



MINISTRY OF EDUCATION AND TRAINING

# FPT UNIVERSITY

## Capstone Project Document

### ChildCare Support System

#### Group 11

**Group members** Le Duy Thanh – Team Leader – SE60939

Doan Hieu Nghia – Team Member – SE61274

Nguyen Duy Anh (Drop) – Team Member – SE61077

**Supervisor** Mr. Nguyen Huy Hung

**Ext. Supervisor** N/A

**Capstone Project code** CCSS

-Ho Chi Minh City, 10/05/2016-

*This page is intentionally left blank*

## **ACKNOWLEDGEMENTS**

We wish to thank various people for their contribution to this project: Our teachers for their advice and participation in the final review, our friends for their valuable technical support.

Special thanks should be given to Mr. Nguyen Huy Hung, our research supervisor for his professional guidance and the useful, constructive recommendations throughout the course of this project.

# Table of Contents

Table of Contents.....	1
List of Tables.....	5
List of Figures.....	5
Definitions, Acronyms, and Abbreviations.....	7
A. Introduction.....	8
1. Project Information.....	8
2. Introduction.....	8
3. Current Situation.....	8
4. Problem Definition.....	8
5. Proposed Solution.....	9
5.1. Feature functions.....	9
5.2. Advantages and disadvantages.....	9
6. Functional Requirements.....	10
7. Role and Responsibility.....	10
B. Project Management Plan (PMP).....	11
1. Problem Definition.....	11
1.1. Name of this Capstone Project.....	11
1.2. Problem Abstract.....	11
1.3. Project Overview.....	11
1.3.4. Future Plan.....	13
2. Project organization.....	14
2.1. Software Process Model.....	15
2.2. Roles and responsibilities.....	15
2.3. Tools and techniques.....	16
3. Project Management Plan.....	17
3.1. Software development life cycle.....	17
3.2. Phase Detail.....	18
3.3. All Meeting Minutes.....	19
4. Coding Convention.....	19
C. Software Requirement Specification.....	21
1. User Requirement Specification.....	21

2.	System Requirement Specification.....	21
2.1.	System Overview Use Case.....	21
2.2.	List of Use Case.....	22
3.	Software system Attribute.....	128
<b>3.5.</b>	<b>Maintainability.....</b>	<b>129</b>
-	The system should be divided into modules and code should be easy to maintain and extend.....	129
<b>3.6.</b>	<b>Portability.....</b>	<b>129</b>
-	Website can be access everywhere which have internet and web browser.....	129
<b>3.7.</b>	<b>Performance.....</b>	<b>129</b>
	All requests should be response in no more than 30s.....	129
<b>4.</b>	<b>Conceptual Diagram.....</b>	<b>130</b>
D.	Software Design Description.....	132
1.	Design overview.....	132
2.	System Architectural Design.....	133
3.	Component Diagram.....	134
4.	Detailed Description of Components.....	135
4.1.	Class Diagram.....	135
4.2.	Class Diagram Description.....	136
4.2.1.	Account.....	136
4.2.2.	ChildProfile.....	137
4.2.3.	ChildImmunization.....	138
4.2.4.	ImmunizationSchedule.....	139
4.2.5.	Immunization.....	140
4.2.6.	ChildHealth.....	141
4.2.7.	HealthRate.....	142
4.2.8.	NutritionContent.....	143
4.2.9.	NutritionAmount.....	144
4.2.10.	Nutrition.....	145
4.2.11.	Class.....	145
4.2.12.	AccountClass.....	146
4.2.13.	ChildClass.....	147

Attribute.....	147
4.2.14. Attendance.....	148
4.2.15. Report.....	148
4.2.16. Report comment.....	150
4.2.17. ClassGrade.....	150
4.2.18. Activity.....	151
4.2.19. Menu.....	152
4.2.20. MenuSchedule.....	152
4.2.21. Post.....	153
4.2.22. Notification.....	153
4.2.23. AbnormalCase.....	154
4.3. Sequence Diagram.....	156
4.3.1. <User> Login.....	156
4.3.2. <System Admin/Operation Admin> Create parent/teacher.....	157
4.3.3. <Parent> Create child profile.....	158
4.3.5. <Parent> Edit child's immunization.....	160
4.3.6. <Parent> View child's health.....	161
4.3.7. <Parent> View Study History.....	162
4.3.8. <Parent> View child's report.....	163
4.3.9. <Teacher> View class detail.....	164
4.3.10. <Teacher> Check Attendance.....	165
4.3.11. <Teacher>Health Check.....	166
5. Database Designed.....	167
5.1. Entity Relationship Diagram.....	167
5.2. Entity Dictionary.....	168
E. System Implementation & Test.....	169
1. Introduction.....	169
1.1. Overview.....	169
1.2. Test Approach.....	169
2. Database Relationship.....	170
2.1. Database Physical Diagram.....	170
2.2. Data Dictionary.....	171

2.2.1.	Table Data Dictionary.....	171
2.2.2.	Attribute Data Dictionary.....	172
F.	Software User's Manual.....	193
1.	Installation Guide.....	193
1.1	Setting up Environment at Server Side.....	193
1.2	Deployment at Server Side.....	193
1.3	Setting up the Environment at Client Side.....	196
2.	User Guide.....	197
2.1.	Operation Adminitrator Screens.....	197
2.1.1.	Login screen for operation administrator.....	197
2.1.2.	Manage User Account Screens.....	197
2.1.2.1.	Manage Parent Account Screen.....	197
2.1.2.2.	Manage Teacher Account Screen.....	201
2.2.	System Adminitrator Screens.....	204
2.2.1.	Login screen for system administrator.....	204
2.2.2.	Manage User Account.....	205
2.2.2.1.	Manage Operation Administrator Account.....	205
2.2.3.	Class management screen.....	208
2.3.	Parent Screens.....	209
2.3.1.	Parent Login Screen.....	209
2.3.2.	Children List Screen.....	211
2.3.3.	Create New Child Profile Screen.....	212
2.3.4.	Child Profile Detail Screen.....	213
	.....	213
2.4.	Teacher Screens.....	215
2.4.1.	Teaching schedule screen.....	215
2.4.2.	Class Detail Screen.....	217
2.4.3.	Check attendance screen.....	219
2.4.4.	Child health screen.....	220
2.4.5.	Input child health data screen.....	221
2.4.6.	Create child report screen.....	222
G.	Appendix.....	225

---

1. Scrum model:.....	225
2. Coding Convention:.....	225
3. Data:.....	225
4. Design:.....	225
5. Store project data:.....	225
6. Media Server.....	225

## List of Table

Table 1: Definitions.....	7
Table 2: Roles and Responsibilities.....	10
Table 3: Hardware Requirement for Server.....	13

## List of Figures

Figure 1: Scrum Development Model.....	15
Figure 2: System overview usecase.....	21
Figure 3: List of usecase.....	22
Figure 4: Conceptual Diagram.....	130
Figure 5: System Architecture.....	133
Figure 6: Component diagram.....	134
Figure 7: Class diagram.....	135
Figure 8: User login sequence diagram.....	156
Figure 9: Create user account sequence diagram.....	157
Figure 10: Create child profile sequence diagram.....	158
Figure 11: Edit child profile sequence diagram.....	159
Figure 12: Edit child immunization sequence diagram.....	160
Figure 13: View child health sequence diagram.....	161
Figure 14: View study history sequence diagram.....	162
Figure 15: View child report sequence diagram.....	163
Figure 16: View class detail sequence diagram.....	164
Figure 17: Check attendance sequence diagram.....	165
Figure 18: Health check sequence diagram.....	166
Figure 19: Entity Relationship Diagram.....	167
Figure 20: Database Physical Diagram.....	170
Figure 21: Prepare deployment package for web application.....	193
Figure 22: Database deployment.....	194
Figure 23: Deploy website in server.....	195
Figure 24: Deploy website in server.....	195
Figure 25: Operation Administrator Login Screen.....	197

Figure 26: Manage Parent Account Screen.....	198
Figure 27: Create new parent account pop-up.....	199
Figure 28: Edit parent account pop-up.....	200
Figure 29: Manage Teacher Account Screen.....	201
Figure 30: Create new teacher account pop-up.....	202
Figure 31: Edit teacher account pop-up.....	203
Figure 32: System administrator login screen.....	204
Figure 33: Operation Administrator Management Screen.....	205
Figure 34: Create new operation administrator account pop-up.....	206
Figure 35: Edit operation administrator account pop-up.....	207
Figure 36: Class management screen.....	208
Figure 37: Class management screen.....	209
Figure 38: Parent login pop-up.....	210
Figure 39: Paren login screen.....	210
Figure 40: Children List Screen.....	211
Figure 41: Create new child profile screen.....	212
Figure 42: Child profile detail screen.....	213
Figure 43: Teaching schedule screen.....	215
Figure 44: Teaching schedule – Class list tab.....	216
Figure 45: Class detail screen.....	217
Figure 46: Check attendance screen.....	219
Figure 47: Child health screen.....	220
Figure 48: Input data child health screen.....	221
Figure 49: Create child report screen.....	222
Figure 50: Create child report popup.....	223

## **Definitions, Acronyms, and Abbreviations**

Name	Definition
CCSS	ChildCare Support System
CCSSProjectPage	ChildCare Support System Project Page
PMP	Project Management Plan
GUI	Graphic User Interface
ERD	Entity Relationship Diagram

**Table 1: Definitions**

## A. Introduction

### 1. Project Information

- Project name: **ChildCare Support System**
- Project Code: **CCSS**
- Product Type: **Website & Mobile Application (Android)**
- Start Date: **May 10<sup>th</sup>, 2016**
- End Date: **August 18<sup>th</sup>, 2016**

### 2. Introduction

Nowaday, any parent is always wanted to create the best conditions for their children. But they do not always have time with their children. Therefore they must send their children to the kindergarten.

In addition, when a child didn't want to eat or cry many at kindergarten, the teacher gets angry and has some inappropriate behavior with the children. Violence against children also a matter of concern.

Another problem is immunization schedule, they were managed by manual so easily lost and forgotten.

As well as the monitoring of nutrition, usually the information from parents and schools are often inconsistent. This can influence child development and improved plan.

"Childcare Support System" help parents are easier to take care about their children. CCSS can allow parents to see their children's activities during the day through camera. Besides that, management base on child's profile CCSS can tracking which child upcoming immunization schedule and notify through mobile application. This system also help teacher create report about children.

### 3. Current Situation

The majority of current kindergarten offers parents an account to watch camera at home or work. Activities schedule or menu will often post in the school site every week. Immunization records almost hold by parents.

### 4. Problem Definition

- There is no reminding of the child immunization schedule, based primarily on the parents and teachers of children cooperation.
- Parents do not know clearly about the health situation of children and do not have nutrition suggestion.
- Keeping children in kindergarten can also cause worry for the child's parents if the child information about preferences, allergies, immunization ... inconsistency between families and schools.

## **5. Proposed Solution**

"Childcare Support System" which use a children profile (parents do it themselves) to tracking activities of the children, notified instantly, view report every semester, track and remind parents immunization schedule, nutrition suggestion base on periodic medical.

### **5.1. Feature functions**

We will build a web application and a mobile application:

- Profile: Parents can create and view their children profile.
- Check attendance: check children in and out in classroom. Parents get notified instantly of the event on their mobile.
- Visual concerns: allow parents to see their children's activities during the day through photo, camera.
- Report: parents can view report about their children: food, play, drop-out....
- Immunization: keep track and remind parents immunization schedules
- Nutrition suggestion: based on children profile, report and suggest parents regarding the current status (weight, height ...), improved plans....

### **5.2. Advantages and disadvantages**

The advantages and disadvantages of the proposed solution:

- Advantages:
  - The parents and the kindergarten can keep track of the entire child's information in a uniform manner.
  - The parents and directly teacher of the child will be reminding about child's immunization schedules.
  - The child will have a right nutrition suggestion which based on children profile.
  - Parents can observe their children when they are home or work.
  - Some emergency case, the child's teacher may notice a detailed and fastest way state child for parents.
- Disadvantages:
  - Just focus on objects as small children from 5 years old to 5 years old.
  - Devices need connect to the internet.

## **6. Functional Requirements**

All of Function requirements are listed as below:

- Admin:
  - Manage immunization schedules base on Standard immunization system.

- Manage system for child nutrition.
- Manage user in system: parent account, teacher account and child profile.
- Manage weekly menu and outdoor activities for children.
- Manage teaching scheduler.
- Parent:
  - Create, update and view their child profile.
  - Observe their children in anywhere.
  - View teaching scheduler.
  - View teacher profile.
  - Get notification about immunization or some emergency case from kindergarten.
  - Get report about their child activity in kindergarten.
  - Get suggestion about nutrition of children.
- Teacher:
  - View teaching scheduler.
  - View child profile.
  - Check attendance of each child.
  - Create report for each child in each period about height, weight, weekly menu...
  - Make notify to child's parent.
  - Get notification about immunization.
  - Get suggestion about nutrition of children.
  - Get notified of some note about the child such as allergies.

## 7. Role and Responsibility

No	Full Name	Role	Position	Contact
1	Nguyen Huy Hung	Scrum Master/ Product Owner	Instructor	<a href="mailto:hunghn@fpt.edu.vn">hunghn@fpt.edu.vn</a>
2	Le Duy Thanh	Developer	Leader	<a href="mailto:thanhldse60939@fpt.edu.vn">thanhldse60939@fpt.edu.vn</a>
3	Doan Hieu Nghia	Developer	Member	<a href="mailto:nghiadhse61274@fpt.edu.vn">nghiadhse61274@fpt.edu.vn</a>
4	Nguyen Duy Anh (Drop)	Developer	Member	<a href="mailto:anhndse61077@fpt.edu.vn">anhndse61077@fpt.edu.vn</a>

**Table 2: Roles and Responsibilities**

## B. Project Management Plan (PMP)

### 1. Problem Definition

#### 1.1. Name of this Capstone Project

- Official name: ChildCare Support System
- Vietnamese name: Hệ Thống Hỗ Trợ Chăm Sóc Trẻ Em
- Abbreviation: CCSS

#### 1.2. Problem Abstract

To solve those problems, which mentioned above, we provide a system that help parents easy to manage their children at school. The system includes real time camera system by using IP camera, children profile to manage: reports, immunization, nutrition... and different manage functions: manage account, manage profile, and manage immunization...

#### 1.3. Project Overview

### 2. Current situation

The problems encountered in this project:

- Disadvantages:
  - Knowledge: some parents not expert in computer and mobile devices.
  - Child nutrition: child from 3 years old to 5 years old are the ages developing all aspects and sensitive. All nutrition suggestion should consult doctor. Difficult to build an automatic system.

### 3. The Proposed System

According to actual needs, our system will provide to parents an account. Parents sign in and create profile for their children. Parent can input information even so follow report from teacher, immunization schedule, nutrition suggestion base on health tracking periodic. In Addition, they also can view camera at school, weekly menu, outdoor activity schedule, etc.

Our system includes two main subsystems:

#### 1.3.2.1 Web Site

Website is common place that have all function of CCSS system:

- Admin:
  - Manage account: manage teacher account and parents account.

- Manage outdoor activity schedule: outdoor activity schedule will posted by admin.
- Manage weekly menu: weekly menu will posted by admin.
- Manage immunization schedule: immunization schedule managed by admin from standard immunization schedule of Ministry of Health.
- Manage class: class schedule managed by admin.
- Parents:
  - Create child profile: create child profile and input information of children.
  - View camera: view camera real time.
  - View report: view report of children make by teacher.
  - Immunization tracking: input injections that children have done and schedule for next injections.
  - Health tracking: follow health tracking and nutrition suggestion
- Teacher:
  - Check attendance: check attendance for children every days.
  - Make report: teacher make report for each child.
  - View camera: view camera real time.
  - View child profile: view child profile.

### **1.3.2.2 Mobile application**

Majority of functions similar web application, some additional function:

- Teacher:
  - Stream video real time: when have outdoor activity, teacher can stream video real time.

## **4. Boundaries of the System**

The system can be used by everyone with a smart phone which was connected to Internet.

The language of the system is Vietnamese.

The complete product includes:

- The website for admin, school staff to interact with the system.
- The website for parents to interact with the child profile.
- The website for teachers to manages classes and follows each child in class.

All the process involved document.

#### 4.1.1. Future Plan

Currently, the system only deploy on a single platform: Android. Besides that, the system has not store video record. We design the system to make it easy to scale to 2 models:

- Run on multiple platforms on client side: iOS, Windows phone.
- Create function to store video record.

### 5. Development Environment

#### 1.3.5.1 Hardware requirements

- For server / web development

Windows	Minimum requirements	Recommended
<b>Internet Connection</b>	Cable, Wi-fi (4Mbps)	Cable, Wi-fi (8Mbps)
<b>Operating System</b>	Ubuntu Server 12 LTS	Ubuntu Server 14.04.2 LTS
<b>Computer Processor</b>	Intel® CORE i3 Quad core 2.1 GHz	Intel® CORE i7 Quad core 2.4 GHz
<b>Computer Memory</b>	2GB RAM	4GB or more

**Table 3:** Hardware Requirement for Server

- For mobile development

Android	Minimum requirements	Recommended
<b>Internet Connection</b>	Wi-fi (4Mbps)	Wi-fi (8Mbps)
<b>Operating System</b>	Android 4.4 Kitkat	Android 6.0 Marshmallow
<b>Processor</b>	Snapdragon 410 1.2GHz Quad Core	Snapdragon 615 1.5GHz Octa-core or higher
<b>Memory</b>	1GB RAM	2GB RAM or more

**Table 4:** Hardware Requirement for client (Mobile device) development

#### 1.3.5.1 Software requirements

	Name / Version	Description
<b>Environment</b>	AngularJS 1.5.x	Specification for developing web application
<b>Modeling tool</b>		
<b>IDE</b>	IntelliJ 2016.1.2	Programming tools
<b>DBMS</b>	MySQL Workbench 5.6	Used to create & manage the database for system
<b>Source control</b>	SmartGit 7.1.3	Used for source control
<b>Web browser</b>	Chrome 50 or above	Testing browser

**Table 5:** Software requirements for develop web site

	Name / Version	Description
--	----------------	-------------

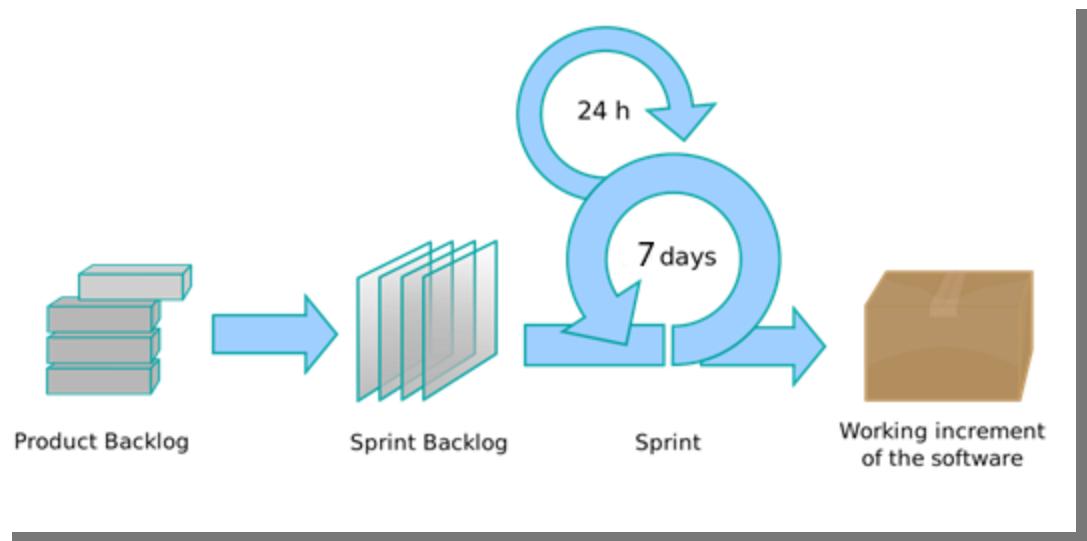
<b>IDE</b>	Android Studio 1.5.1	Programming tools
<b>SDK</b>		
<b>Source Control</b>	SmartGit 7.1.3	Used for source control
<b>Testing OS</b>	Android 5.1.1; Lollipop	Testing Client Operation System

**Table 6: Software requirements for develop client application**

## 2. Project organization

## 2.1. Software Process Model

Project is developed under Scrum model.



**Figure 1: Scrum Development Model**

For more information: [http://en.wikipedia.org/wiki/Scrum\\_%28software\\_development%29](http://en.wikipedia.org/wiki/Scrum_%28software_development%29)  
<https://www.scrum.org/>

## 2.2. Roles and responsibilities

No	Full name	Role in group	Responsibilities
1	Nguyen Huy Hung	Project manager	<ul style="list-style-type: none"><li>• Specify user requirement</li><li>• Specifying the business</li><li>• Control the development process</li><li>• Give advices on techniques, solutions and business analysis support</li></ul>
2	Le Duy Thanh	Team Leader, BA, DEV, Tester	<ul style="list-style-type: none"><li>• Managing process</li><li>• Clarifying requirements</li><li>• Researching solutions and techniques</li><li>• Assigning task for members</li><li>• Design architecture</li><li>• Support team members</li><li>• Reviewing the task result of members</li><li>• Creating/ Editing documents and reports</li><li>• Reviewing documents and reports</li><li>• Coding Web</li><li>• Designing database</li></ul>

			<ul style="list-style-type: none"> <li>• Creating test plan</li> <li>• Creating test case</li> <li>• Testing</li> </ul>
3	Doan Hieu Nghia	Team Member, BA, DEV, Tester	<ul style="list-style-type: none"> <li>• Clarifying requirements</li> <li>• Researching solutions and techniques</li> <li>• Creating/ Editing documents and report</li> <li>• Reviewing documents and reports</li> <li>• Coding Web</li> <li>• Coding Mobile</li> <li>• Reviewing test plan</li> <li>• Reviewing test case</li> <li>• Testing</li> </ul>
4	Nguyen Duy Anh (Drop)	Team Member, BA, DEV, Tester	<ul style="list-style-type: none"> <li>• Clarifying requirements</li> <li>• Researching solutions and techniques</li> <li>• Creating/ Editing documents and report</li> <li>• Reviewing documents and reports</li> <li>• Designing database</li> <li>• Coding Web</li> <li>• Coding Mobile</li> <li>• Reviewing test plan</li> <li>• Reviewing test case</li> <li>• Testing</li> </ul>

**Table 7: Roles and Responsibilities Details**

### 2.3. Tools and techniques

Tool / Technique	Name / version
Frontend IDE	IntelliJ 2016.1.2, Android Studio 1.5.1
Technology	AngularJS 1.5.x, Spring boot 1.3.5 GA
Managing database	My SQL Workbench 5.6
Managing the project	SmartGit 7.1.3
Managing documents, reports, models and diagrams	Microsoft Office 2013

**Table 8: Tools and Techniques**

## 3. Project Management Plan

### 3.1. Software development life cycle

Phase	Description	Deliverables	Resource needed	Dependencies and Constraints	Risks
Specification	Identify and define system spec in	-Introduction of proposed	20 mandays	N/A	- Lack of member share

	general	system. -General software requirement specification.			of understands. - Lack of experience.
<b>Development</b>	-Design the current architecture -Choose technology - Implement module	-Task plan -Software design document -Technology notes - Actual software of each module	60 man-days	Base on specification	- Lack of experience. - Code dose not work.
<b>Validation</b>	- Integrate modules of system. - Release the version. - Create test case. - Test the version. - Note changes.	- Actual software of the whole system. - Test case. - Changes log / notes.	20 man days	Depend on software of each module	- Modules can't connect with others. - Test case doesn't cover all core functions.

**Table 9: Software Development Life Cycle Detail**

If the result of current version in validation phrase is not satisfied, loop the process for the next version until result of the version is approved.

### 3.2. Phase Detail

#### 3.2.1 Specification

Task	Description	Author
<b>Identify and define system spec in general.</b>	Define which main functions system should provide.	ThanhLD, NghiaDH, AnhND

**Table 10: Phase 1: Specification**

#### 3.2.2 Development

Task	Description	Author
<b>Design the current architecture</b>	Design the architecture for the current system base on current definition of specification.	ThanhLD, NghiaDH, AnhND
<b>Choose technology</b>	Choose technology to implement the current system	ThanhLD, NghiaDH, AnhND
<b>Implement modules</b>	Implement modules base the designs and chosen technology	ThanhLD, NghiaDH, AnhND

**Table 11: Phase 2: Implementation**

### 3.2.3 Validation

Task	Description	Author
<b>Integrate all modules of the system</b>	Integrate all separate modules	ThanhLD, NghiaDH, AnhND
<b>Release the version</b>	Release a version after integrate all modules into a system	ThanhLD, NghiaDH, AnhND
<b>Create test case</b>	Create test case base current specification which was determined in Specification phrase	ThanhLD, NghiaDH, AnhND
<b>Test the version</b>	Execute the created test case	ThanhLD, NghiaDH, AnhND
<b>Note changes</b>	Note the changes in changes log for the next version.	ThanhLD, NghiaDH, AnhND

**Table 12: Phase 3: Validation**

### 3.3. All Meeting Minutes

## 4. Coding Convention

- AngularJS: Using to develop UI of website.  
Javascript coding conventions:
  - Naming conventions
    - Always use the same naming convention for all your code
    - Hyphens in HTML and CSS
    - camelCase is used by JavaScript itself, by jQuery, and other JavaScript libraries
  - Variable Names
    - We use **camelCase** for identifier names (variables and functions).
    - All names start with a **letter**.
  - Spaces Around Operators
    - Always put spaces around operators (= + - \* /), and after commas.
  - Code Indentation
    - Always use 4 spaces for indentation of code blocks
  - Statement Rules
    - Always end a simple statement with a semicolon.
- Spring boot: Using to develop rest API server.  
Java coding conventions:
  - Naming Conventions:
    - Variable name should be short yet meaningful. If the name is more than one word, it must be in mixed case, starting word with a lowercase.
    - Constants name should be all uppercase with words separated by underscores.

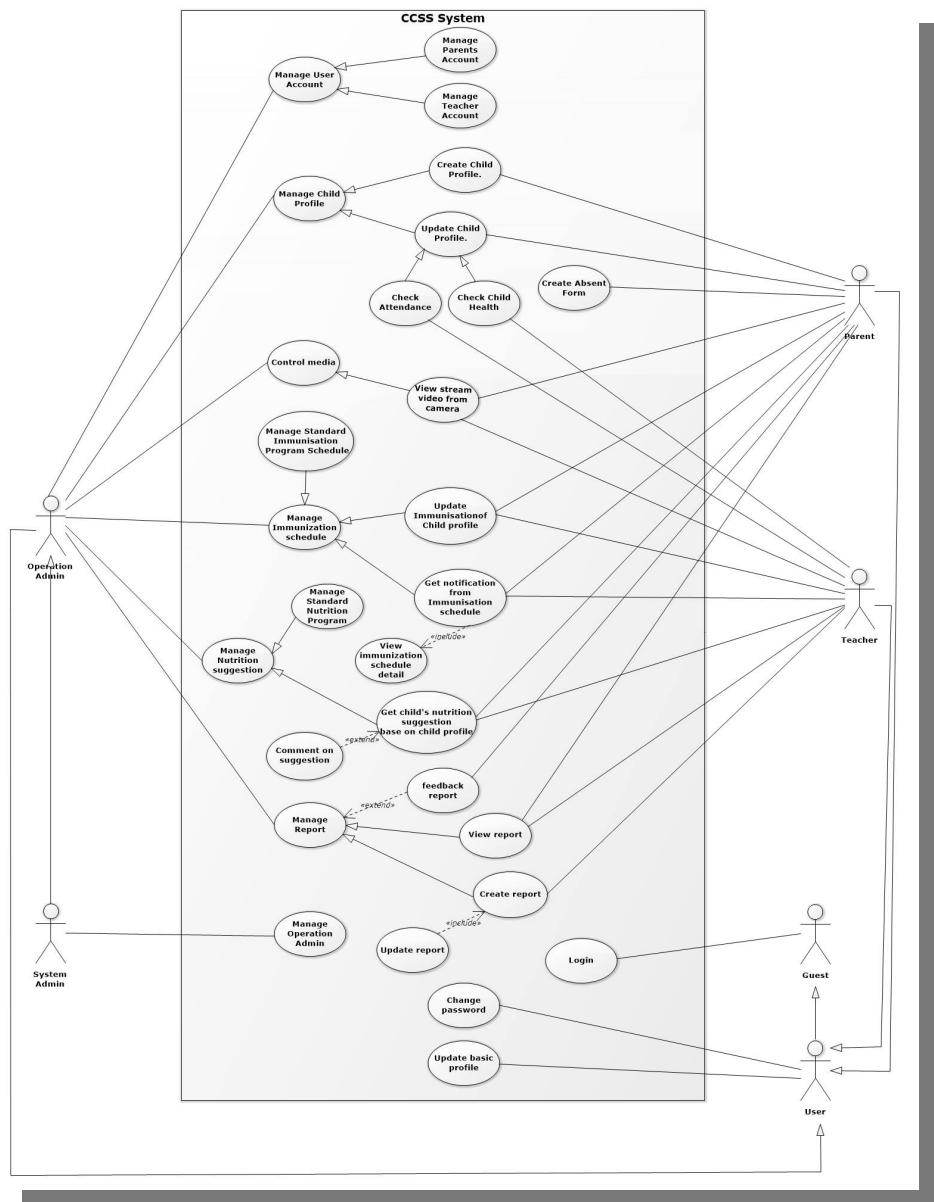
- Methods name should be verbs, in mixed case with the first word lowercase, the first letter of each internal word capitalized.
- Class name should be nouns, in mixed case with the first letter of each internal word capitalized.
- Package and import statements:
  - Package statement is the first non-comment line. -Import statement is after package statement.
- Constants
  - Numerical constants should not be coded directly.
- Variable Assignments:
  - Avoid assigning several variables to the same value in a single statement.
- Comments:
  - Using /\* \*/ for block comments
  - Using // for line comments
- Return Statements:
  - A return statement with a value should not use parentheses.

### **c. Software Requirement Specification**

## **1. User Requirement Specification**

## 2. System Requirement Specification

## 2.1. System Overview Use Case



**Figure 2: System overview usecase**

## 2.2. List of Use Case

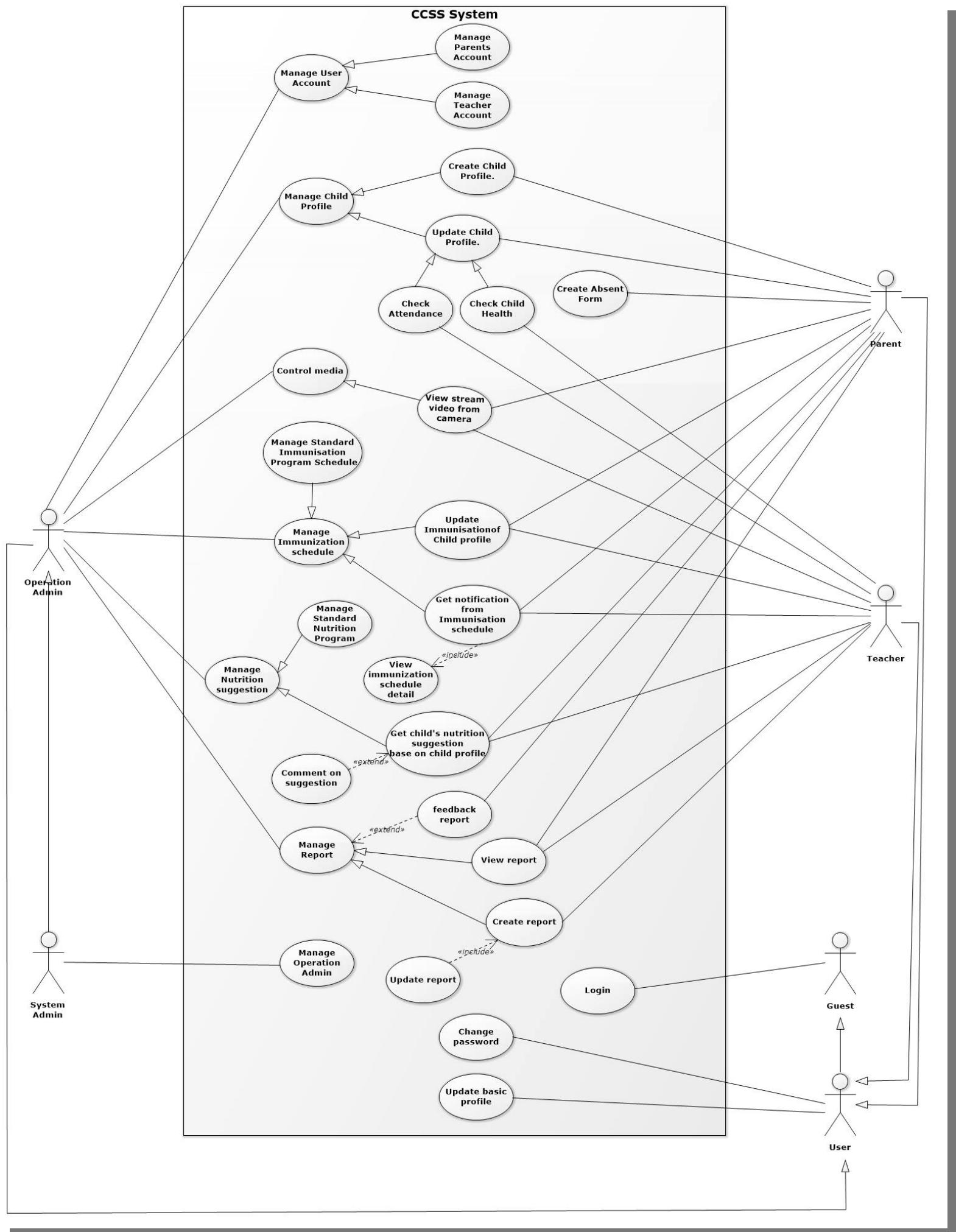
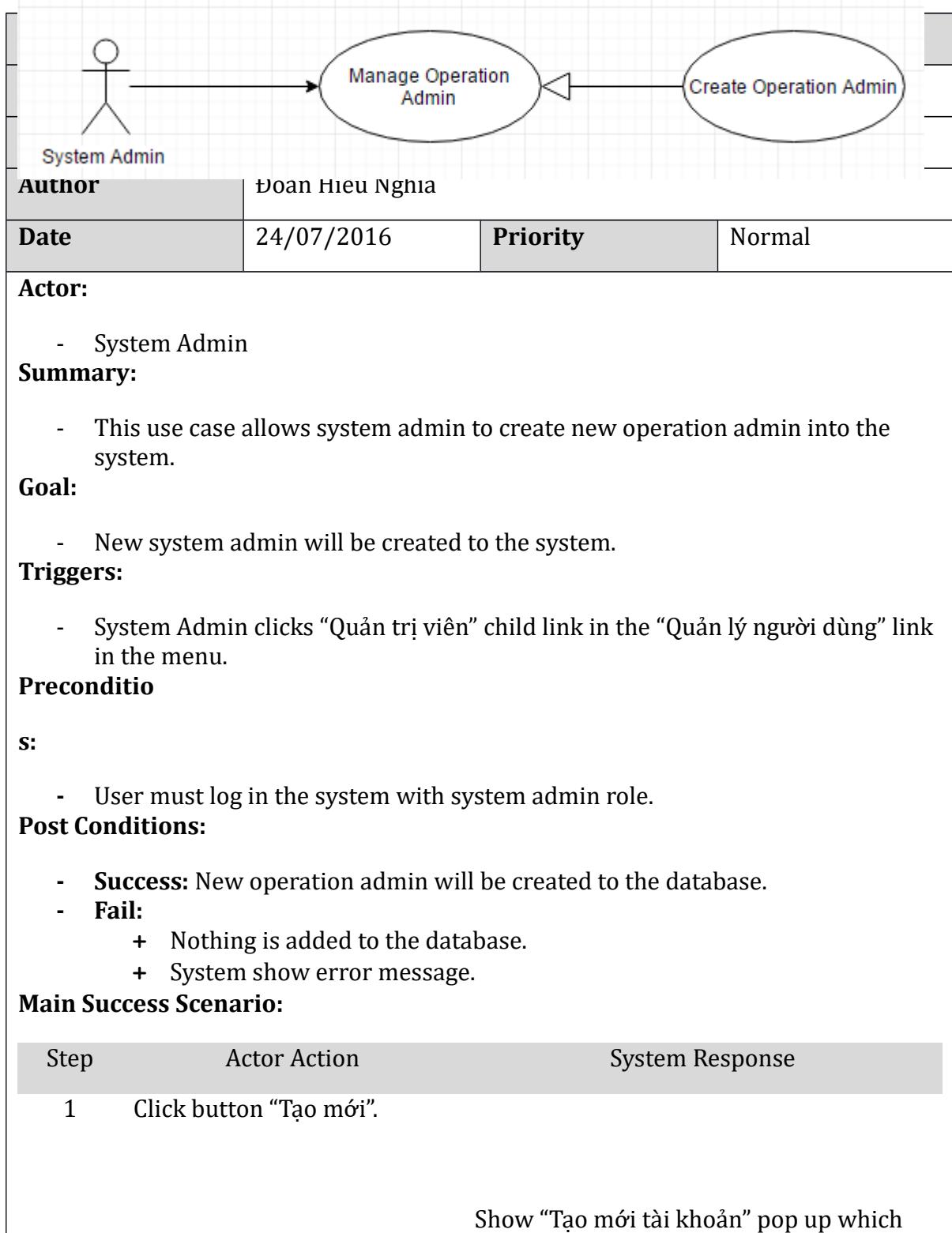


Figure 3: List of usercase

## 2.2.1.<System Admin & Operation Admin> Overview Use Cases

### 2.2.1.1. <System Admin> Create Operation Admin



contains:

- "Email": textbox, must allow email format, required.
- "Mật khẩu": textbox, min length = 6, required.
- "Xác nhận mật khẩu": textbox, have to equals with Mật khẩu, required.
- "Họ và tên": textbox, required.
- "Ngày sinh": textbox, date time, min: 01/01/1950, max: 01/01/2000.
- "Giới tính": radio-button(Nam, nữ & khác)
- "Số điện thoại": textbox, min length: 10, max length: 11, number only, required.
- "Số điện thoại 2": textbox, min length: 10, max length: 11, number only.
- "Địa chỉ": textbox, required.
- "Địa chỉ 2": textbox.
- "Tạo": button.
- "Hủy": button

- 2 System Admin fills data to the form.
- 3 System Admin clicks "Lưu" button. [Alternative 1], [Exception 1, 2, 3, 4, 5]

Insert new operation admin to the database.

#### Alternative Scenario:

No	Actor Action	System Response
1	System Admin clicks "Hủy" button	System closes the current popup.

**Exceptions:**

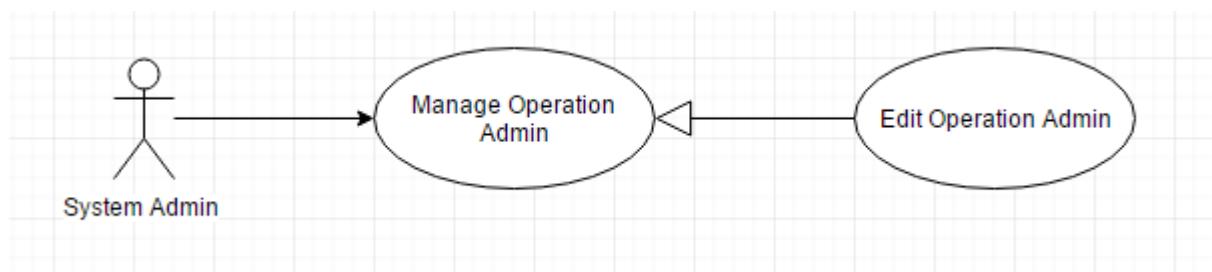
No	Actor Action	System Response
1	Length of “Mật khẩu” is not in range.	Show error message: “Mật khẩu phải từ 6 – 25 ký tự!”
2	Data of “Mật khẩu” and “Xác nhận mật khẩu” is not equals	Show error message: “Xác nhận mật khẩu và mật khẩu chưa trùng khớp”
3	Data of “Ngày sinh” is not in range.	Show error message: “Ngày sinh từ 01/01/1950 đến 01/01/2000!”
4	Length of “Số điện thoại” or “Số điện thoại 2” is not in range.	Show error message: “Số điện thoại phải gồm 10 hoặc 11 số!”
5	Data of “Số điện thoại” or “Số điện thoại 2” is not only number.	Show error message: “Số điện thoại chỉ bao gồm con số!”

