



MINISTRY OF
EDUCATION AND TRAINING

FPT UNIVERSITY

Capstone Project Document

Corporate Training System

GROUP 02	
Group members	Lê Phúc Thịnh – SE61759 (Leader) Võ Thạch Nguyên – SE61778 Phạm Chánh Hưng – SE61807 Đặng Thế Anh – SE61372
Supervisor	Nguyễn Huy Hùng
Ext. Supervisor	N/A
Capstone Project code	CTS

- Ho Chi Minh City, **8th January, 2018** -

This page is intentionally left blank

Table of Contents

List of Table	4
List of Figure	8
Definitions, Acronyms, and Abbreviations.....	11
A. Report No. 1 Introduction.....	12
1. Project Information.....	12
2. Introduction.....	12
3. Current Situation.....	12
4. Problem Definition	12
5. Proposed Solution.....	13
5.1 Feature functions.....	13
5.2 Advantages and Disadvantages.....	13
6. Functional Requirements	13
7. Role and Responsibility.....	14
B. Report No. 2 Software Project Management Plan	15
1. Problem Definition	15
1.1 Name of this Capstone Project.....	15
1.2 Problem Abstract.....	15
1.3 Project Overview	15
2. Project organization.....	18
2.1 Software Process Model.....	18
2.2 Roles and responsibilities.....	19
2.3 Tools and Techniques.....	20
3. Project Management Plan.....	20
3.1 Product Backlog.....	20
3.2 Sprint Backlog.....	20
3.3 All Meeting Minutes.....	20
4. Coding Convention.....	20
C. Report No. 3 Software Requirement Specification	21
1. User Requirement Specification.....	21
1.1 Guest Requirement	21
1.2 Learner Requirement	21

1.3 Author Requirement	21
1.4 System Admin Requirement.....	22
1.5 Manager Requirement	22
2. System Requirement Specification.....	22
2.1 External Interface Requirement	22
2.2 System Overview Use Case.....	23
2.3 List of Use Case.....	23
3. Software System Attribute.....	107
3.1 Usability	107
3.2 Reliability.....	107
3.3 Availability	107
3.4 Security.....	108
3.5 Maintainability.....	108
3.6 Portability.....	108
3.7 Performance	108
4. Conceptual diagram	109
D. Report No. 4 Software Design Description	110
1. Design Overview	110
2. System Architectural Design.....	111
2.1 System Architecture Overview.....	111
3. Component Diagram	113
4. Detail Description.....	114
4.1 Class Diagram	114
4.2 Class Diagram Explanation.....	117
4.3 Interactive Diagram.....	124
4.3.1 Web Application	124
5. User Interface Design	134
5.1 Web Application.....	134
6. Database Design.....	155
6.1 Entity relationship diagram (ERD)	155
6.2 Entity dictionary	157
7. Algorithms	157
7.1 Duplicate course	157
E. Report No. 5 System Implementation & Test	159

1. Introduction.....	159
1.1 Overview.....	159
1.2 Test approach.....	160
2. Database Relationship Diagram.....	160
2.1 Physical diagram	160
2.2 Data dictionary	162
3. Test plan	167
3.1 Features to be tested	167
3.2 Features not to be tested.....	167
4. System Testing Test Case	167
4.1 Test cases.....	167
F. Report No. 6 Software User's Manual.....	176
1. Installation Guide	176
1.1 Setting up environment at server side	176
1.2 Web Applications Deployment Process	176
2. Installation Guide	177
2.1 Admin	184
2.2 Manager.....	185
2.3 Author.....	188
2.3 Learner.....	191

List of Table

Table 1: Roles and Responsibilities.....	14
Table 2: Hardware Requirement for Server	17

Table 3: Hardware Requirement for PC.....	17
Table 4: Software requirement	18
Table 5: Roles and Responsibilities Details.....	20
Table 6: Tools and Techniques.....	20
Table 7: Login specification	25
Table 8: Log out specification	26
Table 9: View plan list specification	28
Table 10: View plan specification	29
Table 11: Update profile specification.....	31
Table 12: View learning activity specification.....	32
Table 13: Search course specification.....	34
Table 14: View course specification	36
Table 15: View program specification	37
Table 16: Enroll course specification.....	39
Table 17: Learn lecture specification.....	41
Table 18: Take quiz specification.....	43
Table 19: View quiz result specification.....	44
Table 20: Add discussion specification	46
Table 21: Update discussion specification.....	47
Table 22: Remove discussion specification.....	49
Table 23: View learner statistic specification	50
Table 24: Create course specification	52
Table 25: Remove course specification.....	53
Table 26: Update course specification.....	55
Table 27: Add lecture specification.....	56
Table 28: Update lecture specification	58
Table 29: Remove lecture specification	59
Table 30: Update quiz specification.....	60
Table 31: Add question specification	62
Table 32: Update question specification	64
Table 33: Remove question specification	65
Table 34: Add answer specification	67
Table 35: Update answer specification	69
Table 36: Remove answer specification	71
Table 37: Add section specification	72
Table 38: Update section specification	73
Table 39: Remove section specification	74
Table 40: Add brand specification.....	76
Table 41: Update brand specification	78
Table 42: Remove brand specification	80
Table 43: View report specification	82
Table 44: Update user's information specification.....	83
Table 45: Create user specification	85

Table 46: Create plan specification	86
Table 47: Update plan specification.....	88
Table 48: Add learner to plan	90
Table 49: Remove plan specification.....	92
Table 50: Create program specification.....	93
Table 51: Update program specification	95
Table 52: Remove program specification	97
Table 53: Create skill specification.....	98
Table 54: Update skill specification	99
Table 55: Remove skill specification	101
Table 56: Add feedback specification.....	102
Table 57: Update feedback specification.....	102
Table 58: Remove feedback specification.....	105
Table 59: Update brand specification	107
<i>Table 60: Conceptual diagram data dictionary.....</i>	110
Table 61: Component Dictionary	114
Table 62: Class Dictionary	116
Table 63:User attributes.....	117
Table 64: Brand attributes	117
Table 65: Plan attributes	118
Table 66: PlanParticipant attributes	118
Table 67: Program attributes.....	119
Table 68: Course attributes.....	119
Table 66: CourseEnroll attributes.....	119
Table 70: Skill attributes	120
Table 71: CourseSkill attributes	120
Table 72: CourseSection attributes.....	120
Table 70: CourseDiscussion attributes.....	121
Table 74: CourseLecture discussion.....	121
Table 75: LearnerProcess attributes.....	122
Table 76: Quiz attributes.....	122
Table 77: Question attributes	122
Table 78: Answer attributes	123
Table 79: LearnerQuiDetail attributes.....	123
Table 80: LearnerQuiz attributes.....	123
Table 81: Role attributes.....	123
Table 82: RolePermission attributes.....	124
Table 83: Login page – Field, Button, Hyperlink.....	134
Table 84: View list plan – Field, Button, Hyperlink	135
Table 85: Plan's overview page – Field, Button, Hyperlink	137
Table 86: User's information page – Field, Button, Hyperlink.....	137
Table 87: Take quiz page – Field, Button, Hyperlink	138
Table 88: Brand detail page – Field, Button, Hyperlink	140

Table 89: View list user – Field, Button, Hyperlink	141
Table 90: List plans – Field, Button, Hyperlink	142
Table 91: Update/Remove plan – Field, Button, Hyperlink	143
Table 92: Add author to plan – Field, Button, Hyperlink.....	144
Table 93: List programs section – Field, Button, Hyperlink.....	145
Table 94: Create new program modal – Field, Button, Hyperlink.....	146
Table 95: Course's feedback – Field, Button, Hyperlink.....	147
Table 96: View list skills – Field, Button, Hyperlink.....	148
Table 97: Create new skill modal – Field, Button, Hyperlink	149
Table 98: View plan report – Field, Button, Hyperlink.....	151
Table 99: Course detail page – Field, Button, Hyperlink.....	152
Table 100: Manage lecture section – Field, Button, Hyperlink	154
Table 101: Manage lecture content page – Field, Button, Hyperlink.....	155
Table 102: Entity data dictionary	157
Table 103: Data Table Dictionary	162
Table 104: Data Table Description.....	167
Table 105: Test case <System Admin> Add branch	168
Table 106: Test case <Manager> Create user	169
Table 107: Test case <Author> Create course – Create lecture – Create quiz.....	172
Table 108: Test case <Learner> Search Course – Update Profile.....	175
Table 109: Hardware requirement for server.....	176
Table 110: Hardware requirement for client.....	176
Table 111: Software requirement for server.....	176
Table 112: Software requirement for client	176
Table 113: Create plan – User manual.....	185
Table 114: Update plan – User manual	187
Table 115: Create program – User manual.....	188
Table 116: Create course – User manual.....	189
Table 117: Update course – User manual.....	190
Table 118: Create quiz – User manual	191
Table 119: Update profile – User manual.....	192
Table 120: Learning quiz – User manual	192

List of Figure

Figure 1: Agile Scrum Methodology.....	19
Figure 2: System Overview Use Case.....	23
Figure 3: <Guest> Login (UC_TRAISY1).....	23
Figure 4: <Learner> Overview Use Case.....	25
Figure 5: <Learner> Logout (UC_TRAISY2)	25
Figure 6: <Learner> View plan list (UC_TRAISY3)	27
Figure 7: <Learner> View plan (UC_TRAISY4).....	28
Figure 8: <Learner> Update profile (UC_TRAISY5).....	30
Figure 9: <Learner> View Learning Activity (UC_TRAISY6).....	32
Figure 10: <Learner> Search course (UC_TRAISY7).....	33
Figure 11: <Learner> View course (UC_TRAISY8)	34
Figure 12: <Learner> View program (UC_TRAISY9).....	36
Figure 13: <Learner> Enroll course (UC_TRAISY10)	37
Figure 14: <Learner> Learn lecture (UC_TRAISY11)	39
Figure 15: <Learner> Take quiz (UC_TRAISY12)	41
Figure 16: <Learner> View quiz result (UC_TRAISY13)	43
Figure 17: <Learner> Add discussion (UC_TRAISY14).....	45
Figure 18: <Learner> Update discussion (UC_TRAISY15).....	46
Figure 19: <Learner> Remove discussion (UC_TRAISY16).....	48
Figure 20: <Author> Overview Use Case.....	49
Figure 21: <Author> View learner statistic (UC_TRAISY17)	49
Figure 22: <Author> Create course(UC_TRAISY18)	51
Figure 23: <Author> Remove course(UC_TRAISY19)	52
Figure 24: <Author> Update course (UC_TRAISY20).....	53
Figure 25: <Author> Add lecture (UC_TRAISY21)	55
Figure 26: <Author> Update lecture (UC_TRAISY22)	56
Figure 27: <Author> Remove lecture (UC_TRAISY23)	58
Figure 28: <Author> Update quiz (UC_TRAISY24)	59
Figure 29: <Author> Add question(UC_TRAISY25)	61
Figure 30: <Author> Update question(UC_TRAISY26).....	62
Figure 31: <Author> Remove question(UC_TRAISY27)	64
Figure 32: <Author> Add answer(UC_TRAISY28)	66
Figure 33: <Author> Update answer (UC_TRAISY28)	68
Figure 34: <Author> Remove answer (UC_TRAISY29)	69
Figure 35: <Author> Add section(UC_TRAISY30).....	71
Figure 36: <Author> Update section(UC_TRAISY31).....	72
Figure 37: <Author> Remove section (UC_TRAISY32).....	73
Figure 38: <System admin> Overview Use Case.....	75
Figure 39: <System Admin> Add brand (UC_TRAISY33)	75
Figure 40: <System Admin> Update brand (UC_TRAISY34).....	77
Figure 41: <System Admin> Update brand (UC_TRAISY35).....	78

Figure 42: <Manager> Overview Use Case	80
Figure 43: <Manager> View report (UC_TRAISY36)	81
Figure 44: <Manager> Update user's information (UC_TRAISY37)	82
Figure 45: <Manager> Create user (UC_TRAISY38).....	84
Figure 46: <Manager> Create plan (UC_TRAISY39).....	85
Figure 47: <Manager> Update plan (UC_TRAISY40)	87
Figure 48: <Manager> Add learner to plan (UC_TRAISY41).....	89
Figure 49: <Manager> Remove plan (UC_TRAISY42)	91
Figure 50: <Manager> Create program (UC_TRAISY43)	92
Figure 51: <Manager> Update program (UC_TRAISY44).....	94
Figure 52: <Manager> Remove program (UC_TRAISY45).....	95
Figure 53: <Manager> Create skill (UC_TRAISY46)	97
Figure 54: <Manager> Update skill (UC_TRAISY47).....	98
Figure 55: <Manager> Remove skill (UC_TRAISY48).....	100
Figure 56: <Manager> Add feedback (UC_TRAISY49).....	101
Figure 57: <Manager> Update feedback (UC_TRAISY50)	102
Figure 58: <Manager> Remove feedback (UC_TRAISY51)	104
Figure 59: <Admin> Overview Use Case.....	105
Figure 60: <Admin> Update brand (UC_TRAISY52)	106
Figure 61: Conceptual diagram.....	109
Figure 62: System architecture overview	111
Figure 64: Class Diagram.....	115
Figure 65: Sequence Diagram - <SystemAdmin> Create brand	124
Figure 66: Sequence Diagram - <SystemAdmin> Remove brand.....	125
Figure 67: Sequence Diagram - <Admin> Update brand	125
Figure 68: Sequence Diagram - <Manager> Update user information	126
Figure 69: Sequence Diagram - <Manager> Add learner to plan.....	126
Figure 70: Sequence Diagram - <Manager> Create new user	127
Figure 71: Sequence Diagram - <Manager> Create plan.....	127
Figure 72: Sequence Diagram - <Manager> Update plan	128
Figure 73: Sequence Diagram - <Manager> Remove plan	128
Figure 74: Sequence Diagram - <Manager> Add author to plan.....	129
Figure 75: Sequence Diagram - <Manager> Create program	129
Figure 76: Sequence Diagram - <Manager> Update program.....	130
Figure 77: Sequence Diagram - <Manager> Remove program.....	130
Figure 78: Sequence Diagram - <Manager> View report	131
Figure 79: Sequence Diagram - <Author> Create lecture.....	131
Figure 80: Sequence Diagram - <Author> Update lecture	132
Figure 81: Sequence Diagram - <Author> Remove lecture	132
Figure 82: Sequence Diagram - <Learner> Log in.....	133
Figure 83: Sequence Diagram - <Learner> Learn lecture	133
Figure 84: Login Page.....	134
Figure 85: List participated plans page.....	135

Figure 86: Plan's overview page.....	136
Figure 87: User's information page.....	137
Figure 88: Take quiz page.....	138
Figure 89: Brand detail page (1).....	139
Figure 90: Brand detail page (2).....	139
Figure 91: View list user.....	140
Figure 93: Manage course (1).....	142
Figure 94: Manage course (2).....	142
Figure 107: Entity data dictionary.....	159

Definitions, Acronyms, and Abbreviations

Name	Definition
CTS	Corporate Training System

A. Report No. 1 Introduction

1. Project Information

- Project name: **Corporate Training System**
- Project Code: **CTS**
- Product Type: **Website application**
- Start Date: **8th January, 2018**
- End Date: **April, 2018**

2. Introduction

In this document, we will introduce a solution for corporates to train their employees more effectively. Nowadays, many companies are having problem to find a system that can provide them an environment to train their staffs easier.

Based on our research and analysis, we proposed a solution for this problem. We build a system, which allows any corporates to post their training program on our website. Through our system, staffs can access to training anywhere and anytime, making learning and training more effective.

Besides, this document also shows our working process and our perspective in the system, designs, architecture and workflow.

3. Current Situation

In recent years, e-learning or studying online is becoming more and more widely used. Corporations with an online training system find out their employees are training faster and better than those who do not have access to online learning. However, it seems not easy for each company to build their own training system.

Whenever a corporate want to develop a system to train their employees, they usually hire a software company to build a system that suitable for their needs with a cheap price. Nevertheless, it's hard to seek a software company to acquire the two requirements above.

4. Problem Definition

From above current situation, there are some disadvantages:

- There are not many systems for corporates to train its employees online
- Learner cannot communicate with author to understand the content of that course if they get trouble in learning.
- Learner have to wait for their manager to accept the request before they can begin to learn a course.

- When an author creates a course, they do not need to receive a permission from manager or administrator. Therefore, that course may have a bad influence on learners or inappropriate content.
- An author maybe wonders what plan or program that their courses belongs to.

5. Proposed Solution

Our proposed solution is to build a system named Corporate Training System (CTS) who has troubles with finding a place to train their employees online. Corporate Training System is a web application with following functions:

5.1 Feature functions

- Manage plan, programs, skills and courses.
- Manage users.
- Manage corporate brand and information.
- Participate to a plan or program.
- Enroll in a course.
- Learn a lecture.
- Take quiz and view quiz result.
- Make statistic report.
- View learners' report on learning activities
- Manage personal information
- Manage discussion.

5.2 Advantages and Disadvantages

- Advantages:
 - Necessary for all corporate
 - New idea
 - Easy to use application
 - Friendly interface
- Disadvantages:
 - Finding a suitable price for customer.
 - The cost in web application maintenance is high
 - Users' demand change

6. Functional Requirements

Function requirements of the system are listed as below:

- Learner component:
 - View program
 - View study plan
 - Search study course
 - View study course
 - Enroll course

- Manage profile
- Manage discussion
- View learning activity
- Sign in, Sign out
- Author component:
 - View learner statistic
 - Manage course
- Manager component:
 - Manage training plan
 - Manage user
 - Manage program
 - Manage skill
 - View learners' report
 - Add learner to program/plan
 - Remove learner from program/plan
- Admin component:
 - Manage corporate's information and brand
 - Change user's role
 - Ban user from study

7. Role and Responsibility

No	Full Name	Role	Position	Contact
1	Nguyễn Huy Hùng	Project Manager	Supervisor	hungnh@fpt.edu.vn
2	Lê Phúc Thịnh	Developer	Leader	thinhlpse61759@fpt.edu.vn
3	Võ Thạch Nguyên	Developer	Member	nguyenvtse61778@fpt.edu.vn
4	Phạm Chánh Hưng	Developer	Member	hungpcse61807@fpt.edu.vn
5	Đặng Thế Anh	Developer	Member	anhdtse61372@fpt.edu.vn

Table 1: Roles and Responsibilities

B. Report No. 2 Software Project Management Plan

1. Problem Definition

1.1 Name of this Capstone Project

- Official name: Corporate Training System
- Vietnamese name: Hệ thống đào tạo cho doanh nghiệp
- Abbreviation: CTS

1.2 Problem Abstract

This project is our concern about development of training online for corporation. We call it Corporate Training System (Traisy). As current in Vietnam, as well as other developed countries, websites and applications for e-study are limited in functions and often focus on only certain companies. For example, many platforms for learning online only contains video lecture. Meanwhile, some systems support many functions but only available to a specific company, such as SAKAI for FPT SOFTWARE.

Those websites that focus on several tasks for e-learning are often designed for international users and thus, are not particularly suitable for Vietnamese people.

The solution is to provide a system for all corporates, especially Vietnamese Corporates, a way to satisfy all their needs of a simple and effective e-training for employees.

1.3 Project Overview

1.3.1 Current Situation

Below are some problems encountered in this project:

- **Business logic:** Some scopes of project are not clear to every team member in early stage of developments. For example, learners cannot self-enroll their selves into a training program but being enrolled by their manager or admin.
- **Schedule of team member:** team members can have a conflict in meeting schedule because of sick, or class and work schedule, etc.
- **Framework study:** team members have a problem when applying the play framework into project. The team needs an amount of time to get familiar new techniques.
- **Some tools require premium package to access:** For convenience for play framework, we use IntelliJ IDE, which requires premium account and expensive for our team members.
- **Lack of UI (user interface), UX (user experience) design skill:** Our team members all study IS major and no one has studied UI, UX design. Therefore, that some UI may misunderstand or hard to use with normal user.

- **Knowledge about training plan:** this project is about how a corporation or company trains their employees. The knowledge of this issue is new to members.

1.3.2 The Proposed System

The proposed system is Corporate Training System. The system contains a web application for learner, author, manager and admin.

In detail, the web application provides following features:

- For learners:
 - Learners can view study courses. In this function, they can view all resources and lectures of a course they are assigned and through this, they can study any lecturer.
 - Learners can view their learning activities. After a course is enrolled for learners, they can view process of their study.
 - Learners can update their info and manage their discussion.
- For author:
 - Authors can manage their courses. Author has full function to edit their lessons such as update, delete and add.
 - Author can view learner statistic. In this function, author can view all learners' progress who takes their lectures. By doing this, author can easily improve their lesions.
- For manager:
 - Manager can manage learners. Manager can add a learner to a program/plan or remove learner from any program/plan of their company.
 - Manager can create, edit or delete a skill.
- For admins:
 - Manage corporate's information. Admin can edit their corporate's related information and change company's logo or brand name.
 - Admin can edit all users' role.

1.3.3 Boundaries of the System

This section supposes that users of the system have stable Internet connection while using.

The system can:

- Allow any corporate to register their training program online.
- Allow admin, manager, author to create courses
- Allow only admin to change role of users

- Learners can search for courses in plans that they already participated in

Besides, the system cannot:

- Let the learners to self-enroll their selves
- Let the learners to register their accounts

1.3.4 Development Environment

1.3.4.1 Hardware requirements

For server

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

Table 2: Hardware Requirement for Server

For PC

PC	Minimum Requirements	Recommended
Internet Connection	Cable, Wi-Fi (4 Mbps)	Cable, Wi-Fi (8 Mbps)
Operating System	Window 7	Window 7 or more.
Computer Processor	Intel® Core i3 1.4GHz	Intel® Core i5 2.50GHz
Computer Memory	1GB RAM	2GB RAM or more
Web Browser	Firefox (v52 or higher), Chromes (v28 or higher)	Chrome latest stable version

Table 3: Hardware Requirement for PC.

1.3.4.2 Software requirements

Name	Name/Version	Description
Environment	Java JDK 8	Specification for developing web application
Operating System	Window 10	Operating system and platform for development
Integrated development environment	- IntelliJ IDEA 2017.2.3 - Bracket (Front-end Application Development)	Programming tools
Modeling tool	StarUML 2.8, Lucidchart.com	Software modeling tool

Database Management System	MySQL 5.7	Used to create & manage the database for system
Source control	Git hosted by Github.com	Used to source control & version control.
Web browser	Chrome 28 or higher.	Testing browser

Table 4: Software requirement.

2. Project organization

2.1 Software Process Model

This project is developed under Scrum model (Scrum Methodology). We apply Scrum model to capable with current situation in our team. We choose this model due to following reasons:

- + Easy in tracking process: In Scrum, a product backlog (a wish list of tasks that need to be prioritized in a project) is created. Thanks to product backlog, our team can create sprint backlog so that we can tracking member's tasks easier in each sprint.
- + Quality: We agreed with each other that sprint is one week. In specific week, we always have a sprint planning meeting that takes place at the beginning of each sprint, a daily meeting that is held for a brief period (about 15 minutes) each day to allow each member to brief the team on what they did the previous day and a sprint review meeting, this meeting is held at the end of each sprint. With all of this meeting, we can review and guarantee quality of all our works.
- + Fast and convenient: Scrum Methodology is very suitable with our project and team because scrum can work with any technology/programming language.
- + Easy to manage: Due to short sprints and constant feedback, it becomes easier to cope with the changes. Besides, it is also a lightly controlled method which insists on frequent updating of work progress through regular meetings. Thus, there is clear visibility of the development project.

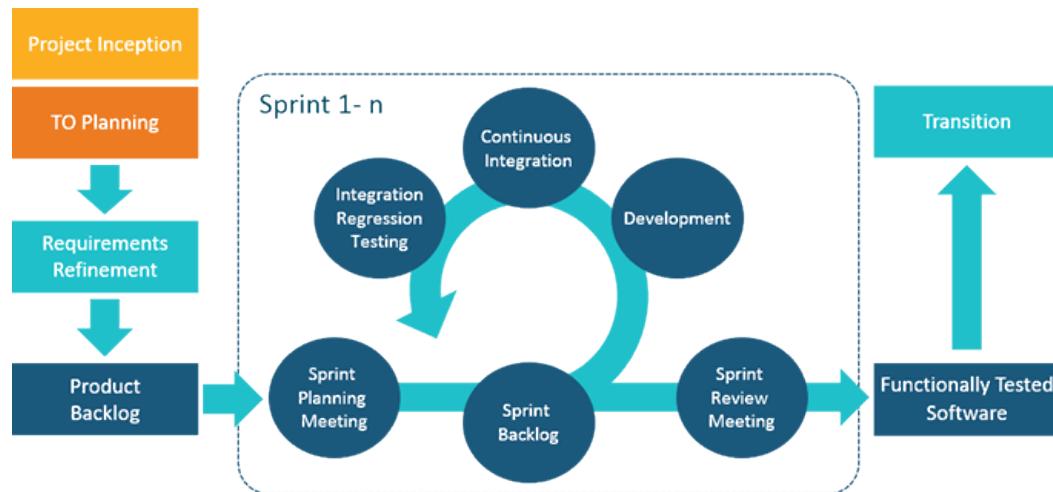


Figure 1: Agile Scrum Methodology

For more information:

- <http://scrummethodology.com/>

2.2 Roles and responsibilities

No	Full name	Role in Group	Responsibilities
1	Nguyễn Huy Hùng	Project manager	<ul style="list-style-type: none"> Specify user requirement Control the development process Give out technique and business analysis support
2	Lê Phúc Thịnh	Team Leader, BA, DEV, Tester	<ul style="list-style-type: none"> Managing process Designing database Clarifying requirements Prepare documents GUI Design Create test plan Coding Testing
3	Võ Thạch Nguyên	Team member, BA, DEV, Tester	<ul style="list-style-type: none"> Designing database Clarifying requirements Prepare documents GUI Design Create test plan Coding Testing

4	Phạm Chánh Hưng	Team member, BA, DEV, Tester	<ul style="list-style-type: none"> • Designing database • Clarifying requirements • Prepare documents • GUI Design • Create test plan • Coding • Testing
5	Đặng Thế Anh	Team member, BA, DEV, Tester	<ul style="list-style-type: none"> • Designing database • Clarifying requirements • Prepare documents • GUI Design • Create test plan • Coding • Testing

Table 5: Roles and Responsibilities Details

2.3 Tools and Techniques

	Tools	Techniques
Front-end	Bracket	HTML5 CSS3 Javascript jQuery Bootstrap
Back-end	IntelliJ IDEA	JavaEE Play Framework JPA Wowza streaming engine

Table 6: Tools and Techniques

3. Project Management Plan

3.1 Product Backlog

Refer to Appendix

3.2 Sprint Backlog

Refer to Appendix

3.3 All Meeting Minutes

<https://drive.google.com/drive/folders/1WAf1QdqQeW3tomYAzrgIgt0PDhu-GKNg?usp=sharing>

4. Coding Convention

Java: Using to develop web application and web service.

- Naming Convention:

- + Follows camelcase syntax for naming the class, interface, method and variable.

- + If name is combined with two words, second word will start with uppercase letter always e.g. actionPerformed(), firstName, ActionEvent, ActionListener etc.
- Method names should be verbs.
- Class names should be nouns, in mixed case with first letter of each internal word capitalized.
- Constant names should be all uppercase with words separated by underscore.
- Comment:
 - Using /* */ for block comments.
 - Using // for line comments.
- Follow Field Naming Conventions
 - + Other fields start with a lowercase letter.
 - + Public static final fields (constants) are ALL_CAPS_WITH_UNDERSCORES.

Using Java coding convention from:

<http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>

C. Report No. 3 Software Requirement Specification

1. User Requirement Specification

1.1 Guest Requirement

Guest is a person who doesn't access to the system. Guest can use some functions in the system, to use all functions, guest must login. These are some functions guest can use:

- Login

1.2 Learner Requirement

Learner is a person who access to the system as learner role. Learner can use following functions:

- Login
- View course
- View plan
- View program
- Learn lecture
- Take quiz
- View learning activity
- Manage discussion

1.3 Author Requirement

- View learner statistic
- Manage course
- Manage lecture
- Manage section
- Manage quiz

1.4 System Admin Requirement

- Manage brand

1.5 Manager Requirement

- View report
- Manage user
- Manage plan
- Manage program
- Manage skill
- Manage feedback

2. System Requirement Specification

2.1 External Interface Requirement

2.1.1 User Interface

- The user interface uses Vietnamese.
- The user interface displays best on 1366x768 resolutions on desktop.

2.1.2 Hardware Interface

Desktop PC.

2.1.3 Software Interface

Web application: work with browsers Firefox (v52 or higher), Chromes (v28 or higher), Internet Explorer (v10 or above) or with any web browser that supports HTML5 & CSS3

2.1.4 Communication Protocol

Use HTTP protocol 1.1 for communication between the web browser and the web.

2.2 System Overview Use Case

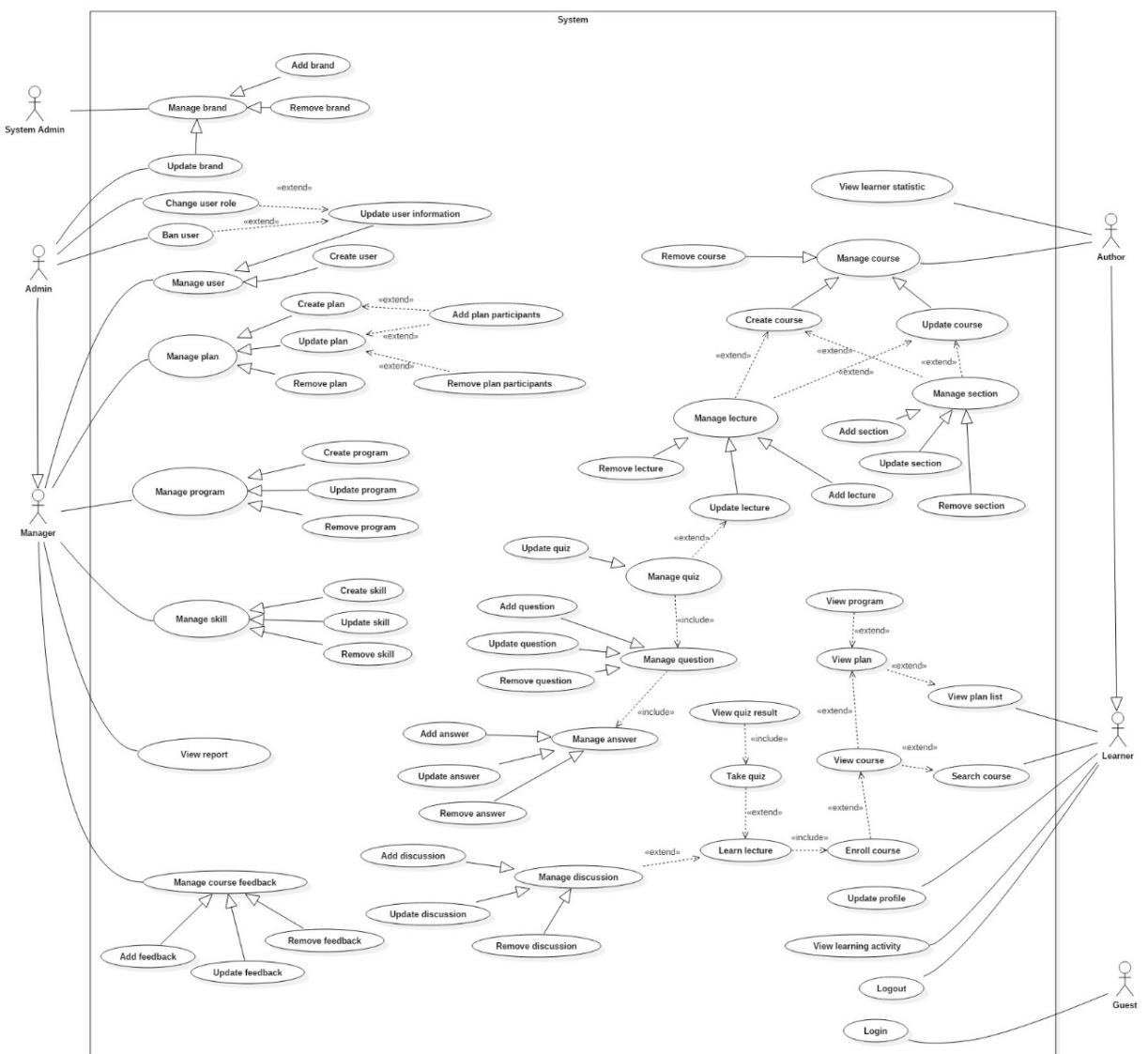


Figure 2: System Overview Use Case

2.3 List of Use Case

2.3.1 <Guest> Overview Use Case

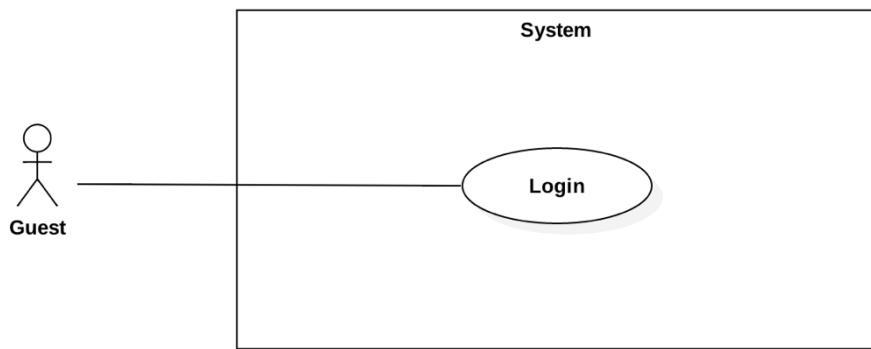


Figure 3: <Guest> Login (UC_TRAISY1)

USE CASE - UC_TRAISY1															
Use Case No.	UC_TRAISY1	Use Case Version	0.2												
Use Case Name	Login														
Author	Đặng Thế Anh														
Date	23/3/2018	Priority	High												
Actor:	<ul style="list-style-type: none"> Guest 														
Summary:	<ul style="list-style-type: none"> This use case allows authorized guest to login to system. 														
Goal:	<ul style="list-style-type: none"> To login to the system. 														
Triggers:	<ul style="list-style-type: none"> Actor sends login command. 														
Preconditions:	<ul style="list-style-type: none"> User does not login yet. 														
Post conditions:	<ul style="list-style-type: none"> Success: Actor accesses the system successfully. Fail: System shows error messages. 														
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Guest goes to login page</td><td> System requires information: <ul style="list-style-type: none"> Tên đăng nhập: text input Mật khẩu: text input </td></tr> <tr> <td>2</td><td>Guest inputs information (Username and password)</td><td></td></tr> <tr> <td>3</td><td>Guest sends command to login (Click “Đăng nhập” button after input login form).</td><td> [Exception no.1] [Exception no.2] System displays the home page. </td></tr> </tbody> </table>			Step	Actor Action	System Response	1	Guest goes to login page	System requires information: <ul style="list-style-type: none"> Tên đăng nhập: text input Mật khẩu: text input 	2	Guest inputs information (Username and password)		3	Guest sends command to login (Click “Đăng nhập” button after input login form).	[Exception no.1] [Exception no.2] System displays the home page.
Step	Actor Action	System Response													
1	Guest goes to login page	System requires information: <ul style="list-style-type: none"> Tên đăng nhập: text input Mật khẩu: text input 													
2	Guest inputs information (Username and password)														
3	Guest sends command to login (Click “Đăng nhập” button after input login form).	[Exception no.1] [Exception no.2] System displays the home page.													
Alternative Scenario:	N/A														
Exceptions:	<table border="1"> <thead> <tr> <th>No</th><th>Cause</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Guest inputs wrong information.</td><td>System notices that “Thông tin đăng nhập không hợp lệ”</td></tr> <tr> <td>2</td><td>Guest has been inactive by admin/manager</td><td>System notices that “Tài khoản đã bị khoá, vui lòng liên hệ quản trị/quản lý để mở khoá”</td></tr> </tbody> </table>			No	Cause	System Response	1	Guest inputs wrong information.	System notices that “Thông tin đăng nhập không hợp lệ”	2	Guest has been inactive by admin/manager	System notices that “Tài khoản đã bị khoá, vui lòng liên hệ quản trị/quản lý để mở khoá”			
No	Cause	System Response													
1	Guest inputs wrong information.	System notices that “Thông tin đăng nhập không hợp lệ”													
2	Guest has been inactive by admin/manager	System notices that “Tài khoản đã bị khoá, vui lòng liên hệ quản trị/quản lý để mở khoá”													
Relationships:	N/A														
Business Rules:	<ul style="list-style-type: none"> After signing in to system, guest will be redirected to specific view based on their role on the system: System Admin, Admin, Manager, Author or Learner. <ul style="list-style-type: none"> If role is “System Admin”, the system will display System Admin view. 														

- If role is “Admin”, the system will display Admin view.
- If role is “Manager”, the system will display Manager View.
- If role is “Author”, the system will display Author view.
- If role is “Learner”, the system will navigate to Learner view
- Username must be at least 5 characters.
- Password must be at least 4 characters.
- Password must be encrypted.

Table 7: Login specification

2.3.2 <Learner> Overview Use Case

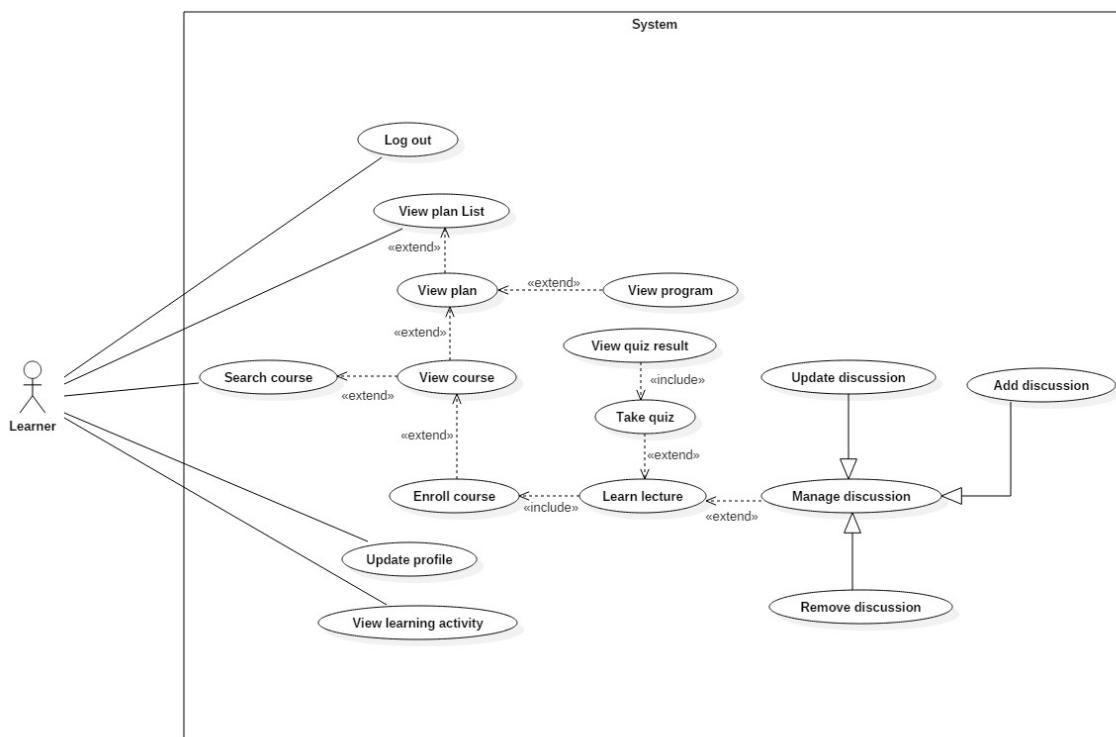


Figure 4: <Learner> Overview Use Case

2.3.2.1 <Learner> Log out

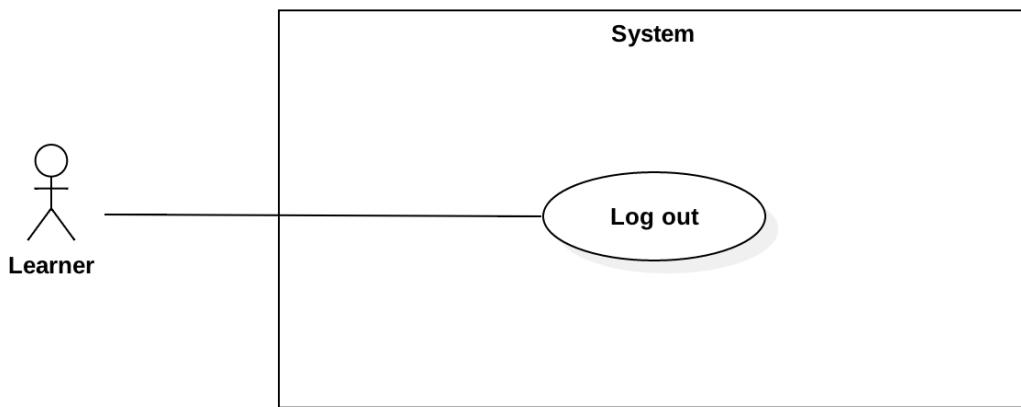


Figure 5: <Learner> Logout (UC_TRAISY2)

USE CASE – UC_TRAISY2

Use Case No.	UC_TRAISY2	Use Case Version	0.2
Use Case Name	Logout		
Author	Đặng Thế Anh		
Date	23/03/2013	Priority	High
Actor:			
<ul style="list-style-type: none"> Learner 			
Summary:			
<ul style="list-style-type: none"> This use case allows user to logout the system. 			
Goal:			
<ul style="list-style-type: none"> To logout the system. 			
Triggers:			
<ul style="list-style-type: none"> Actor sends logout command. 			
Preconditions:			
<ul style="list-style-type: none"> Learner logged in the system. 			
Post conditions:			
<ul style="list-style-type: none"> Success: User session is removed, and system displays login page. Fail: System displays error messages. 			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Actor clicks “Đăng xuất” button.	User session is removed, and system displays login page.	
Alternative Scenario: N/A			
Exceptions:			
No	Cause	System Response	
1	Learner is not participated in any plan.	System displays page with message: “Bạn chưa tham gia khóa học nào.”	
Relationships: N/A			
Business Rules:			
<ul style="list-style-type: none"> Actor has already accessed to the system. After signing out, Authenticated user will become “guest” 			

Table 8: Log out specification

2.3.2.2 <Learner> View plan list

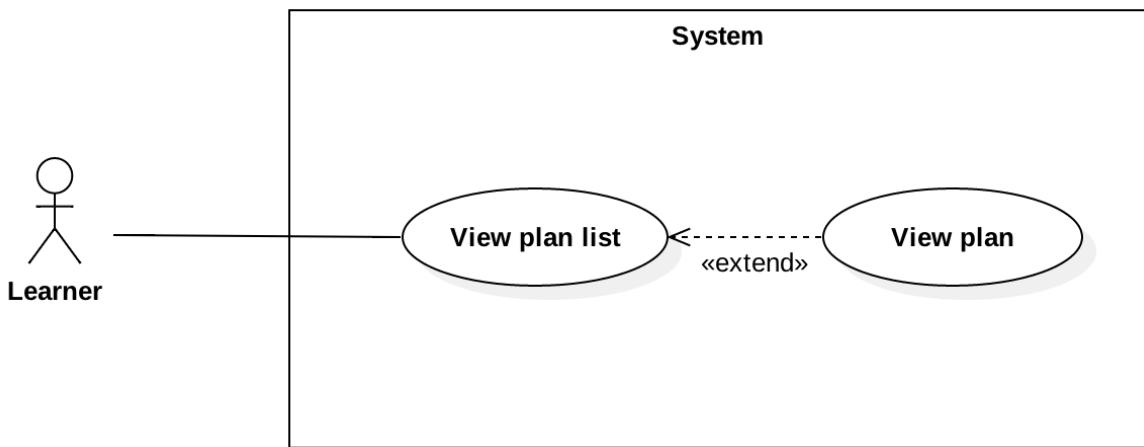


Figure 6: <Learner> View plan list (UC_TRAISY3)

USE CASE - UC_TRAISY3					
Use Case No.	UC_TRAISY3	Use Case Version	0.2		
Use Case Name	View plan list				
Author	Phạm Chánh Hưng				
Date	23/03/2013	Priority	High		
Actor:					
• Learner					
Summary:					
• This use case allows learners to view all their participated plans.					
Goal:					
• To view information of all plans that he/she has participated in.					
Triggers:					
• Actor sends view plan list command.					
Preconditions:					
• Learner logged in the system.					
Post conditions:					
• Success: System displays all participated plans.					
• Fail: System displays error messages.					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Actor clicks on “Kế hoạch của tôi”.	System navigates to list of plans that learner participated.			
Alternative Scenario:					
Step	Actor Action	System Response			
1	Actor clicks on “Kế hoạch của tôi” but actor has not participated in any plan.	System redirects to page with notice “Bạn chưa tham gia kế hoạch nào”.			
Exceptions: N/A					
Relationships:					

- Extended by: View Plan

Business Rules:

- User already logged in the system as “Learner” role.
- Learners only view plans that they already participated.
- Learners only view plans that have published by manager.
- Participated plans will have progress bar which indicates the learning activity of learner.
- Participated plans which are overdue are not accessible by learner.

Table 9: View plan list specification

2.3.2.3 <Learner> View plan

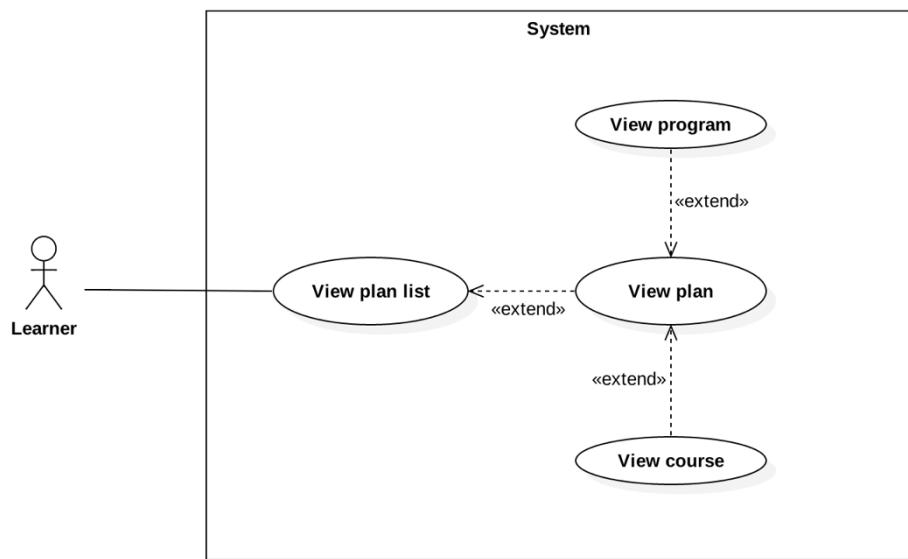


Figure 7: <Learner> View plan (UC_TRAISY4)

USE CASE - UC_TRAISY4			
Use Case No.	UC_TRAISY4	Use Case Version	0.2
Use Case Name	View plan		
Author	Phạm Chánh Hưng		
Date	23/03/2013	Priority	High
Actor:			
• Learner			
Summary:			
• This use case allows learners to view a participated plan.			
Goal:			
• To view information of a specific plan that he/she has participated in.			
Triggers:			
• Actor sends view plan command.			
Preconditions:			
• Learner logged in the system.			
Post conditions:			

- Success: System displays information of the participated plans.
- Fail: System displays error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor clicks on a specific plan	[Exception no.1] [Exception no.2]
		System navigates to the selected plan's page with details of plan.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner has already been removed from the selected plan.	System displays notice: "Bạn chưa tham gia vào khóa học". Then redirects learner to blank plan page.
2	The selected plan has been removed by admin/manager of that brand.	System displays notice: "Khóa học đã bị xóa bởi người quản lý". Then redirects learner to blank plan page

Relationships:

- Extend from: View Plan List
- Extended by: View course
- Extended by: View program

Business Rules:

- User already logged in the system as "Learner" role.
- Learner must be a participant to view the selected plan.
- The selected plan must be active (In duration time and not be deleted).

Table 10: View plan specification

2.3.2.4 <Learner> Update profile

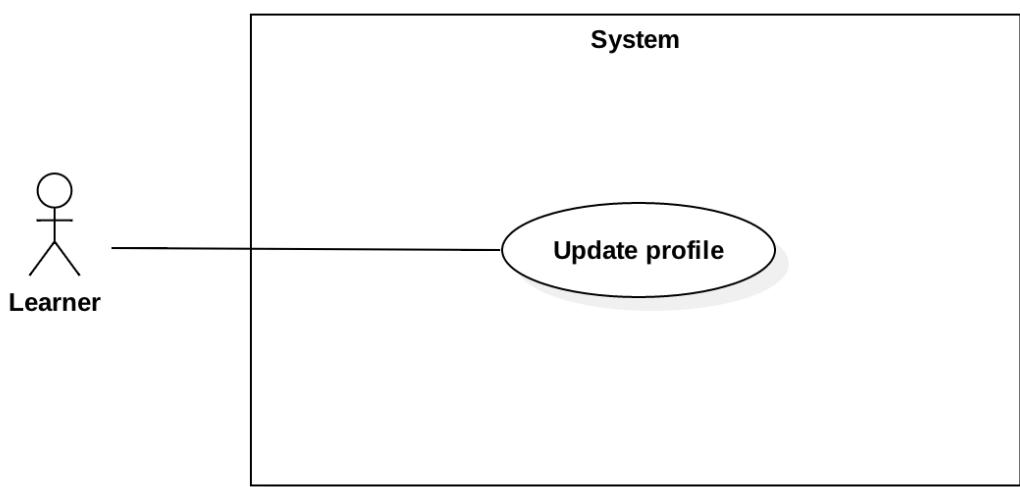


Figure 8: <Learner> Update profile (UC_TRAISY5)

USE CASE - UC_TRAISY5					
Use Case No.	UC_TRAISY5	Use Case Version	0.2		
Use Case Name	Update profile				
Author	Phạm Chánh Hưng				
Date	23/03/2013	Priority	High		
Actor:					
<ul style="list-style-type: none"> Learner 					
Summary:					
<ul style="list-style-type: none"> This use case allows learners to update their information. 					
Goal:					
<ul style="list-style-type: none"> To update profile. 					
Triggers:					
<ul style="list-style-type: none"> Learner sends update profile command. 					
Preconditions:					
<ul style="list-style-type: none"> Learner logged in the system. 					
Post conditions:					
<ul style="list-style-type: none"> Success: Account is updated successfully, and system displays the user profile page. Fail: System displays error messages. 					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Actor clicks “Thông tin người dùng”	System displays user information page with: <ul style="list-style-type: none"> Họ tên: Text Input Chức danh: Text Input Mật khẩu hiện tại: Text Input Mật khẩu mới: Text Input Xác nhận mật khẩu: Text Input 			
2	Actor clicks “Cập nhật” button in the update profile page.	[Exception 1] [Exception 2]			
		Account is updated successfully and system redirects the user to profile page.			
Alternative Scenario: N/A					
Exceptions:					
No	Cause	System Response			
1	Learner does not input required fields.	System notices that guest need to input all these field: - “Họ tên”: System displays warning message: “Họ tên không được trống”.			

