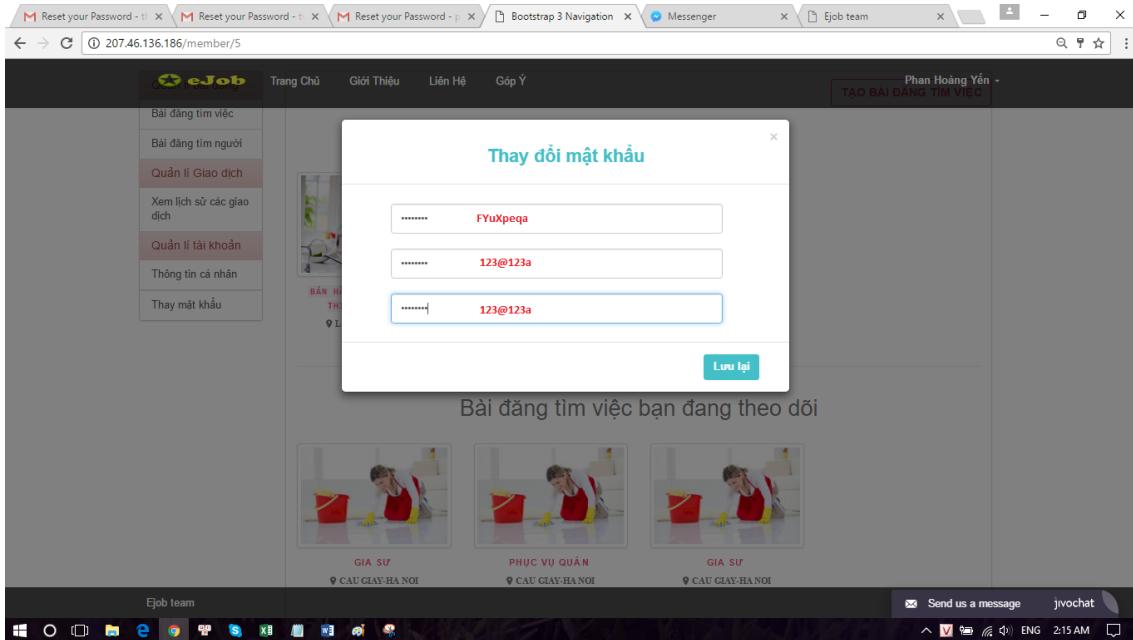
Step 7: Go to eJob homepage

Step 8: Click “Đăng nhập” button on header.

Step 9: Enter account name and new password.



Step 10: Change password. Follow Member change password



6.2.2.1.5 Search and view Employer's Posts

Follow Guest – Search and view Emplpyer's Post

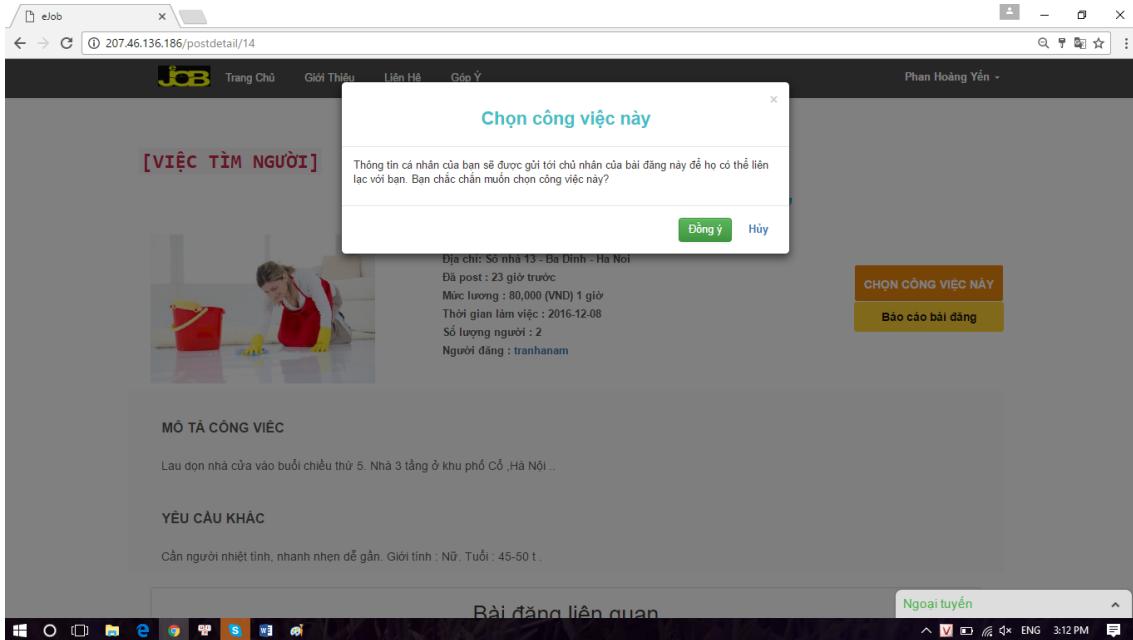
6.2.2.1.6 Search and view Job Seeker's Posts

Follow Guest – Search and view Job Seeker's Post

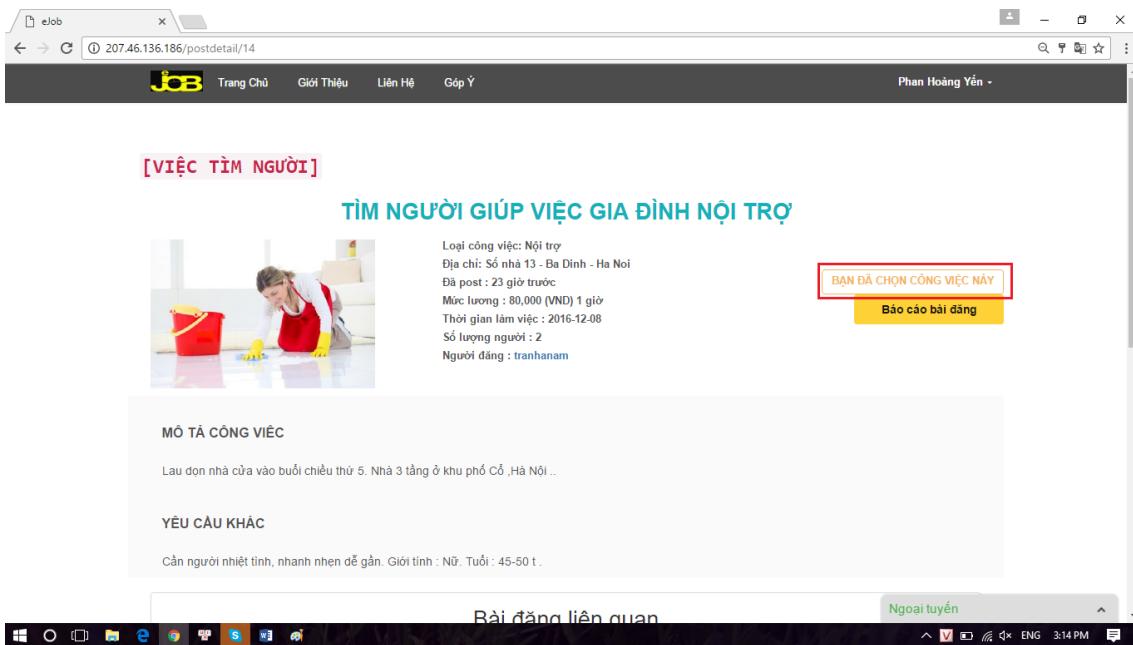
6.2.2.1.7 Apply suitable Job

Step 1: View detail Employer's Post

Step 2: Click button “Chọn công việc này”. Confirm popup is displayed.



Step 3: Click button “Đồng ý” then button “Chọn công việc này” change to “Bạn đã chọn công việc này”

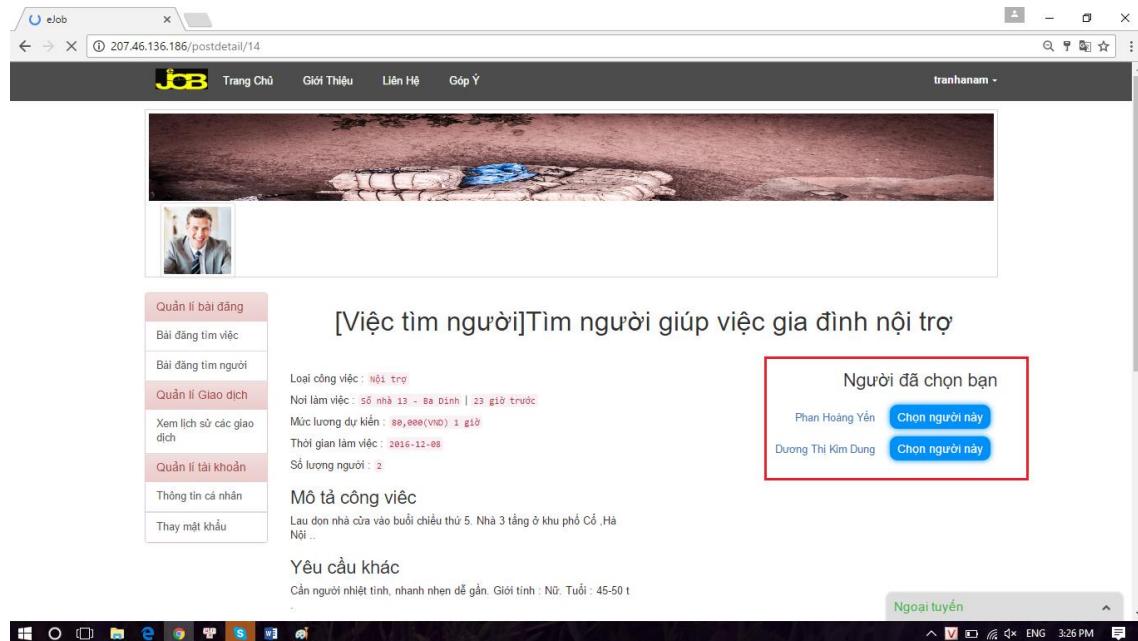


6.2.2.1.8 Pay to get hidden Profile of Job Seekers

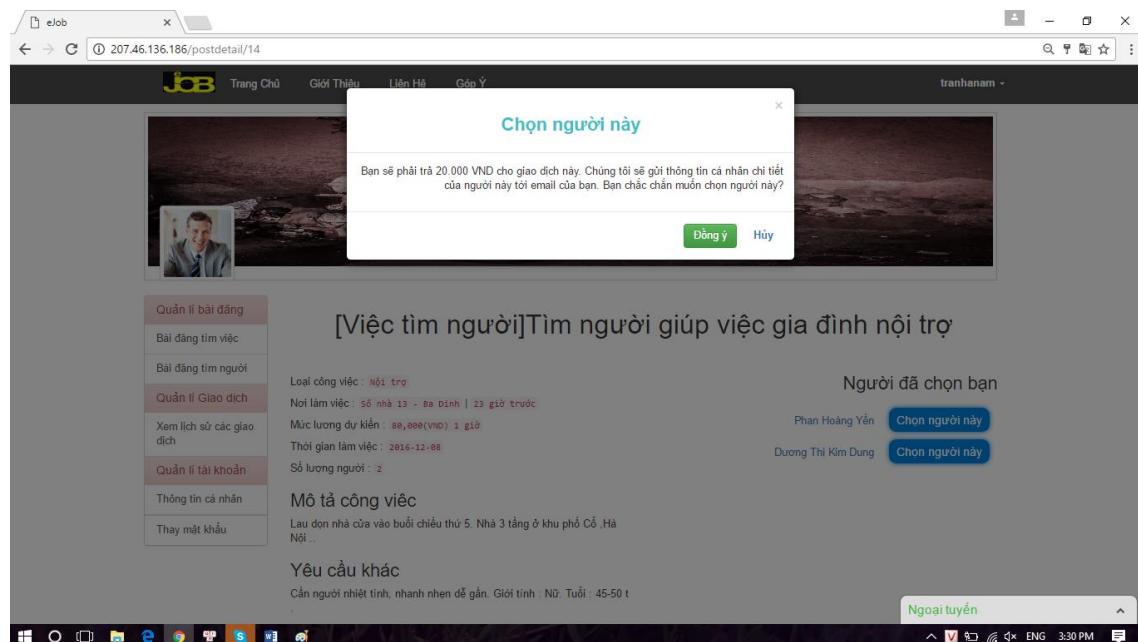
There are 2 ways to view Job Seeker's information:

Way 1: From the displaying of the Members who had chosen Employer's Post, go to see their information.

Step 1: Click button “Chọn người này”

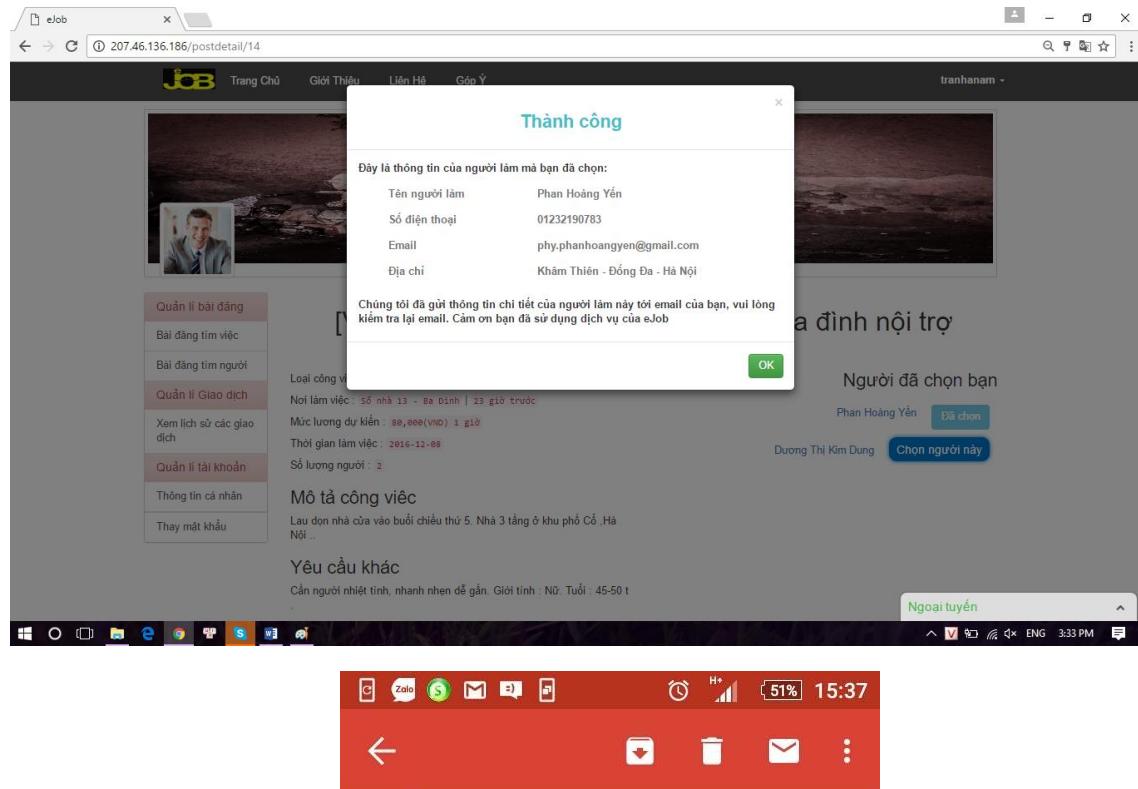


Step 2: Popup confirm is displayed.



Step 3: Click button “Đồng ý”

Step 4: Hidden information of Job Seeker is displayed and also be sent to Employer’s email.



eJob gửi thông tin cá nhân

Hộp thư đến

 ejobteams@gmail.com

tới tôi

15:33 Xem chi tiết

eJob xin gửi thông tin cá nhân

Tên người làm : Phan Hoàng Yến
Email : phy.phanhoangyen@gmail.com

SĐT : 01232190783

Địa chỉ : Khâm Thiên - Đống Đa - Hà Nội

Cảm ơn bạn đã sử dụng dịch vụ của chúng tôi.

Trả lời

Trả lời tất cả

Chuyển tiếp



Way 2: Follow 2.2.1.12 Get suitable Job Seeker.

6.2.2.1.9 Get suitable Job Seeker

Step 1: View detail Job Seeker's Post

The screenshot shows a web browser window for 'eJob' at the URL 207.46.136.186/postdetail/29. The page displays a job listing titled 'GIAO HÀNG TẬN NHÀ HOẶC CHUYỂN ĐỒ, CHUYỂN NHÀ' (Delivery or Moving Services). The post details include:

- Loại công việc:** Chuyển đồ
- Địa chỉ:** Hoàn Kiếm - Hà Nội |
- Đã post:** 2 phút trước
- Mức lương mong muốn:** 50,000 (VND) 1 giờ
- Thời gian làm việc:** 2016-12-11
- Người đăng:** tranhanam

On the right side, there are two buttons: 'CHỌN NGƯỜI NÀY' (Select Person) and 'Bảo cáo bài đăng' (Report Post).

MÔ TẢ CÔNG VIỆC

Giao hàng. Chuyển đồ, chuyển nhà theo yêu cầu..

MÔ TẢ BẢN THÂN

Nhanh nhẹn và cẩn thận. Thông thạo đường đi ở Hà Nội. Có kinh nghiệm chuyển đồ, giao hàng..

At the bottom of the browser window, the taskbar shows icons for various applications like File Explorer, Internet Explorer, and Task Manager.

Step 2: Click button “Chọn người này” then popup confirm is displayed.

The screenshot shows the same job listing page after clicking the 'CHỌN NGƯỜI NÀY' button. A modal dialog box titled 'Chọn người này' (Select Person) is displayed in the center. The message inside the dialog reads: 'Bạn sẽ phải trả 10.000 VND cho giao dịch này. Chúng tôi sẽ gửi thông tin cá nhân chi tiết của người này tới email của bạn ngay khi họ chấp nhận. Bạn chắc chắn muốn chọn người này?' (You will have to pay 10,000 VND for this transaction. We will send the personal information of this person to your email as soon as they accept. Do you want to select this person?).