**Relationships:** Manage Operation Admin

**Business Rules:**

- Each added operation admin must be in active state when insert to database.

### 2.2.1.2. <System Admin> Edit Information of Operation Admin



**USE CASE - UC 02**

<b>Use Case No.</b>	UC02	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Edit Operation Admin		
<b>Author</b>	Đoàn Hiếu Nghĩa		

Date	24/07/2016	Priority	Normal
<b>Actor:</b>			
- System Admin			
<b>Summary:</b>			
- This use case allows system admin to edit information of an existing operation admin in the system.			
<b>Goal:</b>			
- Information of chosen operation admin will be updated to the system.			
<b>Triggers:</b>			
- System Admin clicks “Sửa” button of the chosen operation admin on admin list in the “Quản trị viên” page.			
<b>Preconditions:</b>			
- User must log in the system with system admin role.			
<b>Post Conditions:</b>			
- <b>Success:</b> Information of the chosen operation admin will be updated to the database.			
- <b>Fail:</b> <ul style="list-style-type: none"> <li>+ Information is not updated to the database.</li> <li>+ System show error message.</li> </ul>			
<b>Main Success Scenario:</b>			
Step	Actor Action	System Response	
1	Click button “Sửa”.	Show “Thông tin người dùng” pop up which contains:	
		<ul style="list-style-type: none"> <li>- “Email”: label, can't edit.</li> <li>- “Họ và tên”: textbox, required.</li> <li>- “Ngày sinh”: textbox, date time, min: 01/01/1950, max: 01/01/2000.</li> <li>- “Giới tính”: radio-button(Nam, nữ &amp; khác)</li> <li>- “Số điện thoại”: textbox, min length: 10, max length: 11, number only, required.</li> </ul>	

		<ul style="list-style-type: none"> <li>- “Số điện thoại 2”: textbox, min length: 10, max length: 11, number only.</li> <li>- “Địa chỉ”: textbox, required.</li> <li>- “Địa chỉ 2”: textbox.</li> <li>- “Lưu”: button.</li> <li>- “Hủy”: button</li> </ul> <p>All field have data of the chosen operation admin.</p>
2	System Admin edit data in the form.	
3	System Admin clicks “Lưu” button. [Alternative 1], [Exception 1, 2, 3]	

Update new information of chosen operation admin to the database.

#### Alternative Scenario:

No	Actor Action	System Response
1	System Admin clicks “Hủy” button	System closes the current popup.

#### Exceptions:

No	Actor Action	System Response
1	Data of “Ngày sinh” is not in range.	Show error message: “Ngày sinh từ 01/01/1950 đến 01/01/2000!”
2	Length of “Số điện thoại” or	Show error message: “Số điện thoại phải

	"Số điện thoại 2" is not in range.	gồm 10 hoặc 11 số!"
3	Data of "Số điện thoại" or "Số điện thoại 2" is not only number.	Show error message: "Số điện thoại chỉ bao gồm con số!"

**Relationships:** Manage Operation Admin

**Business Rules:**

- Operation admin has been choose to edit must be in active state.

### 2.2.1.3. <System Admin> Delete Operation Admin



#### USE CASE – UC 03

<b>Use Case No.</b>	UC03	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Delete Operation Admin		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

**Actor:**

- System Admin

**Summary:**

- This use case allows system admin to delete an existing operation admin in the system.

**Goal:**

- Chosen operation admin will be deactivating to the system.

**Triggers:**

- System Admin clicks “Xóa” button of the chosen operation admin on admin list in the “Quản trị viên” page.

**Preconditions:**

- User must log in the system with system admin role.

**Post Conditions:**

- **Success:** Status of the chosen operation admin will be change to deactivate in database.
- **Fail:**
  - + Status is not updated to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click button “Xóa”.	Show alert message confirm: “Bạn muốn xóa tài khoản này?”
2	System Admin click “Ok” button. [Alternative 1]	Change status of chosen operation admin to deactivate in database.

**Alternative Scenario:**

No	Actor Action	System Response
1	System Admin clicks “Hủy” button	

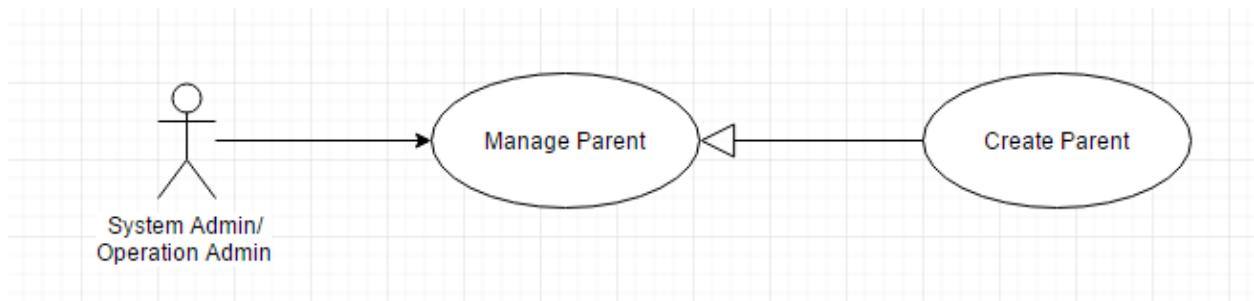
System closes the alert message.

**Relationships:** Manage Operation Admin

**Business Rules:**

- Operation admin has been choose to delete must be in active state.

#### 2.2.1.4. <System Admin & Operation Admin> Create Parent



#### USE CASE - UC 04

Use Case No.	UC04	Use Case Version	1.0
Use Case Name	Create Parent		
Author	Đoàn Hiếu Nghĩa		
Date	24/07/2016	Priority	Normal

**Actor:**

- System Admin, Operation Admin

**Summary:**

- This use case allows user to create new parent in the system.

**Goal:**

- New parent will be created into the system.

**Triggers:**

- User clicks “Phụ huynh” child link into the “Quản lý người dùng” link in the menu.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** New parent will be created to the database.
- **Fail:**
  - + Nothing is added to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click button “Tạo mới”.	Show “Tạo mới tài khoản” pop up which contains: <ul style="list-style-type: none"><li>- “Email”: textbox, must allow email format, required.</li><li>- “Mật khẩu”: textbox, min length = 6, required.</li><li>- “Xác nhận mật khẩu”: textbox, have to equals with Mật khẩu, required.</li><li>- “Họ và tên”: textbox, required.</li><li>- “Ngày sinh”: textbox, date time, min: 01/01/1950, max: 01/01/2000.</li><li>- “Giới tính”: radio-button(Nam, nữ &amp; khác)</li><li>- “Số điện thoại”: textbox, min length: 10, max length: 11, number only, required.</li><li>- “Số điện thoại 2”: textbox, min length: 10, max length: 11, number only.</li><li>- “Địa chỉ”: textbox, required.</li><li>- “Địa chỉ 2”: textbox.</li><li>- “Tạo”: button.</li><li>- “Hủy”: button</li></ul>
2	User fills data to the form.	
3	User clicks “Lưu” button. [Alternative 1], [Exception 1],	

2, 3, 4, 5]

Insert new parent in to the database.

#### Alternative Scenario:

No	Actor Action	System Response
1	User clicks “Hủy” button	

System closes the current popup.

#### Exceptions:

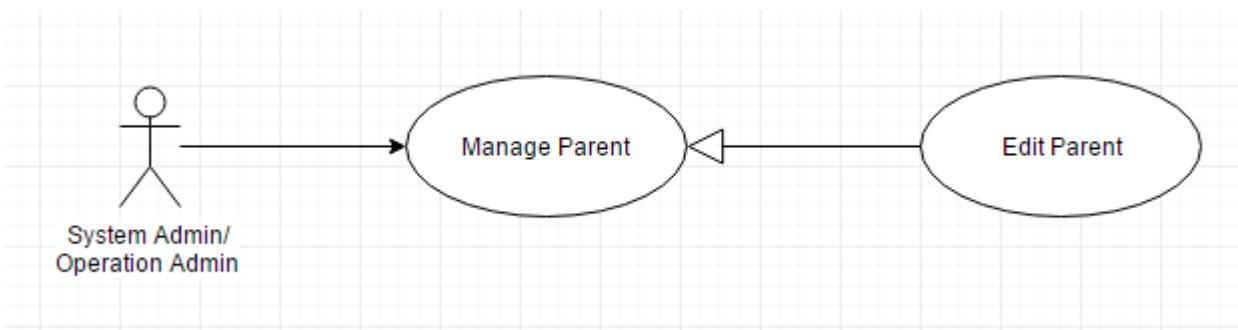
No	Actor Action	System Response
1	Data of “Ngày sinh” is not in range.	Show error message: “Ngày sinh từ 01/01/1950 đến 01/01/2000!”
2	Length of “Số điện thoại” or “Số điện thoại 2” is not in range.	Show error message: “Số điện thoại phải gồm 10 hoặc 11 số!”
3	Data of “Số điện thoại” or “Số điện thoại 2” is not only number.	Show error message: “Số điện thoại chỉ bao gồm con số!”

#### Relationships: Manage Parent

#### Business Rules:

- Each added parent must be in active state when insert to database.

#### 2.2.1.5. <System Admin & Operation Admin > Edit Information of Parent



## USE CASE – UC 05

<b>Use Case No.</b>	UC05	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Edit Parent		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

### Actor:

- System Admin, Operation Admin

### Summary:

- This use case allows user to edit information of an existing parent in the system.

### Goal:

- Information of chosen parent will be updated to the system.

### Triggers:

- User clicks “Sửa” button of the chosen parent on parent list in the “Phụ huynh” page.

### Preconditions:

- User must log in the system with system admin or operation admin role.

### Post Conditions:

- **Success:** Information of the chosen parent will be updated to the database.
- **Fail:**
  - + Information is not updated to the database.
  - + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
------	--------------	-----------------

1 Click button “Sửa”.

Show “Thông tin người dùng” pop up which contains:

- “Email”: label, can't edit.
- “Họ và tên”: textbox, required.
- “Ngày sinh”: textbox, date time, min: 01/01/1950, max: 01/01/2000.
- “Giới tính”: radio-button(Nam, nữ & khác)
- “Số điện thoại”: textbox, min length: 10, max length: 11, number only, required.
- “Số điện thoại 2”: textbox, min length: 10, max length: 11, number only.
- “Địa chỉ”: textbox, required.
- “Địa chỉ 2”: textbox.
- “Lưu”: button.
- “Hủy”: button

All field have data of the chosen operation admin.

2 User edits data in the form.

3 User clicks “Lưu” button.

[Alternative 1], [Exception 1, 2, 3]

Update new information of chosen parent to the database.

#### Alternative Scenario:

No	Actor Action	System Response
1	User clicks “Hủy” button	

System closes the current popup.

### Exceptions:

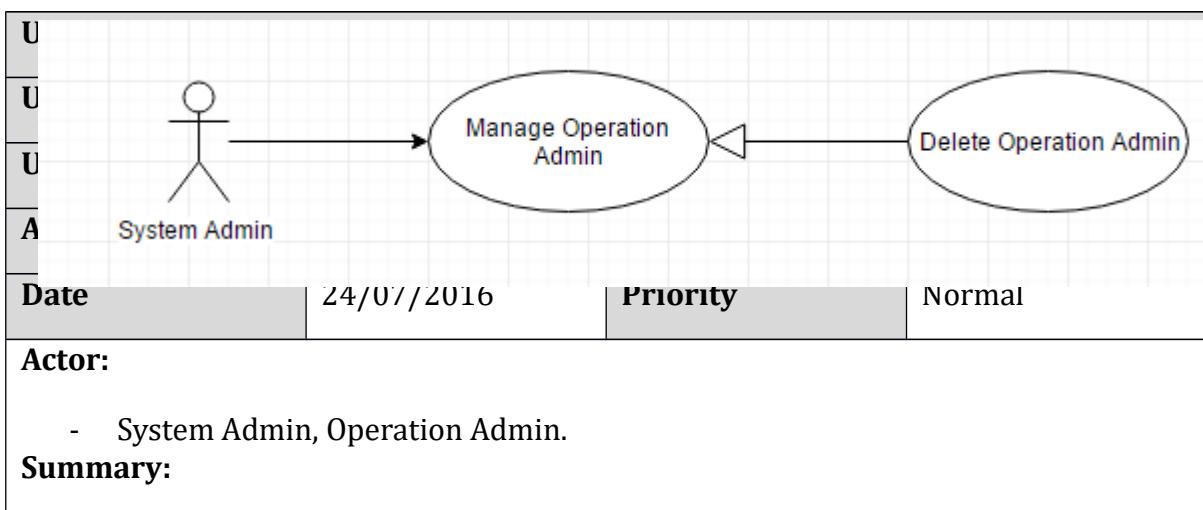
No	Actor Action	System Response
1	Data of “Ngày sinh” is not in range.	Show error message: “Ngày sinh từ 01/01/1950 đến 01/01/2000!”
2	Length of “Số điện thoại” or “Số điện thoại 2” is not in range.	Show error message: “Số điện thoại phải gồm 10 hoặc 11 số!”
3	Data of “Số điện thoại” or “Số điện thoại 2” is not only number.	Show error message: “Số điện thoại chỉ bao gồm con số!”

### Relationships: Manage Parent

### Business Rules:

- Parent has been choose to edit must be in active state.

### 2.2.1.6. <System Admin & Operation Admin> Delete Parent



- This use case allows system user to delete an existing parent in the system.

**Goal:**

- Chosen parent will be deactivating to the system.

**Triggers:**

- User clicks “Xóa” button of the chosen parent on parent list in the “Phụ huynh” page.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** Status of the chosen parent will be change to deactivate in database.
- **Fail:**
  - + Status is not updated to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click button “Xóa”.	Show alert message confirm: “Bạn muốn xóa tài khoản này?”
2	User click “Ok” button. [Alternative 1]	Change status of chosen parent to deactivate in database.

**Alternative Scenario:**

No	Actor Action	System Response
1	User clicks “Hủy” button	

System closes the alert message.

**Exceptions:**

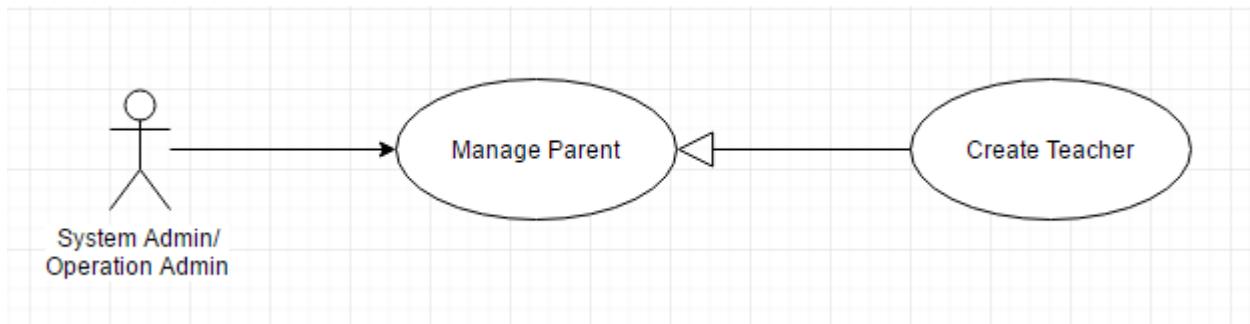
No	Actor Action	System Response
		N/A

**Relationships:** Manage Parent

**Business Rules:**

- Parent has been choose to delete must be in active state.

### 2.2.1.7. <System Admin & Operation Admin> Create Teacher



#### USE CASE – UC 07

Use Case No.	UC07	Use Case Version	1.0
Use Case Name	Create Teacher		
Author	Đoàn Hiếu Nghĩa		
Date	24/07/2016	Priority	Normal

**Actor:**

- System Admin, Operation Admin

**Summary:**

- This use case allows user to create new teacher in the system.

**Goal:**

- New teacher will be created into the system.

**Triggers:**

- User clicks “Giáo viên” child link into the “Quản lý người dùng” link in the menu.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** New teacher will be created to the database.
- **Fail:**
  - + Nothing is added to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click button “Tạo mới”.	

Show “Tạo mới tài khoản” pop up which contains:

- “Email”: textbox, must allow email format, required.
- “Mật khẩu”: textbox, min length = 6, required.
- “Xác nhận mật khẩu”: textbox, have to equals with Mật khẩu, required.
- “Họ và tên”: textbox, required.
- “Ngày sinh”: textbox, date time, min: 01/01/1950, max: 01/01/2000.
- “Giới tính”: radio-button(Nam, nữ & khác)
- “Số điện thoại”: textbox, min length: 10, max length: 11, number only, required.
- “Số điện thoại 2”: textbox, min length: 10, max length: 11, number only.
- “Địa chỉ”: textbox, required.

- “Địa chỉ 2”: textbox.
  - “Tạo”: button.
  - “Hủy”: button
- 2 User fills data to the form.
  - 3 User clicks “Lưu” button.  
[Alternative 1], [Exception 1,  
2, 3, 4, 5]

Insert new teacher in to the database.

#### **Alternative Scenario:**

No	Actor Action	System Response
1	User clicks “Hủy” button	

System closes the current popup.

#### **Exceptions:**

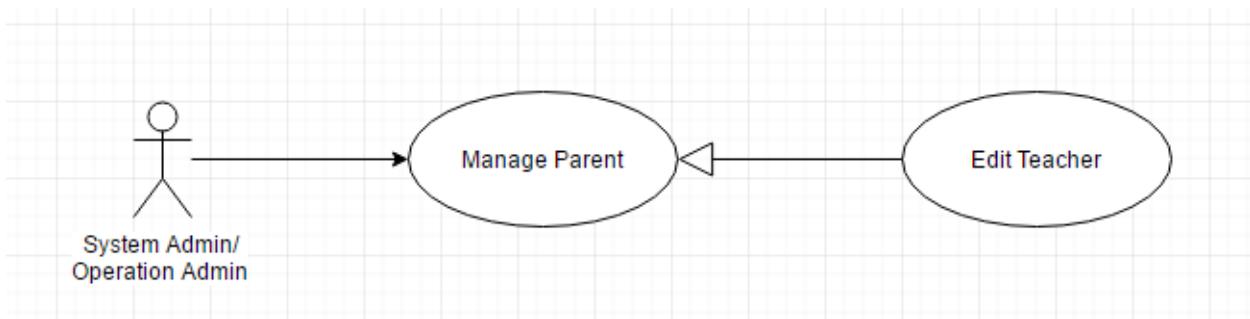
No	Actor Action	System Response
1	Data of “Ngày sinh” is not in range.	Show error message: “Ngày sinh từ 01/01/1950 đến 01/01/2000!”
2	Length of “Số điện thoại” or “Số điện thoại 2” is not in range.	Show error message: “Số điện thoại phải gồm 10 hoặc 11 số!”
3	Data of “Số điện thoại” or “Số điện thoại 2” is not only number.	Show error message: “Số điện thoại chỉ bao gồm con số!”

**Relationships:** Manage Teacher

**Business Rules:**

- Each added teacher must be in active state when insert to database.

### 2.2.1.8. <System Admin & Operation Admin > Edit Information of Teacher



#### USE CASE – UC 08

<b>Use Case No.</b>	UC08	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Edit Teacher		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

#### Actor:

- System Admin, Operation Admin

#### Summary:

- This use case allows user to edit information of an existing teacher in the system.

#### Goal:

- Information of chosen teacher will be updated to the system.

#### Triggers:

- User clicks “Sửa” button of the chosen teacher on teacher list in the “Giáo viên” page.

#### Preconditions:

- User must log in the system with system admin or operation admin role.

#### Post Conditions:

- **Success:** Information of the chosen teacher will be updated to the database.

- **Fail:**
  - + Information is not updated to the database.
  - + System show error message.

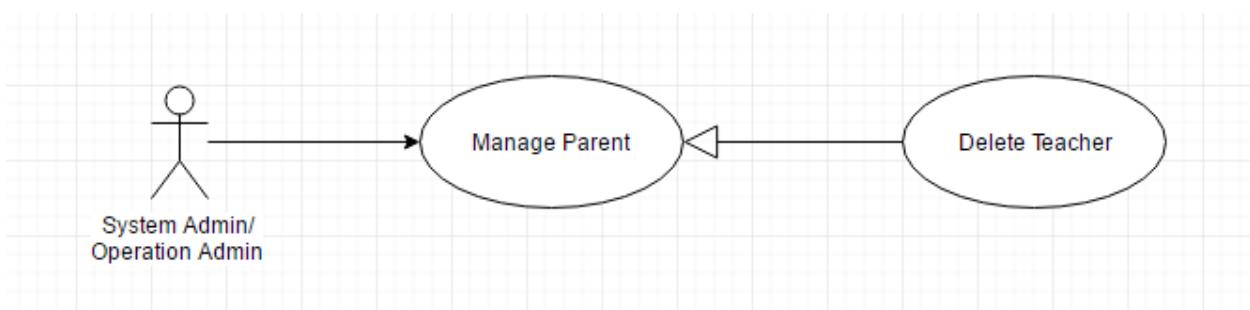
### Main Success Scenario:

Step	Actor Action	System Response
1	Click button “Sửa”.	Show “Thông tin người dùng” pop up which contains:  <ul style="list-style-type: none"> <li>- “Email”: label, can't edit.</li> <li>- “Họ và tên”: textbox, required.</li> <li>- “Ngày sinh”: textbox, date time, min: 01/01/1950, max: 01/01/2000.</li> <li>- “Giới tính”: radio-button(Nam, nữ &amp; khác)</li> <li>- “Số điện thoại”: textbox, min length: 10, max length: 11, number only, required.</li> <li>- “Số điện thoại 2”: textbox, min length: 10, max length: 11, number only.</li> <li>- “Địa chỉ”: textbox, required.</li> <li>- “Địa chỉ 2”: textbox.</li> <li>- “Lưu”: button.</li> <li>- “Hủy”: button</li> </ul> All fields have data of the chosen operation admin.
2	User edits data in the form.	
3	User clicks “Lưu” button. [Alternative 1], [Exception 1, 2, 3]	Update new information of chosen teacher to the database.

### Alternative Scenario:

No	Actor Action	System Response
1	User clicks “Hủy” button	System closes the current popup.
<b>Exceptions:</b>		
No	Actor Action	System Response
1	Data of “Ngày sinh” is not in range.	Show error message: “Ngày sinh từ 01/01/1950 đến 01/01/2000!”
2	Length of “Số điện thoại” or “Số điện thoại 2” is not in range.	Show error message: “Số điện thoại phải gồm 10 hoặc 11 số!”
3	Data of “Số điện thoại” or “Số điện thoại 2” is not only number.	Show error message: “Số điện thoại chỉ bao gồm con số!”
<b>Relationships:</b> Manage Teacher		
<b>Business Rules:</b>		
- Teacher has been choose to edit must be in active state.		

### 2.2.1.9. <System Admin & Operation Admin> Delete Teacher



### USE CASE - UC 09

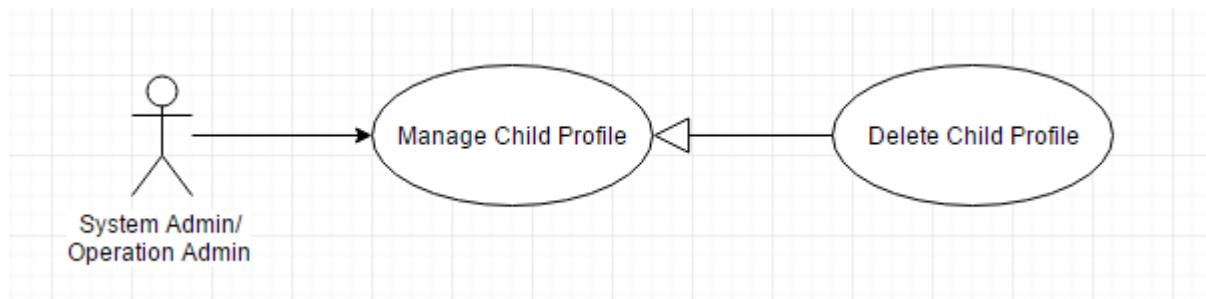
<b>Use Case No.</b>	UC09	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Delete Teacher		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal
<b>Actor:</b>			
<ul style="list-style-type: none"> <li>- System Admin, Operation Admin.</li> </ul>			
<b>Summary:</b>			
<ul style="list-style-type: none"> <li>- This use case allows system user to delete an existing teacher in the system.</li> </ul>			
<b>Goal:</b>			
<ul style="list-style-type: none"> <li>- Chosen teacher will be deactivating to the system.</li> </ul>			
<b>Triggers:</b>			
<ul style="list-style-type: none"> <li>- User clicks “Xóa” button of the chosen teacher on teacher list in the “Giáo viên” page.</li> </ul>			
<b>Preconditions:</b>			
<ul style="list-style-type: none"> <li>- User must log in the system with system admin or operation admin role.</li> </ul>			
<b>Post Conditions:</b>			
<ul style="list-style-type: none"> <li>- <b>Success:</b> Status of the chosen teacher will be change to deactivate in database.</li> <li>- <b>Fail:</b> <ul style="list-style-type: none"> <li>+ Status is not updated to the database.</li> <li>+ System show error message.</li> </ul> </li> </ul>			
<b>Main Success Scenario:</b>			
Step	Actor Action	System Response	
1	Click button “Xóa”.	Show alert message confirm: “Bạn muốn xóa tài khoản này?”	
2	User click “Ok” button. [Alternative 1]	Change status of chosen teacher to deactivate in database.	

**Alternative Scenario:**

No	Actor Action	System Response
1	User clicks "Hủy" button	System closes the alert message.

**Relationships:** Manage Teacher**Business Rules:**

- Teacher has been choose to delete must be in active state.

**2.2.1.10. <System Admin & Operation Admin> Delete Child Profile****USE CASE – UC 10**

<b>Use Case No.</b>	UC10	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Delete Child Profile		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal
<b>Actor:</b>			

- System Admin, Operation Admin.

**Summary:**

- This use case allows system user to delete an existing child profile in the system.

**Goal:**

- Chosen child profile will be deactivating to the system.

**Triggers:**

- User clicks “Xóa” button of the chosen child profile on child profile list in the “Danh sách học sinh” page.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** Status of the chosen child profile will be change to deactivate in database.
- **Fail:**
  - + Status is not updated to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click button “Xóa”.	Show alert message confirm: “Bạn muốn xóa tài khoản này?”
2	User click “Ok” button. [Alternative 1]	Change status of chosen child profile to deactivate in database.

**Alternative Scenario:**

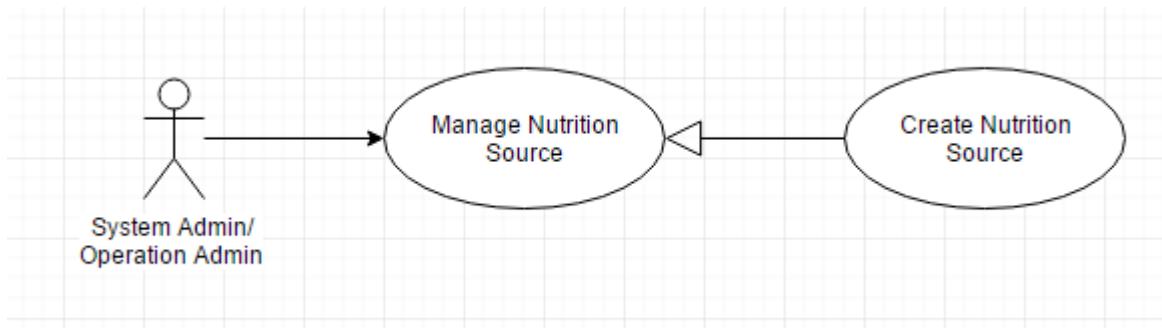
No	Actor Action	System Response
1	User clicks "Hủy" button	System closes the alert message.

**Relationships:** Manage Child Profile

**Business Rules:**

- Child profile has been choose to delete must be in active state.

### 2.2.1.11. <System Admin & Operation Admin> Create Nutrition Source



#### USE CASE – UC 11

Use Case No.	UC11	Use Case Version	1.0
Use Case Name	Create Nutrition Source		
Author	Đoàn Hiếu Nghĩa		
Date	24/07/2016	Priority	Normal

**Actor:**

- System Admin, Operation Admin.

**Summary:**

- This use case allows system user to create a nutrition source in the system.

**Goal:**

- New nutrition source will be created into the database.

**Triggers:**

- User clicks “Chất dinh dưỡng” child link into the “Dinh dưỡng” link in the menu.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** New nutrition source will be creating to the database.
- **Fail:**
  - + Nothing added to the database.
  - + System show error message.

**Main Success Scenario:**

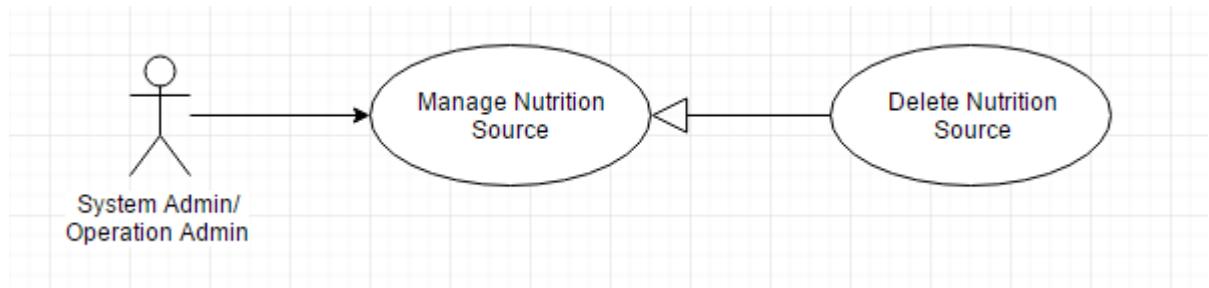
Step	Actor Action	System Response
1	Click link “Chất dinh dưỡng”.	Show “Chất dinh dưỡng” page which contains: <ul style="list-style-type: none"><li>- “Chất dinh dưỡng”: textbox, required.</li><li>- “Thực phẩm tiêu biểu”: textbox, required.</li><li>- “Danh sách các chất dinh dưỡng”: table.</li><li>- “Thêm chất dinh dưỡng”: button.</li><li>- “Xóa”: button.</li></ul>
2	User fill data to the form	
3	User click “Thêm chất dinh dưỡng” button.	Insert new nutrition source into the database.

**Relationships:** Manage Nutrition Source

## **Business Rules:**

- Each added nutrition source must be in active state when insert to database.

### **2.2.1.12. <System Admin & Operation Admin> Delete Nutrition Source**



#### **USE CASE – UC 12**

<b>Use Case No.</b>	UC12	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Delete Nutrition Source		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

#### **Actor:**

- System Admin, Operation Admin.

#### **Summary:**

- This use case allows system user to delete an existing nutrition source in the system.

#### **Goal:**

- All chosen nutrition source will be deactivate to the system.

#### **Triggers:**

- User clicks “Xóa” button, all chosen nutrition source on nutrition source list in the “Chất dinh dưỡng” page.

#### **Preconditions:**

- User must log in the system with system admin or operation admin role.

#### **Post Conditions:**

- **Success:** Status of all chosen nutrition source will be change to deactivate in

- database.
- **Fail:**
    - + Status is not updated to the database.
    - + System show error message.

#### Main Success Scenario:

Step	Actor Action	System Response
1	User check to the check box of all nutrition source which need to delete.	
2	User click button “Xóa”.	Show alert message confirm: “Bạn chắc chắn muốn xóa các chất dinh dưỡng này?”
3	User click “Ok” button. [Alternative 1]	Change status of all chosen nutrition source to deactivate in database.

#### Alternative Scenario:

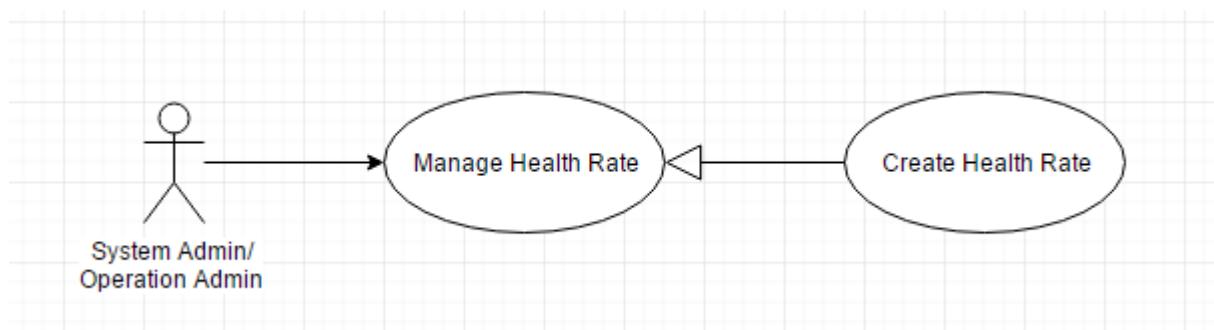
No	Actor Action	System Response
1	User clicks “Hủy” button	System closes the alert message.

#### Relationships: Manage Nutrition Source

#### Business Rules:

- Nutrition source has been choose to delete must be in active state.
- 2.2.1.13. <System Admin & Operation Admin> Create Health Rate**

#### USE CASE – UC 13



#### Actor:

- System Admin, Operation Admin.

#### Summary:

- This use case allows system user to create a health rate in the system.

#### Goal:

- New health rate will be created into the database.

#### Triggers:

- User clicks “Định mức sức khỏe” child link into the “Dinh dưỡng” link in the menu.

#### Preconditions:

- User must log in the system with system admin or operation admin role.

#### Post Conditions:

- **Success:** New health rate will be creating to the database.
- **Fail:**
  - + Nothing added to the database.
  - + System show error message.

#### Main Success Scenario:

Step	Actor Action	System Response
1	Click “Tạo mới định mức sức khỏe” button.	<p>Show “Tạo mới định mức sức khỏe” pop up which contains:</p> <ul style="list-style-type: none"> <li>- “Tên định mức sức khỏe”: textbox, required.</li> <li>- “Độ tuổi từ (tháng tuổi)": textbox, required, min length = 1, max length = 2, number only.</li> <li>- “đến (tháng tuổi)": textbox,</li> </ul>

- required, min length = 1, max length = 2, number only.
- “Chiều cao từ (cm)": textbox, required, min length = 2, max length = 3, number only.
- “đến (cm)": textbox, required, min length = 2, max length = 3, number only.
- “Cân nặng từ (kg)": textbox, required, min length = 2, max length = 2, number only.
- “đến (kg)": textbox, required, min length = 2, max length = 2, number only.
- “Ghi chú": textbox.
- “Tạo định mức sức khỏe": button.
- “Hủy": button.

2 User fill data to the form

3 User click “Tạo định mức sức khỏe” button. [Alternative 1]

Insert new health rate into the database.

### **Alternative Scenario:**

No	Actor Action	System Response
1	User click “Hủy” button.	System closes the current pop up.

### **Exceptions:**

No	Actor Action	System Response
1	Length of “Độ tuổi từ (tháng tuổi)” or “đến (tháng tuổi)” is not in range.	Show error message: “Độ tuổi phải có 1 hoặc 2 con số và đơn vị là tháng tuổi!”

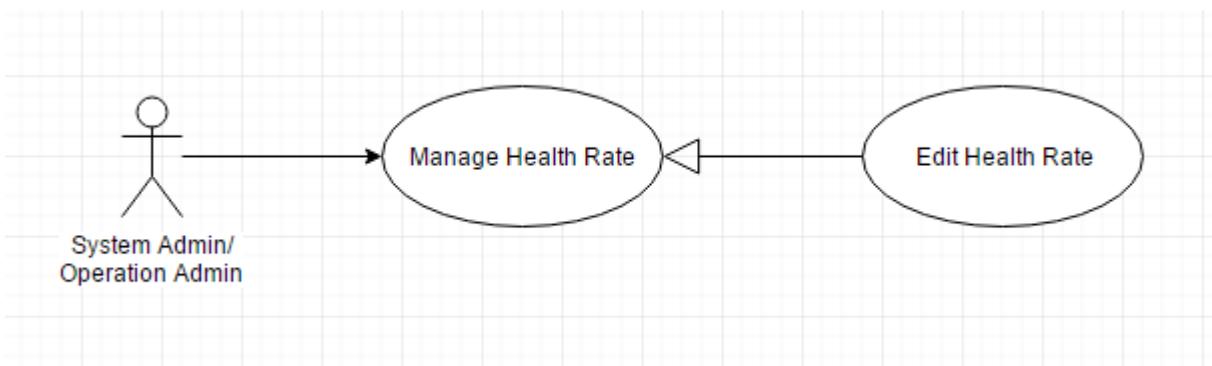
2	Data of “Độ tuổi từ (tháng tuổi)” or “đến (tháng tuổi)” is not only number.	Show error message: “Độ tuổi phải có 1 hoặc 2 con số và đơn vị là tháng tuổi!”
3	Length of “Chiều cao từ (cm)” or “đến (cm)” is not in range.	Show error message: “Chiều cao phải từ 2 đến 3 con số và đơn vị là Cm.”
4	Data of “Chiều cao từ (cm)” or “đến (cm)” is not in range.	Show error message: “Chiều cao phải từ 2 đến 3 con số và đơn vị là Cm.”
5	Length of “Cân nặng từ (kg)” or “đến (kg)” is not in range.	Show error message: “Cân nặng phải là 2 con số và đơn vị là Kg.”
6	Data of “Cân nặng từ (kg)” or “đến (kg)” is not in range.	Show error message: “Cân nặng phải là 2 con số và đơn vị là Kg.”

### Relationships: Manage Health Rate

#### Business Rules:

- Each added health rate must be in active state when insert to database.

#### 2.2.1.14. <System Admin & Operation Admin> Edit Information of Health Rate



#### USE CASE – UC 14

<b>Use Case No.</b>	UC14	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Edit Information of Health Rate		
<b>Author</b>	Đoàn Hiếu Nghĩa		

<b>Date</b>	24/07/2016	<b>Priority</b>	Normal
-------------	------------	-----------------	--------

**Actor:**

- System Admin, Operation Admin.

**Summary:**

- This use case allows system user to edit information of an existing health rate in the system.

**Goal:**

- Information of chosen health rate will be updated to the system.

**Triggers:**

- User clicks “Sửa” button of the chosen health rate on health rate list in the “Định mức sức khỏe” page.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** Information of the chosen health rate will be updated to the database.
- **Fail:**
  - + Information is not updated to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click “Sửa” button.	Show “Tạo mới định mức sức khỏe” pop up which contains:

Show “Tạo mới định mức sức khỏe” pop up which contains:

- “Tên định mức sức khỏe”: textbox, required.
- “Độ tuổi từ (tháng tuổi)": textbox, required, min length = 1, max length = 2, number only.
- “đến (tháng tuổi)": textbox, required, min length = 1, max length = 2, number only.
- “Chiều cao từ (cm)": textbox, required, min length = 2, max

- length = 3, number only.
  - “đến (cm)": textbox, required, min length = 2, max length = 3, number only.
  - “Cân nặng từ (kg)": textbox, required, min length = 2, max length = 2, number only.
  - “đến (kg)": textbox, required, min length = 2, max length = 2, number only.
  - “Ghi chú": textbox.
  - “Tạo định mức sức khỏe": button.
  - “Hủy": button.
- All fields have data of the chosen health rate.

2 User edit data in the form

3 User click “Lưu” button.

[Alternative 1]

Update new information of chosen health rate into the database.

### **Alternative Scenario:**

No	Actor Action	System Response
1	User click “Hủy” button.	System closes the current pop up.

### **Exceptions:**

No	Actor Action	System Response
1	Length of “Độ tuổi từ (tháng tuổi)” or “đến (tháng tuổi)” is not in range.	Show error message: “Độ tuổi phải có 1 hoặc 2 con số và đơn vị là tháng tuổi!”
2	Data of “Độ tuổi từ (tháng	Show error message: “Độ tuổi phải có 1

"tuổi)" or "đến (tháng tuổi)" is not only number. hoặc 2 con số và đơn vị là tháng tuổi!"