		<ul style="list-style-type: none"> - “Chức danh”: System displays warning message: “Chức danh không được trống”. - “Mật khẩu mới”: System displays warning message: “Mật khẩu không được trống”. - “Xác nhận mật khẩu”: System displays warning message: “Xác nhận mật khẩu không được trống”.
2	Learner input wrong some fields with requirement.	<p>System notices that guest need to re-input all these field:</p> <ul style="list-style-type: none"> - “Mật khẩu”: System displays warning message: “Mật khẩu phải từ 4 tới 30 ký tự”. - “Xác nhận mật khẩu”: System displays warning message: “Mật khẩu và xác nhận mật khẩu phải giống nhau”.

Relationships: N/A

Business Rules:

- User already logged in the system as “Learner” role.
- If learner wants to update password, he/she has to enter correct current password, and confirm password.
- Learner can update Fullname and Title without enter current password.
- Fullname is not null and in range of 4-45 characters.
- Title is at least 4 characters.
- Password must be at least 4 characters.

Table 11: Update profile specification

2.3.2.5 <Learner> View Learning Activity

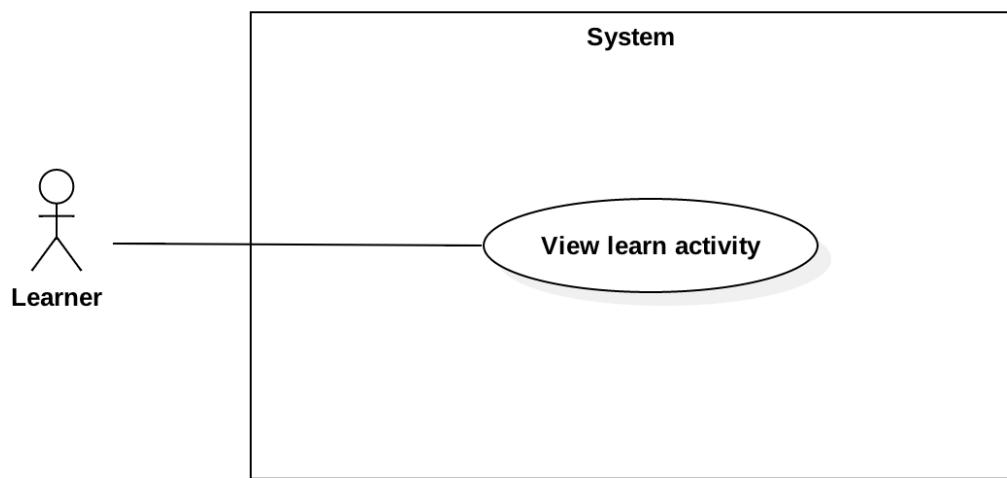


Figure 9: <Learner> View Learning Activity (UC_TRAISY6)

USE CASE - UC_TRAISY6									
Use Case No.	UC_TRAISY6	Use Case Version	0.2						
Use Case Name	View Learning Activity								
Author	Phạm Chánh Hưng								
Date	23/3/2018	Priority	High						
Actor:	<ul style="list-style-type: none"> Learner 								
Summary:	<ul style="list-style-type: none"> This use case allows authorized learner to view process of his/her learning activity. 								
Goal:	<ul style="list-style-type: none"> To view process of learning. 								
Triggers:	<ul style="list-style-type: none"> Actor sends view process command. 								
Preconditions:	<ul style="list-style-type: none"> Learner logged in the system. 								
Post conditions:	<ul style="list-style-type: none"> Success: Learners can see their learning process. Fail: System shows error messages. 								
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Actor clicks to any plan or course in plan/course page</td><td>System displays learning activity/process bar for each course and plan that learner has studied.</td></tr> </tbody> </table>			Step	Actor Action	System Response	1	Actor clicks to any plan or course in plan/course page	System displays learning activity/process bar for each course and plan that learner has studied.
Step	Actor Action	System Response							
1	Actor clicks to any plan or course in plan/course page	System displays learning activity/process bar for each course and plan that learner has studied.							
Alternative Scenario:	N/A								
Relationships:	N/A								
Business Rules:	<ul style="list-style-type: none"> User already logged in the system as "Learner" role. Learner must participate in plan to view learning activity of plan. Learner must enroll in course to view learning activity of course. Learner must study lecture in order to view learning process. If learner hasn't studied the course, the learning process will not appear, enroll button will appear instead of process bar for that course. The process bar of plan will be calculated based on finished programs of that plan. The process bar of course will be calculated based on finished lectures of that course. 								

Table 12: View learning activity specification

2.3.2.6 <Learner> Search course

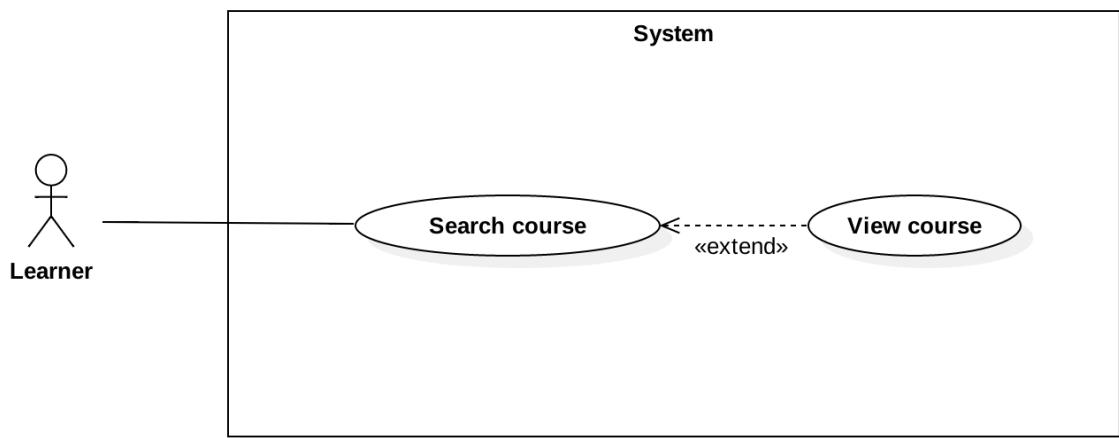


Figure 10: <Learner> Search course (UC_TRAISY7)

USE CASE - UC_TRAISY7			
Use Case No.	UC_TRAISY7	Use Case Version	0.2
Use Case Name	Search course		
Author	Phạm Chánh Hưng		
Date	24/3/2018	Priority	High
Actor:			
• Learner			
Summary:			
• This use case allows authorized learner to search course of plan that he/she has participated in.			
Goal:			
• To search course.			
Triggers:			
• Actor sends search course command.			
Preconditions:			
• Learner logged in the system.			
Post conditions:			
• Success: Learners can see courses with title similar to keyword that learners input. • Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Actor inputs keyword to searching form and clicks “Tìm khóa học”.	System displays courses of participated plans with title similar to keyword input.	
Alternative Scenario:			
Step	Actor Action	System Response	

1	Actor input keyword to searching form and clicks “Tìm khóa học”.	System displays no result found with notice “Không tìm thấy khóa học nào”.
---	--	--

Relationships:

- Extended by: View Course

Business Rules:

- User already logged in the system as “Learner” role.
- Learner must participate in plan to search for courses that published in that plan
- Courses have to be published and active (not overdue) to be searched by learner

Table 13: Search course specification

2.3.2.7 <Learner> View course

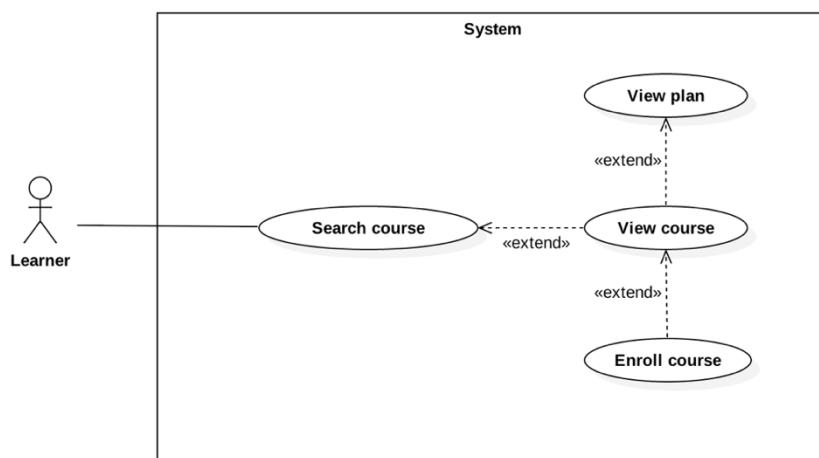


Figure 11: <Learner> View course (UC_TRAISY8)

USE CASE - UC_TRAISY8			
Use Case No.	UC_TRAISY8	Use Case Version	0.2
Use Case Name	View course		
Author	Lê Phúc Thịnh		
Date	24/3/2018	Priority	High
Actor:	<ul style="list-style-type: none"> Learner 		
Summary:	<ul style="list-style-type: none"> This use case allows authorized learner to view information of a specific course 		
Goal:	<ul style="list-style-type: none"> To view information of course. 		
Triggers:	<ul style="list-style-type: none"> Actor sends view course command. 		
Preconditions:	<ul style="list-style-type: none"> Learner logged in the system. 		

Post conditions:

- Success: Learners can see information of that course.
- Fail: System shows error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor clicks on a specific course	<p>[Exception no.1] [Exception no.2] [Exception no.3]</p> <p>System navigates to the selected course's page with details of that course:</p> <ul style="list-style-type: none"> • Title of course • Author of course • Start-end date of course • Skills that course belong to • List of lectures that published

Alternative Scenario:

Step	Actor Action	System Response
1	Actor clicks on an empty course	System navigates to the selected course's page with no detail of that course.

Exceptions:

No	Cause	System Response
1	Learner has already been removed from the selected course or plan.	System displays notice: "Bạn chưa tham gia vào khóa học". Then redirects learner to a blank course page.
2	The selected course has been removed by admin/manager or author of that brand.	System displays notice: "Khóa học đã bị xóa bởi người quản lý". Then redirects learner to a blank course page
3	The plan that course belong to has been removed by admin	System displays notice: "Kế hoạch đã bị xóa bởi người quản lý". Then redirects learner to a blank course page

Relationships:

- Extended from: View Plan
- Extended by: Enroll course
- Extend from: Search course

Business Rules:

- User already logged in the system as "Learner" role.

- Learner must participate in plan that course belongs to in order to view information of course.
- Courses have to be published and active (not overdue) to be viewed by learner.
- If learner hasn't enrolled in course, the course will render page with "Enroll" button, otherwise, course will render page with "Continue learning" button.

Table 14: View course specification

2.3.2.8 <Learner> View program

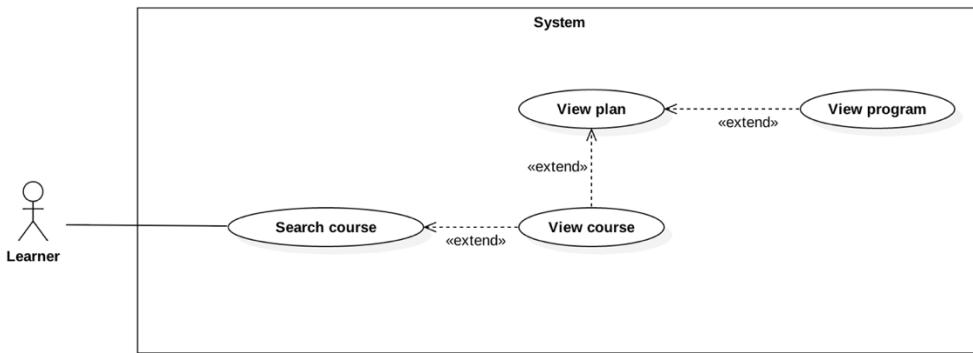


Figure 12: <Learner> View program (UC_TRAISY9)

USE CASE – UC_TRAISY9			
Use Case No.	UC_TRAISY9	Use Case Version	0.2
Use Case Name	View program		
Author	Lê Phúc Thịnh		
Date	24/3/2018	Priority	High
Actor:			
• Learner			
Summary:			
• This use case allows authorized learner to view information of a specific program			
Goal:			
• To view information of program.			
Triggers:			
• Actor sends search program command.			
Preconditions:			
• Learner logged in the system.			
Post conditions:			
• Success: Learners can see information of that program			
• Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Actor clicks on a specific program	[Exception no.1] [Exception no.2]	

		System navigates to the selected program's page with title of that program and list of active courses that program contains.
--	--	--

Alternative Scenario:

Step	Actor Action	System Response
1	Actor clicks on an empty program	System navigates to the selected program's page with only title of that program.

Exceptions:

No	Cause	System Response
1	Learner has already been removed from the selected plan.	System displays notice: "Bạn chưa tham gia vào khóa học". Then redirects learner to a blank plan page.
2	The selected plan has been removed by admin/manager or author of that brand.	System displays notice: "Kế hoạch đã bị xóa bởi người quản lý". Then redirects learner to a blank course page

Relationships:

- Extend from: View plan

Business Rules:

- User already logged in the system as "Learner" role.
- Learner must participate in plan that program belongs to in order to view information of program.
- Plan of the specific program must be active for learner to view information.
- Program with no courses published or activated will not be appeared to learner.

Table 15: View program specification

2.3.2.9 <Learner> Enroll course

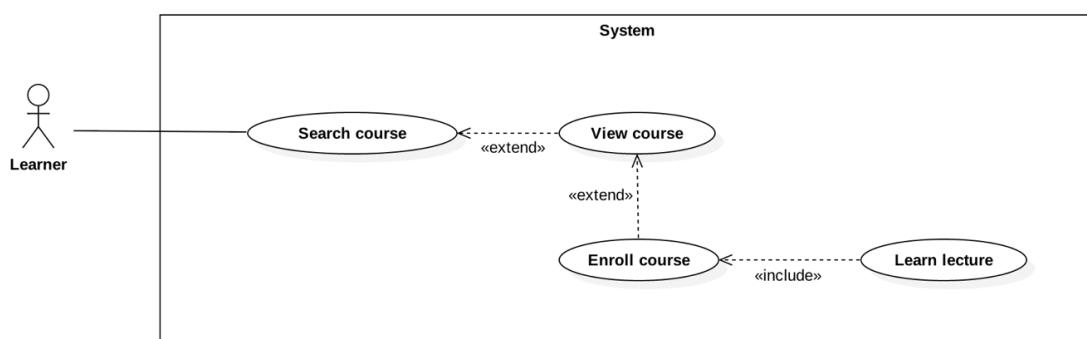


Figure 13: <Learner> Enroll course (UC_TRAISY10)

USE CASE - UC_TRAISY10			
Use Case No.	UC_TRAISY10	Use Case Version	0.2

Use Case Name	Enroll course		
Author	Lê Phúc Thịnh		
Date	24/3/2018	Priority	High

Actor:

- Learner

Summary:

- This use case allows authorized learner to enroll in a specific course

Goal:

- To enroll in course.

Triggers:

- Actor clicks “Tham gia” button.

Preconditions:

- Learner logged in the system.

Post conditions:

- Success: Learners are redirected to learning lecture page
- Fail: System shows error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor clicks “Tham gia” button of a specific course	[Exception no.1] [Exception no.2] [Exception no.3] System navigates to learning page with the first published lecture of course.

Alternative Scenario:

Step	Actor Action	System Response
1	Actor clicks “Học tiếp” button of a specific course if actor already enroll in course	System navigates to learning page with the next published lecture of course that learner has studied.

Exceptions:

No	Cause	System Response
1	Learner has already been removed from the selected plan.	System displays notice: “Bạn chưa tham gia vào khóa học”. Then redirects learner to a blank learning page.
2	The selected plan has been removed by admin/manager or author of that brand.	System displays notice: “Kế hoạch đã bị xóa bởi người quản lý”. Then redirects learner to a blank learning page
3	The course that user enroll in does not have any publish lecture	System redirects learner to information of that course.

Relationships:

- Extend from: View course
- Include: Learn lecture

Business Rules:

- User already logged in the system as “Learner” role.
- Learner must participate in plan that courses belong to in order to enroll in selected course.
- Courses have to be published and active (not overdue) to be enrolled by learner.
- If learner already enrolled in course, the enroll button will be replaced by continue learning button.
- When ever learner enrolls in course, the learning process for this course is updated.

Table 16: Enroll course specification

2.3.2.10 <Learner> Learn lecture

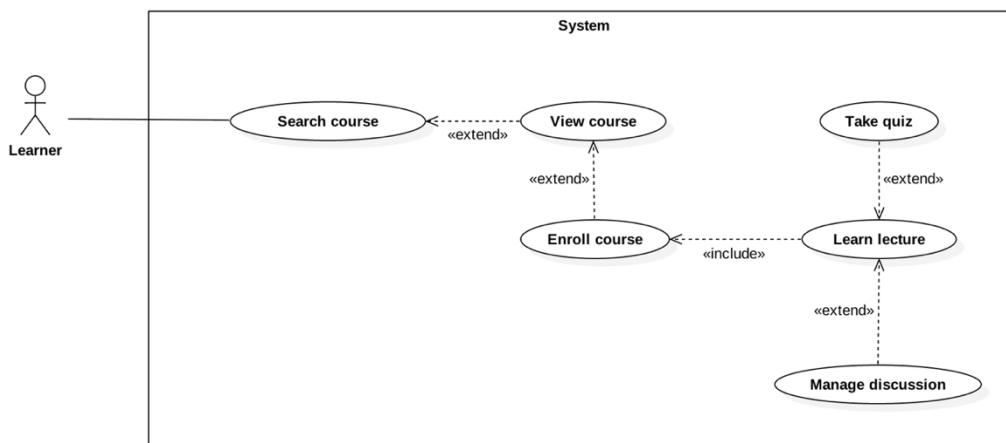


Figure 14: <Learner> Learn lecture (UC_TRAISY11)

USE CASE - UC_TRAISY11			
Use Case No.	UC_TRAISY11	Use Case Version	0.2
Use Case Name	Learn lecture		
Author	Lê Phúc Thịnh		
Date	24/3/2018	Priority	High
Actor:			
<ul style="list-style-type: none"> Learner 			
Summary:			
<ul style="list-style-type: none"> This use case allows authorized learner to learn a specific lecture 			
Goal:			
<ul style="list-style-type: none"> To learn a course. 			
Triggers:			
<ul style="list-style-type: none"> Actor sends learn lecture command. 			
Preconditions:			
<ul style="list-style-type: none"> Learner logged in the system. 			

Post conditions:

- Success: Learners are redirected to learning page of that lecture type
- Fail: System shows error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor clicks a specific lecture in a course.	[Exception no.1] [Exception no.2] [Exception no.3]
		System navigates to: <ul style="list-style-type: none"> • Learning slideshow page if that lecture is PowerPoint type • Learning video page if that lecture is video type • Learning text page if that lecture is of text type • Quiz page if that lecture is of quiz type

Alternative Scenario:

Scenario 1:

Step	Actor Action	System Response
1	Actor clicks "Tham gia" button in view course page	System navigates to learning page with the first published lecture of course.

Scenario 2:

Step	Actor Action	System Response
1	Actor clicks "Học tiếp" button in view course page	System navigates to learning page with the next published lecture of course that learner has studied.

Exceptions:

No	Cause	System Response
1	Learner has already been removed from the selected plan.	System displays notice: "Bạn chưa tham gia vào khóa học". Then redirects learner to a blank learning page.
2	The selected plan has been removed by admin/manager or author of that brand.	System displays notice: "Kế hoạch đã bị xóa bởi người quản lý". Then redirects learner to a blank learning page
3	The lecture that learner clicked has been removed or unpublished by author/manager/admin	System displays notice: "Bài học đã bị xóa bởi người quản lý". Then redirects learner to course page

Relationships:

- Extend by: Take quiz
- Extend by: Manage discussion
- Included in: Enroll course

Business Rules:

- User already logged in the system as “Learner” role.
- Learner must participate in plan that courses belong to in order to learn lecture
- Lectures have to be published and active (not overdue) to be accessed by learner
- Learner will be redirected to specific learning page depends on lecture type: Video, text, slideshow or quiz.
- If the learner clicks on finish studying lecture, the learning activity of learner for this lecture will be updated and the page will redirect to the next lecture in course. If not, this lecture will be counted as “Not finish learning”.

Table 17: Learn lecture specification

2.3.2.11 <Learner> Take quiz

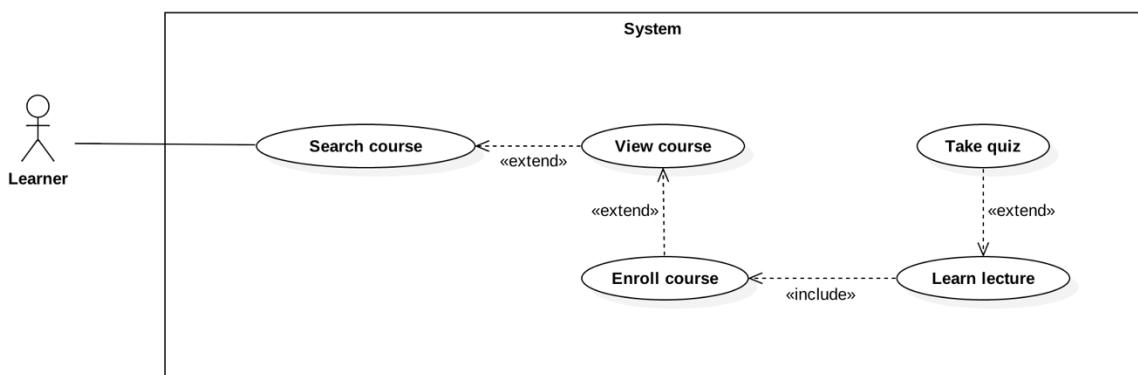


Figure 15: <Learner> Take quiz (UC_TRAISY12)

USE CASE – UC_TRAISY12			
Use Case No.	UC_TRAISY12	Use Case Version	0.2
Use Case Name	Take quiz		
Author	Lê Phúc Thịnh		
Date	24/3/2018	Priority	High
Actor:			
• Learner			
Summary:			
• This use case allows authorized learner to take quiz			
Goal:			
• To take a quiz.			
Triggers:			
• Actor sends take quiz command.			
Preconditions:			
• Learner logged in the system.			

Post conditions:

- Success: Learners are redirected to take quiz page
- Fail: System shows error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor clicks on a specific quiz in a course.	[Exception no.1] [Exception no.2] [Exception no.3]
		System navigates to: Quiz page

Alternative Scenario:

Scenario 1:

Step	Actor Action	System Response
1	Actor clicks "Tham gia" button in view course page	System navigates to quiz page if the first lecture of course is quiz.

Scenario 2:

Step	Actor Action	System Response
1	Actor clicks "Học tiếp" button in view course page	System navigates to quiz page to quiz page if the next lecture of course is quiz.

Exceptions:

No	Cause	System Response
1	Learner has already been removed from the selected plan.	System displays notice: "Bạn chưa tham gia vào khóa học". Then redirects learner to a blank learning page.
2	The selected plan has been removed by admin/manager or author of that brand.	System displays notice: "Kế hoạch đã bị xóa bởi người quản lý". Then redirects learner to a blank learning page
3	The quiz that learner clicked has been removed or unpublished by author/manager/admin	System displays notice: "Bài quiz đã bị xóa bởi người quản lý". Then redirects learner to course page

Relationships:

- Extend from: Learn lecture
- Included: View quiz result

Business Rules:

- User already logged in the system as "Learner" role.
- Learner must participate in plan that courses belong to in order to take quiz.
- Quiz have to be published and active (not overdue) to be accessed by learner.

- If the learner clicks on finish, the learning activity of learner for this quiz will be updated and the page will redirect to the next lecture in course. If not, this lecture will be counted as “Not finish learning”.
- At the last question of the quiz, there is a submit button. If learner clicks on submit button, the result of quiz will be calculated immediately and redirects user to result page of that quiz.

Table 18: Take quiz specification

2.3.2.12 <Learner> View quiz result

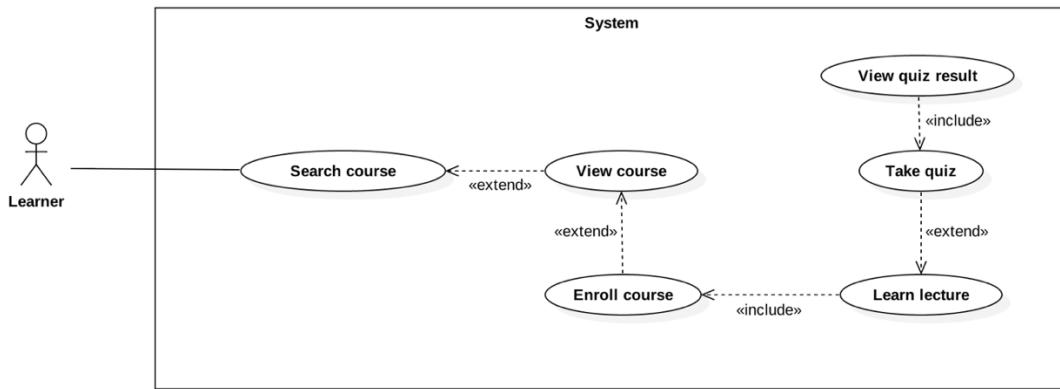


Figure 16: <Learner> View quiz result (UC_TRAISY13)

USE CASE - UC_TRAISY13			
Use Case No.	UC_TRAISY13	Use Case Version	0.2
Use Case Name	View quiz result		
Author	Lê Phúc Thịnh		
Date	24/3/2018	Priority	High
Actor:			
• Learner			
Summary:			
• This use case allows authorized learner to view result of quiz			
Goal:			
• To view result of quiz.			
Triggers:			
• Actor sends view result of quiz command.			
Preconditions:			
• Learner logged in the system.			
Post conditions:			
• Success: Learners are redirected to result of quiz page			
• Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Actor clicks on a specific quiz in a course.	[Exception no.1] [Exception no.2]	

		[Exception no.3]
		System navigates to quiz page.
2	Actor finishes a quiz	System navigates to quiz result page

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner has already been removed from the selected plan.	System displays notice: "Bạn chưa tham gia vào khóa học". Then redirects learner to a blank learning page.
2	The selected plan has been removed by admin/manager or author of that brand.	System displays notice: "Kế hoạch đã bị xóa bởi người quản lý". Then redirects learner to a blank learning page
3	The quiz that learner clicked has been removed or unpublished by author/manager/admin	System displays notice: "Bài quiz đã bị xóa bởi người quản lý". Then redirects learner to course page

Relationships:

- Included in: Take quiz

Business Rules:

- User already logged in the system as "Learner" role.
- If the learner clicks on finish, the learning activity of learner for this quiz will be updated and the page will redirect to the next lecture in course. If not, this lecture will be counted as "Not finish learning".
- Learner must participate in plan that courses belong to in order to view quiz result.
- Quiz have to be published and active (not overdue) to be accessed by learner.
- Learner can also see the right answers of quiz.

Table 19: View quiz result specification

2.3.2.13 <Learner> Add discussion

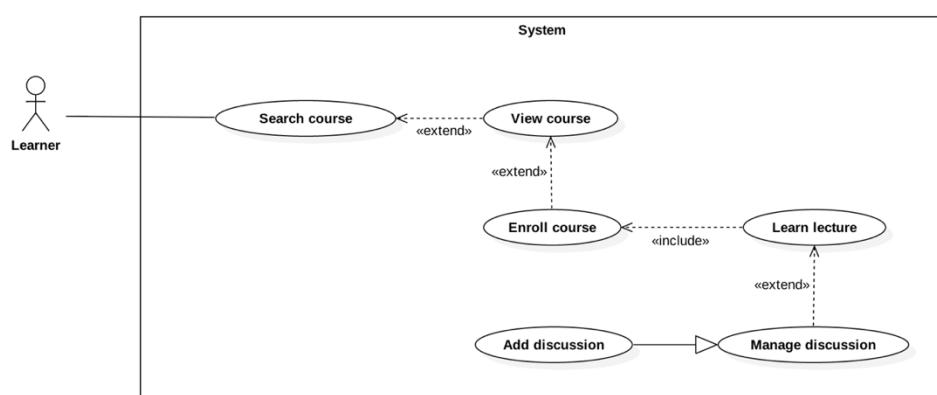


Figure 17: <Learner> Add discussion (UC_TRAISY14)

USE CASE - UC_TRAISY14					
Use Case No.	UC_TRAISY14	Use Case Version	0.2		
Use Case Name	Add discussion				
Author	Phạm Chánh Hưng				
Date	24/3/2018	Priority	High		
Actor:	<ul style="list-style-type: none"> Learner 				
Summary:	<ul style="list-style-type: none"> This use case allows authorized learner to add discussion of a lecture 				
Goal:	<ul style="list-style-type: none"> To add discussion of lecture 				
Triggers:	<ul style="list-style-type: none"> Actor sends add discussion command. 				
Preconditions:	<ul style="list-style-type: none"> Learner logged in the system. 				
Post conditions:	<ul style="list-style-type: none"> Success: Learners are successfully adding discussion to lecture page Fail: System shows error messages. 				
Main Success Scenario:					
Step	Actor Action	System Response			
1	Actor clicks “Thêm thảo luận” button in lecture page	System displays modal with: “Nội dung thảo luận”: text area input			
2	Actor clicks “Thêm” button	[Exception 1] [Exception 2]			
		Discussion is added successfully and system notices the player with message “Thêm thảo luận thành công”.			
Alternative Scenario: N/A					
Exceptions:					
No	Cause	System Response			
1	The lecture that learner add discussion to has been removed or unpublished	System displays notice: “Không thể thêm thảo luận”. Then redirects learner to previous link.			
2	Learner does not input required fields.	System notices that guest need to input all these field: - “Nội dung thảo luận”: System display warning message: “nội dung thảo luận không được trống”.			
Relationships:					

- Abstract use case: Manage discussion

Business Rules:

- User already logged in the system as “Learner” role.
- The content of lecture must be over 5 characters.
- Learner must participate in plan that courses belong to in order to add discussion.
- Plan/course/lecture of the discussion belongs to must be published and active in order to add discussion.
- Learner can not add discussion to a quiz lecture.

Table 20: Add discussion specification

2.3.2.14 <Learner> Update discussion

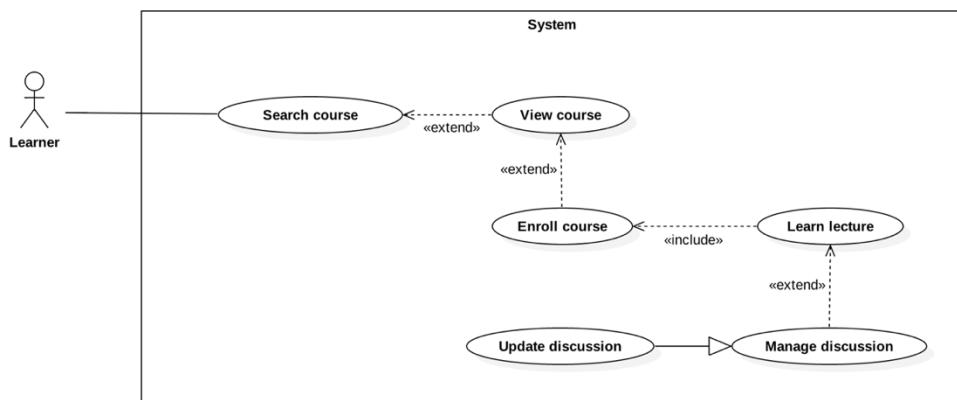


Figure 18: <Learner> Update discussion (UC_TRAISY15)

USE CASE - UC_TRAISY15			
Use Case No.	UC_TRAISY15	Use Case Version	0.2
Use Case Name	Update discussion		
Author	Phạm Chánh Hưng		
Date	24/3/2018	Priority	High
Actor:	<ul style="list-style-type: none"> Learner 		
Summary:	<ul style="list-style-type: none"> This use case allows authorized learner to update his/her discussion of a lecture 		
Goal:	<ul style="list-style-type: none"> To update discussion of lecture 		
Triggers:	<ul style="list-style-type: none"> Actor sends update discussion command. 		
Preconditions:	<ul style="list-style-type: none"> Learner logged in the system. 		
Post conditions:	<ul style="list-style-type: none"> Success: Learners are successfully update discussion in lecture page Fail: System shows error messages. 		

Main Success Scenario:

Step	Actor Action	System Response
1	Actor clicks “Sửa thảo luận” button of a discussion in lecture page	System displays text area with: “Nội dung thảo luận”: text area input
2	Actor clicks “Lưu” button in page	[Exception 1] [Exception 2] Discussion is updated successfully and system notices the player with message “Sửa thảo luận thành công”.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	The lecture that updated discussion belongs to has been removed or unpublished	System displays notice: “Không thể sửa thảo luận”. Then redirects learner to previous link.
2	Learner does not input required fields.	System do nothing

Relationships:

- Abstract use case: Manage discussion

Business Rules:

- User already logged in the system as “Learner” role.
- Learner can only update his own discussion; learner can not update another users’ discussion.
- The content of discussion must be over 5 characters.
- Learner must participate in plan that courses belong to in order to update discussion.
- Plan/course/lecture of the discussion belongs to must be published and active in order to update discussion.

Table 21: Update discussion specification

2.3.2.15 <Learner> Remove discussion

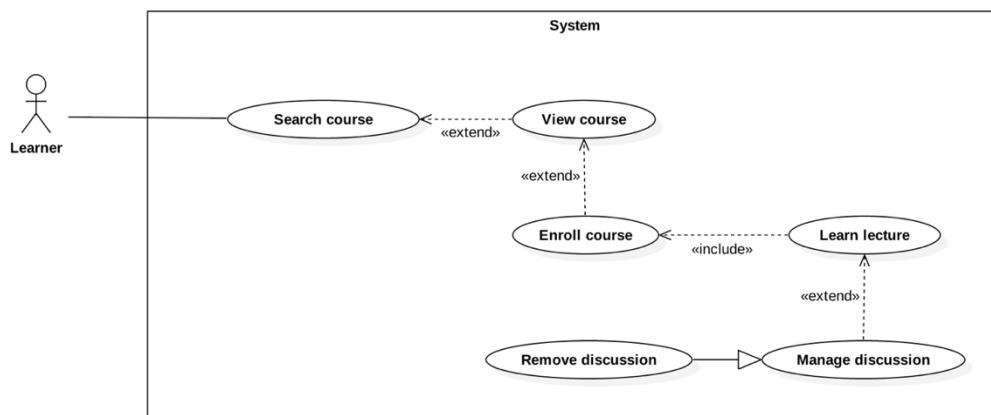


Figure 19: <Learner> Remove discussion (UC_TRAISY16)

USE CASE - UC_TRAISY16									
Use Case No.	UC_TRAISY16	Use Case Version	0.2						
Use Case Name	Remove discussion								
Author	Phạm Chánh Hưng								
Date	24/3/2018	Priority	High						
Actor:	<ul style="list-style-type: none"> Learner 								
Summary:	<ul style="list-style-type: none"> This use case allows authorized learner to remove his/her discussion of a lecture 								
Goal:	<ul style="list-style-type: none"> To remove discussion of lecture 								
Triggers:	<ul style="list-style-type: none"> Actor sends remove discussion command. 								
Preconditions:	<ul style="list-style-type: none"> Learner logged in the system. 								
Post conditions:	<ul style="list-style-type: none"> Success: Learners are successfully remove discussion in lecture page Fail: System shows error messages. 								
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Actor clicks “Xoá thảo luận” button of a discussion in lecture page</td><td> <p>[Exception 1] [Exception 2]</p> <p>Discussion is removed sucessfully and system notices the player with message “Xoá thảo luận thành công”.</p> </td></tr> </tbody> </table>			Step	Actor Action	System Response	1	Actor clicks “Xoá thảo luận” button of a discussion in lecture page	<p>[Exception 1] [Exception 2]</p> <p>Discussion is removed sucessfully and system notices the player with message “Xoá thảo luận thành công”.</p>
Step	Actor Action	System Response							
1	Actor clicks “Xoá thảo luận” button of a discussion in lecture page	<p>[Exception 1] [Exception 2]</p> <p>Discussion is removed sucessfully and system notices the player with message “Xoá thảo luận thành công”.</p>							
Alternative Scenario: N/A									
Exceptions:	<table border="1"> <thead> <tr> <th>No</th><th>Cause</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>The lecture that removed discussion belongs to has been removed or unpublished</td><td>System do nothing</td></tr> </tbody> </table>			No	Cause	System Response	1	The lecture that removed discussion belongs to has been removed or unpublished	System do nothing
No	Cause	System Response							
1	The lecture that removed discussion belongs to has been removed or unpublished	System do nothing							
Relationships:	<ul style="list-style-type: none"> Abstract use case: Manage discussion 								
Business Rules:	<ul style="list-style-type: none"> User already logged in the system as “Learner” role. Learner can only remove his own discussion; learner can not remove another users' discussion. When ever learner removes a discussion, all replies of that discussion will also be removed. 								

- Learner must participate in plan that courses belong to in order to update discussion.
- Plan/course/lecture of the discussion belongs to must be published and active in order to update discussion.

Table 22: Remove discussion specification

2.3.3 <Author> Overview Use Case

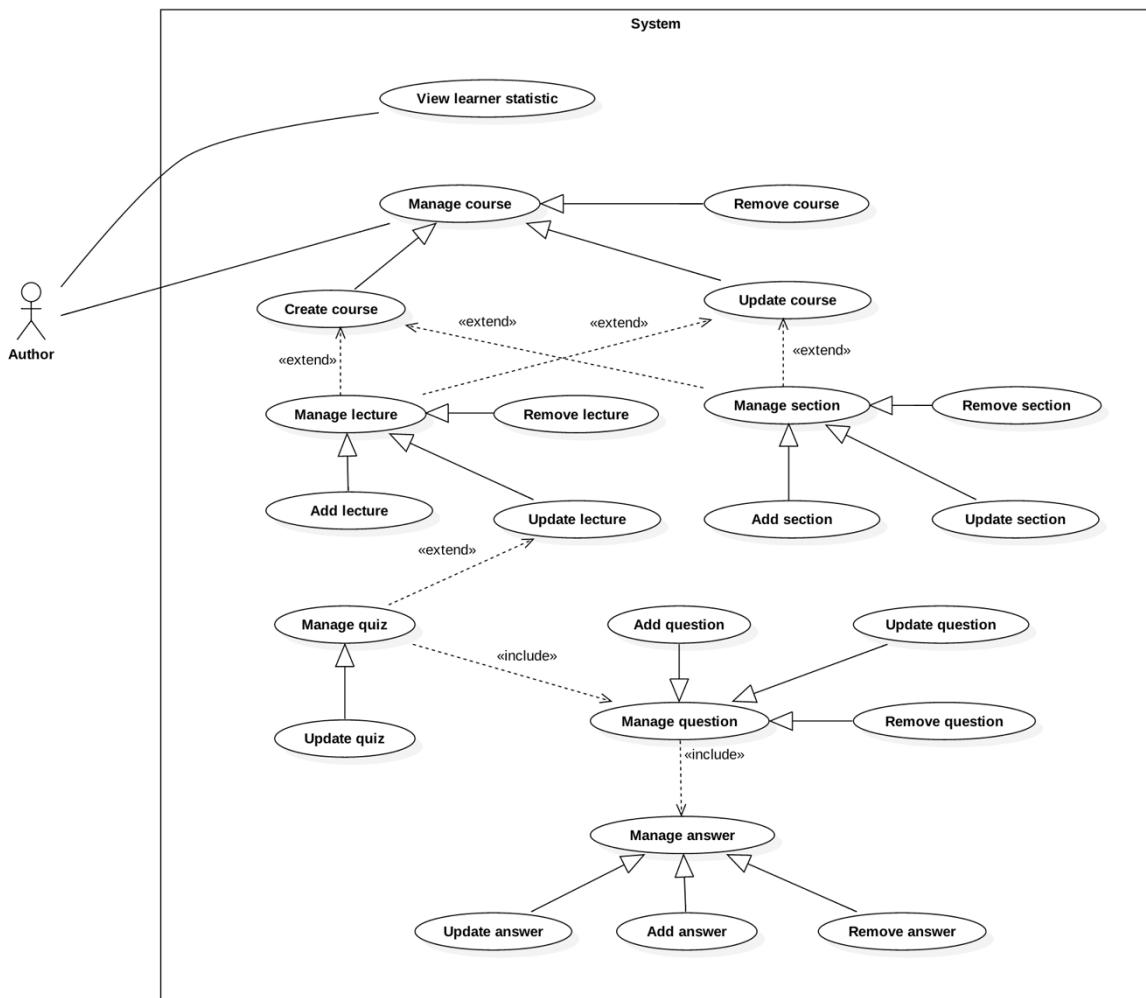


Figure 20: <Author> Overview Use Case

2.3.3.1 <Author> View learner statistic

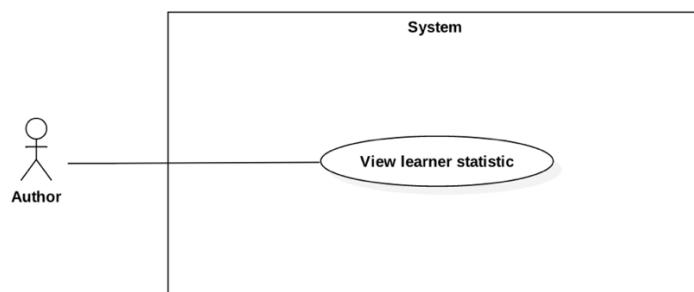


Figure 21: <Author> View learner statistic (UC_TRAISY17)

USE CASE - UC_TRAISY17									
Use Case No.	UC_TRAISY17	Use Case Version	0.2						
Use Case Name	View learner statistic								
Author	Lê Phúc Thịnh								
Date	25/03/2018	Priority	High						
Actor:	<ul style="list-style-type: none"> Author 								
Summary:	<ul style="list-style-type: none"> This use case allows author to view learner statistic in author's course. 								
Goal:	<ul style="list-style-type: none"> Author views learner statistic to get more details about that course. 								
Triggers:	<ul style="list-style-type: none"> Author clicks course's page which include learner statistic. 								
Preconditions:	<ul style="list-style-type: none"> User logged in as author role. 								
Post conditions:	<ul style="list-style-type: none"> Success: System displays learner's process of each course with details of learner. Fail: System displays error messages. 								
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Actor clicks "Học viên" tab from manage course page.</td><td> <p>System displays page which include:</p> <ul style="list-style-type: none"> List of learners Learner's name Enroll date Process of learner on each course </td></tr> </tbody> </table>			Step	Actor Action	System Response	1	Actor clicks "Học viên" tab from manage course page.	<p>System displays page which include:</p> <ul style="list-style-type: none"> List of learners Learner's name Enroll date Process of learner on each course
Step	Actor Action	System Response							
1	Actor clicks "Học viên" tab from manage course page.	<p>System displays page which include:</p> <ul style="list-style-type: none"> List of learners Learner's name Enroll date Process of learner on each course 							
Alternative Scenario:	N/A								
Exceptions:	N/A								
Relationships:	N/A								
Business Rules:	<ul style="list-style-type: none"> User already logged in the system as "Author" role. Learner's statistic only displays learners who already enrolled in course. Author can only view statistic of learners belongs to author's courses. 								

Table 23: View learner statistic specification

2.3.3.2 <Author> Create course

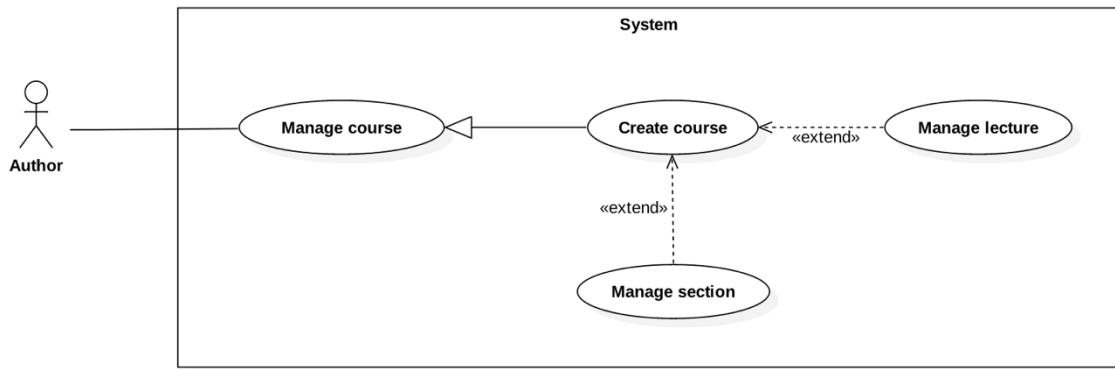


Figure 22: <Author> Create course(UC_TRAISY18)

USE CASE - UC_TRAISY18					
Use Case No.	UC_TRAISY18	Use Case Version	0.2		
Use Case Name	Create course				
Author	Lê Phúc Thịnh				
Date	25/03/2018	Priority	High		
Actor:					
• Author					
Summary:					
• This use case allows author to create course.					
Goal:					
• Author successfully creates course in a specific plan.					
Triggers:					
• Author sends create course command.					
Preconditions:					
• User logged in as author role.					
Post conditions:					
• Success: New course is created into database and redirects user to that course's information page.					
• Fail: System displays error messages.					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Actor clicks "Thêm khóa học" button from manage course page.	System displays modal with field: "Nhập tên khóa học" requires user to input course name.			
2	Actor input course name and clicks "Tạo khóa học"	[Exception no.1] System redirects actor to that course's page including details about that course			
Alternative Scenario: N/A					
Exceptions:					
No	Cause	System Response			

1	Learner does not input required field or input wrong field's requirement.	System notices that “Không thể tạo khóa học. Tên khoá học phải lớn hơn 5 ký tự”
---	---	---

Relationships:

- Abstract use case: Manage course
- Extended by: Manage lecture
- Extended by: Manage section

Business Rules:

- User already logged in the system as “Author” role.
- The name of course has to be over 5 characters.
- When created, course will have default course image. Author can change it in update course.
- The course created will be in a specific program if author select create course in that program. Otherwise, the course created will be in “Another program”.
- List course in “Manage course” page will be updated with new course that just created.
- When created, the course will be “Unpublished” or “Inactive” as default. Author has to send the course for review to be assessable by learners.
- Course title length must be over 5 characters

Table 24: Create course specification

2.3.3.3 <Author> Remove course

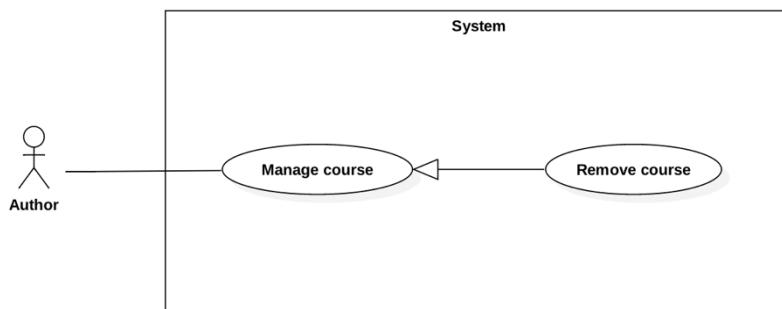


Figure 23: <Author> Remove course(UC_TRAISY19)

USE CASE - UC_TRAISY19			
Use Case No.	UC_TRAISY19	Use Case Version	0.2
Use Case Name	Remove course		
Author	Lê Phúc Thịnh		
Date	25/03/2018	Priority	High
Actor:			
• Author			
Summary:			
• This use case allows author to remove a specific course.			

Goal:						
<ul style="list-style-type: none"> Author successfully remove course in a specific plan. 						
Triggers:						
<ul style="list-style-type: none"> Author sends remove course command. 						
Preconditions:						
<ul style="list-style-type: none"> User logged in as author role. 						
Post conditions:						
<ul style="list-style-type: none"> Success: The course's status in database will be changed to “-1” and display success message. Fail: System displays error messages. 						
Main Success Scenario:						
<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Actor clicks "Xoá vĩnh viễn khóa học" button from manage course page.</td><td>System displays popup “Xoá khóa học thành công”.</td></tr> </tbody> </table>	Step	Actor Action	System Response	1	Actor clicks "Xoá vĩnh viễn khóa học" button from manage course page.	System displays popup “Xoá khóa học thành công”.
Step	Actor Action	System Response				
1	Actor clicks "Xoá vĩnh viễn khóa học" button from manage course page.	System displays popup “Xoá khóa học thành công”.				
Alternative Scenario: N/A						
Exceptions: N/A						
Relationships:						
<ul style="list-style-type: none"> Abstract use case: Manage course 						
Business Rules:						
<ul style="list-style-type: none"> User already logged in the system as “Author” role. The removed course's status will be change to “-1” in database. Therefore, any data such as lecture, section relating to course is not completely deleted from database. The removed course can be changed to active/publish. Learner cannot see or access to course that has been removed. Only author who owns the course can modify. Otherwise, other authors cannot modify. 						

