**VIETNAM NATIONAL UNIVERSITY, HANOI**

**UNIVERSITY OF ENGINEERING AND TECHNOLOGY**

[](http://test.uet.vnu.vn/)

**Trần Ngọc Linh**

**BUILDING REPORT MODULE FOR ELEMENTARY SCHOOL MANAGEMENT SYSTEM**

**Major: Computer Science**

**HA NOI - 2016**

**VIETNAM NATIONAL UNIVERSITY, HANOI**

**UNIVERSITY OF ENGINEERING AND TECHNOLOGY**

**Trần Ngọc Linh**

**BUILDING REPORT MODULE FOR ELEMENTARY SCHOOL MANAGEMENT SYSTEM**

**Major: Computer Science**

**Supervisor: Assoc. Prof. Dr. Truong Anh Hoang**

**Co**-**Supervisor: Dr. Dang Duc Hanh**

**HA NOI - 2016**

**AUTHORSHIP**

*“I hereby declare that the work contained in this thesis is of my own and has not been previously submitted for a degree or diploma at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no materials previously published or written by another person except where due reference or acknowledgement is made.”*

Signature:………………………………………………

**SUPERVISOR’S APPROVAL**

*“I hereby approve that the thesis in its current form is ready for committee examination as a requirement for the Bachelor of Computer Science degree at the University of Engineering and Technology.”*

Signature:………………………………………………

**ACKNOWLEDGEMENT**

I would like to acknowledge the following people for their support and assistance in my final thesis.

First of all, I would like to give a special thankfulness to my supervisors, Associate Professor Truong Anh Hoang and Doctor Dang Duc Hanh, Department of Software and Engineering, Faculty of Information Technology, University of Engineering and Technology, for giving me the instructions and explanation of several problems I have faced related to this thesis. I could not complete my thesis without dozens of helpful advices from him in weekly meetings and via emails. They teaches me how to put my best effort to solve problems by myself. Throughout a year I have learnt so much from them, both the academic work and the proper lifestyles as well.

I would like to thank my teammates, Nguyen Hong Son, Tran Van Tuan and Nguyen Thi Thuy for their support as well as collaboration, i worked in a friendly and efficient enviroment.

I greatly appreciate Sao Khue company. Thanks to Sao Khue company, especially Mr. Nguyen, this thesis has the opportunity to be deployed in the actual enviroment. This is one of my main motivation to conduct this application. I choose this moment to acknowledge his contribution gratefully.

Last but not least I want to thank my family and friends for their supportive role on background, despite all the long days and nights I have spent with my studies.

Ha Noi, April 2016

**Tran Ngoc Linh**

**ABSTRACT**

In this thesis, I will introduce an elementary school management system, was named Truong Nha. It is a set of technological solutions, which helps schools connect to students’ families. Therefore, information about the learning process of the students will be promptly updated. Besides, it supports teaching as well as tasks in education to be getting more simple and more effective. There are four sub-modules in Truong Nha system. A module about teaching and evaluating activities was developed by Mr. Tuan. The second module about API, connecting between server and mobile, was developed by Mr. Son. While the third module about the mobile client with Ionic Framework was developed by Ms. Thuy. The last module about building up report module for elementary school management system was developed by me. So, I only focus on analysis, design and other steps to develop report module in the system.

***Keywords:*** Education management solution, management system, report.

**Tóm tắt**

Trong bài báo cáo khóa luận này, tôi sẽ giới thiệu về một hệ thống quản lý trường tiểu học có tên là Trường Nhà . Đây là một giải pháp công nghệ trợ giúp kết nối trường nhà với gia đình học sinh. Bởi vậy, thông tin về quá trình học tập của học sinh được cập nhật thường xuyên và kịp thời. Bên cạnh đó, hệ thống giúp cho việc giảng dạy cũng như các hoạt động khác trong lĩnh vực giáo dục trở nên đơn giản và hiệu quả hơn. Hệ thống Trường Nhà được có 4 mô-đun chính. Module thứ nhất là về hoạt động giảng dạy và đánh giá được phát triển bởi Trần Văn Tuấn. Module thứ hai về việc xây dựng API để kết nối phía server và phía mobile được phát triển bởi Nguyễn Hồng Sơn. Mô-đun thứ ba được Nguyễn Thị Thủy phát triển để xây dựng ứng dụng di động với Ionic Framework. Còn mô-đun cuối về phần xây dựng các báo cáo được tôi phát triển. Vậy nên trong các phần sau của khóa luận thì tôi chỉ tập trung vào việc phân tích, thiết kế và những bước cần thiết khác để xây dựng các báo cáo trong hệ thống.