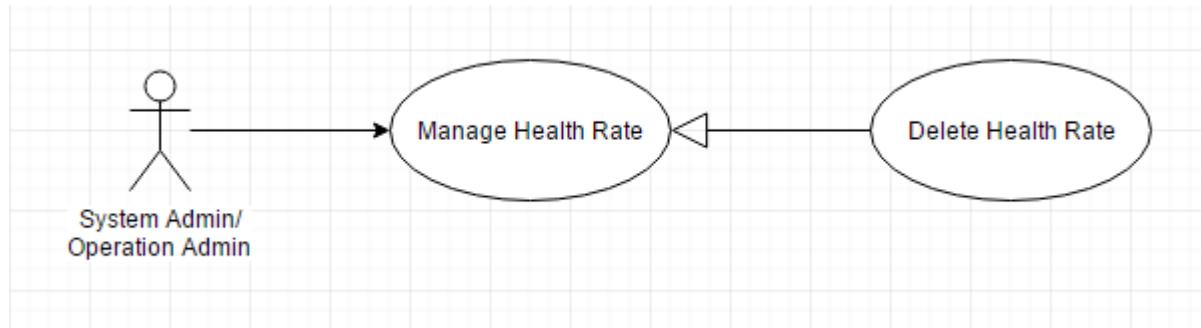
- |   |  |   |
|---|--|---|
| 3 | Length of "Chiều cao từ (cm)" or "đến (cm)" is not in range. | Show error message: "Chiều cao phải từ 2 đến 3 con số và đơn vị là Cm." |
| 4 | Data of "Chiều cao từ (cm)" or "đến (cm)" is not in range.   | Show error message: "Chiều cao phải từ 2 đến 3 con số và đơn vị là Cm." |
| 5 | Length of "Cân nặng từ (kg)" or "đến (kg)" is not in range.  | Show error message: "Cân nặng phải là 2 con số và đơn vị là Kg."        |
| 6 | Data of "Cân nặng từ (kg)" or "đến (kg)" is not in range.    | Show error message: "Cân nặng phải là 2 con số và đơn vị là Kg."        |

### **Relationships:** Manage Health Rate

#### **Business Rules:**

- Health rate has been choose to edit must be in active state.

#### **2.2.1.15. <System Admin & Operation Admin> Delete Health Rate**



#### **USE CASE – UC 15**

<b>Use Case No.</b>	UC15	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Delete Health Rate		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

**Actor:**

- System Admin, Operation Admin.

**Summary:**

- This use case allows system user to delete an existing health rate in the system.

**Goal:**

- Chosen health rate will be deactivating to the system.

**Triggers:**

- User clicks “Xóa” button of the chosen health rate on health rate list in the “Định mức sức khỏe” page.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** Status of chosen health rate will be change to deactivate in database.
- **Fail:**
  - + Status is not updated to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	User click button “Xóa”.	Show alert message confirm: “Bạn muốn xóa định mức sức khỏe này?”
2	User click “Ok” button. [Alternative 1]	Change status of all chosen health rate to deactivate in database.

**Alternative Scenario:**

No	Actor Action	System Response
1	User clicks “Hủy” button	

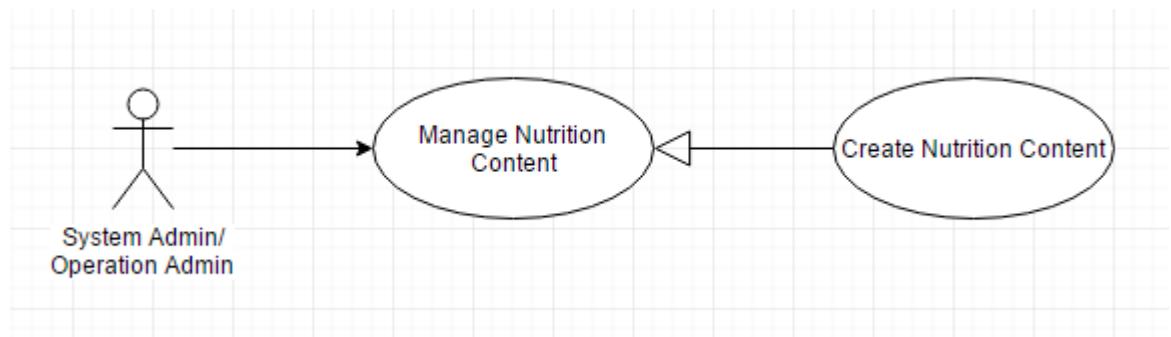
System closes the alert message.

### **Relationships:** Manage Health Rate

#### **Business Rules:**

- Health Rate has been choose to delete must be in active state.

#### **2.2.1.16. <System Admin & Operation Admin> Create Nutrition Content**



#### **USE CASE – UC 16**

<b>Use Case No.</b>	UC16	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Create Nutrition Content		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

#### **Actor:**

- System Admin, Operation Admin.

#### **Summary:**

- This use case allows system user to create a nutrition content in the system.

#### **Goal:**

- New nutrition content will be created into the database.

**Triggers:**

- User clicks “Dinh dưỡng cho trẻ” child link into the “Dinh dưỡng” link in the menu.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** New nutrition content will be create to the database.
- **Fail:**
  - + Nothing added to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click "Dinh dưỡng cho trẻ" link.	Show "Dinh dưỡng cho trẻ" page which contains:

Show “Dinh dưỡng cho trẻ” page which contains:

- “Chọn định mức sức khỏe”: dropdown list, contain all active health rate in system, required.
- “Độ tuổi từ (tháng tuổi)": textbox, disabled, contain data of chosen health rate in dropdownlist.
- “đến (tháng tuổi)": textbox, disabled, contain data of chosen health rate in dropdownlist.
- “Chiều cao từ (cm)": textbox, disabled, contain data of chosen health rate in dropdownlist.
- “đến (cm)": textbox, disabled, contain data of chosen health rate in dropdownlist.
- “Cân nặng từ (kg)": textbox, disabled, contain data of chosen health rate in dropdownlist.
- “đến (kg)": textbox, disabled, contain data of chosen health rate in dropdownlist.
- “Ghi chú": textbox, disabled, contain data of chosen health rate

- in dropdownlist.
- “Thành phần dinh dưỡng cần thiết”: table, contains:
  - + “Chất dinh dưỡng”: textbox, disabled.
  - + “Hàm lượng”: textbox.
  - + “Đơn vị”: dropdown list, 3 option (mg/ngày, lít/ngày, ml/ngày)
  - + “Lưu ý”: textbox.
- “Lưu lại”: button.

2 User fill data to the table  
“Thành phần dinh dưỡng cần thiết”

User click “Lưu lại” button.

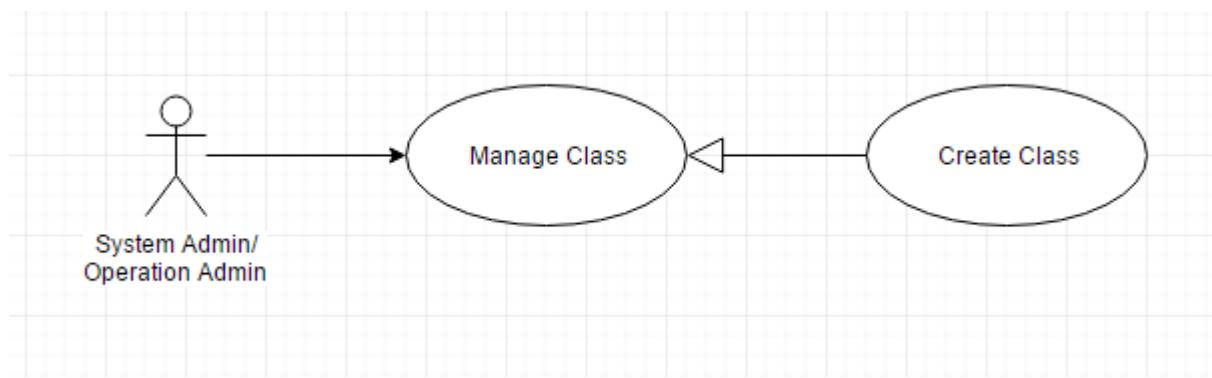
3 Insert nutrition contents into the database with each nutrition source have data in “Thành phần dinh dưỡng cần thiết” table.

### **Relationships:** Manage Nutrition Content

#### **Business Rules:**

- Each added nutrition content must be in active state when insert to database.

#### **2.2.1.17. <System Admin & Operation Admin> Create Class**



## USE CASE - UC 17

<b>Use Case No.</b>	UC17	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Create Class		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

### Actor:

- System Admin, Operation Admin.

### Summary:

- This use case allows system user to create a class in the system.

### Goal:

- New class will be created into the database.

### Triggers:

- User clicks “Lịch giảng dạy” in the menu.

### Preconditions:

- User must log in the system with system admin or operation admin role.

### Post Conditions:

- **Success:** New class will be create to the database.
- **Fail:**
  - + Nothing added to the database.
  - + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
1	Click “Tạo mới lớp học” button.	Show “Tại mới lớp học” pop up which contains:

Show “Tại mới lớp học” pop up which contains:

- “Tên lớp học”: textbox, required.
- “Ngày bắt đầu”: textbox, date, required, min = today.
- “Ngày kết thúc”: textbox, date, required.
- “Phòng”: textbox, required.
- “Mô tả”: textbox.

- “Cân nặng từ (kg)": textbox, disabled, contain data of chosen health rate in dropdownlist.
- “Loại lớp học": dropdown list, required, contain 4 value (giữ trẻ, mầm, chồi, lá).
- “Lưu lại": button.
- “Hủy": button.

## 2 User fill data to the form

User click “Lưu lại” button.

[Alternative Scenario 1]

[Exception 1]

Insert new class into the database.

### **Alternative Scenario:**

No	Actor Action	System Response
1	User click “Hủy” button.	System closes the current pop up.

### **Exceptions:**

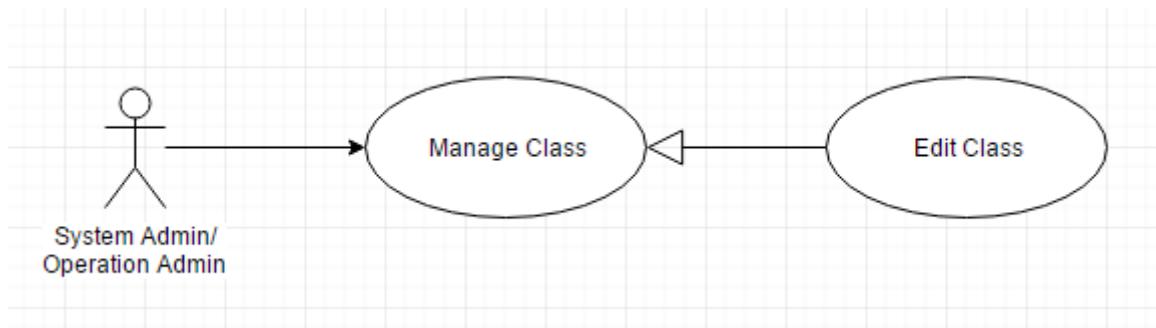
No	Actor Action	System Response
1	Data of “Ngày bắt đầu” greater than to day	Show error message: “Ngày bắt đầu nhỏ nhất là hôm nay!”

### **Relationships:** Manage Class

### **Business Rules:**

- Each added class must be in active state when insert to database.

### 2.2.1.18. <System Admin & Operation Admin> Edit Information of Class



#### USE CASE - UC 18

<b>Use Case No.</b>	UC18	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Edit Information of Class		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

#### Actor:

- System Admin, Operation Admin.

#### Summary:

- This use case allows system user to edit information of an existing class in the system.

#### Goal:

- Information of chosen class will be updated to the system.

**Triggers:**

- User clicks “Sửa” button of the chosen class on class list in the “Quản lý lịch giảng dạy” page.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** Information of the chosen class will be updated to the database.
- **Fail:**
  - + Information is not updated to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click "Sửa" button.	Show “Tại mới lớp học” pop up which contains: <ul style="list-style-type: none"><li>- “Tên lớp học”: textbox, required.</li><li>- “Ngày bắt đầu”: textbox, date, required, min = today.</li><li>- “Ngày kết thúc”: textbox, date, required.</li><li>- “Phòng”: textbox, required.</li><li>- “Mô tả”: textbox.</li><li>- “Cân nặng từ (kg)": textbox, disabled, contain data of chosen health rate in dropdownlist.</li><li>- “Loại lớp học”: dropdown list, required, contain 4 values (giữ trẻ, mầm, chồi, lá).</li><li>- “Lưu lại”: button.</li><li>- “Hủy”: button.</li></ul> All fields have data of the chosen class.
2	User edit data in the form	
3	User click “Lưu” button.	

[Alternative 1] [ Exception 1]

Update new information of chosen class into the database.

**Alternative Scenario:**

No	Actor Action	System Response
1	User click "Hủy" button.	System closes the current pop up.

**Exceptions:**

No	Actor Action	System Response
1	Data of "Ngày bắt đầu" greater than to day.	Show error message: "Ngày bắt đầu nhỏ nhất là hôm nay!"

**Relationships:** Manage Class

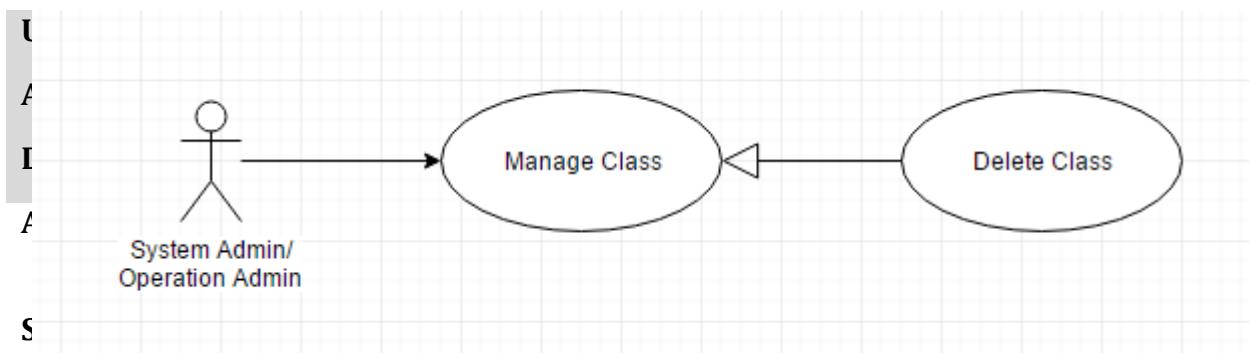
**Business Rules:**

- Class has been choose to edit must be in active state.

**2.2.1.19. <System Admin & Operation Admin> Delete Class**

**USE CASE – UC 19**

<b>Use Case No.</b>	UC19	<b>Use Case Version</b>	1.0
---------------------	------	-------------------------	-----



- This use case allows system user to delete an existing class in the system.

#### Goal:

- Chosen class will be deactivating to the system.

#### Triggers:

- User clicks “Xóa” button of the chosen class on class list in the “Quản lý lịch giảng dạy” page.

#### Preconditions:

- User must log in the system with system admin or operation admin role.

#### Post Conditions:

- **Success:** Status of chosen class will be change to deactivate in database.
- **Fail:**
  - + Status is not updated to the database.
  - + System show error message.

#### Main Success Scenario:

Step	Actor Action	System Response
1	User click button “Xóa”.	Show alert message confirm: “Bạn muốn xóa lớp này?”
2	User click “Ok” button. [Alternative 1]	Change status of all chosen class to deactivate in database.

### Alternative Scenario:

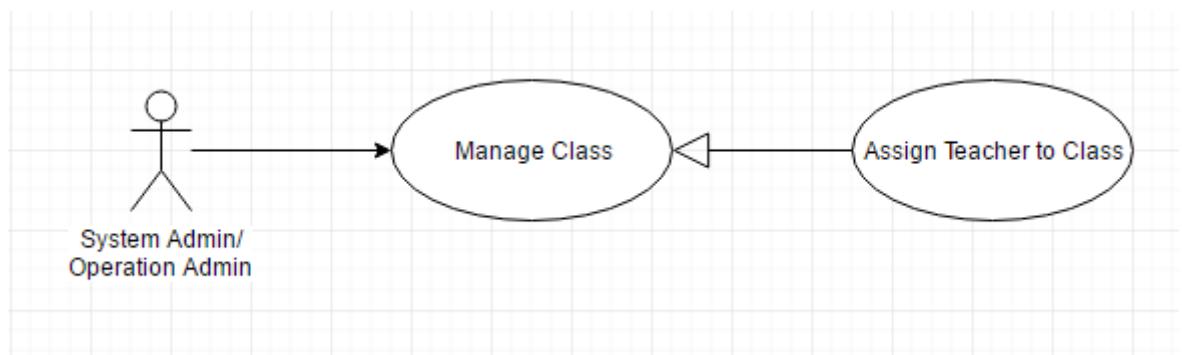
No	Actor Action	System Response
1	User clicks "Hủy" button	
		System closes the alert message.

**Relationships:** Manage Class

**Business Rules:**

- Class has been choose to delete must be in active state.

### 2.2.1.20. <System Admin & Operation Admin> Assign Teacher to Class



### USE CASE - UC 20

<b>Use Case No.</b>	UC20	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Assign Teacher to Class		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

**Actor:**

- System Admin, Operation Admin.

### **Summary:**

- This use case allows system user to assign existing teacher to a class in the system.

### **Goal:**

- Teacher will be assigned to chosen class.

### **Triggers:**

- User clicks “Quản lý lớp” link of the chosen class on class list in the “Quản lý lịch giảng dạy” page.

### **Preconditions:**

- User must log in the system with system admin or operation admin role.

### **Post Conditions:**

- **Success:** Information of teacher will be assigned to the chosen class.
- **Fail:**
  - + Information will be created in the database.
  - + System shows error message.

### **Main Success Scenario:**

Step	Actor Action	System Response
1	User click “Quản lý lớp” link.	<p>Show “Quản lý lịch giảng dạy” page which contains:</p> <ul style="list-style-type: none"> <li>- Information of chosen class:           <ul style="list-style-type: none"> <li>+ “Tên lớp”: name of chosen class.</li> <li>+ “Ngày bắt đầu”: start date of chosen class.</li> <li>+ “Ngày kết thúc”: end date of chosen class.</li> <li>+ “Khối lớp”: grade of chosen class.</li> <li>+ “Phòng học”: room of chosen class.</li> </ul> </li> <li>- “Giáo viên”: table, list of teacher assigned in chosen class, which contains:</li> </ul>

- + “Tên”: name of teacher.
- + “Ngày sinh”: birthday of teacher.
- + “Email”: email of teacher.
- + “Số điện thoại”: phone of teacher.
- + “Xóa”: button.
- “Thêm giáo viên”: button.
- “Học sinh”: table, list of children assigned in chosen class, which contains:
  - + “Tên”: name of child.
  - + “Ngày sinh”: birthday of child.
  - + “Tên phụ huynh”: parent's name of child.
  - + “Số điện thoại phụ huynh”: parent's phone of child.
  - + “Xóa”: button
- “Thêm học sinh”: button.

2 User click “Thêm giáo viên” button.

Show “Thêm giáo viên” pop up which contains:

- “Danh sách đã được thêm”: table, list of teacher assigned in chosen class which contains:
  - + “Tên”: name of teacher.
  - + “Email”: email of teacher.
  - + “Số điện thoại”: phone of teacher.
  - + “Red button”: button.
- “Danh sách có thể thêm”: table, list of teacher unassigned in any class which contains:
  - + “Tên”: name of teacher.
  - + “Email”: email of teacher.
  - + “Số điện thoại”: phone of teacher.
  - + “Green button”: button.

		<ul style="list-style-type: none"> <li>- “Ok”: button.</li> <li>- “Hủy”: button.</li> <li>- “Search”: textbox</li> </ul>
3	User click “Green button” button of an unassigned teacher need to assign to chosen class. [Alternative 1].	Teacher has been chosen will move to “Danh sách đã được thêm” and “Green button” button will change to “Red button”.
4	User click “Ok” button. [Alternative 2]	All teacher in “Danh sách đã được thêm” will be assigned or unassigned to chosen class. Information will update to database.

### Alternative Scenario:

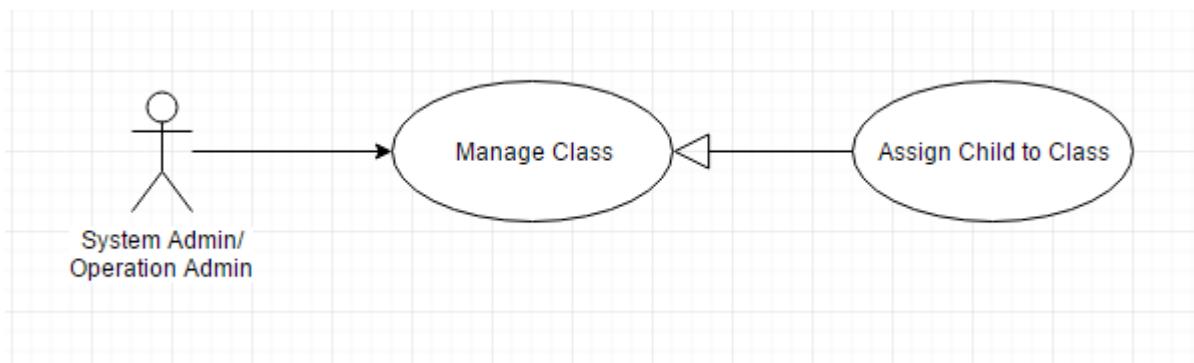
No	Actor Action	System Response
1	User click “Red button” button of an assigned teacher need to remove from chosen class.	Teacher has been chosen will move to “Danh sách có thể thêm” and “Red button” will change to “Green button”.
2	User click “Hủy” button.	System will close the current pop up.

### Relationships: Manage Class

### Business Rules:

- Class has been chosen to assign teacher must be in active state.

#### 2.2.1.21. <System Admin & Operation Admin> Assign Child to Class



#### USE CASE - UC 21

<b>Use Case No.</b>	UC21	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Assign Child to Class		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal
<b>Actor:</b>			

- System Admin, Operation Admin.

#### Summary:

- This use case allows system user to assign existing child to a class in the system.

#### Goal:

- Child will be assign to chosen class.

**Triggers:**

- User clicks “Quản lý lớp” link of the chosen class on class list in the “Quản lý lịch giảng dạy” page.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** Information of child will be assign to the chosen class.
- **Fail:**
  - + Information will be creating to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	User click “Quản lý lớp” link.	<p>Show “Quản lý lịch giảng dạy” page which contains:</p> <ul style="list-style-type: none"> <li>- Information of chosen class: <ul style="list-style-type: none"> <li>+ “Tên lớp”: name of chosen class.</li> <li>+ “Ngày bắt đầu”: start date of chosen class.</li> <li>+ “Ngày kết thúc”: end date of chosen class.</li> <li>+ “Khối lớp”: grade of chosen class.</li> <li>+ “Phòng học”: room of chosen class.</li> </ul> </li> <li>- “Giáo viên”: table, list of teacher assigned in chosen class, which contains: <ul style="list-style-type: none"> <li>+ “Tên”: name of teacher.</li> <li>+ “Ngày sinh”: birthday of teacher.</li> <li>+ “Email”: email of teacher.</li> <li>+ “Số điện thoại”: phone of teacher.</li> <li>+ “Xóa”: button.</li> </ul> </li> <li>- “Thêm giáo viên”: button.</li> </ul>

- “Học sinh”: table, list of children assigned in chosen class, which contains:
  - + “Tên”: name of child.
  - + “Ngày sinh”: birthday of child.
  - + “Tên phụ huynh”: parent’s name of child.
  - + “Số điện thoại phụ huynh”: parent’s phone of child.
  - + “Xóa”: button
- “Thêm học sinh”: button.

2 User click “Thêm học sinh” button.

Show “Thêm học sinh” pop up which contains:

- “Danh sách đã được thêm”: table, list of teacher assigned in chosen class which contains:
  - + “Tên”: name of child.
  - + “Ngày sinh”: birthday of child.
  - + “Tên phụ huynh”: parent’s name of child.
  - + “Số điện thoại phụ huynh”: parent’s phone of child.
  - + “Red button”: button.
- “Danh sách có thể thêm”: table, list of teacher unassigned in any class which contains:
  - + “Tên”: name of child.
  - + “Ngày sinh”: birthday of child.
  - + “Tên phụ huynh”: parent’s name of child.
  - + “Số điện thoại phụ huynh”: parent’s phone of child.
  - + “Green button”: button.
- “Ok”: button.
- “Hủy”: button.

		- “Search”: textbox
3	User click “Green button” button of an unassigned child need to assign to chosen class. [Alternative 1].	Child has been chosen will move to “Danh sách đã được thêm” and “Green button” button will change to “Red button”.
4	User click “Ok” button. [Alternative 2]	All child in “Danh sách đã được thêm” will be assigned or unassigned to chosen class. Information will update to database.

### Alternative Scenario:

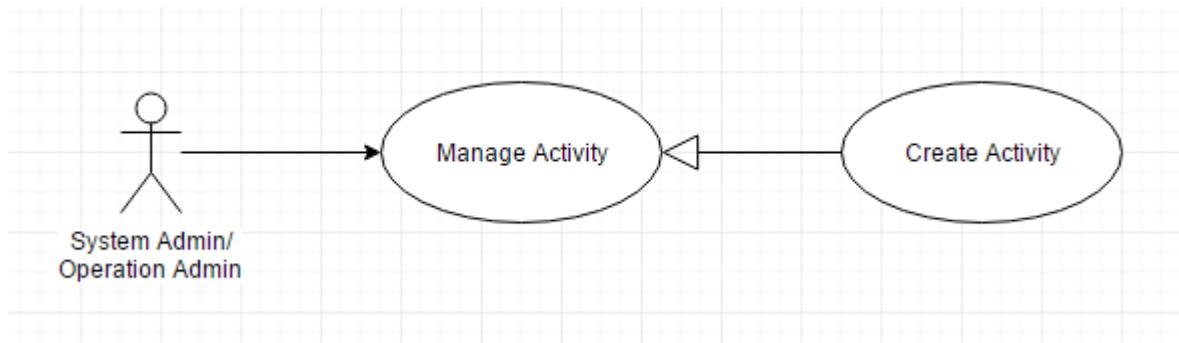
No	Actor Action	System Response
1	User click “Red button” button of an assigned child need to remove from chosen class.	Child has been chosen will move to “Danh sách có thể thêm” and “Red button” will change to “Green button”.
2	User click “Hủy” button.	System will close the current pop up.

### Relationships: Manage Class

### Business Rules:

- Class has been choose to assign child must be in active state.

### 2.2.1.22. <System Admin & Operation Admin> Create Activity



#### USE CASE - UC 22

<b>Use Case No.</b>	UC22	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Create Activity		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

#### Actor:

- System Admin, Operation Admin.

#### Summary:

- This use case allows system user to create an activity in the system.

#### Goal:

- New activity will be created into the database.

#### Triggers:

- User clicks “Quản lý hoạt động” link in the menu.

#### Preconditions:

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** New activity will be creating to the database.
- **Fail:**
  - + Nothing added to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click "Tạo mới hoạt động" button.	Show "Tại mới hoạt động" pop up which contains: <ul style="list-style-type: none"> <li>- "Tiêu đề": textbox, required.</li> <li>- "Nội dung": textbox.</li> <li>- "Địa điểm": textbox, required.</li> <li>- "Ngày bắt đầu": textbox, date, required, min = today.</li> <li>- "Ngày kết thúc": textbox, date, required</li> <li>- "Tạo": button.</li> <li>- "Hủy": button.</li> </ul>
2	User fill data to the form User click "Tạo" button. [Alternative Scenario 1] [Exception 1]	Insert new activity into the database.

**Alternative Scenario:**

No	Actor Action	System Response

1 User click “Hủy” button.

System closes the current pop up.

### Exceptions:

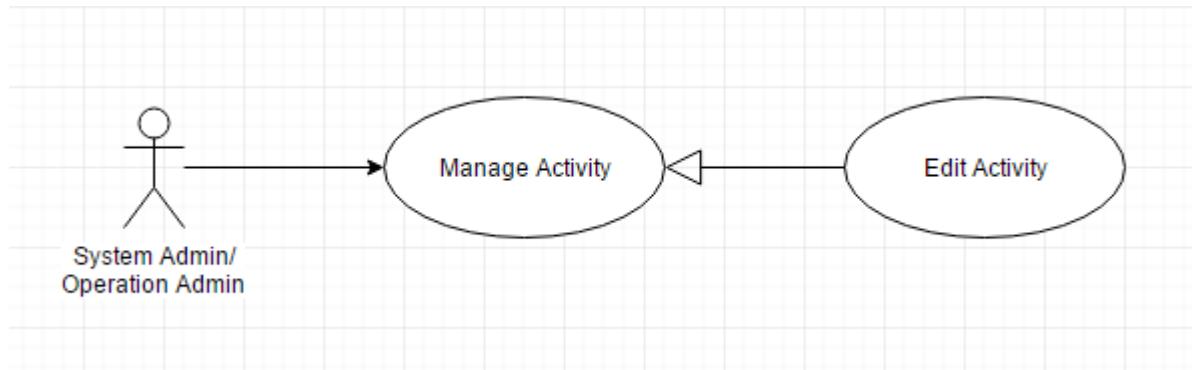
No	Actor Action	System Response
1	Data of “Ngày bắt đầu” greater than to day.	Show error message: “Ngày bắt đầu nhỏ nhất là hôm nay!”

### Relationships: Manage Activity

#### Business Rules:

- Each added activity must be in active state when insert to database.

### 2.2.1.23. <System Admin & Operation Admin> Edit Information of Activity



### USE CASE – UC 23

<b>Use Case No.</b>	UC23	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Edit Information of Activity		
<b>Author</b>	Đoàn Hiếu Nghĩa		

<b>Date</b>	24/07/2016	<b>Priority</b>	Normal
-------------	------------	-----------------	--------

**Actor:**

- System Admin, Operation Admin.

**Summary:**

- This use case allows system user to edit information of an existing activity in the system.

**Goal:**

- Information of chosen activity will be updated to the system.

**Triggers:**

- User clicks “Sửa” button of the chosen activity on activity list in the “Quản lý hoạt động” page.

**Preconditions:**

- User must log in the system with system admin or operation admin role.

**Post Conditions:**

- **Success:** Information of the chosen activity will be updated to the database.
- **Fail:**
  - + Information is not updated to the database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	Click "Sửa" button.	<p>Show “Thông tin hoạt động” pop up which contains:</p> <ul style="list-style-type: none"> <li>- “Tiêu đề”: textbox, required.</li> <li>- “Nội dung”: textbox.</li> <li>- “Địa điểm”: textbox, required.</li> <li>- “Ngày bắt đầu”: textbox, date, required, min = today.</li> <li>- “Ngày kết thúc”: textbox, date, required</li> <li>- “Tạo”: button.</li> <li>- “Hủy”: button.</li> </ul>

		All field have data of the chosen activity.
2	User edit data in the form	
3	User click “Lưu” button. [Alternative 1] [ Exception 1]	Update new information of chosen activity into the database.

### Alternative Scenario:

No	Actor Action	System Response
1	User click “Hủy” button.	System closes the current pop up.

### Exceptions:

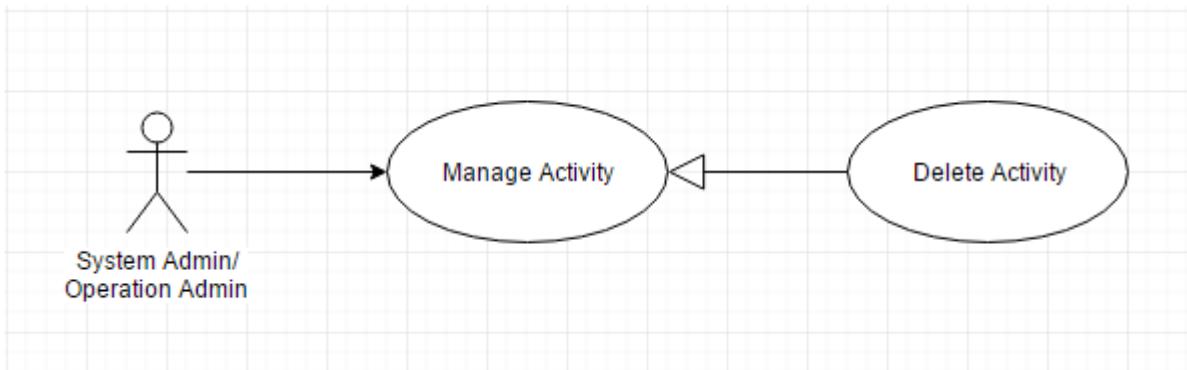
No	Actor Action	System Response
1	Data of “Ngày bắt đầu” greater than to day.	Show error message: “Ngày bắt đầu nhỏ nhất là hôm nay!”

### Relationships: Manage Activity

#### Business Rules:

- Activity has been choose to edit must be in active state.

#### 2.2.1.24. <System Admin & Operation Admin> Delete Activity



## USE CASE – UC 24

<b>Use Case No.</b>	UC24	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Delete Activity		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

### Actor:

- System Admin, Operation Admin.

### Summary:

- This use case allows system user to delete an existing activity in the system.

### Goal:

- Chosen activity will be deactivating to the system.

### Triggers:

- User clicks “Xóa” button of the chosen activity on activity list in the “Quản lý hoạt động” page.

### Preconditions:

- User must log in the system with system admin or operation admin role.

### Post Conditions:

- **Success:** Status of chosen activity will be change to deactivate in database.
- **Fail:**
  - + Status is not updated to the database.
  - + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
CCSS Project	Page 79	

1 User click button “Xóa”.

Show alert message confirm: “Bạn muốn xóa hoạt động này?”

2 User click “Ok” button.  
[Alternative 1]

Change status of chosen activity to deactivate in database.

#### **Alternative Scenario:**

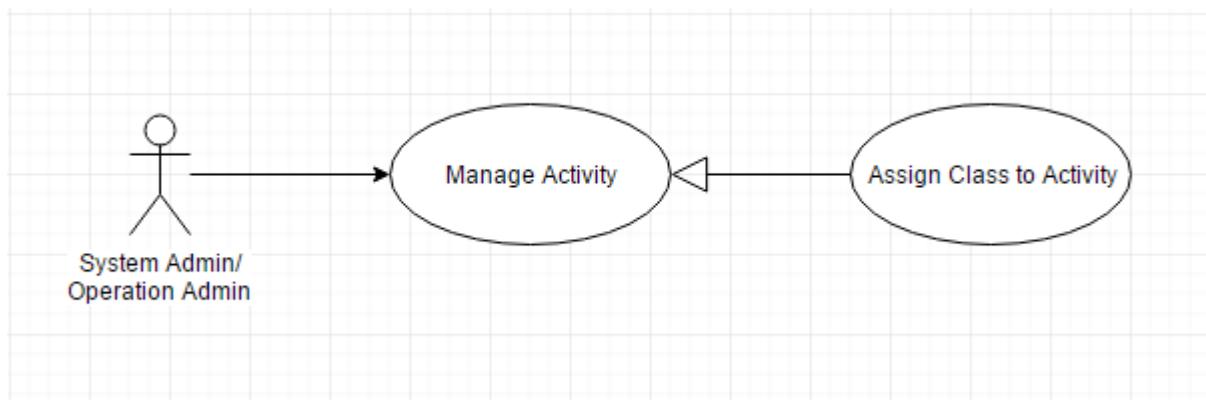
No	Actor Action	System Response
1	User clicks “Hủy” button	
		System closes the alert message.

#### **Relationships:** Manage Activity

#### **Business Rules:**

- Activity has been choose to delete must be in active state.

#### **2.2.1.25. <System Admin & Operation Admin> Assign Class to Activity**



## USE CASE – UC 25

<b>Use Case No.</b>	UC25	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Assign Class to Activity		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

### Actor:

- System Admin, Operation Admin.

### Summary:

- This use case allows system user to assign existing class to an activity in the system.

### Goal:

- Class will be assign to chosen activity.

### Triggers:

- User clicks “Chọn lớp” button of the chosen activity on activity list in the “Quản lý hoạt động” page.

### Preconditions:

- User must log in the system with system admin or operation admin role.

### Post Conditions:

- **Success:** Information of class will be assign to the chosen activity.
- **Fail:**
  - + Information will be creating to the database.
  - + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
1	User click “Chọn lớp” button.	<p>Show “Thêm lớp tham gia hoạt động” pop up which contains:</p> <ul style="list-style-type: none"> <li>- Information of chosen activity:           <ul style="list-style-type: none"> <li>+ “Tiêu đề”: title of chosen activity.</li> <li>+ “Nội dung hoạt động”: content of the chosen activity.</li> <li>+ “Địa điểm”: location of the chosen activity.</li> <li>+ “Ngày bắt đầu”: start date of chosen activity.</li> <li>+ “Ngày kết thúc”: end date of chosen activity.</li> </ul> </li> <li>- “Danh sách lớp tham gia”: table, list of class assigned in chosen activity, which contains:           <ul style="list-style-type: none"> <li>+ “Tên lớp”: name of class.</li> <li>+ “Khối lớp”: grade of class.</li> <li>+ “Ngày bắt đầu”: start date of class.</li> <li>+ “Ngày kết thúc”: end date of class.</li> <li>+ “Red button”: button.</li> </ul> </li> <li>- “Danh sách lớp chưa tham gia”: table, list of class unassigned in chosen activity, which contains:           <ul style="list-style-type: none"> <li>+ “Tên lớp”: name of class.</li> <li>+ “Khối lớp”: grade of class.</li> <li>+ “Ngày bắt đầu”: start date of class.</li> <li>+ “Ngày kết thúc”: end date of class.</li> <li>+ “Green button”: button.</li> </ul> </li> <li>- “Search”: textbox.</li> <li>- “Hủy”: button.</li> </ul>

3	User click "Green button" button of an unassigned class need to assign to chosen activity. [Alternative 1].	Class has been chosen will move to "Danh sách lớp tham gia" and "Green button" button will change to "Red button". Information of this class assign to activity will update to database
4	User click "Hủy" button.	System closes the current popup.

### **Alternative Scenario:**

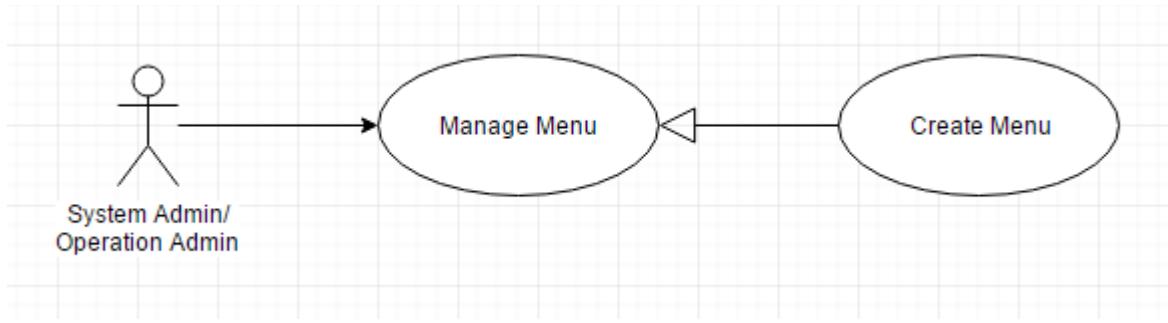
No	Actor Action	System Response
1	User click "Red button" button of an assigned class need to remove from chosen activity.	Class has been chosen will move to "Danh sách lớp chưa tham gia" and "Red button" will change to "Green button".

### **Relationships:** Manage Activity

### **Business Rules:**

- Activity has been choose to assign class must be in active state.

### **2.2.1.26. <System Admin & Operation Admin> Create Menu**



## USE CASE – UC 26

<b>Use Case No.</b>	UC26	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Create Menu		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

### Actor:

- System Admin, Operation Admin.

### Summary:

- This use case allows system user to create an activity in the system.

### Goal:

- New menu will be created into the database.

### Triggers:

- User clicks “Thực đơn” link in the menu.

### Preconditions:

- User must log in the system with system admin or operation admin role.

### Post Conditions:

- **Success:** New menu will be creating to the database.
- **Fail:**
  - + Nothing added to the database.
  - + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
1	User click “Tạo mới” button.	Show “Tại mới thực đơn” pop up which

contains:

- “Nội dung”: textbox, required.
- “Ngày bắt đầu”: textbox, date, required, min = today.
- “Ngày kết thúc”: textbox, date, required
- “Tạo”: button.
- “Hủy”: button.

3 User fill data to the form

User click “Tạo” button.

[Alternative Scenario

1] [Exception 1]

Insert new activity into the database.