Table 25: Remove course specification

2.3.3.4 <Author> Update course

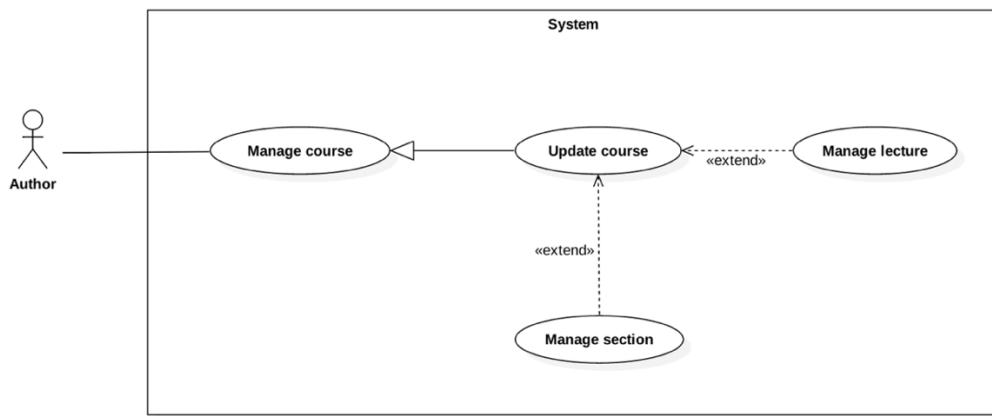


Figure 24: <Author> Update course (UC_TRAISY20)

USE CASE - UC_TRAISY20					
Use Case No.	UC_TRAISY20	Use Case Version	0.2		
Use Case Name	Update course				
Author	Lê Phúc Thịnh				
Date	25/03/2018	Priority	High		
Actor:					
<ul style="list-style-type: none"> • Author 					
Summary:					
<ul style="list-style-type: none"> • This use case allows author to update a specific course. 					
Goal:					
<ul style="list-style-type: none"> • Author successfully update course in a specific plan. 					
Triggers:					
<ul style="list-style-type: none"> • Author sends update course command. 					
Preconditions:					
<ul style="list-style-type: none"> • User logged in as author role. 					
Post conditions:					
<ul style="list-style-type: none"> • Success: The course with new information will be updated in database. • Fail: System displays error messages. 					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Actor go to “Quản lý khóa học” page	System displays page with following information: <ul style="list-style-type: none"> • “Tên khóa học”: text box • “Mô tả”: text area • “Kỹ năng khóa học”: drop down list which list all skills available • Button “Lưu chỉnh sửa” 			
2	Actor clicks “Lưu chỉnh sửa” button from manage course page.	[Exception no.1] System displays popup “Cập nhật thành công”.			
Alternative Scenario: N/A					
Exceptions:					
No	Cause	System Response			
1	Learner does not input required field or input wrong field's requirement.	System notices that “Vui lòng nhập tên khóa học lớn hơn 5 ký tự”			
Relationships:					
<ul style="list-style-type: none"> • Abstract use case: Manage course • Extended by: Manage lecture • Extended by: Manage section 					
Business Rules:					
<ul style="list-style-type: none"> • User already logged in the system as “Author” role. • If update successfully, course will be updated in database immediately. 					

- New information will be updated in manage course page.
- Course title length must be over 5 characters.
- Skills of course cannot exceed 5.
- Only author who owns the course can modify. Otherwise, other authors cannot modify.

Table 26: Update course specification

2.3.3.5 <Author> Add lecture

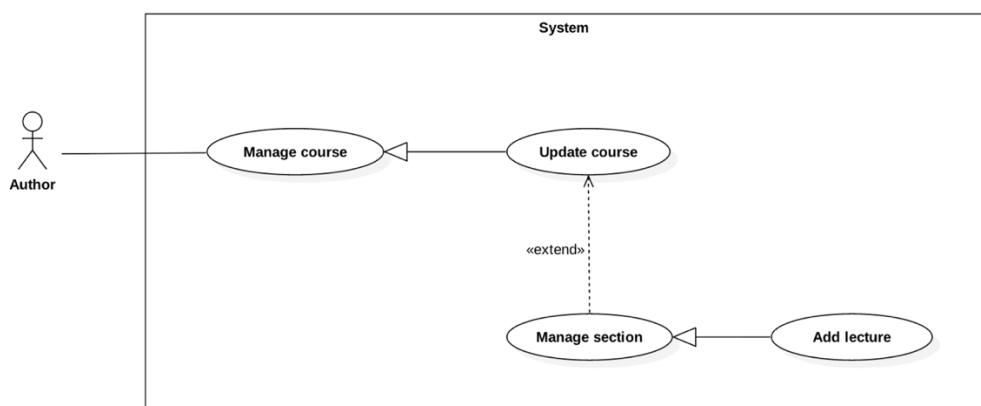


Figure 25: <Author> Add lecture (UC_TRAISY21)

USE CASE – UC_TRAISY21			
Use Case No.	UC_TRAISY21	Use Case Version	0.2
Use Case Name	Add lecture		
Author	Lê Phúc Thịnh		
Date	25/03/2018	Priority	High
Actor:			
• Author			
Summary:			
• This use case allows author to add a lecture to course.			
Goal:			
• Author successfully add lecture in a specific course.			
Triggers:			
• Author sends add lecture command.			
Preconditions:			
• User logged in as author role.			

Post conditions:

- Success: The new lecture is created.
- Fail: System displays error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor clicks "Thêm bài học" button from manage course page.	System displays modal with field: "Nhập tên bài học" requires user to input lecture name.
2	Actor clicks "Thêm bài mới" button.	[Exception no.1] System displays popup "Thêm bài mới thành công".

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner does not input required field or input wrong field's requirement.	System notices that "Vui lòng nhập tên bài học lớn hơn 5 ký tự"

Relationships:

- Abstract use case: Manage lecture

Business Rules:

- User already logged in the system as "Author" role.
- The lecture created must be in a specific course.
- Author can only add lecture in their own's course and can not add lecture in another authors' course.
- List lecture in "Manage course" page will be updated with new lecture that just created.
- When created, the lecture will be "Unpublished" or "Inactive" as default. Author can publish the lecture through Update Lecture use case
- Lecture title length must be over 5 characters and below 80.

Table 27: Add lecture specification

2.3.3.6 <Author> Update lecture

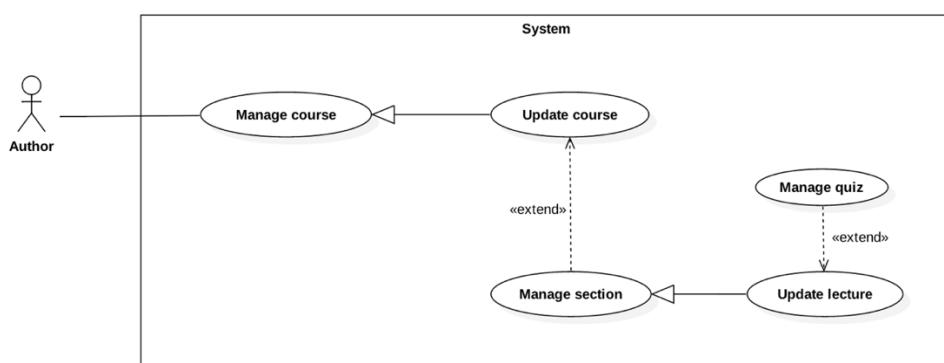


Figure 26: <Author> Update lecture (UC_TRAISY22)

USE CASE - UC_TRAISY22

Use Case No.	UC_TRAISY22	Use Case Version	0.2
Use Case Name	Update lecture		
Author	Lê Phúc Thịnh		
Date	25/03/2018	Priority	High

Actor:

- Author

Summary:

- This use case allows author to update a specific lecture.

Goal:

- Author successfully update lecture.

Triggers:

- Author sends update lecture command.

Preconditions:

- User logged in as author role.

Post conditions:

- Success: The lecture with new information will be updated in database.
- Fail: System displays error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor goes to “Quản lý khóa học” page and clicks “Nội dung bài học” tab	System displays page with: <ul style="list-style-type: none"> • List of lectures • Publish/unpublish button • Edit button • Delete button
2	Actor clicks Edit button from manage course page.	System displays page with: <ul style="list-style-type: none"> • Name of lecture • Publish/unpublish lecture • Contains of lecture
3	Actor clicks “Lưu bài học” button from manage course page.	[Exception no.1] System displays popup “Cập nhật thành công”.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner does not input required field or input wrong field's requirement.	System notices that “Vui lòng nhập tên bài học lớn hơn 5 ký tự”

Relationships:

- Abstract use case: Manage lecture
- Extend by: Manage quiz

Business Rules:

- User already logged in the system as “Author” role.
- If author changes lecture type (e.g.: text, video, PowerPoint), the old content of this lecture will be removed.

- If update successfully, lecture will be updated in database immediately.
- Lecture title length must be over 5 characters
- Only author who owns the lecture can modify. Otherwise, other authors cannot modify

Table 28: Update lecture specification

2.3.3.7 <Author> Remove lecture

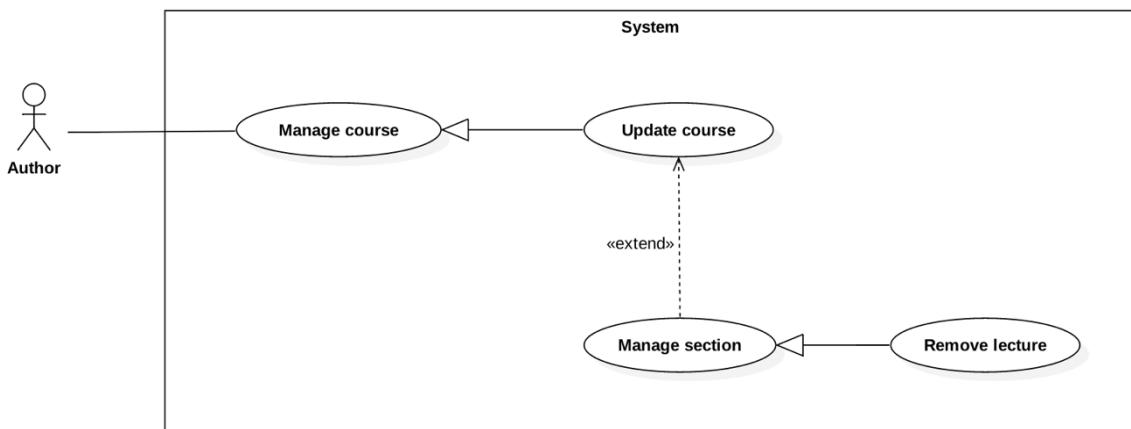


Figure 27: <Author> Remove lecture (UC_TRAISY23)

USE CASE - UC_TRAISY23			
Use Case No.	UC_TRAISY23	Use Case Version	0.2
Use Case Name	Remove lecture		
Author	Lê Phúc Thịnh		
Date	25/03/2018	Priority	High
Actor:			
• Author			
Summary:			
• This use case allows author to remove a specific lecture.			
Goal:			
• Author successfully remove lecture.			
Triggers:			
• Author sends remove lecture command.			
Preconditions:			
• User logged in as author role.			

Post conditions:

- Success: The lecture will be removed from database.
- Fail: System displays error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor goes to “Quản lý khóa học” page and clicks “Nội dung bài học” tab	System displays page with: <ul style="list-style-type: none"> • List of lectures • Publish/unpublish button • Edit button • Delete button
2	Actor clicks delete lecture button from manage course page.	[Exception no.1] System displays popup “Xoá bài học thành công”.

Alternative Scenario: N/A

Exceptions: N/A

Relationships:

- Abstract use case: Manage lecture

Business Rules:

- User already logged in the system as “Author” role.
- The lecture will be completely removed from database including related information of lecture.
- Learner cannot see or access to lecture that has been removed.
- Only author who owns the lecture can modify. Otherwise, other authors cannot modify it.

Table 29: Remove lecture specification

2.3.3.8 <Author> Update quiz

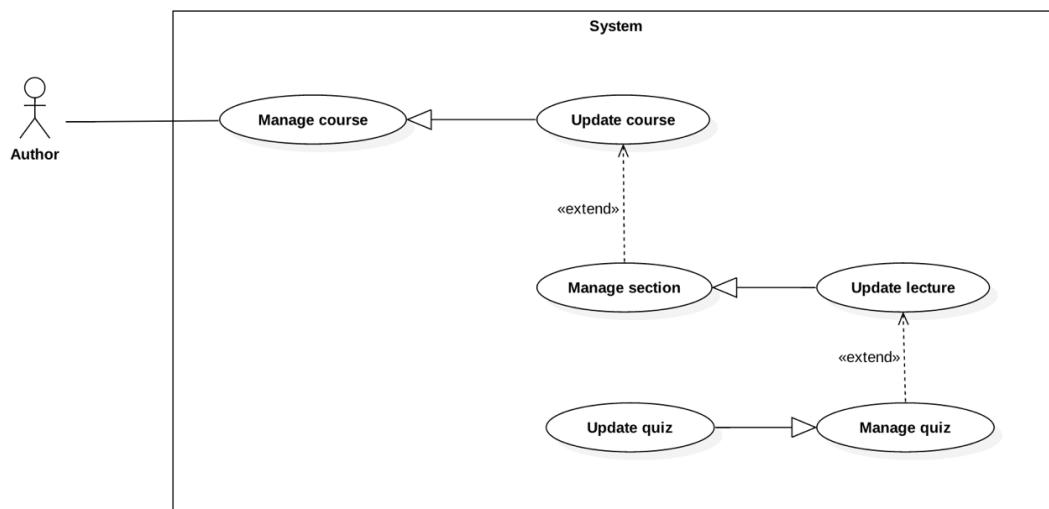


Figure 28: <Author> Update quiz (UC_TRAISY24)

USE CASE - UC_TRAISY24			
Use Case No.	UC_TRAISY24	Use Case Version	0.2

Use Case Name	Update quiz				
Author	Lê Phúc Thịnh				
Date	25/03/2018	Priority	High		
Actor:					
<ul style="list-style-type: none"> • Author 					
Summary:					
<ul style="list-style-type: none"> • This use case allows author to update a specific quiz. 					
Goal:					
<ul style="list-style-type: none"> • Author successfully update quiz. 					
Triggers:					
<ul style="list-style-type: none"> • Author sends update quiz command. 					
Preconditions:					
<ul style="list-style-type: none"> • User logged in as author role. 					
Post conditions:					
<ul style="list-style-type: none"> • Success: The quiz with new information will be update to database. • Fail: System displays error messages. 					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Actor goes to “Quản lý bài học”	System displays page with: <ul style="list-style-type: none"> • “Thời gian làm bài” • “Điểm thấp nhất” • List of questions • “Lưu” button 			
2	Actor clicks “Lưu” button	[Exception no.1] System displays popup “Cập nhật thành công”			
Alternative Scenario: N/A					
Exceptions:					
No	Cause	System Response			
1	Learner does not input required field or input wrong field's requirement.	System notices that “Không thể chỉnh sửa bài học”			
Relationships:					
<ul style="list-style-type: none"> • Abstract use case: Manage quiz 					
Business Rules:					
<ul style="list-style-type: none"> • User already logged in the system as “Author” role. • If author changes lecture type (e.g.: text, video, PowerPoint, Quiz), the old content of this lecture will be removed. • Minimum score for quiz has to be larger than 0. • Duration of quiz can not be negative; the default of duration time is 0. • If update successfully, quiz will be updated in database immediately. • Only author who owns the quiz can update, other authors cannot modify. 					

Table 30: Update quiz specification

2.3.3.9 <Author> Add question

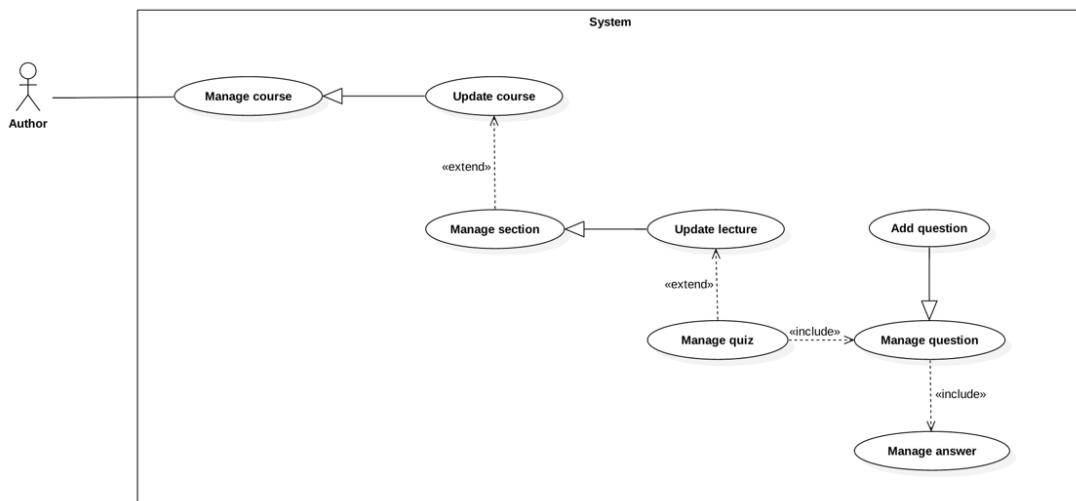


Figure 29: <Author> Add question (UC_TRAISY25)

USE CASE – UC_TRAISY25			
Use Case No.	UC_TRAISY25	Use Case Version	0.2
Use Case Name	Add question		
Author	Phạm Chánh Hưng		
Date	25/03/2018	Priority	High
Actor:			
• Author			
Summary:			
• This use case allows author to add a question.			
Goal:			
• Author successfully add question.			
Triggers:			
• Author sends add question command.			
Preconditions:			
• User logged in as author role.			
Post conditions:			
• Success: The new question is added to database.			
• Fail: System displays error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Actor goes to “Quản lý bài học”	System displays page with: • “Thời gian làm bài” • “Điểm thấp nhất” • List of questions • “Thêm câu hỏi” button	
2	Actor clicks “Thêm câu hỏi” button	System displays modal with fields: • “Nội dung câu hỏi”: text input	

		<ul style="list-style-type: none"> “Loại câu hỏi”: dropdown list with types of question “Câu trả lời”: text input
3	Actor clicks “Thêm” button	<p>[Exception no.1] System displays popup “Thêm câu hỏi thành công”</p>

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner does not input required field or input wrong field's requirement.	<p>System displays notices:</p> <ul style="list-style-type: none"> “Nội dung câu hỏi” field: “Nội dung câu hỏi từ 5 tới 500 ký tự” “Câu trả lời” field: “Nội dung câu trả lời từ 5-500 ký tự”

Relationships:

- Abstract use case: Manage question

Business Rules:

- User already logged in the system as “Author” role.
- If add successfully, question will be added in database immediately.
- Content of question must be over 5 and below 500 characters.
- There are 2 types of question that author can create: multiple-choice, text type question.
- Only author who owns the quiz can add question, other authors cannot modify.

Table 31: Add question specification

2.3.3.10 <Author> Update question

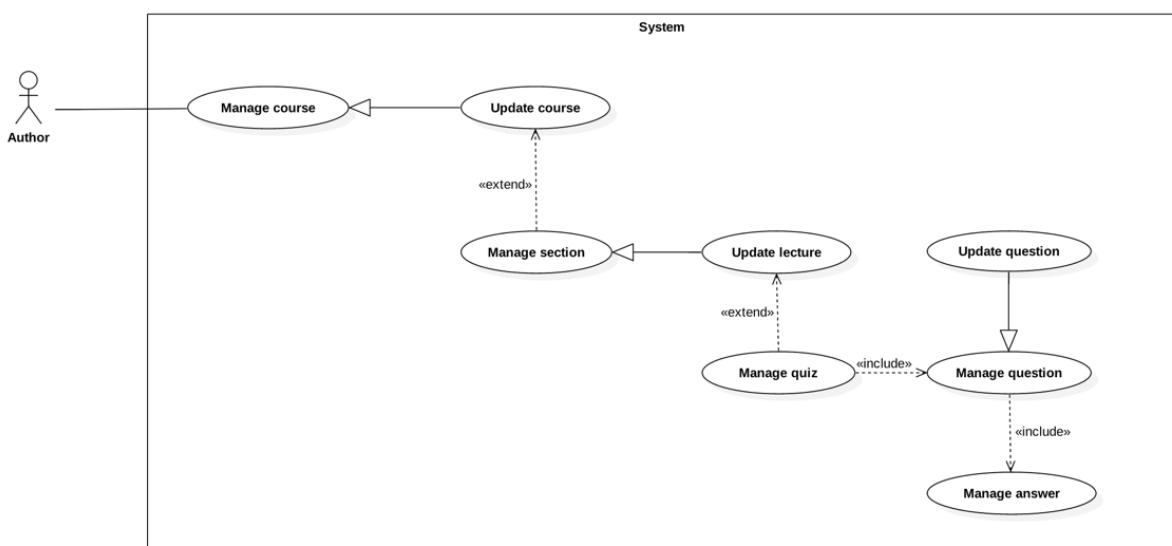


Figure 30: <Author> Update question(UC_TRAISY26)

USE CASE - UC_TRAISY26			
Use Case No.	UC_TRAISY26	Use Case Version	0.2

Use Case Name	Update question		
Author	Phạm Chánh Hưng		
Date	25/03/2018	Priority	High

Actor:

- Author

Summary:

- This use case allows author to update a question.

Goal:

- Author successfully update question.

Triggers:

- Author sends update question command.

Preconditions:

- User logged in as author role.

Post conditions:

- Success: The question with new details is updated to database.
- Fail: System displays error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor goes to “Quản lý bài học”	System displays page with: <ul style="list-style-type: none"> • “Thời gian làm bài” • “Điểm thấp nhất” • List of questions: with dropdown list for each question when clicked • “Thêm câu hỏi” button
2	Actor clicks on a specific question	System will display information about that question which contains: <ul style="list-style-type: none"> • Question name: text box • Delete button • List of answer
3	Actor remove focus on question	[Exception no.1] System displays popup “Chỉnh sửa thành công”

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner does not input required field or input wrong field's requirement.	System displays notices: <ul style="list-style-type: none"> • “Nội dung câu hỏi” field: “Nội dung câu hỏi từ 5 tới 500 ký tự” • “Câu trả lời” field: “Nội dung câu trả lời từ 5-500 ký tự”

Relationships:

- Abstract use case: Manage question

Business Rules:

- User already logged in the system as “Author” role.
- Content of question must be over 5 and below 500 characters.
- There are 2 types of question that author can update: multiple-choice, text type question.
- If update successfully, question will be updated in database immediately.
- Only author who owns the quiz can update question, other authors cannot modify.

Table 32: Update question specification

2.3.3.11 <Author> Remove question

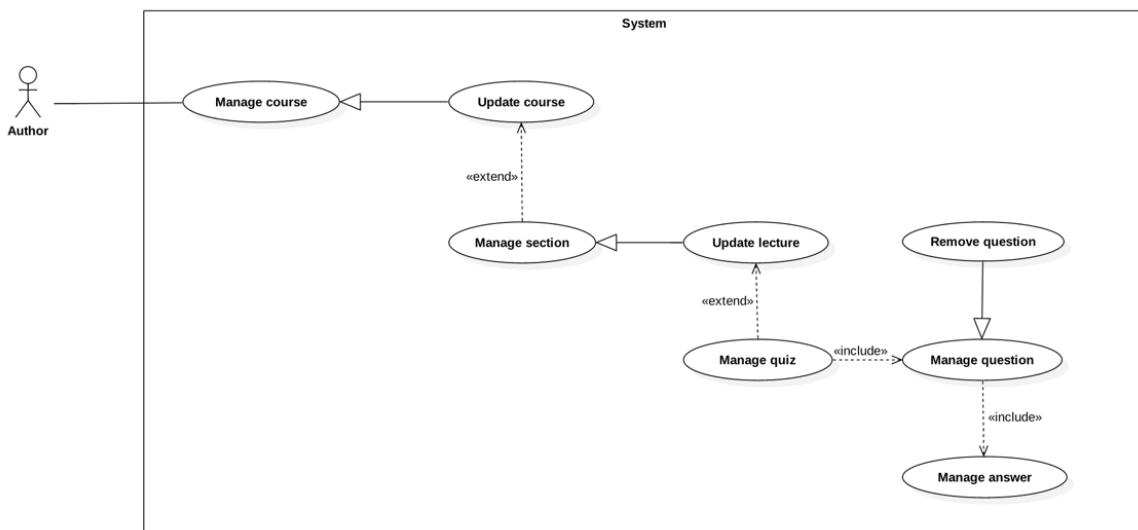


Figure 31: <Author> Remove question(UC_TRAISY27)

USE CASE - UC_TRAISY27			
Use Case No.	UC_TRAISY27	Use Case Version	0.2
Use Case Name	Remove question		
Author	Phạm Chánh Hưng		
Date	25/03/2018	Priority	High
Actor:	<ul style="list-style-type: none"> • Author 		
Summary:	<ul style="list-style-type: none"> • This use case allows author to remove a question. 		

Goal:		
<ul style="list-style-type: none"> • Author successfully remove question from a quiz. 		
Triggers:		
<ul style="list-style-type: none"> • Author sends remove question command. 		
Preconditions:		
<ul style="list-style-type: none"> • User logged in as author role. 		
Post conditions:		
<ul style="list-style-type: none"> • Success: The question will be deleted from database. • Fail: System displays error messages. 		
Main Success Scenario:		
Step	Actor Action	System Response
1	Actor goes to “Quản lý bài học”	<p>System displays page with:</p> <ul style="list-style-type: none"> • “Thời gian làm bài” • “Điểm thấp nhất” • List of questions: with delete button for each question • “Thêm câu hỏi” button
2	Actor clicks delete a specific question	System displays modal to confirm delete question
3	Actor clicks confirm	<p>[Exception no.1]</p> <p>System displays popup “Xóa câu hỏi thành công”</p>
Alternative Scenario: N/A		
Exceptions: N/A		
Relationships:		
<ul style="list-style-type: none"> • Abstract use case: Manage question 		
Business Rules:		
<ul style="list-style-type: none"> • User already logged in the system as “Author” role. • If delete successfully, question will be delete in database immediately. • Only author who owns the quiz can delete question, other authors cannot modify. • When delete, all answers or learner process following the question will also be deleted. 		

Table 33: Remove question specification

2.3.3.12 <Author> Add answer

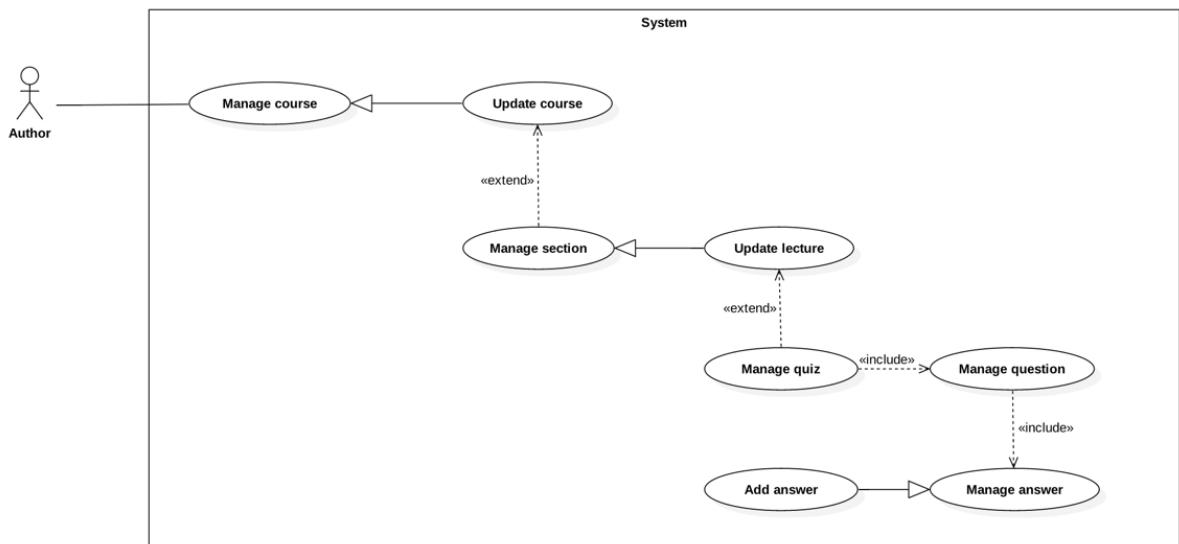


Figure 32: <Author> Add answer(UC_TRAISY28)

USE CASE - UC_TRAISY28			
Use Case No.	UC_TRAISY28	Use Case Version	0.2
Use Case Name	Add answer		
Author	Phạm Chánh Hưng		
Date	25/03/2018	Priority	High

Actor:

- Author

Summary:

- This use case allows author to add an answer to a specific question.

Goal:

- Author successfully adds answer to a question.

Triggers:

- Author sends add answer command.

Preconditions:

- User logged in as author role.

Post conditions:

- Success: The answer will be added to database.
- Fail: System displays error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor goes to “Quản lý bài học”	System displays page with: <ul style="list-style-type: none"> • “Thời gian làm bài” • “Điểm thấp nhất” • List of questions: with delete button for each question • “Thêm câu hỏi” button
2	Actor clicks “Thêm câu hỏi” button	System displays modal with fields:

		<ul style="list-style-type: none"> “Nội dung câu hỏi”: text input “Loại câu hỏi”: dropdown list with types of question “Câu trả lời”: text input
3	Actor input question and answers then click “Thêm” button	[Exception no.1] System displays popup “Thêm câu hỏi thành công”

Alternative Scenario:

Step	Actor Action	System Response
1	Actor goes to “Quản lý bài học”	System displays page with: <ul style="list-style-type: none"> “Thời gian làm bài” “Điểm thấp nhất” List of questions: with dropdown list for each question when clicked “Thêm câu hỏi” button
2	Actor clicks on a specific question	System will display information about that question which contains: <ul style="list-style-type: none"> Question name: text box Delete button List of answer
3	Actor add answer through update the question (UC_TRAISY26)	System displays popup “Chỉnh sửa thành công”

Exceptions:

No	Cause	System Response
1	Learner does not input required field or input wrong field's requirement.	System displays notices: <ul style="list-style-type: none"> “Nội dung câu hỏi” field: “Nội dung câu hỏi từ 5 tới 500 ký tự” “Câu trả lời” field: “Nội dung câu trả lời từ 5-500 ký tự”

Relationships:

- Abstract use case: Manage answer

Business Rules:

- User already logged in the system as “Author” role.
- Answer content has to be over 5 and below 500 characters.
- Author can decide multiple choices or single choice base on number of correct answers of that question.
- If add successfully, answer will be added in database immediately.
- There are 2 types of answer that author can create: choice answer, text answer.
- Only author who owns the quiz can add answer, other authors cannot modify.

Table 34: Add answer specification

2.3.3.13 <Author> Update answer

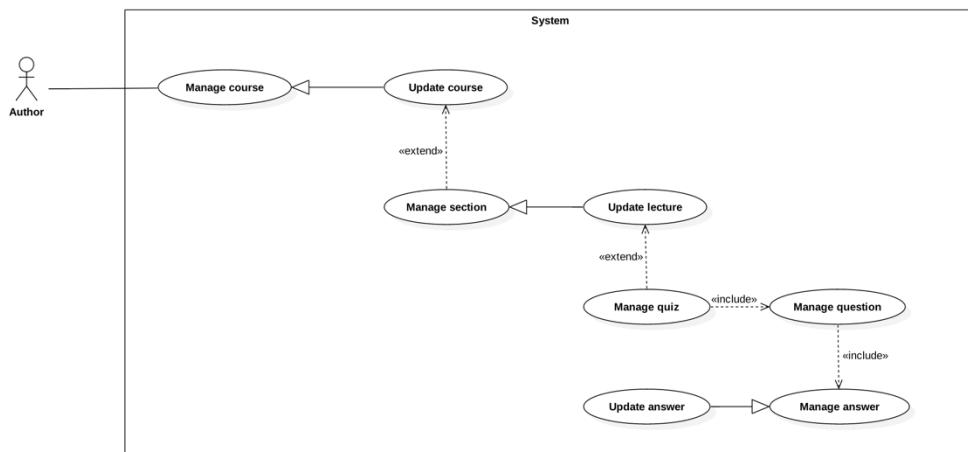


Figure 33: <Author> Update answer (UC_TRAISY28)

USE CASE - UC_TRAISY28			
Use Case No.	UC_TRAISY28	Use Case Version	0.2
Use Case Name	Update answer		
Author	Phạm Chánh Hưng		
Date	25/03/2018	Priority	High
Actor:			
• Author			
Summary:			
• This use case allows author to update an answer.			
Goal:			
• Author successfully updates a specific answer.			
Triggers:			
• Author sends update answer command.			
Preconditions:			
• User logged in as author role.			
Post conditions:			
• Success: The answer will be updated to database.			
• Fail: System displays error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Actor goes to “Quản lý bài học”	System displays page with: • “Thời gian làm bài” • “Điểm thấp nhất” • List of questions: with dropdown list for each question when clicked • “Thêm câu hỏi” button	
2	Actor clicks on a specific question	System will display information about that question which contains:	

		<ul style="list-style-type: none"> • Question name: text box • Delete button • List of answer
3	Actor remove focus on a specific answer	[Exception no.1] System displays popup “Chỉnh sửa thành công”

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner does not input required field or input wrong field's requirement.	System displays notices: <ul style="list-style-type: none"> • “Câu trả lời” field: “Nội dung câu trả lời từ 5-500 ký tự”

Relationships:

- Abstract use case: Manage answer

Business Rules:

- User already logged in the system as “Author” role.
- Answer content has to be over 5 and below 500 characters.
- If update successfully, answer will be updated in database immediately.
- Only author who owns the quiz can update answer, other authors cannot modify.
- Type of answer cannot be changed (from choice type to text type or otherwise).
- Author can update the correct answers for the question.

Table 35: Update answer specification

2.3.3.14 <Author> Remove answer

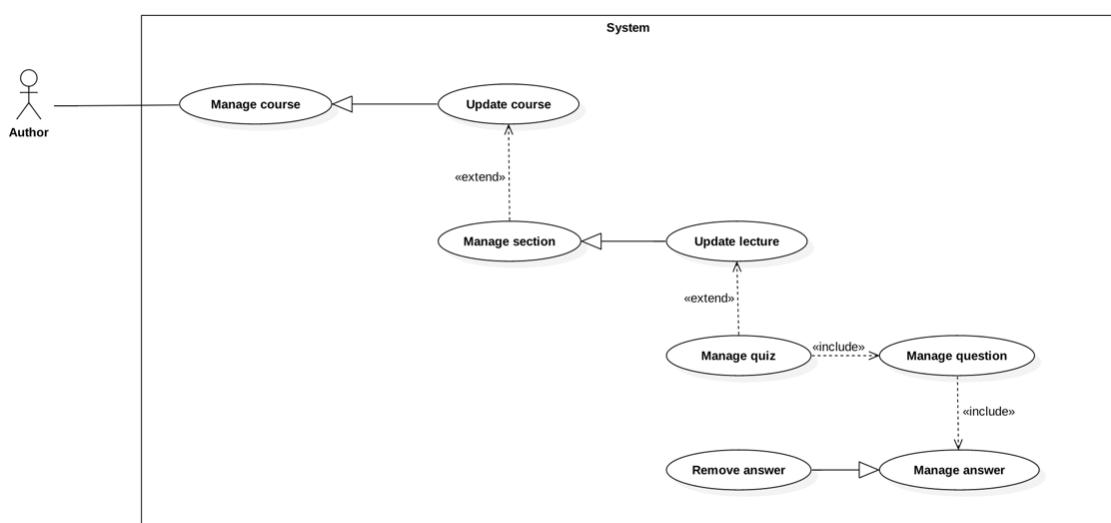


Figure 34: <Author> Remove answer (UC_TRAISY29)

USE CASE – UC_TRAISY29

Use Case No.	UC_TRAISY29	Use Case Version	0.2
Use Case Name	Remove answer		
Author	Phạm Chánh Hưng		
Date	25/03/2018	Priority	High

Actor:

- Author

Summary:

- This use case allows author to remove an answer from a question.

Goal:

- Author successfully removes a specific answer of a question.

Triggers:

- Author sends remove answer command.

Preconditions:

- User logged in as author role.

Post conditions:

- Success: The answer of question will be removed from database.
- Fail: System displays error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor goes to “Quản lý bài học”	System displays page with: <ul style="list-style-type: none"> • “Thời gian làm bài” • “Điểm thấp nhất” • List of questions: with dropdown list for each question when clicked • “Thêm câu hỏi” button
2	Actor clicks on a specific question	System will display information about that question which contains: <ul style="list-style-type: none"> • Question name: text box • Delete button • List of answer with delete button for each answer
3	Actor clicks on delete button on a specific answer	System displays popup “Chỉnh sửa thành công”

Alternative Scenario: N/A

Exceptions: N/A

Relationships:

- Abstract use case: Manage answer

Business Rules:

- User already logged in the system as “Author” role.
- If delete successfully, answer will be deleted in database immediately.

- Only author who owns the quiz can delete answer, other authors cannot modify.
- When delete, all learner process or related information following the answer will also be deleted.

Table 36: Remove answer specification

2.3.3.15 <Author> Add section

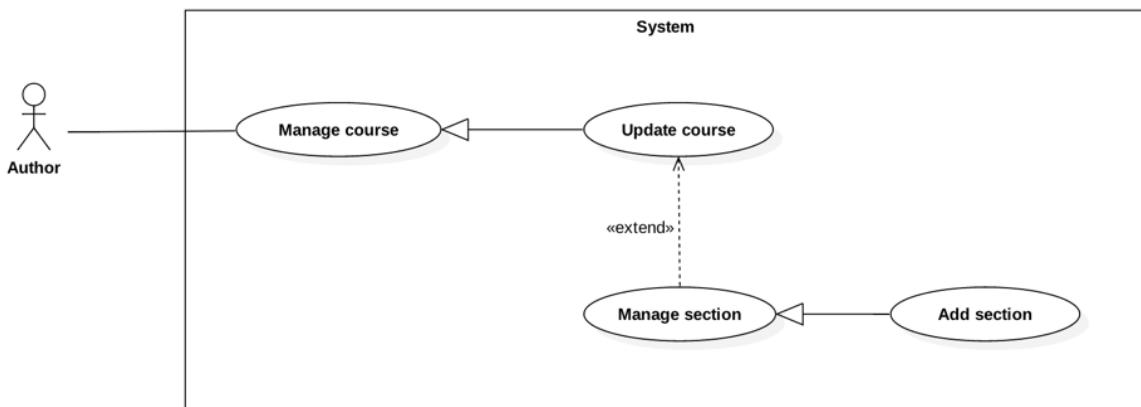


Figure 35: <Author> Add section(UC_TRAISY30)

USE CASE - UC_TRAISY30												
Use Case No.	UC_TRAISY30	Use Case Version	0.2									
Use Case Name	Add section											
Author	Võ Thạch Nguyên											
Date	25/03/2018	Priority	High									
Actor:	<ul style="list-style-type: none"> Author 											
Summary:	<ul style="list-style-type: none"> This use case allows author to add a section to a specific course. 											
Goal:	<ul style="list-style-type: none"> Author successfully adds a section to database. 											
Triggers:	<ul style="list-style-type: none"> Author sends add section command. 											
Preconditions:	<ul style="list-style-type: none"> User logged in as author role. 											
Post conditions:	<ul style="list-style-type: none"> Success: The section of course will be successfully added to database. Fail: System displays error messages. 											
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Actor clicks "Thêm chương mới" button from manage course page.</td><td>System displays modal with field: "Nhập tên chương" requires user to input section name.</td></tr> <tr> <td>2</td><td>Actor clicks "Thêm chương mới" button.</td><td>[Exception no.1] System displays popup "Thêm chương mới thành công".</td></tr> </tbody> </table>			Step	Actor Action	System Response	1	Actor clicks "Thêm chương mới" button from manage course page.	System displays modal with field: "Nhập tên chương" requires user to input section name.	2	Actor clicks "Thêm chương mới" button.	[Exception no.1] System displays popup "Thêm chương mới thành công".
Step	Actor Action	System Response										
1	Actor clicks "Thêm chương mới" button from manage course page.	System displays modal with field: "Nhập tên chương" requires user to input section name.										
2	Actor clicks "Thêm chương mới" button.	[Exception no.1] System displays popup "Thêm chương mới thành công".										

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner does not input required field or input wrong field's requirement.	System notices that "Vui lòng nhập tên chương học lớn hơn 5 ký tự"

Relationships:

- Abstract use case: Manage section

Business Rules:

- User already logged in the system as "Author" role.
- Author can only add section in their own's course and can not add section in another authors' course.
- The section created must be in a specific course.
- List section in "Manage course" page will be updated with new section that just created.
- Section title length must be over 5 characters and below 80.

Table 37: Add section specification

2.3.3.16 <Author> Update section

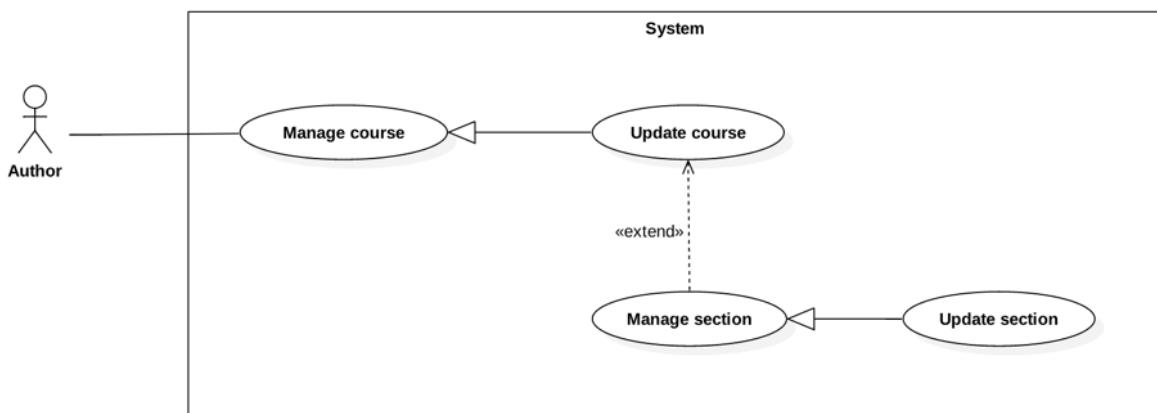


Figure 36: <Author> Update section(UC_TRAISY31)

USE CASE - UC_TRAISY31			
Use Case No.	UC_TRAISY31	Use Case Version	0.2
Use Case Name	Update section		
Author	Võ Thạch Nguyên		
Date	25/03/2018	Priority	High
Actor:	<ul style="list-style-type: none"> Author 		
Summary:	<ul style="list-style-type: none"> This use case allows author to update a specific section. 		
Goal:	<ul style="list-style-type: none"> Author successfully updates a section to database. 		
Triggers:	<ul style="list-style-type: none"> Author sends update section command. 		
Preconditions:			

- User logged in as author role.

Post conditions:

- Success: The section with new information will be updated to database.
- Fail: System displays error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	Actor goes to “Quản lý khóa học” page and clicks “Nội dung bài học” tab	System displays page with: • List of section • Edit button next to each section
2	Actor clicks Edit button from a specific section.	System displays edit text that can edit section's title and confirm button
3	Actor clicks confirm button.	[Exception no.1] System displays popup “Cập nhật thành công”.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Learner does not input required field or input wrong field's requirement.	System notices that “Vui lòng nhập tên chương học lớn hơn 5 ký tự”

Relationships:

- Abstract use case: Manage section

Business Rules:

- User already logged in the system as “Author” role.
- If update successfully, section will be updated in database immediately.
- List section in “Manage course” page will be updated with new details of section that just updated.
- Section title length must be over 5 characters and below 80.
- Only author who owns the course can modify. Otherwise, other authors cannot modify.

Table 38: Update section specification

2.3.3.17 <Author> Remove section

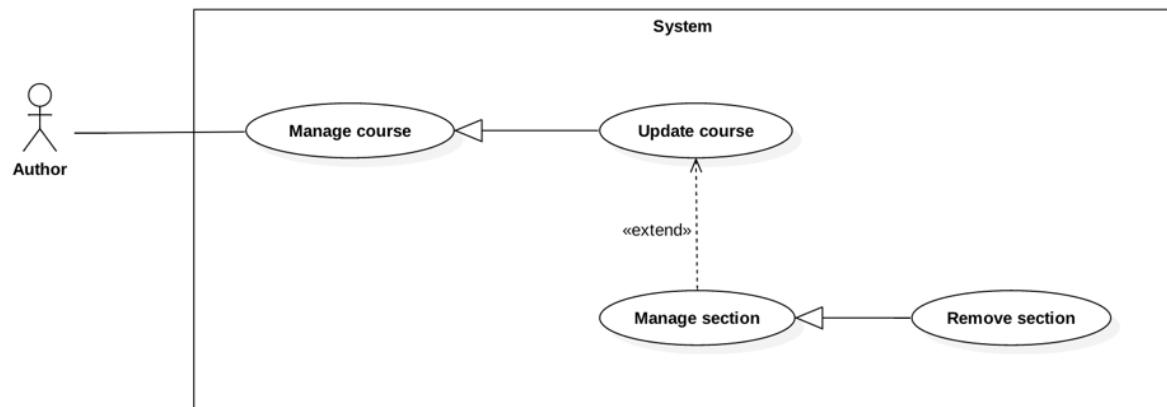


Figure 37: <Author> Remove section (UC_TRAISY32)

USE CASE - UC_TRAISY32					
Use Case No.	UC_TRAISY32	Use Case Version	0.2		
Use Case Name	Remove section				
Author	Võ Thạch Nguyên				
Date	25/03/2018	Priority	High		
Actor:					
<ul style="list-style-type: none"> • Author 					
Summary:					
<ul style="list-style-type: none"> • This use case allows author to remove a specific section. 					
Goal:					
<ul style="list-style-type: none"> • Author successfully removes a section from database. 					
Triggers:					
<ul style="list-style-type: none"> • Author sends remove section command. 					
Preconditions:					
<ul style="list-style-type: none"> • User logged in as author role. 					
Post conditions:					
<ul style="list-style-type: none"> • Success: The section will be removed from database. • Fail: System displays error messages. 					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Actor goes to “Quản lý khóa học” page and clicks “Nội dung bài học” tab	System displays page with: <ul style="list-style-type: none"> • List of section • Delete button for each section 			
2	Actor clicks delete button from a specific section.	System displays modal to confirm author's action			
3	Actor clicks confirm button.	System displays popup “Xoá thành công”			
Alternative Scenario: N/A					
Exceptions: N/A					
Relationships:					
<ul style="list-style-type: none"> • Abstract use case: Manage section 					
Business Rules:					
<ul style="list-style-type: none"> • User already logged in the system as “Author” role. • The section will be completely removed from database including related information of section. • Learner cannot see or access to section that has been removed • Only author who owns the course can delete the section. Otherwise, other authors cannot modify 					

Table 39: Remove section specification

2.3.4 <System Admin> Overview Use Case

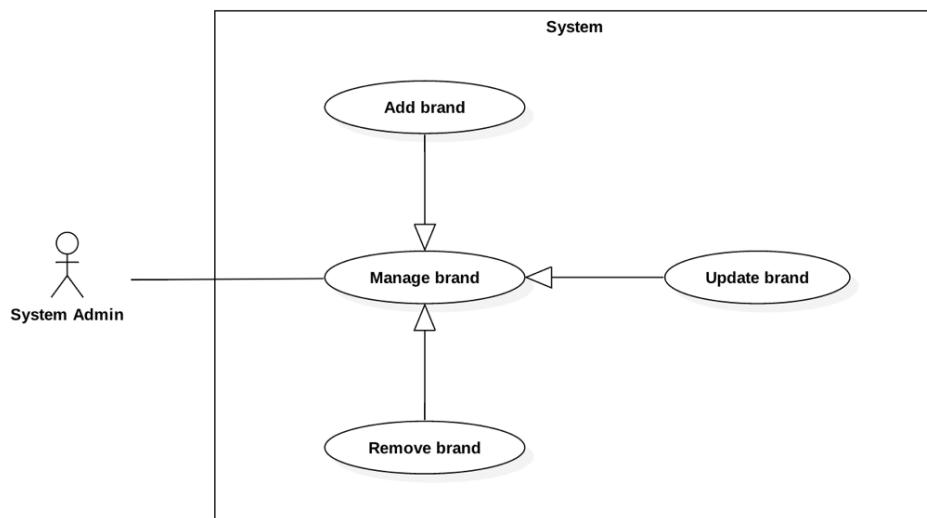


Figure 38: <System admin> Overview Use Case

2.3.4.1 <System Admin> Add brand

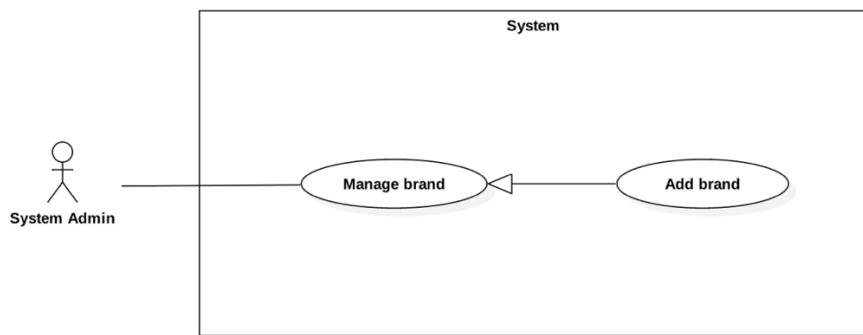


Figure 39: <System Admin> Add brand (UC_TRAISY33)

USE CASE - UC_TRAISY33			
Use Case No.	UC_TRAISY33	Use Case Version	0.2
Use Case Name	Add brand		
Author	Đặng Thế Anh		
Date	26/3/2018	Priority	High
Actor:			
<ul style="list-style-type: none"> • System admin 			
Summary:			
<ul style="list-style-type: none"> • This use case allows authorized system admin to create a brand. 			
Goal:			
<ul style="list-style-type: none"> • To add a new brand to the system. 			
Triggers:			
<ul style="list-style-type: none"> • Actor sends add brand command. 			
Preconditions:			
<ul style="list-style-type: none"> • User logged in as System Admin role. 			
Post conditions:			

- Success: Actor successfully adds new brand to the database and system redirects actor to that brand's details page.
- Fail: System shows error messages.