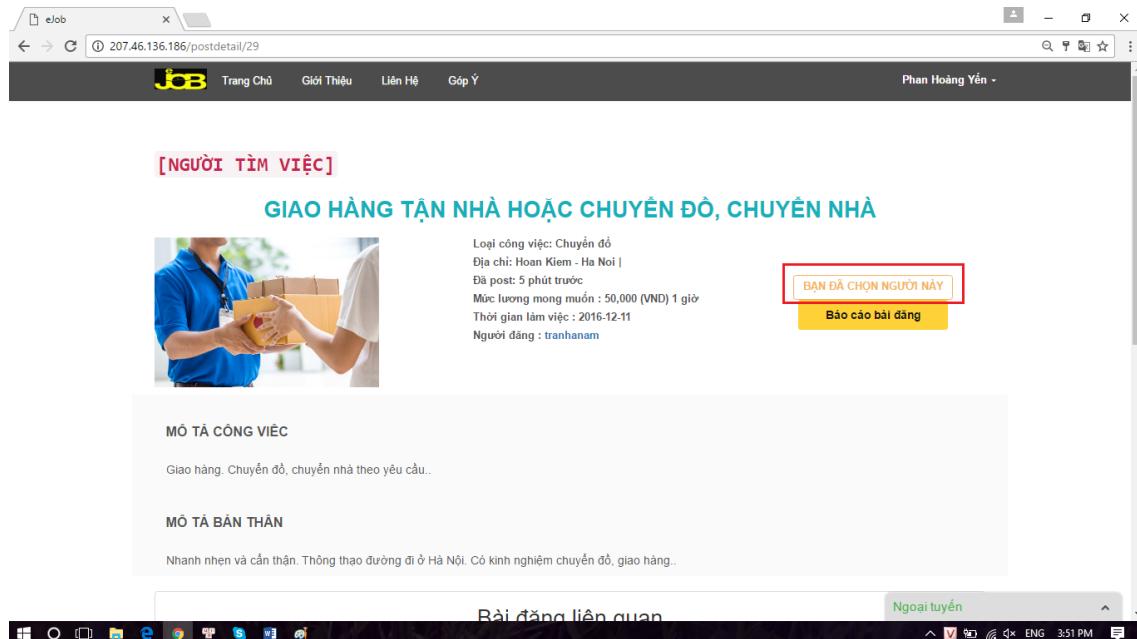
At the bottom of the dialog are two buttons: 'Đồng ý' (Agree) and 'Hủy' (Cancel).

Below the dialog, the original job listing is partially visible again.

At the bottom of the browser window, the taskbar shows icons for various applications like File Explorer, Internet Explorer, and Task Manager.

Step 3: Click button “Đồng ý” then button “Chọn người này” changes to “Bạn đã chọn người này”.

Step 4: Send notify mail to Job Seeker's email.



The screenshot shows a job listing for "GIAO HÀNG TẬN NHÀ HOẶC CHUYỂN ĐỔ, CHUYỂN NHÀ". The job details include: Type of work: Chuyển đồ; Location: Hoan Kiem - Ha Noi; Post time: 5 minutes ago; Desired salary: 50,000 VND/hour; Work period: 2016-12-11; Poster: tranhanam. Buttons shown are "BẠN ĐÃ CHỌN NGƯỜI NÀY" (You have chosen this person) and "Bảo cáo bài đăng" (Report post).

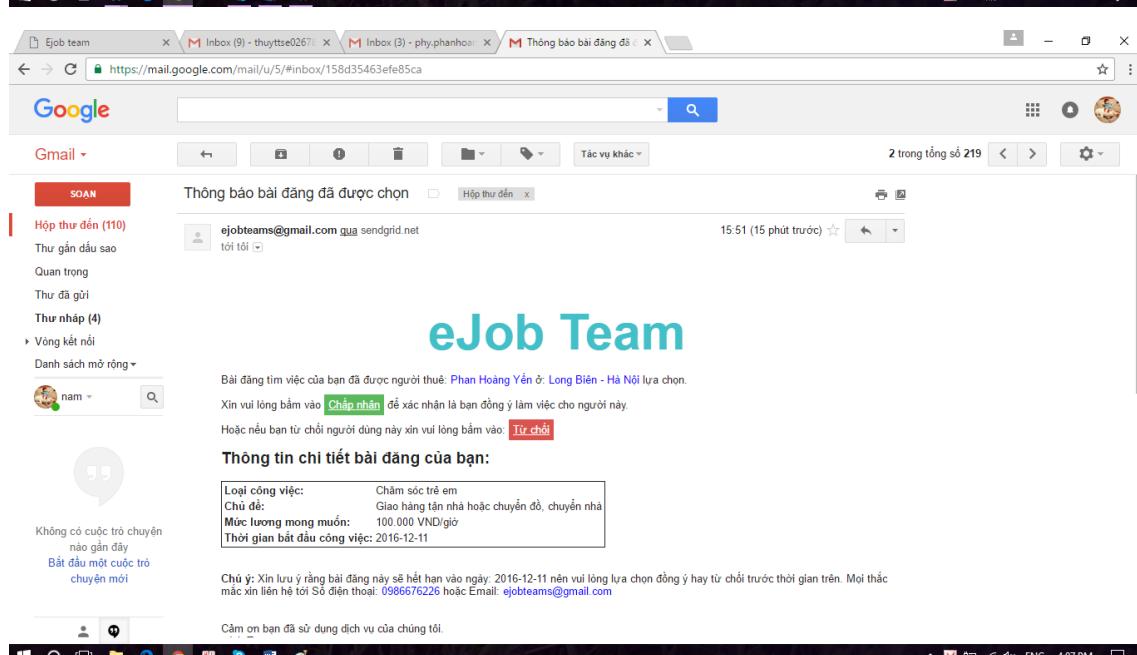
MÔ TẢ CÔNG VIỆC

Giao hàng. Chuyển đồ, chuyển nhà theo yêu cầu..

MÔ TẢ BẢN THÂN

Nhanh nhẹn và cẩn thận. Thông thạo đường đi ở Hà Nội. Có kinh nghiệm chuyển đồ, giao hàng..

Bài đăng liên quan



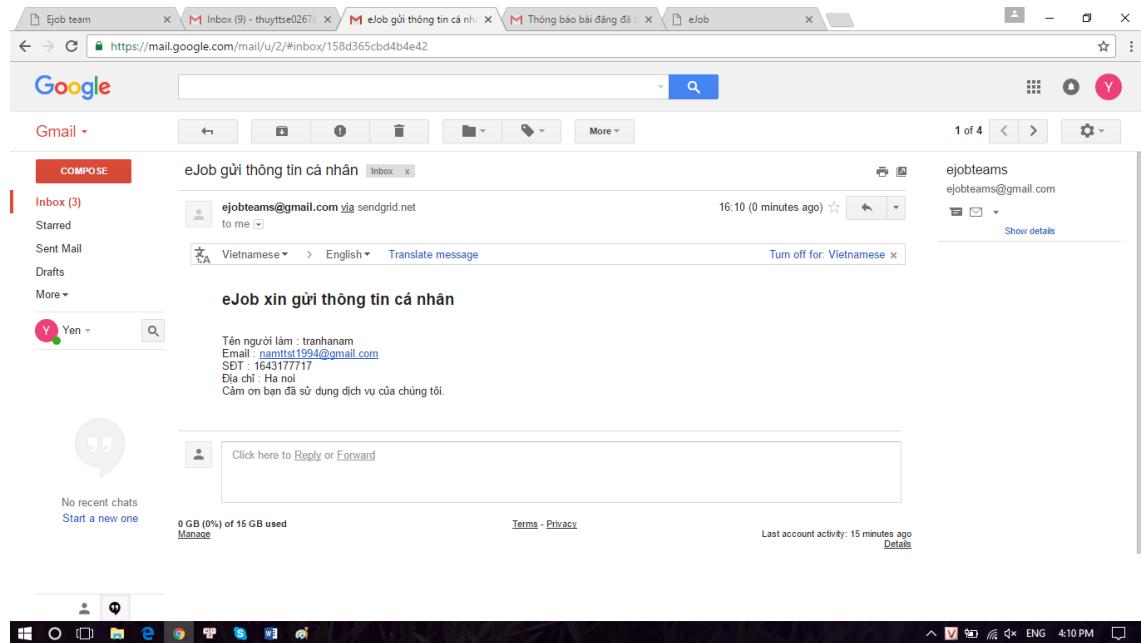
The email subject is "Thông báo bài đăng đã được chọn". The message body says: "Bài đăng tìm việc của bạn đã được người thuê: Phan Hoàng Yến ở: Long Biên - Hà Nội lựa chọn. Xin vui lòng bấm vào [Chấp nhận](#) để xác nhận là bạn đồng ý làm việc cho người này. Hoặc nếu bạn từ chối người dùng này xin vui lòng bấm vào [Từ chối](#)". Below this is a summary table:

Loại công việc:	Chăm sóc trẻ em
Chủ đề:	Giao hàng tận nhà hoặc chuyển đồ, chuyển nhà
Mức lương mong muốn:	100.000 VND/giờ
Thời gian bắt đầu công việc:	2016-12-11

Notes at the bottom: "Chú ý: Xin lưu ý rằng bài đăng này sẽ hết hạn vào ngày: 2016-12-11 nên vui lòng lựa chọn đồng ý hay từ chối trước thời gian trên. Mọi thắc mắc xin liên hệ tới Số điện thoại: 0986676226 hoặc Email: ejobteams@gmail.com"

Step 5: Job Seeker click button “Chấp nhận”

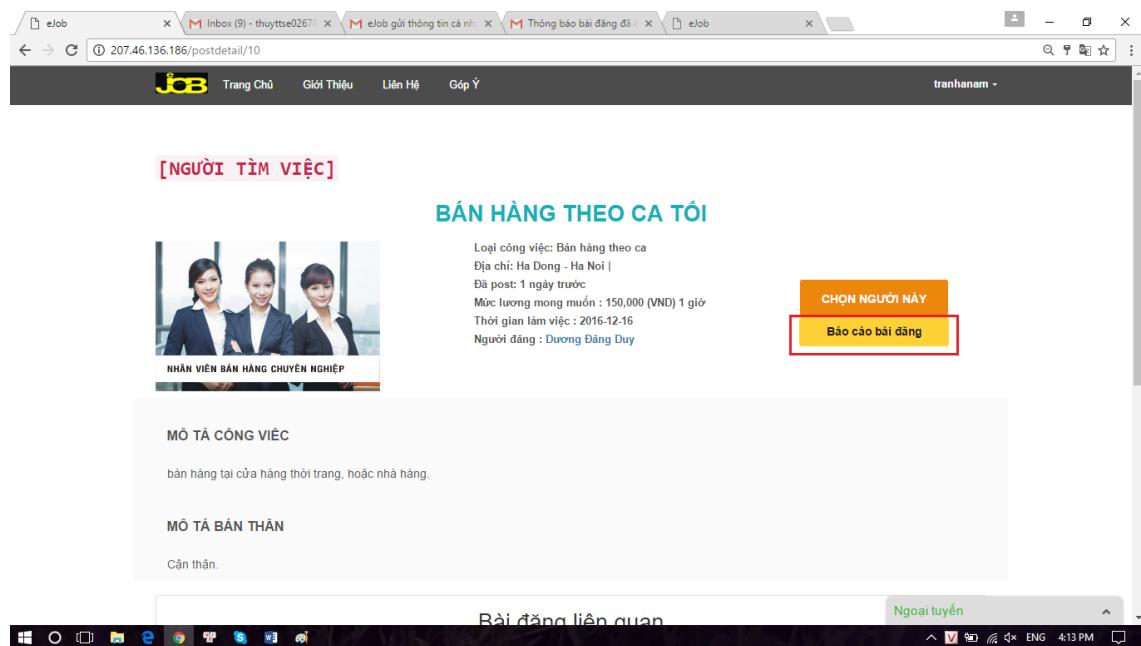
Step 6: System send Job Seeker's information to Employer's email.



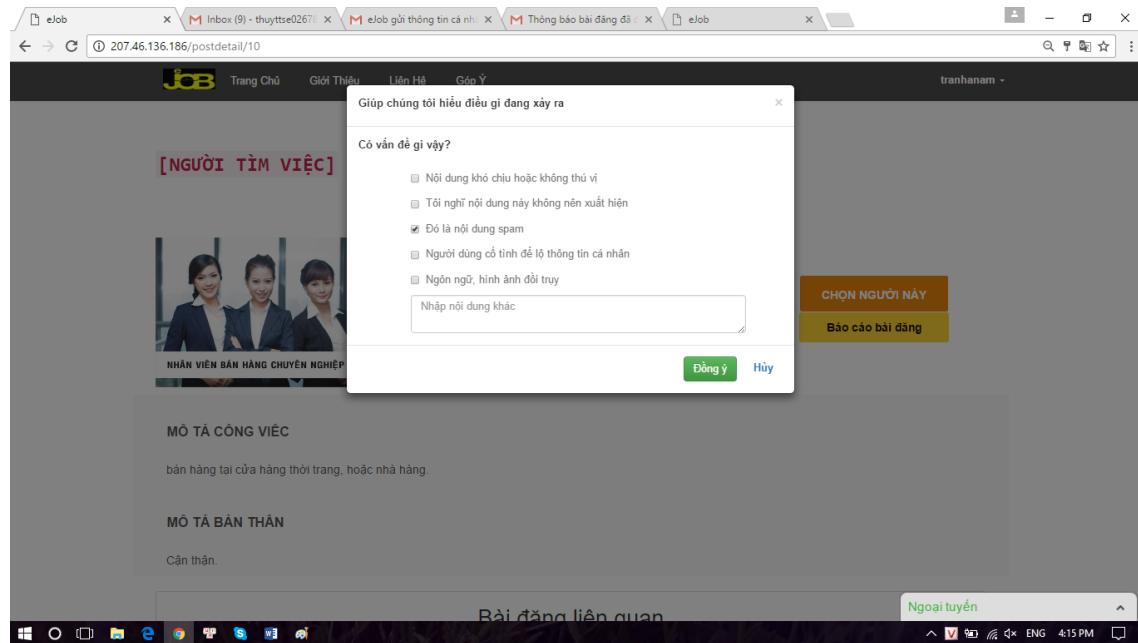
6.2.2.1.10 Report other Member's Post

Step 1: View detail Post

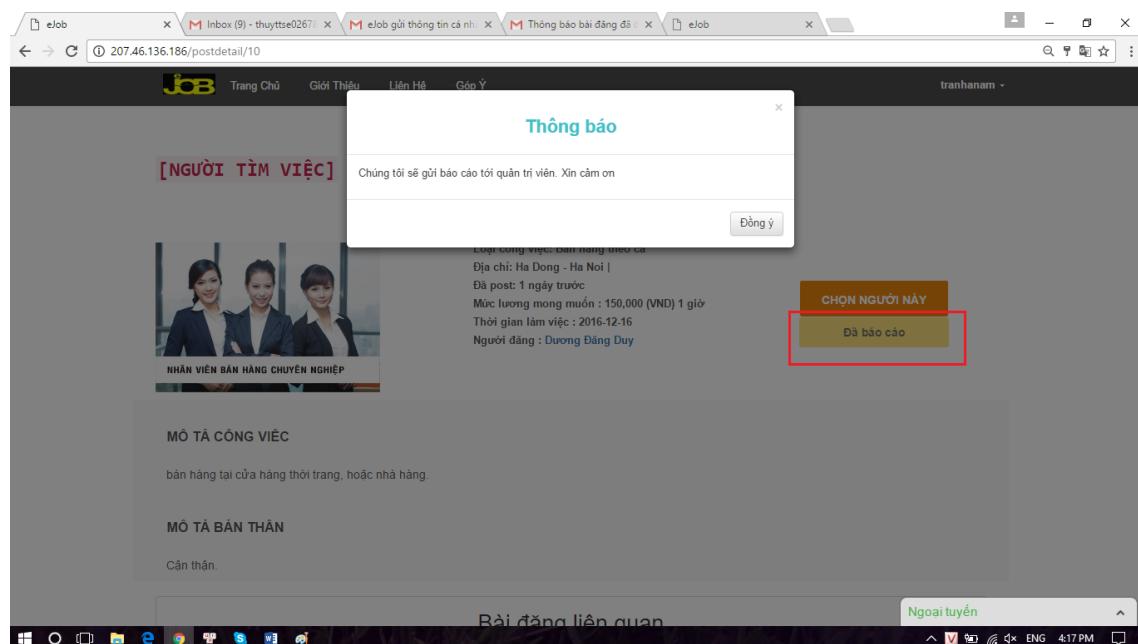
Step 2: Click button “Báo cáo bài đăng”



Step 3: Choose report's reason.