Redirect to create menu schedules page which contains:

- Information of new menu:
  - + “Tên menu”: content of new menu.
  - + “Từ ngày”: start date of new menu.
  - + “Đến ngày”: end date of new menu.
- Table create menu schedule:
  - + “Ngày”: labels, dates of schedule, auto create from start date to end date of menu.
  - + “Bữa sáng”: textbox.
  - + “Bữa trưa”: textbox.
  - + “Bữa xế”: textbox.
  - + “Lưu menu”: button.

3

Insert menu schedules of new menu to database.

User click “Lưu menu” button. [Exception 1]

**Alternative Scenario:**

No	Actor Action	System Response
1	User click “Hủy” button.	System closes the current pop up.

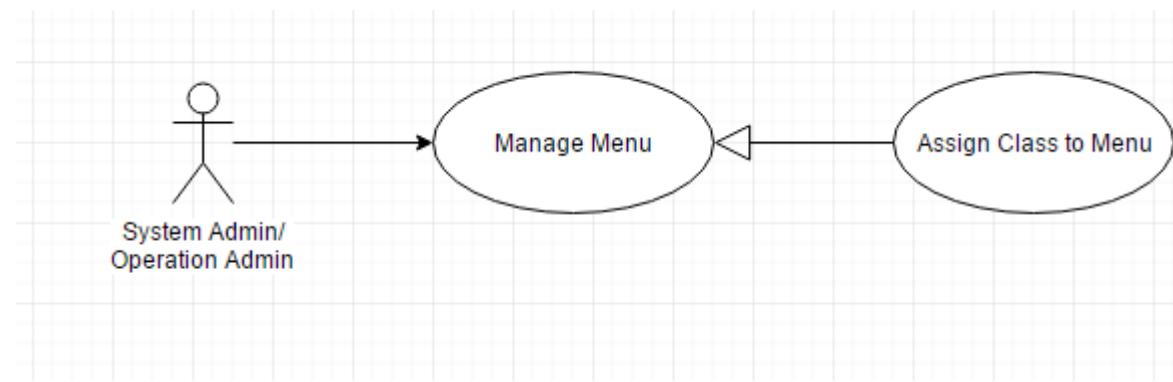
**Exceptions:**

No	Actor Action	System Response
1	Data of “Ngày bắt đầu” greater than to day.	Show error message: “Ngày bắt đầu nhỏ nhất là hôm nay!”

**Relationships:** Manage Menu**Business Rules:**

- Each added menu and menu schedule must be in active state when insert to database.

**2.2.1.27. <System Admin & Operation Admin> Assign Class to Menu**



## USE CASE – UC 27

<b>Use Case No.</b>	UC27	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Assign Class to Menu		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

### Actor:

- System Admin, Operation Admin.

### Summary:

- This use case allows system user to assign existing class to a menu in the system.

### Goal:

- Class will be assign to chosen menu.

### Triggers:

- User clicks name of the chosen menu on menu list in the “Thực đơn” page.

### Preconditions:

- User must log in the system with system admin or operation admin role.

### Post Conditions:

- **Success:** Information of class will be assign to the chosen menu.
- **Fail:**
  - + Information will be creating to the database.
  - + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
------	--------------	-----------------

- 1 User click name of chosen menu.

Show detail menu page which contains:

- Information of chosen menu:
- Table creates menu schedules.
- “Chọn lớp học cho menu”: table, list of class assigned in chosen menu:
  - + “Tên lớp”: name of class.
  - + “Ngày bắt đầu”: start date of class.
  - + “Ngày kết thúc”: end date of class.
  - + “Khối”: Grade of class
- “Thêm lớp học”: button.

- 3 User click “Thêm lớp học”

Show “Danh sách lớp” pop up which contains:

- “Danh sách lớp đã được thêm”: table, list of class assigned in chosen menu, which contains:
  - + “Tên lớp”: name of class.
  - + “Khối”: grade of class.
  - + “Ngày bắt đầu”: start date of class.
  - + “Ngày kết thúc”: end date of class.
  - + “Red button”: button.
- “Danh sách lớp chưa được thêm”: table, list of class unassigned in chosen menu, which contains:
  - + “Tên lớp”: name of class.
  - + “Khối”: grade of class.
  - + “Ngày bắt đầu”: start date of class.
  - + “Ngày kết thúc”: end date of class.
  - + “Green button”: button.

		<ul style="list-style-type: none"> <li>- “Search”: textbox.</li> <li>- “Ok”: button.</li> <li>- “Hủy”: button.</li> </ul>
4	User click “Green button” button of an unassigned class need to assign to chosen menu. [Alternative 1].	Class has been chosen will move to “Danh sách lớp đã được thêm” and “Green button” button will change to “Red button”.
5	User click “Ok” button. [Alternative 2].	Update information of all class assigned in chosen menu to database.

### Alternative Scenario:

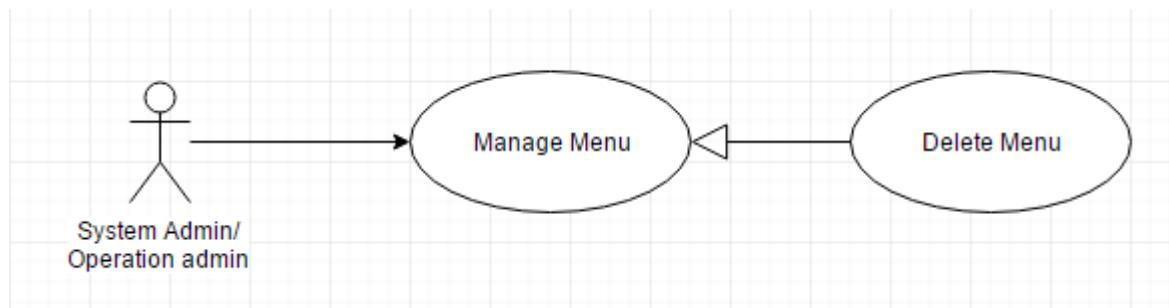
No	Actor Action	System Response
1	User click “Red button” button of an assigned class need to remove from chosen activity.	Class has been chosen will move to “Danh sách lớp chưa tham gia” and “Red button” will change to “Green button”.
2	User click “Hủy” button.	System will close the current pop up.

### Relationships: Manage Menu

## **Business Rules:**

- Menu has been choose to assign class must be in active state.

### **2.2.1.28. <System Admin & Operation Admin> Delete Menu**



## **USE CASE – UC 28**

<b>Use Case No.</b>	UC28	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Delete Menu		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal
<b>Actor:</b>			

- System Admin, Operation Admin.

### **Summary:**

- This use case allows system user to delete an existing menu in the system.

### **Goal:**

- Chosen menu will be deactivating to the system.

### **Triggers:**

- User clicks “Xóa” button of the chosen menu on menu list in the “Thực đơn” page.

### **Preconditions:**

- User must log in the system with system admin or operation admin role.

### **Post Conditions:**

- **Success:** Status of chosen menu and its menu schedules will be change to deactivate in database.
- **Fail:**

- + Status is not updated to the database.
- + System show error message.

#### **Main Success Scenario:**

Step	Actor Action	System Response
1	User click button “Xóa”.	Show alert message confirm: “Bạn muốn xóa thực đơn này?”
2	User click “Ok” button. [Alternative 1]	Change status of chosen menu and its menu schedules to deactivate in database.

#### **Alternative Scenario:**

No	Actor Action	System Response
1	User clicks “Hủy” button	System closes the alert message.

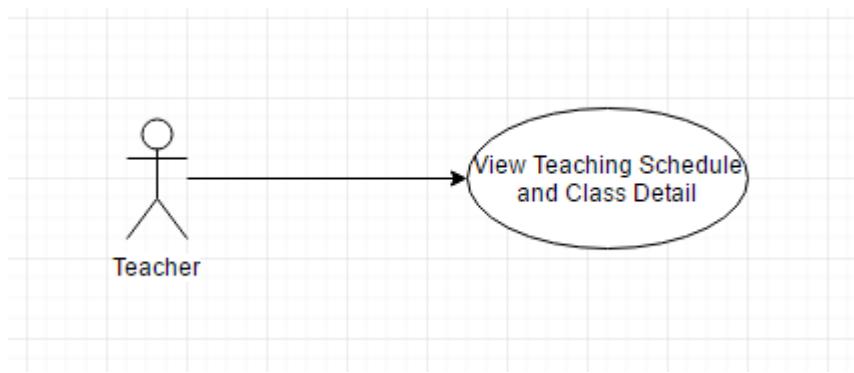
#### **Relationships:** Manage Menu

#### **Business Rules:**

- Menu has been choose to delete must be in active state.

#### **2.2.2. <Teacher> Overview Use Cases**

##### **2.2.2.1. <Teacher> View Teaching Schedule and Class detail**



## USE CASE – UC 29

<b>Use Case No.</b>	UC29	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	View Teaching Schedule and Class detail		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

### Actor:

- Teacher.

### Summary:

- This use case allows teacher to view teaching schedule and class detail.

### Goal:

- Teacher can view current class, class list in the pass and view information of class that current teacher has taught.

### Triggers:

- User clicks “Lịch giảng dạy” link into “Giáo viên” button on menu at home page.

### Preconditions:

- User must log in the system with teacher role.

### Post Conditions:

- **Success:** Data of current class and class list in the past will be show for user.
- **Fail:**
  - + Data can't load from server.
  - + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
1	User click link “Lịch giảng dạy” into “Giáo	

viên” button on menu. Show “Lịch giảng dạy” page which contains:

- Tab “Lớp hiện tại”, contains:
  - + “Tên lớp”: label, name of class.
  - + “Khối lớp”: label, grade of class.
  - + “Phòng”: label, location of class.
  - + “Thời gian”: label, start date and end date of class.
  - + “Giáo viên”: label, current teacher's name of class.
  - + “Mô tả”: label, description of class.
  - + “Chi tiết”: button.
- Tab “Danh sách lớp”, contains table with data of classes that current teacher is teaching and has taught in the past:
  - + “Tên lớp”: name of class.
  - + “Khối lớp”: grade of class.
  - + “Phòng”: location of class.
  - + “Thời gian”: start date and end date of class.
  - + “Giáo viên”: current teacher's name of class.
  - + “Mô tả”: description of class.

2 User click “Chi tiết” button. [Alternative 1]

Show class detail page which contains:

- Information of class:
  - + Class name as title, big text.
  - + “Khối”: label, grade of class.
  - + “Phòng”: label, location of class.
  - + “Khai giảng”: start date of class.
  - + “Kết thúc”: end date of class.
  - + “Điểm danh hôm nay”: button.
  - + “Kiểm tra sức khỏe”: button.
  - + “Tạo báo cáo”: button.
- “Danh sách trẻ” contains:
  - + Tab “Bình thường” contains table as list of children have normal weight in current class.
  - + Tab “Thiếu cân” contains table as

3

- list of children have underweight in current class.
- + Tab “Thùa cân” contains table as list of children have overweight in current class.
- “Hoạt động của lớp” as list activities of current class, contains:
  - + “Hoạt động”: title of activity.
  - + “Địa điểm”: location of activity.
  - + “Thời gian”: start date and end date of activity.
  - + “Mô tả”: description of activity.
- “Thực đơn của lớp” as list menu of current class, contains:
  - + “Ngày”: date of menu.
  - + “Bữa sáng”: dish for breakfast.
  - + “Bữa trưa”: dish for afternoon.
  - + “Bữa xế”: dish for lunch.
- “Trở lại danh sách”: button.

[Return to “Lịch giảng dạy” page.](#)

User click “Trở lại danh sách” button.

#### **Alternative Scenario:**

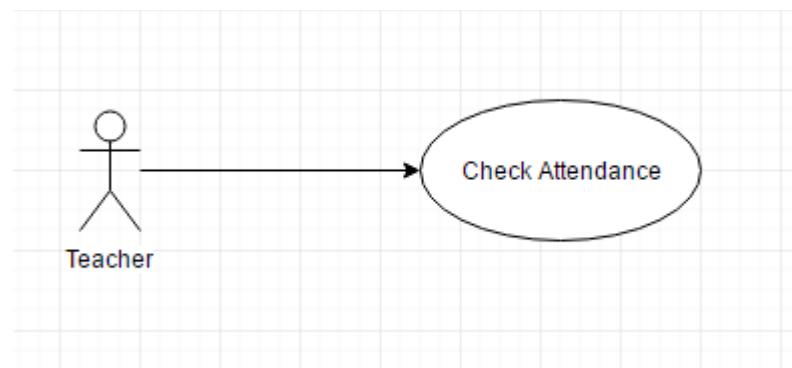
No	Actor Action	System Response
1	User clicks on class information row in tab “Danh sách lớp”.	Show class detail page.

**Relationships:** View Teaching Schedule and Class detail.

### **Business Rules:**

- Class choose to view detail must be in active state.

#### **2.2.2.2. <Teacher> Check Attendance**



#### **USE CASE – UC 30**

<b>Use Case No.</b>	UC30	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Check attendance.		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal
<b>Actor:</b>			

- Teacher.

#### **Summary:**

- This use case allows teacher to check attendance.

#### **Goal:**

- Teacher can check attendance of current date for children.

#### **Triggers:**

- User clicks “Điểm danh hôm nay” button on class detail page.

#### **Preconditions:**

- User must log in the system with teacher role.

#### **Post Conditions:**

- **Success:** Attendance of current date for children will be created to database.
- **Fail:**

- + Nothing added to database.
- + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
1	User click “Điểm danh hôm nay” button in class detail page.	<p>Show “Điểm danh cho lớp” page which contains:</p> <ul style="list-style-type: none"> <li>- Date of today.</li> <li>- Class information:           <ul style="list-style-type: none"> <li>+ Class name as title.</li> <li>+ “Khối”: label, grade of class.</li> <li>+ “Phòng”: label, location of class.</li> <li>+ “Khai giảng”: start date of class.</li> <li>+ “Kết thúc”: end date of class.</li> </ul> </li> <li>- Table list child in current class contains:           <ul style="list-style-type: none"> <li>+ “Tên bé”: name of child.</li> <li>+ “Ngày sinh”: birthday of child.</li> <li>+ “Giới tính”: gender of child.</li> <li>+ “Phụ huynh”: parent's name of child.</li> <li>+ “Số đt phụ huynh”: parent's phone number of child.</li> <li>+ “Điểm danh”: contains:               <ul style="list-style-type: none"> <li>▪ “Có mặt”: button.</li> <li>▪ “Vắng mặt”: button.</li> </ul> </li> </ul> </li> <li>- “Trở về”: link</li> </ul>
2	User click “Có mặt” button of a child. [Alternative 1, 2]	<p>Insert to database attendance of chosen child.</p> <p>Show success alert “Điểm danh có mặt”.</p>
3	User click “Trở về” link	Return to class detail page.

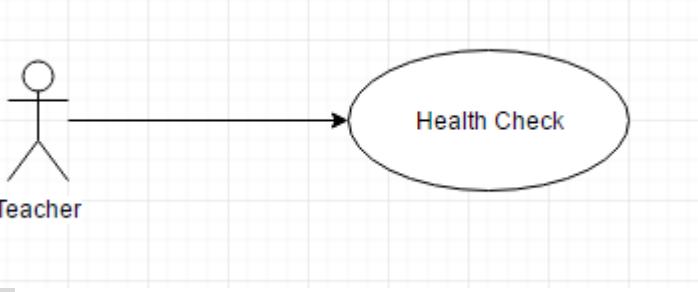
**Alternative Scenario:**

No	Actor Action	System Response
1	User clicks on “Vắng mặt” button.	Show “Lý do vắng mặt” pop up which contains: <ul style="list-style-type: none"><li>- “Lý do vắng mặt”: textbox.</li><li>- “Cập nhật”: button.</li><li>- “Hủy”: button.</li></ul>
2	User fill “Lý do vắng mặt” text box.  User click “Cập nhật” button.  [Alternative 2.1]	Insert to database attendance off chosen child.  Show success alert “Điểm danh vắng mặt”.
2.1	User click “Hủy” button.	System closes the current pop up.

**Relationships:** Check Attendance**Business Rules:**

- Each child chooses to check attendance must be in active state.

**2.2.2.3. <Teacher> Health check**

<b>USE CASE – U</b>	
<b>Use Case No.</b>	
<b>Use Case Name</b>	
<b>Author</b>	
<b>Date</b>	24/07/2016
<b>Priority</b>	Normal

#### Actor:

- Teacher.

#### Summary:

- This use case allows teacher to check child's health.

#### Goal:

- Teacher can check health of children and see the current health's situation.

#### Triggers:

- User clicks "Kiểm tra sức khỏe" button on class detail page.

#### Preconditions:

- User must log in the system with teacher role.

#### Post Conditions:

- **Success:** New health check will be creating to database.

#### Fail:

- + Nothing added to database.
- + System show error message.

#### Main Success Scenario:

Step	Actor Action	System Response
1	User click "Điểm danh hôm nay" button in class detail page.	Show "Thông tin kiểm tra sức khỏe cho lớp" page which contains: <ul style="list-style-type: none"> <li>- Class information:               <ul style="list-style-type: none"> <li>+ Class name as title.</li> <li>+ "Khối": label, grade of class.</li> <li>+ "Phòng": label, location of class.</li> <li>+ "Khai giảng": start date of</li> </ul> </li> </ul>

- class.
  - + “Kết thúc”: end date of class.
- “Nhập dữ liệu kiểm tra sức khỏe”: button.
- “Chọn ngày kiểm tra sức khỏe”: dropdown list, contains all health checking date.
- Table child’s health check in the past of current class contains:
  - + “Tên bé”: name of child.
  - + “Tuổi”: age of child.
  - + “Giới tính”: gender of child.
  - + “Cân nặng”: weight of child.
  - + “Chiều cao”: height of child.
  - + “Tình trạng”: health’s condition of child.
  - + “Định mức sức khỏe”: child’s assessment of weight and height.
  - + “Cập nhật lần cuối”: the last updated date of this health’s check.
- “Trở về”: link.

- 2 User chooses a date in “Chọn ngày kiểm tra sức khỏe” dropdown list.

Show all health check at the chosen date.

- 3 User click “Nhập dữ liệu kiểm tra sức khỏe” button.  
[Alternative 1]

Show “Kiểm tra sức khỏe cho lớp” page, which contains:

- Class information:
  - + Class name as title.
  - + “Khối”: label, grade of class.
  - + “Phòng”: label, location of class.
  - + “Khai giảng”: start date of class.

- + “Kết thúc”: end date of class.
- Table child's health check in the past of current class contains:
  - + “Tên bé”: name and age of child.
  - + “Giới tính”: gender of child.
  - + “Cân nặng”: textbox, required.
  - + “Chiều cao”: textbox, required.
  - + “Tình trạng”: textbox.
  - + “Không thể kiểm tra”: checkbox.
  - + “Định mức sức khỏe”: child's assessment of weight and height.
  - + “Cập nhật lần cuối”: the last updated date of this health's check.
- “Lưu lại”: button.
- “Trở về”: link.

4

User fill data to table child's health check.[Alternative 2]

User click “Lưu lại” button.  
[Alternative 1]

Insert to database all health check.

Show child's assessment of weight and height into “Định mức sức khỏe” of each row.

#### **Alternative Scenario:**

No	Actor Action	System Response
1	User clicks “Trở về” link.	Return to class detail page.

- 2 User checks “Không thể kiểm tra” checkbox.

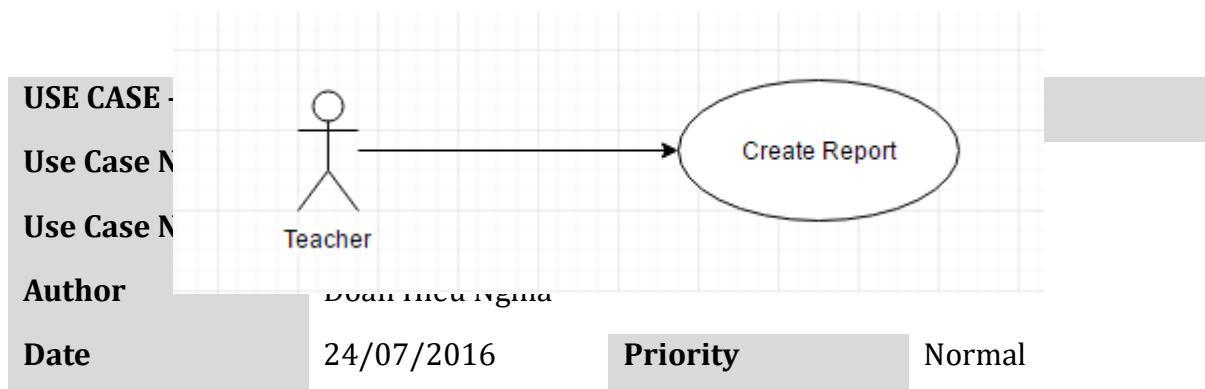
“Chiều cao”, “Cân nặng”, “Tình trạng” of health check are disabled and system won’t insert this health check to database.

### **Relationships:** Health check

### **Business Rules:**

- Each added health check must be in active state when insert to database.

### **2.2.2.4. <Teacher> Create Report**



#### **Actor:**

- Teacher.

#### **Summary:**

- This use case allows teacher to create report.

#### **Goal:**

- Teacher can create report for child.

#### **Triggers:**

- User clicks “Tạo báo cáo” button on class detail page.

#### **Preconditions:**

- User must log in the system with teacher role.

#### **Post Conditions:**

- **Success:** New report will be creating to database.

- **Fail:**

- + Nothing added to database.

- + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	User click “Tạo báo cáo” button in class detail page.	<p>Show “Tạo báo cáo cho lớp” page which contains:</p> <ul style="list-style-type: none"> <li>- Class name as title.</li> <li>- “Loại và thời gian báo cáo” form contains: <ul style="list-style-type: none"> <li>+ “Loại báo cáo”: dropdown list, type of report.</li> <li>+ “Thời gian báo cáo”: dropdown list, time of report.</li> <li>+ Label start date and end date of report</li> </ul> </li> <li>- Table list child in current class, contains: <ul style="list-style-type: none"> <li>+ “Tên bé”: name of child.</li> <li>+ “Ngày sinh”: birthday of child.</li> <li>+ “Giới tính”: gender of child.</li> <li>+ “Ghi chú”: description of child.</li> <li>+ “Nhận xét chung”: comment of teacher for child.</li> <li>+ “Tạo báo cáo cho bé”: button.</li> </ul> </li> <li>- “Trở về”: link.</li> </ul>
2	User chooses options in 2 dropdown list “Loại báo cáo” and “Thời gian báo cáo”.  User click “Tạo báo cáo cho bé” button. [Alternative 1]	Show “Nhận xét cho bé” pop up, which

contains:

- “Thông tin kiểm tra sức khỏe” contains:
  - + Date of health check.
  - + “Cân nặng”: label, weight of child.
  - + “Chiều cao”: label, height of child.
  - + “Lưu ý”: health condition of child.
  - + “Xếp loại”: rating of child.
- “Thông tin tiêm chủng” contains data of incoming immunization schedule.
- “Nhận xét chung” contains:
  - + “Nhận thức”: textarea, required.
  - + “Ngôn ngữ”: text area, required.
  - + “Vận động – Tự phục vụ bản thân”: text area, required.
  - + “Lễ phép – Giao tiếp”: textarea, required.
  - + “Thẩm mỹ - năng khiếu”: textarea, required.
  - + “Nhận xét chung – Trao đổi với phụ huynh”: textarea, required.
- “Lưu lại”: button.
- “Trở về”: link.

User fills data to the form.

- 4      User click “Lưu lại” button.  
[Alternative 2]

Insert to database new report.

### **Alternative Scenario:**

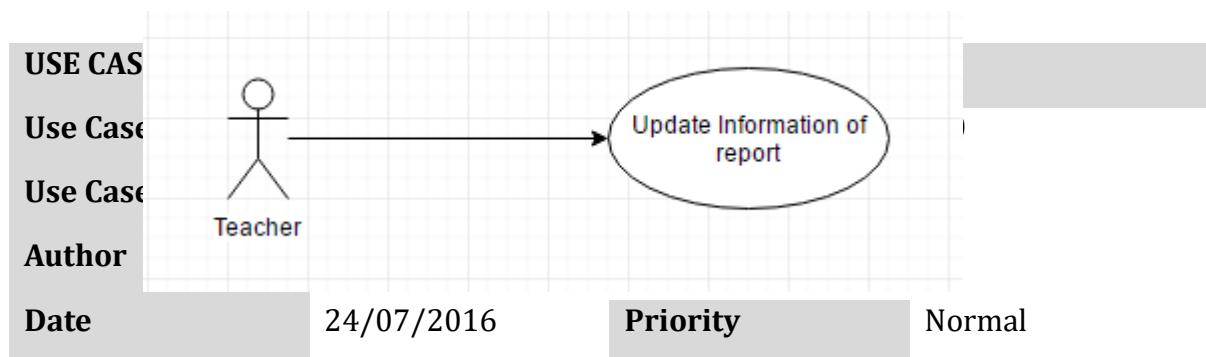
No	Actor Action	System Response
1	User clicks “Trở về” link.	Return to class detail page.
2	User click “Hủy” button.	System closes current pop up.

### **Relationships:** Create Report

### **Business Rules:**

- Each added report must be in active state when insert to database.

### **2.2.2.5. <Teacher> Update Information of Report**



### **Actor:**

- Teacher.

### **Summary:**

- This use case allows teacher to update report.

### **Goal:**

- Teacher can update information of report.

### **Triggers:**

- User clicks “Báo cáo” link into “Giáo viên” button on menu.

### **Preconditions:**

- User must log in the system with teacher role.

### **Post Conditions:**

- **Success:** Information of report will be updated to database.
- **Fail:**
  - + Nothing updated to database.
  - + System show error message.

### **Main Success Scenario:**

Step	Actor Action	System Response
1	User click “Báo cáo” link into “Giáo viên” button on menu.	Show “Danh sách báo cáo” page which contains: <ul style="list-style-type: none"><li>- Class name as title.</li><li>- Table list of report contains:<ul style="list-style-type: none"><li>+ “Tiêu đề”: title of report.</li><li>+ “Thời gian”: start date and end date of report.</li><li>+ “Loại báo cáo”: type of report.</li><li>+ “Cập nhật”: button.</li><li>+ “Xóa”: button.</li></ul></li></ul>
2	User clicks “Cập nhật” button.	Show “Chỉnh sửa báo cáo” page, which contains: <ul style="list-style-type: none"><li>- “Thông tin kiểm tra sức khỏe” contains:<ul style="list-style-type: none"><li>+ Date of health check.</li><li>+ “Cân nặng”: label, weight of child.</li><li>+ “Chiều cao”: label, height of child.</li><li>+ “Lưu ý”: health condition of child.</li><li>+ “Xếp loại”: rating of child.</li></ul></li><li>- “Thông tin tiêm chủng” contains data of incoming immunization schedule.</li><li>- “Nhận xét chung” contains:<ul style="list-style-type: none"><li>+ “Nhận thức”: textarea,</li></ul></li></ul>

- required.
- + “Ngôn ngữ”: text area, required.
  - + “Vận động – Tự phục vụ bản thân”: text area, required.
  - + “Lễ phép – Giao tiếp”: textarea, required.
  - + “Thẩm mỹ - năng khiếu”: textarea, required.
  - + “Nhận xét chung – Trao đổi với phụ huynh”: textarea, required.
  - “Lưu lại”: button.
  - “Trở về”: link.

All fields have detail of chosen report.

3 User edits information of report.

User click “Lưu lại” button.

[Alternative 1]

Update information of report to database.

Show success alert “Cập nhật thành công”.

Return to “Danh sách báo cáo” page.

#### **Alternative Scenario:**

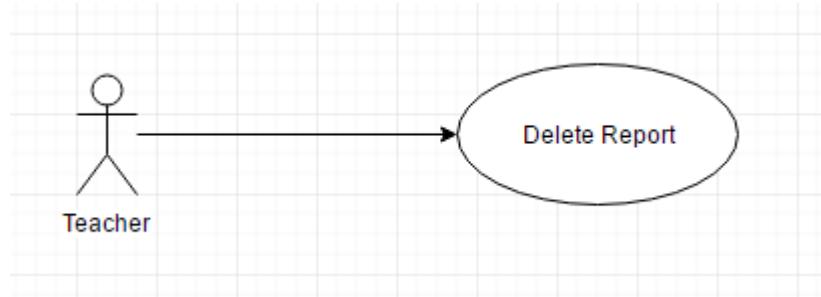
No	Actor Action	System Response
1	User clicks “Hủy” button.	Return to “Danh sách báo cáo” page.

#### **Relationships:** Update Information of Report

#### **Business Rules:**

- Report choose to update must be in active state when insert to database.

### 2.2.2.6. <Teacher> Delete Report



#### USE CASE - UC 34

<b>Use Case No.</b>	UC34	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	Delete Report.		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

#### Actor:

- Teacher.

#### Summary:

- This use case allows teacher to delete an existing report in the system.

#### Goal:

- Teacher can delete report.

#### Triggers:

- User clicks “Báo cáo” link into “Giáo viên” button on menu.

#### Preconditions:

- User must log in the system with teacher role.

#### Post Conditions:

- **Success:** Status of the chosen report will be change to deactivate in database.
- **Fail:**
  - + Nothing changed to database.
  - + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
1	User click “Báo cáo” link into “Giáo viên” button on menu.	Show “Danh sách báo cáo” page which contains:  <ul style="list-style-type: none"> <li>- Class name as title.</li> <li>- Table list of report contains: <ul style="list-style-type: none"> <li>+ “Tiêu đề”: title of report.</li> <li>+ “Thời gian”: start date and end date of report.</li> <li>+ “Loại báo cáo”: type of report.</li> <li>+ “Cập nhật”: button.</li> <li>+ “Xóa”: button.</li> </ul> </li> </ul>
2	User clicks “Xóa” button.	Show confirms alert “Bạn có muốn xóa báo cáo này không?”
3	User click “Ok” button. [Alternative 1]	Change status of chosen report to deactivate in database.

### Alternative Scenario:

No	Actor Action	System Response
1	User clicks “Hủy” button.	System closes confirm alert.

### Exceptions:

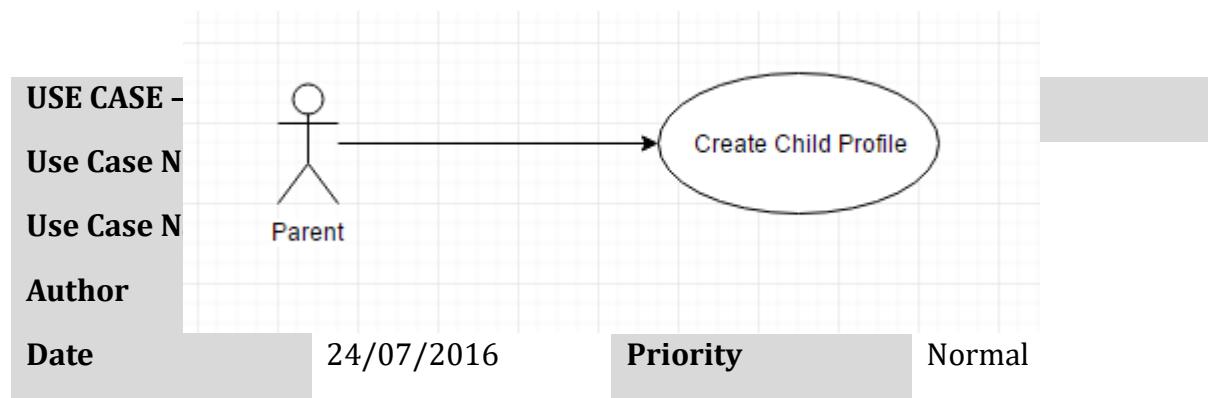
**Relationships:** Delete Report

**Business Rules:**

- Report choose to delete must be in active state when insert to database.

### 2.2.3. <Parent> Overview Use Cases

#### 2.2.3.1. <Parent> Create Child Profile



- Parent.

**Summary:**

- This use case allows parent to create their child's profile.

**Goal:**

- Parent can create profile for their child.

**Triggers:**

- User clicks “Tạo hồ sơ thông tin trẻ” button in “Thông tin trẻ” page.

**Preconditions:**

- User must log in the system with parent role.

**Post Conditions:**

- **Success:** New child profile will be creating to database.
- **Fail:**
  - + Nothing added to database.

- + System show error message.

### Main Success Scenario:

Step	Actor Action	System Response
1	User click “Tạo hồ sơ thông tin trẻ” button in “Thông tin trẻ” page.	<p>Show “Tạo hồ sơ thông tin trẻ” page which contains:</p> <ul style="list-style-type: none"> <li>- “Chọn ảnh đại diện cho bé”: uploadfile, required.</li> <li>- “Tên bé”: textbox, required.</li> <li>- “Tên thường gọi”: textbox.</li> <li>- “Ngày sinh”: textbox, datetime, required.</li> <li>- “Giới tính”: radio button, 2 options (“nam”, “nữ”), required.</li> <li>- “Quan hệ của phụ huynh với bé”: dropdown list, required.</li> <li>- “Sở thích của bé”: textarea.</li> <li>- “Những thú bé sở”: textarea.</li> <li>- “Những thú bé dị ứng hoặc những điều cần lưu ý”: textarea.</li> <li>- “Các kỹ năng của bé” contains: <ul style="list-style-type: none"> <li>+ “Biết tự mặc đồ”: checkbox.</li> <li>+ “Biết thưa gửi/ chào hỏi”: checkbox.</li> <li>+ “Biết tô màu”: checkbox.</li> <li>+ “Biết hát những bài đơn giản”: checkbox.</li> <li>+ “Khác”: checkbox, contains textarea.</li> </ul> </li> <li>- “Ghi chú”: textarea.</li> <li>- “Nhập thông tin phụ huynh thứ 2”: checkbox.</li> <li>- “Tạo”: button.</li> <li>- “Hủy”: button.</li> </ul>
2	User check “Nhập thông tin phụ huynh thứ 2” check box	Show “Thông tin phụ huynh thứ 2” form

below, which contains:

- “Quan hệ của phụ huynh với bé”: dropdown list, required.
- “Họ tên phụ huynh”: textbox, required.
- “Năm sinh”: textbox, datetime, required.
- “Giới tính”: radio button, 2 option (“nam”, “nữ”), required.
- “Số điện thoại liên lạc”: textbox, required.
- “Địa chỉ”: textbox, required.

3 User fill data to the form.

User click “Tạo” button.  
[Alternative 1]

Insert new child profile to database.

Return to “Thông tin trẻ” page.

#### **Alternative Scenario:**

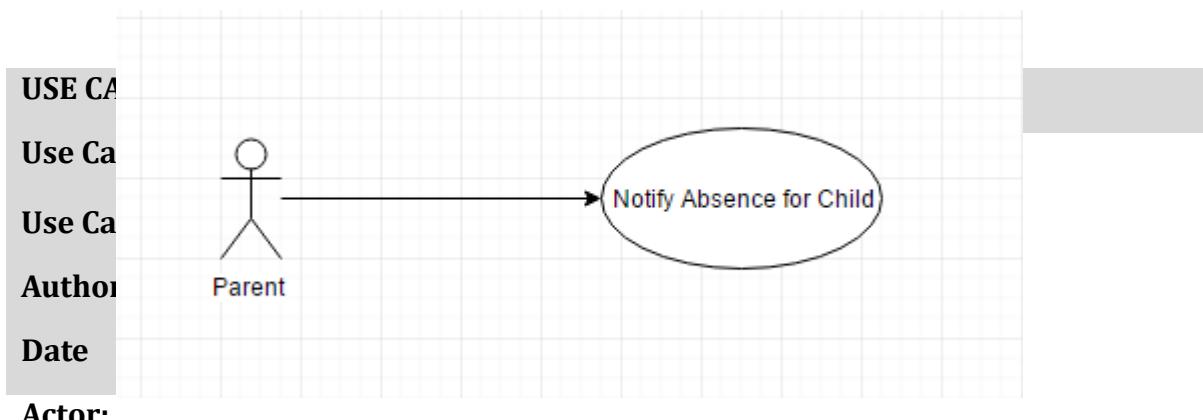
No	Actor Action	System Response
1	User clicks “Hủy” button.	Return to “Thông tin trẻ” page.

#### **Relationships: Create Child Profile**

#### **Business Rules:**

- Each added child profile must be in active state when insert to database.

#### **2.2.3.2. <Parent> Notify absence for child**



**Actor:**

- Parent.

**Summary:**

- This use case allows parent to notify and create absence attendance for their child in a future's date.

**Goal:**

- Parent can notify and create absence attendance for their child.

**Triggers:**

- User clicks “Báo hôm nay bé vắng” button in tab “Thông tin trẻ”.

**Preconditions:**

- User must log in the system with parent role.

**Post Conditions:**

- **Success:** New absence attendance will be created to database.

- **Fail:**

- + Nothing created to database.
- + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	User clicks “Báo hôm nay bé vắng” button in tab “Thông tin trẻ”.	<p>Show “Báo vắng cho trẻ” pop up which contains:</p> <ul style="list-style-type: none"> <li>- “Bé vắng ngày”: textbox, datetime, required, the date that child will be absenced.</li> <li>- “Lý do”: textarea, required, reason</li> </ul>

		that child will be absenced.
	- "Xong": button.	
	- "Hủy": button.	
2	User fill data to the form.	
	User clicks "Xong" button. [Alternative 1]	Insert new absence attendance to database.
3	User clicks "Ok" button.	Show alert message "Báo vắng cho bé thành công".
		Close alert message.

#### Alternative Scenario:

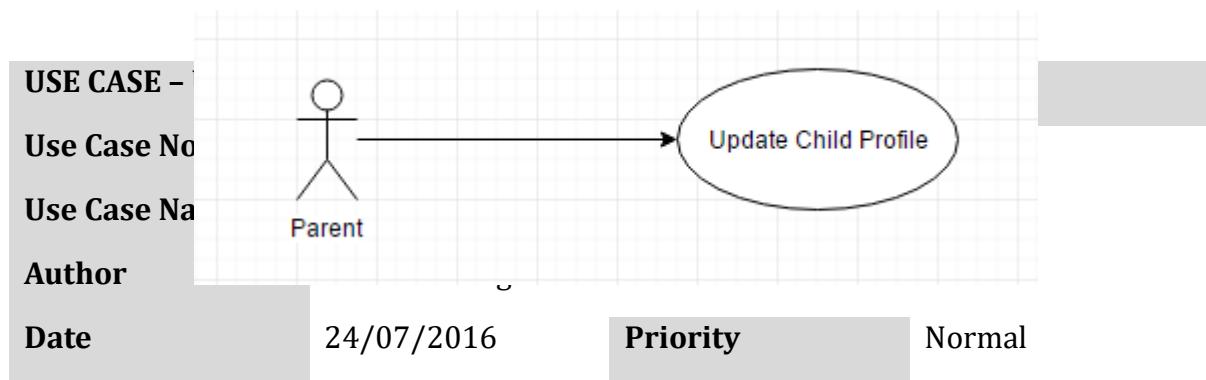
No	Actor Action	System Response
1	User clicks "Hủy" button.	System closes the current pop up.

**Relationships:** Notify absence for child.

#### Business Rules:

- The date create new absence attendate must be in the future.

#### 2.2.3.3. <Parent> Update Child Profile



**Actor:**

- Parent.

**Summary:**

- This use case allows parent to update their child's profile.

**Goal:**

- Parent can update profile for their child.

**Triggers:**

- User clicks "Chi tiết" button of child in "Thông tin trẻ" page to view detail at child detail page and clicks "Chỉnh sửa thông tin của bé" button in tab "Thông tin trẻ".

**Preconditions:**

- User must log in the system with parent role.

**Post Conditions:**

- **Success:** Information of child profile will be updated to database.
- **Fail:**
  - + Nothing update to database.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	User click "Chi tiết" button of child in "Thông tin trẻ" page.	Show child detail page which contains: <ul style="list-style-type: none"><li>- Avatar of child.</li><li>- Name of child, big text.</li><li>- Age of child.</li><li>- "Ngày sinh": label, birthday of child.</li><li>- "Sở thích": label, hobby of child.</li><li>- "Bé sợ": label, fear of child.</li><li>- "Dị ứng hoặc lưu ý về bé": label, allergic of child.</li><li>- "Kỹ năng của bé": label, skill of child.</li><li>- 4 Tab: "Thông tin trẻ", "Thông tin tiêm chủng", "Theo dõi tình hình", "Lịch sử học tập".</li><li>- "Báo bé vắng hôm nay": button.</li></ul>
2	User clicks on tab "Thông tin trẻ".	