Main Success Scenario:

Step	Actor Action	System Response
1	System admin goes to “Quản lý thương hiệu” page	System displays page which include: <ul style="list-style-type: none"> • List of all brands that system contains • Add new brand button
2	System admin clicks add new brand button	System displays modal with field: “Nhập tên thương hiệu” requires user to input brand's name.
3	System admin clicks “Thêm” button	[Exception no.1] System redirects user to details of that brand.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	System admin does not input required field	System notices that “Tên thương hiệu không được trống”

Relationships:

- Abstract use case: Manage brand

Business Rules:

- User already logged in the system as “System admin” role.
- Brand's name must not be empty and between 2-80 characters.
- When created, the brand will have default brand's logo. System admin can change the logo in brand overview page (Update brand usecase).
- When created, the brand will be “Active” as default. System admin can change the status in brand overview page (Update brand usecase).

Table 40: Add brand specification

2.3.4.2 <System Admin> Update brand

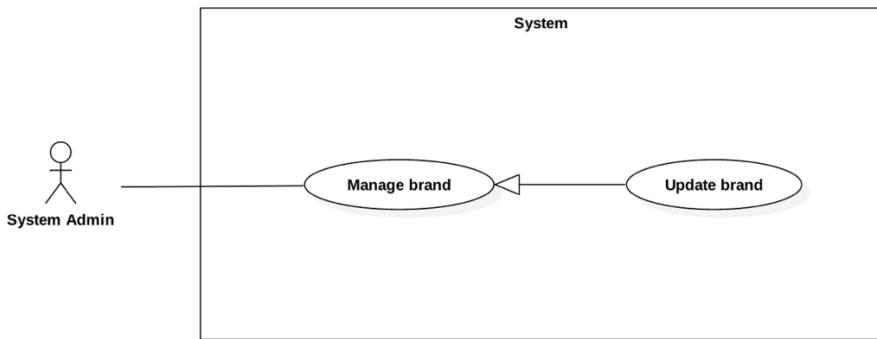


Figure 40: <System Admin> Update brand (UC_TRAISY34)

USE CASE – UC_TRAISY34			
Use Case No.	UC_TRAISY34	Use Case Version	0.2
Use Case Name	Update brand		
Author	Đặng Thế Anh		
Date	26/3/2018	Priority	High
Actor:			
<ul style="list-style-type: none"> • System admin 			
Summary:			
<ul style="list-style-type: none"> • This use case allows authorized system admin to update a specific brand. 			
Goal:			
<ul style="list-style-type: none"> • To update a brand from the system. 			
Triggers:			
<ul style="list-style-type: none"> • Actor sends update brand command. 			
Preconditions:			
<ul style="list-style-type: none"> • User logged in as System Admin role. 			
Post conditions:			
<ul style="list-style-type: none"> • Success: Actor successfully updates a brand to the database. • Fail: System shows error messages. 			
Main Success Scenario:			
Step	Actor Action	System Response	
1	System admin goes to “Quản lý thương hiệu” page	System displays page which include: <ul style="list-style-type: none"> • List of all brands that system contains • Add new brand button 	
2	System admin clicks on a specific brand in the list	System redirects actor to page which include: <ul style="list-style-type: none"> • “Tên thương hiệu”: text field • “Mô tả”: text area • Picture of brand • “Lưu chỉnh sửa” (save) button • Delete brand button 	

3	System admin clicks “Lưu chỉnh sửa” button	[Exception no.1] System shows “Cập nhật thành công” popup
---	--	--

Alternative Scenario:

Step	Actor Action	System Response
1	System admin goes to “Quản lý thương hiệu” page	System displays page which include: <ul style="list-style-type: none">• List of all brands that system contains• Add new brand button
2	System admin clicks add new brand button	System displays modal with field: “Nhập tên thương hiệu” requires user to input brand's name.
3	System admin clicks “Thêm” button	System redirects user to details of that brand.
4	System admin can update information of brand through add brand use case (UC_TRAISY33)	System shows “Cập nhật thành công” popup

Exceptions:

No	Cause	System Response
1	System admin does not input required field	System notices that “Tên thương hiệu không được trống”

Relationships:

- Abstract use case: Manage brand

Business Rules:

- User already logged in the system as “System admin” role.
- Brand's name must not be empty and between 2-80 characters.
- System admin can update any brand in the system.

Table 41: Update brand specification

2.3.3.3 <System Admin> Remove brand

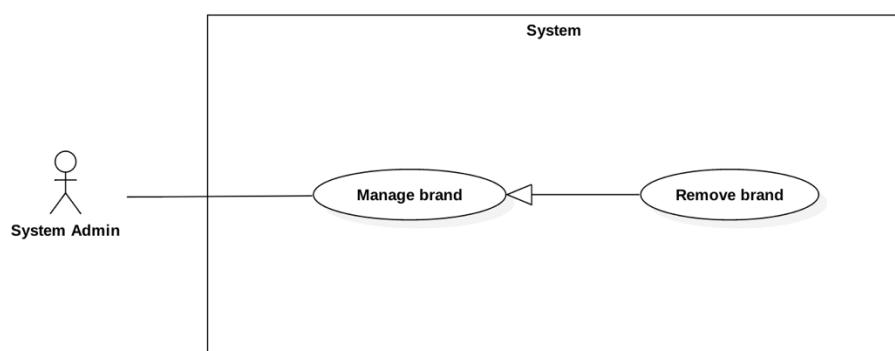


Figure 41: <System Admin> Update brand (UC_TRAISY35)

USE CASE - UC_TRAISY35																		
Use Case No.	UC_TRAISY35	Use Case Version	0.2															
Use Case Name	Remove brand																	
Author	Đặng Thế Anh																	
Date	26/3/2018	Priority	High															
Actor:	<ul style="list-style-type: none"> System admin 																	
Summary:	<ul style="list-style-type: none"> This use case allows authorized system admin to update a specific brand. 																	
Goal:	<ul style="list-style-type: none"> To delete a brand from the system. 																	
Triggers:	<ul style="list-style-type: none"> Actor sends delete brand command. 																	
Preconditions:	<ul style="list-style-type: none"> User logged in as System Admin role. 																	
Post conditions:	<ul style="list-style-type: none"> Success: Actor successfully deletes a brand from the database. Fail: System shows error messages. 																	
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>System admin goes to “Quản lý thương hiệu” page</td><td> System displays page which include: <ul style="list-style-type: none"> List of all brands that system contains Add new brand button </td></tr> <tr> <td>2</td><td>System admin clicks on a specific brand in the list</td><td> System redirects actor to page which include: <ul style="list-style-type: none"> “Tên thương hiệu”: text field “Mô tả”: text area Picture of brand “Lưu chỉnh sửa” (save) button Delete brand button </td></tr> <tr> <td>3</td><td>System admin clicks delete brand button</td><td>System shows modal that confirm deletion.</td></tr> <tr> <td>4</td><td>System admin clicks confirm button</td><td>System shows “Xoá thành công” and redirects actor to list brand page.</td></tr> </tbody> </table>			Step	Actor Action	System Response	1	System admin goes to “Quản lý thương hiệu” page	System displays page which include: <ul style="list-style-type: none"> List of all brands that system contains Add new brand button 	2	System admin clicks on a specific brand in the list	System redirects actor to page which include: <ul style="list-style-type: none"> “Tên thương hiệu”: text field “Mô tả”: text area Picture of brand “Lưu chỉnh sửa” (save) button Delete brand button 	3	System admin clicks delete brand button	System shows modal that confirm deletion.	4	System admin clicks confirm button	System shows “Xoá thành công” and redirects actor to list brand page.
Step	Actor Action	System Response																
1	System admin goes to “Quản lý thương hiệu” page	System displays page which include: <ul style="list-style-type: none"> List of all brands that system contains Add new brand button 																
2	System admin clicks on a specific brand in the list	System redirects actor to page which include: <ul style="list-style-type: none"> “Tên thương hiệu”: text field “Mô tả”: text area Picture of brand “Lưu chỉnh sửa” (save) button Delete brand button 																
3	System admin clicks delete brand button	System shows modal that confirm deletion.																
4	System admin clicks confirm button	System shows “Xoá thành công” and redirects actor to list brand page.																
Alternative Scenario:	N/A																	
Exceptions:	N/A																	
Relationships:	<ul style="list-style-type: none"> Abstract use case: Manage brand 																	
Business Rules:																		

- User already logged in the system as “System admin” role.
- The removed brand’s status will be change to “-1” in database. Therefore, any data such as course, program, and lecture relating to brand is not completely deleted from database.
- Users that relating to brand cannot see or access to brand that has been removed. The system will show an error page for this.

Table 42: Remove brand specification

2.3.4 <Manager> Overview Use Case

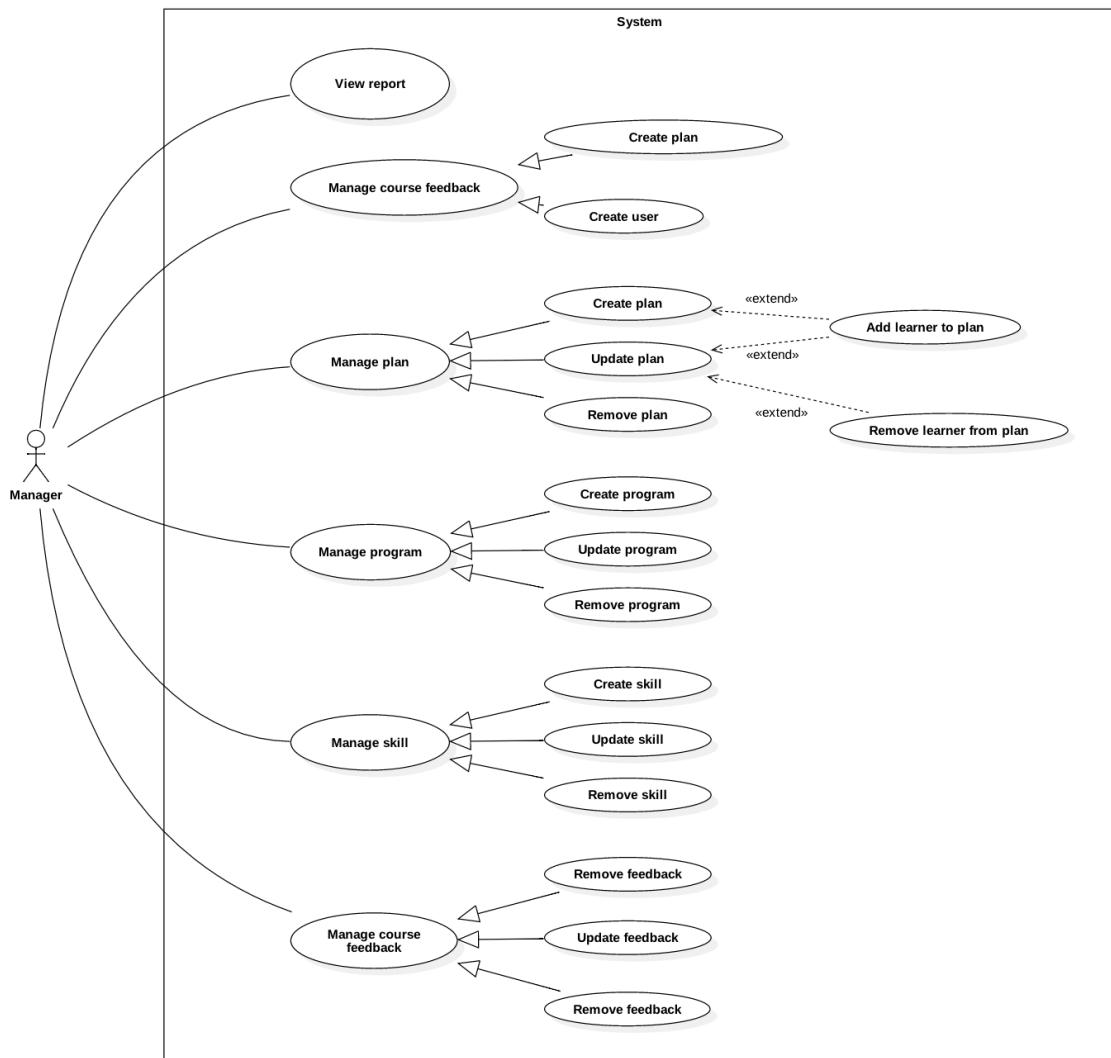


Figure 42: <Manager> Overview Use Case

2.3.4.1 <Manager> View report

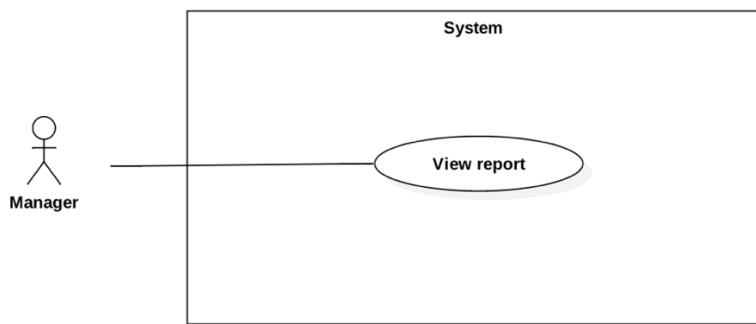


Figure 43: <Manager> View report (UC_TRAISY36)

USE CASE - UC_TRAISY36					
Use Case No.	UC_TRAISY36	Use Case Version	0.2		
Use Case Name	View report of learners and authors				
Author	Lê Phúc Thịnh				
Date	26/3/2018	Priority	Normal		
Actor:					
• Manager					
Summary:					
• This use case allows manager view report.					
Goal:					
• View report in more detail.					
Triggers:					
• Manager clicks on view report.					
Preconditions:					
• Manager logged in the system.					
Post conditions:					
• Success: Manager views report of learners and authors.					
• Fail: System shows error messages.					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Manager visits to 'Báo cáo' page	System redirect to "Báo cáo" page			
2	Manager clicks on specific plan	System redirect to "Chi tiết báo cáo kế hoạch" page			
Alternative Scenario: N/A					
Exceptions: N/A					
Relationships: N/A					
Business Rules:					
• User already logged in the system as "Manager" role.					
• The pie chart shows the number of active plans and the number of plans that have expired					
• Another pie chart shows the progress of the plans, including completed and unfinished					

- A horizontal bar chart shows the number of students and trainers in each plan
- A list of plans showing the number of learner completed in that plan
- A table contains list of learners and authors is included in the plan, and details of those learners / authors can be viewed
- Detailed information about each trainee and trainer

Table 43: View report specification

2.3.4.2 <Manager> Update user's information

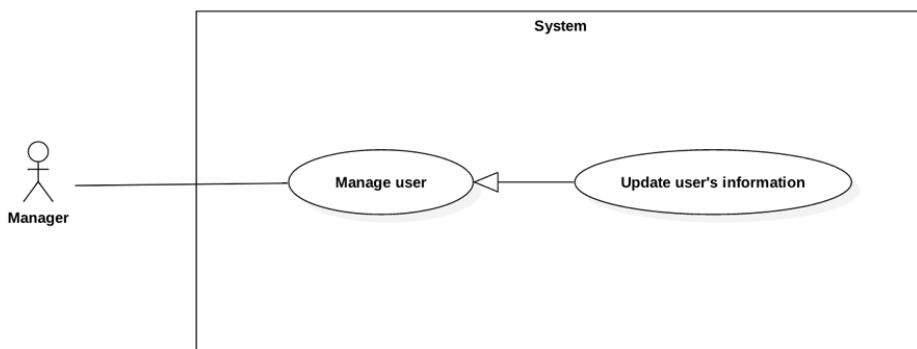


Figure 44: <Manager> Update user's information (UC_TRAISY37)

USE CASE - UC_TRAISY37			
Use Case No.	UC_TRAISY37	Use Case Version	0.2
Use Case Name	Update user's information		
Author	Đặng Thế Anh		
Date	26/3/2018	Priority	Normal
Actor:			
• Manager			
Summary:			
• This use case allows manager updates user's information.			
Goal:			
• Update user's information successfully.			
Triggers:			
• Manager clicks on “Chỉnh sửa” button of specific user.			
Preconditions:			
• Manager logged in the system.			
Post conditions:			
• Success: Manager update user's information successfully.			
• Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager visits to “Quản lý người dùng” page.	System redirects to “Quản lý người dùng” page.	

2	Manager clicks on “Chỉnh sửa” button.	System displays “Chỉnh sửa thông tin người dùng” popup with: <ul style="list-style-type: none"> Tên đăng nhập: text input Mật khẩu: text input Họ và tên: text input Chức danh: text input Vai trò: dropdown list Trạng thái: dropdown list
3	Manager updates necessary information of that user.	
4	Manager clicks on “Lưu” button.	System saves updated information and displays them immediately.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
	Manager enters wrong some fields with requirement.	System notices that guest need to re-input all these field: <ul style="list-style-type: none"> “Họ tên”: System displays warning message: “Họ và tên phải từ 5 tới 80 ký tự”. “Mật khẩu”: System displays warning message: “Mật khẩu phải từ 4 tới 30 ký tự”.

Relationships:

- Abstract use case: Manage user

Business Rules:

- User already logged in the system as “Admin” role.
- Admin can only update users that belongs to his brand; Admin can not update other brands’ users.
- “Họ tên” must be from 5 to 80 characters.
- “Mật khẩu” must be from 4 to 30 characters.
- “Mật khẩu” must be encrypted.
- “Tên đăng nhập” must not be edited.
- “Chức vụ” must be selected.
- “Trạng thái” must be selected.

Table 44: Update user's information specification

2.3.4.3 <Manager> Create user

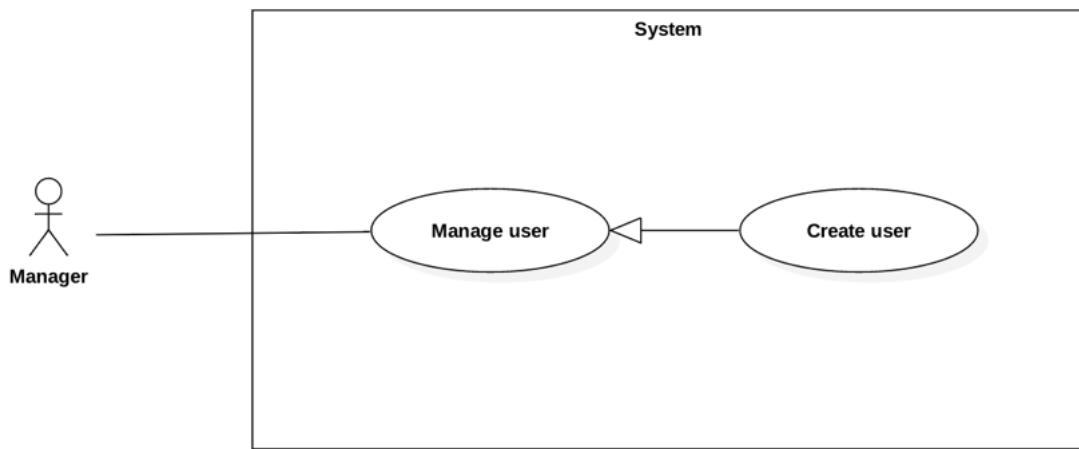


Figure 45: <Manager> Create user (UC_TRAISY38)

USE CASE - UC_TRAISY38			
Use Case No.	UC_TRAISY38	Use Case Version	0.2
Use Case Name	Create user		
Author	Đặng Thế Anh		
Date	26/3/2018	Priority	High
Actor:			
<ul style="list-style-type: none"> Manager 			
Summary:			
<ul style="list-style-type: none"> This use case allows manager creates new user. 			
Goal:			
<ul style="list-style-type: none"> Create new user successfully. 			
Triggers:			
<ul style="list-style-type: none"> Manager clicks on “Thêm người dùng” button. 			
Preconditions:			
<ul style="list-style-type: none"> Manager logged in the system. 			
Post conditions:			
<ul style="list-style-type: none"> Success: Manager creates new user successfully. Fail: System shows error messages. 			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager visits to “Quản lý người dùng” page.	System redirects to “Quản lý người dùng” page.	
2	Manager clicks on “Thêm người dùng” button.	System displays “Thông tin người dùng mới” popup with: <ul style="list-style-type: none"> Tên đăng nhập: text input Mật khẩu: text input Họ và tên: text input Chức danh: text input Vai trò: dropdown list 	

		<ul style="list-style-type: none"> Trạng thái: dropdown list
3	Manager enters necessary information for new user.	
4	Manager clicks on “Lưu” button.	System saves that information and displays them immediately.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
	Manager enters wrong some fields with requirement.	<p>System notices that guest need to re-input all these field:</p> <ul style="list-style-type: none"> “Tên đăng nhập”: System displays warning message: “Tên đăng nhập phải từ 4 tới 35 ký tự”. “Họ tên”: System displays warning message: “Họ và tên phải từ 5 tới 80 ký tự”. “Mật khẩu”: System displays warning message: “Mật khẩu phải từ 4 tới 30 ký tự”.

Relationships:

- Abstract use case: Manage user

Business Rules:

- “Tên đăng nhập” must be from 4 to 35 characters.
- “Họ tên” must be from 5 to 80 characters.
- “Mật khẩu” must be from 4 to 30 characters.
- “Mật khẩu” must be encrypted.

Table 45: Create user specification

2.3.4.4 <Manager> Create plan

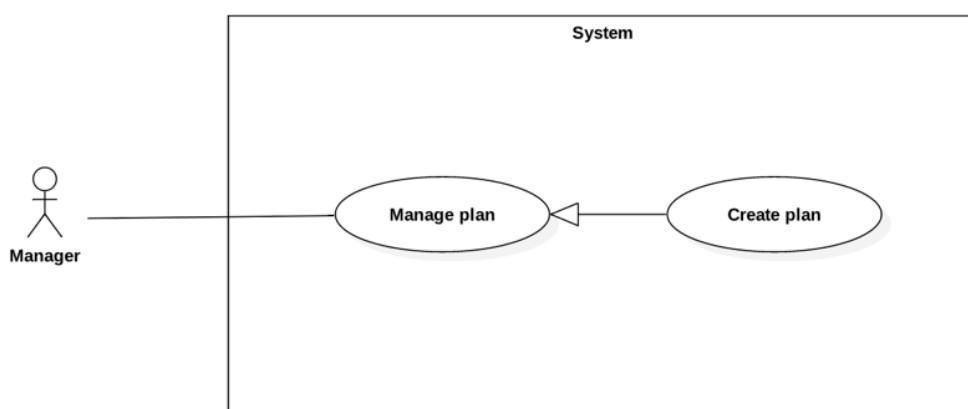


Figure 46: <Manager> Create plan (UC_TRAISY39)

USE CASE – UC_TRAISY39

Use Case No.	UC_TRAISY39	Use Case Version	0.2
Use Case Name	Create plan		
Author	Võ Thạch Nguyên		
Date	26/3/2018	Priority	High
Actor:			
<ul style="list-style-type: none"> Manager 			
Summary:			
<ul style="list-style-type: none"> This use case allows manager creates new plan. 			
Goal:			
<ul style="list-style-type: none"> Create new plan successfully. 			
Triggers:			
<ul style="list-style-type: none"> Manager clicks on “Thêm người dùng” button. 			
Preconditions:			
<ul style="list-style-type: none"> Manager logged in the system. 			
Post conditions:			
<ul style="list-style-type: none"> Success: Manager creates new plan successfully. Fail: System shows error messages. 			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager visits to “Quản lý kế hoạch” page.	System redirects to “Quản lý kế hoạch” page.	
2	Manager clicks on “Thêm kế hoạch” button.	System displays “Nhập tên kế hoạch” popup with: <ul style="list-style-type: none"> Tên kế hoạch: text input 	
3	Manager enters name for new plan		
4	Manager clicks on “Thêm” button.	System creates new plan and displays them immediately.	
Alternative Scenario: N/A			
Exceptions:			
No	Cause	System Response	
	Manager enters wrong some fields with requirement.	System notices that guest need to re-input all these field: <ul style="list-style-type: none"> “Tên kế hoạch”: System displays warning message: “Tên kế hoạch phải từ 5 đến 80 ký tự”. 	
Relationships:			
<ul style="list-style-type: none"> Abstract use case: Manage plan 			
Business Rules:			
<ul style="list-style-type: none"> “Tên kế hoạch” must be from 5 to 80 characters. 			

Table 46: Create plan specification

2.3.4.5 <Manager> Update plan

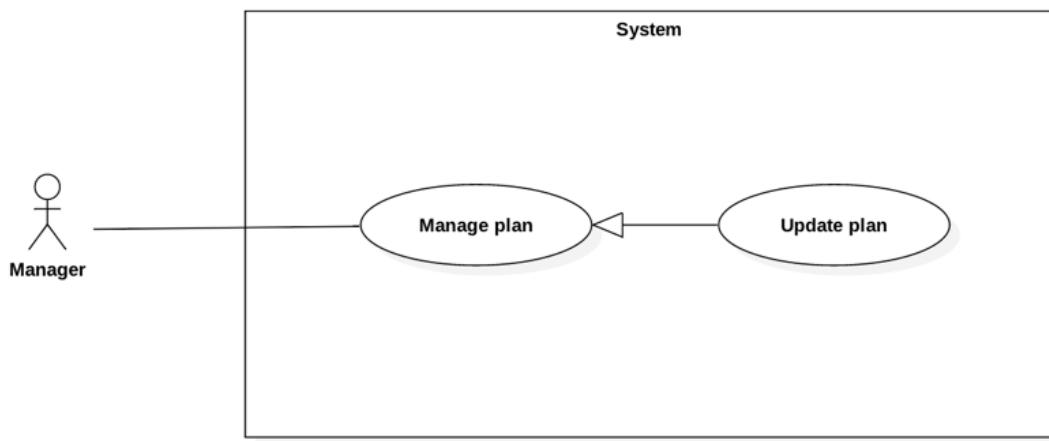


Figure 47: <Manager> Update plan (UC_TRAISY40)

USE CASE - UC_TRAISY40			
Use Case No.	UC_TRAISY40	Use Case Version	0.2
Use Case Name	Update plan		
Author	Võ Thạch Nguyên		
Date	26/3/2018	Priority	High
Actor:			
• Manager			
Summary:			
• This use case allows manager update plan's information.			
Goal:			
• Update plan successfully.			
Triggers:			
• Manager clicks on “Chi tiết” button.			
Preconditions:			
• Manager logged in the system.			
Post conditions:			
• Success: Manager updates plan's information successfully.			
• Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager visits to “Quản lí kế hoạch” page.	System redirects to “Quản lí kế hoạch” page.	
2	Manager clicks on “Chi tiết” button of a specific plan.	System redirects to “Quản lí chi tiết kế hoạch” of that plan, includes five sections: <ul style="list-style-type: none">• “Giới thiệu”• “Hình ảnh”• “Giảng viên”• “Học viên”	

		<ul style="list-style-type: none"> “Nội dung kế hoạch”
3	Manager selects a specific section.	<p>System displays corresponding section:</p> <ul style="list-style-type: none"> “Giới thiệu” <ul style="list-style-type: none"> “Tên kế hoạch” “Thời gian” “Mô tả” “Hình ảnh” <ul style="list-style-type: none"> “Chọn ảnh” “Giảng viên” <ul style="list-style-type: none"> “Thêm giảng viên” “Học viên” <ul style="list-style-type: none"> “Thêm học viên” “Nội dung kế hoạch”
4	Manager updates necessary information	
5	Manager clicks on “Lưu chỉnh sửa” button.	System saves updated information immediately and display it.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
	Manager enters wrong some fields with requirement.	<p>System notices that guest need to re-input all these field:</p> <ul style="list-style-type: none"> “Tên kế hoạch”: System displays warning message: “Tên kế hoạch phải từ 5 tới 80 ký tự”. “Thời gian”: System displays warning message: “Thời gian bắt đầu phải trước thời gian kết thúc”

Relationships:

- Abstract use case: Manage plan

Business Rules:

- “Tên kế hoạch” must be from 5 to 80 characters
- The start time must be before the end time
- Manager can easily add learners and authors
- Manager can easily change the image which representing that plan
- Managers can see which programs are in the plan, and easily manage them

Table 47: Update plan specification

2.3.4.6 <Manager> Add learner to plan

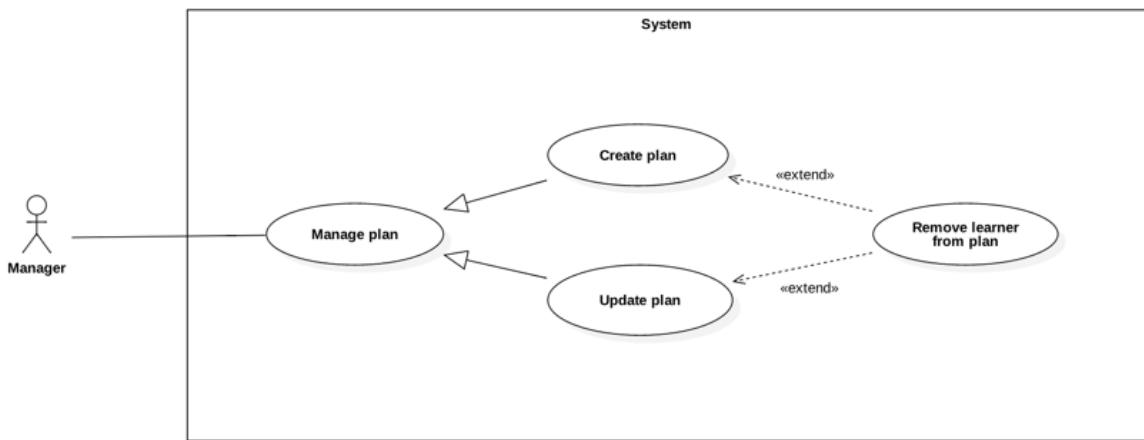


Figure 48: <Manager> Add learner to plan (UC_TRAISY41)

USE CASE - UC_TRAISY41			
Use Case No.	UC_TRAISY41	Use Case Version	0.2
Use Case Name	Add learner to plan		
Author	Võ Thạch Nguyên		
Date	26/3/2018	Priority	High
Actor:			
• Manager			
Summary:			
• This use case allows manager add learner to plan.			
Goal:			
• Add learner to plan successfully.			
Triggers:			
• Manager clicks on “Thêm học viên” button.			
Preconditions:			
• Manager logged in the system.			
Post conditions:			
• Success: Manager adds learner to plan successfully.			
• Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager visits to “Quản lý kế hoạch” page.	System redirects to “Quản lý kế hoạch” page.	
2	Manager clicks on “Thêm kế hoạch” button.	System displays “Nhập tên kế hoạch” popup with: • “Tên kế hoạch”: text input	
3	Manager enters name for new plan		
4	Manager clicks on “Thêm” button.	System creates new plan and redirect to “Quản lý chi tiết kế hoạch page”.	

		System display five sections: <ul style="list-style-type: none">“Giới thiệu”“Hình ảnh”“Giảng viên”“Học viên”“Nội dung kế hoạch”
5	Manager visits to “Quản lí kế hoạch” page.	System redirects to “Quản lí kế hoạch” page.
6	Manager selects “Học viên” section	System displays “Học viên” section
7	Manager clicks on “Thêm học viên” button	System displays “Danh sách học viên” popup.
8	Manager clicks on “Add” button of specific learner	System add this learner to plan immediately

Alternative Scenario:

Step	Actor Action	System Response
1	Manager visits to “Quản lí kế hoạch” page.	System redirects to “Quản lí kế hoạch” page.
2	Manager clicks on “Chi tiết” button of a specific plan.	System redirects to “Quản lí chi tiết kế hoạch” of that plan, includes five sections: <ul style="list-style-type: none">“Giới thiệu”“Hình ảnh”“Giảng viên”“Học viên”“Nội dung kế hoạch”
3	Manager selects “Học viên” section	System displays “Học viên” section
4	Manager clicks on “Thêm học viên” button	System displays “Danh sách học viên” popup.
5	Manager clicks on “Add” button of specific learner	System add this learner to plan immediately

Exceptions: N/A

Relationships:

- Abstract use case: Manage plan

Business Rules:

- Managers can add trainees and add them to the plan, and can also remove trainees from the plan

Table 48: Add learner to plan

2.3.4.7 <Manager> Remove plan

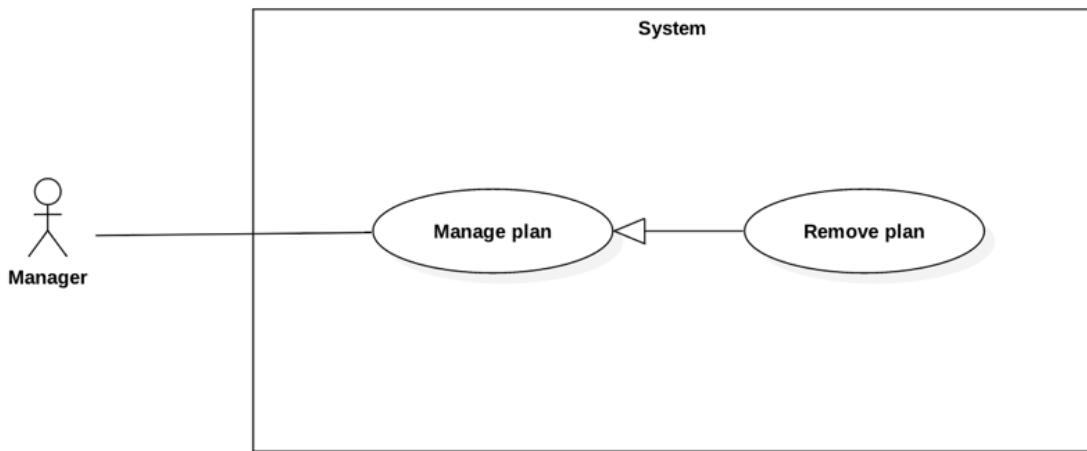


Figure 49: <Manager> Remove plan (UC_TRAISY42)

USE CASE - UC_TRAISY42					
Use Case No.	UC_TRAISY42	Use Case Version	0.2		
Use Case Name	Remove plan				
Author	Võ Thạch Nguyên				
Date	26/3/2018	Priority	High		
Actor:					
• Manager					
Summary:					
• This use case allows manager removes a plan.					
Goal:					
• Remove plan successfully.					
Triggers:					
• Manager clicks on “Xoá kế hoạch” button.					
Preconditions:					
• Manager logged in the system.					
Post conditions:					
• Success: Manager removes plan successfully.					
• Fail: System shows error messages.					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Manager visits to “Quản lý kế hoạch” page.	System redirects to “Quản lý kế hoạch” page.			
2	Manager clicks on “Xoá kế hoạch” button.	System displays “Xác nhận xoá kế hoạch” popup.			
3	Manager click on “Xoá” button	System removes plan and redirects to “Quản lý kế hoạch” page			
Alternative Scenario: N/A					
Exceptions: N/A					
Relationships:					

- Abstract use case: Manage plan

Business Rules:

- The removed plan's status will be change to "-1" in database. Therefore, any data such as course, program, and lecture relating to plan is not completely deleted from database.
- Users included author and learner that relating to plan cannot see or access to plan that has been removed.
- After plan was removed, web must be redirected to list plan page immediately.

Table 49: Remove plan specification

2.3.4.8 <Manager> Create program

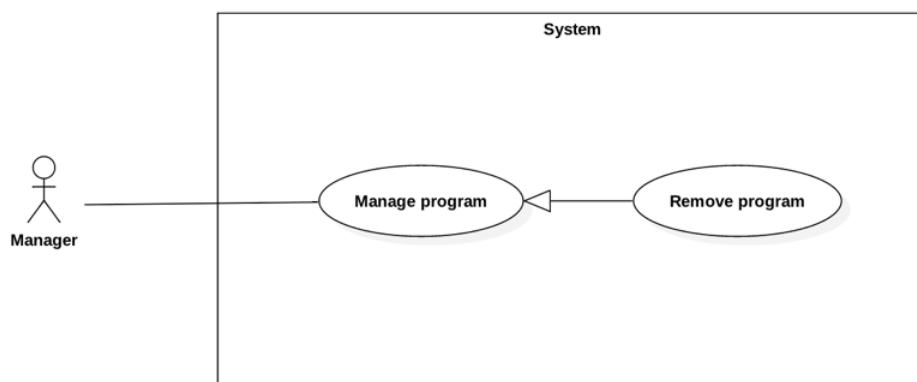


Figure 50: <Manager> Create program (UC_TRAISY43)

USE CASE - UC_TRAISY43			
Use Case No.	UC_TRAISY43	Use Case Version	0.2
Use Case Name	Create program		
Author	Võ Thạch Nguyên		
Date	26/3/2018	Priority	High
Actor:			
• Manager			
Summary:			
• This use case allows manager creates new program.			
Goal:			
• Create new program successfully.			
Triggers:			
• Manager clicks on "Thêm kế hoạch" button.			
Preconditions:			
• Manager logged in the system.			
Post conditions:			
• Success: Manager creates new program successfully.			
• Fail: System shows error messages.			
Main Success Scenario:			

Step	Actor Action	System Response
1	Manager visits to “Quản lý kế hoạch” page.	System redirects to “Quản lý kế hoạch” page.
2	Manager selects “Nội dung kế hoạch” section.	System displays “Nội dung kế hoạch” section.
2	Manager clicks on “Thêm chương trình” button.	System displays “Nhập tên chương trình” popup with: <ul style="list-style-type: none"> Tên chương trình: text input Thời gian: datetime picker
3	Manager enters name for new program	
4	Manager clicks on “Tạo chương trình” button.	System creates new program and displays them immediately.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
	Manager enters wrong some fields with requirement.	System notices that guest need to re-input all these field: <ul style="list-style-type: none"> “Tên chương trình”: System displays warning message: “Tên chương trình phải từ 5 tới 80 ký tự”. “Thời gian”: System displays warning message: “Thời gian bắt đầu phải trước thời gian kết thúc”

Relationships:

- Abstract use case: Manage program

Business Rules:

- “Tên chương trình” must be from 5 to 80 characters
- The start time must be before the end time
- After adding a new program, the browser will automatically redirect to manage program page.

Table 50: Create program specification

2.3.4.9 <Manager> Update program

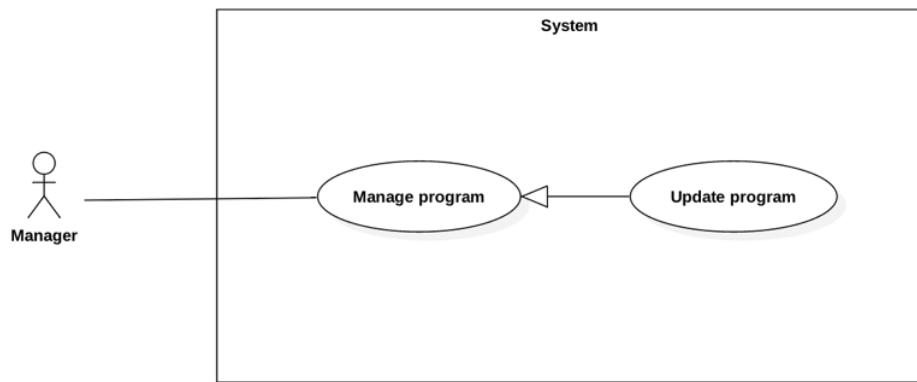


Figure 51: <Manager> Update program (UC_TRAISY44)

USE CASE - UC_TRAISY44			
Use Case No.	UC_TRAISY44	Use Case Version	0.2
Use Case Name	Update program		
Author	Võ Thạch Nguyên		
Date	26/3/2018	Priority	High
Actor:			
<ul style="list-style-type: none"> Manager 			
Summary:			
<ul style="list-style-type: none"> This use case allows manager update program's information. 			
Goal:			
<ul style="list-style-type: none"> Update program successfully. 			
Triggers:			
<ul style="list-style-type: none"> Manager clicks on “Quản lý” button. 			
Preconditions:			
<ul style="list-style-type: none"> Manager logged in the system. 			
Post conditions:			
<ul style="list-style-type: none"> Success: Manager updates program's information successfully. Fail: System shows error messages. 			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager visits to “Quản lý kế hoạch” page.	System redirects to “Quản lý kế hoạch” page.	
2	Manager clicks on “Chi tiết” button of a specific plan.	System redirects to “Quản lý chi tiết kế hoạch” of that plan, includes five sections: <ul style="list-style-type: none"> “Giới thiệu” “Hình ảnh” “Giảng viên” “Học viên” “Nội dung kế hoạch” 	

3	Manager selects “Nội dung kế hoạch” section.	System displays list of program that belongs to plan.
4	Manager clicks on “Chỉnh sửa” button of specific program	System displays “Chỉnh sửa kế hoạch” popup with: <ul style="list-style-type: none"> Tên chương trình: text input Thời gian: datetime picker
5	Manager clicks on “Lưu chỉnh sửa” button.	System saves updated information immediately and display it.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
	Manager enters wrong some fields with requirement.	System notices that guest need to re-input all these field: <ul style="list-style-type: none"> “Tên kế hoạch”: System displays warning message: “Tên chương trình phải từ 5 tới 80 ký tự”. “Thời gian”: System displays warning message: “Thời gian bắt đầu phải trước thời gian kết thúc”

Relationships:

- Abstract use case: Manage program

Business Rules:

- “Tên chương trình” must be from 5 to 80 characters
- The start time must be before the end time

Table 51: Update program specification

2.3.4.10 <Manager> Remove program

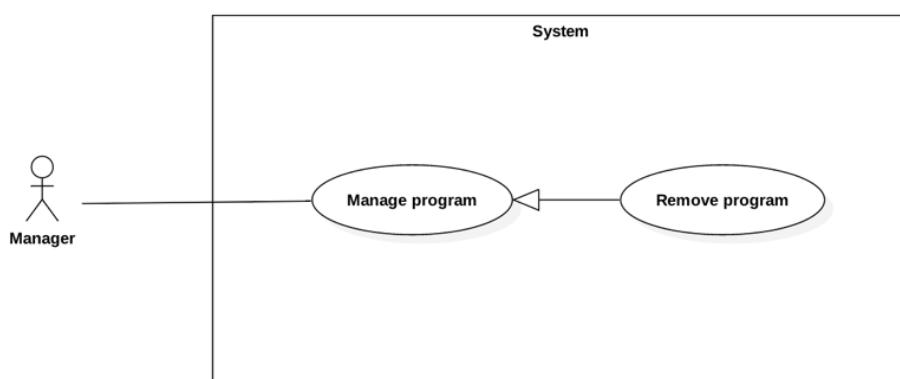


Figure 52: <Manager> Remove program (UC_TRAISY45)

USE CASE - UC_TRAISY45			
Use Case No.	UC_TRAISY45	Use Case Version	0.2
Use Case Name	Remove program		

Author	Võ Thạch Nguyên				
Date	26/3/2018	Priority	High		
Actor:					
<ul style="list-style-type: none"> Manager 					
Summary:					
<ul style="list-style-type: none"> This use case allows manager removes a program. 					
Goal:					
<ul style="list-style-type: none"> Remove plan successfully. 					
Triggers:					
<ul style="list-style-type: none"> Manager clicks on “Xoá chương trình” button. 					
Preconditions:					
<ul style="list-style-type: none"> Manager logged in the system. 					
Post conditions:					
<ul style="list-style-type: none"> Success: Manager removes program successfully. Fail: System shows error messages. 					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Manager visits to “Quản lý kế hoạch” page.	System redirects to “Quản lý kế hoạch” page.			
2	Manager clicks on “Chi tiết” button of a specific plan.	System redirects to “Quản lý chi tiết kế hoạch” of that plan, includes five sections: <ul style="list-style-type: none"> “Giới thiệu” “Hình ảnh” “Giảng viên” “Học viên” “Nội dung kế hoạch” 			
3	Manager selects “Nội dung kế hoạch” section.	System displays list of program that belongs to plan.			
4	Manager click on “Xoá chương trình” button of specific program	System displays “Xác nhận xoá chương trình” popup.			
5	Manager click on “Xoá” button	System removes that program immediately.			
Alternative Scenario: N/A					
Exceptions: N/A					
Relationships:					
<ul style="list-style-type: none"> Abstract use case: Manage program 					
Business Rules:					
<ul style="list-style-type: none"> The removed program's status will be change to “-1” in database. Therefore, any data such as course, author, learner, and lecture relating to program is not completely deleted from database. 					

- Users included author and learner that relating to plan cannot see or access to plan that has been removed.
- After the program is removed from the plan, the list of programs will be refreshed.

Table 52: Remove program specification

2.3.4.11 <Manager> Create skill

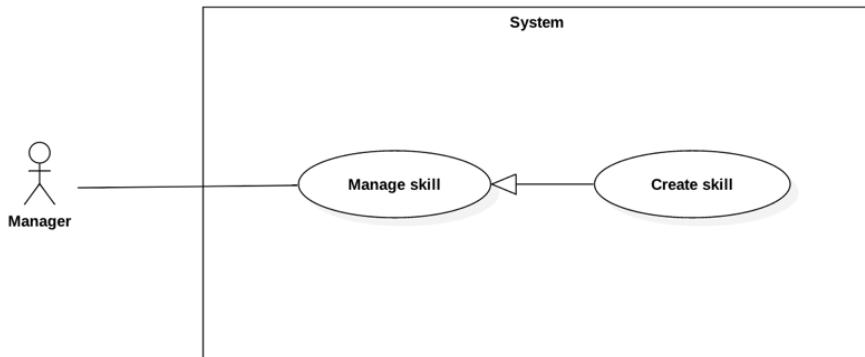


Figure 53: <Manager> Create skill (UC_TRAISY46)

USE CASE – UC_TRAISY46			
Use Case No.	UC_TRAISY46	Use Case Version	0.2
Use Case Name	Create skill		
Author	Đặng Thế Anh		
Date	26/3/2018	Priority	High
Actor:			
• Manager			
Summary:			
• This use case allows manager creates new skill.			
Goal:			
• Create new skill successfully.			
Triggers:			
• Manager clicks on “Thêm kỹ năng” button.			
Preconditions:			
• Manager logged in the system.			
Post conditions:			
• Success: Manager creates new skill successfully.			
• Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager visits to “Quản lý kỹ năng” page.	System redirects to “Quản lý kỹ năng” page.	
2	Manager clicks on “Thêm kỹ năng” button	System displays “Thêm kỹ năng” popup with: <ul style="list-style-type: none"> “Tên kỹ năng”: text input 	

		<ul style="list-style-type: none"> “Mô tả kỹ năng”: text input
3	Manager enters necessary information.	
4	Manager click on “Lưu” button	System saves new skill immediately and display it on page.

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Manager enters wrong some fields with requirement.	<p>System notices that guest need to re-input all these field:</p> <ul style="list-style-type: none"> “Tên kỹ năng”: System displays warning message: “Tên kỹ năng không được để trống”.

Relationships:

- Abstract use case: Manage skill

Business Rules:

- “Tên kỹ năng” must be not be empty.
- “Tên kỹ năng” must be not be empty and duplicated.
- After the skill is added, the modal will automatically close and the skill list will be updated and displayed immediately

Table 53: Create skill specification

2.3.4.12 <Manager> Update skill

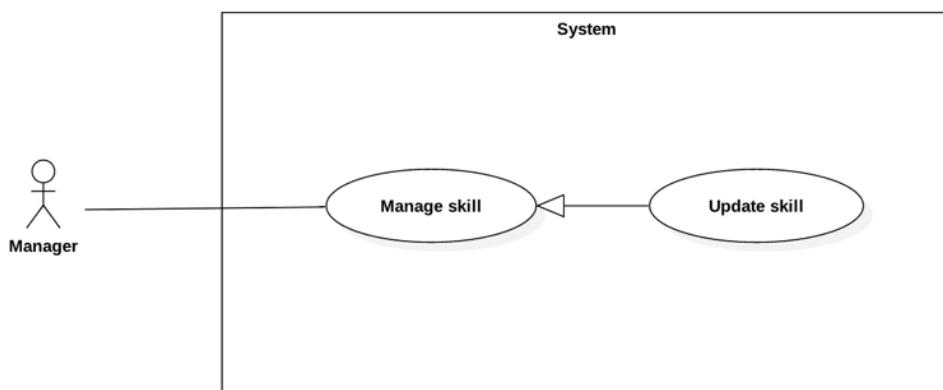


Figure 54: <Manager> Update skill (UC_TRAISY47)

USE CASE - UC_TRAISY47			
Use Case No.	UC_TRAISY47	Use Case Version	0.2
Use Case Name	Update skill		
Author	Đặng Thế Anh		
Date	26/3/2018	Priority	High
Actor:			

<i>Table 54: Update skill specification</i>		

2.3.4.13 <Manager> Remove skill

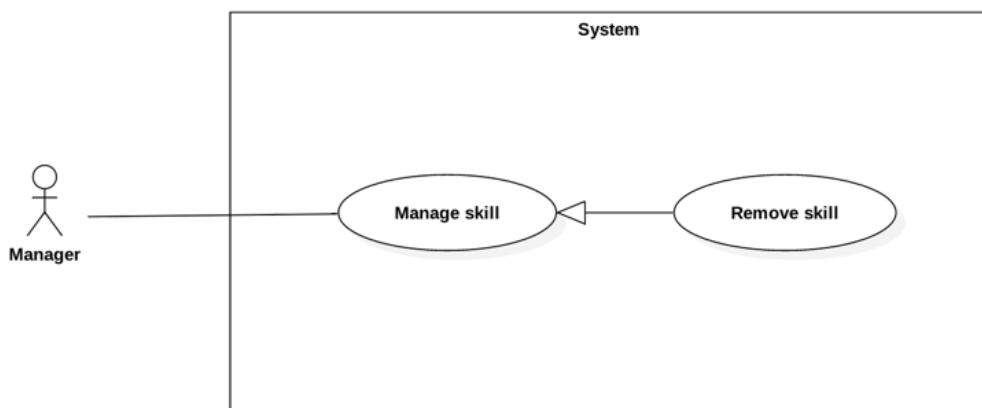


Figure 55: <Manager> Remove skill (UC_TRAISY48)

USE CASE – UC_TRAISY48															
Use Case No.	UC_TRAISY48	Use Case Version	0.2												
Use Case Name	Update skill														
Author	Đặng Thế Anh														
Date	26/3/2018	Priority	High												
Actor:	<ul style="list-style-type: none"> Manager 														
Summary:	<ul style="list-style-type: none"> This use case allows manager remove specific skill. 														
Goal:	<ul style="list-style-type: none"> Remove skill successfully. 														
Triggers:	<ul style="list-style-type: none"> Manager clicks on “Xoá” button. 														
Preconditions:	<ul style="list-style-type: none"> Manager logged in the system. 														
Post conditions:	<ul style="list-style-type: none"> Success: Manager removes skill successfully. Fail: System shows error messages. 														
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Manager visits to “Quản lý kỹ năng” page.</td><td>System redirects to “Quản lý kỹ năng” page.</td></tr> <tr> <td>2</td><td>Manager clicks on “Xoá” button</td><td>System displays “Xác nhận xoá kỹ năng” popup.</td></tr> <tr> <td>3</td><td>Manager click on “Xoá” button</td><td>System removes skill immediately.</td></tr> </tbody> </table>			Step	Actor Action	System Response	1	Manager visits to “Quản lý kỹ năng” page.	System redirects to “Quản lý kỹ năng” page.	2	Manager clicks on “Xoá” button	System displays “Xác nhận xoá kỹ năng” popup.	3	Manager click on “Xoá” button	System removes skill immediately.
Step	Actor Action	System Response													
1	Manager visits to “Quản lý kỹ năng” page.	System redirects to “Quản lý kỹ năng” page.													
2	Manager clicks on “Xoá” button	System displays “Xác nhận xoá kỹ năng” popup.													
3	Manager click on “Xoá” button	System removes skill immediately.													
Alternative Scenario:	N/A														
Exceptions:	N/A														
Relationships:	<ul style="list-style-type: none"> Abstract use case: Manage skill 														
Business Rules:															

- After the skill is removed, the modal will automatically close and the skill list will be updated and displayed immediately.