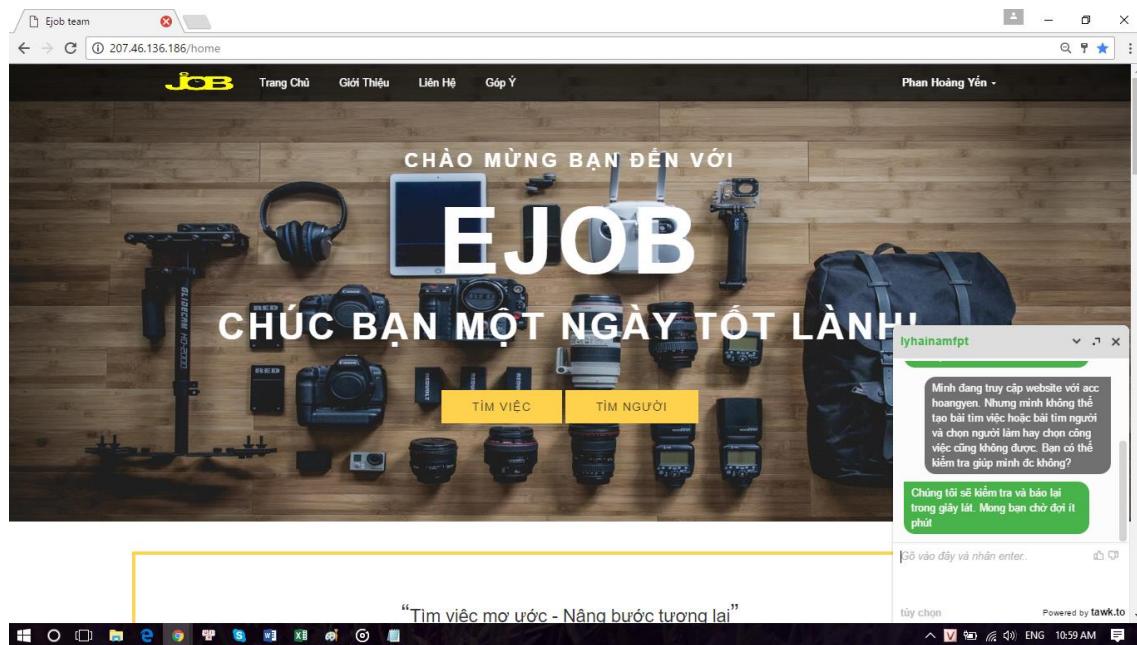


Step 4: Click button “Đồng ý”



Step 5: Popup notify is displayed and button “Báo cáo bài đăng” change to “Đã báo cáo”.

6.2.2.1.11 Online chat with Admin

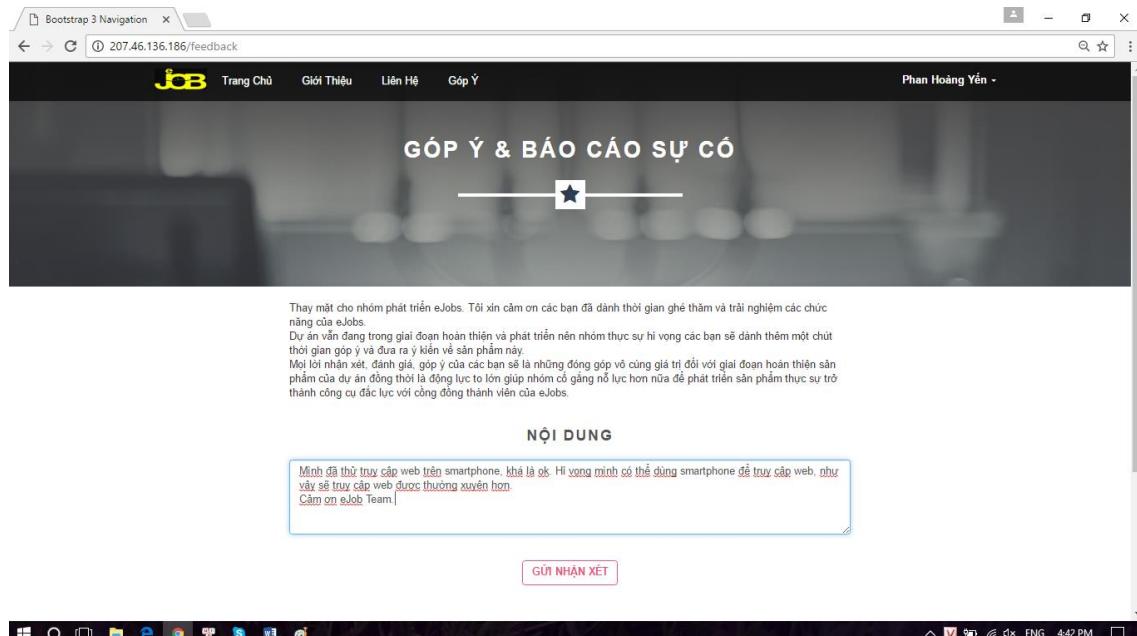


6.2.2.1.12 Feedback

Step 1: Open browser and go to <http://207.46.136.186/home>, then Homepage is displayed.

Step 2: Click link “Góp ý” on header.

Step 4: Input content of feedback to textbox.



Step 5: Click button “Gửi nhận xét”

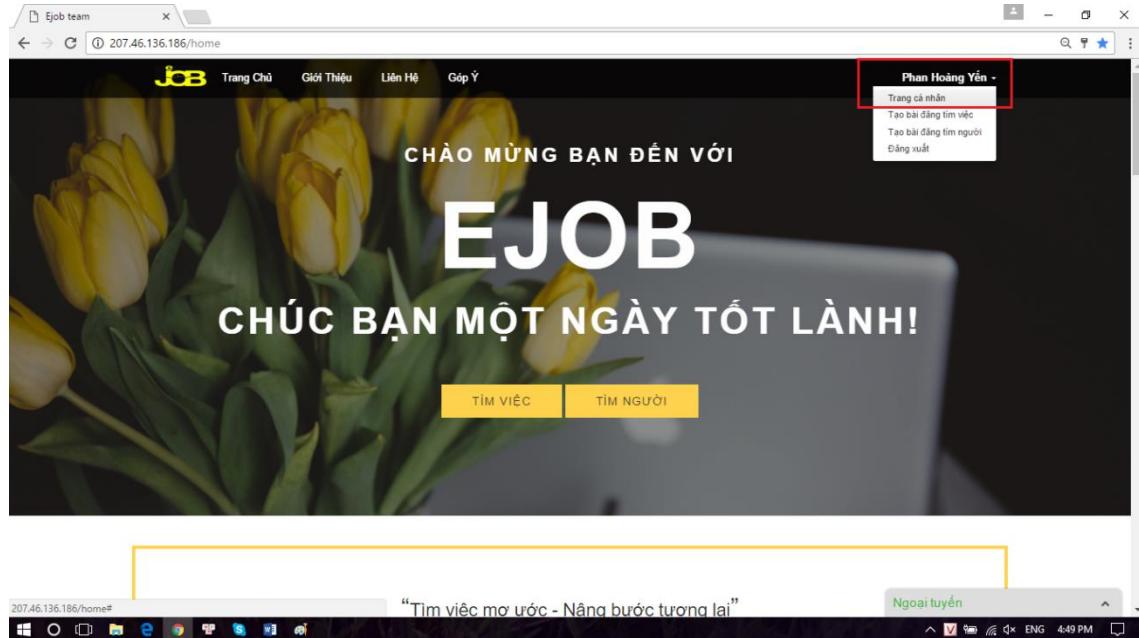
6.2.2.2 Manage Personal Page

6.2.2.2.1 View Personal Page

Step 1: Open browser go to: <http://207.46.136.186/home>, then Homepage is displayed.

Step 2: Login

Step 3: Move the cursor to Account name at top of screen, and then click link “Trang cá nhân”



Step 4: Personal Page is displayed.

6.2.2.2.2 View and edit Profile

Step 1: View Profile

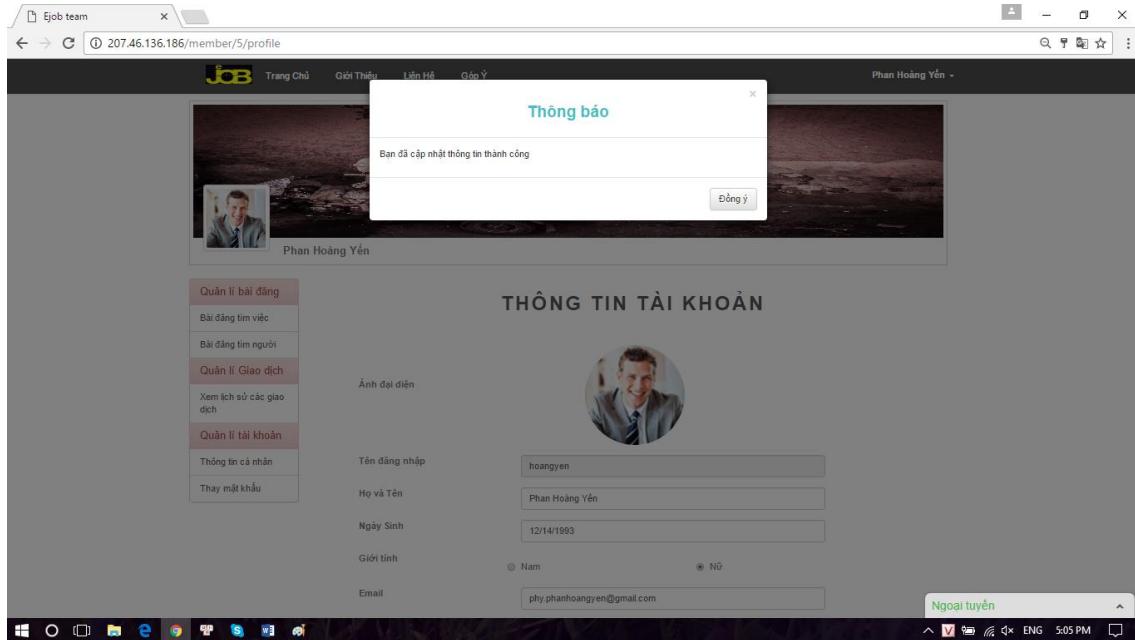
Step 2: Click on link “Thông tin cá nhân”

The screenshot shows a web browser window with the URL 207.46.136.186/member/5/EmployerPosts. The page title is "Bootstrap 3 Navigation". The main content area displays a profile for "Phan Hoàng Yến" with a placeholder image. On the left, a sidebar menu lists several options: Quản lý bài đăng, Bài đăng tìm việc, Bài đăng tìm người, Quản lý Giao dịch, Xem lịch sử các giao dịch, Quản lý tài khoản, Thông tin cá nhân (which is highlighted with a red box), and Thay mật khẩu. Below the sidebar, there are two small images: one of a dog and another of a child at a desk. A button labeled "TẠO BÀI ĐĂNG TÌM NGƯỜI" is visible. At the bottom of the page, a message says "Bạn chưa quan tâm bài đăng nào." and a green "Ngoại tuyển" button is present. The taskbar at the bottom shows various application icons.

Step 3: Personal information page is displayed.

The screenshot shows a web browser window with the URL 207.46.136.186/member/5/profile. The page title is "Ejob team". The main content area displays a "THÔNG TIN TÀI KHOẢN" section. It includes a placeholder image for "Ảnh đại diện" (Profile Picture). Form fields for "Tên đăng nhập" (Login Name) containing "hoangyen", "Họ và Tên" (Full Name) containing "Phan Hoàng Yến", and "Ngày Sinh" (Date of Birth) containing "12/05/2016". Under "Giới tính" (Gender), there are radio buttons for "Nam" and "Nữ", with "Nam" selected. Form fields for "Email" (Email) containing "phy.phanhoangyen@gmail.com" and "Số điện thoại" (Phone Number) containing "01232190783". The "Địa chỉ" (Address) field contains "Khâm Thiên - Đông Đa - Hà Nội". A large text area for "Mô tả bản thân" (Self Description) is empty. A yellow "CẬP NHẬT THÔNG TIN" (Update Information) button is located at the bottom. The taskbar at the bottom shows various application icons.

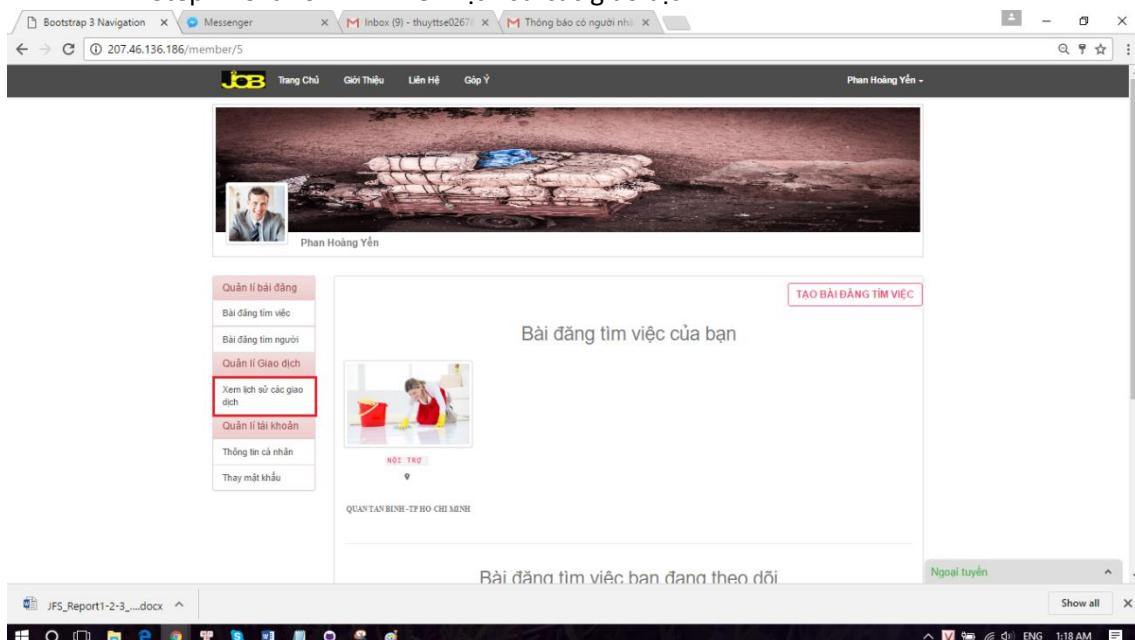
Step 5: Change information then click button “Cập nhật thông tin”



6.2.2.2.3 View History Transaction

Step 1: View Personal Post

Step 2: Click on link "Xem lịch sử các giao dịch"



Step 3: History Transaction page is displayed.

The screenshot shows a web browser window with multiple tabs open. The active tab displays a member's profile for 'Phan Hoàng Yến'. The profile includes a small photo of a man, the member's name, and a banner image of a person working at a desk. On the left, a sidebar menu lists several options: 'Quản lý bài đăng' (selected), 'Bài đăng tìm việc', 'Bài đăng tìm người', 'Quản lý Giao dịch' (selected), 'Xem lịch sử các giao dịch', 'Quản lý tài khoản', 'Thông tin cá nhân', and 'Thay mật khẩu'. The main content area shows a table titled 'Xem lịch sử giao dịch' with the following data:

Tác vụ	Giao dịch với	Số tiền (VND)	Thời gian	Trạng thái
Tôi cần một công việc như nội trợ hoặc chăm sóc trẻ em	dangduy	0	2016-12-06 18:23:50	Đang chờ
Tôi cần một công việc như nội trợ hoặc chăm sóc trẻ em	dangduy	0	2016-12-06 18:23:47	Đang chờ
Nấu ăn Ăn Ăn	dangduy	0	2016-12-06 18:23:31	Đang chờ
Tôi cần một công việc như nội trợ hoặc chăm sóc trẻ em	dangduy	0	2016-12-06 18:22:47	Đang chờ
Giá sư tiếng Nhật trình độ NS	vietvq	0	2016-12-06 18:18:33	Đang chờ
Chăm sóc trẻ em	dangduy	0	2016-12-06 18:17:53	Đang chờ
Giao hàng tận nhà hoặc chuyển đồ, chuyển nhà	tranhanam	10,000	2016-12-06 08:51:11	Thành công

6.2.2.3 View other Member's Personal Page

6.2.2.3.1 View other Member's Posts

Step 1: View detail Post

Step 2: Click on name of Member who posted this post.

The screenshot shows a detailed post from a member named 'Dương Đăng Duy'. The post is titled 'BÁN HÀNG THEO CA TỐI' (Sell goods by shift). It includes a photo of three women in business attire. The post details are as follows:

- Loại công việc: Bán hàng theo ca
- Địa chỉ: Hà Đông - Hà Nội | Đã post: 1 ngày trước
- Mức lương mong muốn: 150.000 (VND) 1 giờ
- Thời gian làm việc: 2016-12-16
- Người đăng: Dương Đăng Duy

Below the post, there are sections for 'MÔ TẢ CÔNG VIỆC' (Job Description) and 'MÔ TẢ BẢN THÂN' (Self Description), both containing placeholder text. A 'Bài đăng liên quan' (Related posts) section is also visible at the bottom.

Step 3: Go to other member's Personal Post page

Dương Đăng Duy

Bài đăng tìm việc của Dương Đăng Duy

Chăm sóc trẻ em
Ba Đình-Hà Nội

Chăm sóc trẻ em
Quận 1-Tp Hồ Chí Minh

Chăm sóc trẻ em nấu ăn
Hà Đông-Hà Nội

Nội trú
Đống Đa-Hà Nội

6.2.2.3.2 View other Member's basic Profile

Step 1: View detail Post

Step 2: Click on name of Member who posted this post

[NGƯỜI TÌM VIỆC]

BÁN HÀNG THEO CA TỐI

Loại công việc: Bán hàng theo ca
Địa chỉ: Hà Đông - Hà Nội | Đã post: 1 ngày trước
Mức lương mong muốn : 150.000 (VND) 1 giờ
Thời gian làm việc : 2016-12-16
Người đăng : Dương Đăng Duy

CHỌN NGƯỜI NÀY

Báo cáo bài đăng

MÔ TẢ CÔNG VIỆC
bán hàng tại cửa hàng thời trang, hoặc nhà hàng.