Show child detail form which contains:

- “Điểm danh hôm nay”: child's attendance information of today.
- “Lớp” class detail contains:
  - + Class name.
  - + “Giáo viên”: label, name of teacher(s) is (are) teaching at this time.
  - + “Cấp lớp”: label, grade of current class.
  - + “Thời gian”: label, start date and end date of current class.
  - + “Hoạt động trong ngày”: detail of activity of current class today, contains “Xem chi tiết” button.
  - + “Thực đơn ngày”: detail of menu of current class today, contains “Xem chi tiết” button.
- “Phụ huynh” parent's detail, contains:
  - + Parent's name and reference role.
  - + “Ngày sinh”: label, birthday of parent.
  - + “Số điện thoại”: label, phone number of parent.
  - + “Địa chỉ”: label, address of parent.
- “Ghi chú”: textarea, description of child, disabled.
- “Báo hôm nay bé vắng”: button.
- “Chỉnh sửa thông tin của bé”: button.
- “Trở lại danh sách trẻ”: link

Insert new child profile to database.

Return to “Thông tin trẻ page”.

3

User clicks “Chỉnh sửa thông tin của bé” button.  
[Alternative 1]

Show “Chỉnh sửa hồ sơ cho trẻ”, which contains:

- “Chọn ảnh đại diện cho bé”: uploadfile, required.
- “Tên bé”: textbox, required.
- “Tên thường gọi”: textbox.
- “Ngày sinh”: textbox, datetime, required.
- “Giới tính”: radio button, 2 options (“nam”, “nữ”), required.
- “Quan hệ của phụ huynh với bé”: dropdown list, required.
- “Sở thích của bé”: textarea.
- “Những thứ bé sờ”: textarea.
- “Những thứ bé dị ứng hoặc những điều cần lưu ý”: textarea.
- “Các kỹ năng của bé” contains:
  - + “Biết tự mặc đồ”: checkbox.
  - + “Biết thura gửi/ chào hỏi”: checkbox.
  - + “Biết tô màu”: checkbox.
  - + “Biết hát những bài đơn giản”: checkbox.
  - + “Khác”: checkbox contains textarea.
- “Ghi chú”: textarea.
- “Nhập thông tin phụ huynh thứ 2”: checkbox.
- “Lưu”: button.
- “Hủy”: button.

All fields have data of current child.

Update information of current child to database.  
Return to child detail page

4

User edits information of child.

User click “Lưu” button.  
[Alternative 2]

#### **Alternative Scenario:**

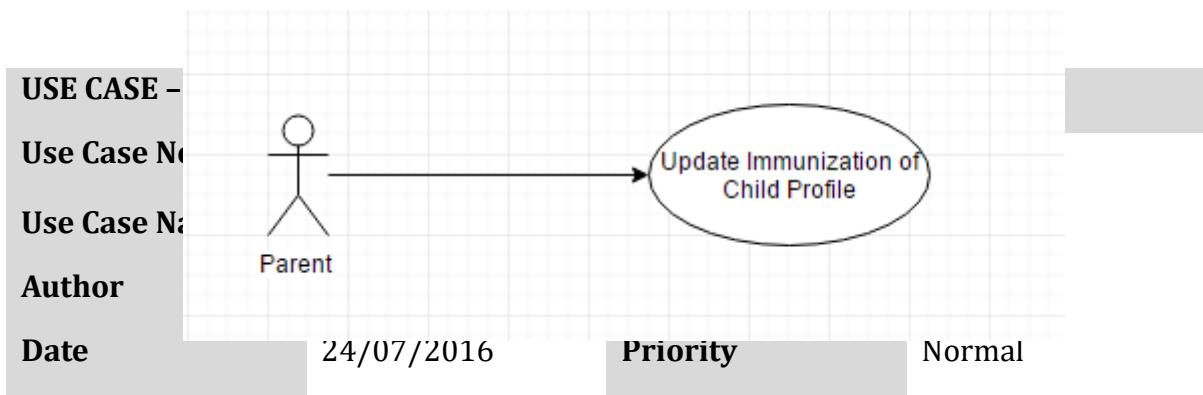
No	Actor Action	System Response
1	User clicks “Trở về danh sách trẻ” link.	Return to “Thông tin trẻ” page.
2	User clicks “Hủy” button.	Return to child detail page.

#### **Relationships:** Update Child Profile

#### **Business Rules:**

- Child profile choose to update information must be in active state.

#### **2.2.3.4. <Parent> Update Immunization of Child Profile**



**Actor:**

- Parent.

**Summary:**

- This use case allows parent to update immunization of their child's profile.

**Goal:**

- Parent can update immunization for their child.

**Triggers:**

- User clicks “Cập nhật” button in tab “Thông tin tiêm chủng”.

**Preconditions:**

- User must log in the system with parent role.

**Post Conditions:**

- **Success:** Information of immunization will be updated to database.
- **Fail:**

- + Nothing update to database.
- + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	User click on tab “Thông tin tiêm chủng”.	<p>Show table child's immunization detail page which contains:</p> <ul style="list-style-type: none"> <li>- “Thời gian dự kiến”: textbox, date time, disabled, intend time of immunization.</li> <li>- “Giai đoạn”: label, stage should perform injections.</li> <li>- “Loại vắc xin phòng bệnh”: label,</li> </ul>

- name of vaccine.
  - “Mũi tiêm”: label, current injection of immunization schedule.
  - “Ghi chú”: textarea, disabled, description of current immunization schedule.
  - “Trạng thái”: dropdown list, disabled, contains 2 stage of immunization schedule (“Đã tiêm”, “Chưa tiêm”), default “Chưa tiêm”.
  - “Thời gian tiêm”: textbox, datetime, injected real time, invisible.
  - “Nơi tiêm”: textarea, location to inject.
  - “Cập nhật”: button.
- 2 User clicks “Cập nhật” button.
- Enable “Ghi chú” textarea and “Trạng thái” dropdown list.
- 3 User change stage of “Trạng thái” from “Chưa tiêm” to “Đã tiêm”.
- Change color of current immunization schedule from red to grey, Visible “Thời gian tiêm” textbox, enable “Nơi tiêm” textarea, replace “Cập nhật” button by “Lưu” button and “Hủy” button.
- 4 User update information of immunization schedule.  
User licks “Lưu” button.  
[Alternative 1]
- Update information of immunization schedule to database.
- Show success alert “Lưu thành công”.

### **Alternative Scenario:**

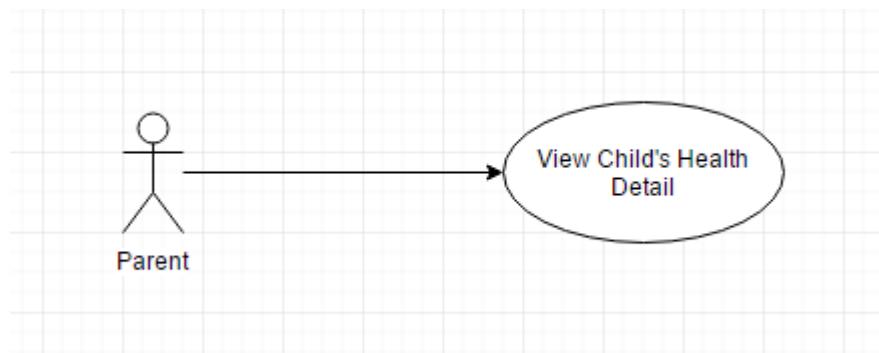
No	Actor Action	System Response
1	User clicks "Hủy" button.	Nothing update to database. Disable "Ghi chú" textarea, "Trạng thái" dropdown list, invisible "Thời gian tiêm" textbox, disable "Nơi tiêm" textarea, replace "Lưu" button and "Hủy" button by "Cập nhật" button.

**Relationships:** Update Immunization of Child Profile

**Business Rules:**

- Immunization choose to update information must be in active state.

#### 2.2.3.5. <Parent> View Child's Health Detail



#### USE CASE – UC 37

<b>Use Case No.</b>	UC37	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	View Child's Health Detail.		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

**Actor:**

- Parent.

**Summary:**

- This use case allows parent to view health detail of their child's profile.

**Goal:**

- Parent can view health's detail.

**Triggers:**

- User clicks on tab "Theo dõi tình hình".

**Preconditions:**

- User must log in the system with parent role.

**Post Conditions:**

- **Success:** Load information from database and show to parent.
- **Fail:**
  - + Nothing loaded.
  - + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	User click on tab "Theo dõi tình hình".	Show child's health detail form which contains: <ul style="list-style-type: none"><li>- "Thời gian từ": contains 2 dropdown list as month and year.</li><li>- "đến": contains 2 dropdown list as month and year.</li><li>- "Lọc": button.</li><li>- "Tình hình phát triển chiều cao của bé (Đơn vị: cm)": chart created by health check data of child from database.</li><li>- "Tình hình phát triển cân nặng của bé (Đơn vị: kg)": chart created by health check data of child from database.</li><li>- Table show detail of child's health, contains:<ul style="list-style-type: none"><li>+ "Độ tuổi": age of child.</li><li>+ "Chiều cao": height of child.</li><li>+ "Cân nặng": weight of child.</li><li>+ "Tình trạng sức khỏe": health's condition of child.</li><li>+ "Đánh giá học tập": rating of</li></ul></li></ul>

child.

All fields have data of health's check from database.

- 2 User choose data for “Thời gian từ” and “đến” dropdown list.

User clicks “Lọc” button.

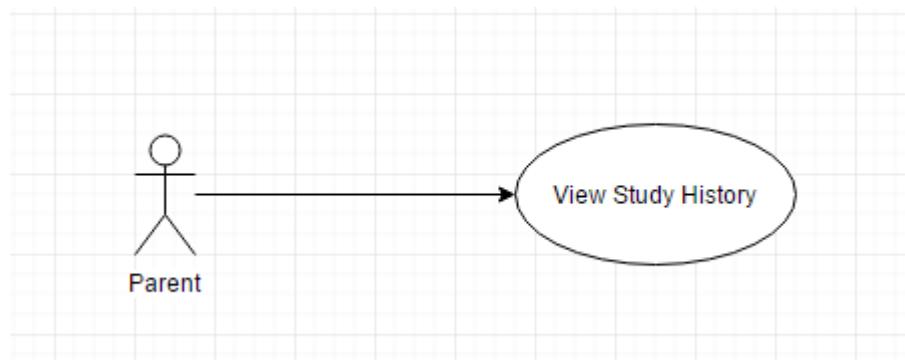
Load data from database and show for parent.

### **Relationships:** View Child's Health Detail

#### **Business Rules:**

- Information of child's health must be load from health check happen in parent's chosen time.

#### **2.2.3.6. <Parent> View Study History**



#### **USE CASE – UC 38**

<b>Use Case No.</b>	UC38	<b>Use Case Version</b>	1.0
<b>Use Case Name</b>	View Study History.		
<b>Author</b>	Đoàn Hiếu Nghĩa		
<b>Date</b>	24/07/2016	<b>Priority</b>	Normal

**Actor:**

- Parent.

**Summary:**

- This use case allows parent to view study history of their child.

**Goal:**

- Parent can view study history.

**Triggers:**

- User clicks on tab "Lịch sử học tập".

**Preconditions:**

- User must log in the system with parent role.

**Post Conditions:**

- **Success:** Load information from database and show to parent.

- **Fail:**

- + Nothing loaded.
- + System show error message.

**Main Success Scenario:**

Step	Actor Action	System Response
1	User click on tab "Lịch sử học tập".	Show child's study history form which contains: <ul style="list-style-type: none"><li>- "Năm học từ": dropdown list.</li><li>- "đến": dropdown list.</li><li>- Button Collapse/Expend for each class current child was studied.</li></ul>
2	User choose data for "Năm học từ" and "đến" dropdown list.	Load data from database and show for parent.

**Relationships:** View Study History.

## **Business Rules:**

- Class load from database must be had start date and end date between parent's chosen time.

## **3. Software system Attribute**

### **3.1. Usability**

#### **3.1.1. Graphic User Interface**

- All texts, labels and should be written in Vietnamese.

#### **3.1.2. Usability**

- Operation admin, teacher should need less than one week of training to use system.
- Parent can use the system easily without training.
- The rule and warning must be clear. User can read and do themselves without developer's help.

### **3.2. Reliability**

- All information must be clear and accurate based on the document's WHO.

### **3.3. Availability**

- Users must have internet connection to use application.

### **3.4. Security**

- All input and data should be validated before saving to database.
- All privacy information, such as password, should be encrypted to ensure security.
- Roles permission should be specify clearly and user should be authenticated and authorized when accessing to the system.

### **3.5. Maintainability**

- The system should be divided into modules and code should be easy to maintain and extend.

### **3.6. Portability**

- Website can be access everywhere which have internet and web browser.

### **3.7. Performance**

All requests should be response in no more than 30s.

#### 4. Conceptual Diagram

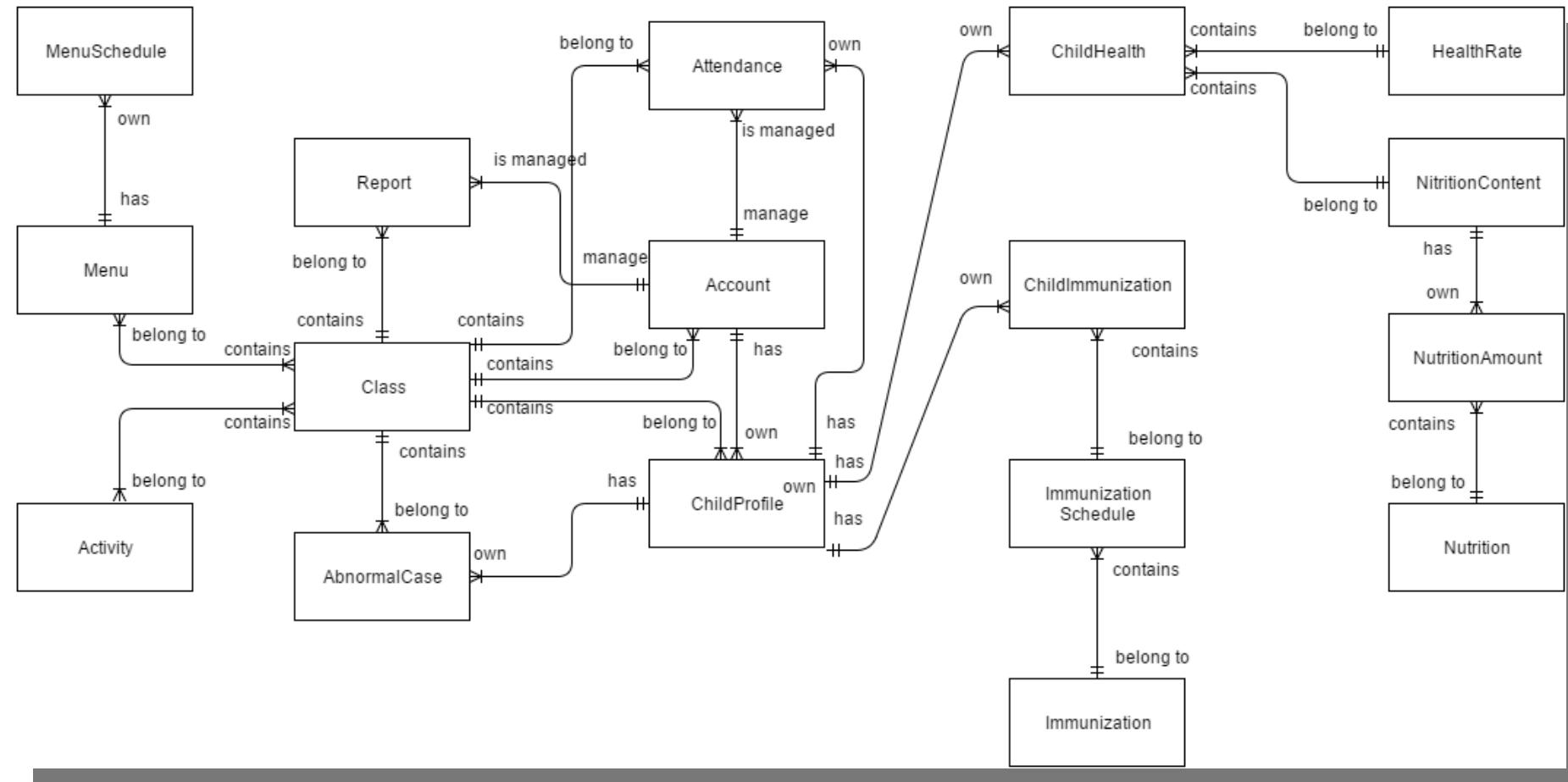


Figure 4: Conceptual Diagram

<b>Entity Data dictionary: describe content of all entities</b>	
<b>Entity Name</b>	<b>Description</b>
Account	Describe all accounts in the system. Account includes: system admin, operation admin, teacher, parent.
ChildProfile	Describe all information of child profile in the system.
ChildHealth	Describe all children's health information in the system.
HealthRate	Describe all healthrate in the system based on WHO's document.
NutritionContent	Describe all nutrition and amount suggestion in the system.
NutritionAmount	Describe all amount of nutrition in the system
Nutrition	Describe all nutrition in the system
ChildImmunization	Describe all children's immunization information in the system.
ImmunizationSchedule	Describe all immunization schedules in the system based on child's birthday and immunization.
Immunization	Describe all basic immunization in the system based on WHO's document.
Attendance	Describe all attendance of children in the system.
Report	Describe all report in the system.
Class	Describe all class in the system.
Menu	Describe all menu of in the system
MenuSchedule	Describe all menu schedule of menu in the system
Activity	Describe all activity in the system.

## D. Software Design Description

### 1. Design overview

- This document describes the technical and user interface design of CCSS. It includes the architectural design, the detailed design of common functions and business functions.
- 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:
  - o **Section 2:** The overall description of the system architectural design.
  - o **Section 3:** Component diagrams that describe the connection and integration of the system.
  - o **Section 4:** Detail design description, which includes class diagram, class explanation, and sequence diagram to describe the application functions.
  - o **Section 5:** Database design with entity relationship diagram.
  - o **Section 6:** The algorithms are used.

## 2. System Architectural Design

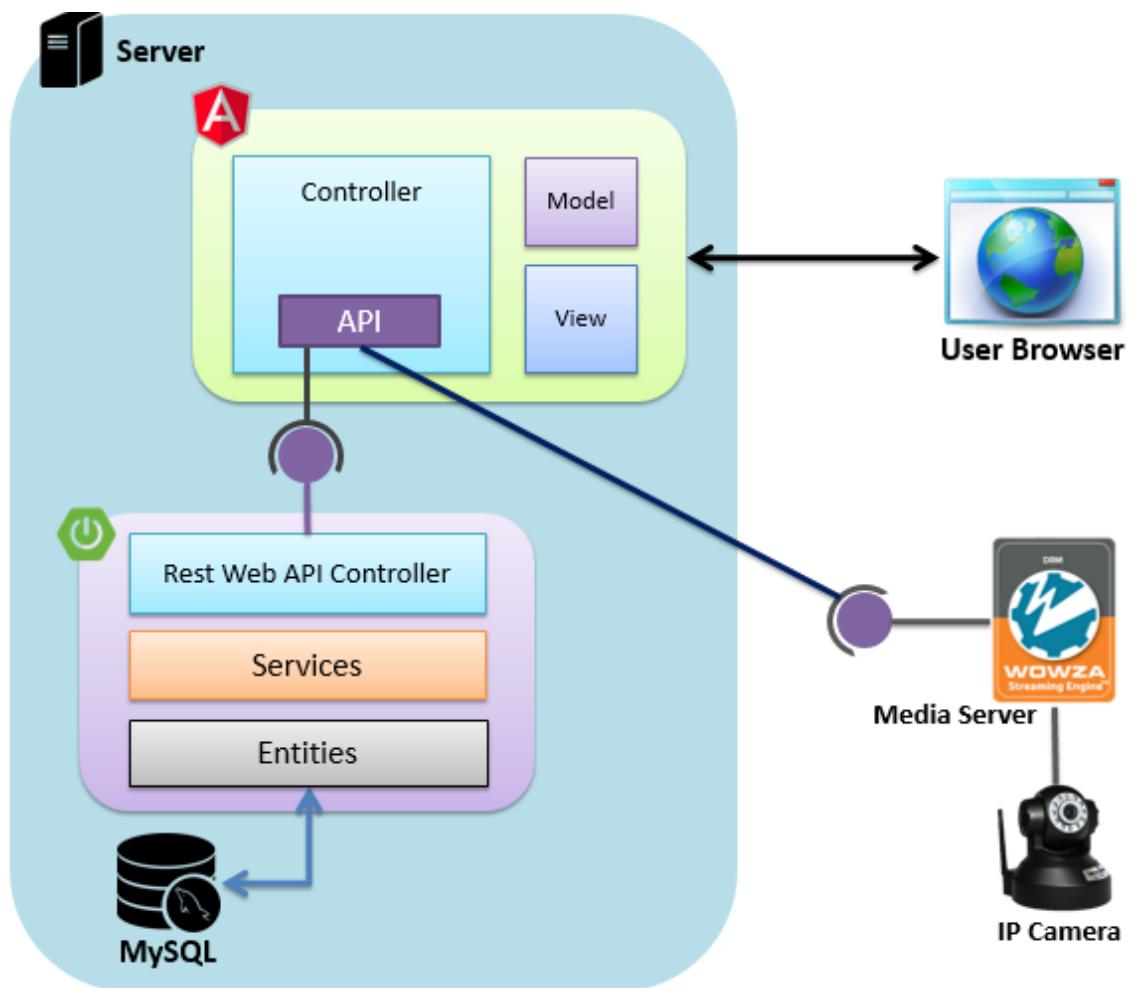
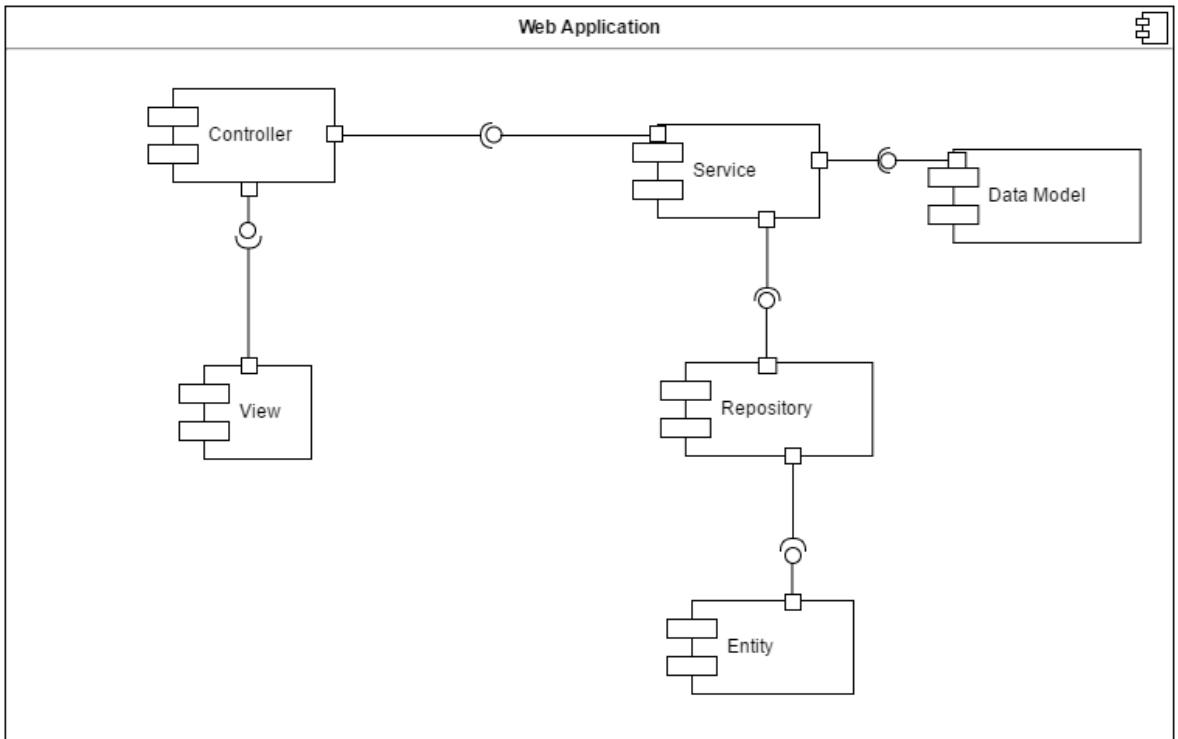


Figure 5: System Architecture

## 3. Component Diagram



**Figure 6: Component diagram**

<b>Component dictionary: describe content of all entities</b>	
<b>Component Name</b>	<b>Description</b>
Entity	Describe all entity of system.
Data Model	Describe all model of system.
Repository	Describe all repository which query data from database of system.
Service	Describe all service which implement all methods of system.
Controller	Describe all controller which implement service and provide api of system.
View	Describe all view of system.

**4.**

## 5. Detailed Description of Components

### 5.1. Class Diagram

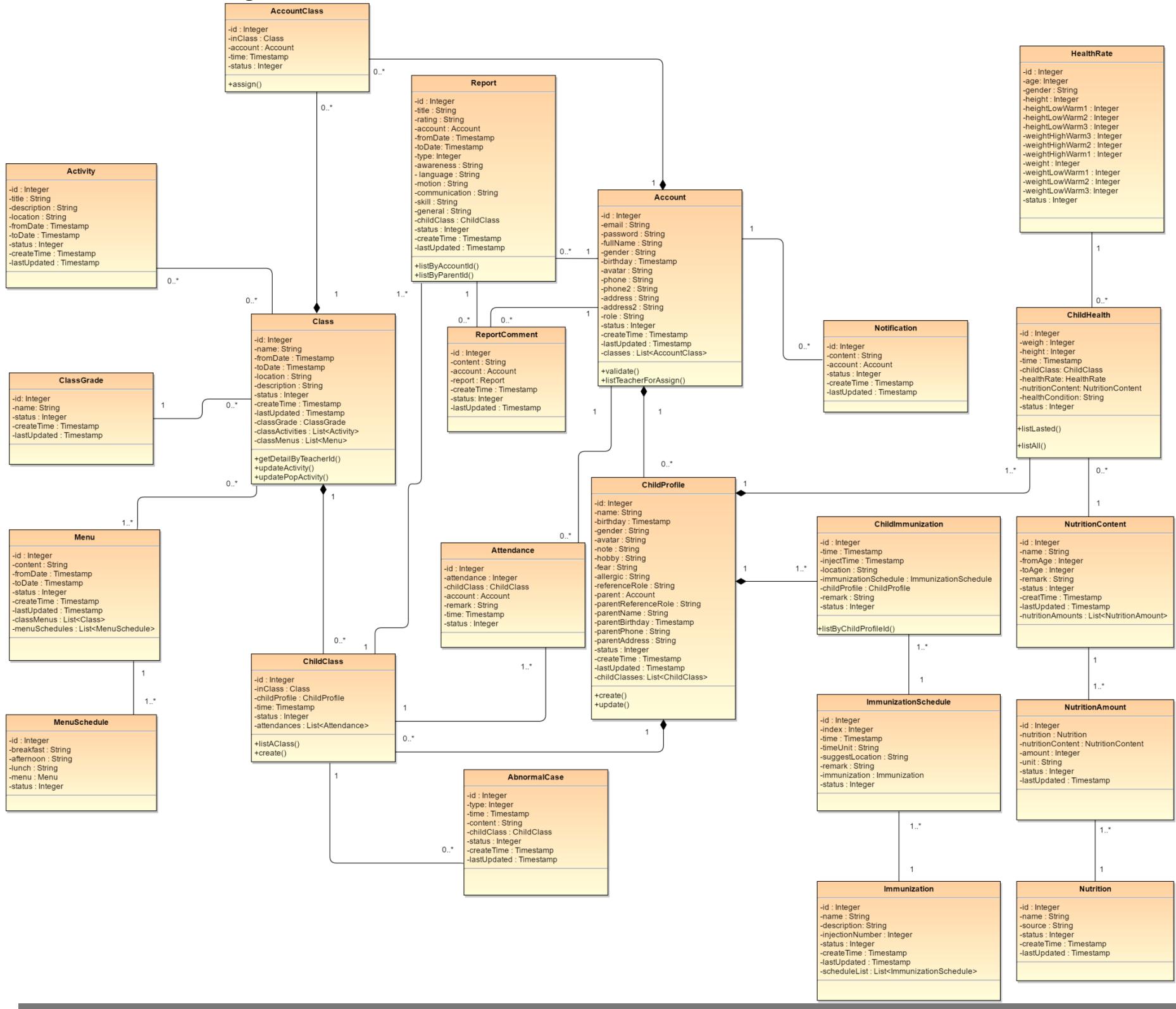


Figure 7: Class diagram

## 5.2. Class Diagram Description

### 5.2.1. Account

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Account
email	String	Public	User's email
password	String	Public	User's password
fullName	String	Public	Full name of user
gender	String	Public	Gender of user
birthday	Timestamp	Public	Birthday of user
avatar	String	Public	Avatar of user
phone	String	Public	Phone number of user
phone2	String	Public	Another phone number of user
address	String	Public	Address of user
address2	String	Public	Another address of user
role	String	Public	Role of user in the system
status	Integer	Public	Status of account
createTime	Timestamp	Public	Create time of account
lastUpdated	Timestamp	Public	Account's last time updated.

#### Method

Method	Return type	Visibility	Description
--------	-------------	------------	-------------

validate	boolean	Public	Validate account
listTeacherForAssign	List<AccountModel>	Public	List all teacher are free for assign to class

### 5.2.2. ChildProfile

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each ChildProfile
name	String	Public	Name of child
birthday	Timestamp	Public	Birthday of child
gender	String	Public	Gender of child
avatar	String	Public	Avatar of child
note	String	Public	Description of child
hobby	String	Public	Hobby of child
fear	String	Public	Fear of child
allergic	String	Public	Allergic of child
referenceRole	String	Public	Reference role between child and parent
parent	Account	Public	Parent's account of child
parentName	String	Public	Name of parent 2
parentReferenceRole	String	Public	Reference roll between child and parent 2
parentBirthday	Timestamp	Public	Birthday of parent 2

parentPhone	String	Public	Phone of parent 2
parentAddress	String	Public	Address of parent 2
status	Integer	Public	Status of child profile
createTime	Timestamp	Public	Create time of child profile
lastUpdated	Timestamp	Public	Child profile's last time updated.
childClasses	List<ChildClass>	Public	All class child has studied

## Method

Method	Return type	Visibility	Description
create	ChildProfileModel	Public	Create child profile
update	ChildProfileModel	Public	Update child profile

### 5.2.3. ChildImmunization

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each ChildImmunization
time	Timestamp	Public	Expected time of injection
injectTime	Timestamp	Public	The actual injection time
location	String	Public	Location of child's immunization
immunizationSchedule	ImmunizationSchedule	Public	Immunization schedule of child's

			immunization
childProfile	ChildProfile	Public	Child of immunization
remark	String	Public	Remark of child's immunization
status	Integer	Public	Status of child profile

## Method

Method	Return type	Visibility	Description
listByChildProfileId	List<ChildImmunization AndChildProfile>	Public	List all immunization of a child

### 5.2.4. ImmunizationSchedule

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Immunization Schedule
index	Integer	Public	Index of immunization
time	Timestamp	Public	Time need to inject.
timeUnit	String	Public	Unit of time.
suggestLocation	String	Public	Suggested location to inject
remark	String	Public	Remark of immunization schedule
status	Integer	Public	Status of

			immunization schedule
--	--	--	-----------------------

### 5.2.5. Immunization

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Immunization
name	Integer	Public	Name of immunization
description	String	Public	Description of immunization
injectionNumber	Integer	Public	Injection's number of immnization
createTime	Timestamp	Public	Create time of immunization
lastUpdated	Timestamp	Public	Immunization's last time updated
status	Integer	Public	Status of immunization schedule
scheduleList	List<Immunization Schedule>	Public	List schedule of immunization

### 5.2.6. ChildHealth

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each ChildHealth
weight	Integer	Public	Weight of child

height	Integer	Public	Height of child
time	String	Public	Time of child's health check
childClass	ChildClass	Public	Class of child
healthRate	HealthRate	Public	HealthRate for child corresponding to child's health
healthCondition	String	Public	Child's health condition at checking
nutritionContent	NutritionContent	Public	Nutrition content for child corresponding to child's health
status	Integer	Public	Status of child profile

## Method

Method	Return type	Visibility	Description
listLasted	List<ChildHealthModel>	Public	List lasted child's health check of all child in a class
listAll	List<ChildHealthModel>	Public	List all child's health check of a child

### 5.2.7. HealthRate

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each HealthRate
age	Integer	Public	Age of child compatible with

			health rate
gender	String	Public	Gender of child compatible with health rate
height	Integer	Public	Standard height
heightLowWarm1	Integer	Public	Height substandard level 1
heightLowWarm2	Integer	Public	Height substandard level 2
heightLowWarm3	Integer	Public	Height substandard level 3
weightHighWarm3	Integer	Public	Exceeds the standard weight level 3
weightHighWarm2	Integer	Public	Exceeds the standard weight level 2
weightHighWarm1	Integer	Public	Exceeds the standard weight level 1
weight	Integer	Public	Standard weight
weightLowWarm1	Integer	Public	Weight substandard level 1
weightLowWarm2	Integer	Public	Weight substandard level 2
weightLowWarm3	Integer	Public	Weight substandard level 3
status	Integer	Public	Status of health rate

## 5.2.8. NutritionContent

### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each

			NutritionContent
name	String	Public	Name of nutrition content
fromAge	Integer	Public	The lowest age of child compatible with nutrition content
toAge	Integer	Public	The highest age of child compatible with nutrition content
remark	String	Public	Remark of nutrition content
status	Integer	Public	Status of health rate
createTime	Timestamp	Public	Create time of nutrition content
lastUpdated	Timestamp	Public	Nutrition content's last time updated
nutritionAmounts	List<NutritionAmount>	Public	List nutrition amount of nutrition content

## 5.2.9. NutritionAmount

**Attribute**

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each NutritionAmount
nutrition	Nutrition	Public	Nutrition of nutrition amount

nutritionContent	NutritionContent	Public	Nutrition Content of nutrition amount
amount	Integer	Public	Amount of nutrition
unit	String	Public	Unit of amount
status	Integer	Public	Status of health rate
lastUpdated	Timestamp	Public	Nutrition content's last time updated

### 5.2.10. Nutrition Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Nutrition
name	String	Public	Name of nutrition
source	String	Public	Source of nutrition
status	Integer	Public	Status of health rate
createTime	Timestamp	Public	Create time of nutrition
lastUpdated	Timestamp	Public	Nutrition content's last time updated

### 5.2.11. Class Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier

			of each Class
name	String	Public	Name of class
fromDate	Timestamp	Public	Start date of class
toDate	Timestamp	Public	End date of class
location	String	Public	Location of class
description	String	Public	Description of class
createTime	Timestamp	Public	Create time of nutrition
lastUpdated	Timestamp	Public	Nutrition content's last time updated
classGrade	ClassGrade	Public	Grade of class
classActivities	List<Activity>	Public	Activities of class
classMenus	List<Menu>	Public	Menus of class

## Method

Method	Return type	Visibility	Description
getDetailByTeacherId	ClassDetailModel	Public	Get detail of class by teacher
updateActivity	ClassModel	Public	Add activity into class's activity list
updatePopActivity	ClassModel	Public	Remove activity from class's activity list

### 5.2.12. AccountClass Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier

			of each AccountClass
inClass	Class	Public	Class
account	Account	Public	Teacher in class
time	Timestamp	Public	Create time of AccountClass
status	Integer	Public	Status of AccountClass

## Method

Method	Return type	Visibility	Description
assign	List<AccountClassModel>	Public	Assign teacher to class

### 5.2.13. ChildClass

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each ChildClass
inClass	Class	Public	Class
childProfile	ChildProfile	Public	Child in class
time	Timestamp	Public	Create time of ChildClass
status	Integer	Public	Status of ChildClass
attendances	List<Attendance>	Public	

## Method

Method	Return type	Visibility	Description
--------	-------------	------------	-------------

listAClass	List<ChildClassModel>	Public	List all child class in a class.
create	List<ChildClassModel>	Public	Create list ChildClass of a class

### 5.2.14. Attendance

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Attendance
attendance	Integer	Public	Attendance of child
childClass	ChildClass	Public	Child has attendance
account	Account	Public	Teacher checks attendance
time	Timestamp	Public	Create time of attendance
status	Integer	Public	Status of attendance

### 5.2.15. Report

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Report
title	String	Public	Title of report
rating	String	Public	Rating of child
account	Account	Public	Teacher creates

			report
fromDate	Timestamp	Public	Start date of report
toDate	Timestamp	Public	End date of report
type	Integer	Public	Type of report
awareness	String	Public	Comment about awareness of child
language	String	Public	Comment about language of child
motion	String	Public	Comment about motion of child
communication	String	Public	Comment about communication of child
skill	String	Public	Comment about skill of child
general	String	Public	General comment of child
childClass	ChildClass	Public	Child and class of report
createTime	Timestamp	Public	Create time of report
lastUpdated	Timestamp	Public	Report's last time updated
status	Integer	Public	Status of attendance

## Method

Method	Return type	Visibility	Description
listByAccountId	List< ReportModel >	Public	List report created by a teacher

listByParentId	List< ReportModel >	Public	List child's report for a parent
----------------	---------------------	--------	----------------------------------

### 5.2.16. Report comment

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Report comment
content	String	Public	Content of comment
report	Report	Public	Report of comment
account	Account	Public	User post comment
childClass	ChildClass	Public	Child and class of report
createTime	Timestamp	Public	Create time of report
lastUpdated	Timestamp	Public	Report's last time updated
status	Integer	Public	Status of attendance

### 5.2.17. ClassGrade

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Class grade
name	String	Public	Name of class's

			grade
createTime	Timestamp	Public	Create time of class's grade
lastUpdated	Timestamp	Public	Class's grade last time updated
status	Integer	Public	Status of class's grade

### 5.2.18. Activity

#### Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Activity
title	String	Public	Title of activity
description	String	Public	Description of activity
location	String	Public	Location of activity
fromDate	String	Public	Start date of activity
toDate	String	Public	End date of activity
createTime	Timestamp	Public	Creat time of activity
lastUpdated	Timestamp	Public	Activity's last time updated
status	Integer	Public	Status of activity

### 5.2.19. Menu

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier

			of each Menu
content	String	Public	Content of menu
fromDate	String	Public	Start date of menu
toDate	String	Public	End date of menu
createTime	Timestamp	Public	Creat time of menu
lastUpdated	Timestamp	Public	Menu's last time updated
status	Integer	Public	Status of menu
classMenus	List<Class>	Public	List class use menu
menuSchedules	List<MenuSchedule>	Public	List schedule of menu

### 5.2.20. **MenuSchedule** Attribute

Attribute	Type	Visibility	Description
id	Integer	Public	Unique identifier of each Menu Schedule
breakfast	String	Public	Dishes for breakfast
afternoon	String	Public	Dishes for afternoon
lunch	String	Public	Dishes for lunch
menu	Menu	Public	Menu contains schedule
status	Integer	Public	Status of menu

### **5.2.21. Post**

#### **Attribute**

<b>Attribute</b>	<b>Type</b>	<b>Visibility</b>	<b>Description</b>
id	Integer	Public	Unique identifier of each Post
category	String	Public	Category of post
content	String	Public	Content of post
account	Account	Public	User creat post
status	Integer	Public	Status of menu
createTime	Timestamp	Public	Create time of post
lastUpdated	Timestamp	Public	Post's last time updated

### **5.2.22. Notification**

#### **Attribute**

<b>Attribute</b>	<b>Type</b>	<b>Visibility</b>	<b>Description</b>
id	Integer	Public	Unique identifier of each Notification
content	String	Public	Content of notification
status	Integer	Public	Status of notification
createTime	Timestamp	Public	Create time of notification
lastUpdated	Timestamp	Public	Notification's last time updated

### **5.2.23. AbnormalCase**

#### **Attribute**

<b>Attribute</b>	<b>Type</b>	<b>Visibility</b>	<b>Description</b>
id	Integer	Public	Unique identifier of each AbnormalCase
type	Integer	Public	Type of AbnormalCase
time	Timestamp	Public	Time of AbnormalCase
content	String	Public	Content of AbnormalCase
childClass	ChildClass	Public	Child and class of AbnormalCase
status	Integer	Public	Status of AbnormalCase
createTime	Timestamp	Public	Create time of AbnormalCase
lastUpdated	Timestamp	Public	AbnormalCase's last time updated