Table 55: Remove skill specification

2.3.4.14 <Manager> Add feedback

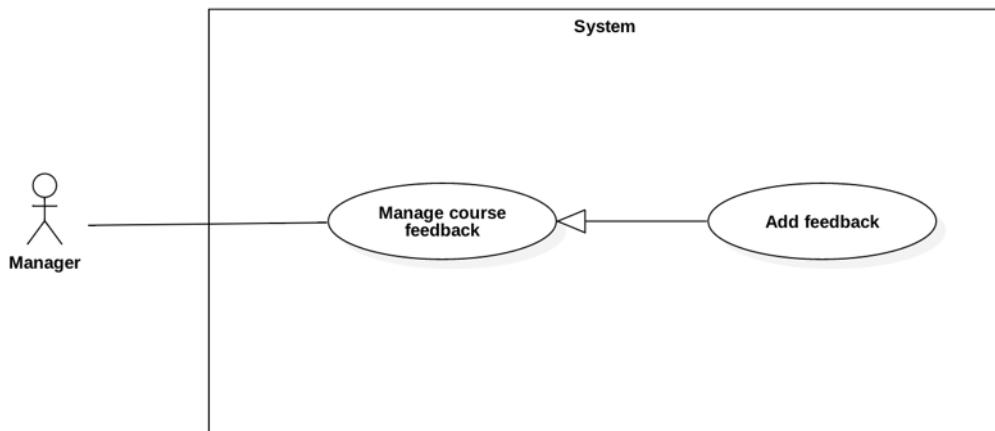


Figure 56: <Manager> Add feedback (UC_TRAISY49)

USE CASE – UC_TRAISY49			
Use Case No.	UC_TRAISY49	Use Case Version	0.2
Use Case Name	Add feedback		
Author	Võ Thạch Nguyên		
Date	26/3/2018	Priority	High
Actor:	<ul style="list-style-type: none"> Manager 		
Summary:	<ul style="list-style-type: none"> This use case allows authorized manager to add a feedback to specific course. 		
Goal:	<ul style="list-style-type: none"> To add a feedback to course. 		
Triggers:	<ul style="list-style-type: none"> Actor sends add feedback command. 		
Preconditions:	<ul style="list-style-type: none"> User logged in as manager role. 		
Post conditions:	<ul style="list-style-type: none"> Success: Actor successfully add a feedback to course. Fail: System shows error messages. 		
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager goes to “Đề xuất chỉnh sửa” page	System displays page which include: <ul style="list-style-type: none"> List of discussion “Thêm phản hồi” button 	

		<ul style="list-style-type: none"> • Menu option for each discussion
2	Manager clicks “Thêm phản hồi”	System displays modal which include: <ul style="list-style-type: none"> • “Nhập tiêu đề”: Text field • “Nội dung đề xuất”: Text area • “Thêm” button
3	Manager clicks “Thêm” button	[Exception no.1] System displays “Thêm thành công”

Alternative Scenario: N/A

Exceptions:

No	Cause	System Response
1	Manager does not input required field	System notices that: - “Nhập tiêu đề”: System displays warning message: “Tiêu đề phải từ 5 tới 80 ký tự”.

Relationships:

- Abstract use case: Manage course feedback

Business Rules:

- “Tiêu đề” must be from 5 to 80 characters.
- Discussion belongs to a specific course.
- Each feedback shows the full name, submission time, title and content of the feedback.
- After adding feedback, users can edit and delete the feedback.

Table 56: Add feedback specification

2.3.4.15 <Manager> Update feedback

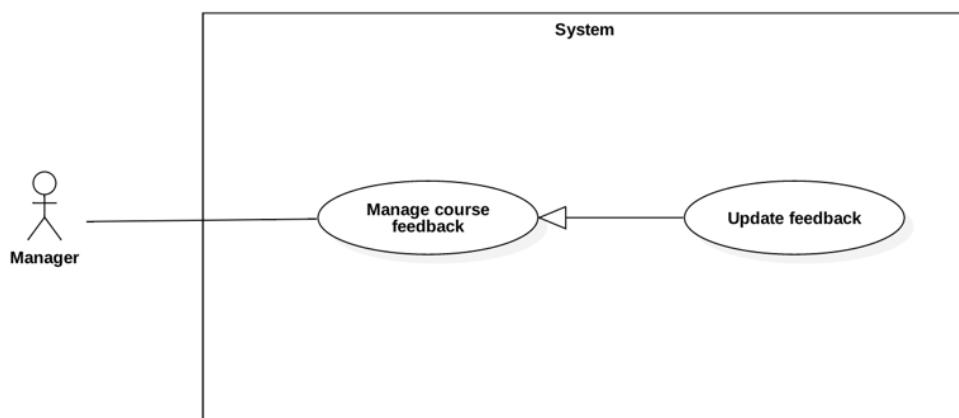


Figure 57: <Manager> Update feedback (UC_TRAISY50)

USE CASE – UC_TRAISY50			
Use Case No.	UC_TRAISY50	Use Case Version	0.2
Use Case Name	Update feedback		

Author	Võ Thạch Nguyên				
Date	26/3/2018	Priority	High		
Actor:					
<ul style="list-style-type: none"> Manager 					
Summary:					
<ul style="list-style-type: none"> This use case allows authorized manager to update a specific feedback. 					
Goal:					
<ul style="list-style-type: none"> To update a feedback to database. 					
Triggers:					
<ul style="list-style-type: none"> Actor sends update feedback command. 					
Preconditions:					
<ul style="list-style-type: none"> User logged in as manager role. 					
Post conditions:					
<ul style="list-style-type: none"> Success: Actor successfully update a specific feedback. Fail: System shows error messages. 					
Main Success Scenario:					
Step	Actor Action	System Response			
1	Manager goes to “Đề xuất chỉnh sửa” page	System displays page which include: <ul style="list-style-type: none"> List of discussion “Thêm phản hồi” button Menu option for each discussion 			
2	Manager clicks “Chỉnh sửa” from menu option for each discussion	System displays modal which include: <ul style="list-style-type: none"> “Nhập tiêu đề”: text field that contains that discussion title “Nội dung đề xuất”: Text area that contains that discussion description “Lưu” button 			
3	Manager clicks “Lưu” button	[Exception no.1] System displays “Sửa phản hồi thành công”			
Alternative Scenario: N/A					
Exceptions:					
No	Cause	System Response			
1	Manager does not input required field	System notices that: <ul style="list-style-type: none"> “Nhập tiêu đề”: System displays warning message: “Tiêu đề phải từ 5 tới 80 ký tự”. 			
Relationships:					
<ul style="list-style-type: none"> Abstract use case: Manage course feedback 					
Business Rules:					

- “Tiêu đề” must be from 5 to 80 characters.
- Discussion belongs to a specific course
- Manager can only edit his/her own discussion. Manager cannot edit another managers' discussion.

Table 57: Update feedback specification

2.3.4.16 <Manager> Remove feedback

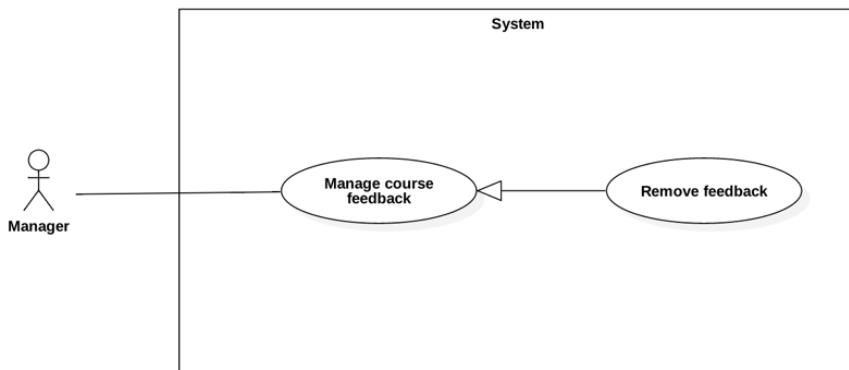


Figure 58: <Manager> Remove feedback (UC_TRAISY51)

USE CASE - UC_TRAISY51			
Use Case No.	UC_TRAISY51	Use Case Version	0.2
Use Case Name	Remove feedback		
Author	Võ Thạch Nguyên		
Date	26/3/2018	Priority	High
Actor:			
• Manager			
Summary:			
• This use case allows authorized manager to remove a specific feedback.			
Goal:			
• To remove a feedback of a course.			
Triggers:			
• Actor sends remove feedback command.			
Preconditions:			
• User logged in as manager role.			
Post conditions:			
• Success: Actor successfully remove a feedback.			
• Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Manager goes to “Đề xuất chỉnh sửa” page	System displays page which include: • List of discussion • “Thêm phản hồi” button • Menu option for each discussion	

2	Manager clicks “Xóa” from menu option for each discussion	System displays modal to confirm user's action
3	Manager clicks confirm button	System displays “Xóa phản hồi thành công”

Alternative Scenario: N/A

Exceptions: N/A

Relationships:

- Abstract use case: Manage course feedback

Business Rules:

- When discussion is deleted, the discussion is completely removed from database. Therefore, all information that related to discussion such as reply discussion will also be removed
- Manager can only delete his/her own discussion. Manager cannot delete another users' discussion

Table 58: Remove feedback specification

2.3.6 <Admin> Overview Use Case

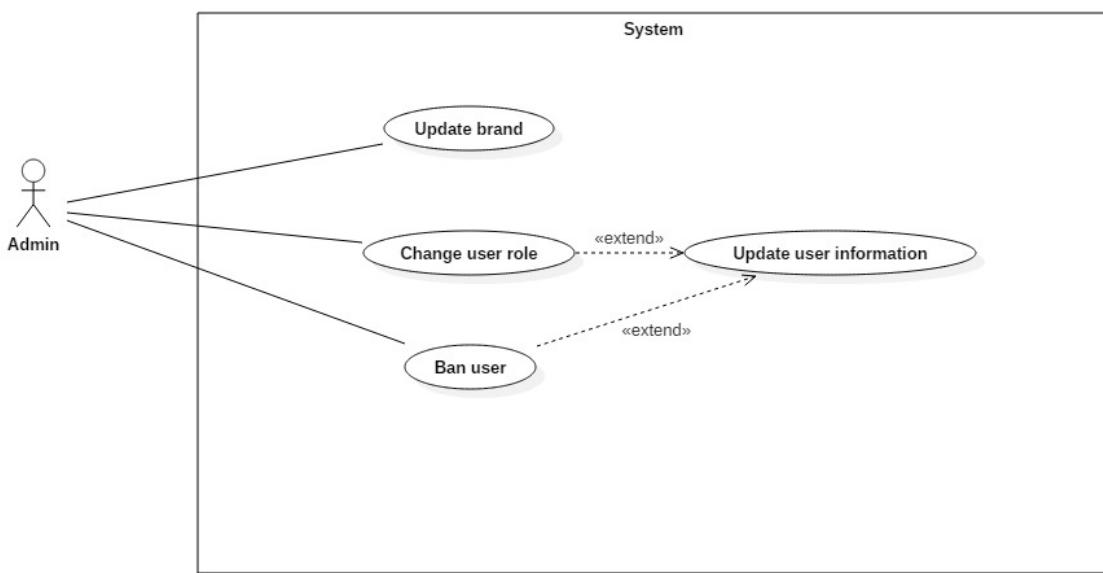


Figure 59: <Admin> Overview Use Case

2.3.6.1 <Admin> Update brand

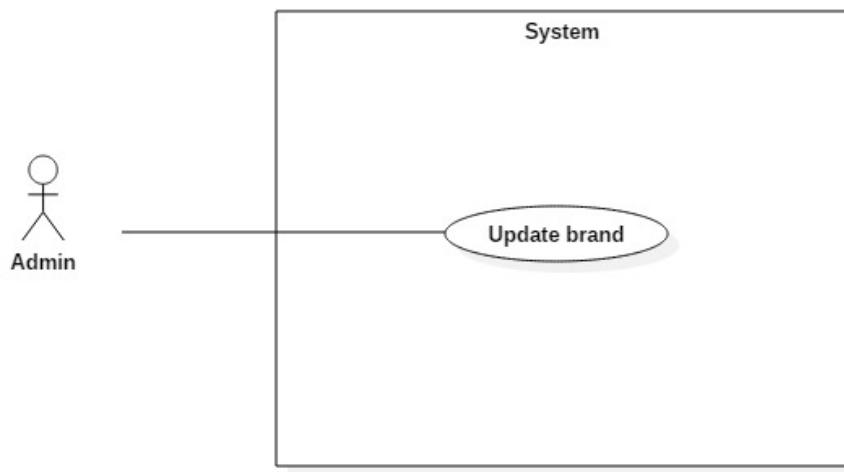


Figure 60: <Admin> Update brand (UC_TRAISY52)

USE CASE - UC_TRAISY52			
Use Case No.	UC_TRAISY52	Use Case Version	0.2
Use Case Name	Update brand		
Author	Phạm Chánh Hưng		
Date	26/3/2018	Priority	Normal
Actor:			
• Admin			
Summary:			
• This use case allows Admin to update his brand.			
Goal:			
• Update his brand's details.			
Triggers:			
• Actor sends update brand command.			
Preconditions:			
• User logged in as Admin role.			
Post conditions:			
• Success: Actor successfully updates a brand to the database.			
• Fail: System shows error messages.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Admin goes to “Quản lý thương hiệu” page	System redirects actor to page which include: • “Tên thương hiệu”: text field • “Mô tả”: text area • Picture of brand • “Lưu chỉnh sửa” (save) button	
2	Admin clicks “Lưu chỉnh sửa” button	[Exception no.1]	

		System shows “Cập nhật thành công” popup
Exceptions:		
No	Cause	System Response
1	Admin does not input required field	System notices that “Tên thương hiệu không được trống”
2	Admin's brand has been deleted by system admin	System redirects to error page.

Relationships: N/A

Business Rules:

- User already logged in the system as “Admin” role.
- Brand’s name must not be empty and between 2-80 characters.
- Admin can only update his brand. Admin can not update another admins’ brand.

Table 59: Update brand specification

3. Software System Attribute

3.1 Usability

3.1.1 Graphic User Interface

All the texts, labels, alerts and messages will be written in Vietnamese GUI for web application is designed base on material design with Navigation Bar on the left contains the main functions of the application.

3.1.2 Usability

The system usability is easy to use that users generally don’t need to look at the document to use including admin and manager.

Icons that indicate the actions should be easy to understand and users will not meet any troubles to recognize the feature of page.

For example:



For learners, each lecture has icon that show what type of lecture next to it, which helps learners understand the lecture even before studying it.

For example:

- ▶ Indicates video lecture
- 📄 Indicates text lecture

3.2 Reliability

The data should be backed up every day.

3.3 Availability

The web application must be available 24/7.

3.4 Security

- System admin has to create each brand for each company to use the system.
- Users has to be registered by their admin/manager in order to access to the application.
- Users can not register their own accounts

3.5 Maintainability

System is divided into modules.

3.6 Portability

The software itself is a web application, therefore it can be used in any platform that has a web browser and can connect to the internet.

3.7 Performance

Requests from users are responded in less than 10 seconds at 5 Mbps bandwidth speed.

4. Conceptual diagram

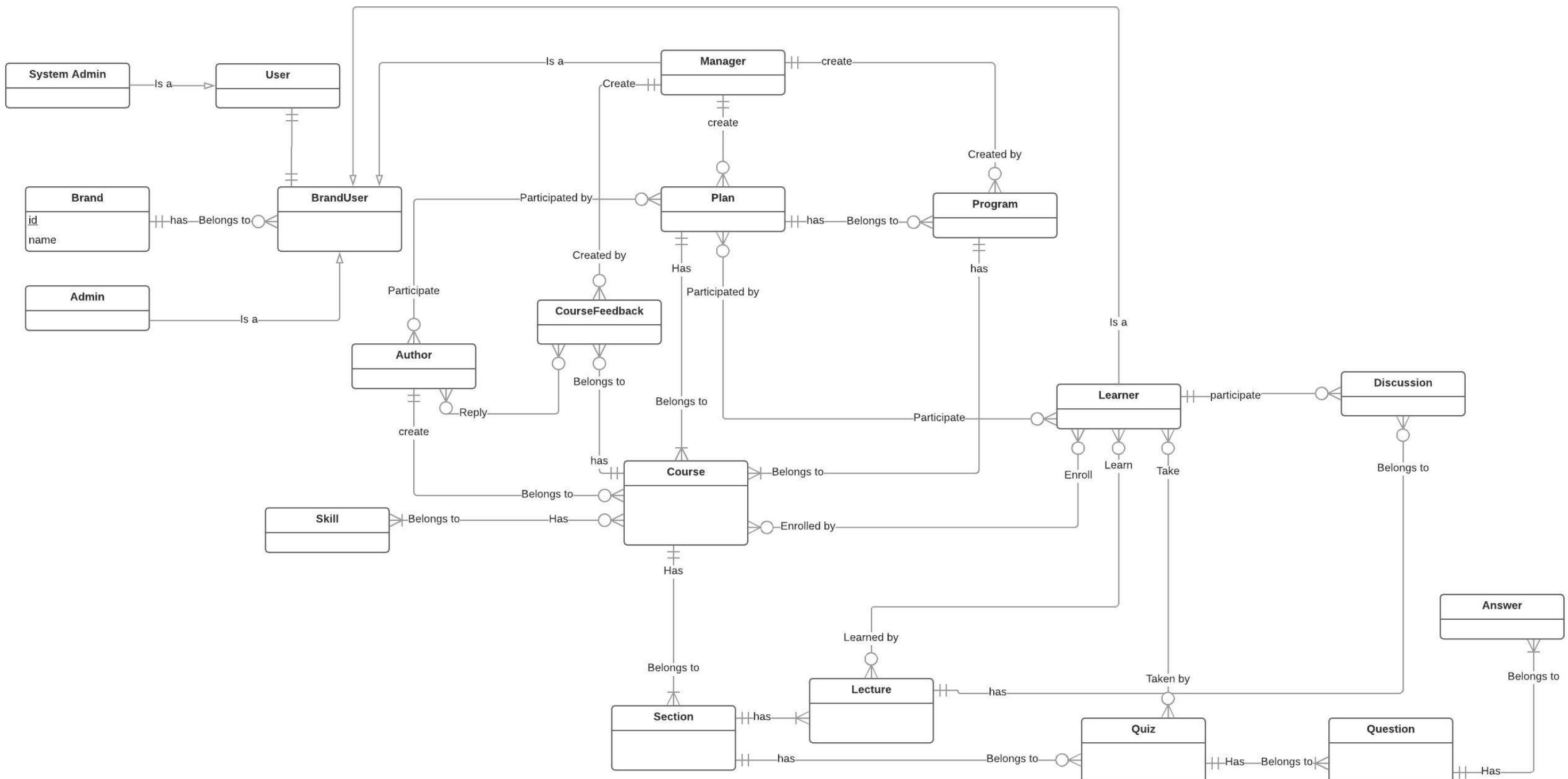


Figure 61: Conceptual diagram

ENTITY DATA DICTIONARY: DESCRIBE ALL CONTENT OF ALL ENTITIES	
Entity Name	Description
User	Abstract entity describes a user in system
System Admin	Contains the system admin information
Admin	Contains the admin user information
Manager	Contains the manager information
Author	Contains the author information
Learner	Contains the learner information
Brand	Contains the brand information
BrandUser	Contains the user information in a brand
AuthorPlan	Contains the information of author in a plan
LearnerPlan	Contains the information of a learner in a plan
Plan	Contains the plan information
Program	Contains the program information
Course	Contains the course information
CourseFeedback	Contains the information of feedback
Skill	Contains the information of skill
Section	Contains the section information
Lecture	Contains the lecture information
LearnerLecture	Contains the information of a learner about lecture
Quiz	Contains the quiz information
Question	Contains the information of question
Answer	Contains the information of an answer
LearnerQuiz	Contains the information of a learner in a quiz
Discussion	Contains the information of a discussion

Table 60: Conceptual diagram data dictionary

D. Report No. 4 Software Design Description

1. Design Overview

This document describes the technical and user interface design of TRAISY Web Application. It includes the architectural design, the detailed design of common functions and business functions and the design of database model.

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

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

The database design describes the relationships between entities and details of each entity.

Document overview:

Section 2: gives an overall description of the system architecture design.

Section 3: gives component diagrams that describe the connection and integration of the system.

Section 4: gives the detail design description which includes class diagram, class explanation, and sequence diagram to details the application functions.

Section 5: describe screens design.

Section 6: describe a fully attributed ERD.

Section 7: describe algorithms.

2. System Architectural Design

2.1 System Architecture Overview

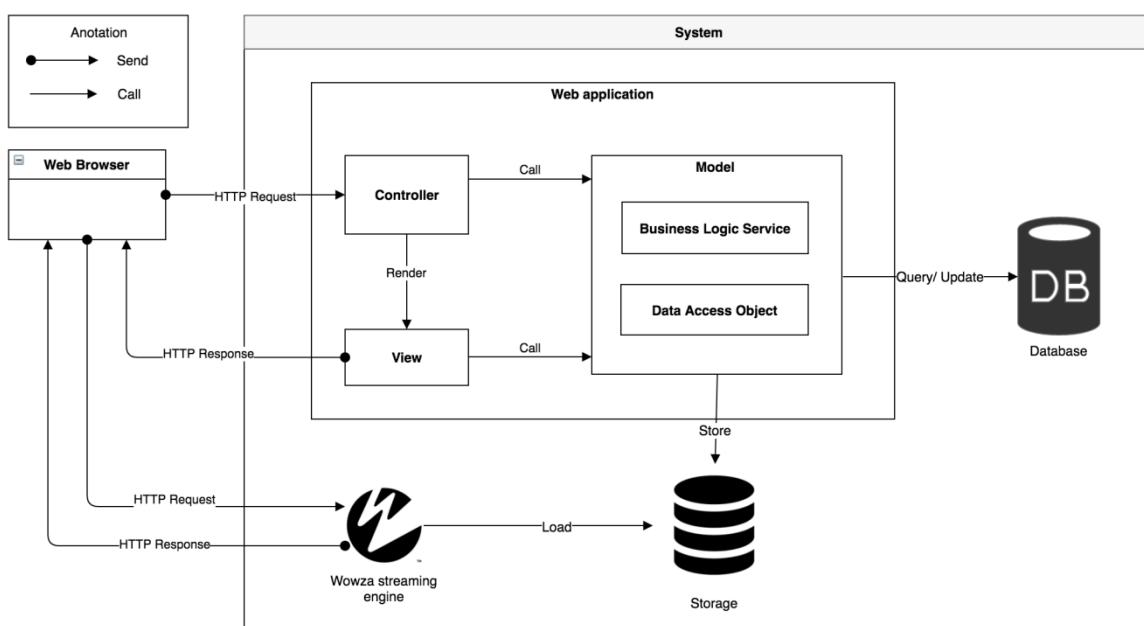


Figure 62: System architecture overview

Our application is developed mainly on Play Framework 2 which follows the MVC architectural pattern. We choose this architecture for web application because of following advantages:

- **Hot reload:** Make a change, refresh the page and we can see the change immediately. As a result, we don't waste time to redeploy the application.
- **Open source:** Play framework is open source; we can see how everything works in the open source code. There is also a large community using it, contributing plugins, asking questions on StackOverflow, and so on.
- **Reactive:** play is built on top of Akka and Netty, so it's fully asynchronous: that is, you can use non-blocking I/O when making calls to remote services. This makes it possible to support many concurrent connections without dealing with thread pools.

This pattern splits the application into separate layers: The model layer, view layer, and controller layer.

- The model layer: Contains data structures and operations that needed for system to operate. Whenever model objects need to be saved into database, they contain JPA annotation for making these operations easier and more flexible.
- The view layer: a HTML file that presents the content and interaction with user. In play framework, views are generated using a templating system. The Controller gets data from the model layer, and then applies a template to render these objects.
- The controller responds to user actions and processes them. For more details, the controller listens for HTTP requests, then extracts relevant data from the request, such as parameter, request headers... and invokes the suitable methods that contains read, update or create the model object. After finishing the process, controllers render the view and then written as an HTTP Response to the user.

Besides the MVC architectural, we also use Wowza streaming engine with HLS streaming protocol for an easier and more efficient way to deliver video content or video lecture.

HTTP Live streaming (HLS) is a streaming protocol that generates multiple versions of the same content (but different resolution/bit rate) and chops these versions into chunks or segments (e.g., two seconds). The segments are provided on a web server and can be downloaded through HTTP standard compliant GET requests. Through multiple versions of same content, HLS also provides “adaptive streaming”, the client can switch to a higher resolution – if bandwidth permits, or to a lower resolution – if bandwidth decreases.

3. Component Diagram

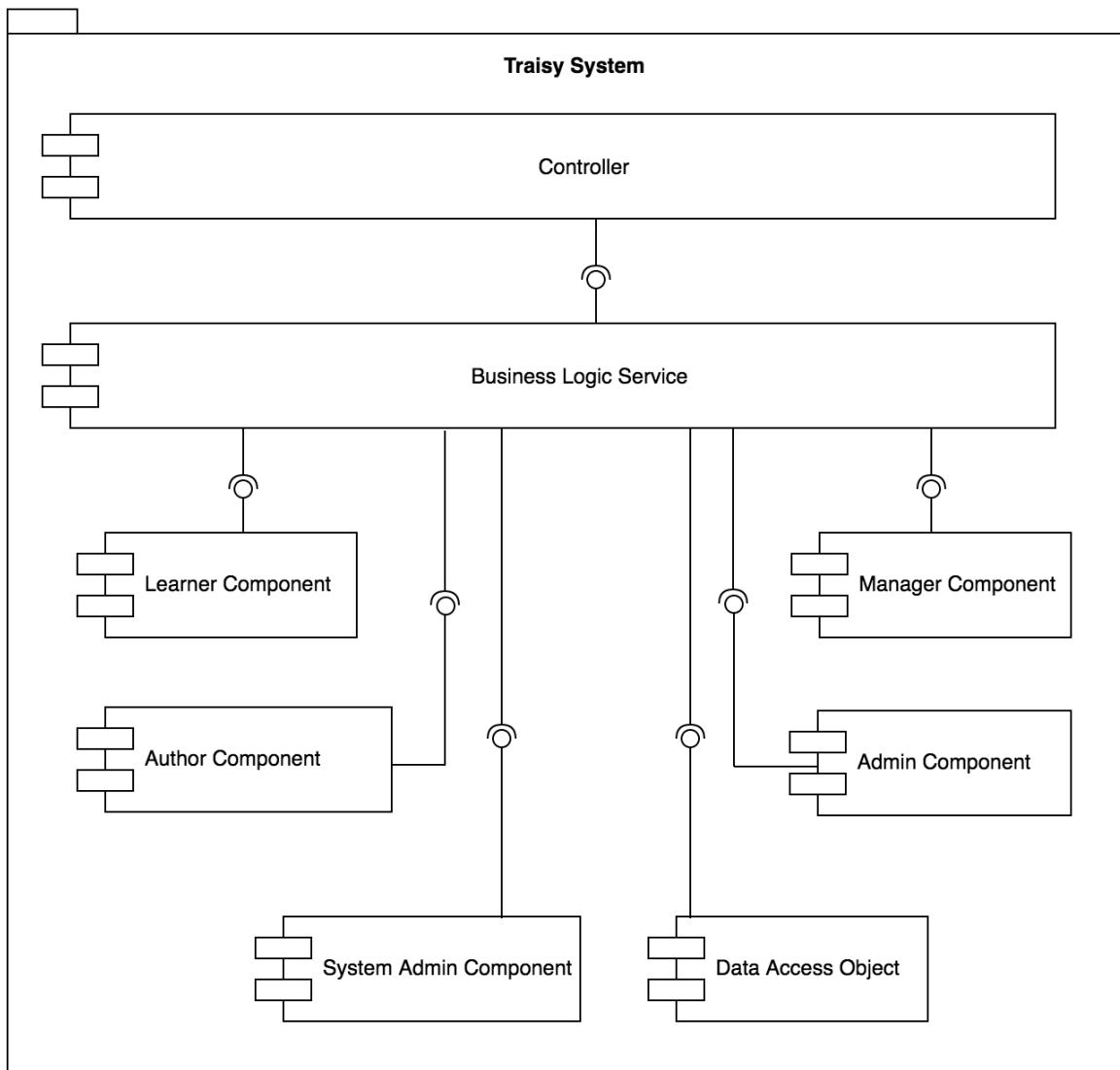


Figure 63: Component diagram

COMPONENT DICTIONARY: DESCRIBES COMPONENTS	
Traisy System	Mobile application package
Controller	Component to handle HTTP request
Business Logic Service	Component to handle system's business operations
Learner Component	Component to handle learner activities in the system
Author Component	Component to handle author activities in the system
Manager Component	Component to handle manager activities in the system
Admin Component	Component to handle admin activities in the system

System Admin Component	Component to handle system admin activities in the system
Data access object	Component to handle interaction between the system and database

Table 61: Component Dictionary

4. Detail Description

4.1 Class Diagram

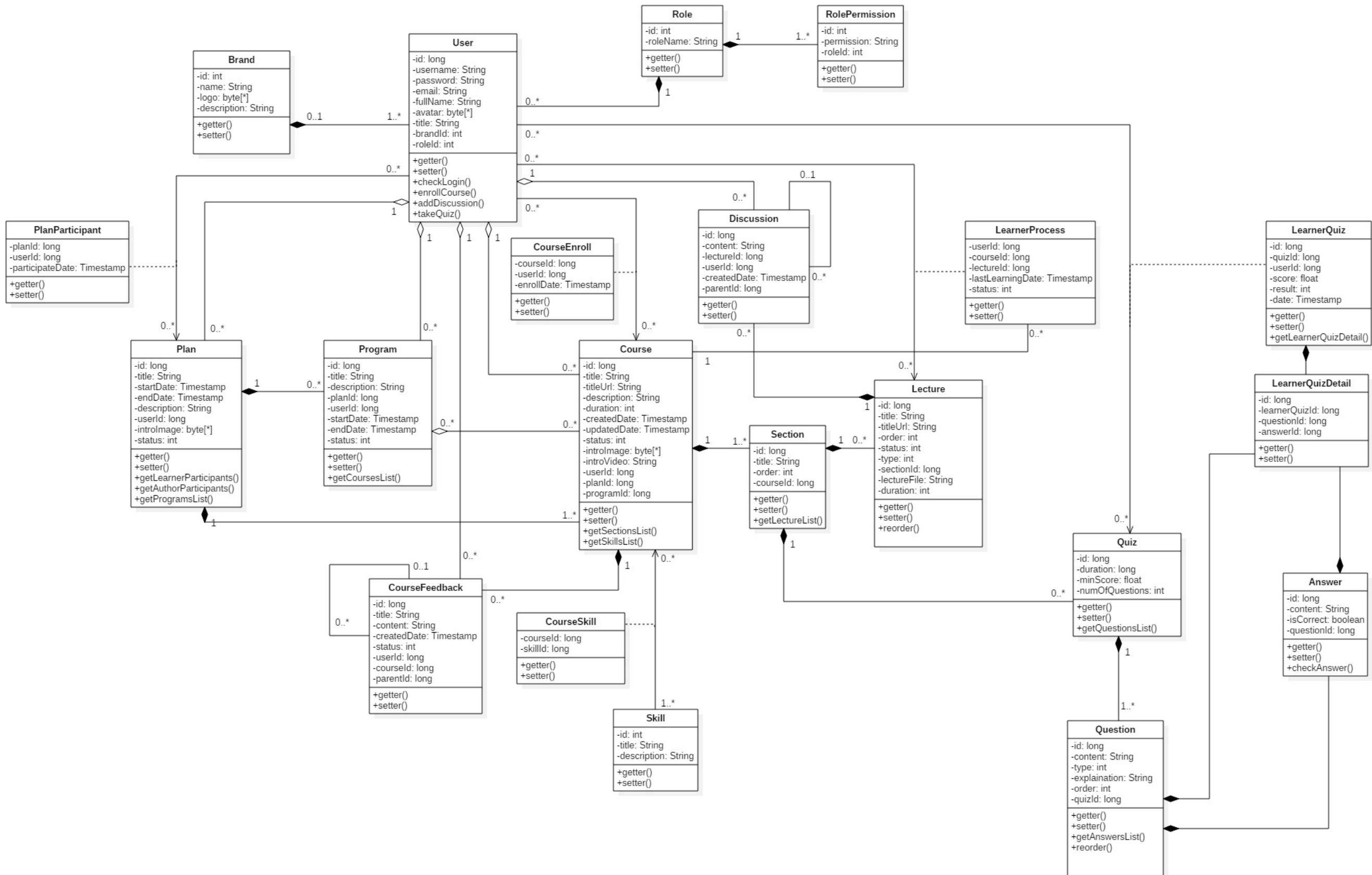


Figure 64: Class Diagram

CLASS DICTIONARY: DESCRIBE CLASS		
Class Name	Mapping column with Conceptual diagram	Description
User	user	Contains the user information
Brand	brand	Contains the brand information
PlanParticipant	N/A	Not exist in conceptual diagram but need this table to contain the information of a user participated in a plan
Plan	Plan	Contains the plan information
Program	Program	Contain the program information
CourseEnroll	N/A	Not exist in conceptual diagram but need this table to contain the information of a user enrolled in a plan
Course	Course	Contains the course information
Discussion	Discussion	Contains the discussion information
Section	Section	Contains the section information
Skill	Skill	Contains the skill information
CourseSkill	N/A	Not exist in the conceptual diagram but need this table to contain the information of a skill in a specific course
Lecture	Lecture	Contains the lecture information
LearnerProcess	N/A	Not exist in the conceptual diagram but need this table to contain the process information when learner study a lecture
Quiz	Quiz	Contains the quiz information
Question	Question	Contains the question information
LearnerQuiz	LearnerQuiz	Contains the information of learner when that learner take quiz
LearnerQuizDetail	N/A	Contains the result of learner's quiz
Answer	Answer	Contains the information of answer
Role	N/A	Not exist in conceptual diagram but need this table to contain the information of a role
RolePermission	N/A	Not exist in conceptual diagram but need this table to contain the permission of each role

Table 62: Class Dictionary

4.2 Class Diagram Explanation

4.2.1 User

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a user
username	String	Private	User's username
password	String	Private	User's password
fullName	String	Private	User's fullname
avatar	Byte[]	Private	User's avatar
brandId	Long	Private	Id of brand that the user belongs to
title	String	Private	User's title
roleId	Integer	Private	Id of user's role
status	Integer	Private	User's status
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 63:User attributes

4.2.2 Brand

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a brand
name	String	Private	Brand's name
logo	byte[]	Private	Brand's logo
description	String	Private	Brand's description
status	Integer	Private	Brand's status
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 64: Brand attributes

4.2.3 Plan

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a plan
title	String	Private	Plan's title
startDate	Timestamp	Private	Plan's start date
endDate	Timestamp	Private	Plan's end date
description	String	Private	Plan's description

userId	Long	Private	Id of user that create the plan
planImage	byte[]	Private	Plan's image
status	int	Private	Plan's status

Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 65: Plan attributes

4.2.4 PlanParticipant

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a plan participant
planId	Long	Private	Id of plan that the user participated in
userId	Long	Private	Id of user that participate in plan
participateDate	Timestamp	Private	User's participate date
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 66: PlanParticipant attributes

4.2.5 Program

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a program
title	String	Private	Program's title
description	String	Private	Program's description
planId	Long	Private	Id of plan that the program belongs to
startDate	Timestamp	Private	Program's start date
endDate	Timestamp	Private	Program's end date
status	int	Private	Program's status
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 67: Program attributes

4.2.6 Course

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a course
title	String	Private	Course's title
titleUrl	String	Private	Course's title url
description	String	Private	Course's description
duration	Integer	Private	Course's duration
createdDate	Timestamp	Private	Program's start date
updatedDate	Timestamp	Private	Program's end date
status	Integer	Private	Course's status
introImage	Byte[]	Private	Course's image
introVideo	String	Private	Course's video
userId	Long	Private	Id of user that create the course
planId	Long	Private	Id of plan that the course belongs to
programId	Long	Private	Id of program that the course belongs to
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 68: Course attributes

4.2.7 CourseEnroll

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a CourseEnroll
courseId	Long	Private	Id of course that the user enrolled to
userId	Long	Private	Id of user that enrolled in course
enrollDate	Timestamp	Private	User's enroll date
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 69: CourseEnroll attributes

4.2.7 Skill

Attribute	Type	Visibility	Description
-----------	------	------------	-------------

id	Integer	Private	Unique identifier of a skill
title	String	Private	Skill's title
description	String	Private	Skill's description
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 70: Skill attributes

4.2.8 CourseSkill

Attribute	Type	Visibility	Description
courseId	Long	Private	Id of course that the skill belongs to
skillId	Integer	Private	Id of skill that belongs to course
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 71: CourseSkill attributes

4.2.8 CourseSection

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a CourseSection
title	String	Private	CourseSection's title
order	Integer	Private	Order of section
courseId	Long	Private	Id of course that the section belongs to
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 72: CourseSection attributes

4.2.9 CourseDiscussion

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a discussion
content	String	Private	Discussion's content
status	Integer	Private	Discussion's status

userId	Long	Private	Id of user that create discussion
createdDate	Timestamp	Private	Discussion's created date
parentId	Long	Private	Discussion's parent
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 73: CourseDiscussion attributes

4.2.10 CourseLecture

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a lecture
title	String	Private	Lecture's title
titleUrl	String	Private	Lecture's title url
order	Integer	Private	Lecture's order
status	Integer	Private	Lecture's status
type	Integer	Private	Lecture's type
sectionId	Long	Private	Id of section that the lecture belongs to
lectureFile	String	Private	Lecture's file
duration	Integer	Private	Lecture's duration
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 74: CourseLecture discussion

4.2.11 LearnerProcess

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a learner process
userId	Long	Private	Id of user that study the lecture
courseId	Long	Private	Id of course that the learner process belongs to
lectureId	Long	Private	Id of lecture that user study
lastLearningDate	Timestamp	Private	Last learning date of user
status	Integer	Private	Process's status

Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 75: LearnerProcess attributes

4.2.12 Quiz

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a quiz
duration	Long	Private	Quiz's duration
minScore	Float	Private	Quiz's minimum score
numberOfQuestions	Integer	Private	Quiz's number of questions
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 76: Quiz attributes

4.2.13 Question

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a question
content	String	Private	Question's content
type	Integer	Private	Question's type
explanation	String	Private	Question's explanation
order	Integer	Private	Order of question
quizId	Long	Private	Id of quiz that the question belongs to
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 77: Question attributes

4.2.14 Answer

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of an answer
content	String	Private	Answer's content
isCorrect	Integer	Private	Answer's is correct or not
questionId	Long	Private	Id of question the answer belongs to
Method	Return Type	Visibility	Description

getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 78: Answer attributes

4.2.15 LearnerQuizDetail

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a learner quiz detail
learnerQuizId	Long	Private	Id of learner quiz
questionId	Long	Private	Id of question
answerId	Long	Private	Id of answer
answerContent	String	Private	Content of answer
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 79: LearnerQuiDetail attributes

4.2.16 LearnerQuiz

Attribute	Type	Visibility	Description
id	Long	Private	Unique identifier of a learner quiz
quizId	Long	Private	Id of quiz
userId	Long	Private	If of user
score	Float	Private	LearnerQuiz's score
result	Integer	Private	LearnerQuiz's result
date	Timestamp	Private	LearnerQuiz's date
Method	Return type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 80: LearnerQuiz attributes

4.2.17 Role

Attribute	Type	Visibility	Description
id	int	Private	Unique identifier of a role
roleName	String	Private	Name of a role
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 81: Role attributes

4.2.18 RolePermission

Attribute	Type	Visibility	Description
-----------	------	------------	-------------

id	int	Private	Unique identifier of a role
permission	String	Private	Name of a role
roleId	int	Private	Id of role
Method	Return Type	Visibility	Description
getter	Attribute type	Public	Get value of attribute
setter	Void	Public	Set value for attribute