***Từ khóa:*** giải pháp quản lý giáo dục, hệ thống quản lý, báo cáo.

**TABLE OF CONTENTS**

List of Figured x

ABBREVATIONS xi

Chapter 1 1

INTRODUCTION 1

1.1. Motivation 1

1.2. Related work 2

1.3. Contributions 3

1.4. Thesis overview 4

Chapter 2 6

SYSTEM REQUIREMENTS 6

2.1. Functional requirements 6

2.2. Non-function requirements 7

2.3. Table form requirements 7

Chapter 3 14

TECHNOLOGIES 14

3.1. Ruby on Rails 14

3.2. PostgreSQL 17

Chapter 4 19

SYSTEM DESIGN 19

4.1. Database design 19

4.2. User-Case Model 21

Chapter 5 30

IMPLEMENTATION AND DEPLOYMENT 30

5.1. Implementation 30

5.2. Deployment 32

Chapter 6 34

DISCUSTIONS AND CONCLUSIONS 34

6.1. Discussions 34

6.2. Conclusions 35

6.3. Future Works 35

References 36

List of Figured

[Figure 2‑1-A non feedback form 8](#_Toc449576581)

[Figure 2‑2-A feedback form 8](#_Toc449576582)

[Figure 2‑3-A reviews by teacher form 8](#_Toc449576583)

[Figure 2‑4-Term end review form 10](#_Toc449576584)

[Figure 2‑5-Statistic of education quality in class form 11](#_Toc449576585)

[Figure 2‑6-Statistic of education quality for school form 11](#_Toc449576586)

[Figure 2‑7-Statistic of score in class form 12](#_Toc449576587)

[Figure 2‑8-Statistic of score for school form 12](#_Toc449576588)

[Figure 2‑9- File EMQS form 13](#_Toc449576589)

[Figure 3‑1-The model-view-controller (MVC) architecture 17](#_Toc449576590)

[Figure 4‑1-Database design (part 1) 19](#_Toc449576591)

[Figure 4‑2-Database design (part 2) 20](#_Toc449576592)

[Figure 4‑3-The user-case model in school manager view 21](#_Toc449576593)

[Figure 4‑4-The user-case model in headmaster view 22](#_Toc449576594)

[Figure 4‑5-The user-case model in teacher view 22](#_Toc449576595)

[Figure 5‑1-Export Excel interface 31](#_Toc449576596)

[Figure 5‑2-Code for write Excel file 32](#_Toc449576597)

[Figure 6‑1-Statistical data via Google Analistics 34](#_Toc449576598)

ABBREVATIONS

MVC Model-View-Controller

AJAX Asynchronous JavaScript and XML

ROR Ruby on Rails

SMAS School Manager System

HTML HyperText Markup Language

CSS Cascading Style Sheets

# 

INTRODUCTION

## Motivation

Nowadays, education is getting more and more innovated in order to enhance abilities and qualities of students. Especially, circular no 30 with evaluating elementary students was promulgated in 2014. It offers efficacy in improving the quality of education in elementary schools. However, teachers, headmasters, and parents also face several problems that they have to work more hard. Firstly, for those teachers, in addition to the traditional work they are doing as testing, evaluating and making comments on all of the student’s activities. Every day, they write comments on student’s notebooks. Monthly, they write reviews in a student’s monitoring book of education quality and give the books to their students. Students will bring them to home and give their parents. After the parents sign and comment into the book, it will be returned to the teacher. Secondly, for those headmasters, they spend a lot of time making many reports with various types in the whole semester to know teaching activities situation and education quality of a school. Especially, at the semester end, theses businesses bring the burden of records that make them feel tired and uncomfortable. Finally, for those parents, information about their children learning process is not updated immediately. So they cannot cooperate with the schools and teachers in educating students.