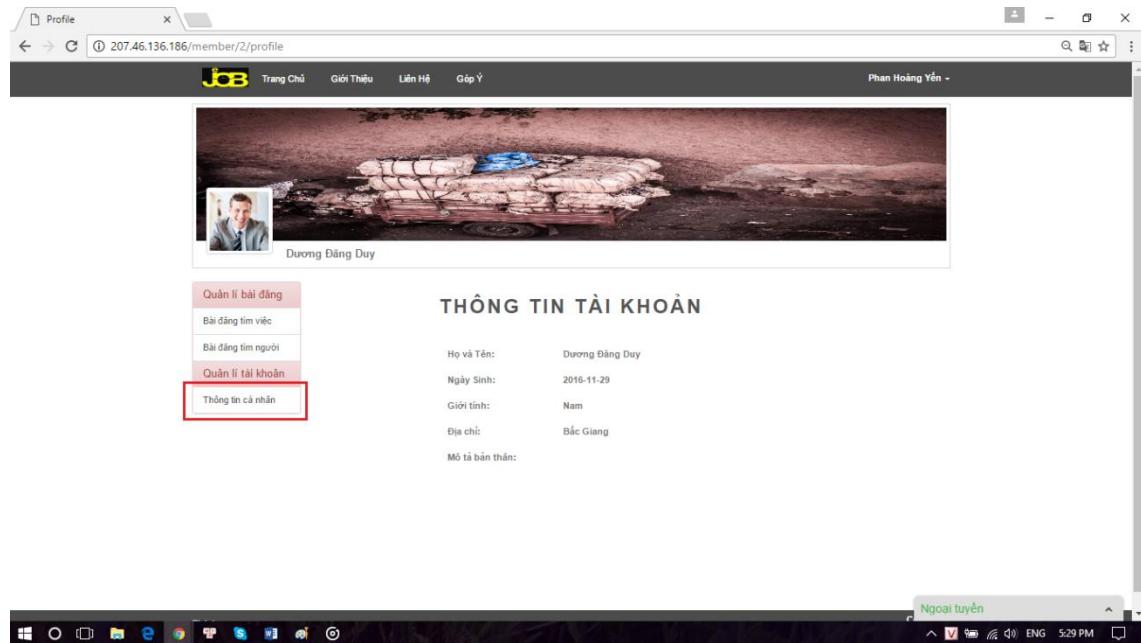
MÔ TẢ BẢN THÂN
Cận thận.

Bài đăng liên quan

Sắp xếp theo thứ tự ưu tiên
Địa chỉ → Thời gian → Mức lương

Step 3: Go to other member's Personal Post page

Step 4: Click on link “Thông tin cá nhân” then basic information of other member is displayed.



6.2.2.4 Manage Post

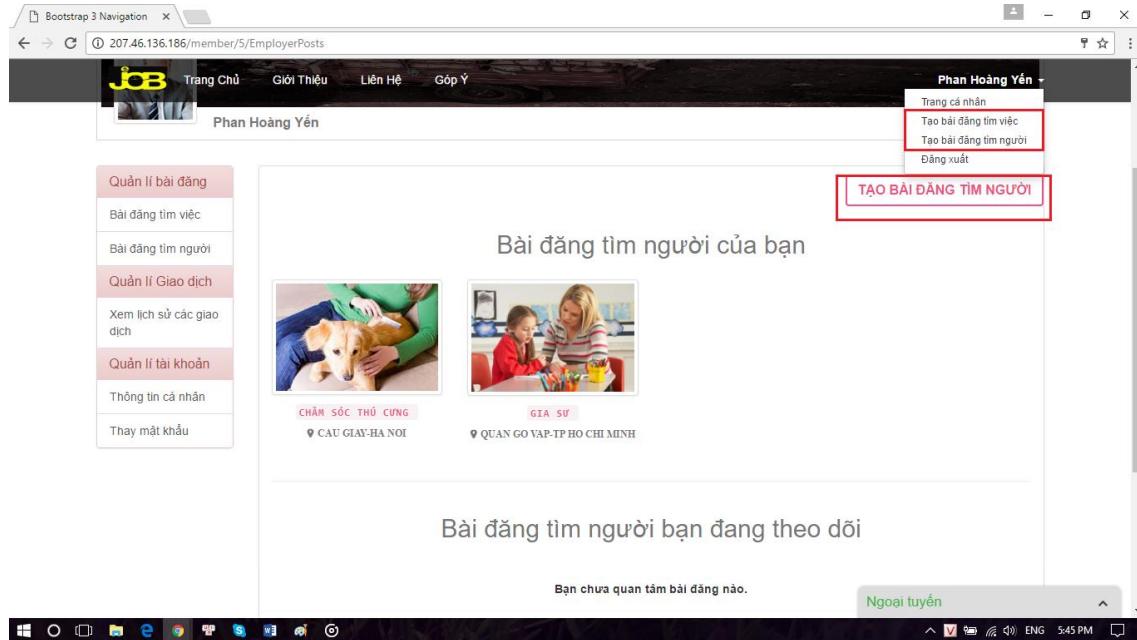
6.2.2.4.1 Create Post

These are 2 ways to create a post.

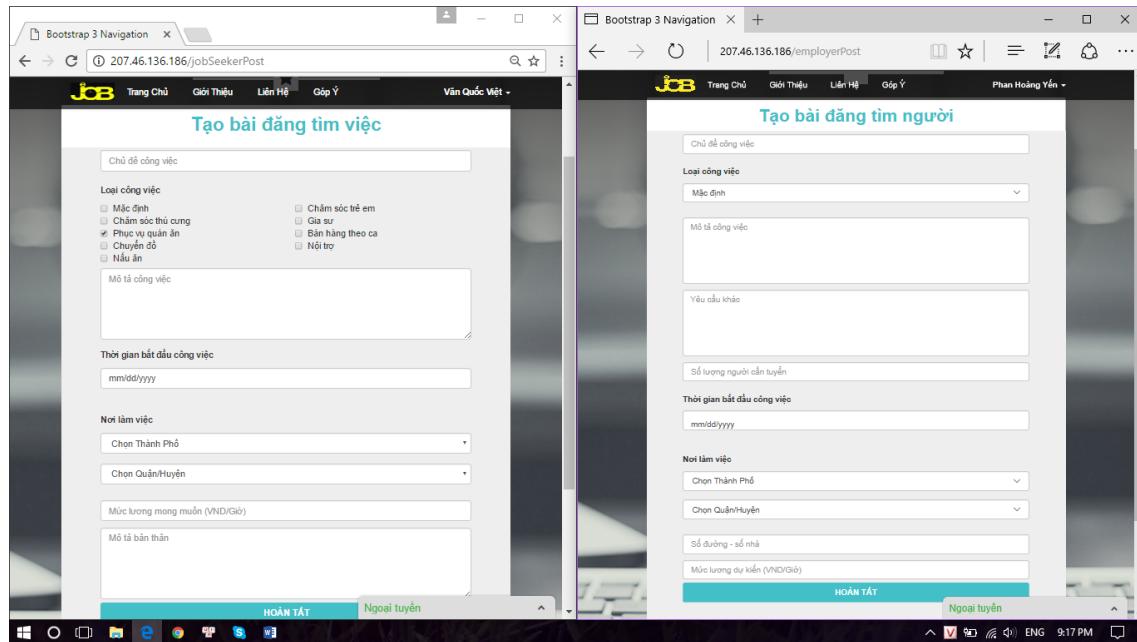
1. Step 1: Move the cursor to Member's name in header and then click link “Tạo bài đăng tìm việc” or “Tạo bài đăng tìm người”
2. Step 1: Move the cursor to Member's name in header and click link “Trang cá nhân”. Go to Personal Page.

Step 2: Click button “Bài đăng tìm việc”. Then click button “Tạo bài đăng tìm việc”

Step 3: Click on button “Bài đăng tìm người”. Then click button “Tạo bài đăng tìm người”



Step 2: Go to Create Post page



Step 3: Fill all needed information to form

Step 4: Click button “Hoàn thành”

6.2.2.4.2 Edit Post

Member can only edit own post when this post wasn't selected by anyother member.

Step 1: Go to Personal Page.

Step 2: Choose own post, and then see post detail.

The screenshot shows a web browser window for 'Bootstrap 3 Navigation' at the URL '207.46.136.186/member/5/jobSeekerPosts'. The page title is 'JOB'. The top navigation bar includes links for 'Trang Chủ', 'Giới Thiệu', 'Liên Hệ', and 'Góp Ý'. A dropdown menu for 'Phan Hoàng Yên' is open. On the left, there's a sidebar with a user profile picture and a list of menu items: 'Quản lý bài đăng' (selected), 'Bài đăng tìm việc', 'Bài đăng tìm người', 'Quản lý Giao dịch' (selected), 'Xem lịch sử các giao dịch', 'Quản lý tài khoản' (selected), 'Thông tin cá nhân', and 'Thay mật khẩu'. The main content area displays a job listing for 'CHĂM SÓC CHUỘT HAMSTER' located in 'QUAN 1 - TP HO CHI MINH'. The listing details include: 'Cho chuột ăn uống, vệ sinh...', 'LƯƠNG: 90000(VND) 1 NGÀY', and 'CHĂM SÓC THÚ CƯNG'. A button labeled 'TẠO BÀI ĐĂNG TÌM VIỆC' is visible. Below the listing, a message says 'Bài đăng tìm việc bạn đang theo dõi'. The status bar at the bottom shows the URL 'http://207.46.136.186/postdetail/30' and the system time '10:25 PM'.

Step 4: Click button "Sửa"

The screenshot shows a web browser window for 'eJob' at the URL '207.46.136.186/postdetail/30'. The top navigation bar includes links for 'Trang Chủ', 'Giới Thiệu', 'Liên Hệ', and 'Góp Ý'. A dropdown menu for 'Phan Hoàng Yên' is open. On the left, there's a sidebar with a user profile picture and a list of menu items: 'Quản lý bài đăng' (selected), 'Bài đăng tìm việc', 'Bài đăng tìm người', 'Quản lý Giao dịch' (selected), 'Xem lịch sử các giao dịch', 'Quản lý tài khoản' (selected), 'Thông tin cá nhân', and 'Thay mật khẩu'. The main content area displays a job listing for 'Chăm sóc chuột hamster'. The listing details include: 'Loại công việc: Chăm sóc thú cưng', 'Nơi làm việc: Quan 1 - TP Hồ Chí Minh | 35 phút trước', 'Mức lương mong muốn: 90,000 (VND) 1 giờ', and 'Thời gian bắt đầu làm việc: 2016-12-06'. Below the listing, sections for 'Mô tả công việc' (describing the job as 'Cho chuột ăn uống, vệ sinh sạch sẽ chỗ ở...') and 'Mô tả bản thân' (describing the person as 'Yêu động vật và đang nuôi chuột hamster.') are shown. At the bottom, there are two buttons: 'Sửa' (Edit) and 'Xóa' (Delete). A section titled 'BÀI ĐĂNG LIÊN QUAN' is visible at the bottom right. The status bar at the bottom shows the URL 'http://207.46.136.186/postdetail/30' and the system time '10:56 PM'.

Step 5: Edit information that you want to change.

The screenshot shows a web browser window with the URL 207.46.136.186/postdetail/30. The title of the page is "Sửa bài đăng". The main content area contains the following fields:

- Chủ đề công việc ***: Chăm sóc chuột hamster
- Loại công việc ***:
 - Mặc định
 - Chăm sóc trẻ em
 - Chăm sóc thú cưng
 - Gia sư
 - Phục vụ quán ăn
 - Bán hàng theo ca
 - Chuyển đồ
 - Nội trợ
 - Nấu ăn
- Mô tả công việc ***: Cho chuột ăn uống, vệ sinh sạch sẽ chỗ ở.
- Mô tả bản thân ***: Yêu động vật, đã và đang nuôi chuột hamster.
- Thời gian bắt đầu làm việc**: 12/06/2016
- Nơi làm việc ***: TP Hồ Chí Minh
- Quận/Huyện ***: Chọn Quận/Huyện
- Mức lương mong**: 90000
- Ngoại tuyển**: (A dropdown menu is open, showing the option "Ngoại tuyển" highlighted in green.)

The left sidebar has a vertical navigation menu with items like Quản lý, Bài đăng, etc. The bottom taskbar shows standard Windows icons for Start, Task View, File Explorer, Edge, Chrome, File, Word, and Excel.

Chủ đề công việc * Chăm sóc bé trên 1 tuổi

Loại công việc *

- Mặc định
- Chăm sóc trẻ em
- Chăm sóc thú cưng
- Gia sư
- Phục vụ quán ăn
- Bán hàng theo ca
- Chuyển đồ
- Nội trợ
- Nấu ăn

Mô tả công việc *

Chăm bé ăn uống, tắm theo yêu cầu của gia đình

Mô tả bản thân *

Yêu trẻ.
Có kinh nghiệm chăm sóc bé sơ sinh.

Thời gian bắt đầu làm việc

12/07/2016

Nơi làm việc *

Ha Noi

Quận/Huyện *

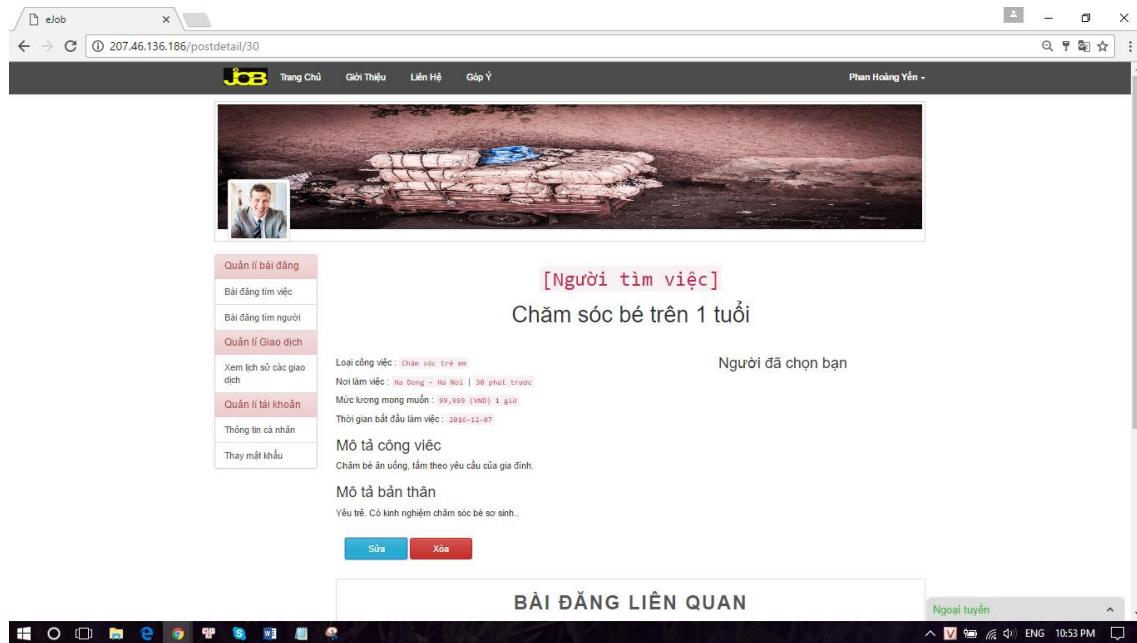
Ha Dong

Mức lương mong muốn (VND/Giờ)

99999

Ngoại tuyển

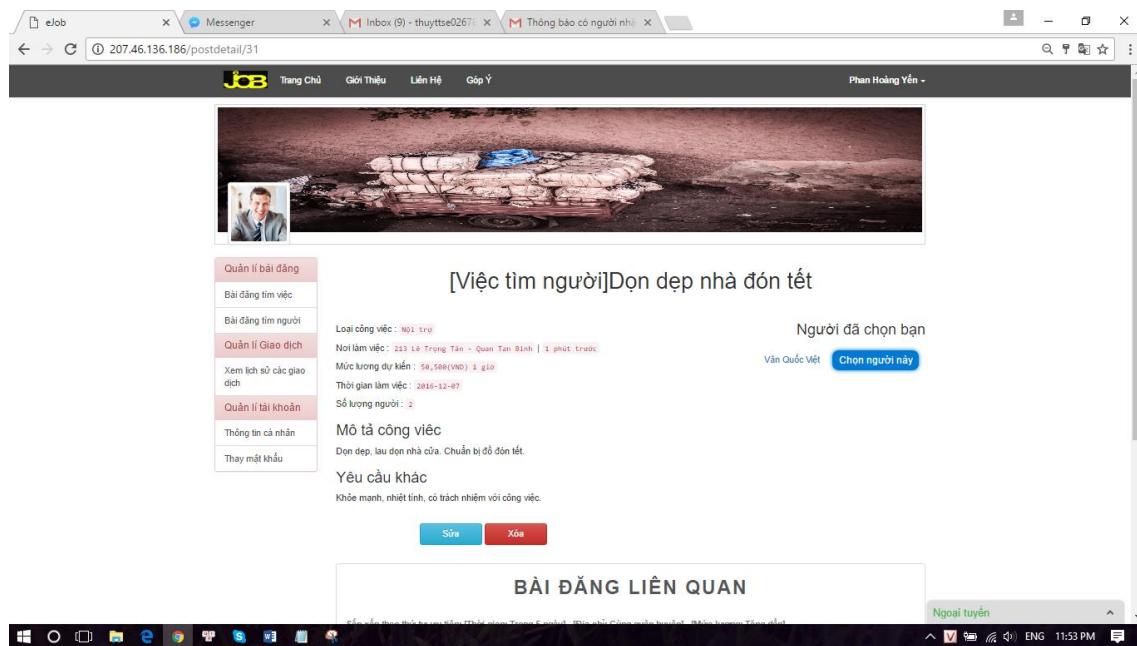
Step 6: Click button “Hoàn tất”



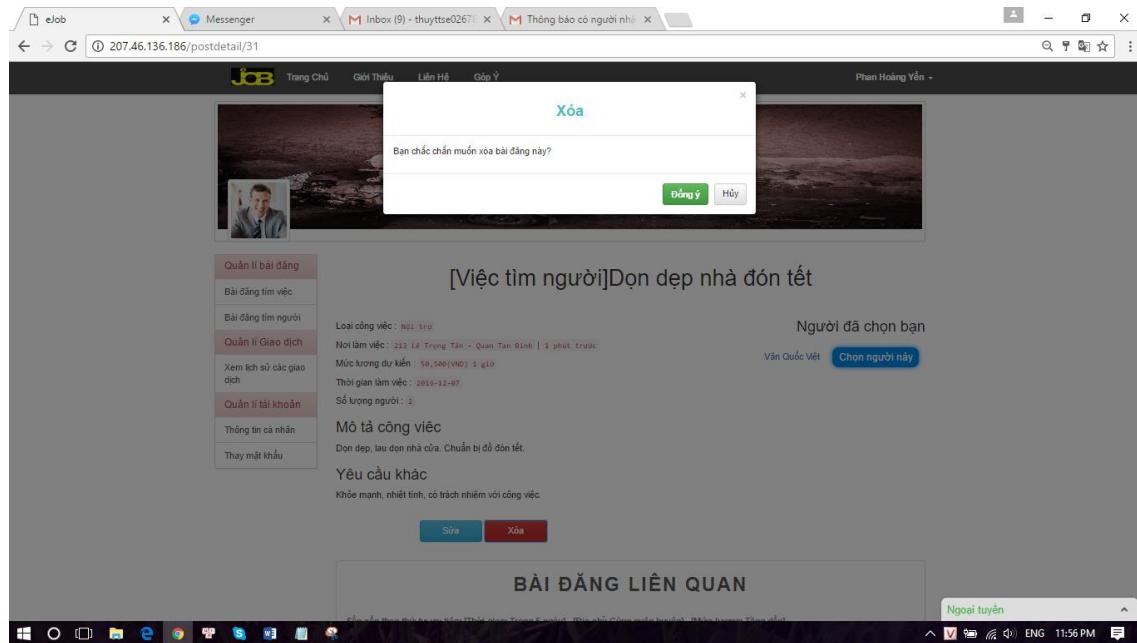
6.2.2.4.3 Delete Post

Step 1: Go to Personal Page.