Table 82: RolePermission attributes

4.3 Interactive Diagram

4.3.1 Web Application

4.3.1.1 System Admin

4.3.1.1.1 Create new brand

Summary: this diagram shows process of creating a brand

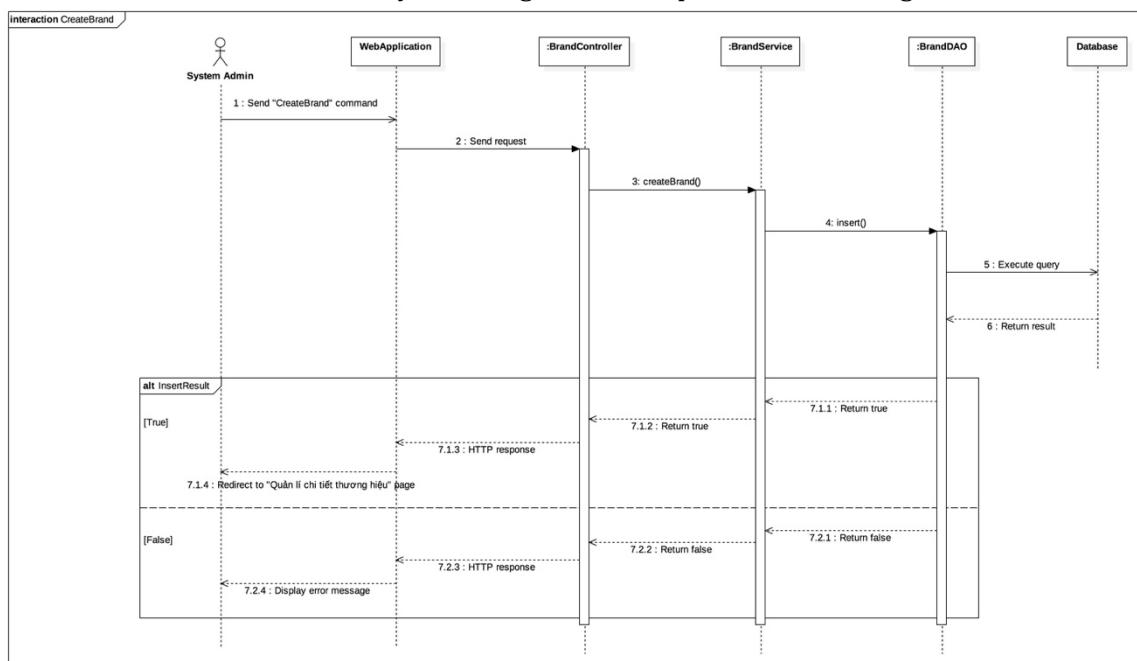


Figure 65: Sequence Diagram - <SystemAdmin> Create brand

4.3.1.1.2 Remove brand

Summary: this diagram shows process of deleting a brand

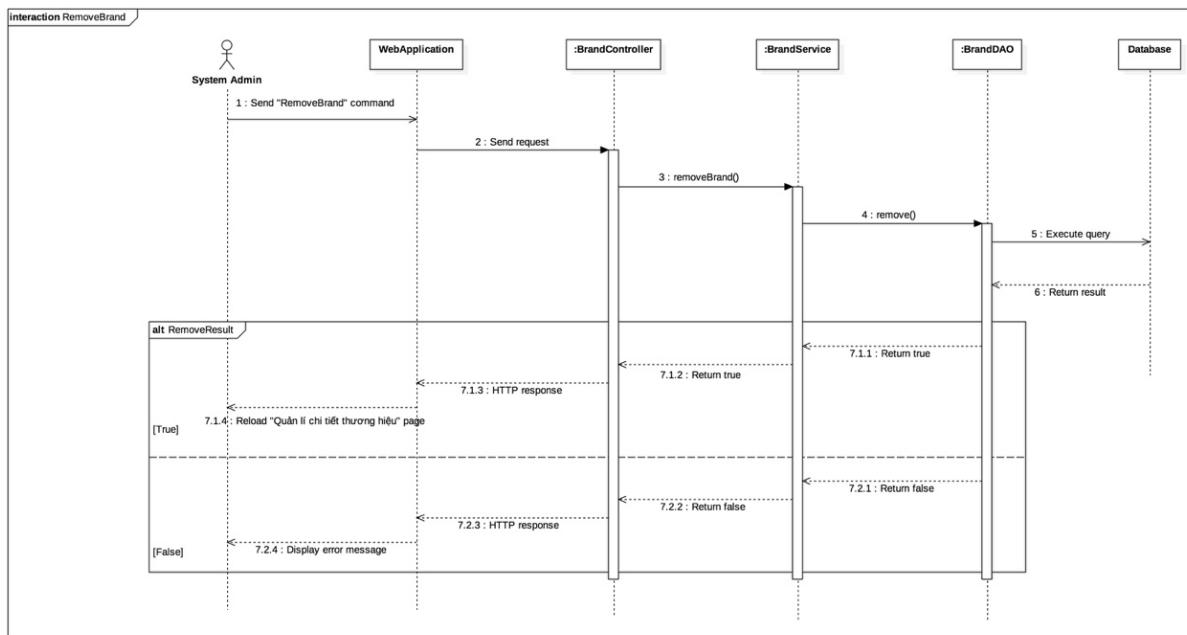


Figure 66: Sequence Diagram - <SystemAdmin> Remove brand

4.3.1.2 Admin

4.3.1.2.1 Update brand

Summary: this diagram shows process of updating admin's brand information

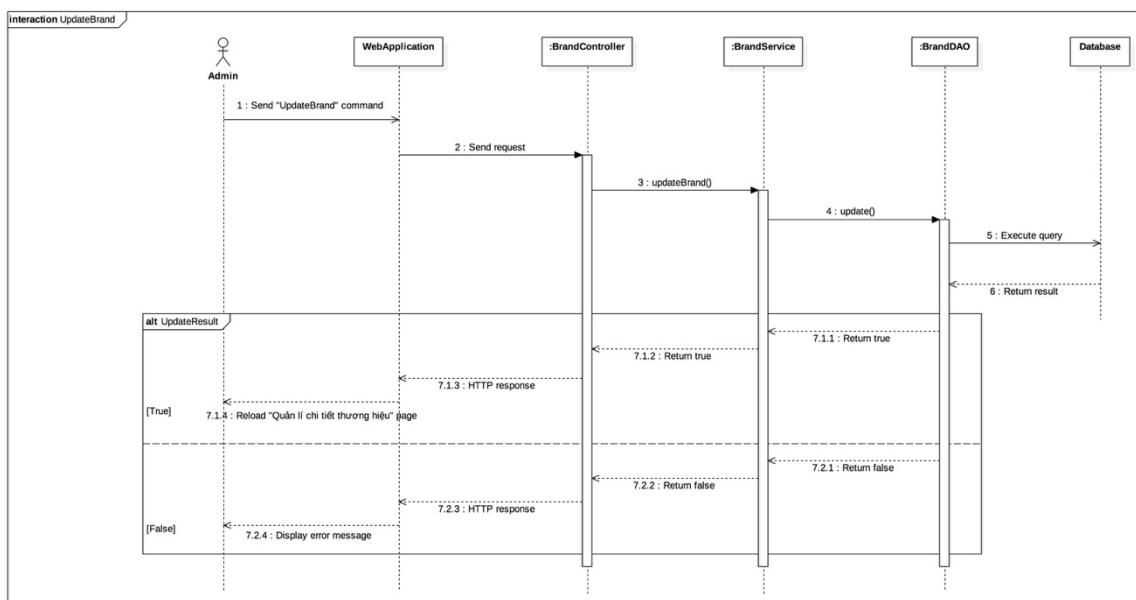


Figure 67: Sequence Diagram - <Admin> Update brand

4.3.1.3 Manager

4.3.1.3.1 Update user information

Summary: this diagram shows process of updating a user's information in brand

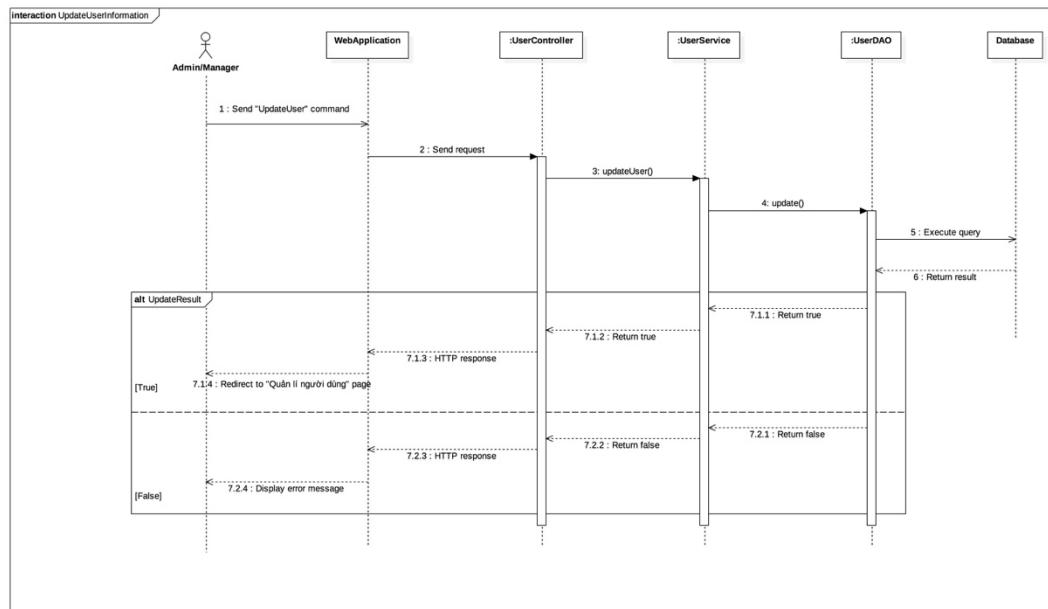


Figure 68: Sequence Diagram - <Manager> Update user information

4.3.1.3.2 Add learner to plan

Summary: this diagram shows process of adding a learner to plan

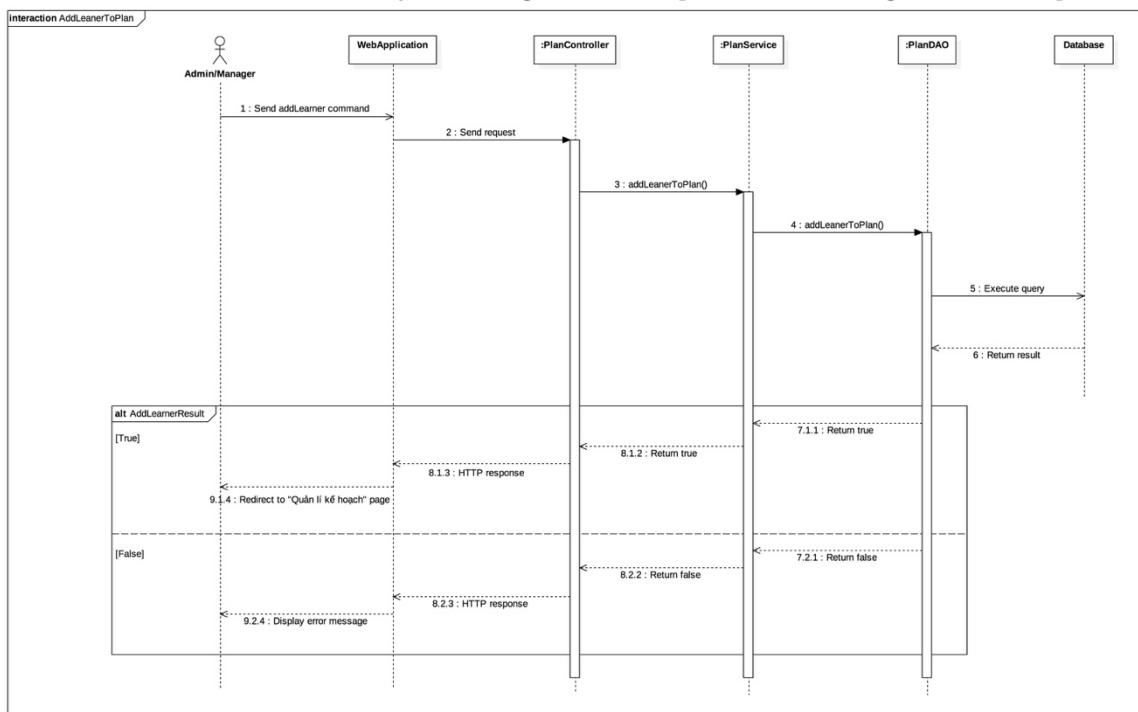


Figure 69: Sequence Diagram - <Manager> Add learner to plan

4.3.1.3.3 Create new user

Summary: this diagram shows process of creating user

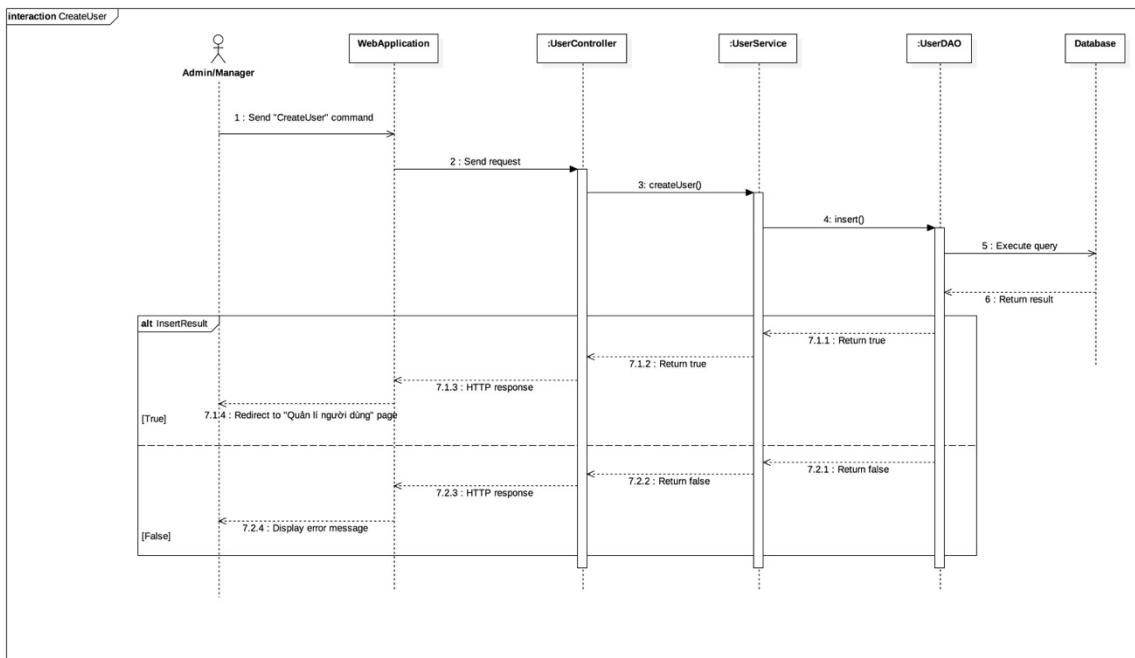


Figure 70: Sequence Diagram - <Manager> Create new user

4.3.1.3.4 Create plan

Summary: this diagram shows process of creating a plan

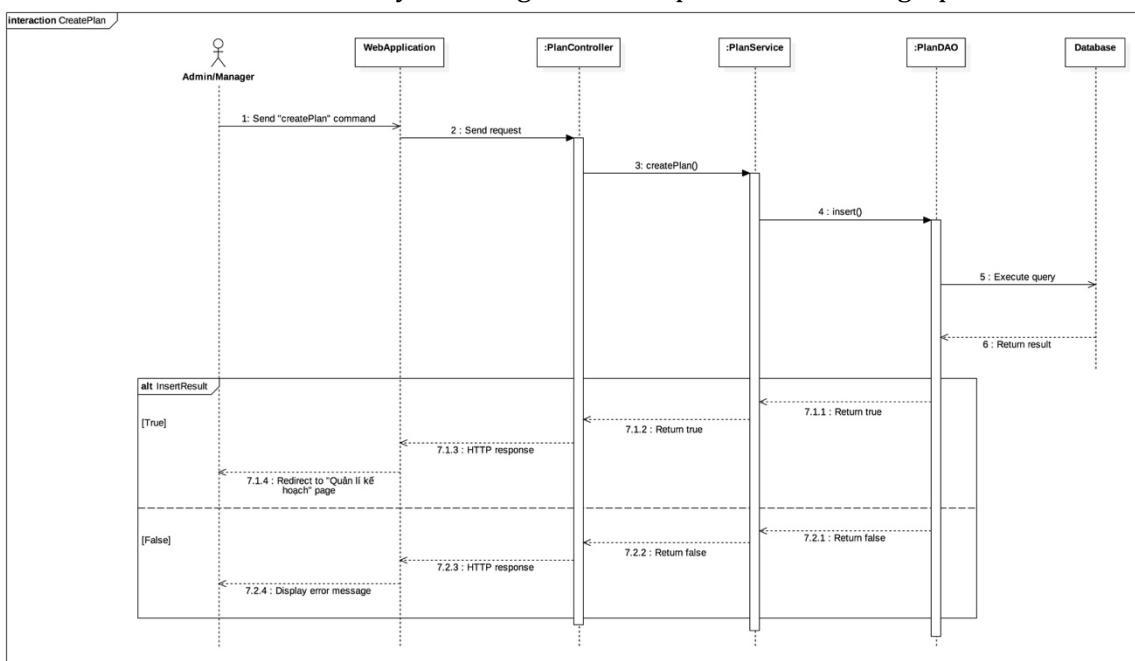


Figure 71: Sequence Diagram - <Manager> Create plan

4.3.1.3.5 Update plan

Summary: this diagram shows process of updating a plan

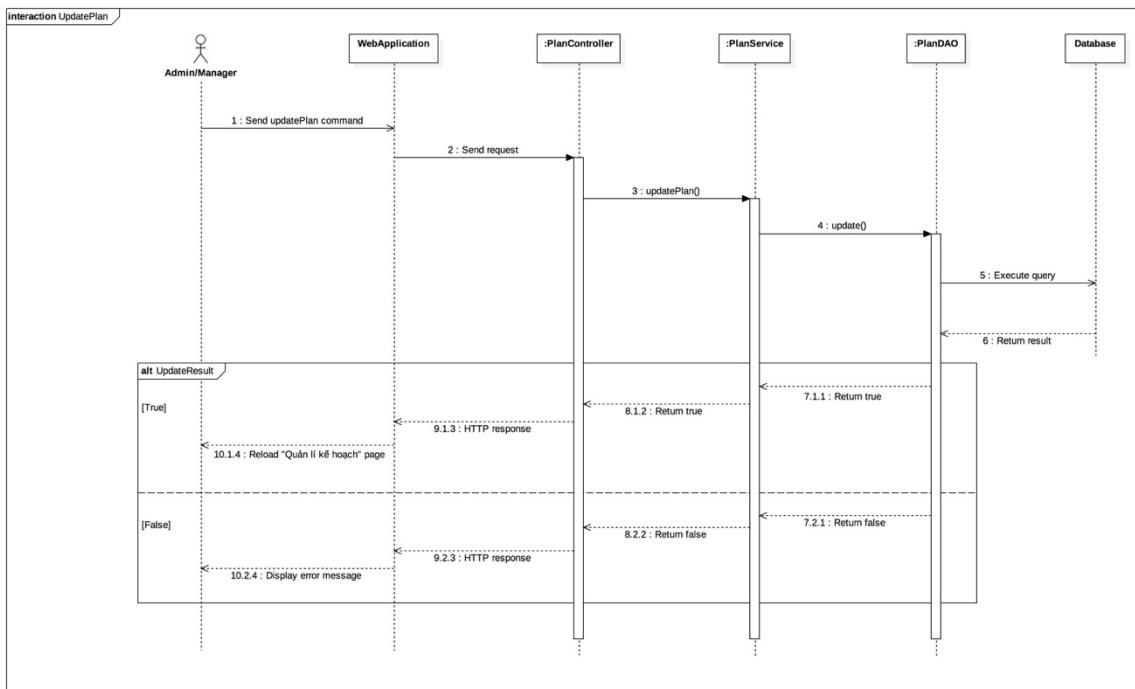


Figure 72: Sequence Diagram - <Manager> Update plan

4.3.1.3.6 Remove plan

Summary: this diagram shows process of removing a plan

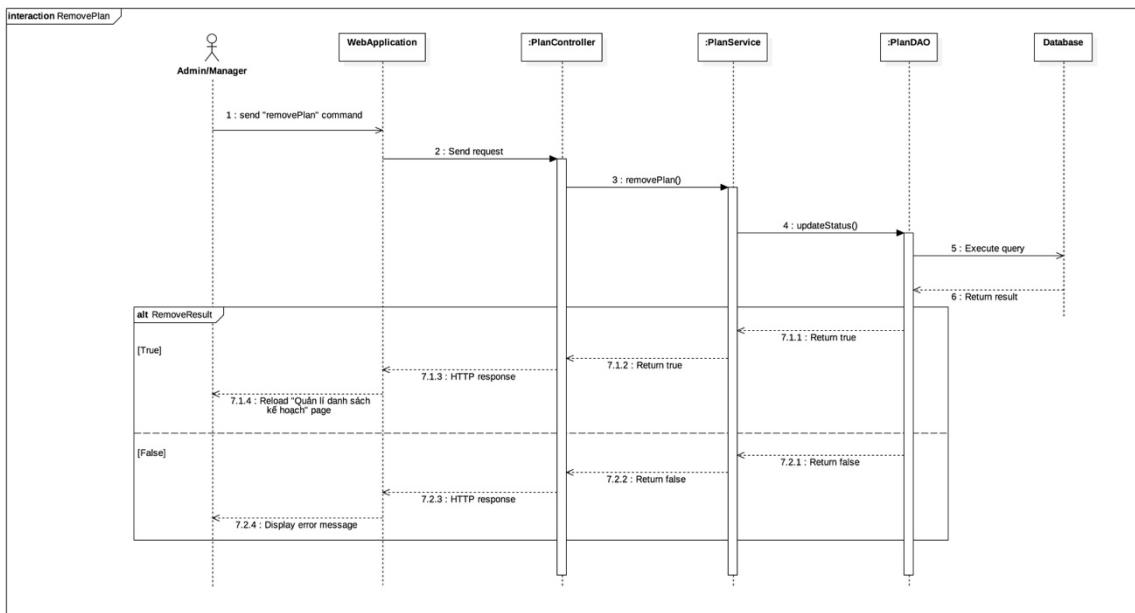


Figure 73: Sequence Diagram - <Manager> Remove plan

4.3.1.3.7 Add author to plan

Summary: this diagram shows process of adding an author to plan

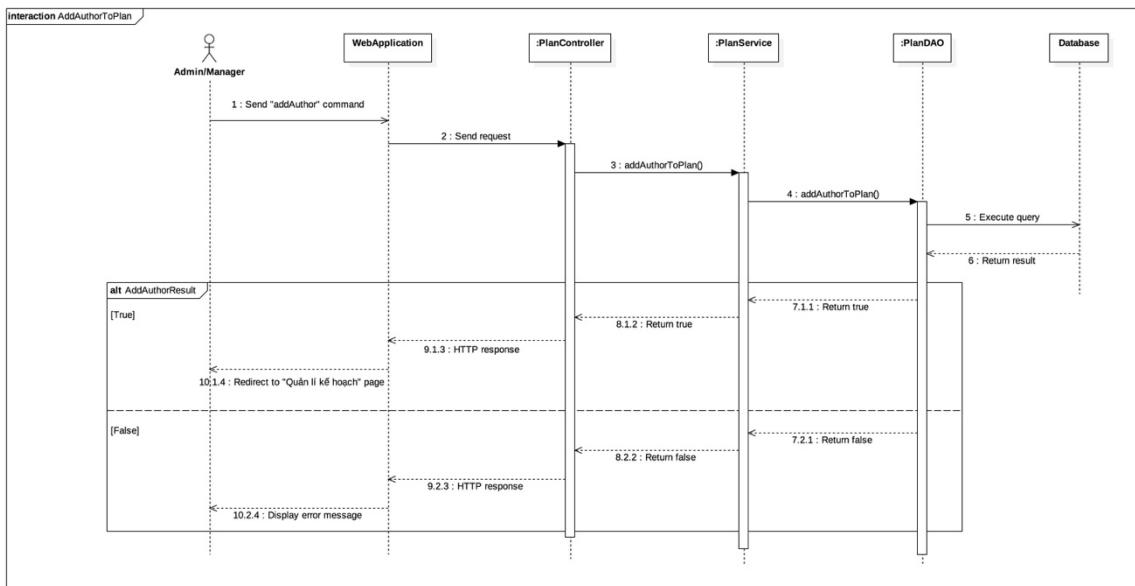


Figure 74: Sequence Diagram - <Manager> Add author to plan

4.3.1.3.8 Create program

Summary: this diagram shows process of creating a program

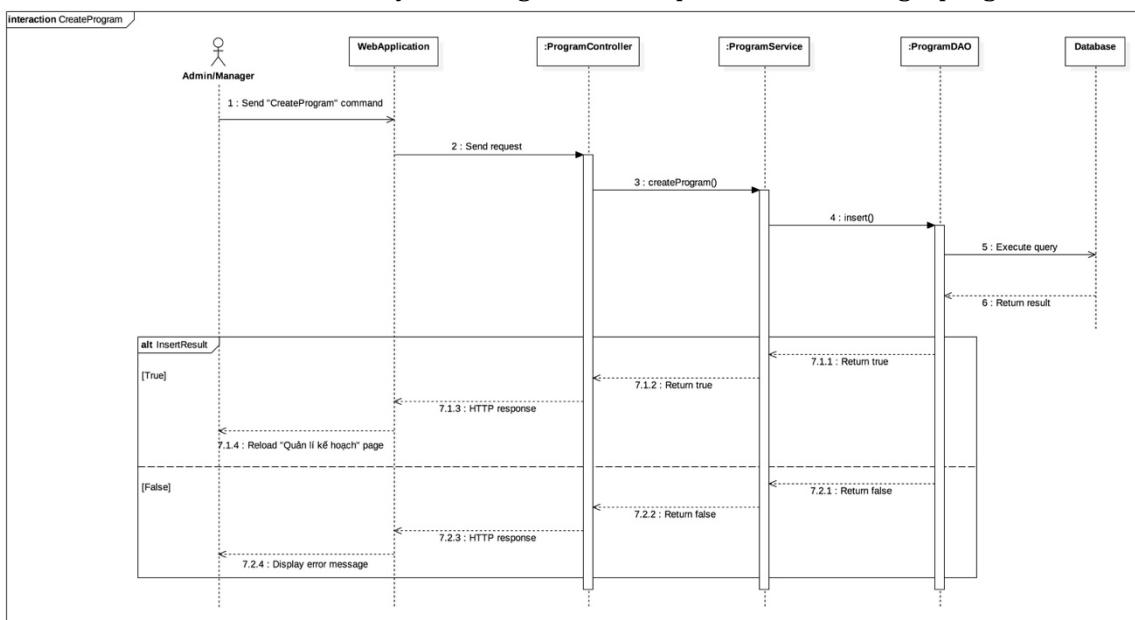


Figure 75: Sequence Diagram - <Manager> Create program

4.3.1.3.9 Update program

Summary: this diagram shows process of updating a program

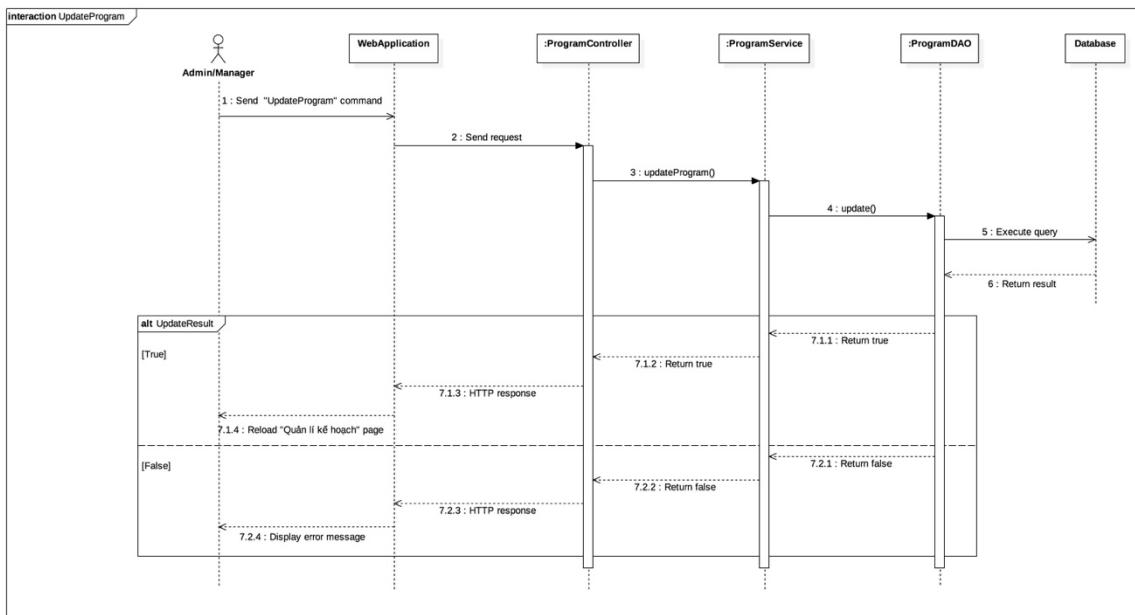


Figure 76: Sequence Diagram - <Manager> Update program

4.3.1.3.10 Remove program

Summary: this diagram shows process of removing a program

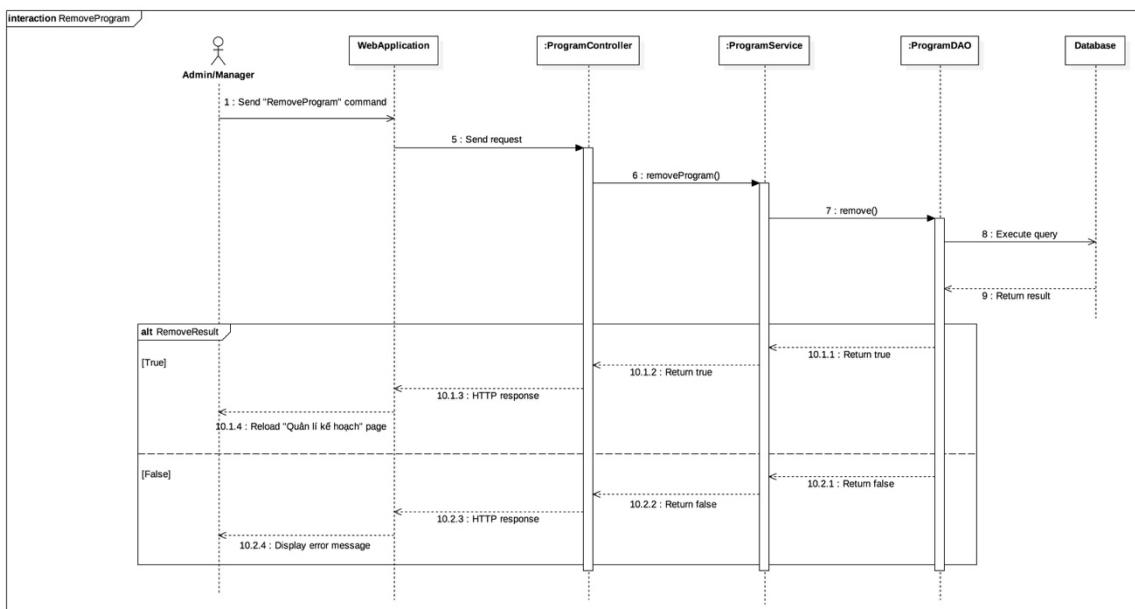


Figure 77: Sequence Diagram - <Manager> Remove program

4.3.1.3.11 View report

Summary: this diagram shows process of getting all reports of each plan in manager's brand

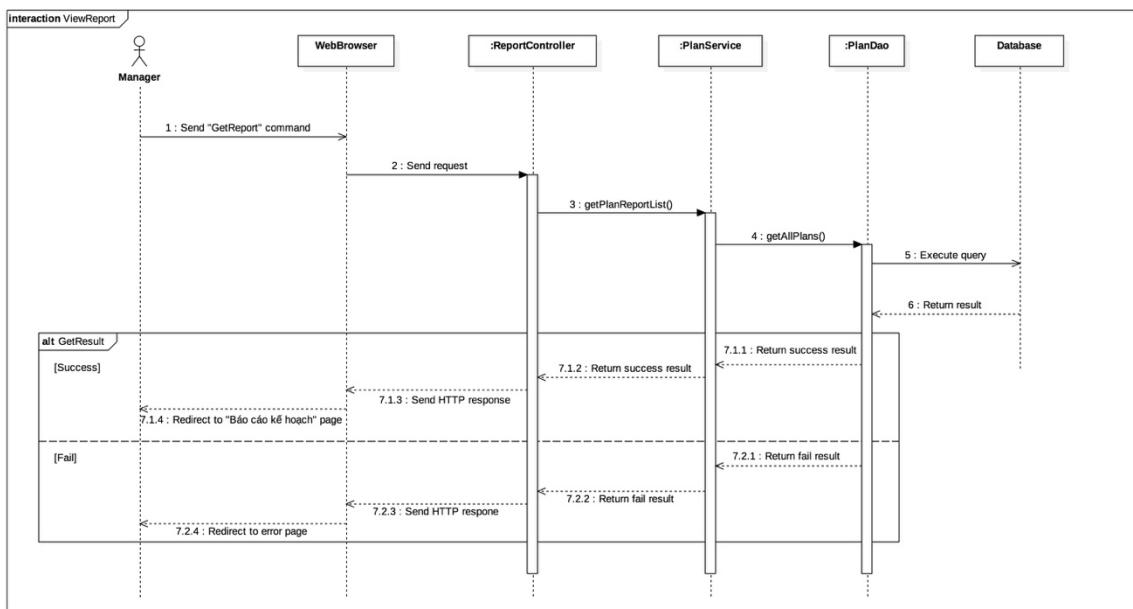


Figure 78: Sequence Diagram - <Manager> View report

4.3.1.4 Author

4.3.1.4.1 Create lecture

Summary: this diagram shows process of creating a lecture

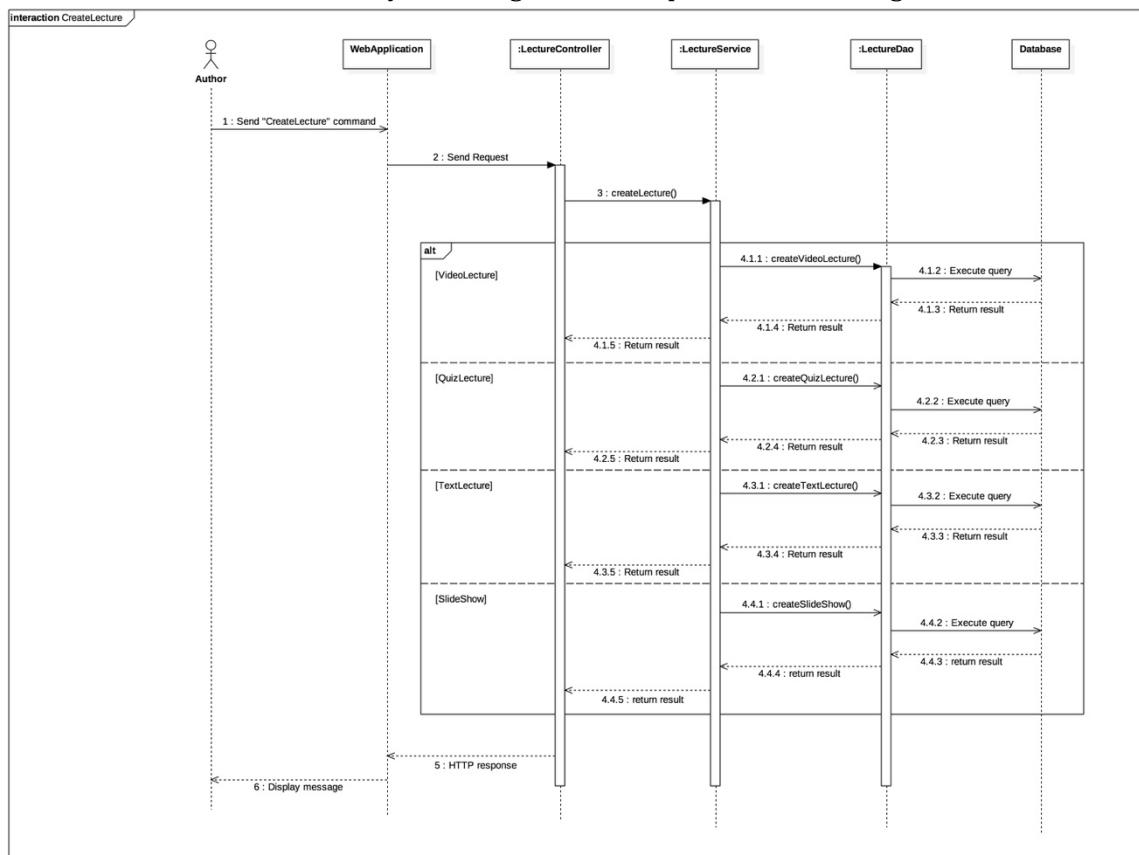


Figure 79: Sequence Diagram - <Author> Create lecture

4.3.1.4.2 Update lecture

Summary: this diagram shows process of updating a lecture

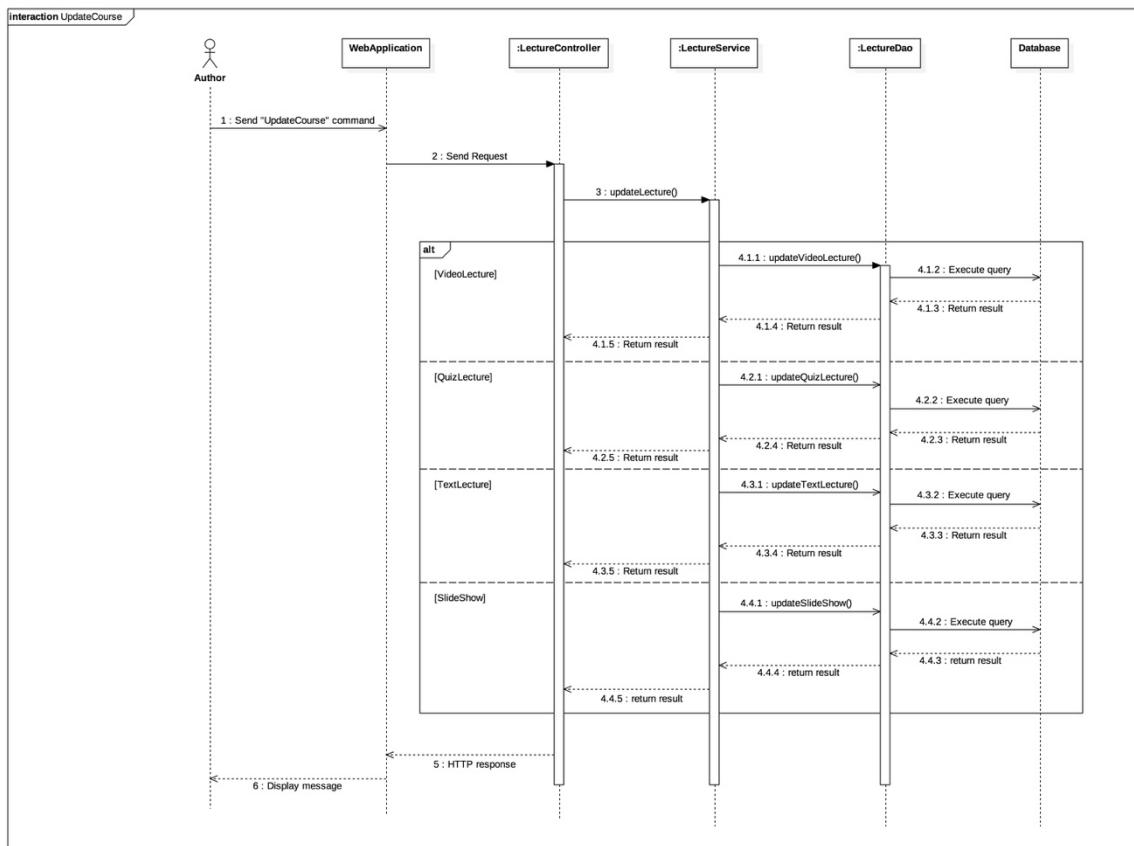


Figure 80: Sequence Diagram - <Author> Update lecture

4.3.1.4.3 Remove lecture

Summary: this diagram shows process of removing a lecture

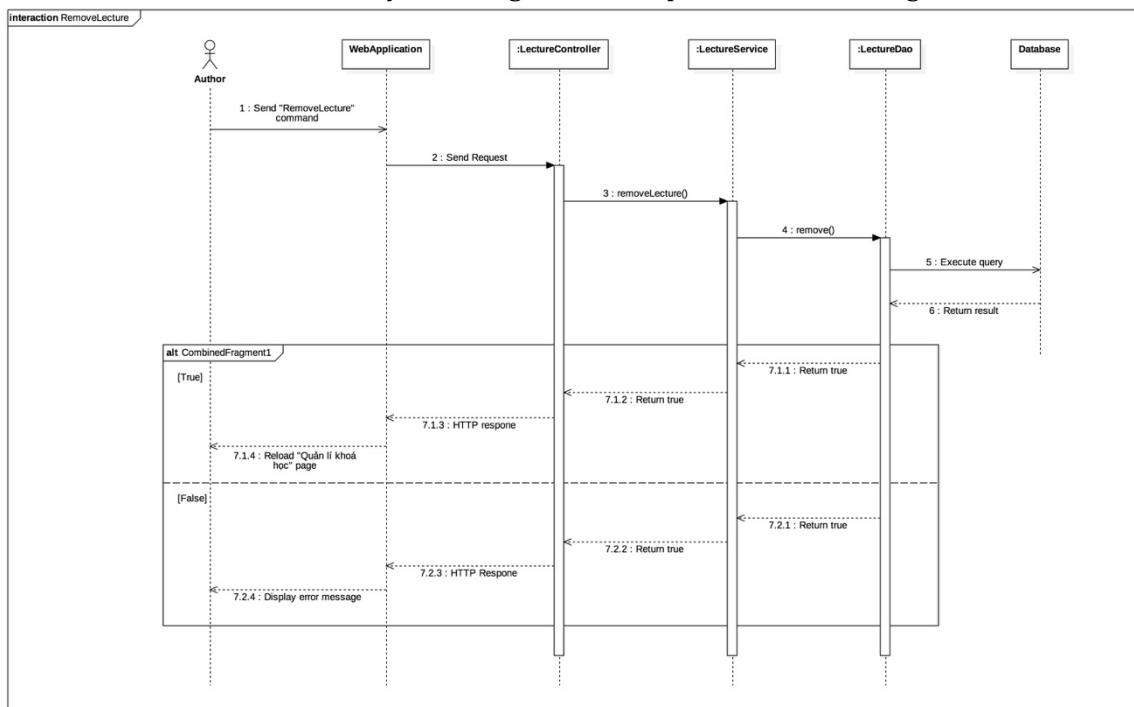


Figure 81: Sequence Diagram - <Author> Remove lecture

4.3.1.5 Learner

4.3.1.5.1 Login

Summary: this diagram shows process of logging into the system

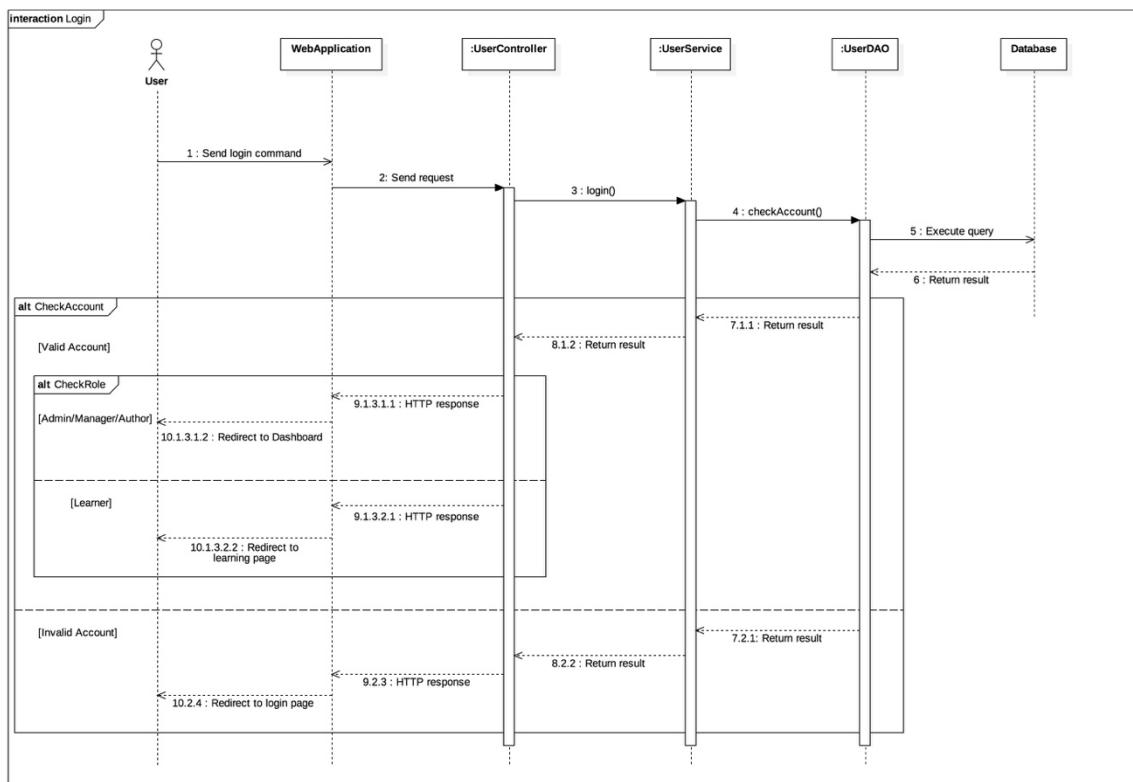


Figure 82: Sequence Diagram - <Learner> Log in

4.3.1.5.2 Learn lecture

Summary: this diagram shows process of learning a lecture

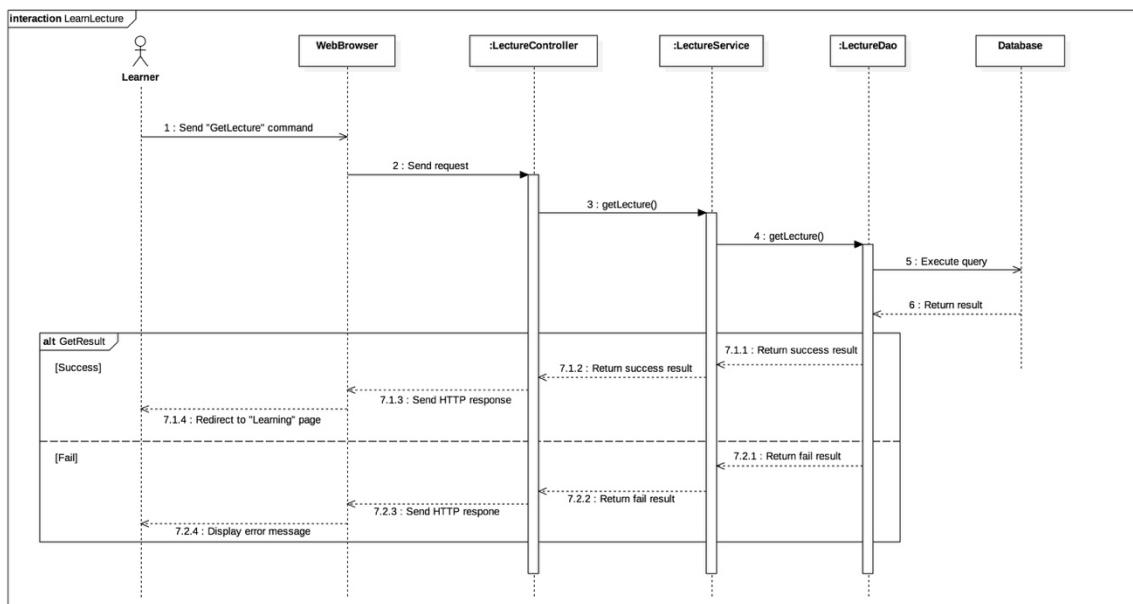


Figure 83: Sequence Diagram - <Learner> Learn lecture

5. User Interface Design

5.1 Web Application

5.1.1 Learner Interface Design

5.1.1.1 Login

Figure 84: Login Page

FIELDS								
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length	
1	txtUsername	Fill username	No	Yes	Textbox	String		
2	txtPassword	Fill password	No	Yes	Textbox	String		
BUTTONS/HYPERLINKS								
No	Function	Description		Validation	Outcome			
3	btnLogin	Login to system		N/A	N/A			

Table 83: Login page – Field, Button, Hyperlink

5.1.1.2 View list plans

Danh sách kế hoạch đang tham gia

Phạm Chánh Hưng
Project Manager

1 Kế hoạch hiện tại

2 Thông tin tài khoản

3 Đăng xuất

4 **Đào tạo Front-End Developer**
Từ 14/03/2018 đến 31/05/2018
Front-end và back-end là các thuật ngữ chỉ các giai đoạn bắt đầu và kết thúc của một quá trình xử lý. Khái niệm này thường sử dụng trong lĩnh vực phát ...

5 **Kế hoạch hết hạn**
Training Fresher Developer
Từ 07/03/2018 đến 31/03/2018
Training full stack developer

Figure 85: List participated plans page

BUTTONS/HYPERLINKS				
No	Function	Description	Validation	Outcome
1	viewListPlan	View list plans	N/A	N/A
2	manageProfile	Manage user profile	N/A	N/A
3	logout	Logout the system	N/A	N/A
4	viewPlanDetail	View detail information of specific plan	N/A	N/A
5	outdatePlan	Plan is outdated	N/A	N/A

Table 84: View list plan – Field, Button, Hyperlink

5.1.1.3 View plan overview

The screenshot displays a user interface for a learning management system. On the left, there's a sidebar with navigation links: 'Kế hoạch hiện tại' (1), 'Thông tin tài khoản' (2), and 'Đăng xuất' (3). The main content area features a course card for 'Đào tạo Front-End Developer' (1). The card includes a thumbnail image of a modern house (4), a start date ('Từ 14/03/2018 đến 31/05/2018') (2), a brief description ('Front-end và back-end là các thuật ngữ chỉ các giai đoạn bắt đầu và kết thúc của một quá trình xử lý. Khái niệm này thường sử dụng trong lĩnh vực phát triển phần mềm') (3), and a progress bar indicating 100% completion (9). Below the main card, there are two smaller cards: 'Làm quen với HTML5 và CSS3' (5) and 'Java Căn Bản' (6). The Java card shows a progress bar at 100% completion (9), a teacher name ('Giảng viên: Võ Thạch Nguyên') (8), and a 'Học tiếp' button (10).

Figure 86: Plan's overview page

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	title	Title of plan	Yes	Yes	Text	N/A	N/A
2	planTime	Plan's start date and end date	Yes	Yes	Text	N/A	N/A
3	planDescription	Plan's description	Yes	Yes	Text	N/A	N/A
4	planImage	Plan's picture	Yes	Yes	Image	N/A	N/A
5	program	Program belongs to plan	Yes	Yes	Text	N/A	N/A
6	coursePicture	Course's picture	Yes	Yes	Image	N/A	N/A
7	courseName	Course's name	Yes	Yes	Text	N/A	N/A
8	courseAuthor	Course's author	Yes	Yes	Text	N/A	N/A
9	courseProcess	Learning progress of learner	Yes	Yes	Text	N/A	N/A

BUTTONS/HYPERLINKS				
No	Function	Description	Validation	Outcome
10	courseAction	Coutinue learning or participate	N/A	N/A

Table 85: Plan's overview page – Field, Button, Hyperlink

5.1.1.4 Update profile

The screenshot shows a user profile update interface. On the left, there is a sidebar with a user picture, the name 'Phạm Chánh Hưng', and the title 'Project Manager'. Below this are three links: 'Kế hoạch hiện tại', 'Thông tin tài khoản', and 'Đăng xuất'. The main area contains a form with the following fields:

- 1: User picture (displayed)
- 2: Họ tên (Full name) field containing 'Phạm Chánh Hưng'
- 3: Chức danh (Title) field containing 'Project Manager'
- 4: Mật khẩu hiện tại (Current password) field
- 5: Mật khẩu mới (New password) field
- 6: Xác nhận lại mật khẩu (Confirm password) field
- 7: Lưu thay đổi (Save changes) button

Figure 87: User's information page

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	userPicture	User's avatar	No	No	TextBox	N/A	N/A
2	txtFullscreen	User's fullname	No	No	TextBox	N/A	N/A
3	txtTitle	User's title	No	No	TextBox	N/A	N/A
4	txtOldPassword	Old password	No	No	TextBox	N/A	N/A
5	txtNewPassword	New password	No	No	TextBox	N/A	N/A
6	txtConfirmPassword	Confirm password again to change	No	No	TextBox	N/A	N/A

BUTTONS/HYPERLINKS				
No	Function	Description	Validation	Outcome
7	btnSubmit	Save updated information	N/A	Save the new information

Table 86: User's information page – Field, Button, Hyperlink

5.1.1.6 Learn quiz lecture

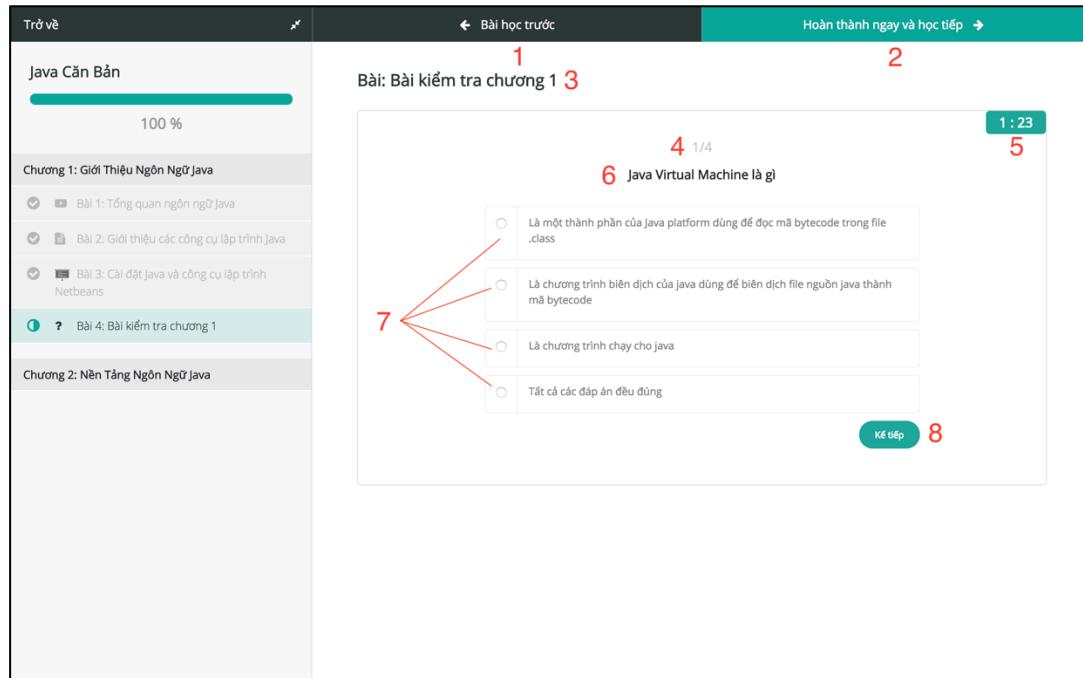


Figure 88: Take quiz page

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	numberOfQuestions	Number of questions	Yes	Yes	Label	N/A	N/A
2	questionContent	Question's content	Yes	Yes	Label	N/A	N/A
3	quizTime	Time for taking quiz	Yes	Yes	Label	N/A	N/A

BUTTONS/HYPERLINKS				
No	Function	Description	Validation	Outcome
1	btnPreviousLecture	Back to previous lecture	N/A	Previous lecture will be displayed
2	btnNextLecture	Skip to next lecture	N/A	Next lecture will be displayed
8	btnNextQuestion	Skip to next question	N/A	Next question will be displayed

Table 87: Take quiz page – Field, Button, Hyperlink

5.1.2 Admin Interface Design

5.1.2.1 Update brand

The screenshot shows the 'Brand Detail' page in the TRAISY Admin interface. The left sidebar contains navigation links like 'Báo cáo', 'Quản lý kế hoạch', 'Quản lý thương hiệu' (highlighted in blue), 'Quản lý người dùng', 'Quản lý kỹ năng', and 'Kiểm duyệt khóa học'. The main content area has a title 'FPT Software'. A red box labeled '1' highlights the top-left corner of the page. A red box labeled '2' highlights the 'Tên thương hiệu' input field containing 'FPT Software'. A red box labeled '3' highlights the 'Mô tả' text area below it. A red box labeled '4' highlights the green 'Lưu chỉnh sửa' button at the bottom right.

Figure 89: Brand detail page (1)

The screenshot shows the 'Brand Detail' page in the TRAISY Admin interface. The left sidebar contains navigation links like 'Báo cáo', 'Quản lý kế hoạch', 'Quản lý thương hiệu' (highlighted in blue), 'Quản lý người dùng', 'Quản lý kỹ năng', and 'Kiểm duyệt khóa học'. The main content area has a title 'FPT Software'. A red box labeled '5' highlights the logo image. A red box labeled '6' highlights the 'Chọn ảnh' (Select Image) button below it. A green 'Lưu chỉnh sửa' (Save changes) button is visible in the top right corner.