## 5.3. Sequence Diagram

### 5.3.1. <User> Login

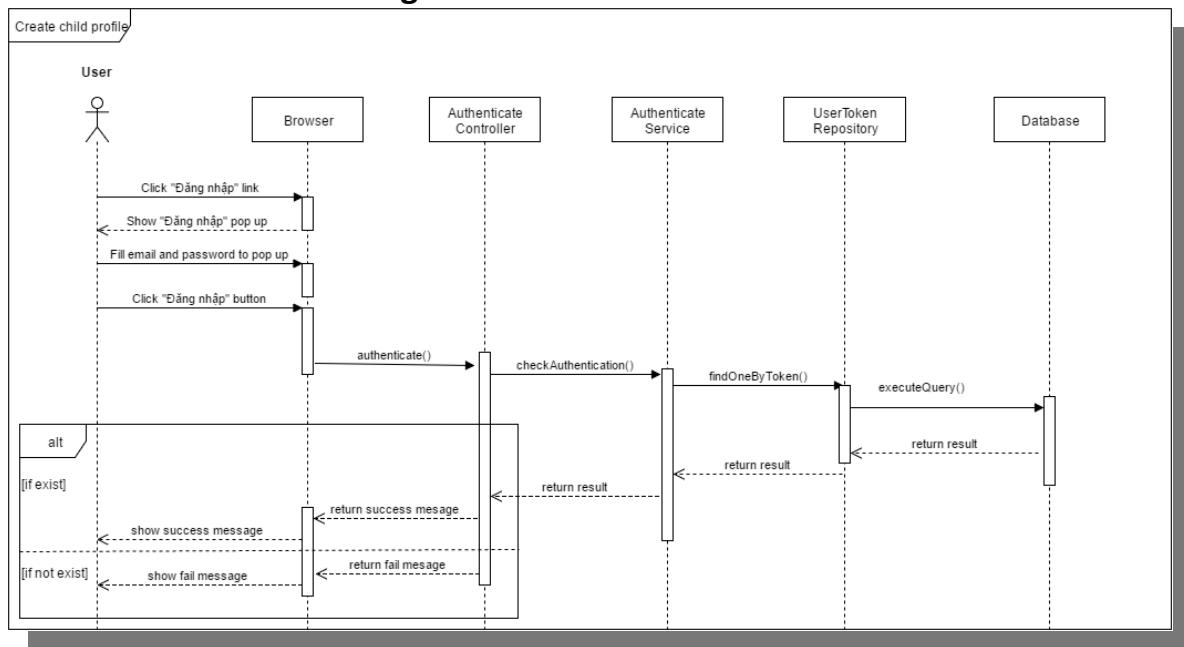


Figure 8: User login sequence diagram

### 5.3.2. <System Admin/Operation Admin> Create parent/teacher

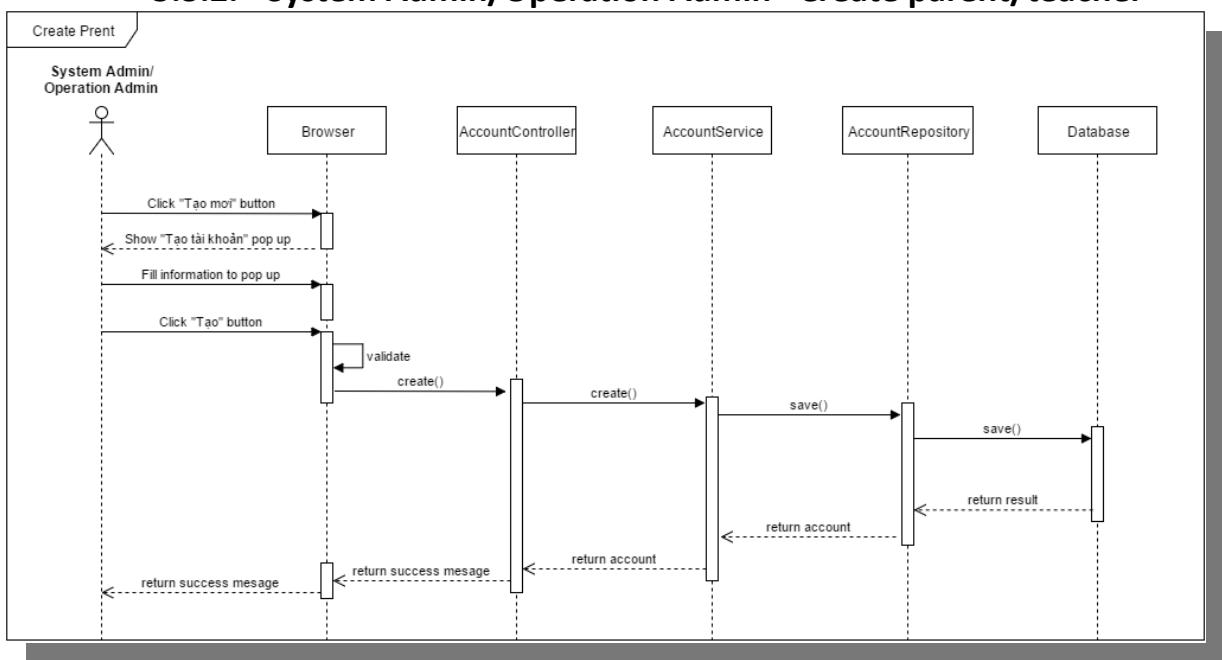


Figure 9: Create user account sequence diagram

### 5.3.3. <Parent> Create child profile

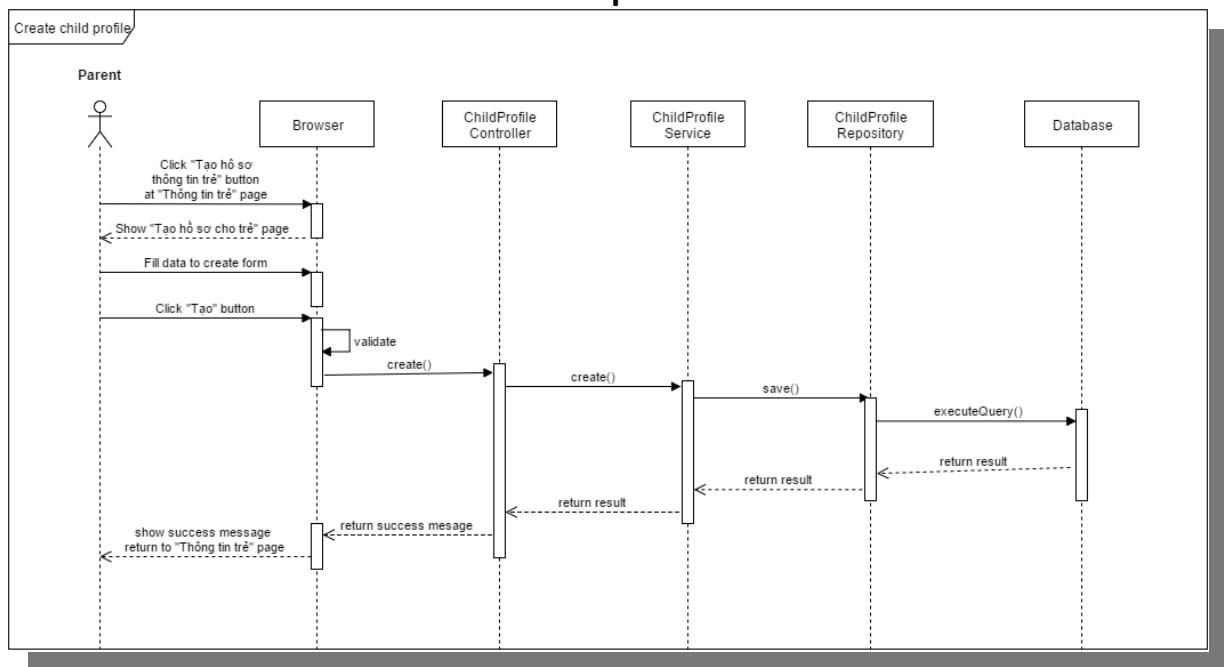


Figure 10: Create child profile sequence diagram

### 5.3.4. <Parent> Edit child profile

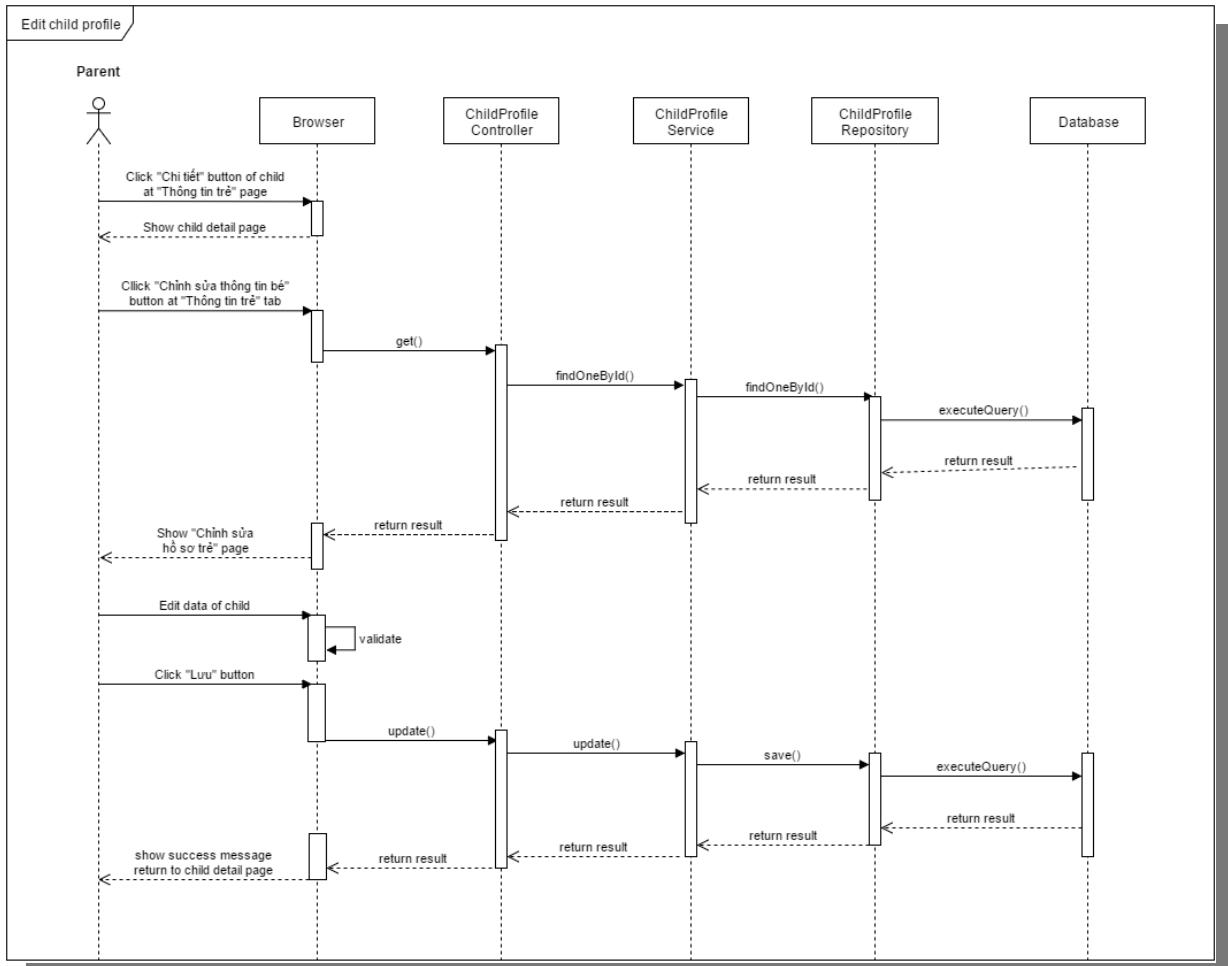
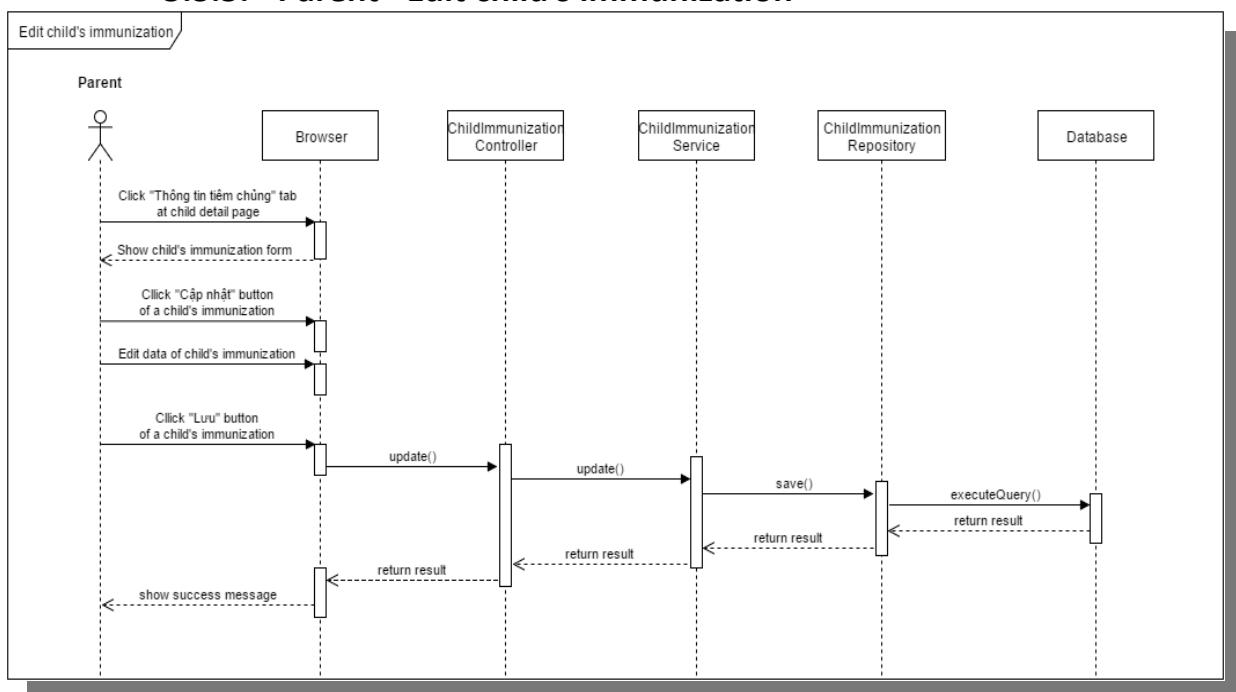


Figure 11: Edit child profile sequence diagram

### 5.3.5. <Parent> Edit child's immunization



**Figure 12: Edit child immunization sequence diagram**

### 5.3.6. <Parent> View child's health

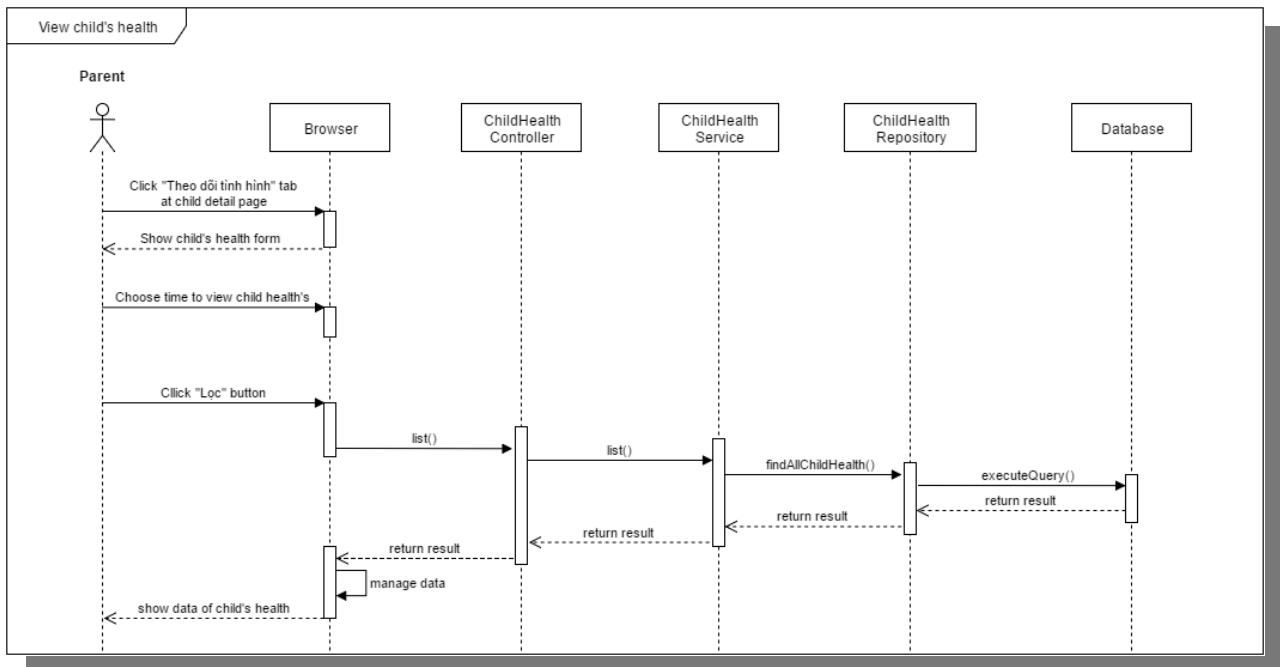


Figure 13: View child health sequence diagram

### 5.3.7. <Parent> View Study History

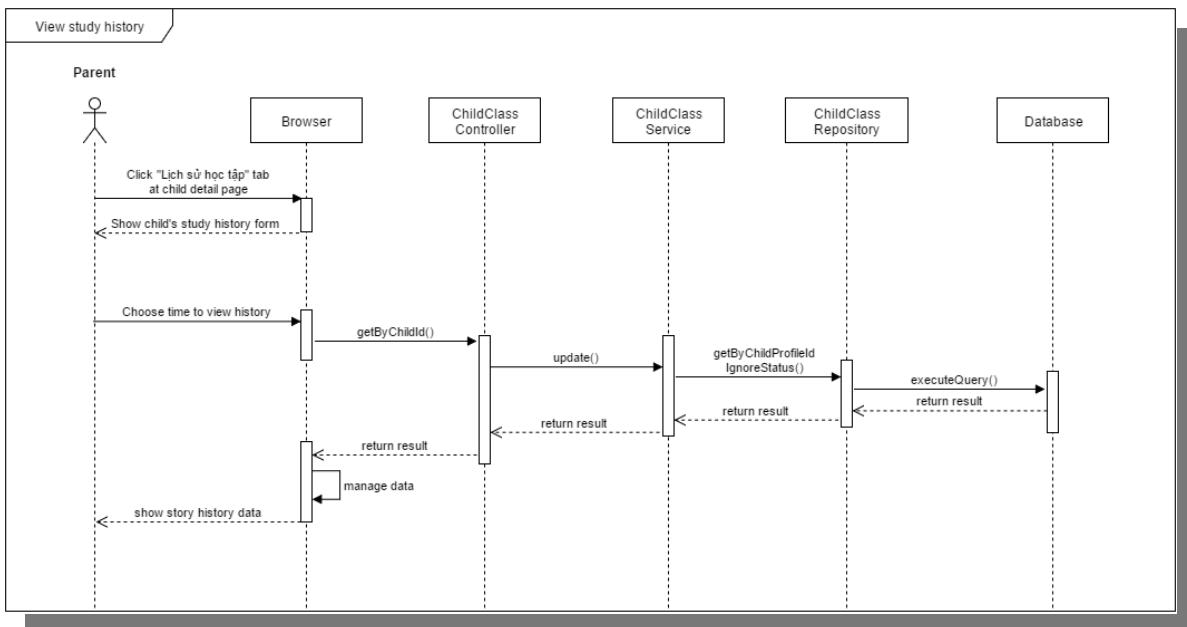


Figure 14: View study history sequence diagram

### 5.3.8. <Parent> View child's report

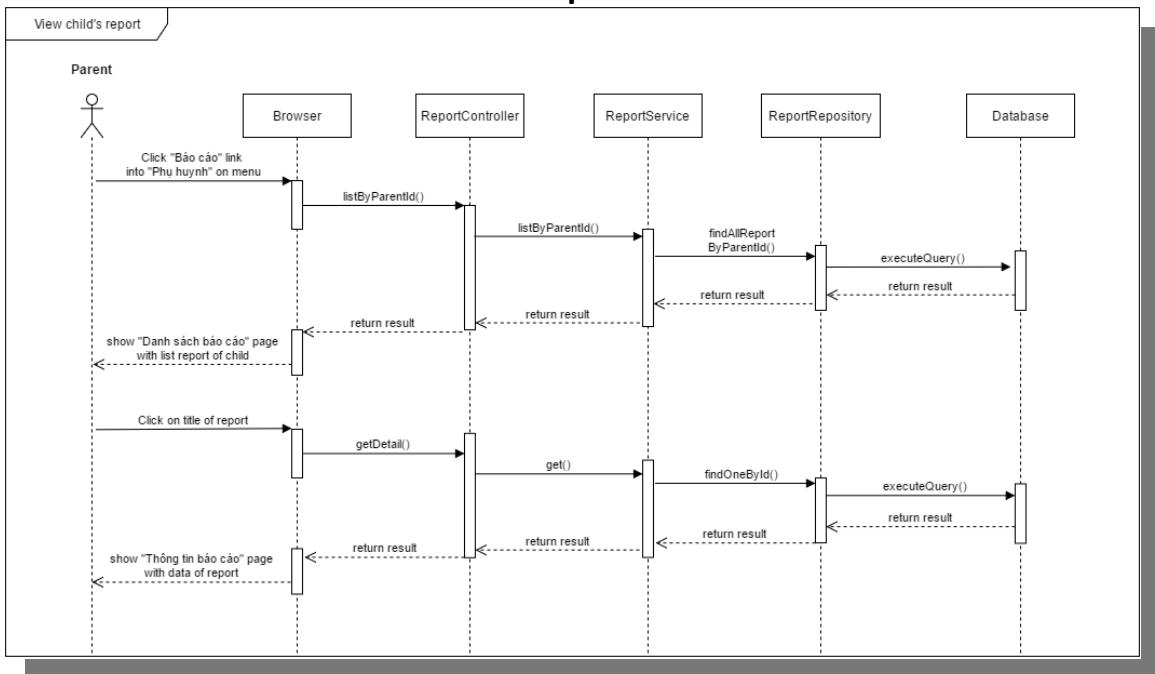
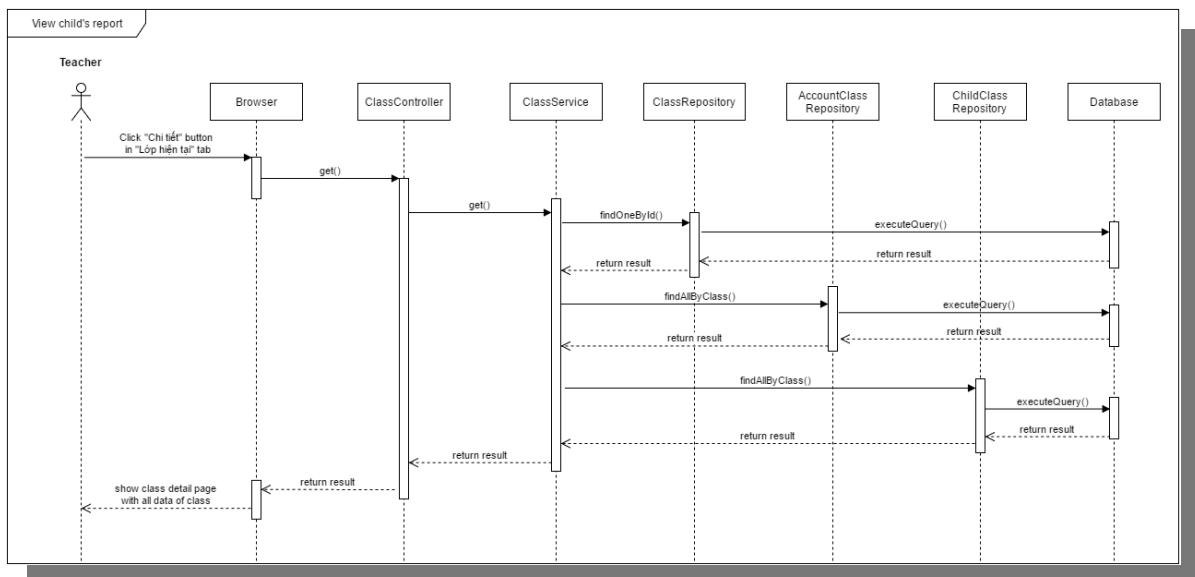


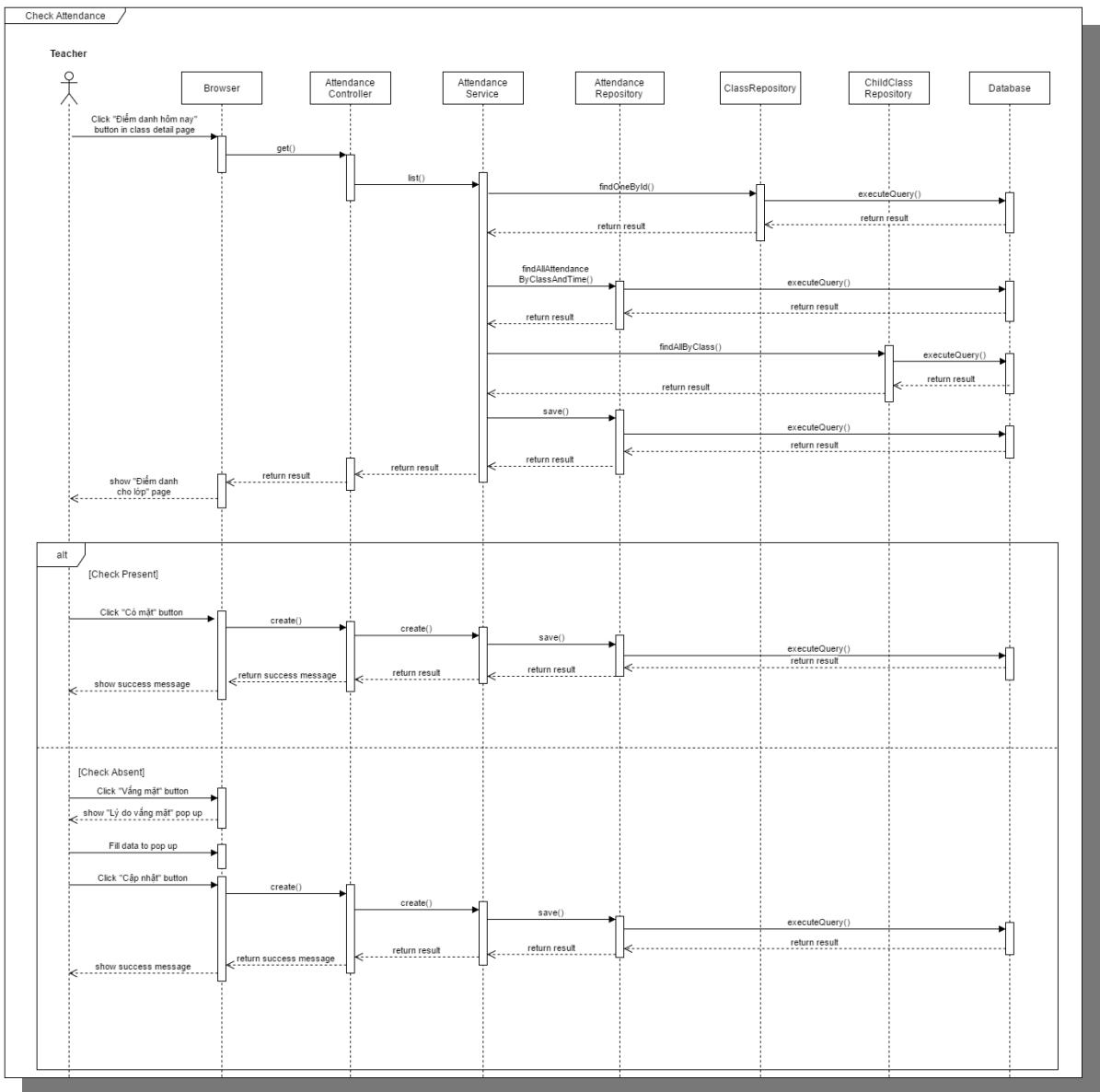
Figure 15: View child report sequence diagram

### 5.3.9. <Teacher> View class detail



**Figure 16: View class detail sequence diagram**

### **5.3.10. <Teacher> Check Attendance**



**Figure 17: Check attendance sequence diagram**

### 5.3.11. <Teacher>Health Check

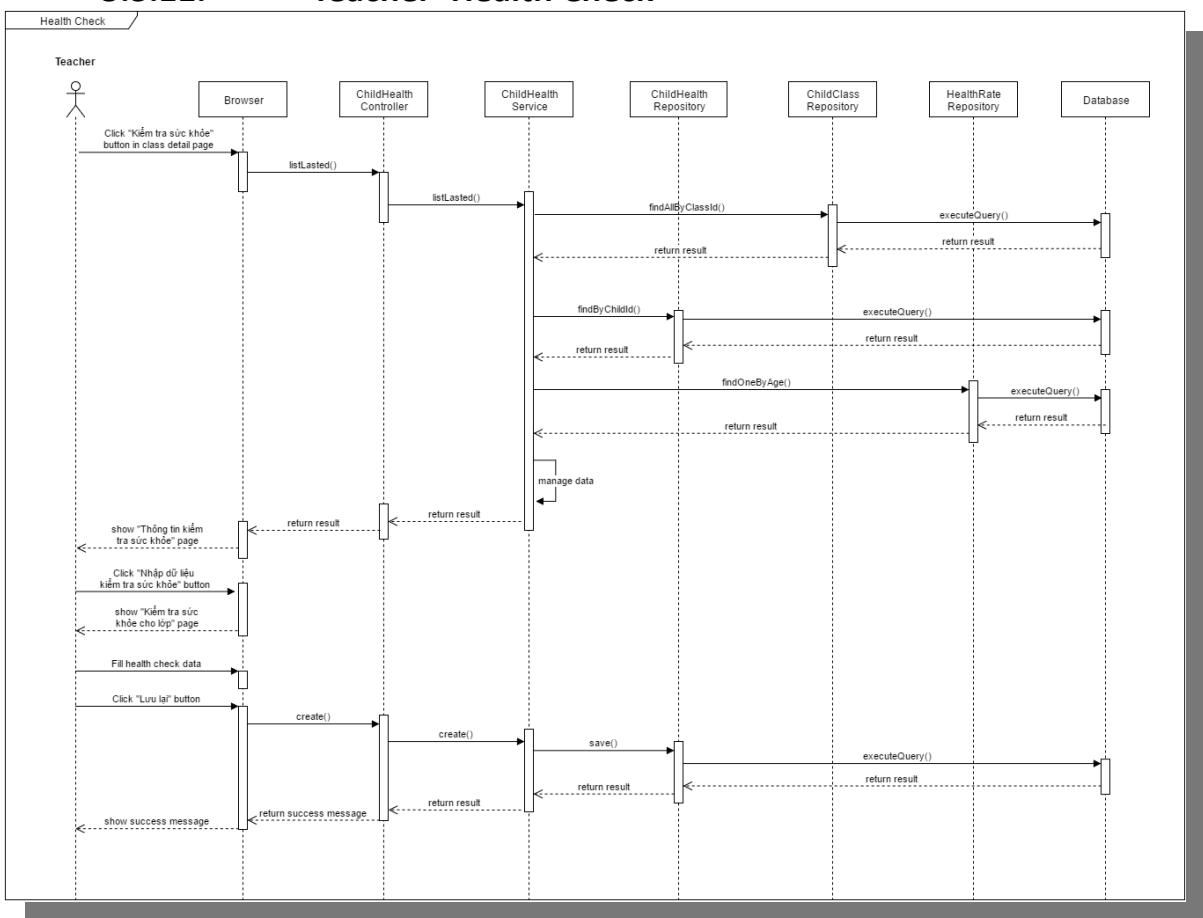


Figure 18: Health check sequence diagram

## 6. Database Designed

### 6.1. Entity Relationship Diagram

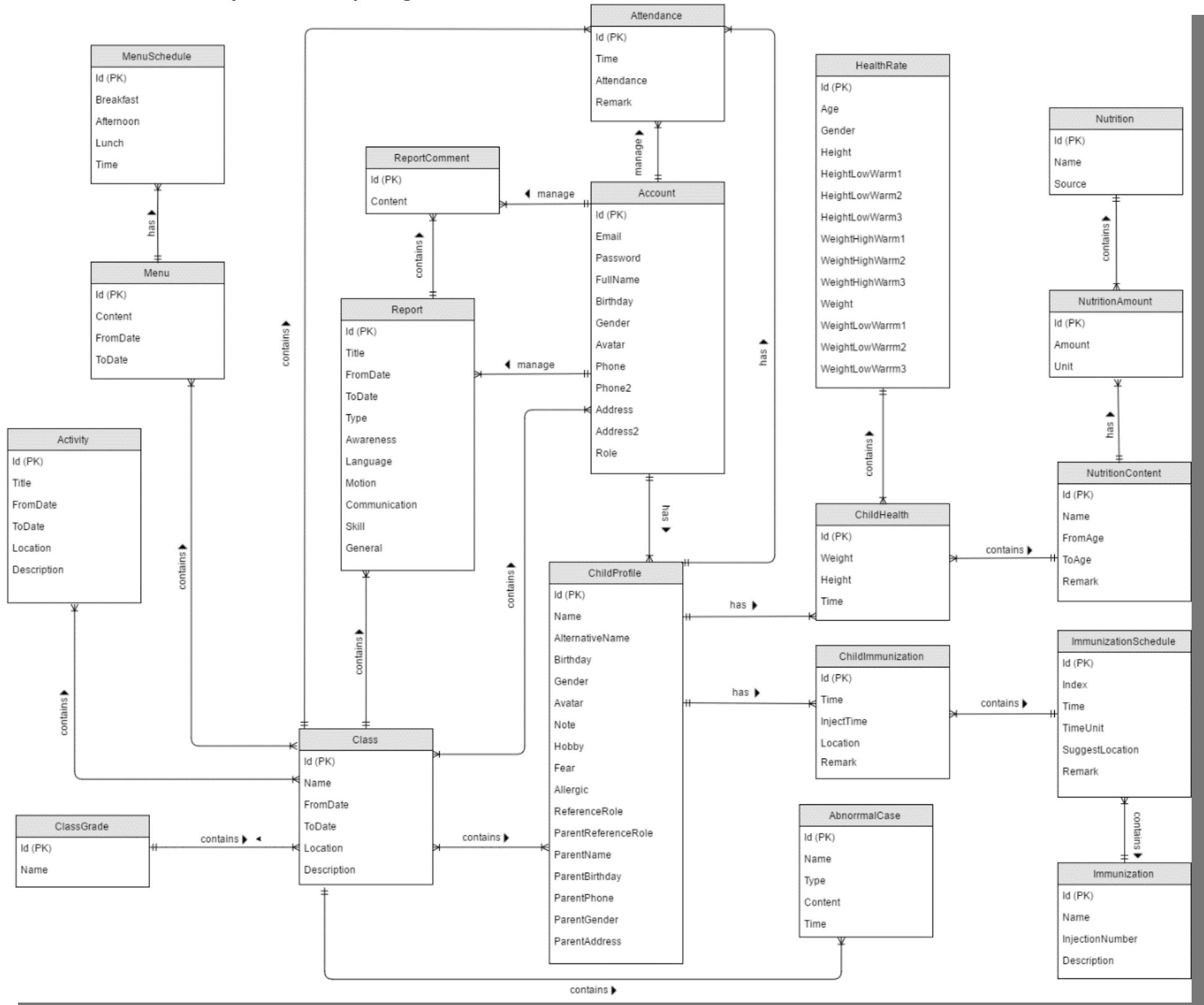


Figure 19: Entity Relationship Diagram

## 6.2. Entity Dictionary

**Entity Data dictionary: describe content of all entities**

Entity Name	Description
Account	Describe all accounts in the system. Account includes: system admin, operation admin, teacher and parent.
ChildProfile	Describe all child profile in the system.
ChildHealth	Describe all child's health in the system
HealthRate	Describe all heath rate for child's health in the system.
NutritionContent	Describe all nutrition content in the system.
NutritionAmount	Describe all nutrition amount in the system.
Nutrition	Describe all nutrition in the system
ChildImmunization	Describe all child's immunization in the system
ImmunizationSchedule	Describe all immunization's schedule in the system
Immunization	Describe all immunization in the system
Notification	Describe all notification in the system
Attendance	Describe all attendance of child in the system
Report	Describe all report of child in the system
ReportComment	Describe all report's comment in the system
Class	Describe all class in the system
ClassGrade	Describe all class's grade in the system
Menu	Describe all menu in the system
MenuSchedule	Describe all menu's schedule in the system
Activity	Describe all activity in the system
AbnormalCase	Describe all unexpected absences of child in the system

## E. System Implementation & Test

## **1. Introduction**

### **1.1. Overview**

This section provides in detail all necessary information about test plans, test cases, test result, test environments, pass/fail criteria and risks estimations as well as a checklist to cover all possible cases of this project.

### **1.2. Test Approach**

- **Goal:** To test the whole system based on the core workflow.
- **Method:** System Testing, Black-box testing.