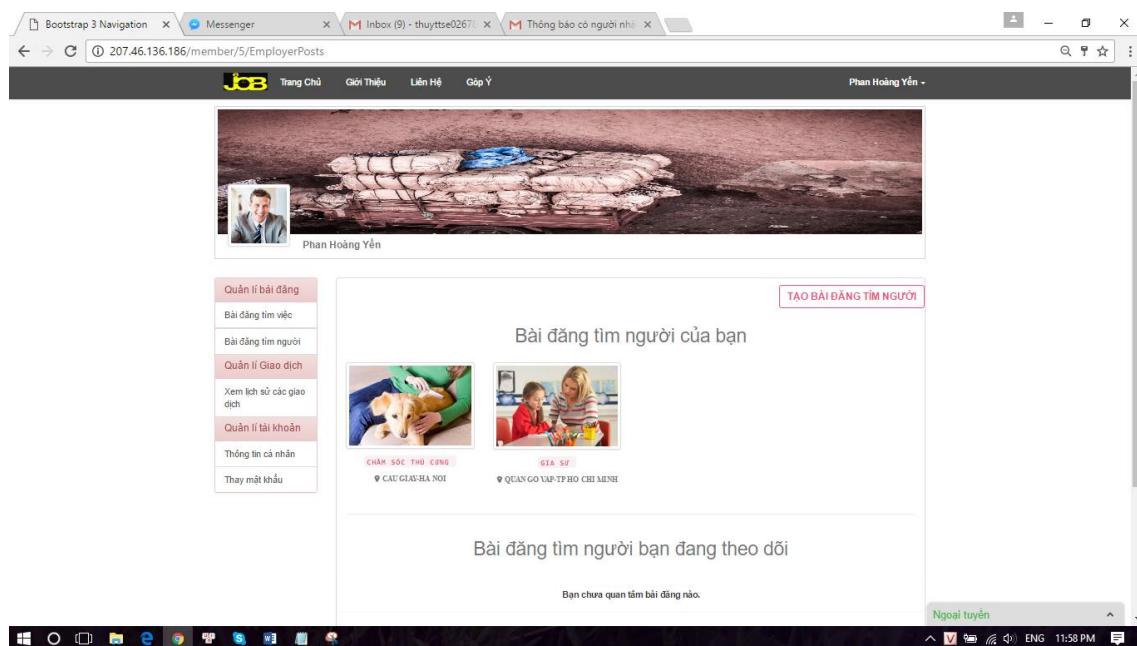
Step 2: Choose own post, and then see post detail.



Step 3: Click button "Xóa". Then confirm popup is displayed.



Step 4: Click button “Đồng ý”. That post is no longer displayed at homepage.



6.2.2.4.4 View selected Posts

6.2.2.4.5 View Posts

Step 1: Go to Personal Page.

Step 2: Click on link “Bài đăng tìm việc” or “Bài đăng tìm người”

The screenshot shows a web browser window with multiple tabs open. The main content area displays a user profile for 'Phan Hoàng Yên' with a small photo and a large banner image of a person working with bags. A sidebar menu on the left includes options like 'Quản lý bài đăng', 'Bài đăng tìm việc', 'Bài đăng tìm người', 'Quản lý Giao dịch', 'Xem lịch sử các giao dịch', 'Quản lý tài khoản', 'Thông tin cá nhân', and 'Thay mật khẩu'. To the right, there's a section titled 'Bài đăng tìm người của bạn' with three thumbnail images: 'MÃI VIỆC KHOA HỌC CHUYÊN MÔN', 'BÁN HÀNG THÉO CÁ', and 'CHĂM SÓC THÚ CƯNG'. Below this is another section titled 'Bài đăng tìm người bạn đang theo dõi' with the message 'Bạn chưa quan tâm bài đăng nào.' A red button labeled 'TAO BÀI ĐĂNG TÌM NGƯỜI' is visible at the top right of the main content area.

6.2.2.4.6 View detail Post

Step 1: View post

Step 2: Click any post.

This screenshot shows the same browser window after clicking on a post. The main content area now displays a detailed job listing for 'CÔNG VIỆC NỘI TRỢ' (Part-time Work) located in 'QUẬN TÂN BÌNH - TP HỒ CHÍ MINH'. The listing includes text: 'Đi chợ, nấu ăn thường ngày...', 'HOURS: 10000 VNĐ / NGÀY', and 'NỘI TRỢ'. A red button labeled 'TAO BÀI ĐĂNG TÌM VIỆC' is visible at the top right of the main content area. The sidebar menu remains the same as in the previous screenshot.

Step 3: Post detail is displayed

[Người tìm việc]
Công việc nội trợ

Loại công việc: Nội trợ
Nơi làm việc: Quận Tân Bình - TP Hồ Chí Minh | 2 phút trước
Mức lương mong muốn: 70,000 (VND) 1 giờ
Thời gian bắt đầu làm việc: 2016-12-07

Mô tả công việc
Đi chợ, nấu ăn thường ngày. Dọn dẹp nhà cửa sạch sẽ.
Mô tả bản thân
Thảo vật, nhanh nhẹn, cần thận, sạch sẽ.

Sửa Xóa

BÀI ĐĂNG LIÊN QUAN

Sắp xếp theo thứ tự ưu tiên: [Thời gian: Trong 5 ngày] - [Địa chỉ: Cùng quận huyện] - [Mức lương: Giảm dần]

Thời gian → Địa chỉ → Mức lương

Không có bài đăng tìm người nào phù hợp với bài đăng này

Ngoài tuyển

6.2.2.4.7 View people choose

View Members who had chosen your own post.

[Người tìm việc]
Công việc nội trợ

Loại công việc: Nội trợ
Nơi làm việc: Quận Tân Bình - TP Hồ Chí Minh | 3 phút trước
Mức lương mong muốn: 70,000 (VND) 1 giờ
Thời gian bắt đầu làm việc: 2016-12-07

Mô tả công việc
Đi chợ, nấu ăn thường ngày. Dọn dẹp nhà cửa sạch sẽ.
Mô tả bản thân
Thảo vật, nhanh nhẹn, cần thận, sạch sẽ.

Sửa Xóa

BÀI ĐĂNG LIÊN QUAN

Sắp xếp theo thứ tự ưu tiên: [Thời gian: Trong 5 ngày] - [Địa chỉ: Cùng quận huyện] - [Mức lương: Giảm dần]

Thời gian → Địa chỉ → Mức lương

Không có bài đăng tìm người nào phù hợp với bài đăng này

Ngoài tuyển

View những người đã chọn bài đăng tìm người của bạn.

[Việc tìm người] Bán hàng tại cửa hàng quần áo.

Loại công việc: Bán hàng theo ca
Nơi làm việc: May - 43 Lê Duẩn - Hai Bà Trưng | 32 phút trước
Mức lương dự kiến: 50,000(VND) 1 giờ
Thời gian làm việc: 2015-12-18
Số lượng người: 3

Mô tả công việc
Trang cài shop. Cập nhật các mẫu mới của shop trên kênh online. Tư vấn giúp khách hàng có những lựa chọn phù hợp với bản thân.

Yêu cầu khác
Năng động, giao tiếp tốt.

Sửa Xóa

BÀI ĐĂNG LIÊN QUAN

Sắp xếp theo thứ tự ưu tiên: [Thời gian: Trong 5 ngày] - [Địa chỉ: Cùng quận huyện] - [Mức lương: Tăng dần]

Không có bài đăng tìm việc nào phù hợp với bài đăng này

Ngoại tuyến

6.2.2.4.8 View matching Posts

The Employer's Posts is matched with related Job Seeker's Posts.

Step 1: View detail post

The related posts will be displayed at the bottom of Post Detail screen.

GIA SƯ ÔN THI TIẾNG NHẬT N2

Loại công việc: Gia sư
Địa chỉ: 31/2 Dương Đình Nghệ - Cầu Giấy - Hà Nội
Đã post: 28 phút trước
Mức lương: 200,000 (VND) 1 giờ
Thời gian làm việc: 2015-12-10
Số lượng người: 1
Người đăng: Phan Hoàng Yến

MÔ TẢ CÔNG VIỆC
Hướng dẫn học và ôn tập trình độ N2.

YÊU CẦU KHÁC
Trình độ tiếng nhật N2 trôi chảy. Có kỹ năng sư phạm. Đúng giờ..

Bài đăng liên quan

Sắp xếp theo thứ tự ưu tiên

Địa chỉ Thời gian Mức lương

GIA SƯ TIẾNG ANH IELTS LUYỆN THI IELTS
Hướng dẫn ôn luyện thi tiếng Anh
Mức lương: 200,000(VND) 1 NGÀY
GIA SƯ
31/2 NGUYỄN TRÃI

GIA SƯ
31/2 ĐƯỜNG BÌNH NGHỆ

Ngoại tuyến

6.2.3 Admin 管理

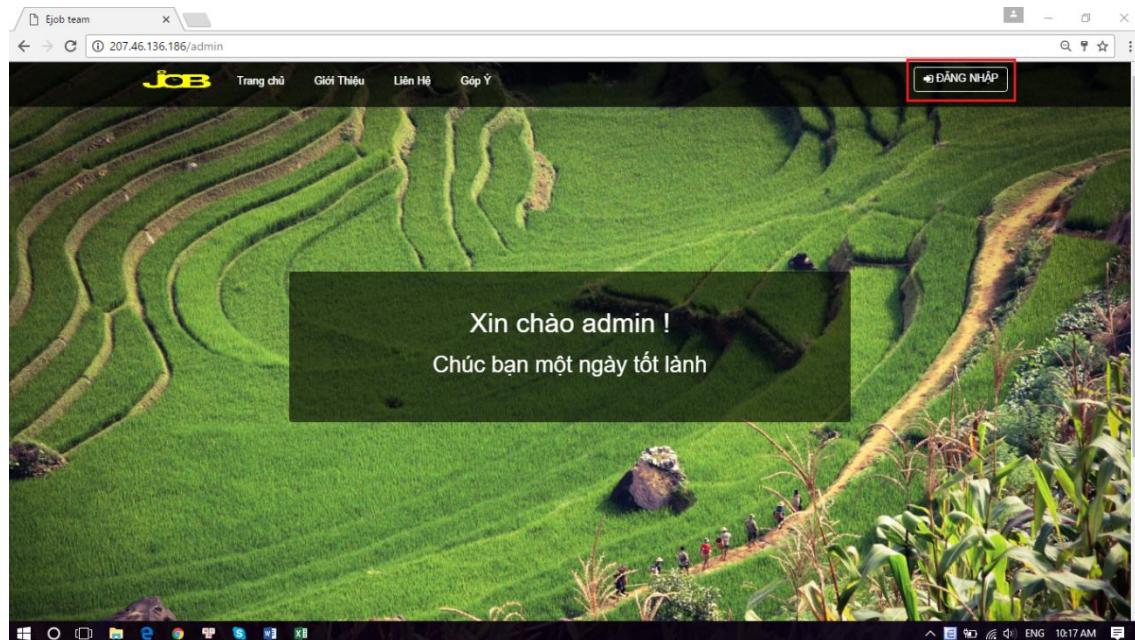
6.2.3.1 Common

6.2.3.1.1 Login

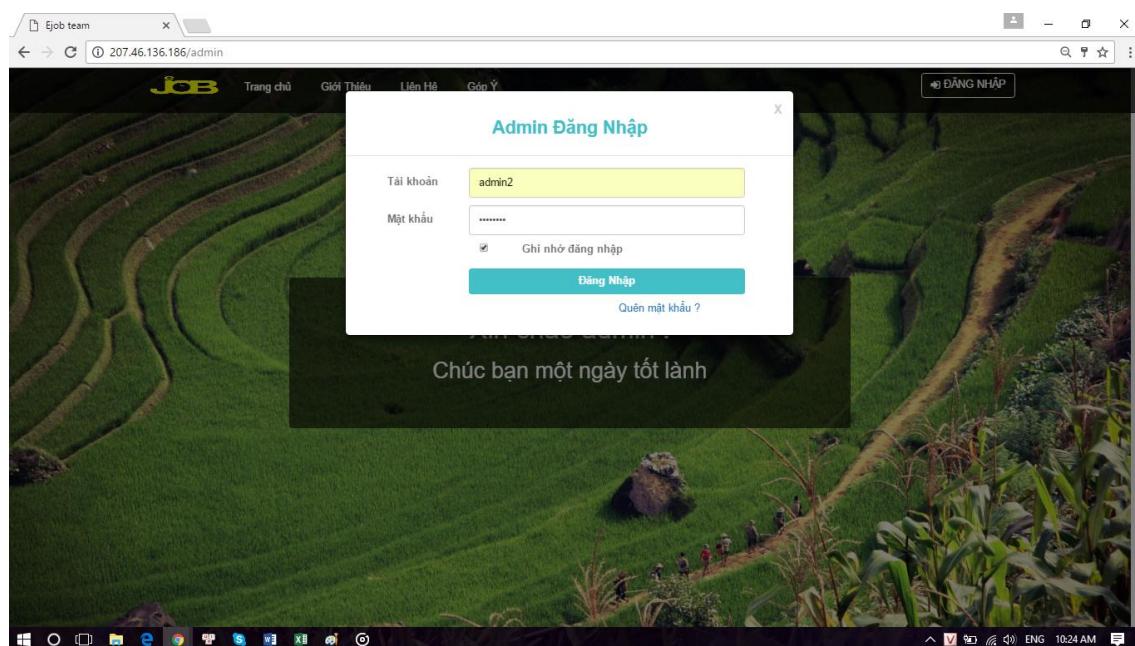
To login using admin account, user do the steps below:

Step 1: Open browser and go to: <http://207.46.136.186/admin>

Step 2: Click on Đăng nhập button in header, Login popup is displayed.



Step 3: Enter admin's account name and password in textbox



Step 4: Click button “Đăng nhập”

The screenshot shows the dashboard of the Jobs Finding System. At the top left is a user profile for 'admin2'. Below it are four main statistics boxes: '8 Thành viên' (red), '36 Bài đăng' (teal), '40,000 VND Số tiền thu được' (green), and '1 Nhận xét - Góp ý' (yellow). To the right of these are two more boxes: '2 Báo cáo' (red) and '9 Loại công việc' (dark grey). Below the statistics is a map of the FPT University, Hoa Lac campus area, showing roads like Quốc lộ 1A and landmarks like Trung Đại học, Hoa Lạc. To the right of the map is a calendar for December 2016.

6.2.3.1.2 Logout

Step 1: Admin Login

Step 2: Click on link “Đăng xuất”

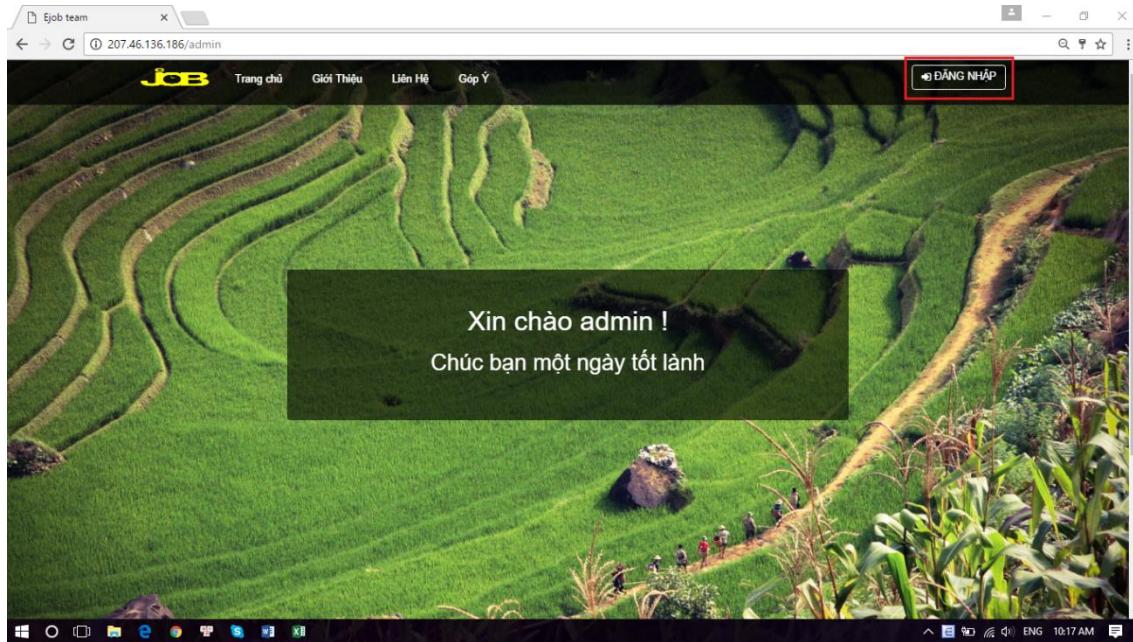
This screenshot is identical to the one above, showing the dashboard with its various statistics and map. However, the 'Đăng xuất' (Logout) button in the sidebar menu is now highlighted with a red box, indicating the next step in the process.