Information technology has been developing rapidly. The application of information technology in the field of management is an inevitable trend and education is no exception to these fields. The application of information technology in education will enhance the quality of education as well as help manage school more efficiently. Currently, there are many technology solutions for education such as the educations technology solutions of IBM. In Viet Nam, Viettel Corporation has developed a system called SMAS. After the investigation, we found that this application has not yet met effectively the needs to solve problems mentioned. From those requirements, our team led by Assoc. Prof. Dr. Truong Anh Hoang develops a software system for elementary school management, called “Truong Nha”. This software not only connects the school with family but also reduces the burden of record for teachers.

Personally, I found that developing an educational software is necessary for industrialization and modernization age. It helps exchanging information between school and family becomes easy and convenient. Especially, building report module supports teachers and headmasters make reports faster and more effective.

## Related work

Nowadays, the development of technology brings many benefits to people’s life as well as in the fields. Especially, in the field of education, some great system is developed to support teachers in teaching activities and managers in managing school such as Education technology solutions (1), SMAS and SMS-Edu (2) and Truong Nha (3), etc. For the details of those systems we will discuss in the next few paragraphs.

For the beginning, I have to start to talk about Education technology solutions of IBM. It is a relatively comprehensive system. It gives teachers and students the technology solutions and services they need to be successful from kindergarten through graduation to employment in the workforce. However, the biggest drawback of this system is not supporting Vietnamese, so it’s not suitable to deploy in Vietnam.

SMAS and SMS-Edu are developing by Viettel Corporation. SMAS, which current version is SMAS 3.0, is built to meet the business of managing schools from kindergarten to high school. SMS-Edu includes SMS-Parent service and SMS-Teacher to support communicating between the school and the parents become convenient and easy. Parents can see the mark, attendance, rewards and other activities of their children after teachers enter marks, attendance, etc. but in the exchange process, sending the message between the school and parents through SMS have to pay for the fee and limited characters. More importantly, in 2014, the evaluation method was changed in elementary schools lead to many functions in the system is not suitable.

TruongNha is a good system, which developed by a group of students of University of Engineering and Technology in 2013. It focuses on the process of information exchanged between teacher and students for secondary schools and high schools. However, this system was not developed for elementary schools and reduces the burden of management in education.

Three of the most common systems have their own pros and cons. Without systems, managing all activities become too hard for the school manager. Similarity, the teacher has to waste a lot of time to write reports for the students.

## Contributions

From the urgency developing a system is more suitable, my team includes: Tran Van Tuan develops functions about teaching and evaluation activities, Nguyen Thi Thuy build a hybrid application for mobile, Nguyen Hong Son have responsibility connecting between the server and mobile client and I develop report module for the system.

The teacher has to make many reports with various type in each semester. For example, they have to write daily reports when their students have special activities in school. They need export some term end evaluations at the end semester. The headmaster has to make some statistic reports to know the quality of teaching activities in schools. This module is an important part of the system to help teachers and headmasters make report conveniently and easily with several clicks.

In this system, my contributions is:

I take part in the design database, which can serve many elementary schools everytime, work well with current tasks and bring the high scalability.

The user interface, which is designed simple and easy to use, bring high user experiments.

I develop export functions having export various of type reports like Microsoft Excel and PDF.

I develop software to ensure security mechanisms and divide projects into small packages to easily maintenance and upgrades.

I fix some error about export reports when a user reports or contribute in the forum.

I research about Ruby language, Ruby on Rails Framework, HTML, JavaSript, AJAX and GEM

## Thesis overview

The rest of this thesis is organized as follows.

**Chapter 1:** Introduction

There is some general information about our research and related work.

**Chapter 2:** System requirements

This chapter shows the requirements about function and non-function in the system

**Chapter 3:** Technology

This chapter introduces the technologies was used in the system and how to it work.

**Chapter 4:** System design

I will analyze and design user requirements in the system.

**Chapter 5:** Implementation

I will provide the necessary information about method I have used to build report module for TruongNha

**Chapter 6:** Deployment and testing

I will clearly specify how TruongNha was deployed and its performance during the actual deployment.