## 2. Database Relationship

### 2.1. Database Physical Diagram

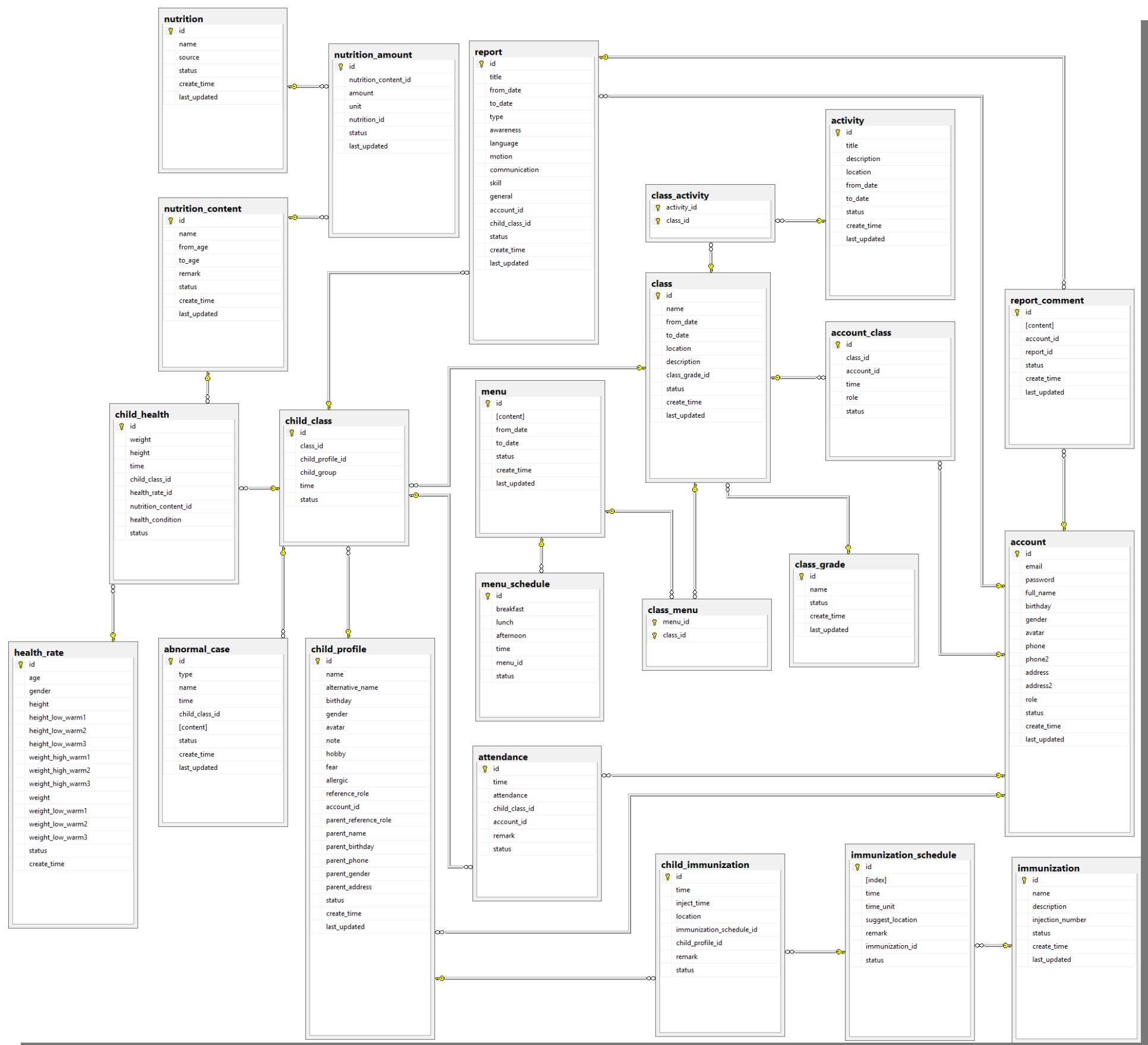


Figure 20: Database Physical Diagram

## 2.2. Data Dictionary

### 2.2.1. Table Data Dictionary

Data dictionary: describe content of all tables	
Table Name	Description
account	Describe all accounts in the system. Account includes: system admin, operation admin, teacher and parent.
child_profile	Describe all children's profile in the system.
child_health	Describe all information of child's health in the system.
health_rate	Describe all health rates for child in the system.
nutrition_content	Describe all nutrition's content for child in the system.
nutrition_amount	Describe amount of nutrition for nutrition's content in the system.
nutrition	Describe all nutrition in the system.
child_immunization	Describe all children's immunization in the system.
immunization_schedule	Describe all immunization's schedule in the system.
immunization	Describe all immunization in the system.
class	Describe all class in the system.
account_class	Describe class's teachers in the system.
child_class	Describe class's children in the system.
attendance	Describe all attendance of child in the system.
report	Describe all report of child in the system.
report_comment	Describe all report's comment in the system.
class_menu	Describe class's menus in the system
menu	Describe all menu in the system
menu_schedule	Describe all menu's schedule in the system
class_activity	Describe all class's activities in the system
activity	Describe all activity in the system

class_grade	Describe all class's grade in the system
abnormal_case	Describe all unexpected absences of child in the system

### 2.2.2. Attribute Data Dictionary

Table name	Attributes	Description	Domain	Null
account	id	Unique identify of account.	int	No
	email	Email of user	varchar(255)	No
	password	Password of user	varchar(255)	No
	full_name	Full name of user	varchar(255)	No
	birthday	Birthday of user	Timestamp	No
	gender	Gender of user	varchar(45)	No
	avatar	Avatar of user	varchar(255)	Yes
	phone	Phone number of user	varchar(45)	No
	phone2	Phone number of user	varchar(45)	Yes
	address	Address of user	varchar(255)	No
	address2	Address of user	varchar(255)	Yes
	role	Role of user	varchar(45)	No
	create_time	Create time of account	Timestamp	No
	last_updated	Last time updated of account	Timestamp	No
	status	Status of account	int	No

child_profile	<b>id</b>	Unique identify of child profile.	int	No
	<b>name</b>	Name of child	varchar(255)	No
	<b>birthday</b>	Birthday of child	Timestamp	No
	<b>gender</b>	Gender of child	varchar(45)	No
	<b>avatar</b>	Avatar of child	longtext	Yes
	<b>note</b>	Note of child	text	
	<b>reference_role</b>	Referemce role between child with parent	varchar(255)	No
	<b>account_id</b>	Id of parent's account	int	No
	<b>parent_name</b>	Name of parent 2	varchar(255)	Yes
	<b>parent_reference_role</b>	Referemce role between child with parent 2	varchar(45)	Yes
	<b>parent_birthday</b>	Birthday of parent 2	Timestamp	Yes
	<b>parent_phone</b>	Phone number of parent 2	varchar(45)	Yes
	<b>parent_gender</b>	Gender number of parent 2	varchar(255)	Yes
	<b>parent_address</b>	Address of parent 2	varchar(255)	Yes
	<b>alternative_name</b>	Alternative name of child	varchar(255)	Yes
	<b>allergic</b>	Allergic of child	varchar(255)	Yes
	<b>fear</b>	Fear of child	varchar(255)	Yes
	<b>hobby</b>	Hobby of child	varchar(255)	Yes
	<b>status</b>	Status of child	int	No

		profile		
	create_time	Create time of child profile	Timestamp	No
	last_Updated	Last time updated of child profile	Timestamp	No
health_rate	id	Unique identify of health rate.	int	No
	age	Age of child	int	No
	gender	Gender of child	varchar(45)	No
	height	Standard height	int	No
	height_low_warm1	Substandard height level 1	int	Yes
	height_low_warm2	Substandard height level 2	int	Yes
	height_low_warm3	Substandard height level 3	int	Yes
	weight_heigh_warm3	Exceeds the standard weight level 3	int	Yes
	weight_heigh_warm2	Exceeds the standard weight level 2	int	Yes
	weight	Standard weight	int	No
	weight_low_warm1	Substandard weight level 1	int	Yes
	weight_low_warm2	Substandard weight level 2	int	Yes
	weight_low_warm3	Substandard weight level 3	int	Yes
	status	Status of health	int	No

		rate		
	create_time	Create time of health rate	Timestamp	No
nutrition_content	id	Unique identify of nutrition content.	int	No
	name	Name of nutrition content	varchar(255)	No
	from_age	Lowest age of child	int	No
	to_age	Highest age of child	Int	No
	remark	Remark of nutrition content	text	Yes
	status	Status of nutrition content	int	No
	create_time	Create time of nutrition content	Timestamp	No
	last_updated	Last time updated of nutrition content	Timestamp	No
nutrition_amount	id	Unique identify of nutrition amount.	int	No
	nutrition_content_id	Id of nutrition content	int	No
	nutrition_id	Id of nutrition	int	No
	amount	Amount of nutrition	Int	No
	unit	Unit of amount	varchar(100)	No
	status	Status of nutrition	int	No

		amount		
	last_updated	Last time updated of nutrition amount	Timestamp	No
nutrition	id	Unique identify of nutrition.	int	No
	name	Name of nutrition	varchar(255)	No
	Source	Source of nutrition	text	No
	status	Status of nutrition	int	No
	create_time	Create time of nutrition	Timestamp	No
	last_updated	Last time updated of nutrition	Timestamp	No
child_immunization	id	Unique identify of child's immunization.	int	No
	time	Expected time of injection	Timestamp	No
	inject_time	The actual injection time	Timestamp	No
	location	Location to inject	text	Yes
	remark	Remark of child's immunization	text	Yes
	immunization_schedule_id	Id of immunization's schedule	int	No

	child_profile_id	Id of child profile	int	No
	status	Status of child's immunization	Int	No
immunization_schedule	id	Unique identify of immunization's schedule.	int	No
	index	Index of injection	int	No
	time	Time need to inject	Int	No
	time_unit	Create time of classify	varchar(255)	No
	suggest_location	Suggested location to inject	text	Yes
	remark	Remark of immunization's schedule	text	Yes
	immunization_id	Id of immunization	int	No
	status	Status of immunization	Int	No
immunization	id	Unique identify of immunization.	int	No
	name	Name of immunization	varchar(255)	No
	description	Description of immunization	text	Yes
	injection_number	Number of injection	int	No
	status	Status of	int	No

		immunization		
	create_time	Create time of immunization	Timestamp	No
	last_updated	Last time updated of immunization	Timestamp	No
Class	id	Unique identify of class.	int	No
	name	Name of class	varchar(255)	No
	from_date	Start date of class	Timestamp	No
	to_date	End date of class	Timestamp	No
	location	Location of class	varchar(255)	No
	description	Description of class	text	Yes
	class_grade_id	Grade of class	int	No
	status	Status of class	int	No
	create_time	Create time of class	Timestamp	No
	last_updated	Last time updated of class	Timestamp	No
account_class	id	Unique identify of accountclass.	int	No
	class_id	Id of class	int	No
	account_id	Id of teacher's account	int	No
	time	Time create accountclass	Timestamp	No
	Status	Status of dish: active, deactive	Int	Yes

child_class	id	Unique identify of childclass.	int	No
	class_id	Id of class	int	No
	child_profile_id	Id of child profile	int	No
	time	Time create accountclass	Timestamp	No
	status	Status of child class	int	No
	child_group	Child's group by health	int	Yes
Report	id	Unique identify of report.	int	No
	title	Title of report	text	No
	from_date	Start date of report	Timestamp	No
	to_date	End date of report	Timestamp	No
	type	Type of report	int	No
	child_group	Child's group by health	int	No
	awareness	Chils's awareness comment	varchar(255)	No
	communication	Chidl's communication comment	varchar(255)	No
	language	Chidl's language comment comment	varchar(255)	No
	motion	Child's motion comment comment	varchar(255)	No

	rating	Child's rating	Varchar(255)	No
	general	Child's general comment	Varchar(255)	No
	status	Status of report	int	No
	create_time	Create time of report	Timestamp	No
	last_updated	Last time updated of report	Timestamp	No
report_comment	id	Unique identify of report's comment.	int	No
	content	Content of comment	text	No
	account_id	Id of account post comment	int	No
	report_id	Id of report contain comment	int	No
	status	Status of comment	int	No
	create_time	Create time of comment	Timestamp	No
	last_updated	Last time updated of comment	Timestamp	No
activity	id	Unique identify of activity.	int	No
	title	Title of activity	varchar(255)	No
	description	Description of activity	text	Yes
	location	Location of	Varchar(255)	No

		activity		
	from_date	Start date of activity	Timestamp	No
	end_date	End date of activity	Timestamp	No
	status	Status of activity	int	No
	create_time	Create time of activity	Timestamp	No
	last_updated	Last time updated of activity	Timestamp	No
class_activity	activity_id	Id of activity.	int	No
	class_id	Id of class	int	No
menu	id	Unique identify of menu.	int	No
	content	Title of menu	varchar(255)	No
	from_date	Start date of menu	Timestamp	No
	from_date	Start date of menu	Timestamp	No
	status	Status of menu	int	No
	create_time	Create time of menu	Timestamp	No
	last_updated	Last time updated of menu	Timestamp	No
class_menu	menu_id	Id of menu	int	No
	class_id	Id of class	int	No
menu_schedule	id	Unique identify of menu's schedule.	int	No

	breakfast	Breakfast of schedule	varchar(255)	No
	afternoon	Afternoon of schedule	varchar(255)	No
	lunch	Lunch of schedule	varchar(255)	No
	time	Time of schedule	Timestamp	No
	status	Status of schedule	int	No
attendance	id	Unique identify of attendance.	int	No
	attendance	Attendance status of child	Int	No
	time	Time check attendance	varchar(255)	No
	child_class_id	Id of child class	int	No
	account_id	Id of account teacher check attendance	int	No
	Remark	Remark of attendance	Timestamp	Yes
	Status	Status of attendance	Int	No
class_grade	id	Unique identify of attendance.	int	No
	name	Name of grade	varchar(255)	No
	status	Status of grade	int	No
	create_time	Create time of grade	Timestamp	No
	last_updated	Last time updated of grade	Timestamp	No

abnormal_case	id	Unique identify of abnormal case.	int	No
	type	Type of abnormal case.	varchar(255)	No
	name	Name of abnormal case	varchar(255)	No
	time	Time check abnormal case	Timestamp	No
	content	Content of abnormal case	text	Yes
	child_class_id	Id of child class	int	No
	status	Status of grade	int	No
	create_time	Create time of grade	Timestamp	No
	last_updated	Last time updated of grade	Timestamp	No

## 1. Test plan

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

### 1.1. <System admin / Operation admin> Create account

ID	Test Case Description	Test Case Procedure	Expected output	Result	Test Date
AC01	Input wrong email format	-input email in label "Email"	Show error message: "Email chưa đúng"	Passed	10/8/2016

AC02	Input wrong birthday	-Input in label: "Ngày sinh".	Show error message: "Ngày sinh chưa nhập đúng"	Passed	10/8/2016
AC03	Input wrong Phone number.	-Not input "Số điện thoại" - Input wrong phone number format	Show error message: "Phải nhập Sđt từ 10 đến 11 số"	Passed	10/8/2016
AC04	Fill data: Right data	-Enter information -Click "Tạo" button	-Close pop up -Create data to database	Passed	10/8/2016

### 1.2. <System admin / Operation dmin> Create class

ID	Test Case Description	Test Case Procedure	Expected output	Result	Test Date
AC01	No input data	-Not input data	Button "Tạo" disabled	Passed	10/8/2016
AC02	Input wrong from date	-Input in label: "Bắt đầu".	Show error message: "Ngày bắt đầu chưa đúng"	Passed	10/8/2016

AC03	Input wrong to date.	- Input in label "Kết thúc"	Show error message: "Ngày kết thúc chưa đúng"	Passed	10/8/2016
AC04	Fill data: Right data	-Enter information -Click "Tạo" button	-Close pop up -Create data to database	Passed	10/8/2016

### 1.3. <System admin / Operation dmin> Create activity

ID	Test Case Description	Test Case Procedure	Expected output	Result	Test Date
AC01	No input data	-Not input data	Button "Tạo" disabled	Passed	10/8/2016
AC02	Input wrong from date	-Input in label: "Bắt đầu".	Show error message: "Ngày bắt đầu chưa đúng"	Passed	10/8/2016

AC03	Input wrong to date.	- Input in label “Kết thúc”	Show error message: “Ngày kết thúc chưa đúng”	Passed	10/8/2016
AC04	Fill data: Right data	-Enter information -Click “Tạo” button	-Close pop up -Create data to database	Passed	10/8/2016

#### 1.4. <System admin / Operation dmin> Create menu

ID	Test Case Description	Test Case Procedure	Expected output	Result	Test Date
AC01	No input data	-Not input data	Button "Tạo" disabled	Passed	10/8/2016
AC02	Input wrong from date	-Input in label: “Bắt đầu”.	Show error message: “Ngày bắt đầu chưa đúng”	Passed	10/8/2016

AC03	Input wrong to date.	- Input in label "Kết thúc"	Show error message: "Ngày kết thúc chưa đúng"	Passed	10/8/2016
AC04	Fill data: Right data	-Enter information -Click "Tạo" button	-Close pop up -Create data to database	Passed	10/8/2016

### 1.5. <Parent> Create child profile

ID	Test Case Description	Test Case Procedure	Expected output	Result	Test Date
AC01	No input data	-Not input data	Button "Tạo" disabled	Passed	10/8/2016
AC02	Input wrong birthday	-Input in label: "Ngày sinh".	Show error message: "Ngày sinh chưa đúng cho bé từ 3 đến 5 tuổi"	Passed	10/8/2016

AC03	Input wrong phone number of parent 2.	- Input in label “Sđt liên lạc”	Show error message: “Số điện thoại chưa đúng”	Passed	10/8/2016
AC04	Fill data: Right data	-Enter information -Click “Tạo” button	-Close pop up -Create data to database	Passed	10/8/2016

### 1.6. <Parent> Edit child profile

ID	Test Case Description	Test Case Procedure	Expected output	Result	Test Date
AC01	No input data	-Not input data	Button "Tạo" disabled	Passed	10/8/2016
AC02	Input wrong birthday	-Input in label: “Ngày sinh”.	Show error message: “Ngày sinh chưa đúng cho bé từ 3 đến 5 tuổi”	Passed	10/8/2016

AC03	Input wrong phone number of parent 2.	- Input in label “Sđt liên lạc”	Show error message: “Số điện thoại chưa đúng”	Passed	10/8/2016
AC04	Fill data: Right data	-Enter information -Click “Tạo” button	-Update data to database -Show success message	Passed	10/8/2016

### 1.7. <Teacher> Health Check

ID	Test Case Description	Test Case Procedure	Expected output	Result	Test Date
AC01	No input data	-Not input data -Click button “Lưu lại”	Show error message: “Cân nặng/ chiều cao cần phải nhập”	Passed	10/8/2016
AC02	Input wrong height	-Input in label: “Chiều cao”.	Show error message: “Chiều cao chưa đúng”	Passed	10/8/2016

AC03	Input wrong weight	- Input in label “Cân nặng”	Show error message: “Cân nặng chưa đúng”	Passed	10/8/2016
AC04	Fill data: Right data	-Enter information -Click “Tạo” button	-Create data to database -Show message.	Passed	10/8/2016

#### 1.8. <Teacher> Create report

ID	Test Case Description	Test Case Procedure	Expected output	Result	Test Date
AC01	No input data of report's time	-Not input data -Click button “Tạo báo cáo”	Show error message: “Chọn thời gian cho báo cáo”	Passed	10/8/2016
AC02	No input comment to all fields	-Not input to all fields comment.	Show error message: “Phải nhập đầy đủ nhận xét”	Passed	10/8/2016
AC03	Fill data: Right data	-Enter information -Click “Tạo” button	-Create data to database -Show message.	Passed	10/8/2016

### **1.8.1.**

# F. Software User's Manual

## 1. Installation Guide

### 1.1 Setting up Environment at Server Side

#### 1.1.1 Hardware Requirements

Server computer's minimum specification for Web Application to be deployed on:

- CPU Intel® Core™ i5 2.53GHz
- 4GB of RAM
- 10GB of free hard disk
- Cable Internet (>= 8Mbps)

#### 1.1.2 Software Requirements

- Java SE Development Kit 8 or Java SE Runtime Environment 8
- MySQL version 5.7.13

## 1.2 Deployment at Server Side

### 1.2.1 Prepare Deployment Package

- Extract file CCSS2016\_Deployment.zip:
  - o ccss2016-v1.0.0.jar
  - o data.sql (default and init data for project)
  - o schemas.sql (schema for project)

Name	Date modified	Type	Size
ccss2016-v1.0.0.jar	8/13/2016 10:13 PM	Executable Jar File	53,101 KB
data.sql	8/12/2016 7:24 PM	SQL Text File	44 KB
schemas.sql	8/13/2016 8:29 PM	SQL Text File	23 KB

Figure 21: Prepare deployment package for web application

### 1.2.2 Configure Server before Deploy

N/A

### 1.2.3 Deploy Database

- Open MySQL Workbench to connect with mysql server.
  - o Must setting mysql in local of server where deploy this website.
  - o Config mysql:
    - Username: root
    - Password: root

- Run two scripts in Deployment Folder (two sql files).
  - o First, run schemas.sql file.
  - o Next, run data.sql file.

```

1 • CREATE DATABASE IF NOT EXISTS `ccss_2016`;
2 • USE `ccss_2016`;
3
4 • /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
5 • /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
6 • /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
7 • /*!40101 SET NAMES utf8 */;
8 • /*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
9 • /*!40103 SET TIME_ZONE='+07:00' */;
10 • /*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
11 • /*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
12 • /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
13 • /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

14
15
16 -- Table structure for table `class_grade`
17 --
18 • DROP TABLE IF EXISTS `class_grade`;
19 • /*!40101 SET @saved_cs_client      = @@character_set_client */;
20 • /*!40101 SET character_set_client = utf8 */;
21 • CREATE TABLE `class_grade` (
22   `id` INT(10) UNSIGNED NOT NULL AUTO_INCREMENT,
23   `name` NVARCHAR(255) NOT NULL,
24   `status` INT(10) NOT NULL,
25   `create_time` TIMESTAMP DEFAULT NOW(),
26   `last_updated` TIMESTAMP DEFAULT NOW(),
27   PRIMARY KEY (`id`)

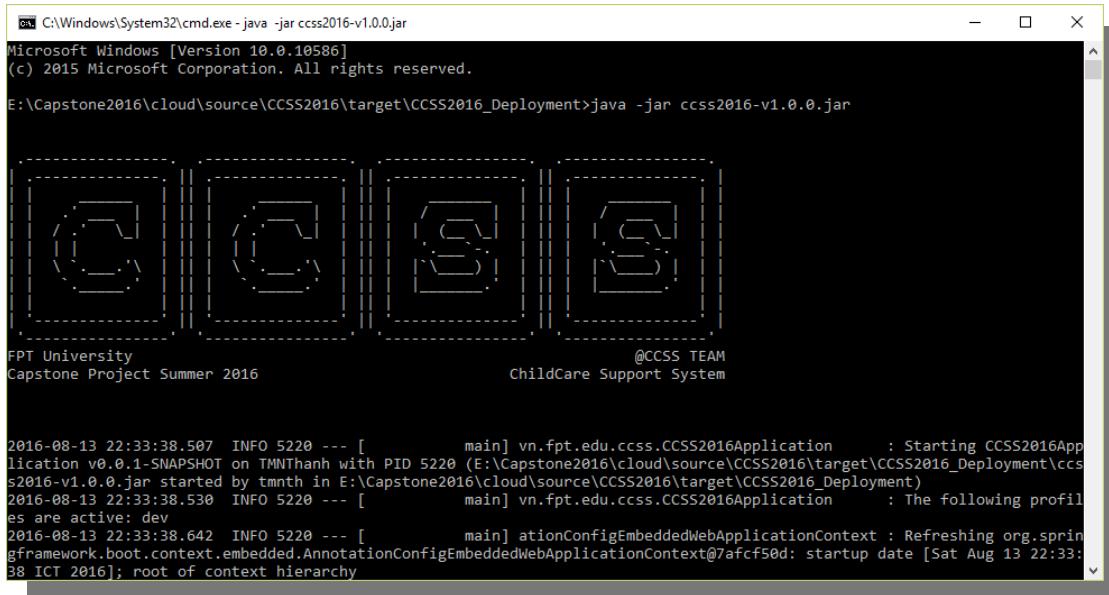
```

**Figure 22: Database deployment**

#### 1.2.4 Deploy Web Application on Server

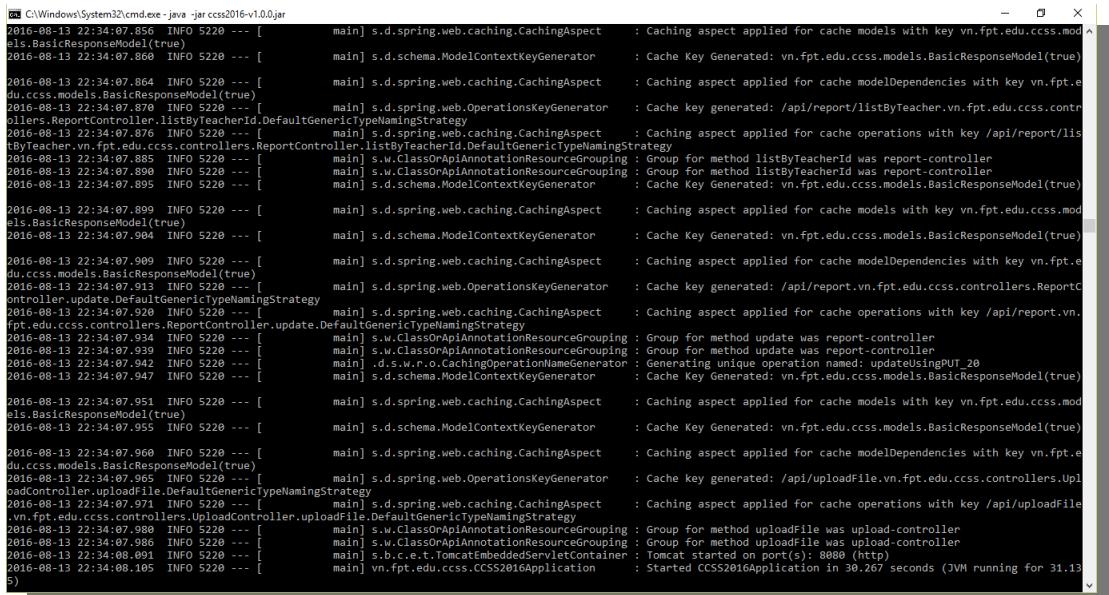
This project is using Spring Boot framework so they just have a jar file, then run them by java virtual machine.

- Open **Command Prompt (CMD)** in CCSS\_Development folder.
- Run command: `java -jar ccss2016-v1.0.0.jar`



**Figure 23: Deploy website in server.**

- Check log if website is ready, open browser and go to address: <http://localhost:8080/>. Then it will go to the home screen of website.



**Figure 24: Deploy website in server.**

## 1.3 Setting up the Environment at Client Side

### 1.3.1 Setting up for Computer

#### 1.1.1.1 Hardware Requirements

- Internet connection (>= 3.6Mbps)

#### 1.1.1.2 Software Requirements

Client need to use one of the following browsers to be able to access the website:

- Google Chrome 38 or newer
- Internet Explorer 10
- Firefox 32 or newer

### **1.3.2 Setting up for Mobile Device**

#### **1.1.1.3 Hardware Requirements**

Minimum specification for Mobile devices

- 512MB of RAM
- 20MB of hard disk
- Wi-Fi Internet or 3G Network ( $\geq$  3.6Mbps)
- Camera
- GPS

#### **1.1.1.4 Software Requirements**

For iPhone and iPad devices:

- iOS 6 or later

For Android devices:

- Android 4.2.1 (Jellybean) or later

#### **1.1.1.5 Installation**

Download application installable package from Website, by select platform according to current device's platform.

Link Google Play for download application.

## 2. User Guide

### 2.1. Operation Administrator Screens

#### 2.1.1. Login screen for operation administrator

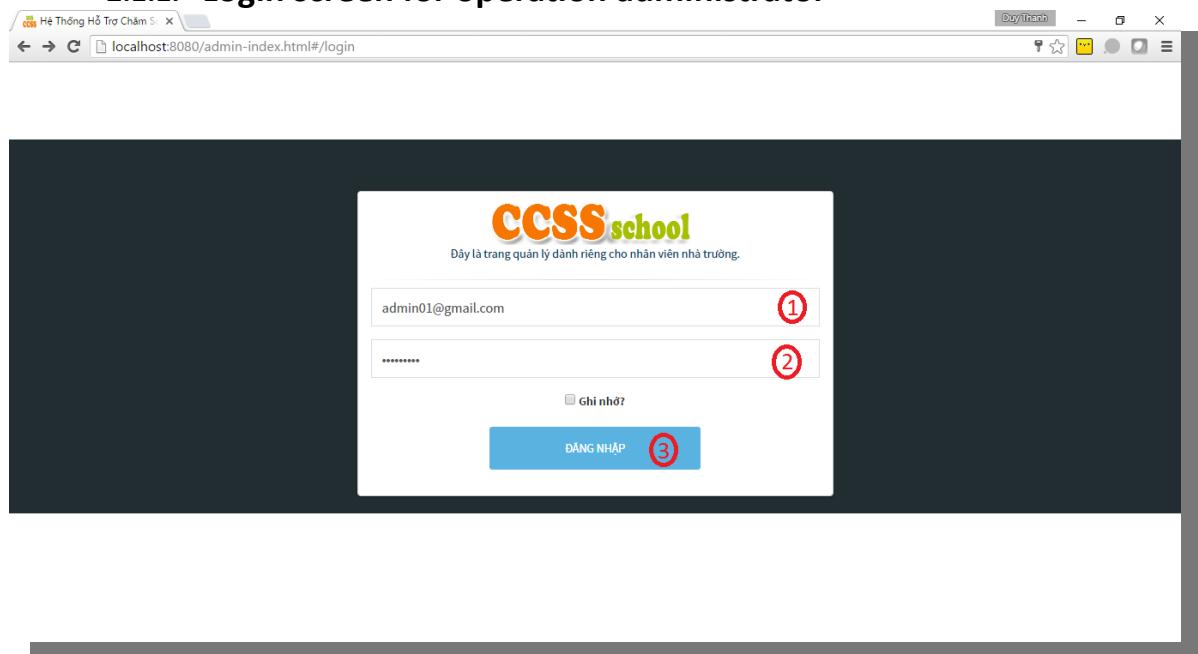


Figure 25: Operation Administrator Login Screen

#	Name	Description
1	Username input	Input the username of user. Username is email of that user.
2	Password input	Input the password of user.
3	Login Button	Button for logging in function.

#### 2.1.2. Manage User Account Screens

##### 2.1.2.1. Manage Parent Account Screen

**Figure 26: Manage Parent Account Screen**

#	Name	Description
1	Create button	Create a new parent account. Open <i>Create new parent account</i> pop-up.
2	Parent Account list	List of current parent accounts in system.
3	Search input	Search a name, phone, address... in list.
4	Edit button	Edit the parent account in that row. Open <i>Edit parent account</i> pop-up.
5	Delete button	Delete the parent account in that row. Confirm message will ask before delete a parent account.

## Tạo mới tài khoản cho phụ huynh

Tài khoản cấp cho phụ huynh, người thân, người chịu trách nhiệm cho trẻ.

(\*): Thông tin bắt buộc nhập

Email(\*):

Email của phụ huynh cung cấp. Mỗi email đại diện cho một tài khoản duy nhất.

Mật khẩu(\*):

Mật khẩu có ít nhất 6 ký tự, mật khẩu sẽ cung cấp cho phụ huynh để đăng nhập vào hệ thống.

Xác nhận mật khẩu(\*):

Xác nhận lại mật khẩu nhằm đảm bảo mật khẩu đã nhập chính xác.

Họ và tên(\*):

Họ tên đầy đủ của phụ huynh cung cấp.

Ngày sinh(\*):

mm/dd/yyyy

Ngày tháng năm sinh của phụ huynh cung cấp.

Giới tính(\*):

Nam  Nữ

Chọn giới tính cho tài khoản phụ huynh.

Số điện thoại(\*):

Số điện thoại liên lạc với phụ huynh.

Số điện thoại 2:

Số điện thoại dự phòng, phòng khi số điện thoại chính không thể liên lạc trong những trường hợp khẩn cấp.

Địa chỉ(\*):

Địa chỉ 1

Địa chỉ của phụ huynh cung cấp (địa chỉ nhà).

Địa chỉ 2:

Địa chỉ 2

Địa chỉ phụ của phụ huynh cung cấp (địa chỉ cơ quan, nhà người thân).

**Tạo** **Hủy**

**Figure 27: Create new parent account pop-up**

## Thông tin tài khoản phụ huynh

Tài khoản cấp cho phụ huynh, người thân, người chịu trách nhiệm cho trẻ.

(\*): Thông tin bắt buộc nhập

Email(\*):

parent01@gmail.com

Email của phụ huynh cung cấp. Mỗi email đại diện cho một tài khoản duy nhất.

Họ và tên(\*):

Trần Văn Sang

Họ tên đầy đủ của phụ huynh cung cấp.

Ngày sinh(\*):

01/12/1990

Ngày tháng năm sinh của phụ huynh cung cấp.

Giới tính(\*):

Nam  Nữ

Chọn giới tính cho tài khoản phụ huynh.

Số điện thoại(\*):

01234567443

Số điện thoại liên lạc với phụ huynh.

Số điện thoại 2:

Số điện thoại dự phòng, phòng khi số điện thoại chính không thể liên lạc trong những trường hợp khẩn cấp.

Địa chỉ(\*):

27 XVNT, Bình Thạnh, HCM

Địa chỉ của phụ huynh cung cấp (địa chỉ nhà).

Địa chỉ 2:

Địa chỉ 2

Địa chỉ phụ của phụ huynh cung cấp (địa chỉ cơ quan, nhà người thân).

Lưu

Hủy

Figure 28: Edit parent account pop-up

### 2.1.2.2. Manage Teacher Account Screen

Figure 29: Manage Teacher Account Screen

#	Name	Description
1	Create button	Create a new teacher account. Open <i>Create new teacher account</i> pop-up.
2	Teacher Account list	List of current teacher accounts in system.
3	Search input	Search a name, phone, address... in list.
4	Edit button	Edit the teacher account in that row. Open <i>Edit teacher account</i> pop-up.
5	Delete button	Delete the teacher account in that row. Confirm message will ask before delete a teacher account.

## Tạo mới tài khoản cho giáo viên

Tài khoản cấp cho giáo viên / nhân viên của nhà trường.

(\*): Thông tin bắt buộc nhập

Email(\*):

Email của giáo viên cung cấp. Mỗi email đại diện cho một tài khoản duy nhất.

Mật khẩu(\*):

Mật khẩu có ít nhất 6 ký tự, mật khẩu sẽ cung cấp cho giáo viên để đăng nhập vào hệ thống.

Xác nhận mật khẩu(\*):

Xác nhận lại mật khẩu nhằm đảm bảo mật khẩu đã nhập chính xác.

Họ và tên(\*):

Họ tên đầy đủ của giáo viên cung cấp.

Ngày sinh(\*):

Ngày tháng năm sinh của giáo viên cung cấp.

Giới tính(\*):

Nam  Nữ

Chọn giới tính cho tài khoản giáo viên.

Số điện thoại(\*):

Số điện thoại liên lạc với giáo viên.

Số điện thoại 2:

Số điện thoại dự phòng, phòng khi số điện thoại chính không thể liên lạc trong những trường hợp khẩn cấp.

Địa chỉ(\*):

Địa chỉ của giáo viên cung cấp (địa chỉ nhà).

Địa chỉ 2:

Địa chỉ phụ của giáo viên cung cấp (địa chỉ cơ quan, nhà người thân).

Tạo

Hủy

Figure 30: Create new teacher account pop-up

## Thông tin tài khoản giáo viên

Tài khoản cấp cho giáo viên / nhân viên của nhà trường.

(\*): Thông tin bắt buộc nhập

Email(\*):

teacher01@gmail.com

Email của giáo viên cung cấp. Mỗi email đại diện cho một tài khoản duy nhất.

Họ và tên(\*):

Võ Thị Tuyết Vân

Họ tên đầy đủ của giáo viên cung cấp.

Ngày sinh(\*):

11/23/1972

Ngày tháng năm sinh của giáo viên cung cấp.

Giới tính(\*):

Nam  Nữ

Chọn giới tính cho tài khoản giáo viên.

Số điện thoại(\*):

01233342234

Số điện thoại liên lạc với giáo viên.

Số điện thoại 2:

Số điện thoại dự phòng, phòng khi số điện thoại chính không thể liên lạc trong những trường hợp khẩn cấp.

Địa chỉ(\*):

233 XVNT, Bình Thạnh, HCM

Địa chỉ của giáo viên cung cấp (địa chỉ nhà).

Địa chỉ 2:

Địa chỉ 2

Địa chỉ phụ của giáo viên cung cấp (địa chỉ cơ quan, nhà người thân).

Lưu

Hủy

**Figure 31: Edit teacher account pop-up**

## 2.2. System Administrator Screens

### 2.2.1. Login screen for system administrator

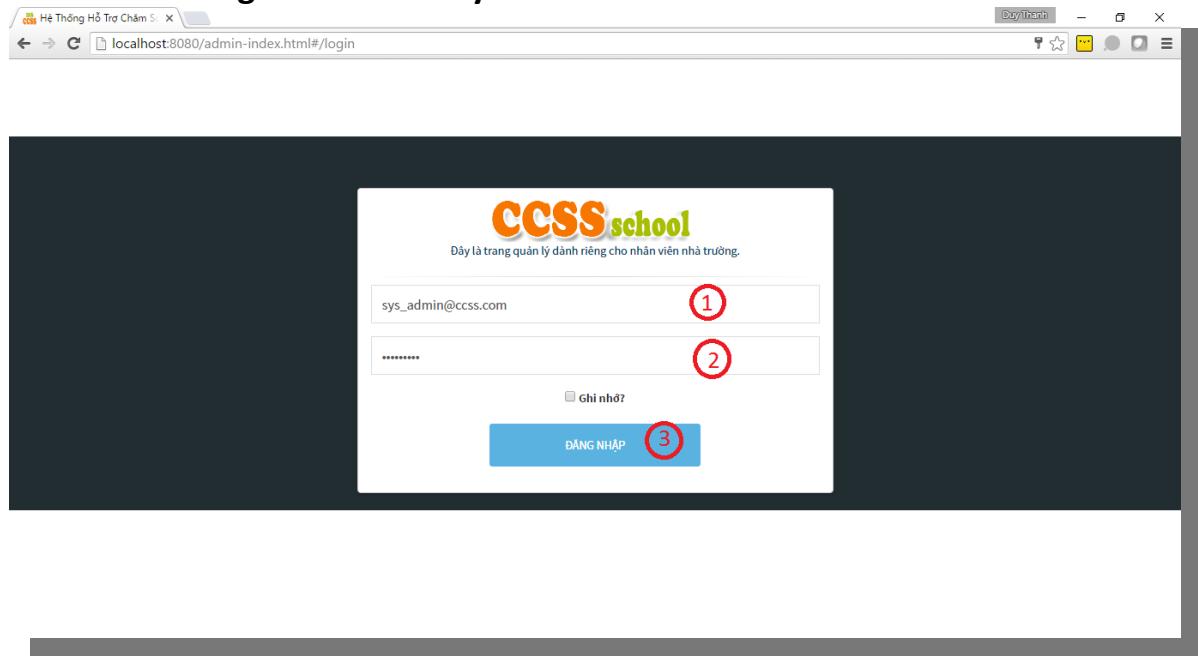


Figure 32: System administrator login screen

#	Name	Description
1	Username input	Input the username of user. System administrator has only one account: <b>sys_admin@ccss.com</b>
2	Password input	Input the password of user.
3	Login Button	Button for logging in function.

## 2.2.2. Manage User Account

### 2.2.2.1. Manage Operation Administrator Account

Figure 33: Operation Administrator Management Screen

#	Name	Description
1	Create button	Create a new operation administrator account. Open <i>Create new operation administrator account</i> pop-up.
2	Operation Administrator Account list	List of current operation administrator accounts in system.
3	Search input	Search a name, phone, address... in list.
4	Edit button	Edit the operation administrator account in that row. Open <i>Edit operation administrator account</i> pop-up.
5	Delete button	Delete the operation administrator account in that row. Confirm message will ask before delete an operation administrator account.

## Tạo mới tài khoản cho quản trị viên

Tài khoản cấp cho quản trị viên, nhân viên quản lý của nhà trường.

(\*): Thông tin bắt buộc nhập

Email(\*):

Email của quản trị viên cung cấp. Mỗi email đại diện cho một tài khoản duy nhất.

Mật khẩu(\*):

Mật khẩu có ít nhất 6 ký tự, mật khẩu sẽ cung cấp cho quản trị viên để đăng nhập vào hệ thống.

Xác nhận mật khẩu(\*):

Xác nhận lại mật khẩu nhằm đảm bảo mật khẩu đã nhập chính xác.

Họ và tên(\*):

Họ tên đầy đủ của quản trị viên cung cấp.

Ngày sinh(\*):

mm/dd/yyyy

Ngày tháng năm sinh của quản trị viên cung cấp.

Giới tính(\*):

Nam  Nữ

Chọn giới tính cho tài khoản quản trị viên.

Số điện thoại(\*):

Số điện thoại liên lạc với quản trị viên.

Số điện thoại 2:

Số điện thoại dự phòng, phòng khi số điện thoại chính không thể liên lạc trong những trường hợp khẩn cấp.

Địa chỉ(\*):

Địa chỉ 1

Địa chỉ của quản trị viên cung cấp (địa chỉ nhà).

Địa chỉ 2:

Địa chỉ 2

Địa chỉ phụ của quản trị viên cung cấp (địa chỉ cơ quan, nhà người thân).

Tạo

Hủy

Figure 34: Create new operation administrator account pop-up

## Thông tin tài khoản quản trị viên

Tài khoản cấp cho quản trị viên, nhân viên quản lý của nhà trường.

(\*): Thông tin bắt buộc nhập

Email(\*):

admin01@gmail.com

Email của quản trị viên cung cấp. Mỗi email đại diện cho một tài khoản duy nhất.

Họ và tên(\*):

Hoàng Thị Lan

Họ tên đầy đủ của quản trị viên cung cấp.

Ngày sinh(\*):

05/12/1991

Ngày tháng năm sinh của quản trị viên cung cấp.

Giới tính(\*):

Nam  Nữ

Chọn giới tính cho tài khoản quản trị viên.

Số điện thoại(\*):

01690394953

Số điện thoại liên lạc với quản trị viên.

Số điện thoại 2:

Số điện thoại dự phòng, phòng khi số điện thoại chính không thể liên lạc trong những trường hợp khẩn cấp.

Địa chỉ(\*):

Quang Trung, Quận 12, HCM

Địa chỉ của quản trị viên cung cấp (địa chỉ nhà).

Địa chỉ 2:

Địa chỉ 2

Địa chỉ phụ của quản trị viên cung cấp (địa chỉ cơ quan, nhà người thân).