Step 3: Go to Admin's homepage.

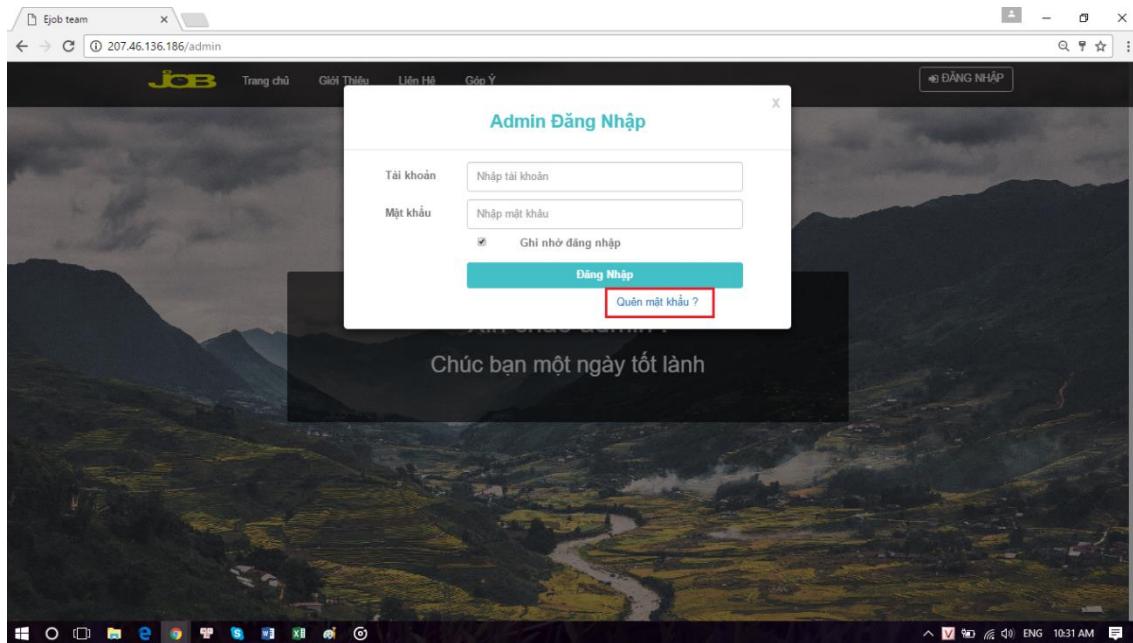
6.2.3.1.3 Forgot password

Step 1: Open browser and go to: <http://207.46.136.186/admin>, then Homepage is displayed.

Step 2: Click on Đăng nhập button in header, Login popup is displayed.

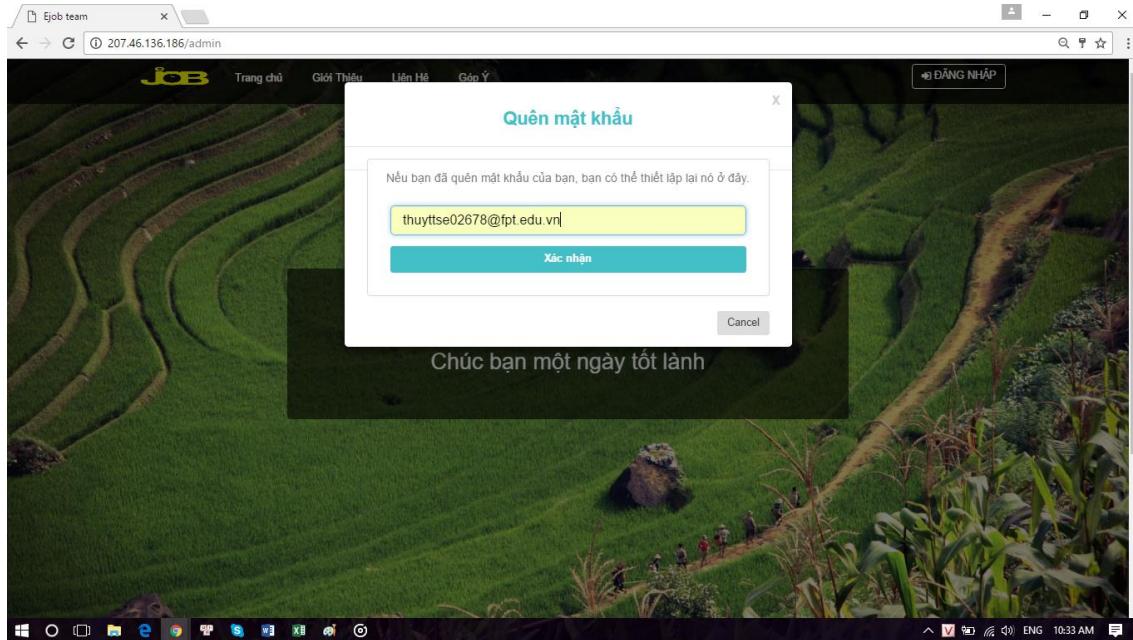


Step 3: Click on link “Quên mật khẩu” in login popup.

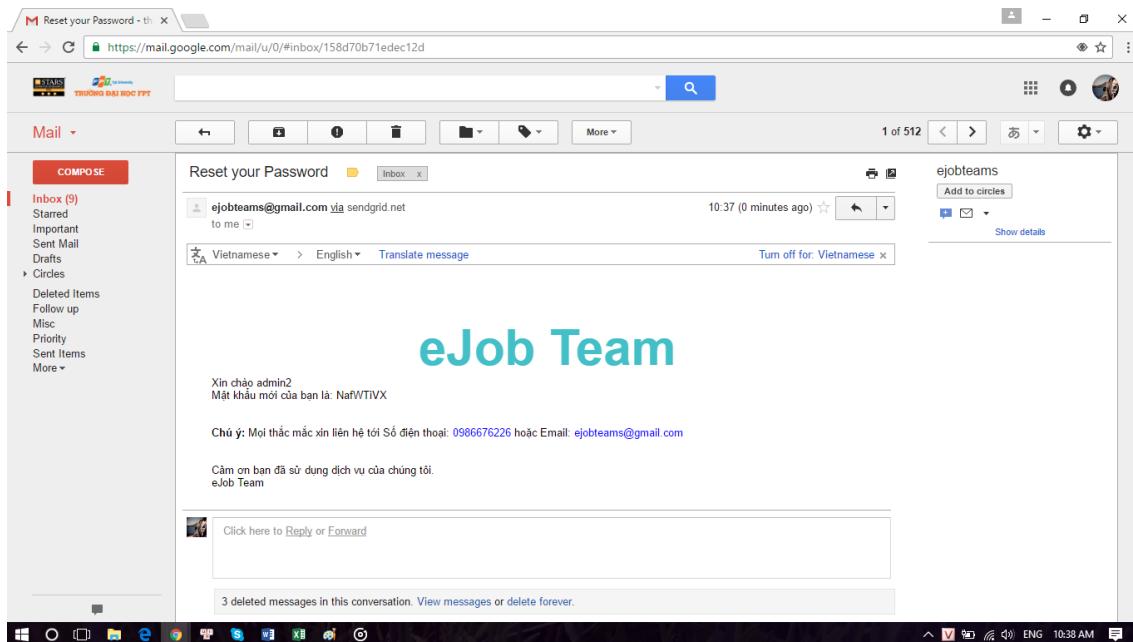


Step 4: Enter registered email.

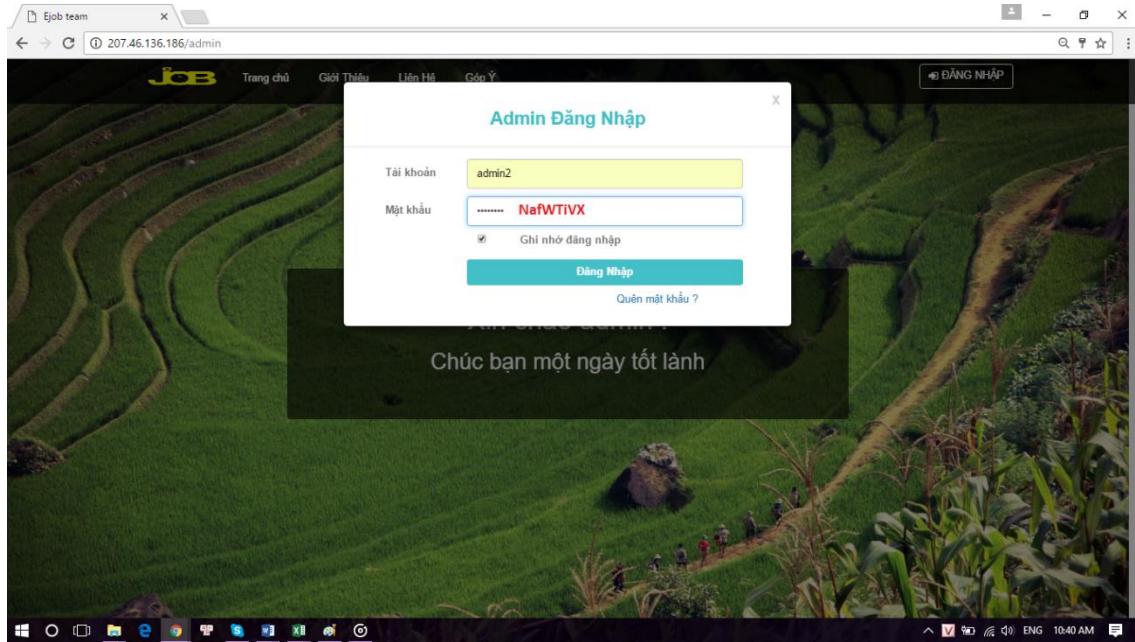
Step 5: Click button “Xác nhận”



Step 6: Check received email to see password.



Step 7: Follow 2.3.1.1 – Login with new password



6.2.3.1.4 Online chat with Member

6.2.3.1.5 Send notify by email

Step 1: Admin login

Step 2: Click on link “Quản lý thành viên”

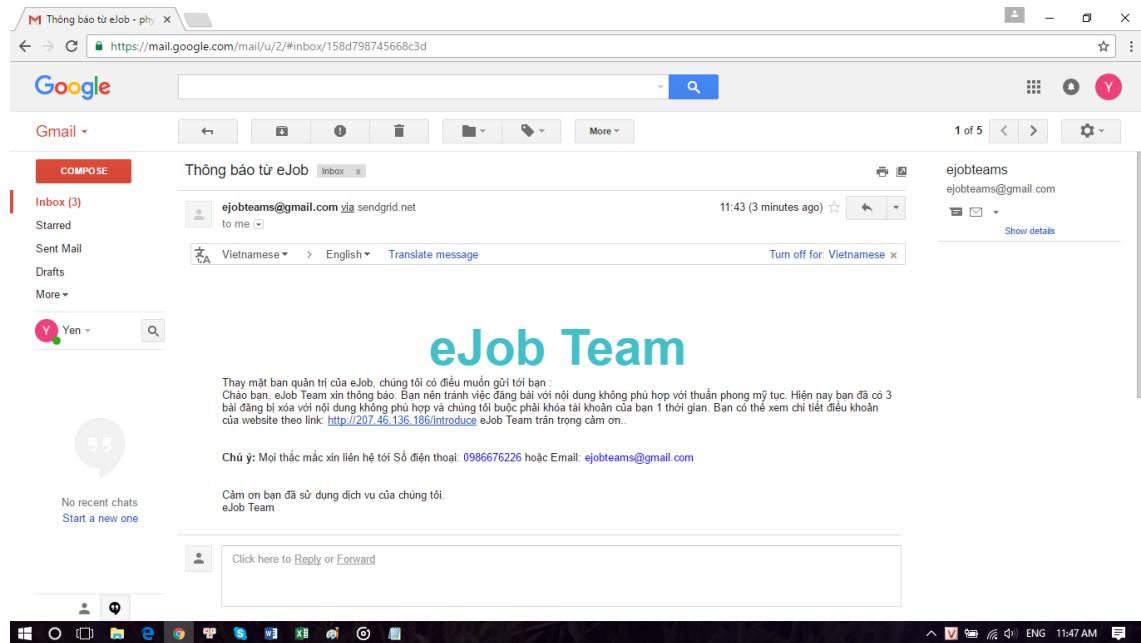
Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
sonpt267	2016-12-07 01:57:52	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
vietvq	2016-12-06 13:38:59	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
kimdung	2016-12-05 15:35:36	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
dangduy	2016-12-05 08:22:30	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam	2016-12-05 08:17:07	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

Step 3: Click button “Gửi thông báo”

Chào bạn, eJob Team xin thông báo. Bạn nên tránh việc đăng bài với nội dung không phù hợp với thuần phong mỹ tục. Hiện nay bạn đã có 3 bài đăng bị xóa với nội dung không phù hợp và chúng tôi buộc phải khóa tài khoản của bạn 1 thời gian.

Step 4: Enter content in textbox.

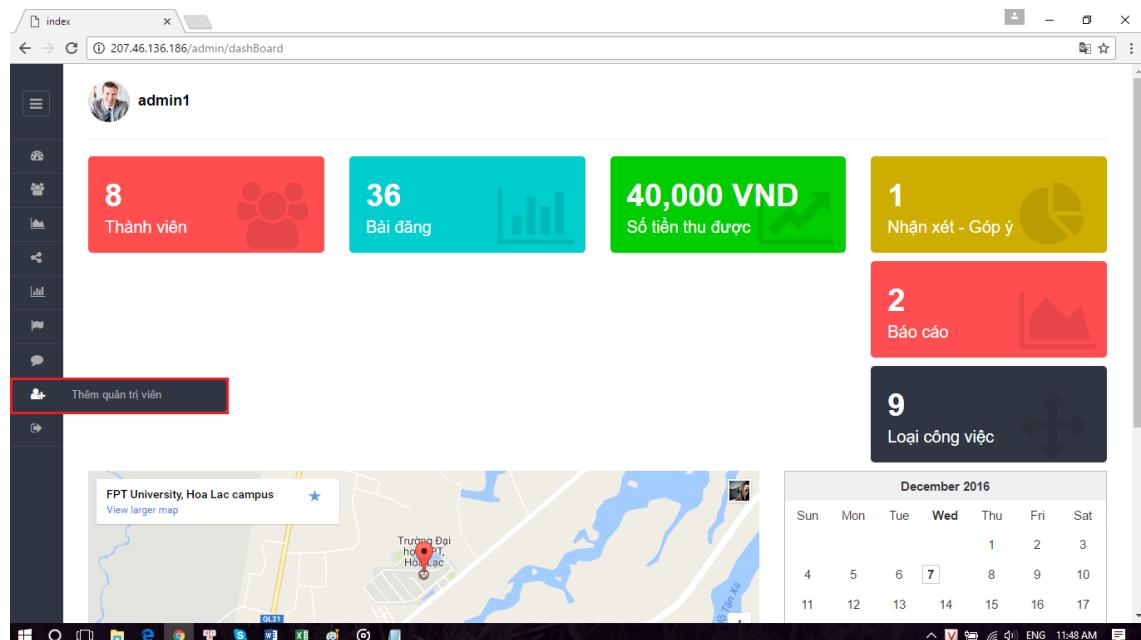
Step 5: Click button “OK”



6.2.3.1.6 Add Admin

Step 1: Addmin Login

Step 2: Click link “Thêm quản trị viên”



Step 3: Input needed informations.

Step 4: Click button “Hoàn tất”

Tạo thêm quản trị viên

Tên đăng nhập *	admin2	Họ và Tên *	Nguyễn Huy Hoàng
Mật khẩu *	*****	Giới tính	<input checked="" type="radio"/> Nam <input type="radio"/> Nữ
Xác nhận mật khẩu *	*****	Ngày sinh	01/02/1993
Email *	nguyễnhuyhoang@gmail.com	Địa chỉ	Khâm Thiên - Đồng Đa - Hà Nội
Số điện thoại *	01231289087	Vui lòng nhập vào các trường có dấu (*)	

HỦY BỎ HOÀN TẤT

6.2.3.2 Manage Members

6.2.3.2.1 Search and View Members

Step 1: Admin Login

Step 2: Click on link “Quản lý thành viên”

Step 3: Enter keyword in search textbox.

Quản lý thành viên

Tất cả (8)

Quản lý tài khoản

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
dangduy	2016-12-05 08:22:30	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam	2016-12-05 08:17:07	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

6.2.3.2.2 View Member Profile

Step 1: Search and view Member

Step 2: Click on link at member's account name

The screenshot shows a web browser window titled 'index' with the URL '207.46.136.186/admin/manage/members'. The page is titled 'Quản lý thành viên' (Manage Members). It displays a table with columns: Tài khoản (Account), Ngày kích hoạt (Activation Date), Bài đăng bị xóa (Posts deleted), Số lần bị Ban (Banned times), and Chức năng (Actions). The table contains five rows, each representing a user account. The first row, 'lyhainam1', is highlighted with a red box.