Figure 90: Brand detail page (2)

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	brandName	Brand's name	Yes	No	Text	N/A	N/A
2	txtBrandName	Brand's name input	No	No	TextBox	String	N/A

3	txtBrandDescription	Brand's description input	No	No	TextBox	String	N/A
4	imgBrandPicture	Brand's picture	Yes	No	Image	N/A	N/A
BUTTONS/HYPERLINKS							
No	Function	Description		Validation	Outcome		
5	btnSaveInfo	Save updated information		N/A	N/A		
6	btnChoosePicture	Choose and save updated picture		N/A	N/A		

Table 88: Brand detail page – Field, Button, Hyperlink

5.1.2.2 View list user

#	Tên đăng nhập	Họ và tên	Chức vụ	Chức danh	Trạng thái
1	admin 3	Võ Thạch Nguyên 4	Manager 5	Quản trị viên 6	Hoạt động 7
2	manager	Nguyễn Xuân Mai	Java Developer	Quản lý	Hoạt động
3	author1	Lê Phúc Thịnh	Web Developer	Tác giả	Hoạt động
4	learner1	Phạm Chính Hưng	Project Manager	Học viên	Hoạt động
5	learner2	Đặng Thế Anh	Tester	Học viên	Hoạt động

Figure 91: View list user

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
3	username	Username	Yes	Yes	Label	N/A	N/A
4	userFullname	User's fullname	Yes	Yes	Label	N/A	N/A
5	userTitle	User's title	Yes	Yes	Label	N/A	N/A
6	userRole	User's role	Yes	Yes	Label	N/A	N/A
7	userStatus	User's status	Yes	Yes	Label	N/A	N/A

BUTTONS/HYPERLINKS				
No	Function	Description	Validation	Outcome
1	btnCreateNewUser	Create new user	N/A	Open “Thêm người dùng mới” modal
2	btnSearchUser	Search user	N/A	Search user
8	btnUpdateUser	Update user's information	N/A	Open “Chỉnh sửa người dùng” modal
9	btnPagination	Pagination	N/A	N/A

Table 89: View list user – Field, Button, Hyperlink

5.1.3.2 View list plan

Figure 92: List plans

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	planInfo	Plan overview	Yes	Yes	N/A	N/A	N/A
3	planName	Plan's name	Yes	No	TextBox	String	N/A
4	planTime	Plan's time	Yes	No	TextBox	String	N/A
5	planDescription	Plan's description	Yes	No	TextBox	String	N/A
6	planPicture	Plan's picture	Yes	No	Image	N/A	N/A

BUTTONS/HYPERLINKS				
No	Function	Description	Validation	Outcome

2	btnCreateNewPlan	Create new plan	N/A	Open "Thêm kế hoạch" modal
7	btnViewPlanDetail	View detail information of plan	N/A	Redirect to plan detail page

Table 90: List plans – Field, Button, Hyperlink

5.1.3.4 Update/Remove plan

The screenshot shows the 'Manage course' interface. On the left is a sidebar with navigation links: Báo cáo, Quản lý kế hoạch (selected), Quản lý người dùng, Quản lý kỹ năng, and Kiểm duyệt khóa học. Below this is a list of course categories: DANH SÁCH KẾ HOẠCH, Đào tạo Tester, Đào tạo Designer chuyên nghiệp, Đào tạo Java Developer, Đào tạo Front-End Developer, and Training Fresher Developer. The main content area has tabs: Giới thiệu (selected), Hình ảnh, Giảng viên, Học viên, and Nội dung kế hoạch. The 'Giới thiệu' tab displays a numbered form for updating a plan. The form includes fields for 'Tên kế hoạch' (Đào tạo Tester), 'Thời gian' (01/04/2018 to 30/04/2018), and 'Mô tả' (description of a software test). A 'Lưu chỉnh sửa' (Save changes) button is at the bottom right.

Figure 93: Manage course (1)



Figure 94: Manage course (2)

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
6	txtPlanName	Plan's name	No	Yes	TextBox	String	N/A
7	txtStartDate	Plan's start date	No	Yes	TextBox	String	N/A
8	txtEndDate	Plan's end date	No	Yes	Textbox	String	N/A
9	planDescription	Plan's description	No	Yes	TextArea	String	N/A
BUTTONS/HYPERLINKS							
No	Function	Description	Validation	Outcome			
1	planOverview	Display plan's overview	N/A	Display plan's overview			
2	planPicture	Display plan's picture	N/A	Display plan's picture			
3	planAuthor	Display list of authors that participated in plan	N/A	Display list of authors that participated in plan			
4	planLearner	Display list of learners that participated in plan	N/A	Display list of learners that participated in plan			
5	planContent	Display programs in that plan	N/A	Display programs in that plan			
10	btnSaveInfo	Save updated information	N/A	N/A			
11	btnRemovePlan	Remove plan	N/A	Open confirm modal			
13	btnChoosePicture	Choose picture	N/A	Open "Select file" dialog			

Table 91: Update/Remove plan – Field, Button, Hyperlink

5.1.3.5 Add author to plan

Figure 95: Add author to plan modal

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
2	authorInformation	Author information	Yes	Yes	N/A	N/A	N/A
3	txtAuthorName	Author's name	Yes	Yes	Label	String	N/A
4	txtAuthorTitle	Author's title	Yes	Yes	Label	String	N/A

BUTTONS/HYPERLINKS							
No	Function	Description		Validation	Outcome		
1	btnSearchAuthor	Search author		N/A	N/A		
5	addAuthor	Add auhot to plan		N/A	N/A		
6	btnSaveInfo	Save information		N/A	Modal will close		
7	btnClose	Close modal		N/A	Modal will close		

Table 92: Add author to plan – Field, Button, Hyperlink

5.1.3.7 View list program

Figure 96: List programs section

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length

2	programName	Program's name	Yes	Yes	Label	String	N/A
3	programDuration	Program's duration	Yes	Yes	Label	String	N/A
7	courseInformation	Course information	Yes	Yes	N/A	N/A	N/A
BUTTONS/HYPERLINKS							
No	Function	Description		Validation	Outcome		
1	btnCreateProgram	Create new program		N/A	Open create program modal		
4	btnCreateCourse	Create new course in that program		N/A	Open create course modal		
5	btnMoreOption	More option for program		N/A	Dropdown option will display		
6	showListCourse	Show list of course		N/A	N/A		
8	manageCourse	View details of course		N/A	Redirect to manage this course page		

Table 93: List programs section – Field, Button, Hyperlink

5.1.3.8 Create program

Figure 97: Create new program modal

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	txtProgramName	Program's name	No	Yes	TextBox	String	N/A

2	txtStartDate	Program's start date	No	Yes	Calender Box	Timest amp	N/A
2	txtEndDate	Program's end date	No	Yes	Calender Box	Timest amp	N/A
BUTTONS/HYPERLINKS							
No	Function	Description		Validation	Outcome		
4	btnSaveInfo	Create new program		N/A	Modal will close and redirect to manage that course page.		
5	btnClose	Close modal		N/A	Modal will close		

Table 94: Create new program modal – Field, Button, Hyperlink

5.1.3.9 Manage course feedback

Figure 98: Course's feedbacks section

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
2	feedbackOwnerAvatar	Avatar of user who create that feedback	Yes	Yes	Image	N/A	N/A
3	feedbackOwnerName	Name of user who create that feedback	Yes	Yes	Label	String	N/A

4	feedbackDuration	Duration that feedback exist	Yes	Yes	Label	String	N/A
5	feedbackTitle	Feedback title	Yes	Yes	Label	String	N/A
6	feedbackStatus	Status of feedback (reviewed/unreviewed)	Yes	Yes	Label	String	N/A
7	feedbackDetail	Detail of feedback	Yes	Yes	Label	String	N/A
BUTTONS/HYPERLINKS							
No	Function	Description	Validation	Outcome			
1	btnCreateFeedback	Create new feedback	N/A	Open create feedback modal			
8	btnFeedbackAction	Action for feedback	N/A	Open menu action for feedback, including update and remove			
9	btnReplyFeedback	Reply for this feedback	N/A	Open a reply dialog that let user reply for this feedback			

Table 95: Course's feedback – Field, Button, Hyperlink

5.1.3.10 View list skills

#	Tên kỹ năng	Mô tả kỹ năng	
2	HTML5	HTML5 là một ngôn ngữ cấu trúc và trình bày nội dung	6 7
3	javascript	JavaScript, theo phiên bản hiện hành, là một ngôn ngữ lập trình kịch bản dựa trên đối tượng	
4	Java	Java là một ngôn ngữ lập trình hướng đối tượng và dựa trên các lớp.	
7	Kotlin	Kotlin là một ngôn ngữ ngữ dụng kiểu tĩnh dành cho Java Virtual Machine đã chính thức phát hành phiên bản 1.0.	

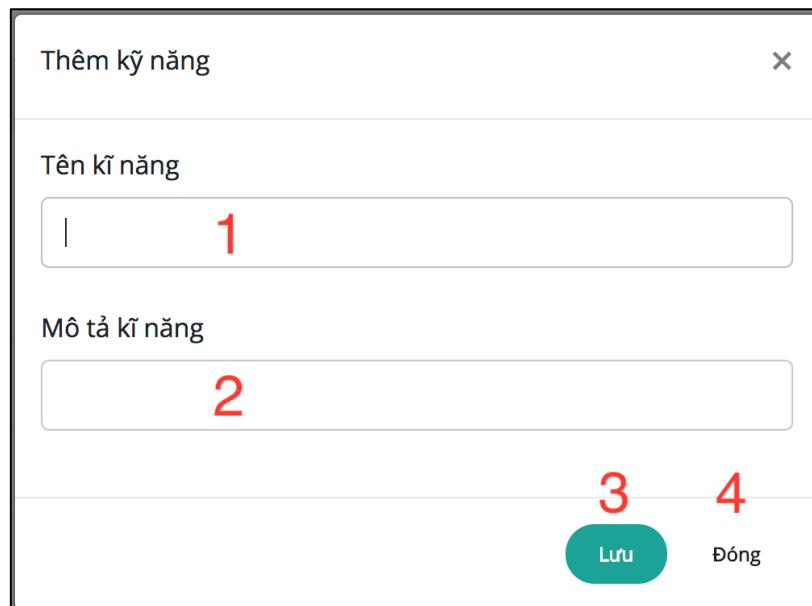
Figure 99: Manage skill page

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
3	skillNumber	Skill's number	Yes	Yes	Textbox	N/A	N/A
4	skillName	Skill's name	Yes	Yes	Textbox	N/A	N/A
5	skillDescription	Skill's description	Yes	Yes	Textbox	N/A	N/A

BUTTONS/HYPERLINKS							
No	Function	Description		Validation	Outcome		
1	btnCreateSkill	Add new skill		N/A	Open "Thêm mới kỹ năng" modal		
2	btnSearchSkill	Search skill		N/A	Search skill		
6	btnUpdateSkill	Update current skill's information		N/A	Open "Chỉnh sửa kỹ năng" modal		
7	btnRemoveSkill	Remove specific skill		N/A	Open confirm modal		
8	btnPagination	Pagination		N/A	View next skills		

Table 96: View list skills – Field, Button, Hyperlink

5.1.3.11 Create skill



The image shows a modal dialog box titled "Thêm kỹ năng". It contains two input fields: "Tên kỹ năng" (Skill name) with the value "1" highlighted in red, and "Mô tả kỹ năng" (Skill description) with the value "2" highlighted in red. At the bottom right are two buttons: a teal "Lưu" (Save) button and a white "Đóng" (Close) button.

Figure 100: Create new skill modal

FIELDS							
--------	--	--	--	--	--	--	--

No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	skillName	Skill's name	No	Yes	Textbox	String	N/A
2	skillDescription	Skill's description	No	Yes	Textbox	String	N/A
BUTTONS/HYPERLINKS							
No	Function	Description		Validation	Outcome		
3	saveInformation	Save new skill's information		N/A	N/A		
4	closeModal	Close modal		N/A	Modal will close		

Table 97: Create new skill modal – Field, Button, Hyperlink

5.1.3.14 View plan report

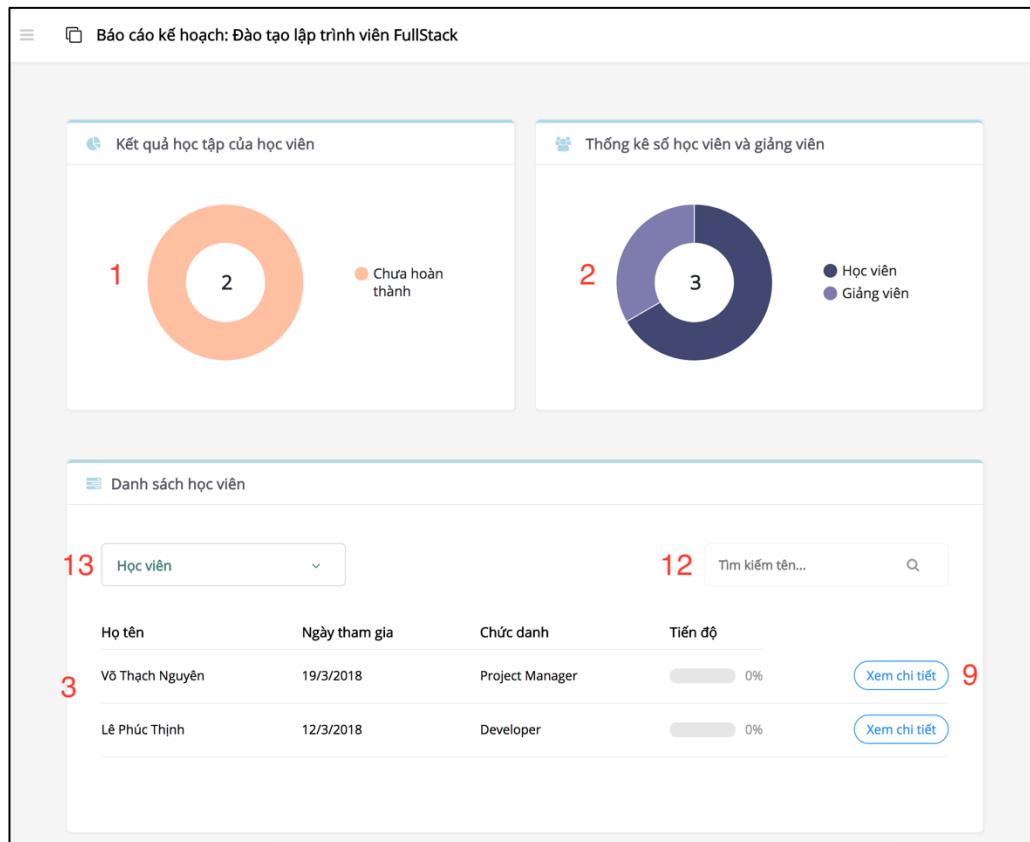


Figure 101: View plan report (1)

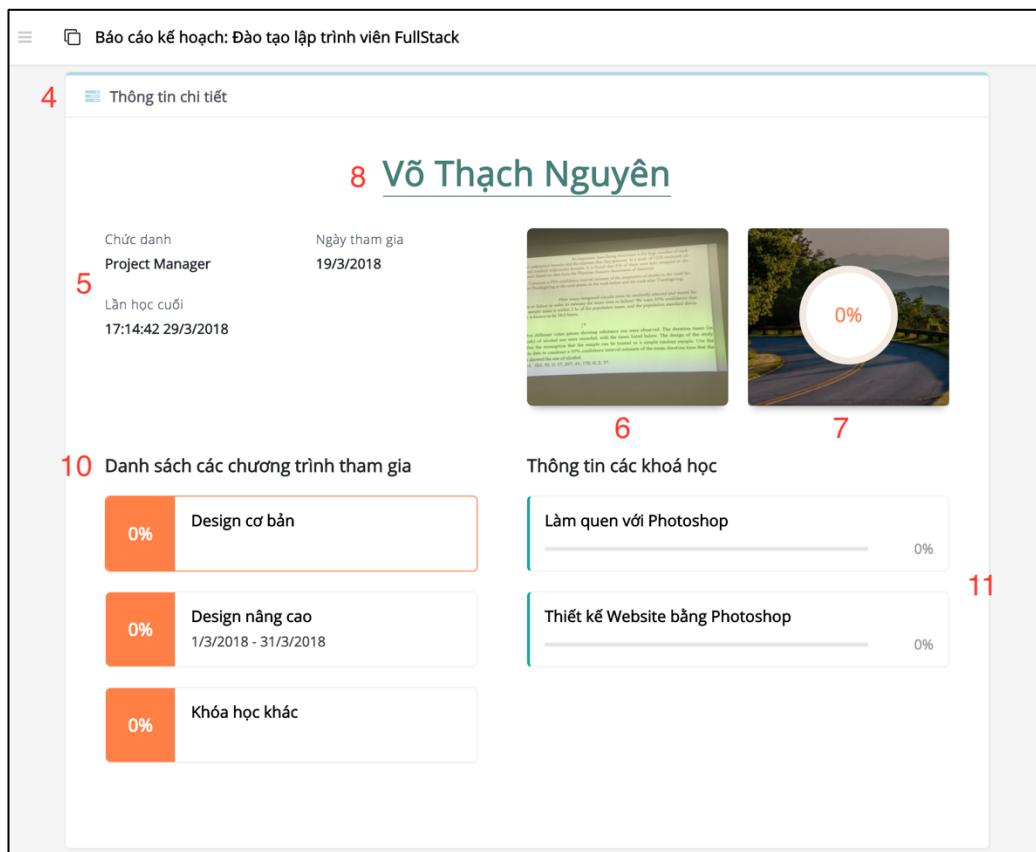


Figure 102: View plan report (2)

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	learnerProgressChart	Pie chart about learning progress of learner	Yes	No	Chart	N/A	N/A
2	numberOfLearnerAndAuthor	Pie chart about number of learner and author in plan	Yes	No	Chart	N/A	N/A
3	learnerList/authorList	List of learners and authors in that plan	Yes	Yes	Table	N/A	N/A
4	userDetailReport	Report and detail information of user (learner, author in plan)	Yes	Yes	N/A	N/A	N/A

5	userInfo	Information of user	Yes	Yes	Label	N/A	N/A
6	userAvatar	Avatar of user	Yes	Yes	Image	N/A	N/A
7	learningProgressOfLearner	Learning progress this plan of a learner	Yes	Yes	Progress Bar	N/A	N/A
8	userName	Fullname of user	Yes	Yes	Label	N/A	N/A
11	courseReport	List of participated course	Yes	Yes	N/A	N/A	N/A
BUTTONS/HYPERLINKS							
No	Function	Description	Validation	Outcome			
9	viewUserDetail	Display user's information	N/A	Display user's information in section 4			
10	participatedProgram	List of participated programs	N/A	Display list of courses in that program			
12	searchUser	Search learner and author	N/A	N/A			
13	selectLeanerOfAuthor	Select table about learner or author	N/A	N/A			

Table 98: View plan report - Field, Button, Hyperlink

5.1.4 Author Interface Design

5.1.4.1 Manage course

The screenshot shows the 'Course's detail page' on the TRAISY platform. The main content area is divided into several sections:

- Section 1 (Top Left):** Includes a search bar and navigation links: 'Giới thiệu', 'Hình ảnh', 'Nội dung bài học', 'Học viên', and 'Đề xuất chỉnh sửa' (with a red notification badge).
- Section 2 (Top Right):** A button labeled 'Công khai'.
- Section 3 (Main Content):**
 - Section 3.1 (Title):** 'Giới thiệu' (Section 1) and 'Thông tin sơ lược về khóa học' (Section 2).
 - Section 3.2 (Input Fields):**
 - 'Tên khóa học': Input field containing 'Java căn bản' (Section 7).
 - 'Mô tả': Text area containing a detailed description of Java (Section 8).
 - 'Kỹ năng khóa học': Input field containing 'Java' (Section 9).
 - Section 3.3 (Buttons):** A green 'Lưu chỉnh sửa' button (Section 11) and a red 'Xoá vĩnh viễn khóa học' button (Section 12).
- Section 4 (Bottom Left):** A sidebar titled 'DANH SÁCH KẾ HOẠCH' containing course categories: 'Đào tạo Tester', 'Đào tạo Designer chuyên nghiệp...', 'Đào tạo Java Developer', 'Đào tạo Front-End Developer', and 'Training Fresher Developer'.
- Section 5 (Bottom Right):** A user profile icon and the name 'Võ Thạch Nguyên'.

Figure 103: Course's detail page

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
7	txtCourseName	Course's name	No	Yes	TextBox	N/A	N/A
8	txtCourseDescription	Course's description	No	No	TextBox	N/A	N/A
9	skillList	Course's skills	No	No	Text	N/A	N/A
BUTTONS/HYPERLINKS							
No	Function	Description	Validation	Outcome			
1	courseIntroduction	View basic information of course	N/A	Open manage information of course			
2	courseImageVideo	View image and video description of course	N/A	Open manage image of course			
3	courseContent	View manage lecture page	N/A	Open manage lecture tab of course			
4	courseStudent	View student that enrolled in course	N/A	Open course's students tab			
5	courseFeedback	View feedbacks in course	N/A	Open manage course's feedback			
6	btnPublishCourse	Publish the course if the course is unpublished or otherwise	N/A	Change the course status			
10	btnChooseSkill	Menu action for course skill	N/A	Open skill manage menu			
11	btnSaveInfo	Save course's information	N/A	N/A			
12	btnRemoveCourse	Remove course	N/A	N/A			

Table 99: Course detail page – Field, Button, Hyperlink

5.1.4.2 Manage section

Figure 104: Manage lecture section

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
2	sectionName	Section's name	Yes	No	Textbox	String	N/A
6	lectureList	Section's list of lectures	Yes	No	N/A	N/A	N/A
BUTTONS/HYPERLINKS							
No	Function	Description	Validation	Outcome			
1	btnCreateSection	Create new section	N/A	Open create section modal			
2	btnEditSection	Edit section	N/A	Open textbox that edit section's name			
3	btnCreateLecture	Create new lecture	N/A	Open create lecture modal			
4	btnRemoveSection	Remove section	N/A	Remove section			
5	btnExpand	Expand section to view its lecture	N/A	Open list lectures of that section			
7	btnViewLecture	View lecture's detail information	N/A	Redirect to manage lecture page			
8	btnEditLecture	Edit lecture's detail information	N/A	Redirect to manage lecture page			

9	btnRemoveLecture	Remove lecture	N/A	Remove lecture
---	------------------	----------------	-----	----------------

Table 100: Manage lecture section – Field, Button, Hyperlink

5.1.4.3 Manage lecture

Figure 105: Manage lecture content page

FIELDS							
No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	txtLectureTitle	Lecture's title	No	Yes	Textbox	String	N/A
3	lectureType	Lecture's type	No	No	N/A	N/A	N/A
4	lectureContent	Lecture's content	No	No	N/A	N/A	N/A
BUTTONS/HYPERLINKS							
No	Function	Description	Validation	Outcome			
2	btnPublishLecture	Publish the lecture if the course is unpublished or otherwise	N/A	Change the lecture status			
3	videoLecture	Video lecture	N/A	Video lecture			
4	textLecture	Text lecture	N/A	Text lecture			

5	powerpointLecture	Powerpoint lecture	N/A	Powerpoint lecture
6	quizLecture	Quiz lecture	N/A	Quiz lecture

Table 101: Manage lecture content page – Field, Button, Hyperlink

6. Database Design

6.1 Entity relationship diagram (ERD)

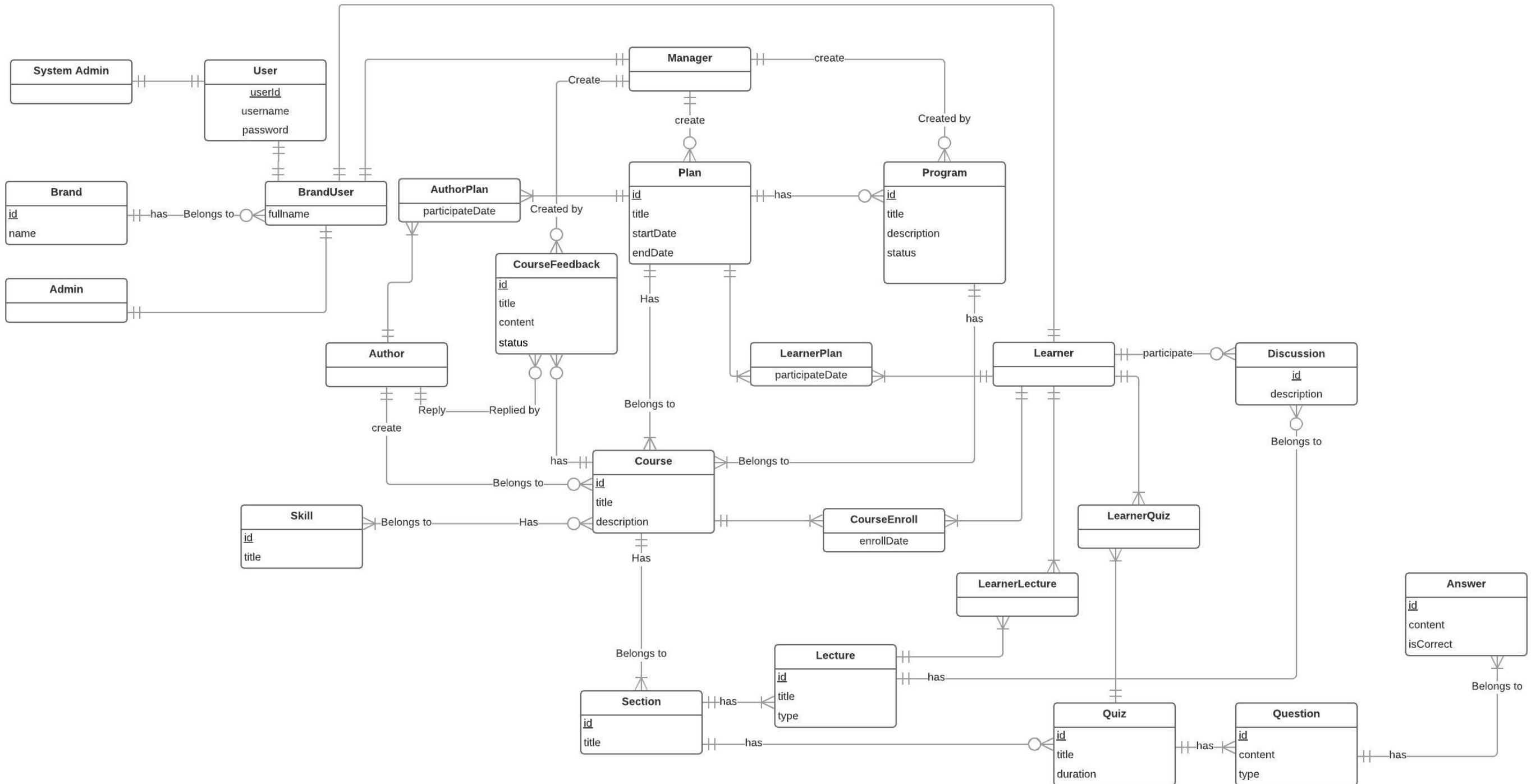


Figure 106: Entity relationship diagram (ERD)

6.2 Entity dictionary

Entity Data Dictionary: describe content of all entities	
Entity name	Description
User	Contains the user's information
System Admin	Contains the system admin's information
Brand	Contains the brand's information
Brand User	Shows relationship between Brand and User
Admin	Contains the admin's information
Manager	Contains the manager's information
Author	Contains the author's information
AuthorPlan	Shows relationship between Author and Plan
Plan	Contains the plan's information
Program	Contains the program's information
CourseFeedback	Contains the feedback's information
Course	Contains the course's information
Skill	Contains the skill's information
Section	Contains the section's information
LearnerPlan	Shows relationship between Learner and Plan
CourseEnroll	Contains the information about what course that learner was enrolled
Lecture	Contains the lecture's information
LearnerLecture	Contains the information about what lecture that learner was learned
Learner	Contains the learner's information
LearnerQuiz	Contains the information about what quiz that learner was taken
Quiz	Contains quiz's information
Discussion	Contains discussion's information
Question	Contains the question's information
Answer	Contains the answer's information

Table 102: Entity data dictionary

7. Algorithms

7.1 Duplicate course

7.1.1 Definition

When a user visits a course, the web will navigate to that course's page with unique URL for each course. Moreover, the course's URL needs to be related to the course's title.

7.1.2 Define problem

Each course has its own title and content. However, we need a unique URL that related to the title for web to navigate between courses. The problem is that when an author creates a course, the course title may be duplicated to other course; therefore, we need an algorithm to differentiate between courses that have the same title but different content.

7.1.3 Solution

To solve this problem, we should follow these steps:

- Each course will be distinguished by its URL that saved in database, which is created through:
 1. Transfer the course title into non-Unicode characters by removing the Unicode accents and diacritics
 2. Replace special characters such as white space with hyphen (-)
 3. Lowercase all the course's title
 4. If the URL is already existed in database, the URL will be appended with a number behind.
 5. The URL structure will be as bellow:
 - Khoa học HTML: khoa-hoc-html (if course title is not duplicated)
 - Khoa học HTML: khoa-hoc-html-2 (if course title is duplicated)
 6. The unique course URL will be inserted to database.

7.1.4 Complexity

In total, the complexity of this algorithm is **N**

7.1.5 Flow chart

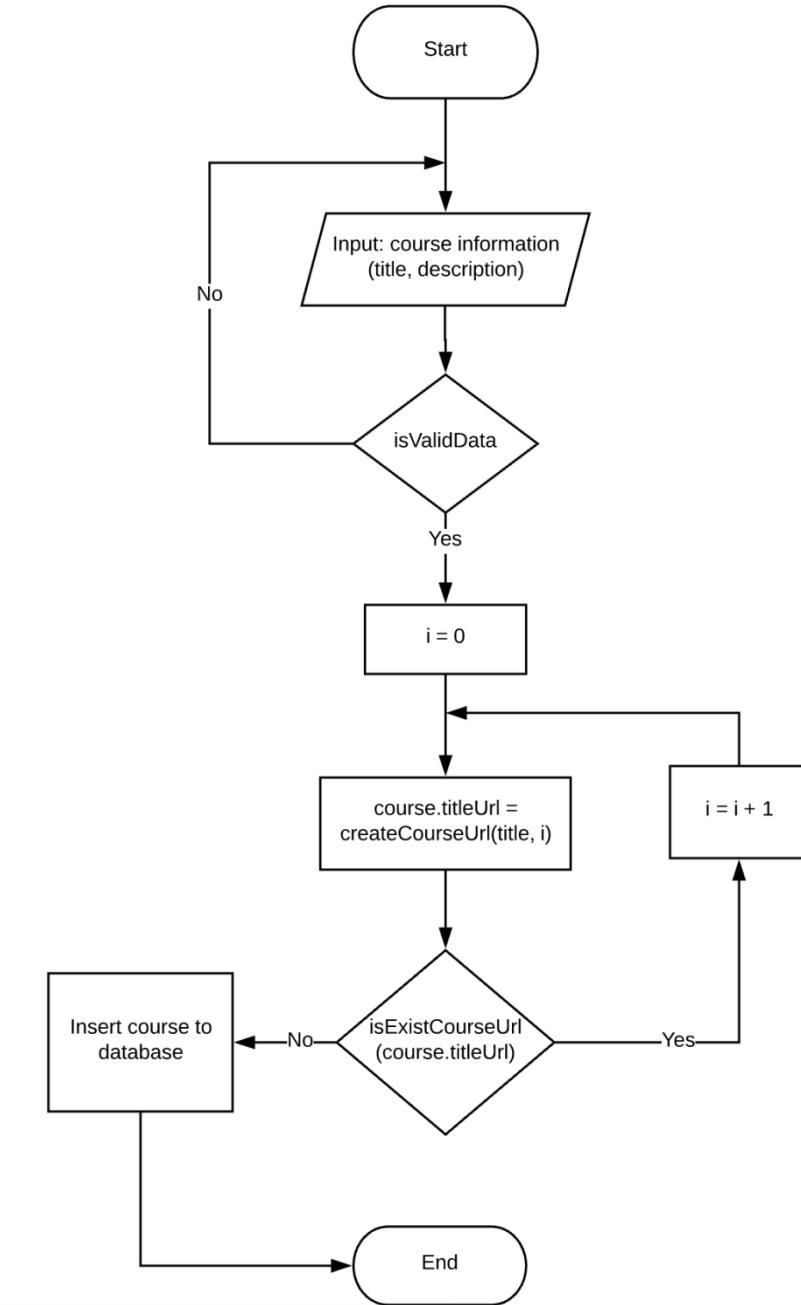


Figure 107: Entity data dictionary

E. Report No. 5 System Implementation & Test

1. Introduction

1.1 Overview

This section describes the approach and methodologies used by group to plan, organize and manage the testing of TRAISY system. It provides in the detail all necessary information about the implementation and testing procedure of the system included test plans, test cases, test result, test environments, pass/fail criteria and risks estimations as well as a checklist to cover all possible cases.

1.2 Test approach

- Goal: Test all features in the whole TRAISY system based on the core flow.
- Method: black-box testing
- Technique: check list

The testing for this project will consists of Integration System test level. Testing the program which was integrated and as a complete system to ensure that the software requirements have been met.

- Integration testing would be performed by all member of team and approved by team leader.

System testing is focused on assessing the system's reliability. This process is concerned with finding errors that result from unanticipated interactions between components and component interface problems

2. Database Relationship Diagram

2.1 Physical diagram

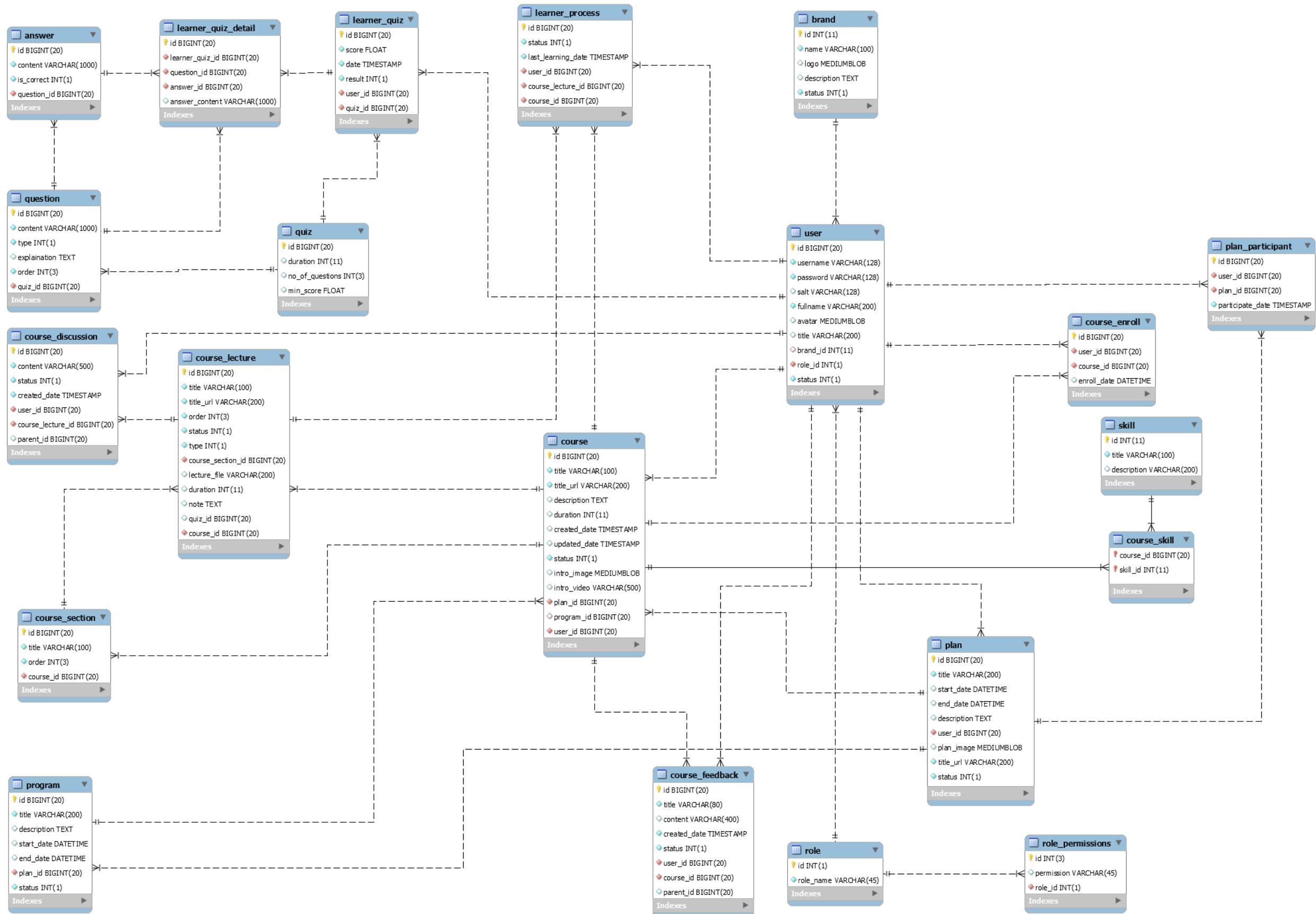


Figure 108: Physical diagram

2.2 Data dictionary

DATA DICTIONARY: DESCRIBE CONTENT OF ALL TABLES	
Table name	Description
user	Contains the user information
brand	Contains the brand information
plan_participant	Contains the information of a user that participated in plan
plan	Contains the plan information
program	Contains the program information
course	Contains the course information
course_discussion	Contains the course's discussion information
course_section	Contains the section information
skill	Contains the skill information
course_skill	Contains the information of skill that belongs to course
course_lecture	Contains the lecture information
learner_process	Contains the process information of a user when learning a lecture
quiz	Contains the quiz information
question	Contains the question information
answer	Contains the answer information
learner_quiz	Contains the information of learner when that learner take quiz
learner_quiz_detail	Contains the result of learner's quiz
course_feedback	Contains the course's feedback information
role	Contains the role's information
role_permissions	Contains the role's permission information
course_enroll	Contains the information of a user that enrolled in course

Table 103: Data Table Dictionary

Table Name	Attributes	Description	Domain	Null
User	id	Unique identifier of an account	BIGINT(20)	NO
	username	Username of user	VARCHAR(128)	NO
	password	Password of user	VARCHAR(128)	NO
	salt	Encryption of password	VARCHAR(128)	YES
	fullname	Fullscreen of user	VARCHAR(200)	NO
	avatar	Avatar of user	MEDIUMBLOB	YES
	title	Title of user	VARCHAR(200)	YES
	brand_id	Identifier of the brand of user	INT(11)	YES
	role_id	Identifier of the role of user	INT	NO
	status	Status of user	INT	NO

brand	id	Identifier of the brand	INT	NO
	name	Name of brand	VARCHAR(100)	NO
	logo	Logo of brand	MEDIUMBLOB	YES
	description	Description of brand	TEXT	YES
	status	Status of brand	INT	NO
plan_participant	id	Identifier of the plan_participant	BIGINT	NO
	user_id	Identifier of the user in plan_participant	BIGINT	NO
	plan_id	Identifier of plan in plan_participant	BIGINT	NO
	participate_date	participate date of user in plan	TIMESTAMP	NO
plan	id	Identifier of plan	BIGINT	NO
	title	Title of plan	VARCHAR(200)	NO
	start_date	Start date of plan	DATETIME	YES
	end_date	End date of plan	DATETIME	YES
	description	Description of plan	TEXT	YES
	user_id	Identify of creator of plan	BIGINT	NO
	plan_image	Plan's introduction image	MEDIUMBLOB	YES
	title_url	Url of plan	VARCHAR(200)	NO
	status	Status of plan	INT	NO
program	id	Identifier of program	BIGINT	NO
	title	Title of program	VARCHAR(200)	NO
	description	Description of program	TEXT	YES
	start_date	Start date of program	DATETIME	YES
	end_date	End date of program	DATETIME	YES
	plan_id	Identifier of plan that program belongs to	BIGINT	NO
	status	Status of program	INT	NO
course	id	Identifier of course	BIGINT	NO
	title	Title of course	VARCHAR(100)	NO
	title_url	Url of course	VARCHAR(200)	NO
	description	Description of course	TEXT	YES
	duration	Duration of course	INT	YES
	created_date	Create date of course	TIMESTAMP	YES

	updated_date	Last updated date of course	TIMESTAMP	YES
	status	Status of course	INT	NO
	intro_image	Introduction image of course	MEDIUMBLOB	YES
	intro_video	Introduction video of course	VARCHAR(500)	YES
	plan_id	Identifier of plan that course belongs to	BIGINT	NO
	program_id	Identifier of program that course belongs to	BIGINT	YES
	user_id	Identifier of user that create course	BIGINT	NO
course_discussion	id	Identifier of course discussion	BIGINT	NO
	content	Content of discussion	VARCHAR(500)	NO
	status	Status of discussion	INT	NO
	created_date	Create date of discussion	TIMESTAMP	NO
	user_id	Identifier of user that create discussion	BIGINT	NO
	course_lecture_id	Identifier of lecture that belongs to course	BIGINT	NO
	parent_id	Identifier of top discussion that contains it	BIGINT	YES
course_section	id	Identifier of course section	BIGINT	NO
	title	Title of course section	VARCHAR(100)	NO
	order	Order of lecture in specific course	INT	NO
	course_id	Identifier of course that contains this lecture	BIGINT	NO
skill	id	Identifier of skill	INT	NO
	title	Title of skill	VARCHAR(100)	NO
	description	Description of skill	VARCHAR(200)	YES
course_skill	course_id	Identifier of course that have skill	BIGINT	NO
	skill_id	Identifier of skill that belongs to that course	INT	NO

course_lecture	id	Identifier of lecture	BIGINT	NO
	title	Title of lecture	VARCHAR(100)	NO
	title_url	Url of lecture	VARCHAR(200)	NO
	order	Order of lecture in specific course	INT	NO
	status	Status of lecture	INT	NO
	type	Type of course	INT	NO
	course_section_id	Identifier of course section that contains lecture	BIGINT	NO
	lecture_file	File belongs to lecture	VARCHAR(200)	YES
	duration	Duration of video lecture	INT	YES
	note	Note of lecture	TEXT	YES
learner_process	quiz_id	Identifier of quiz that belongs to lecture	BIGINT	YES
	course_id	Identifier of course that contains lecture	BIGINT	NO
	id	Identifier of process of specific learner	BIGINT	NO
	status	Status of process	INT	NO
	last_learning_date	Last date that learner learnt	TIMESTAMP	NO
	user_id	Identifier of user that learn lecture	BIGINT	NO
quiz	course_lecture_id	Identifier of course lecture	BIGINT	NO
	course_id	Identifier of course that contains lecture	BIGINT	NO
	id	Identifier of quiz	BIGINT	NO
	duration	Time to finishing a quiz	INT	YES
question	no_of_questions	Number of question in a quiz	INT	YES
	min_score	Min score that need to be passed	FLOAT	YES
	id	Identifier of question	BIGINT	NO
	content	Content of question	VARCHAR(1000)	NO
	type	Type of question	INT	NO
explaination	Explanation for question	TEXT	NO	
	order	Order of question in quiz	INT	NO

	quiz_id	Identifier of quiz that contains quesiton	BIGINT	NO
answer	id	Identifier of answer	BIGINT	NO
	content	Content of answer	VARCHAR(1000)	NO
	is_correct	Right or wrong answer of question	INT	NO
	question_id	Identifier of question that contain the answer	BIGINT	NO
learner_quiz	id	Identifier of a specific learner has taken a specifc quiz	BIGINT	NO
	score	Score of quiz that learner has taken	FLOAT	NO
	date	Date that learner takes the quiz	TIMESTAMP	NO
	result	Result of quiz, include pass and fail	INT	NO
	user_id	Identifier of user who has taken quiz	BIGINT	NO
learner_quiz_detail	quiz_id	Identifier of quiz that learner has taken	BIGINT	NO
	id	Identifier of a specific learner has taken a specifc quiz in more detail	BIGINT	NO
	learner_quiz_id	Identifier of a specific learner has taken a specifc quiz	BIGINT	NO
	question_id	Identifier of question	BIGINT	NO
	answer_id	Identifier of answer	BIGINT	NO
	answer_content	Content of answer	VARCHAR(1000)	YES
course_feedback	id	Identifier of course's feedback	BIGINT	NO
	title	Title of feedback	VARCHAR(80)	NO
	content	Content of feedback	VARCHAR(400)	YES
	created_date	Date that feedback have been created	TIMESTAMP	NO
	status	Status of feedback	INT	NO
	user_id	Identifier of user who create the feedback	BIGINT	NO
	course_id	Identifier of course that contains feedback	BIGINT	NO

	parent_id	Identifier of topic feedback that contains it	BIGINT	YES
role	id	Identifier of role	INT	NO
	role_name	Name of role	VARCHAR(45)	NO
role_permissions	id	Identifier of role's permission	INT	NO
	permission	Permission of the role	VARCHAR(45)	YES
	role_id	Identifier of role	INT	NO
course_enrol l	id	Identifier of courses enroll	BIGINT	NO
	user_id	Identifier of user	BIGINT	NO
	course_id	Identifier of course	BIGINT	NO
	enroll_date	Date that the user has enrolled to the course	DATETIME	YES

Table 104: Data Table Description

3. Test plan

The overall purpose of testing is to ensure TRAISY meets its entire technical, functional and business requirement. The purpose of this document is to describe the overall test plan and strategy for testing the TRAISY. The approach described in this document provides the framework for all testing related to this application. Individual test cases will be written for each version of the application that is released. This document will also be updated as required for each release.

3.1 Features to be tested

- System admin: Add brand
- Manager: Create user, create plan, create program
- Author: Create course, create lecture, create quiz
- Learner: Search course, update profile

3.2 Features not to be tested

- View report
- Learn lecture

4. System Testing Test Case

4.1 Test cases

4.1.1 <System admin> Add brand

ID	Test Case Description	Precondition	Test case procedure	Expected output	Inter-test case dependence	Result	Test Date
Add1	Test system admin add a branch successfully	User already logged in the system as system admin	1.User input required fields 2.User send create command	1.System shows “Tạo thương hiệu thành công”	N/A	Pass	07/04/18
Add2	Test user create a brand with wrong input requirement	User already logged in the system as system admin	1.User input blank or wrong requirement in field “Tên thương hiệu” 2.User send create command	2.System shows error message “Tên thương hiệu không được trống”	N/A	Pass	07/04/18

Table 105: Test case <System Admin> Add branch

4.1.2 <Manager> Create user

ID	Test Case Description	Precondition	Test case procedure	Expected output	Inter-test case dependence	Result	Test Date
Create1	Test manager create a user successfully	User already logged in the system as manager	1. Manager inputs required fields 2. Manager sends create command	1. System shows “Tạo người dùng thành công”	N/A	Pass	07/04/18
Create2	Test manager create a user with wrong input requirement	User already logged in the system as manager	1. Manager inputs blank or wrong requirement 2. Manager sends create command	2. System shows error message for that field	N/A	Pass	07/04/18

Create3	Test manager create a user with username already existed in database	There is already a record in database with that user name	1. Manager inputs username that duplicated with others user 2. Manager sends create command	2. System shows “Vui lòng nhập tên đăng nhập không bị trùng”	N/A	Pass	07/04/18
---------	--	---	--	--	-----	------	----------

Table 106: Test case <Manager> Create user

4.1.1 <Author> Create Course - Create Lecture - Create Quiz

ID	Test Case Description	Precondition	Test case procedure	Expected output	Inter-test case dependence	Result	Test Date
CREATE COURSE							
	Test that user creates course with valid input	User login to system as an author	<ol style="list-style-type: none"> User clicks on “Thêm khoá học” button. User inputs “Javascript course” User clicks on “Tạo khoá học” button 	<ol style="list-style-type: none"> The course has been created successfully. Redirect to course overview page immediately 	N/A		
	Test that user creates course with empty input	User login to system as an author	<ol style="list-style-type: none"> User clicks on “Thêm khoá học” button. User inputs nothing User clicks on “Tạo khoá học” button 	Error message will be shown at the bottom of input with content: “Tên khoá học không được để trống”.	N/A		

	Test that user creates course with invalid input	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thêm khóa học” button. 2. User inputs a string with length longer than 80 3. User clicks on “Tạo khóa học” button 	Error message will be shown at the bottom of input with content: “Tên khóa học không được quá 80 kí tự”.	N/A		
CREATE LECTURE							
	Test that user creates lecture with valid input	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thêm bài học” button. 2. User inputs “Javascript introduction” 3. User clicks on “Thêm bài mới” button 	1. The lecture has been created successfully.	N/A		
	Test that user creates lecture with empty input	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thêm bài học” button. 2. User inputs nothing 3. User clicks on “Thêm bài mới” button 	Error message will be shown at the bottom of input with content: “Tên bài học không được để trống”.	N/A		

	Test that user creates lecture with invalid input	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thêm bài học” button. 2. User inputs a string with length longer than 80 3. User clicks on “Thêm bài mới” button 	Error message will be shown at the bottom of input with content: “Tên bài học không được quá 80 kí tự”.	N/A		
CREATE QUIZ							
	Test that user creates quiz successfully	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thêm bài học” button 2. User input “Javascript Introduction” 3. User click on “Chỉnh sửa” button 4. User select “Bài kiểm tra” tab 5. User clicks on “Thêm câu hỏi” button 6. User inputs valid data 7. User clicks on “Thêm” button 	Question will be created and display immediately	N/A		

	Test that user creates quiz with invalid data	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thêm bài học button” 2. User input “Javascript Introduction” 3. User click on “Chỉnh sửa” button 4. User select “Bài kiểm tra” tab 5. User clicks on “Thêm câu hỏi” button 6. User inputs nothing 7. User clicks on “Thêm” button 	<ol style="list-style-type: none"> 1. Error message will be shown at the bottom of input with content: “Nội dung câu hỏi từ 5 đến 500 ký tự.”. 2. Error message will be shown at the bottom of input with content: “Nội dung câu trả lời từ 5 đến 500 ký tự.”. 	N/A		
--	---	-----------------------------------	--	--	-----	--	--

Table 107: Test case <Author> Create course – Create lecture – Create quiz

4.1.1 <Learner> Search Course - Update Profile

ID	Test Case Description	Precondition	Test case procedure	Expected output	Inter-test case dependence	Result	Test Date
SEARCH COURSE							
	Test that user searchs course successfully	User login to system as a learner	<ol style="list-style-type: none"> 1. User inputs search keyword 2. User clicks on search button 	Modal that contains list of result which suitable with keyword will be displayed.	N/A		

	Test that user searches course fail	User login to system as a learner	<ol style="list-style-type: none"> 1. User inputs search keyword 2. User clicks on search button 	Modal with empty result will be displayed.	N/A		
UPDATE PROFILE							
	Test that user updates profile successfully	User login to system as a learner	<ol style="list-style-type: none"> 1. User clicks on “Thông tin tài khoản” hyperlink 2. User inputs valid data 3. User clicks on “Lưu thay đổi” button 	User's profile will be updated successfully	N/A		
	Test that user updates profile unsuccessfully with empty fullname	User login to system as a learner	<ol style="list-style-type: none"> 1. User clicks on “Thông tin tài khoản” hyperlink 2. User inputs nothing to field “Họ tên” 3. User clicks on “Lưu thay đổi” button 	Error message will be shown at the bottom of input with content: “Họ tên không được để trống”.	N/A		
	Test that user updates profile unsuccessfully with empty title	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thông tin tài khoản” hyperlink 2. User inputs nothing to field “Chức danh” 3. User clicks on “Lưu thay đổi” button 	Error message will be shown at the bottom of input with content: “Chức danh không được để trống”.	N/A		

	Test that user updates profile unsuccessfully with invalid fullname	User login to system as a learner	<ol style="list-style-type: none"> 1. User clicks on “Thông tin tài khoản hyperlink” 2. User inputs string longer than 80 characters in “Họ tên” field 3. User clicks on “Lưu thay đổi” button 	Error message will be shown at the bottom of input with content: “Họ tên không được quá 80 kí tự”.	N/A		
	Test that user updates profile unsuccessfully with invalid title	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thông tin tài khoản hyperlink” 2. User inputs string longer than 80 characters in “Chức danh” field 3. User clicks on “Lưu thay đổi” button 	Error message will be shown at the bottom of input with content: “Chức danh không được được quá 80 kí tự”.	N/A		
	Test that user update profile unsuccessfully with invalid password	User login to system as an author	<ol style="list-style-type: none"> 1. User clicks on “Thông tin tài khoản hyperlink” 2. User inputs “123456” to password field 3. User inputs “789456” to 	Error message will be shown at the bottom of input with content: “Mật khẩu và xác nhận mật khẩu phải giống nhau”.	N/A		

			confirm password field. 4. User clicks on “Lưu thay đổi” button				
--	--	--	--	--	--	--	--

Table 108: Test case <Learner> Search Course – Update Profile

F. Report No. 6 Software User's Manual

1. Installation Guide

1.1 Setting up environment at server side

The specifications are based on the dependencies requirements and performance test result from previous section of this document.