Lưu

Hủy

Figure 35: Edit operation administrator account pop-up

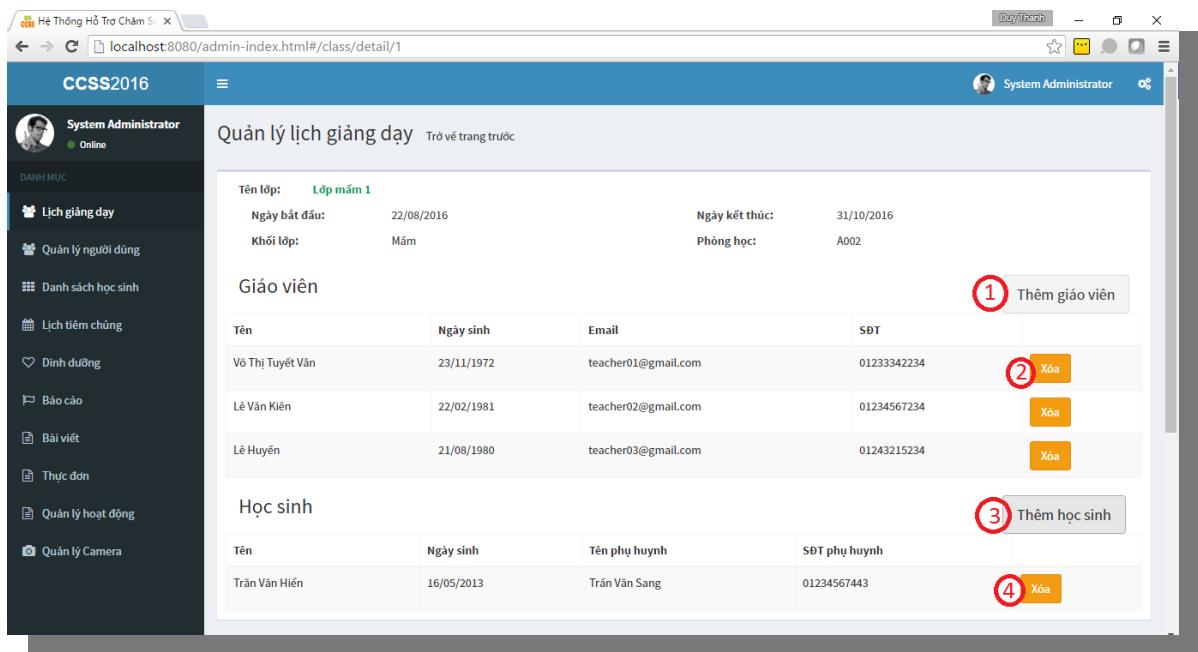
### 2.2.3. Class management screen

The screenshot shows a web-based application interface for managing classes. On the left is a sidebar with a user profile picture and the text "Online". Below this are several menu items under "DANH MỤC": "Lịch giảng dạy", "Quản lý người dùng", "Quản trị viên", "Phụ huynh", "Giáo viên", "Danh sách học sinh", "Lịch tiêm chủng", "Dinh dưỡng", "Báo cáo", "Bài viết", "Thực đơn", "Quản lý hoạt động", and "Quản lý Camera". The main content area has a title "Quản lý lịch giảng dạy". A modal dialog titled "Tạo mới lớp học" (Create new class) is open, with a red circle labeled "1" above it. The main table lists four classes:

② Tên	Ngày bắt đầu	Dự kết thúc	Phòng	Mô tả	Loại	Sửa	Xóa	Quản lý lớp
Lớp mầm 1	22/08/2016	31/10/2016	A002	Lớp mầm 1 cho các bé vừa nhập học.	Mầm	<span>3 Sửa</span>	<span>4 Xóa</span>	<span>5 Quản lý lớp</span>
Lớp mầm 2	22/08/2016	31/10/2016	A002	Lớp mầm 2 cho các bé chuyển từ lớp giữ trẻ.	Mầm	<span>Sửa</span>	<span>Xóa</span>	<span>Quản lý lớp</span>
Lớp chồi 1	22/08/2016	31/12/2016	B02	Từ các bé mầm 1 kỳ trước.	Chồi	<span>Sửa</span>	<span>Xóa</span>	<span>Quản lý lớp</span>
Lớp chồi 2	22/08/2016	31/12/2016	B003	Các bé chuyển từ trường khác.	Chồi	<span>Sửa</span>	<span>Xóa</span>	<span>Quản lý lớp</span>

Figure 36: Class management screen

#	Name	Description
1	Create button	Create a new class. Open <i>Create new class</i> pop-up.
2	Class list	List of current classes in system.
3	Edit button	Edit the class in that row. Open <i>Edit class</i> pop-up.
4	Delete button	Delete the operation administrator account in that row. Confirm message will ask before a class.
5	Manage class button	Go to class manage screen. To assign teachers and children to that class.



**Figure 37: Class management screen**

#	Name	Description
1	Teacher assign button	Assign teachers to this class.
2	Remove teacher assign button	Remove a teacher who has assigned to this class.
3	Child assign button	Assign children to this class.
4	Remove child assign button	Remove a child who has assigned to this class.

## 2.3. Parent Screens

### 2.3.1. Parent Login Screen

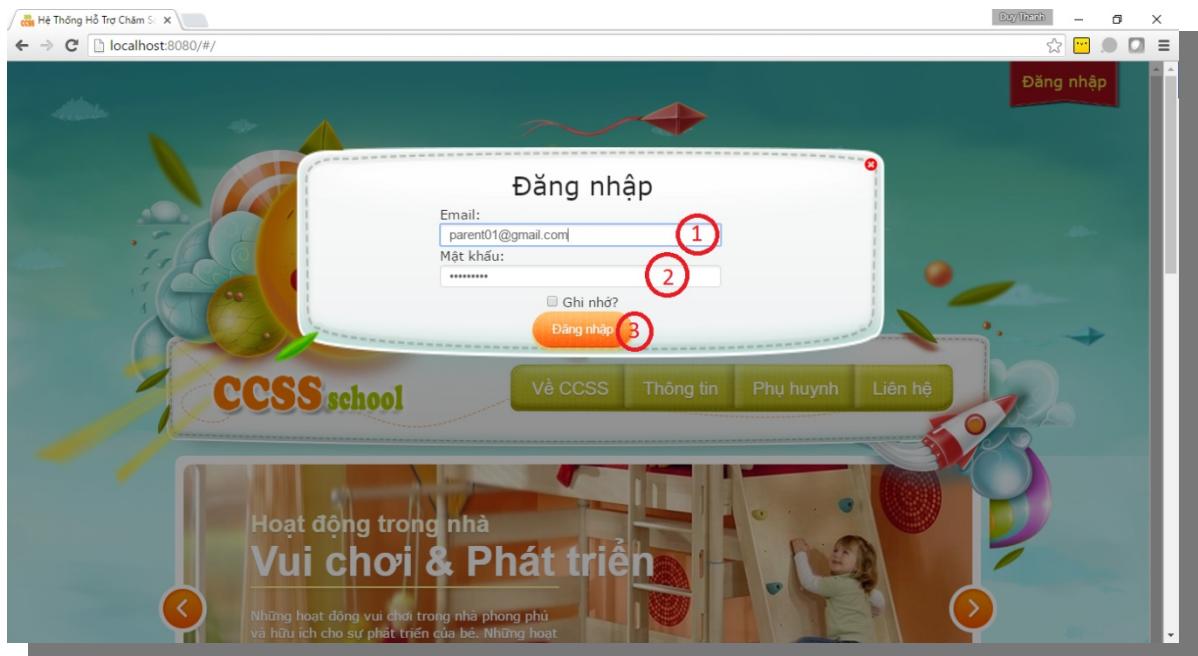


Figure 38: Parent login pop-up

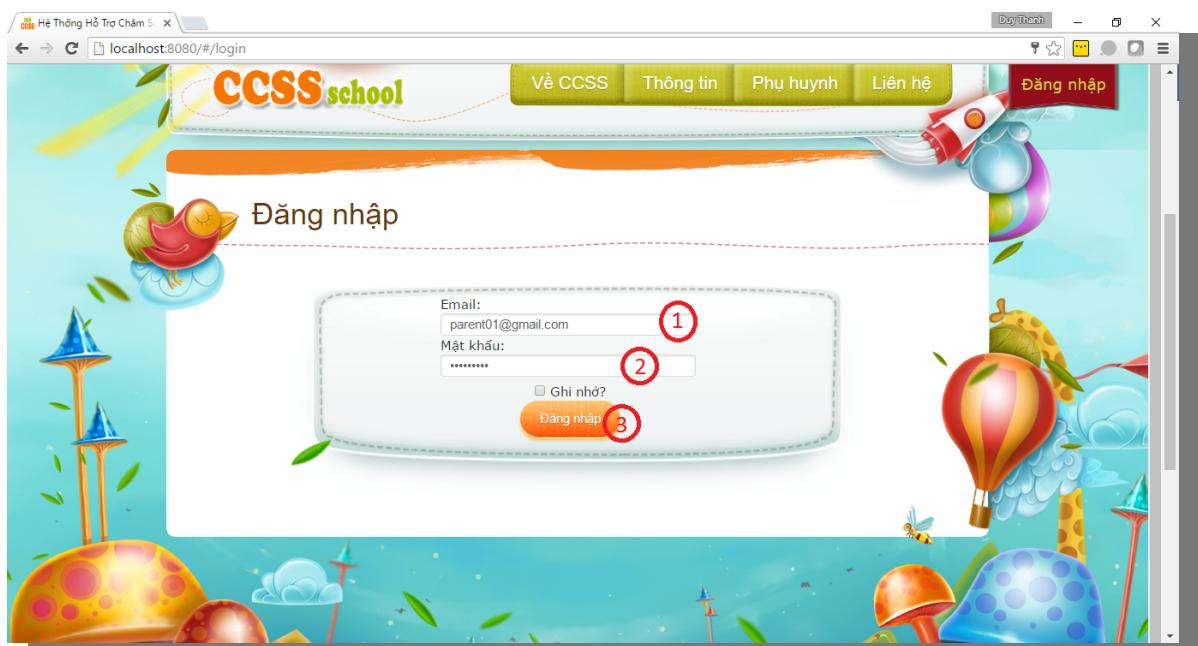


Figure 39: Paren login screen

#	Name	Description
1	Username input	Input the username of user. Username is email of user.
2	Password input	Input the password of user.
3	Login Button	Button for logging in function.

### 2.3.2. Children List Screen

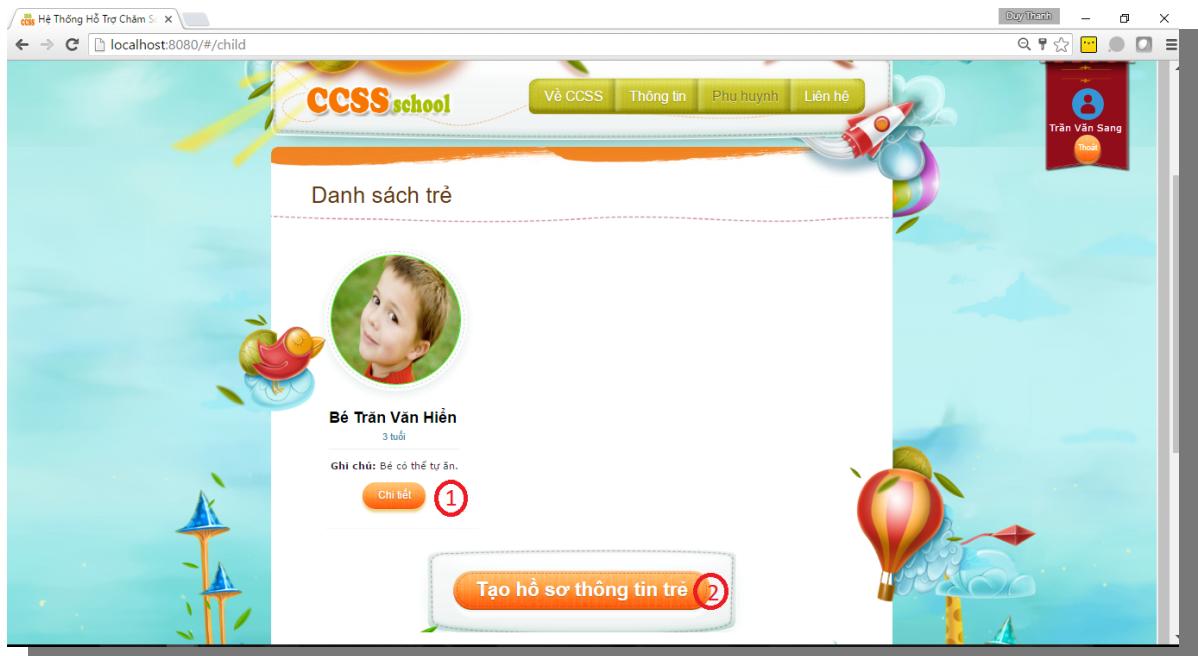


Figure 40: Children List Screen

#	Name	Description
1	Detail button	Go to child detail profile.
2	Create button	Create a new child profile. Go to <i>create new child profile</i> screen.

### 2.3.3. Create New Child Profile Screen

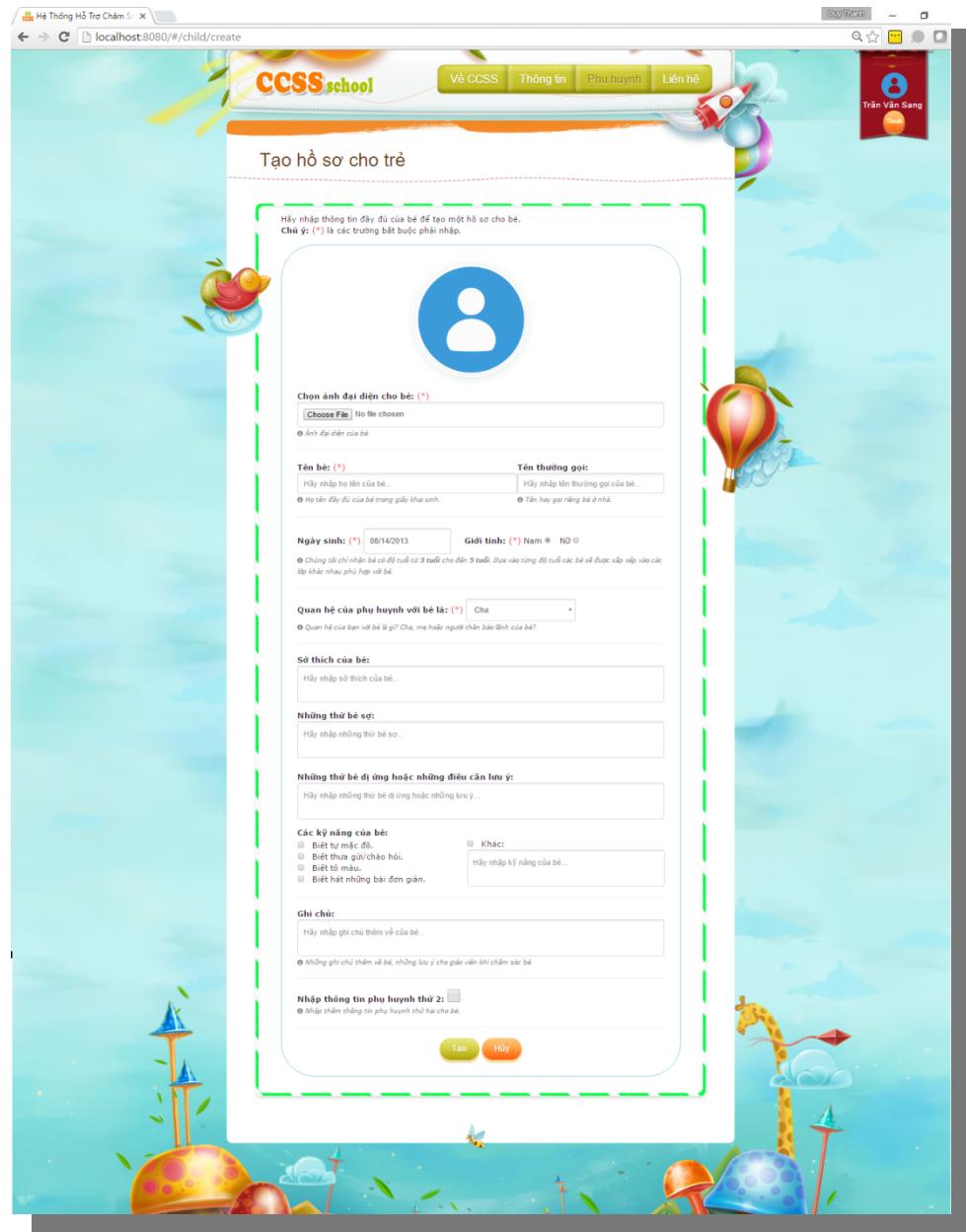
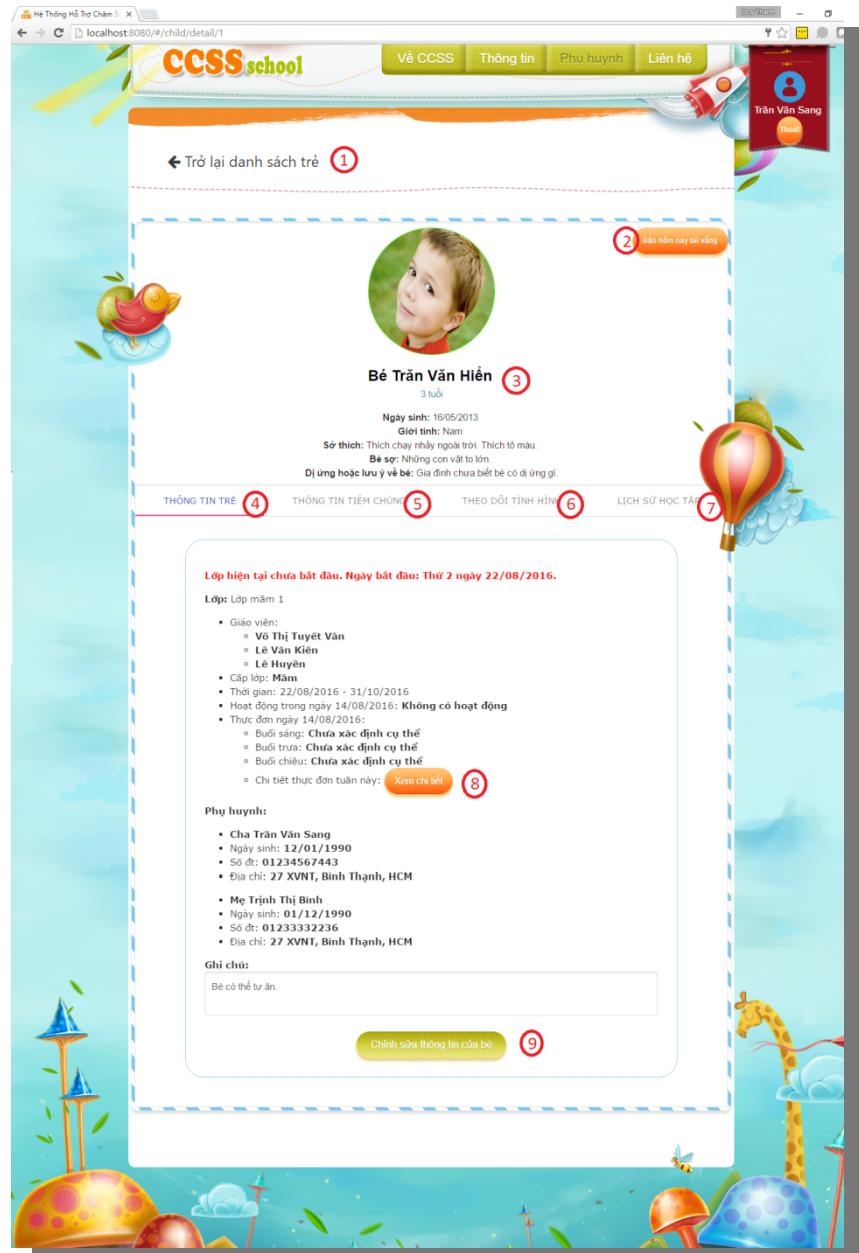


Figure 41: Create new child profile screen

### 2.3.4. Child Profile Detail Screen



**Figure 42: Child profile detail screen.**

#	Name	Description
1	Return child profile list button	Return child profile list.
2	Create Absent Form for child	Create an absent form to notify to teachers the day child can not go to school and reason.
3	Child Information	The information of child. <ul style="list-style-type: none"> <li>- Full name</li> <li>- Age</li> </ul>

		- Hobby, fear...
4	Child Information Tab	This tab contains the current and newest child information in class (attendance, menu, activity).
5	Child Immunization Tab	This tab contains the immunization schedule of child.
6	Child Tracking Tab	This tab contains the child development about child health (height, weight), education rating.
7	Child history class Tab	This tab contains all information about child in class.
8	Menu detail button	Open menu detail popup. Show current menu of current class.
9	Edit child profile information button	Go to edit child profile screen.

## 2.4. Teacher Screens

After login successful to system with role is Teacher, System will redirect to Teaching schedule screen.

### 2.4.1. Teaching schedule screen

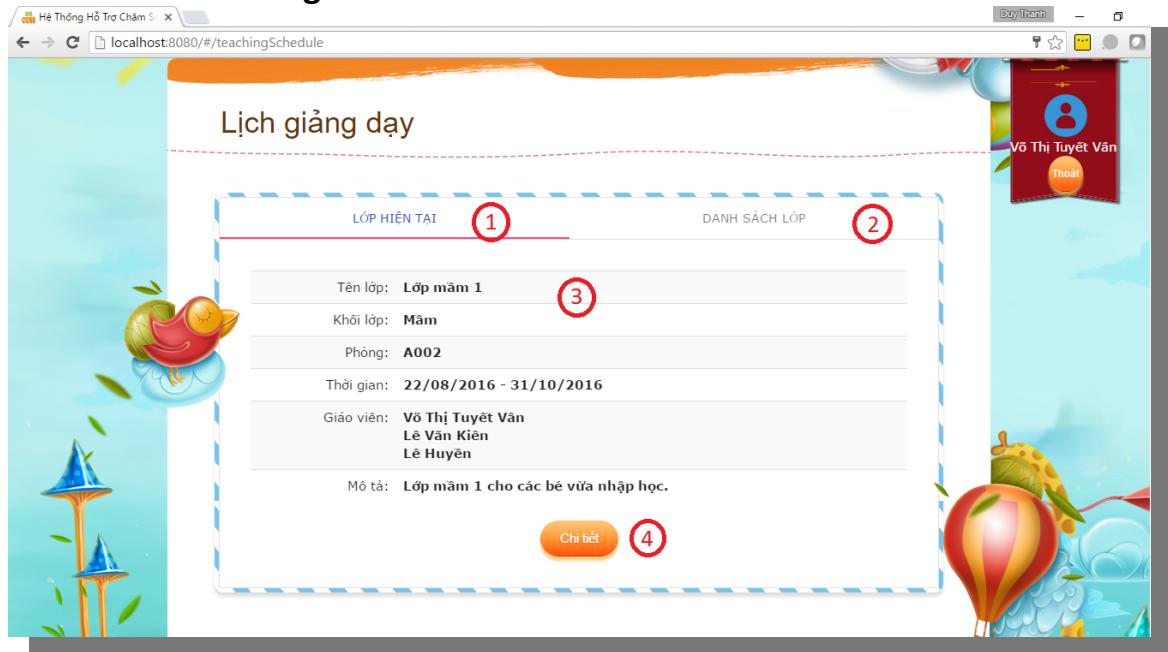
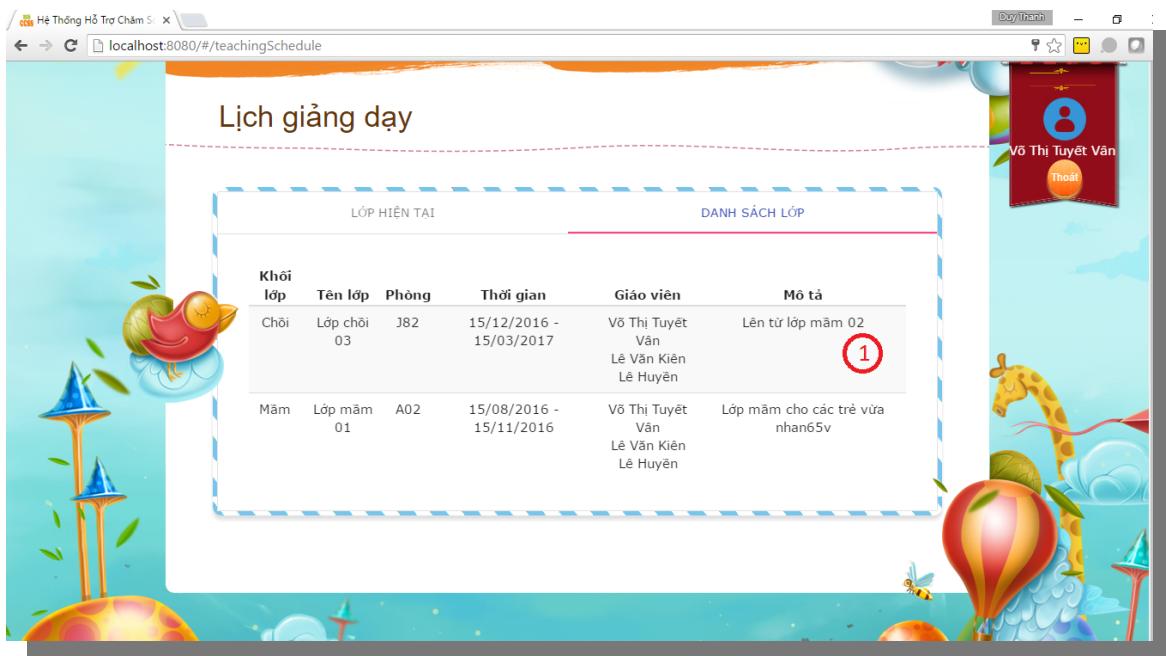


Figure 43: Teaching schedule screen

#	Name	Description
1	Current Class Tab	View the current class information for the teacher.
2	Class List Tab	This tab contains all class which has assigned to current teacher.
3	Current class information	Show all information of current class which this teacher has assigned.
4	Class Detail Button	Go to detail of class for manage all child in that class.



**Figure 44: Teaching schedule – Class list tab**

#	Name	Description
1	Class list	Show information of all class which this teacher has assigned. Click for going to <i>class detail</i> screen.

### 2.4.2. Class Detail Screen

The screenshot displays the 'Class Detail Screen' of a web application. The interface is designed with a colorful, cartoonish background featuring mushrooms, a hot air balloon, and a giraffe.

**Class Information Area (1):** Shows the class name 'Lớp Lớp chồi 03', the teacher's name 'Lê Thị Lành', and the class code '02'. It also displays the room number 'J82', start date '15/12/2016', and end date '15/03/2017'.

**Action Buttons:**

- (2) Điểm danh hôm nay (Check attendance)
- (3) Kiểm tra sức khỏe (Check child health)
- (4) Tạo báo cáo (Create report)

**Danh sách trẻ (5):** A table showing student details. The columns are BÌNH THƯỜNG (Normal), THIẾU CÁN (Underweight), and THỪA CÁN (Overweight). The rows show:

Tên bé	Ngày sinh	Giới tính	Ghi chú
Trần Văn Hiển	04/01/2013	Nam	Bé không thích người lạ
Phạm Trang	09/07/2013	Nữ	

**Hoạt động của lớp (6):** A table showing class activities. The columns are Hoạt động (Activity), Địa điểm (Location), Thời gian (Time), and Mô tả (Description). The row shows:

Hoạt động	Địa điểm	Thời gian	Mô tả
Đi thao công viên	Thảo cầm viên Hồ Chí Minh	Từ ngày 15/12/2016 - đến ngày 17/12/2016	Tham quan thảo cầm viên nhằm kích thích sự hiếu biết của các bé về các loại động vật xung quanh cũng như cho các bé có các hoạt động hữu ích ngoài trời.

**Thực đơn của lớp (7):** A table showing the class menu. The columns are Ngày (Date), Bữa sáng (Breakfast), Bữa trưa (Lunch), and Bữa xế (Dinner). The rows show:

Ngày	Bữa sáng	Bữa trưa	Bữa xế
15/12/2016	Cơm Canh	Cơm Canh	Bánh gạo
16/12/2016	Bánh uýt nóng	Cơm Canh	Bánh mì
19/12/2016	Bánh tiêu	Cơm Canh	Bánh canh
20/12/2016	Bánh chuối	Cơm Canh	Bánh canh
21/12/2016	Bánh gạo	Cơm Canh	Bánh chuối
22/12/2016	Bánh canh	Cơm Canh	Bánh tiêu
23/12/2016	Bánh canh	Cơm Canh	Bánh gạo

**Buttons:**

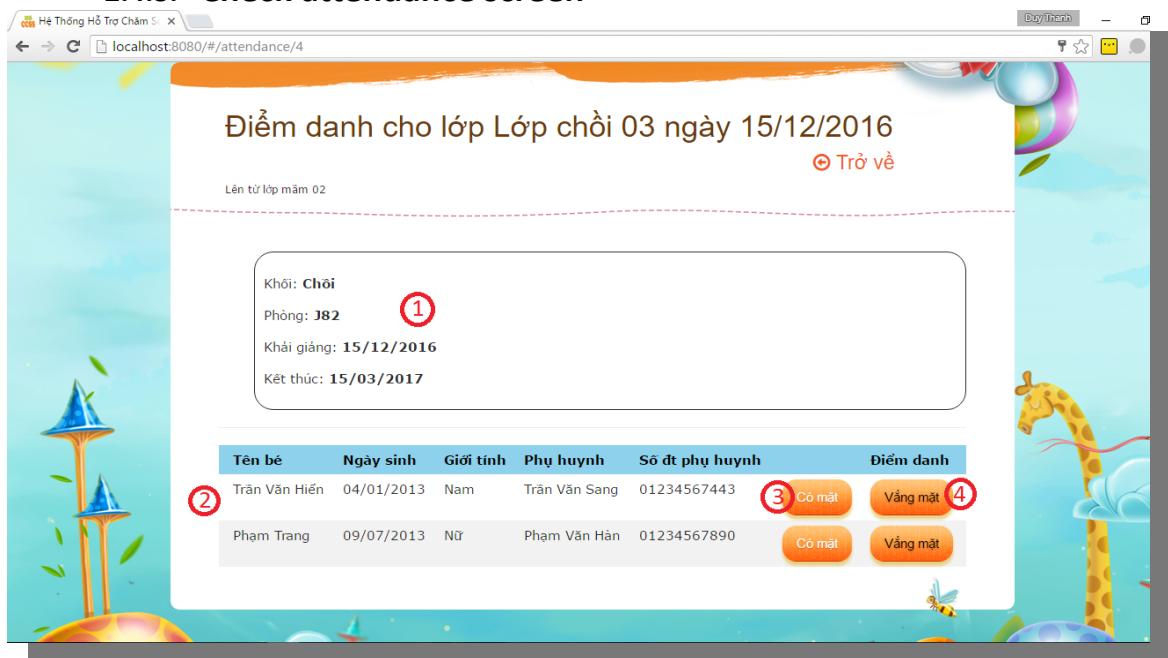
- (8) Trở lại danh sách lớp (Return to class list)

Figure 45: Class detail screen

#	Name	Description
1	Class information area	Current class information
2	Check attendance button	Go to check attendance screen
3	Check child health button	Go to check child health screen

4	Create report button	Go to <i>create report</i> screen
5	Child list in class	This area have three tabs: - Normal child list - Overweight child list - Underweight child list
6	All activity in class	Show all activities of class.
7	Menu in class	Show current menu of class (weekly menu or short time menu)
8	Back to teaching schedule button	Go back to <i>teaching schedule</i> screen

### 2.4.3. Check attendance screen



**Figure 46: Check attendance screen**

#	Name	Description
1	Class information area	Current class information
2	Child list	List of child with basic information of child.
3	Present button	Check present for child.
4	Absent button	Check absent for child, need input the reason.

#### 2.4.4. Child health screen

Tên bé	Tuổi	Giới tính	Cân nặng (kg)	Chiều cao (cm)	Tình trạng	Định mức sức khỏe	Cập nhật lần cuối
Trần Văn Hiển	4 tuổi (47 tháng tuổi)	Nam	3	50	Cân nặng dưới chuẩn cấp 3	Cân nặng dưới chuẩn cấp 3	15/12/2016
Phạm Trang	3 tuổi (41 tháng tuổi)	Nữ	6	60	Cân nặng dưới chuẩn cấp 3	Cân nặng dưới chuẩn cấp 3	15/12/2016

Figure 47: Child health screen

#	Name	Description
1	Class information area	Current class information
2	Input child health data button	Go to <i>input child health data screen</i> .
3	Date check selectbox	The list of date which all the time of checking health for child in current class.
4	Child list	Child list with the child health data (height, weight...) follow the selected date in (3).

#### 2.4.5. Input child health data screen

Hệ Thống Hỗ Trợ Chăm Sóc Mầm Non

localhost:8080/#/healthCheck/4

Lên từ lớp mầm 02

Kiểm tra sức khỏe cho lớp Lớp chồi 03

Trở về

Khối: Chồi  
Phòng: J82  
Khởi giảng: 15/12/2016  
Kết thúc: 15/03/2017

Tên bé	Giới tính	Cân nặng (kg)	Chiều cao (cm)	Tình trạng	Không thể kiểm tra	Định mức sức khỏe	Cập nhật lần cuối
Trần Văn Hiển (47 tháng tuổi)	Nam	3	50	<input type="checkbox"/>	Cân nặng dưới chuẩn cấp 3	15/12/2016	
Phạm Trang (41 tháng tuổi)	Nữ	6	60	<input type="checkbox"/>	Chiều cao bé dưới chuẩn cấp 3	15/12/2016	

Lưu lại

Figure 48: Input data child health screen

#	Name	Description
1	Class information area	Current class information
2	Child list	Child list with height/weight and health situation input of child.
3	Save data button	Save all data. Back to child health screen.

#### 2.4.6. Create child report screen

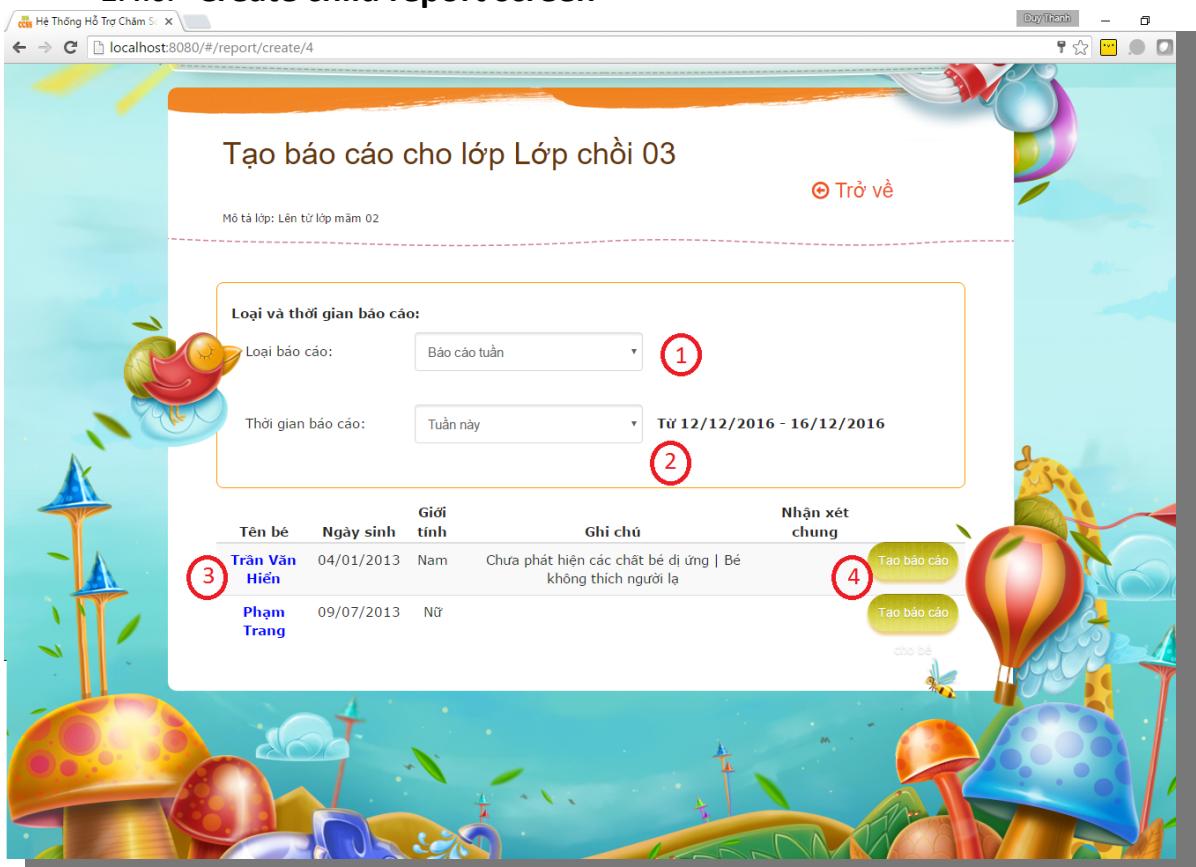


Figure 49: Create child report screen

#	Name	Description
1	Report Type selectbox	Report type: - Weekly report - Monthly report - Final report
2	Report Time selectbox	Report time: - Current time follow by type of report - The 5 before time of report (last week, last year...)
3	Child list	List of child with some information of each child.
4	Report create button	Open the <i>create child report</i> pop-up.

Nhận xét cho bé Trần Văn Hiển X

**Thông tin sức khỏe kiểm tra ngày 15/12/2016**

Cân nặng: **20 kg** 1

Chiều cao: **80 cm**

Lưu ý: Bé dễ sốt khi trời gió

Trong khoảng thời gian này bé không có lịch tiêm chủng. 2

Xếp loại: Bé ngoan cấp trường 3

**Nhận xét chung:** 4

**Nhận thức:**  
Nhận thức của bé tốt.

**Ngôn ngữ:**  
Ngôn ngữ phát triển tốt.

**Vận động - Tự phục vụ bản thân:**  
Bé có khả năng tự phục vụ bản thân cơ bản.

**Lễ giáo - Giao tiếp:**  
Bé tập chào hỏi, thưa gửi tốt.

**Thẩm mỹ - Năng khiếu:**  
Năng khiếu phát triển tốt.

**Nhận xét chung - Trao đổi với phụ huynh:**  
Bé ngoan, lễ phép, hiền và chơi vui vẻ với các bạn.

5 Tao 6 Hủy

Figure 50: Create child report popup

#	Name	Description
1	Child health recent check information area	The last information of child health. - Current weight - Current height - Last comment about child health
2	Child immunization in time	List of child immunization with time scope of report.

	scope of report	
3	Education rating	Selectbox for some popular rating of child, you can input other rating for child.
4	The comments of teacher input	The area, teacher can input the comments in all aspects of child.
5	Create button	Create button for saving all data.
6	Cancel button	Cancel to close current popup.

## G. Appendix

### 1. Scrum model:

- <https://www.atlassian.com/agile/scrum>
- <http://www.netspecglobal.com/approach.html>
- <https://www.braintrustgroup.com/scrum/benefits-of-scrum>

### 2. Coding Convention:

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

### 3. Data:

- <https://vi.wikipedia.org>
- <http://mecon.vn/cham-be/chuan-can-nang-moi-cua-tre-theo-who/>
- <http://baophunuso.com/che-do-dinh-duong-tieu-chuan-cho-be-so-sinh-den-12-thang-tuoi/>
- <http://comvisinhbibo.vn/cau-hoi-thuong-gap/53-bang-tieu-chuan-chieu-cao-can-nang-cua-tre-moi-nhat>
- <http://ductridn.edu.vn/mamnon/chuyen-muc/cac-lop-mam-non>

### 4. Design:

- <https://design.google.com/>

### 5. Store project data:

- <https://drive.google.com/>
- <https://bitbucket.org>

### 6. Media Server

- [http://info.wowza.com/wowza-streaming-engine?utm\\_campaign=ism\\_ppc\\_goog\\_brand\\_wowzaserver\\_try&utm\\_medium=ppc&utm\\_source=ppc&gclid=CjwKEAjwguu5BRDq8uSKhaKlzDkSJACQ7WJlkznkp--3j6hBP7bxuQvRtyu9llss6j0UNGqY3szGbhoCWZ7w\\_wcB](http://info.wowza.com/wowza-streaming-engine?utm_campaign=ism_ppc_goog_brand_wowzaserver_try&utm_medium=ppc&utm_source=ppc&gclid=CjwKEAjwguu5BRDq8uSKhaKlzDkSJACQ7WJlkznkp--3j6hBP7bxuQvRtyu9llss6j0UNGqY3szGbhoCWZ7w_wcB)