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
lyhainam1	2016-12-06 13:47:32	0	0	Ban Gửi thông báo Xóa
hoangyen	2016-12-05 18:48:54	0	0	Ban Gửi thông báo Xóa
tranhanam	2016-12-05 08:58:53	0	1	Ban Gửi thông báo Xóa
dangduy	2016-12-05 08:22:30	0	0	Ban Gửi thông báo Xóa
lyhainam	2016-12-05 08:17:07	0	0	Ban Gửi thông báo Xóa

Step 3: Go to Member personal page.

Step 4: Click on link “Thông tin cá nhân”

The screenshot shows a web browser window titled 'Profile' with the URL '207.46.136.186/admin/manage/members/5/profile'. The page features a header with 'Trang chủ', 'Giới Thiệu', 'Liên Hệ', and a dropdown for 'admin1'. Below the header is a banner image of a person. The main content area is titled 'THÔNG TIN TÀI KHOẢN'. On the left, there is a sidebar with links: 'Quản lý bài đăng', 'Bài đăng tìm việc', 'Bài đăng tìm người', 'Quản lý tài khoản', and 'Thông tin cá nhân', which is highlighted with a red box. The right side displays personal information in a table format:

Họ và Tên:	Phan Hoàng Yên
Ngày Sinh:	1993-12-14
Giới tính:	Nữ
Chuyên môn:	Chăm sóc trẻ em
Email:	phy.phanhoangyen@gmail.com
Số điện thoại:	01232190783
Địa chỉ:	Khâm Thiên - Đống Đa - Hà Nội
Mô tả bản thân:	Có trách nhiệm với công việc.

6.2.3.2.3 Ban Member

Step 1: Admin Login

Step 2: Clink on link “Quản lý thành viên”

Step 3: Clink button “Ban”

The screenshot shows a list of members with their details: Tài khoản, Ngày kích hoạt, Bài đăng bị xóa, Số lần bị Ban, and Các chức năng. The 'Bài đăng bị xóa' and 'Số lần bị Ban' columns show values for each user. The 'Các chức năng' column contains three buttons: 'Ban' (orange), 'Gửi thông báo' (blue), and 'Xóa' (red). A red box highlights the 'Ban' button for all users in the list.

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Các chức năng
sonpt267	2016-12-07 01:57:52	0	0	Ban Gửi thông báo Xóa
lyhaianam1	2016-12-06 13:47:32	0	0	Ban Gửi thông báo Xóa
vietvq	2016-12-06 13:38:59	0	1	Ban Gửi thông báo Xóa
hoangyen	2016-12-05 18:48:54	0	0	Ban Gửi thông báo Xóa
kimdung	2016-12-05 15:35:36	0	0	Ban Gửi thông báo Xóa
tranhanam	2016-12-05 08:58:53	0	1	Ban Gửi thông báo Xóa
dangduy	2016-12-05 08:22:30	0	0	Ban Gửi thông báo Xóa
lyhaianam	2016-12-05 08:17:07	0	0	Ban Gửi thông báo Xóa

Step 4: Popup confirm is displayed then click button “OK”

The screenshot shows a confirmation dialog box titled 'Ban' with the message 'Bạn chắc chắn muốn ban thành viên này?'. Below the message are two buttons: 'OK' and 'Hủy'. The 'OK' button is highlighted with a red box.

Step 5: Member is banned.

Quản lý thành viên

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
sonpl267	2016-12-07 01:57:52	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
vietvq	2016-12-06 13:38:59	0	2	<button>Unban</button> <button>Gửi thông báo</button> <button>Xóa</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
kimdung	2016-12-05 15:35:36	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
dangduy	2016-12-05 08:22:30	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam	2016-12-05 08:17:07	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

6.2.3.2.4 Unban Member

Step 1: Ban Member

Step 2: Click on button “Unban”

Quản lý thành viên

Unban

Bạn chắc chắn muốn unban thành viên này?

OK Hủy

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
sonpl267	2016-12-07 01:57:52	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
vietvq	2016-12-06 13:38:59	0	2	<button>Unban</button> <button>Gửi thông báo</button> <button>Xóa</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
kimdung	2016-12-05 15:35:36	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
dangduy	2016-12-05 08:22:30	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam	2016-12-05 08:17:07	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

Step 3: Confirm popup is displayed then click button “OK”

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
sonpt267	2016-12-07 01:57:52	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
vietvq	2016-12-06 13:38:59	0	2	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
kimdung	2016-12-05 15:35:36	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
dangduy	2016-12-05 08:22:30	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam	2016-12-05 08:17:07	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

6.2.3.2.5 Delete Member

Step 1: Manage Member

Step 2: Click on button “Xóa”

Step 3: Popup confirm is displayed.

Xóa

Bạn chắc chắn muốn xóa thành viên này?

OK Hủy

Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
lamngochan	2016-12-08 12:47:15	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
sonpt267	2016-12-07 01:57:52	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
vietvq	2016-12-06 13:38:59	0	2	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
kimdung	2016-12-05 15:35:36	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

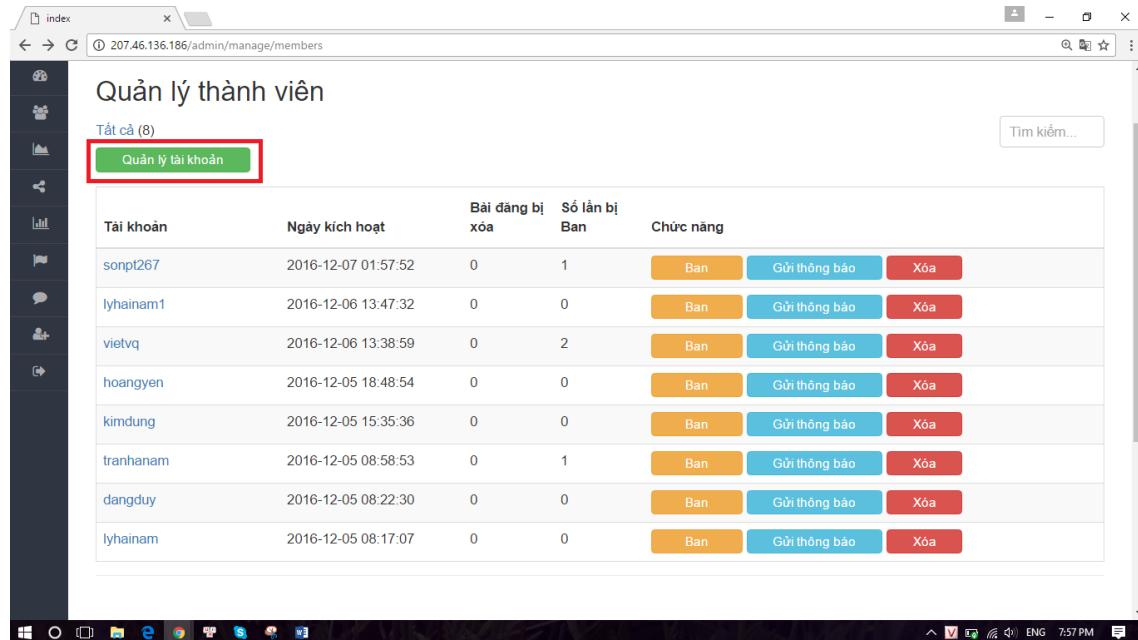
Step 4: Click button “OK”

Step 5: Account is deleted.

6.2.3.2.6 Add money into Employer's account

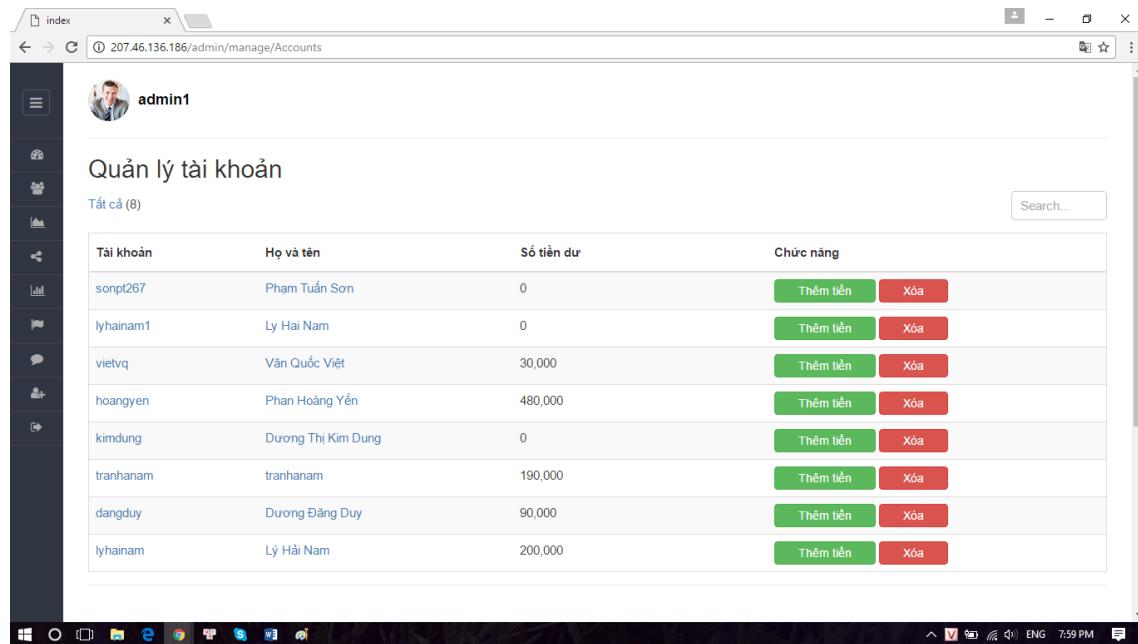
Step 1: Search and View Members

Step 2: Click button “Quản lý tài khoản”



Tài khoản	Ngày kích hoạt	Bài đăng bị xóa	Số lần bị Ban	Chức năng
sontp267	2016-12-07 01:57:52	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam1	2016-12-06 13:47:32	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
vietvq	2016-12-06 13:38:59	0	2	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
hoangyen	2016-12-05 18:48:54	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
kimdung	2016-12-05 15:35:36	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
tranhanam	2016-12-05 08:58:53	0	1	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
dangduy	2016-12-05 08:22:30	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>
lyhainam	2016-12-05 08:17:07	0	0	<button>Ban</button> <button>Gửi thông báo</button> <button>Xóa</button>

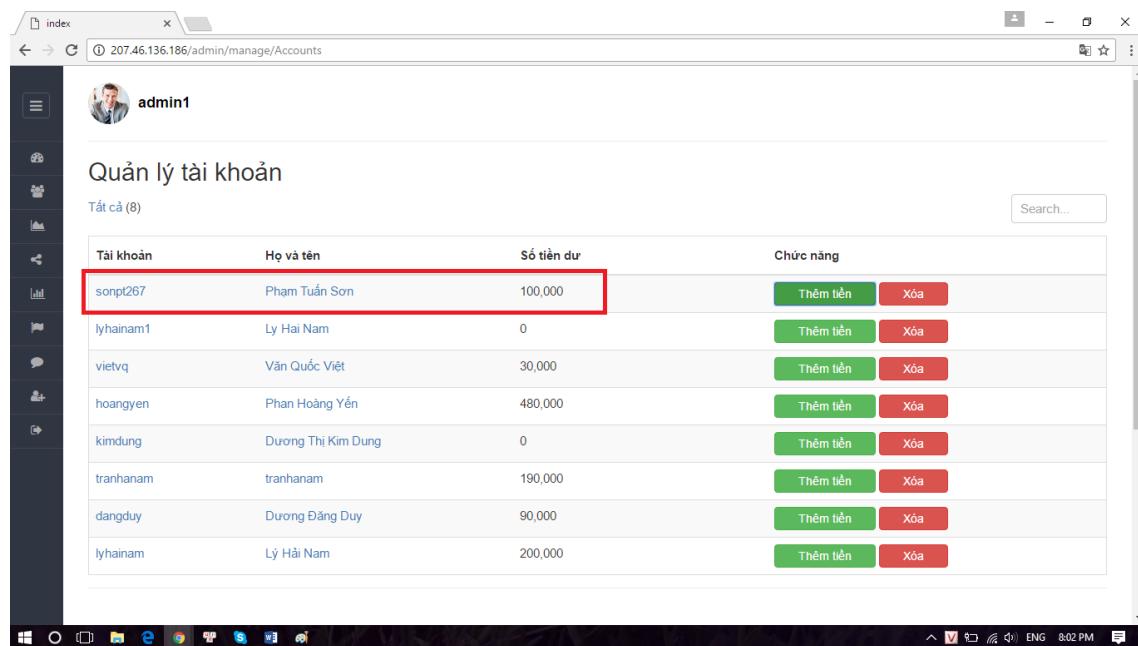
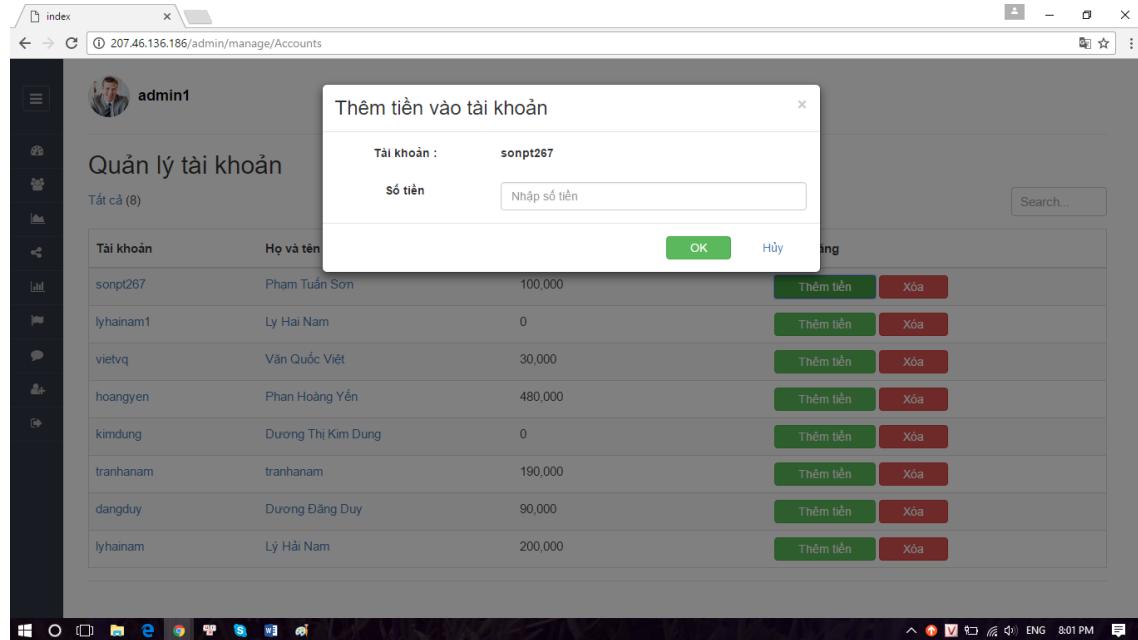
Step 3: Go to manage employer's account, then click button “Thêm tiền”.



Tài khoản	Họ và tên	Số tiền dư	Chức năng
sontp267	Phạm Tuấn Sơn	0	<button>Thêm tiền</button> <button>Xóa</button>
lyhainam1	Lý Hải Nam	0	<button>Thêm tiền</button> <button>Xóa</button>
vietvq	Văn Quốc Việt	30,000	<button>Thêm tiền</button> <button>Xóa</button>
hoangyen	Phan Hoàng Yến	480,000	<button>Thêm tiền</button> <button>Xóa</button>
kimdung	Đương Thị Kim Dung	0	<button>Thêm tiền</button> <button>Xóa</button>
tranhanam	tranhanam	190,000	<button>Thêm tiền</button> <button>Xóa</button>
dangduy	Dương Đăng Duy	90,000	<button>Thêm tiền</button> <button>Xóa</button>
lyhainam	Lý Hải Nam	200,000	<button>Thêm tiền</button> <button>Xóa</button>

Step 4: Enter number of money in textbox.

Step 5: Click button “OK”

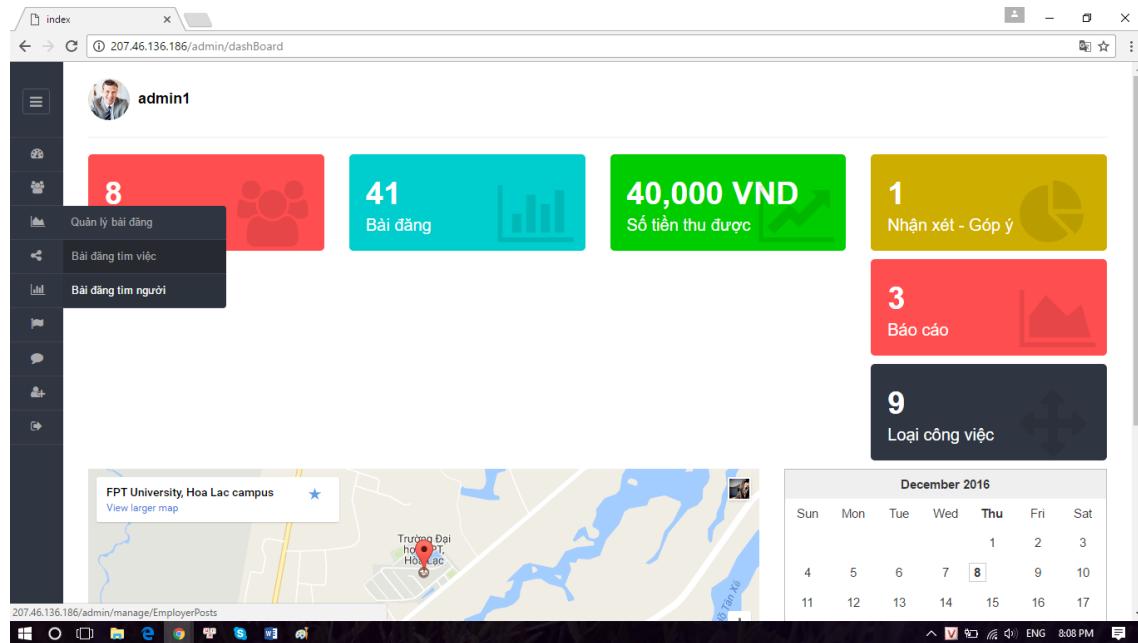


6.2.3.3 Manage Posts

6.2.3.3.1 Search and view Employer's Post

Step 1: Admin Login

Step 2: Click on link “Bài đăng tìm người”



Step 3: Admin can search by dropdown list or enter key word in textbox search.