**Chapter 7:** Conclusion

This is our conclusion after build and deploy this system. This chapter will show what I do to contribute as well as present difficulties and our experiences in the process of deploying an actual product with the relatively large number of users.

# 

SYSTEM REQUIREMENTS

After analyzing requirements from users and other common applications, we divide servers site to two modules. Mr. Tuan is the responsibility of building teaching activity module. I building report module for the elementary school management system. Therefore, I will focus on functional requirements and non-functional requirements which related to building report module.

## Functional requirements

The system provides the making report functions to each specific user.

***For the school manager***

* View and export PDF file of all students’ reviews in a specific class.
* View and export PDF file of all students’ reviews of a specific teacher.
* View and export PDF file of all students’ term end reviews of a specific teacher.
* Export EMQS report with Microsoft Excel file.

***For the headmaster***

* + View and export PDF file of all students’ reviews in a class.
  + View and export PDF file of all students’ reviews of a specific teacher.
  + View and export PDF file of all students’ term end reviews of a specific teacher.
  + View and export Excel file about statistic of education quality by grade
  + View and export Excel file about statistic of score by grade

***For the teacher***

* + View and export PDF file of all students’ reviews in a class which they teach
  + View and export PDF file of all students’ reviews which they write
  + View and export PDF file of all students’ term end reviews of a specific teacher.
  + View and export Excel file about statistic of education quality by class
  + View and export Excel file about statistic of score by class

## Non-function requirements

* The system have to suitable with various size screen.
* The system should be secure.
* The time respond from the server is less than 3 second in normal condition.

## Table form requirements

### Form of reviews in class

The purpose of the form is showing all review of each student in a specific class. There is two type of the form: non-feedback form and feedback form. The non-feedback form includes student’s name, student’s parents, class name, teacher’s name and list review. Feedback form similar non-feedback, however, adding blank lines to parents can write feedback. Each form is printed on a different page.