1.1.1 Hardware requirement

- For server:

Criteria	Minimum Requirements
Internet Connection	Cable (4 Mbps)
Computer Processor	64-bit CPU 1.4GHz with 2 cores
Computer Memory	4 GB RAM

Table 109: Hardware requirement for server

- For client:

Criteria	Minimum Requirements
Internet Connection	Cable (1 Mbps or higher)
Computer Processor	Intel® Core i3 1.4GHz
Computer Memory	1GB RAM

Table 110: Hardware requirement for client

1.1.2 Software requirement

- For server:

Name	Name/Version
Operating System	Windows Server 2008 R2
Environment	Java Runtime Environment 1.8
Web Server	Apache 2.4
Database Systems	MySQL 5.7

Table 111: Software requirement for server

- For client:

Name	Name/Version
Operating System	Windows 7 or higher
Web browser	Firefox (v52 or higher), Chromes (v28 or higher)

Table 112: Software requirement for client

1.2 Web Applications Deployment Process

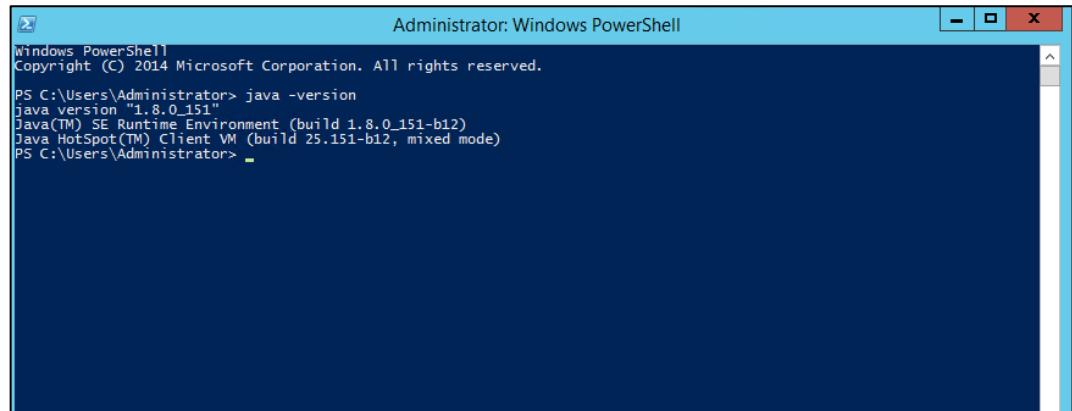
1.2.1 Check environment

Check Java Runtime Environment version by following these steps:

Step 1: Open the PowerShell

Step 2: Check Java Runtime Environment version by running this command: java -version

The result should be same as the example below. If the java version is not 1.8 or higher, please download & install this Java Runtime Environment version.



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2014 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> java -version
java version "1.8.0_151"
Java(TM) SE Runtime Environment (build 1.8.0_151-b12)
Java HotSpot(TM) Client VM (build 25.151-b12, mixed mode)
PS C:\Users\Administrator>
```

Figure 109: Check Java Runtime Environment Version

Make sure that MySQL has been install and running as Windows services.

Step 1: open run windows, type "services.msc" and click ok.

Step 2: In the Services Window, check if MySQL is running.

This picture below is the correct state for MySQL.

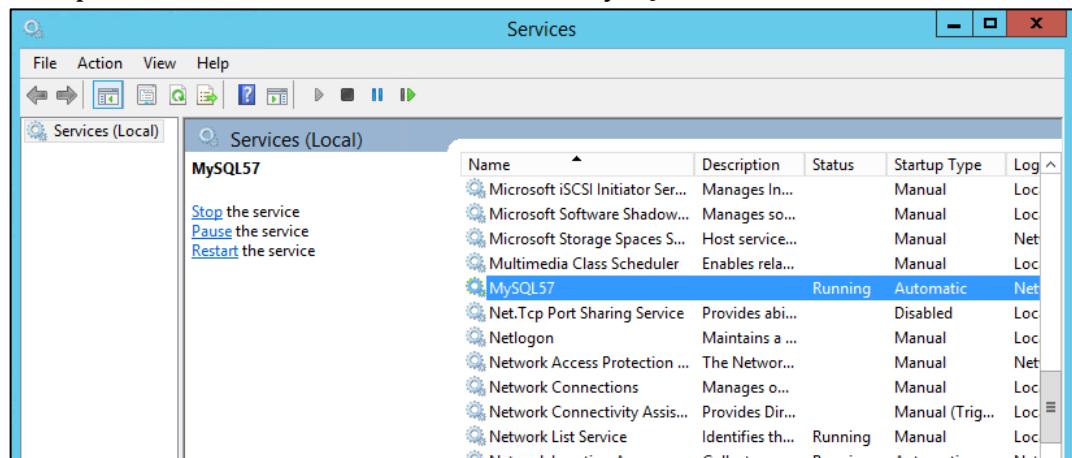


Figure 110: Check MySQL57 status

If MySQL is not running, turn on the service following these steps:

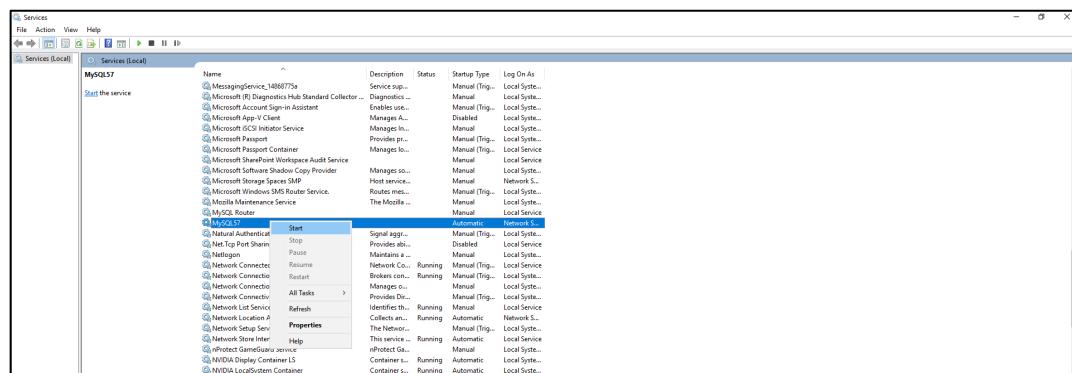


Figure 111: Start MySQL57

1.2.2 Import database

Open MySQL Workbench and follow the following steps to import database:

Using file `traisy.sql` located under Deployment directory from this document as a directory file in step 2.

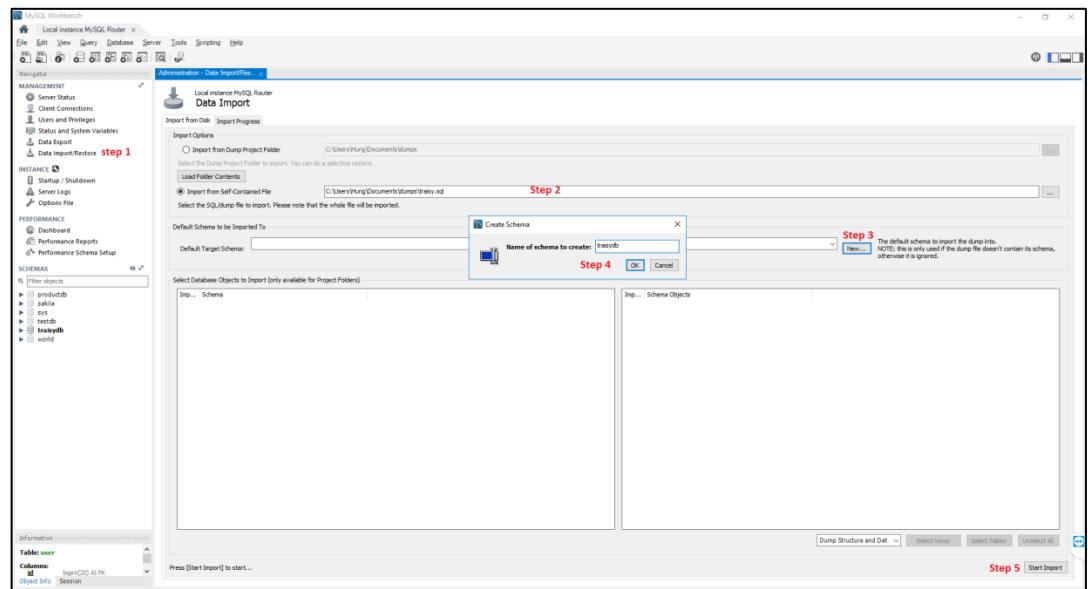


Figure 112: Import Database

Step 1: Click Data Import

Step 2: Select database to import

Step 3: Click new to create new Target Schema

Step 4: Enter “`traisydb`” and click OK

Step 5: Click on start import

1.2.2 Setup wowza streaming engine

This wowza streaming engine is for video streaming.

Step 1: Go to the [Wowza Downloads webpage](#) and click Download for the installer for your desired operating system.

Step 2: When the download completes, double click the installer and follow the installing wizard

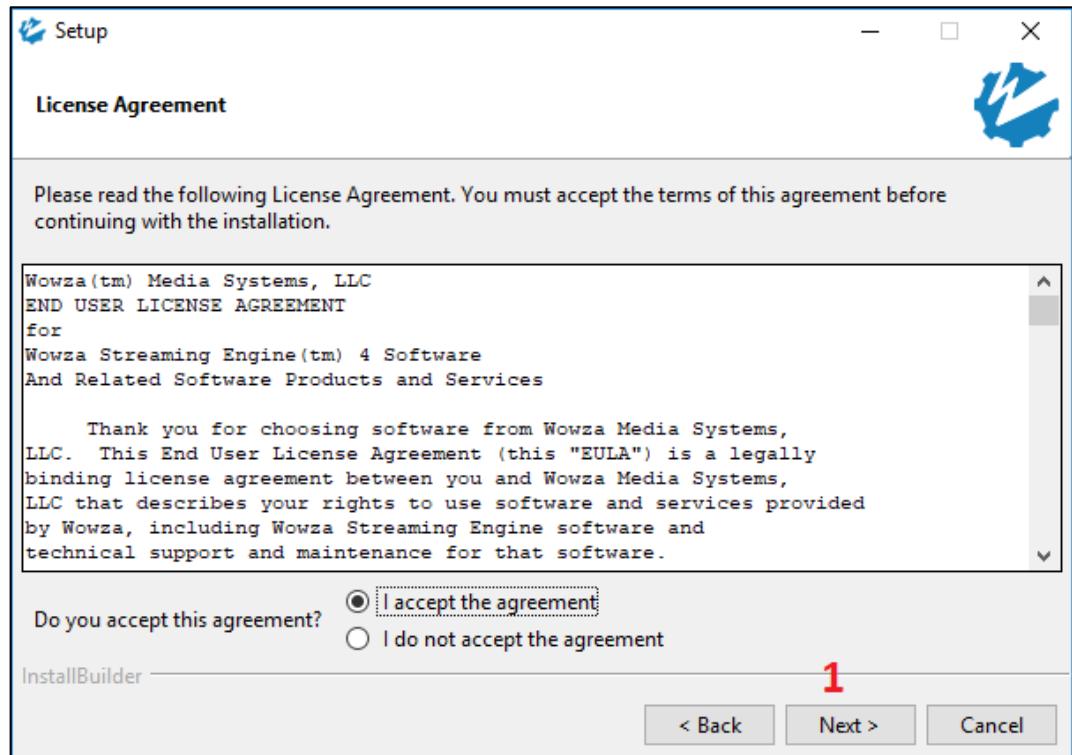


Figure 113: Install Wowza – Step 1

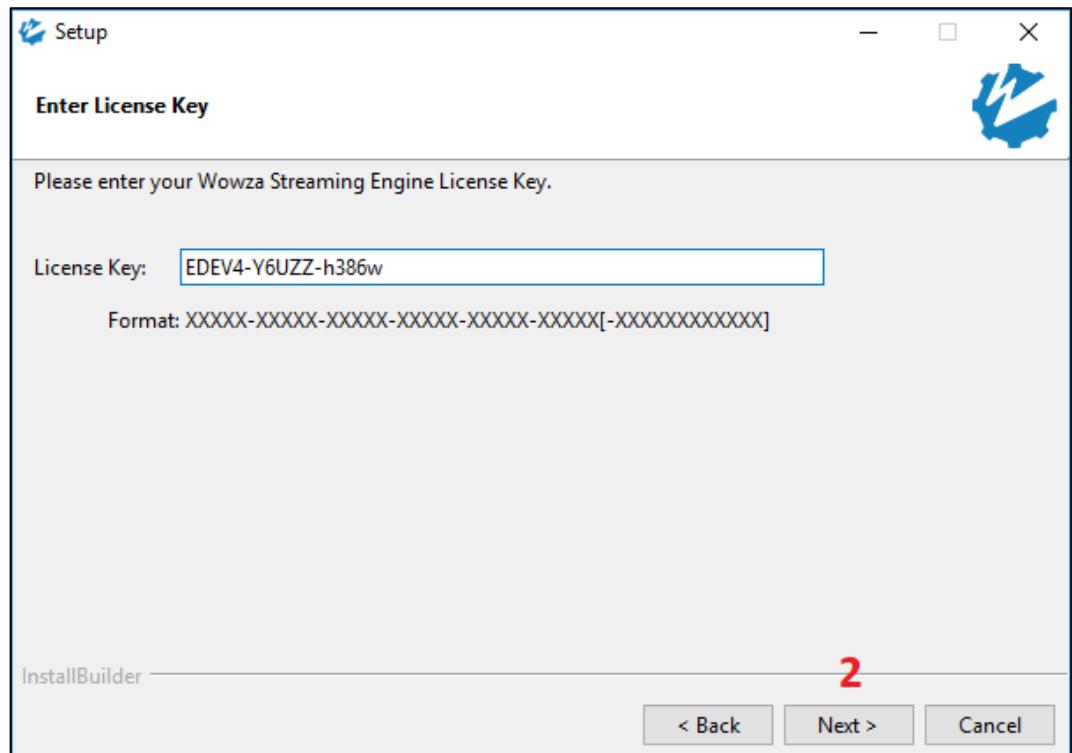


Figure 114: Install Wowza – Step 2

2: Enter product licensed key that wowza send to your email when register account and click next

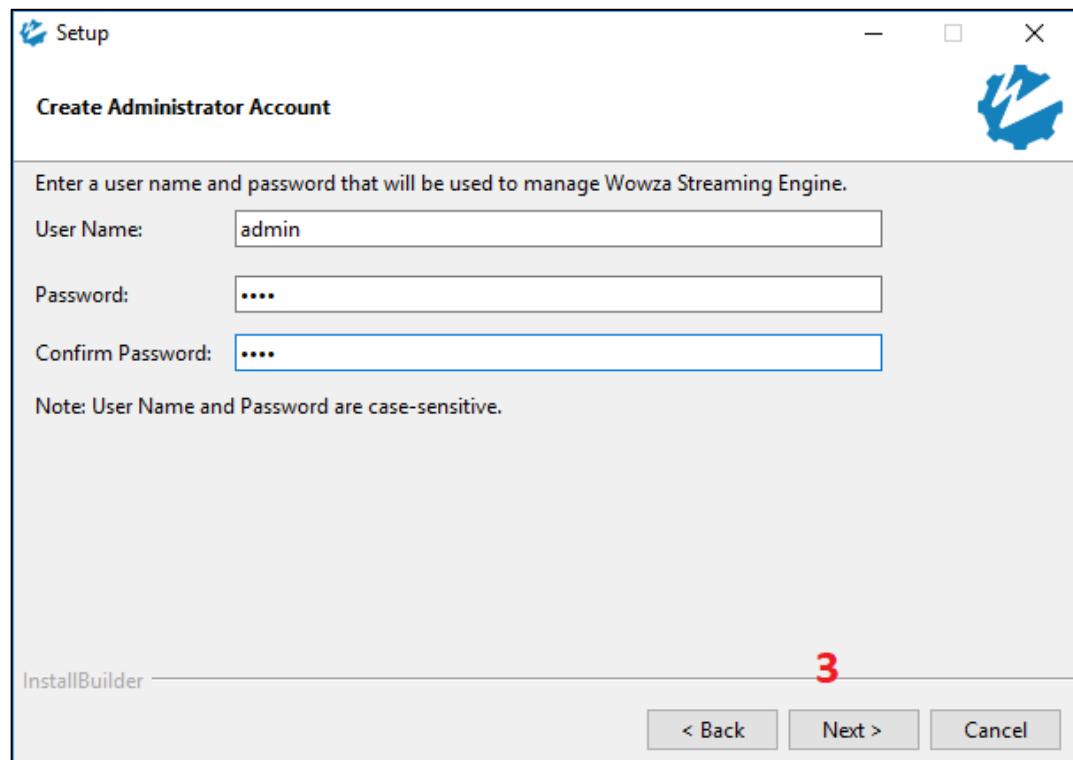


Figure 115: Install Wowza – Step 3

3: Create your admin account. You'll use this account to sign in to the browser-based Wowza Streaming Engine Manager. Click next after you finish.

4: Click next until finish the installing

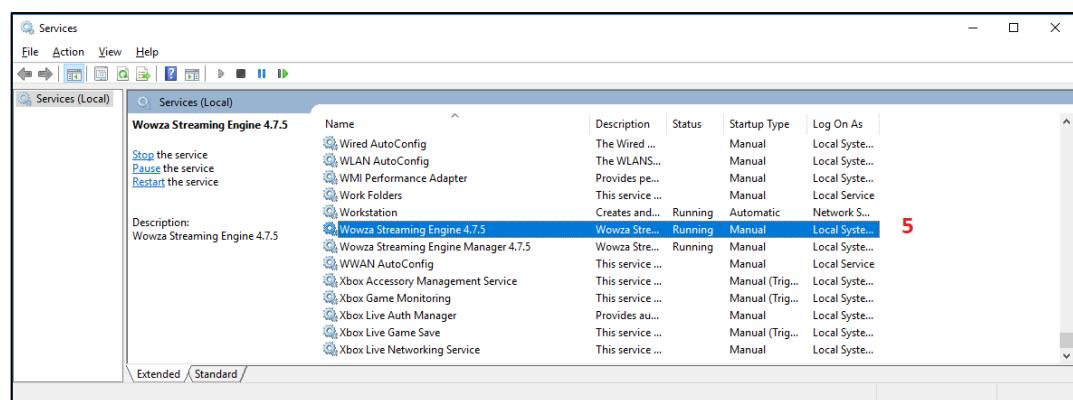


Figure 116: Install Wowza – Step 5

5: start the wowza streaming engine service

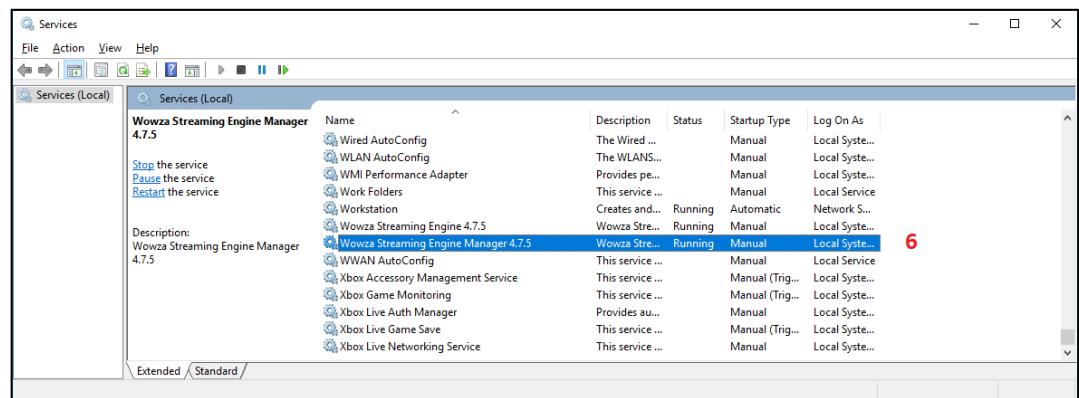


Figure 117: Install Wowza – Step 6

6: Start the wowza streaming engine manager

Step 3: Set up directory for wowza video folder

Open Streaming Engine Software in a web browser with the following URL: <http://localhost:8088/engineManager>

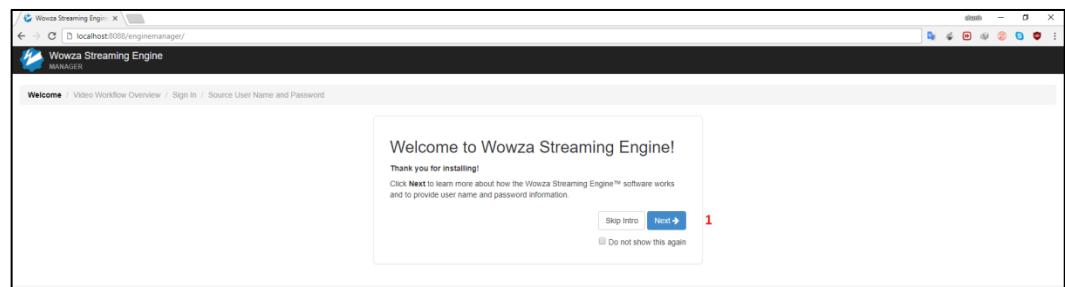


Figure 118: Setup directory – Step 1

1: Follow the installing wizard

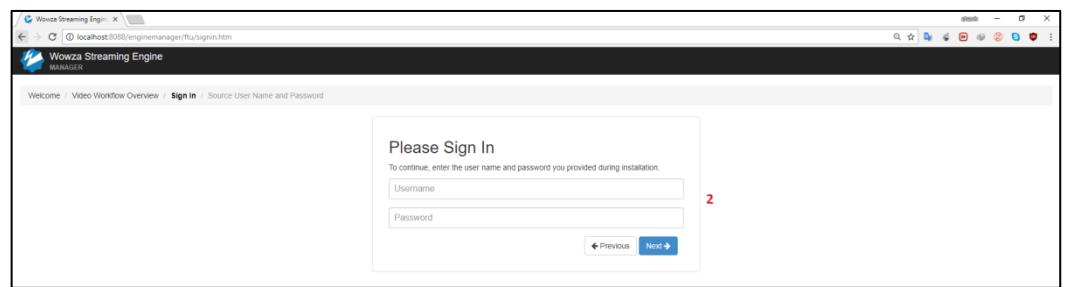


Figure 119: Setup directory – Step 2

2: Enter account that created in step 2 and click next until you reach the home page

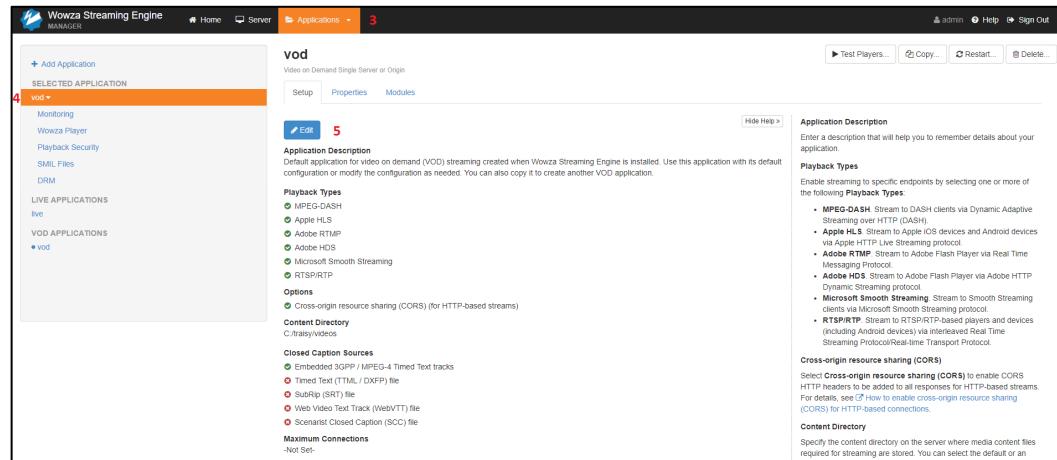


Figure 120: Setup directory - Step 5

- 3: Click on application
- 4: Choose vod in the left panel
- 5: Click "Edit"

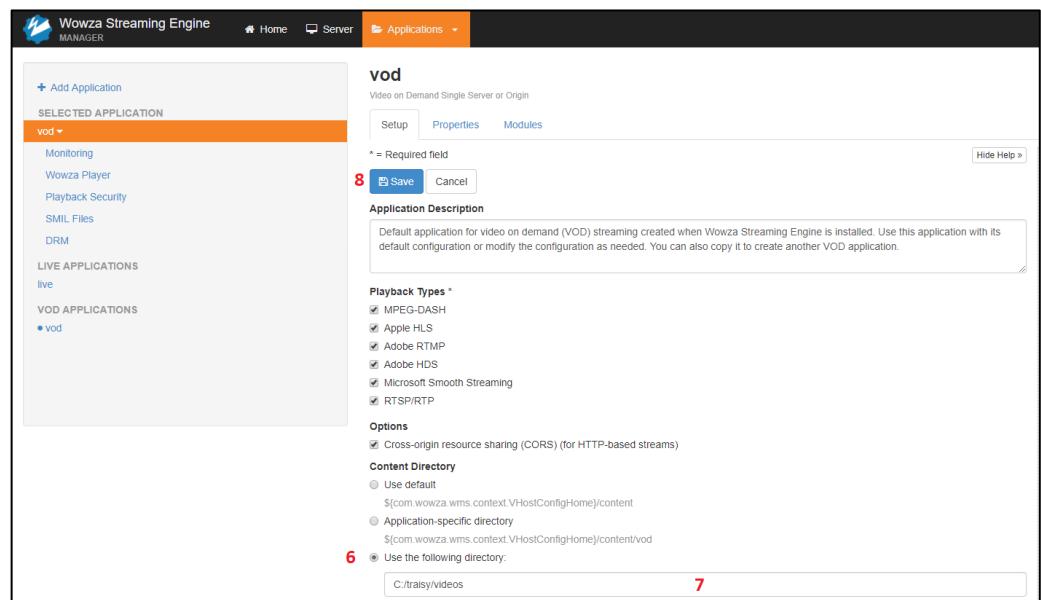


Figure 121: Setup directory - Step 6,7,8

- 6: Select "Use the following directory"
- 7: Enter "C:/traisy/videos" as the picture above
- 8: Click "Save"

Now we have finished the setup for wowza streaming server

1.2.3 Build project

Step 1: Download and install sbt for deployment at: <https://www.scala-sbt.org/download.html>

Step 2: Create Web Application deployment package by open the PowerShell at the project folder and do following steps:

Step 3: Clean the project

sbt clean

Step 4: Run this comment to create target package:

sbt dist

Step 5: Now get the package at path: [project folder path]/target/universal
For more information, please check at here:

<https://www.playframework.com/documentation/2.5.x/Deploying#using-the-dist-task>

1.2.4 Deploy project

Refer Guide for Production Configuration:

<https://www.playframework.com/documentation/2.5.x/ProductionConfiguration>

Deploy the back-end package: (the package can be found in /target/universal)

Assume that the package file name is traisy-1.0-SNAPSHOT.zip. Unzip this file. Open PowerShell, change current folder to the folder that un-zip the package and run this command:

traisy-1.0-SNAPSHOT/bin/traisy Dplay.http.secret.key=[optional_secret_key]

For those who have errors deploying the project, following this alternative way:

- For linux user: simply open shell at project folder and type

sbt start

- For window user: Open PowerShell at project folder and type

sbt run

The server is then deployed at: 127.0.0.1/9000

2. User Manual

2.1 Admin

2.1.1 Update brand

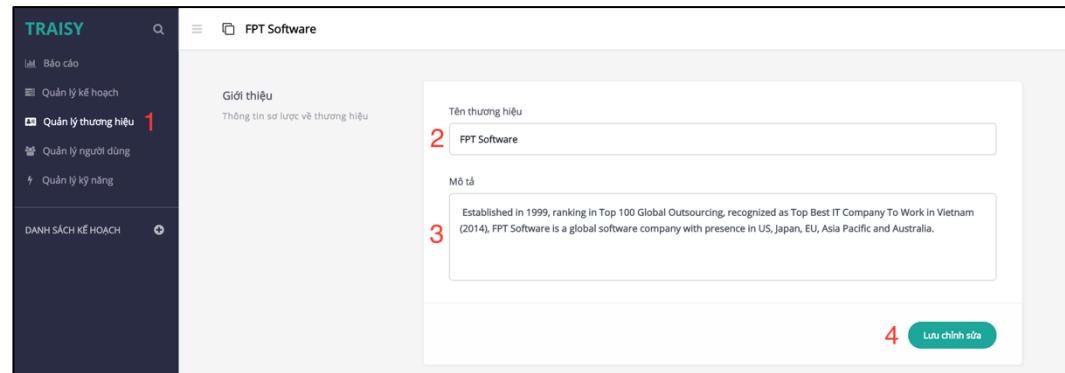


Figure 122: Update brand (1)

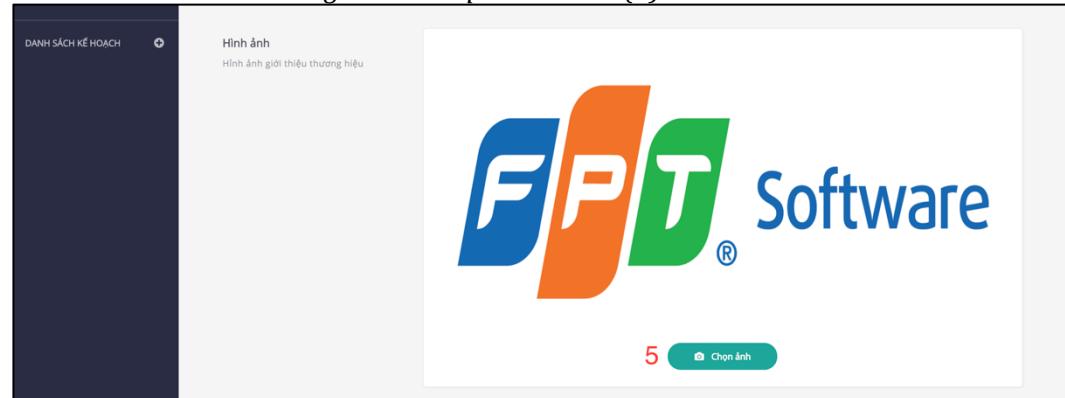


Figure 123: Update brand (2)

Step	Description
1	Click on “Quản lý thương hiệu” hyperlink
2	Fill in field: “Tên thương hiệu”
3	Fill in field: “Mô tả”
4	Click on “Lưu chỉnh sửa” button
5	Click on “Chọn ảnh” button

Table 113: User manual - Update brand

2.2 Manager

2.2.1 Create plan

Figure 124: Create plan (1)

Figure 125: Create plan (2)

Step	Description
1	Click on “Thêm kế hoạch” button
2	Fill in field: “Tên kế hoạch”
3	Click on “Thêm” button

Table 114: User manual - Create plan

2.2.2 Update plan

Figure 126: Update plan (1)

Figure 127: Update plan (2)

Figure 128: Update plan (3)

Figure 129: Update plan (4)

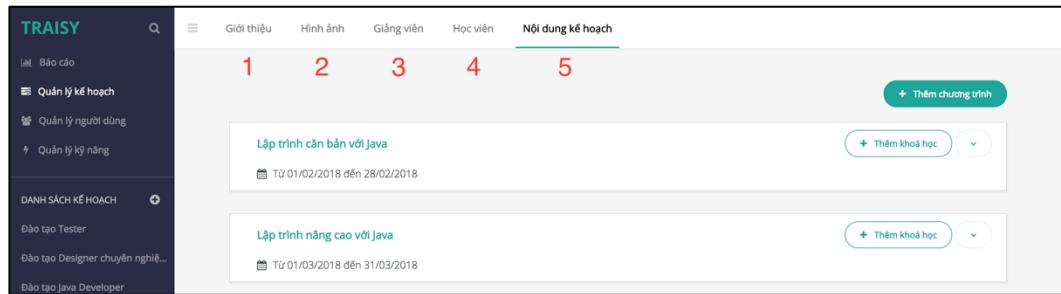


Figure 130: Update plan (5)

Step	Description
1	Click on tab “Giới thiệu”
1.1	Fill in field: “Tên kế hoạch”
1.2	Fill in field: “Thời gian”
1.3	Fill in field: “Mô tả”
1.4	Click on “Lưu chỉnh sửa”
2	Click on tab “Hình ảnh”
2.1	Click on “Chọn ảnh button”
3	Click on tab “Giảng viên”
3.1	Click on “Thêm giảng viên” button
4	Click on tab “Học viên”
4.1	Click on “Thêm học viên” button
5	Click on tab “Nội dung kế hoạch”

Table 115: User manual - Update plan

2.2.3 Create program

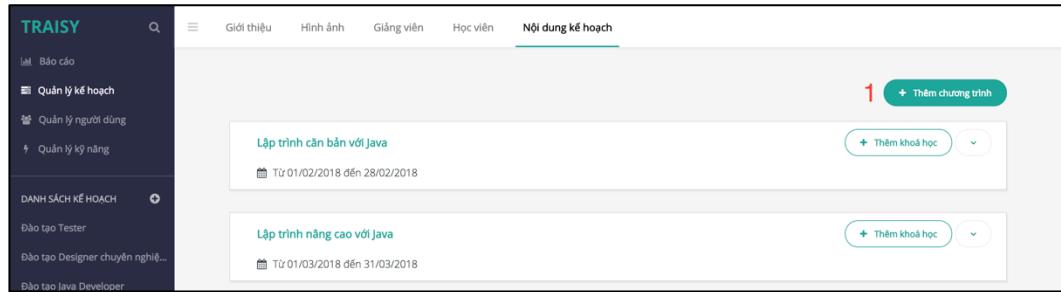


Figure 131: Create program (1)

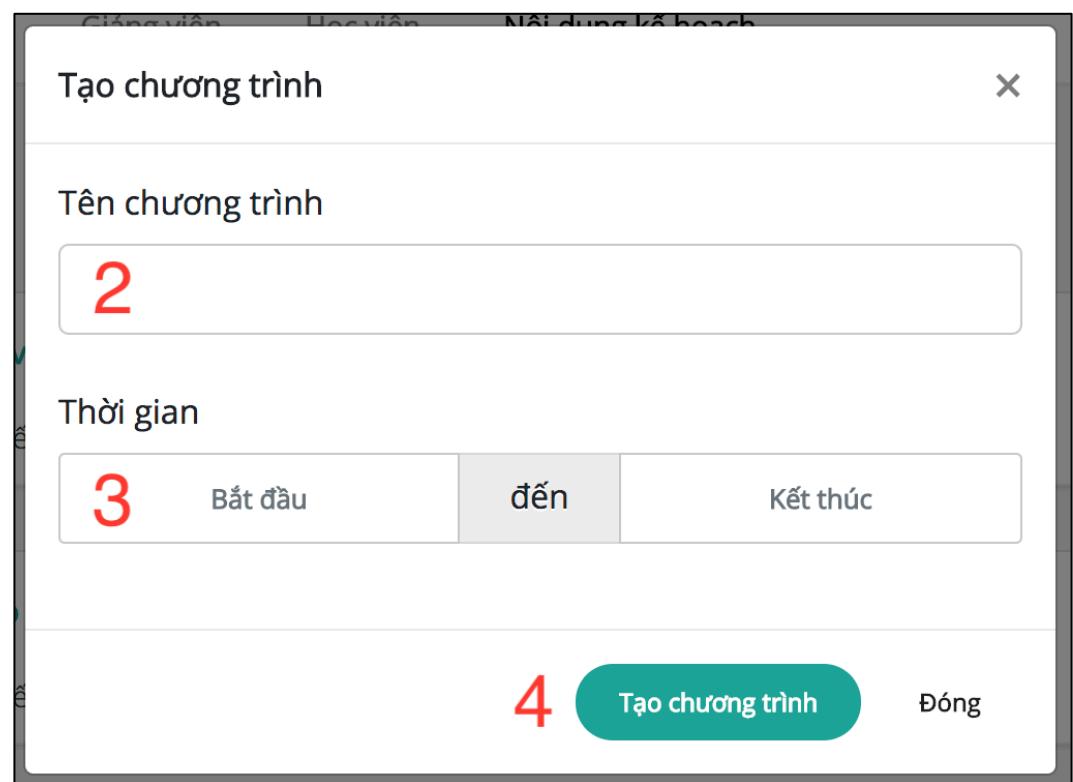


Figure 132: Create program (2)

Step	Description
1	Click on tab “Giới thiệu”
1.1	Fill in field: “Tên kế hoạch”
1.2	Fill in field: “Thời gian”
1.3	Fill in field: “Mô tả”

Table 116: User manual - Create program

2.3 Author

2.3.1 Create course

Figure 133: Create course (1)

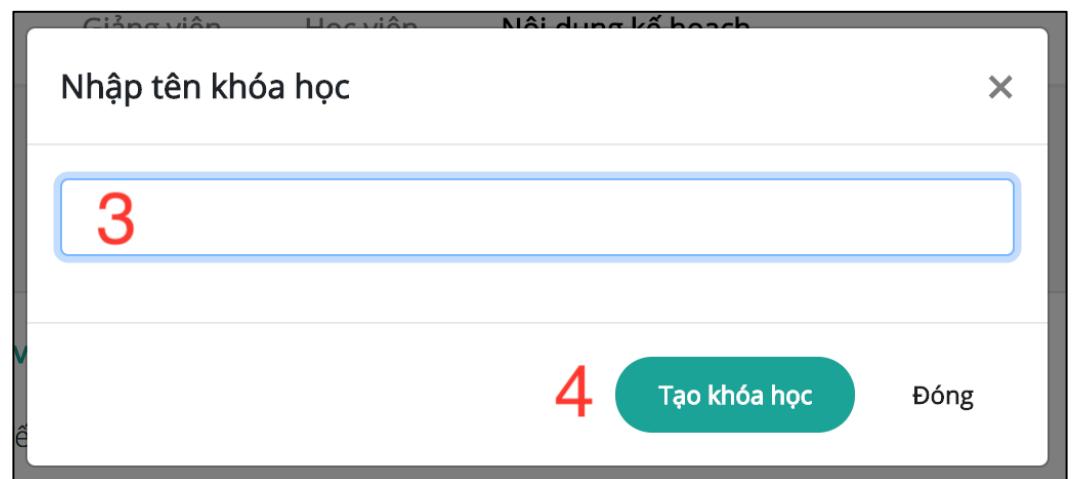


Figure 134: Create course (2)

Step	Description
1	Click on tab “Nội dung kế hoạch”
2	Click on “Thêm khoá học” button
3	Fill in field: “Tên khoá học”
4	Click on “Tạo khoá học” button

Table 117: User manual - Create course

2.3.2 Update course

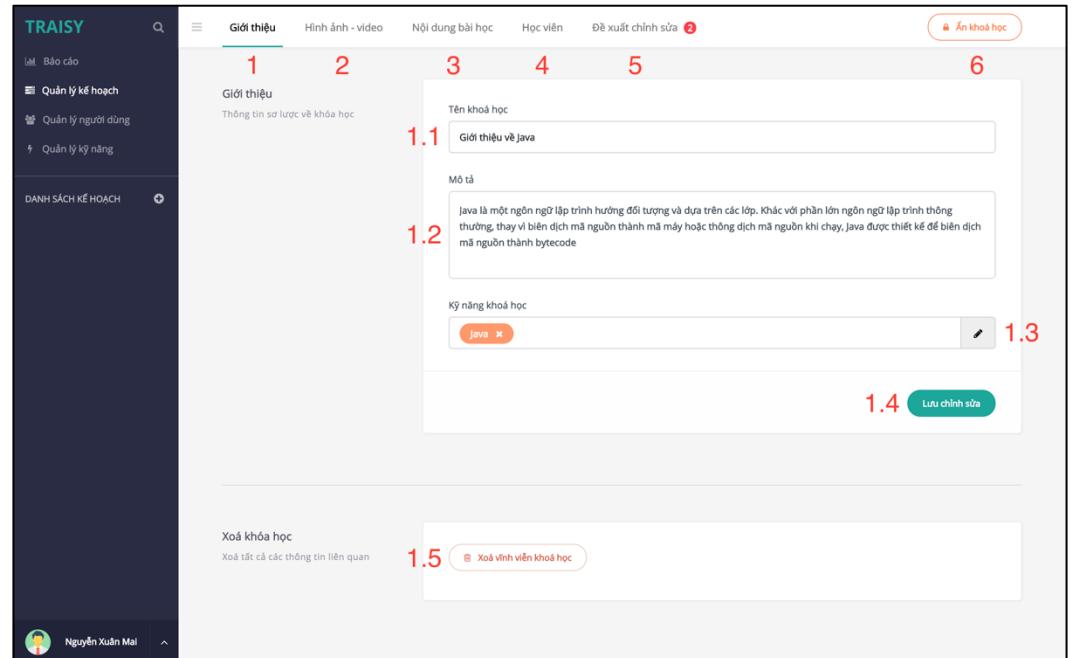


Figure 135: Update course (1)



Figure 136: Update course (2)

Step	Description
1	Click on tab “Giới thiệu”
1.1	Fill in field: “Tên khoá học”
1.2	Fill in field: “Mô tả”
1.3	Click on “Chọn kỹ năng” button
1.4	Click on “Lưu chỉnh sửa” button
1.5	Click on “Xoá vĩnh viễn khoá học” if you want to delete that course
2	Click on tab “Hình ảnh - Video”
2.1	Click on “Chọn ảnh” button
3	Click on tab “Nội dung bài học”
4	Click on tab “Học viên”
5	Click on tab “Đề xuất chỉnh sửa”
6	Click on “Ẩn khoá học” button

Table 118: User manual - Update course

2.3.2 Create quiz

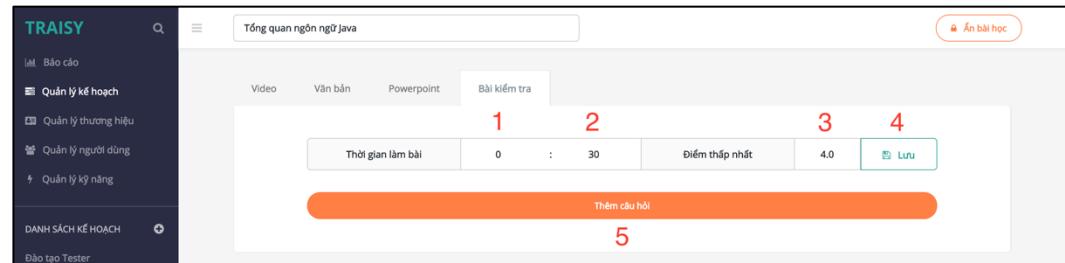


Figure 137: Create quiz (1)

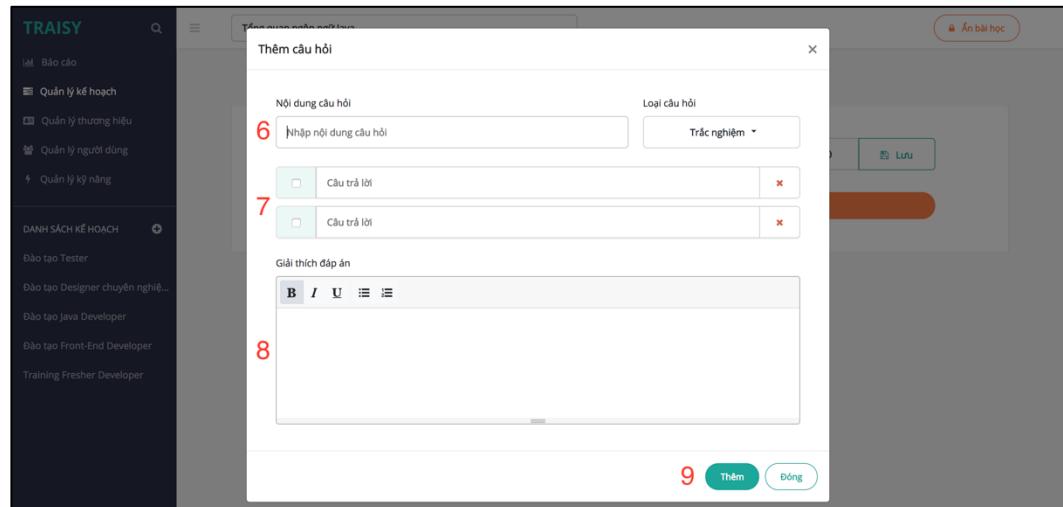


Figure 138: Create quiz (2)

Step	Description
1	Fill in minute field
2	Fill in second field
3	Fill in min score field
4	Click on “Lưu” button
5	Click on “Thêm câu hỏi” button
6	Fill in field: “Nội dung câu hỏi”
7	Fill in field: “Câu trả lời”
8	Fill in field: “Giải thích đáp án”
9	Click on “Thêm” button

Table 119: User manual - Create quiz

2.3 Learner

2.3.1 Update profile

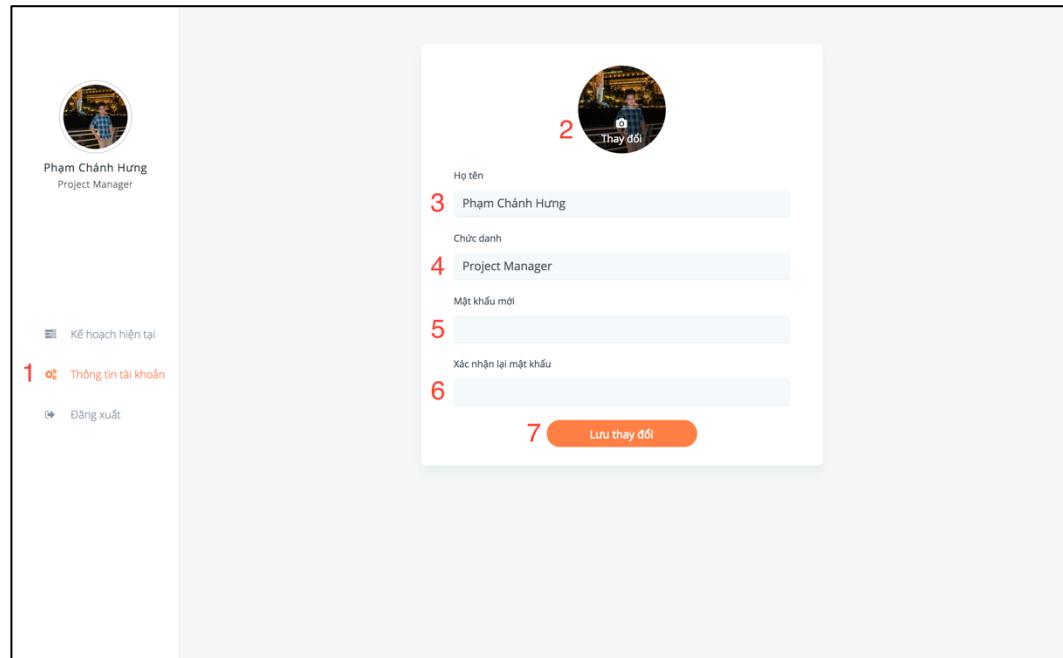


Figure 139: Update profile

Step	Description
1	Click on “Thông tin tài khoản” hyperlink
2	Click on “Thay đổi” button to change avatart
3	Fill in field “Họ tên”
4	Fill in field “Chức danh”
5	Fill in field “Mật khẩu mới” if you want to update password
6	Fill in field “Xác nhận mật khẩu” to update password
7	Click on “Lưu thay đổi” button

Table 120: User manual - Update profile

2.3.2 Learning quiz

Bài: Bài kiểm tra chương 1

1/4 Java Virtual Machine là gì?

- Là một thành phần của Java platform dùng để đọc mã bytecode trong file .class
- Là chương trình biên dịch của java dùng để biên dịch file nguồn java thành mã bytecode
- Là chương trình chạy cho java
- Tất cả các đáp án đều đúng

Kết tiếp

Figure 140: Learning quiz (1)

Số câu trả lời đúng	1/4
Tổng số điểm	2.5
Kết quả	Bạn đã vượt qua bài kiểm tra

Làm lại

Figure 141: Learning quiz (2)

Step	Description
1	Click on quiz lecture
2	Choose answer
3	Click on “Kết tiếp” button
4	Click on “Làm lại” button if you want to take quiz again

Table 121: User manual - Learning quiz