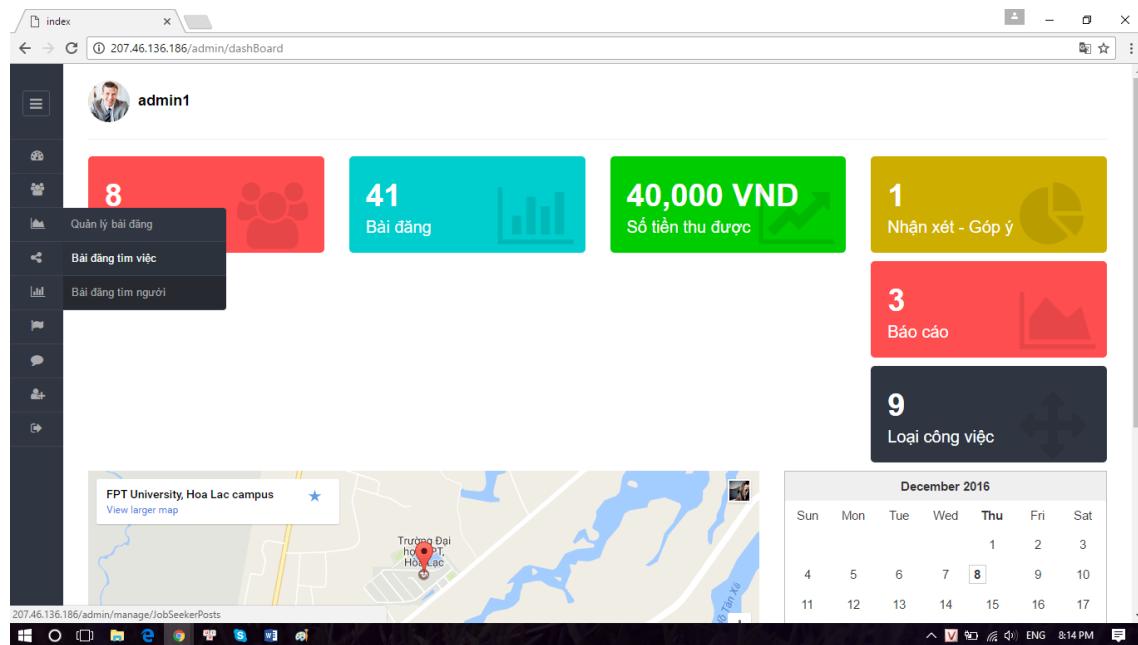
The screenshot shows the 'Bài đăng tìm người' (Job Seeker's Post) search results page. It includes a search bar with dropdowns for 'Tiêu đề' (Job Title), 'Người đăng' (Poster), 'Loại công việc' (Job Type), 'Địa chỉ' (Address), and 'Chức năng' (Function). Below the search bar is a table listing job posts:

Tiêu đề	Người đăng	Loại công việc	Địa chỉ	Chức năng
Gia sư tiếng nhật trình độ N3	vietvq	Gia sư	38A Trần Thái Tông - Cau Giay - Ha Noi	Xóa
Gia sư ôn thi tiếng nhật N2	hoangyen	Gia sư	91/2 Dương Đình Nghệ - Cau Giay - Ha Noi	Xóa

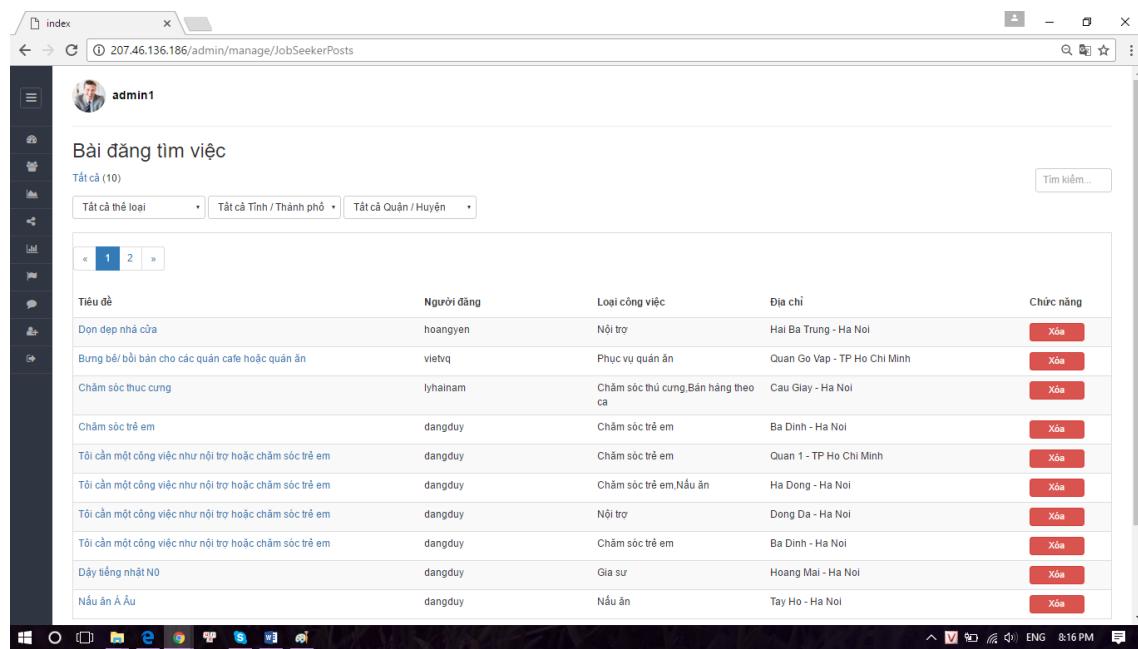
6.2.3.3.2 Search and view Job Seeker's Post

Step 1: Admin Login

Step 2: Click on link "Bài đăng tìm việc"



Step 3: List all posts or search for specific posts by dropdown list or enter key word on textbox search.



6.2.3.3 Delete Post

Step 1: Search and view Posts

Step 1.1: Search and view Employer's Post

Step 1.2: Search and view Job Seeker's Post

Step 2: Click button “Xóa”

Bài đăng tìm người

Tất cả (4)

Tiêu đề	Người đăng	Loại công việc	Địa chỉ	Chức năng
Gia sư tiếng nhật trình độ N3	vietvq	Gia sư	38A Trần Thái Tông - Cau Giay - Ha Noi	Xóa
Gia sư ôn thi tiếng nhật N2	hoangyen	Gia sư	91/2 Dương Đình Nghệ - Cau Giay - Ha Noi	Xóa
Bán hàng tại cửa hàng quần áo.	hoangyen	Bán hàng theo ca	May - 43 Lê Duẩn - Hai Ba Trung - Ha Noi	Xóa
Chăm sóc mèo	hoangyen	Chăm sóc thú cưng	Số nhà 151/2 - Nguyễn Phong Sắc - Cau Giay - Ha Noi	Xóa

Step 3: Popup confirm is displayed. Then click button “OK”

Xóa

Bạn chắc chắn muốn xóa bài đăng tìm người này?

OK Hủy

6.2.3.3.4 View detail Posts

Step 1: Search and view Post

Step 1.1: Search and view Employer's Posts

Step 1.2: Search and view Job Seeker's Posts

Step 2: Click on link Post's title

Step 3: Go to Post detail page

GIA SƯ ÔN THI TIẾNG NHẬT N2

Loại công việc: Gia sư
Địa chỉ: 91/2 Dương Đình Nghệ - Cầu Giấy - Hà Nội
Đã post: 1 ngày trước
Mức lương: 200,000 (VND) 1 giờ
Thời gian làm việc: 2016-12-10
Số lượng người: 1
Người đăng: Phan Hoàng Yến

MÔ TẢ CÔNG VIỆC
Hướng dẫn học và ôn tập trình độ N2.

YÊU CẦU KHÁC
Trình độ tiếng nhật N2 trở lên. Có kỹ năng sư phạm. Đúng giờ..

Bài đăng liên quan

6.2.3.4 Manage Feedback

6.2.3.4.1 View Feedback

Step 1: Admin Login

Step 2: Click on link “Góp ý và báo cáo sự cố”

index

207.46.136.186/admin/dashBoard

admin1

8 Thành viên

41 Bài đăng

40,000 VND Số tiền thu được

1 Nhận xét - Góp ý

3 Báo cáo

9 Loại công việc

Góp ý - Báo cáo sự cố

FPT University, Hoa Lac campus

Trường Đại học FPT

December 2016

Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17

Step 3: Go to manage feedback page.

The screenshot shows a web browser window titled 'index' with the URL '207.46.136.186/admin/manage/feedbacks'. The page is titled 'Góp ý \$ Báo cáo sự cố' (Feedback \$ Report Incident). It displays a table with one row of data:

Số	Người góp ý	Nội dung	Chức năng
1	hoangyen	Minh đã thử truy cập web trên smartphone, khá là ok. Hi vọng mình có thể dùng smartphone để truy cập web, như vậy sẽ truy cập web được thường xuyên hơn. Cảm ơn eJob Team.	Xóa

6.2.3.4.2 Delete Feedback

Step 1: View feedback

Step 2: Click on button “Xóa”

Step 3: Popup confirm is displayed. Then click button “OK”

6.2.3.5 Statistic

6.2.3.5.1 Statistic earned money from Members

Step 1: Admin Login

Step 2: Click on link “Thống kê doanh thu”

The screenshot shows the dashboard of the Jobs Finding System. It features a sidebar with icons for users, posts, statistics, reports, and earnings. The main area has four large cards: '8 Thành viên' (8 Members) in red, '41 Bài đăng' (41 Posts) in teal, '40,000 VND Số tiền thu được' (40,000 VND Earnings) in green, and '1 Nhận xét - Góp ý' (1 Review - Comment) in yellow. Below these are three smaller cards: '3 Báo cáo' (3 Reports) in pink, and '9 Loại công việc' (9 Job Types) in dark grey. A map of FPT University, Hoa Lac campus is displayed with a red pin on 'Trường Đại học FPT'. To the right is a calendar for December 2016.

Step 3: Go to statistic earned money from Members page

The screenshot shows the 'Quản lý doanh thu' (Sales Management) page. It displays a total earnings of 'Tổng số doanh thu : 40,000 VND'. Below this is a dropdown menu 'Thống kê theo ngày'. A table lists five transactions:

Người thuê	Người làm	Bài đăng	Số tiền
hoangyen	vietvq	Gia sư tiếng nhật trình độ N5	10,000
hoangyen	tranhanam	Giao hàng tận nhà hoặc chuyển đồ, chuyển nhà	10,000
tranhanam	hoangyen	Tìm người giúp việc gia đình nội trợ	10,000
dangduy	lyhainam	Gia sư cấp 2	10,000

6.2.3.5.2 View Chart

Step 1: Admin Login

Step 2: Click on link “Thống kê theo loại công việc”

Step 3: Go to chart page



6.2.3.6 Manage Reports

6.2.3.6.1 View reports about Post

Step 1: Admin Login

Step 2: Click on link “Quản lý báo cáo”

Quản Lý Báo Cáo

Tất cả (3)

Bài đăng	Người báo cáo	Nội dung báo cáo	Chức năng
Bán hàng theo ca tối	tranhanam	Đó là nội dung spam,	Xóa
Chăm sóc trẻ em	kimdung	Ngôn ngữ, hình ảnh đồi truy,	Xóa
Chăm sóc trẻ em	tranhanam	Người dùng có thể để lộ thông tin cá nhân, Đó là nội dung spam,	Xóa

6.2.3.6.2 Delete Report about Post

Step 1: View report about post

Step 2: Click button “Xóa”

Step 3: Confirm popup is displayed. Then click button “OK”

6.2.3.7 Manage Category

6.2.3.7.1 View Category

Step 1: Admin Login

Step 2: Click on link “Quản lý các loại công việc”

The screenshot shows the admin dashboard with the following statistics:

- 8 Thành viên
- 41 Bài đăng
- 40,000 VND Số tiền thu được
- 1 Nhận xét - Góp ý
- 3 Báo cáo
- 9 Loại công việc

Below the stats is a map of FPT University, Hoa Lac campus with a red pin on the university grounds. To the right is a calendar for December 2016.

Step 3: Go to manage category page.

The screenshot shows the 'Quản Lý Các Loại Công Việc' (Manage Categories) page with the following details:

Tất cả (5)

Loại công việc	Số bài đăng sử dụng	Chức năng
Máy tính	4	<button>Sửa</button> <button>Xóa</button>
Chăm sóc trẻ em	8	<button>Sửa</button> <button>Xóa</button>
Chăm sóc thú cưng	9	<button>Sửa</button> <button>Xóa</button>
Gia sư	10	<button>Sửa</button> <button>Xóa</button>
Phục vụ quán ăn	4	<button>Sửa</button> <button>Xóa</button>

6.2.3.7.2 Add Category

Step 1: View category

Step 2: Click button “Thêm loại công việc”

Loại công việc	Số bài đăng sử dụng	Chức năng
Máy định	4	<button>Sửa</button> <button>Xóa</button>
Chăm sóc trẻ em	8	<button>Sửa</button> <button>Xóa</button>
Chăm sóc thú cưng	9	<button>Sửa</button> <button>Xóa</button>
Gia sư	10	<button>Sửa</button> <button>Xóa</button>
Phục vụ quán ăn	4	<button>Sửa</button> <button>Xóa</button>

Step 4: Enter category in textbox and upload an image.

Step 5: Click button “Đồng ý”

Thêm loại công việc mới

Loại công việc:

Ảnh công việc:

Đồng ý Hủy

6.2.3.7.3 Delete Category

Step 1: View category

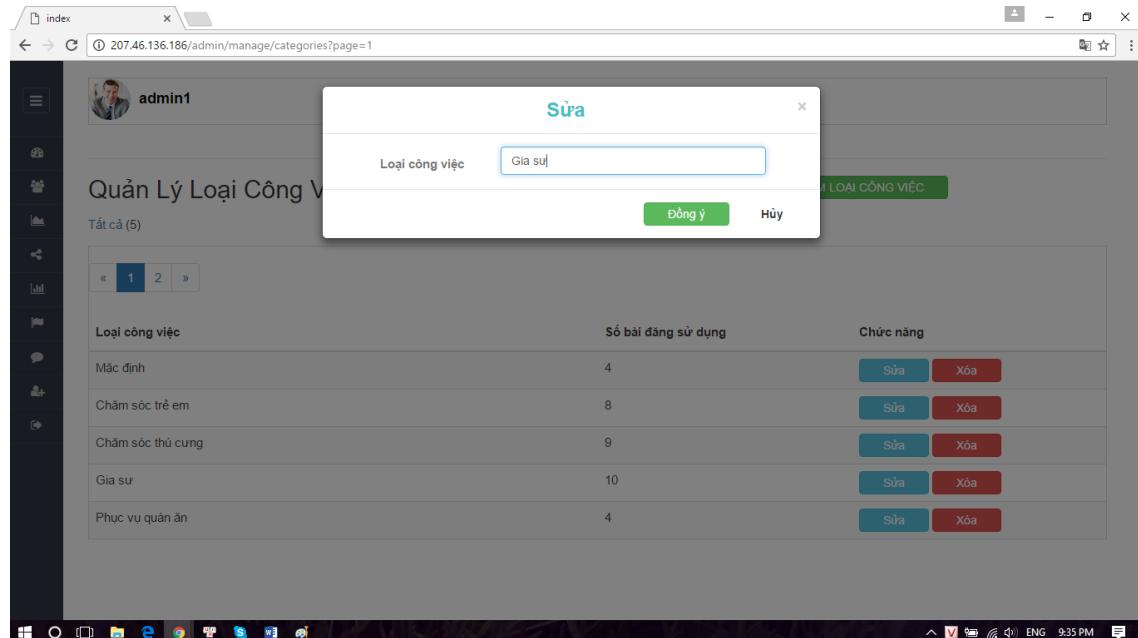
Step 2: Click “Xóa” button

Step 3: Popup confirm is displayed. Click button “OK”

6.2.3.7.4 Edit Category

Step 1: Follow 2.3.7.1 View category

Step 2: Click button “Sửa”



Step 3: Edit information displayed at textbox.

Step 4: Click button “Đồng ý”

6.2.4 System システム

- **Send hidden profile of a Job Seeker to Employer:** After System deduct money in Employer's Account successful, system display a pop up contain Profile of Job Seeker. System also send email to Employer, this Employer be able to access to pick Job Seeker's Profile Page.
- Send notification to Member's email in the following cases:
 - Delete Member banned over 3 times.
 - Ban and unban Member based on System's Rule: 1 member will be banned if got 3 deleted posts by admin. Unban after 1 week.
 - Deduct Employer's money when they are received Job Seeker's hidden profile.
- Delete user when that user is banned 3 times.
- Ban user based in website's rule.
System active event "Delete User" and "Ban User" after event "Delete Post" occurs (delete by admin).
- Deduct Employer's money when they are received Job Seeker's hidden information. (Event "Deduct Money" occurs when an Employer accept a transaction".
- Create chart to compare between job types. (Event "Create Chart" occurs when Admin click on link "View Chart".
- List matching posts.