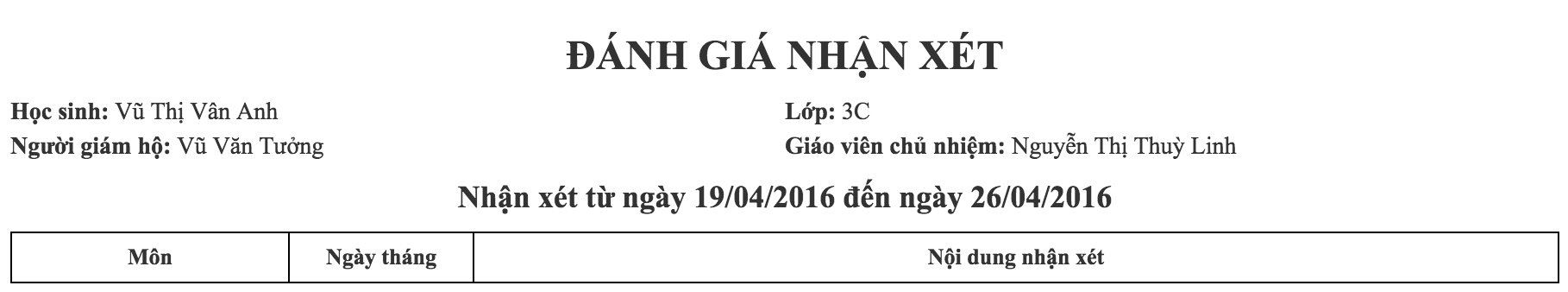


Figure 2‑1-A non feedback form

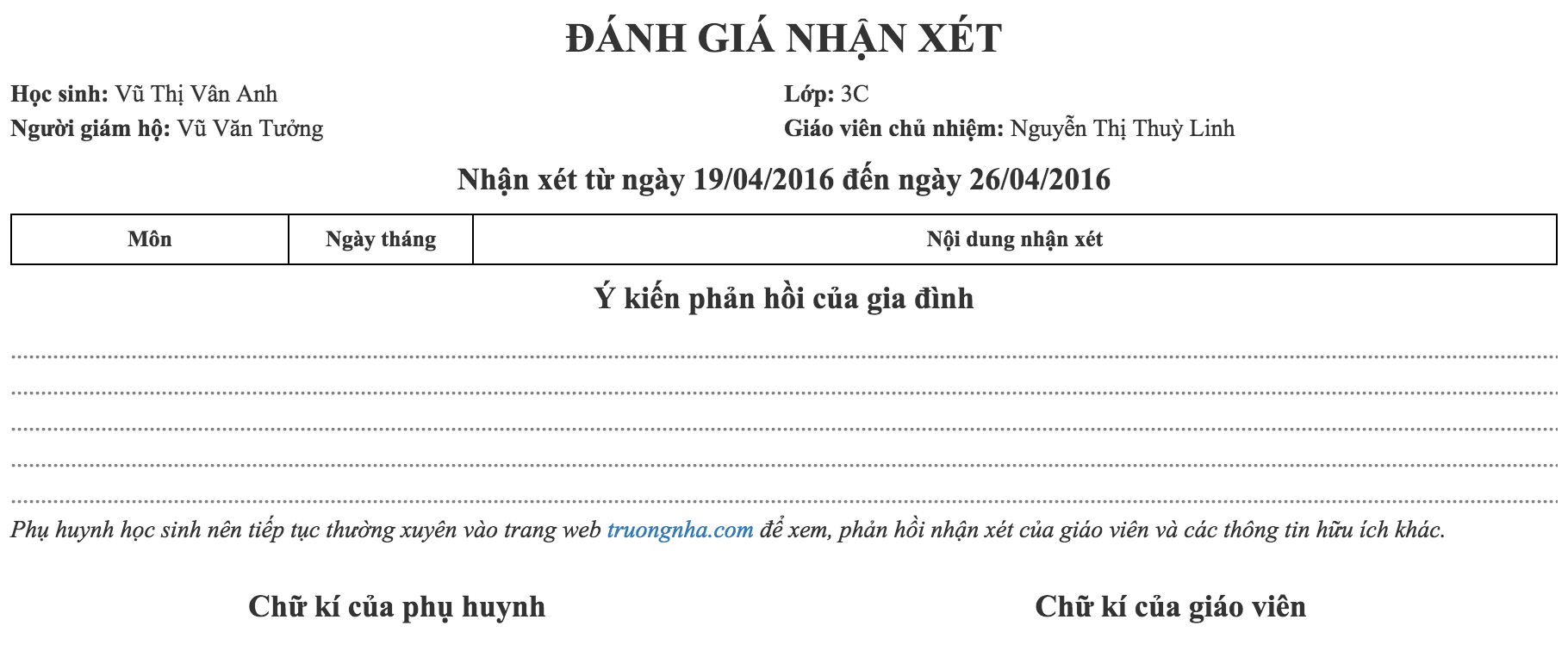


Figure 2‑2-A feedback form

### Form of reviews by teacher

The purpose of the form is showing all reviews which teacher writes. The form includes teacher’s name and list student’s name with reviews

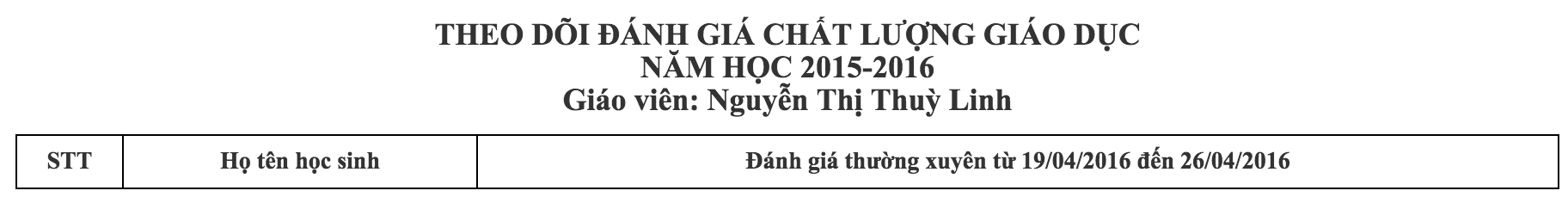


Figure 2‑3-A reviews by teacher form

### Form of term end reviews

This form shows student’s result after semester end. It includes student’s name, student’s parents, class name, teacher’s name, etc. the figure below show details the form

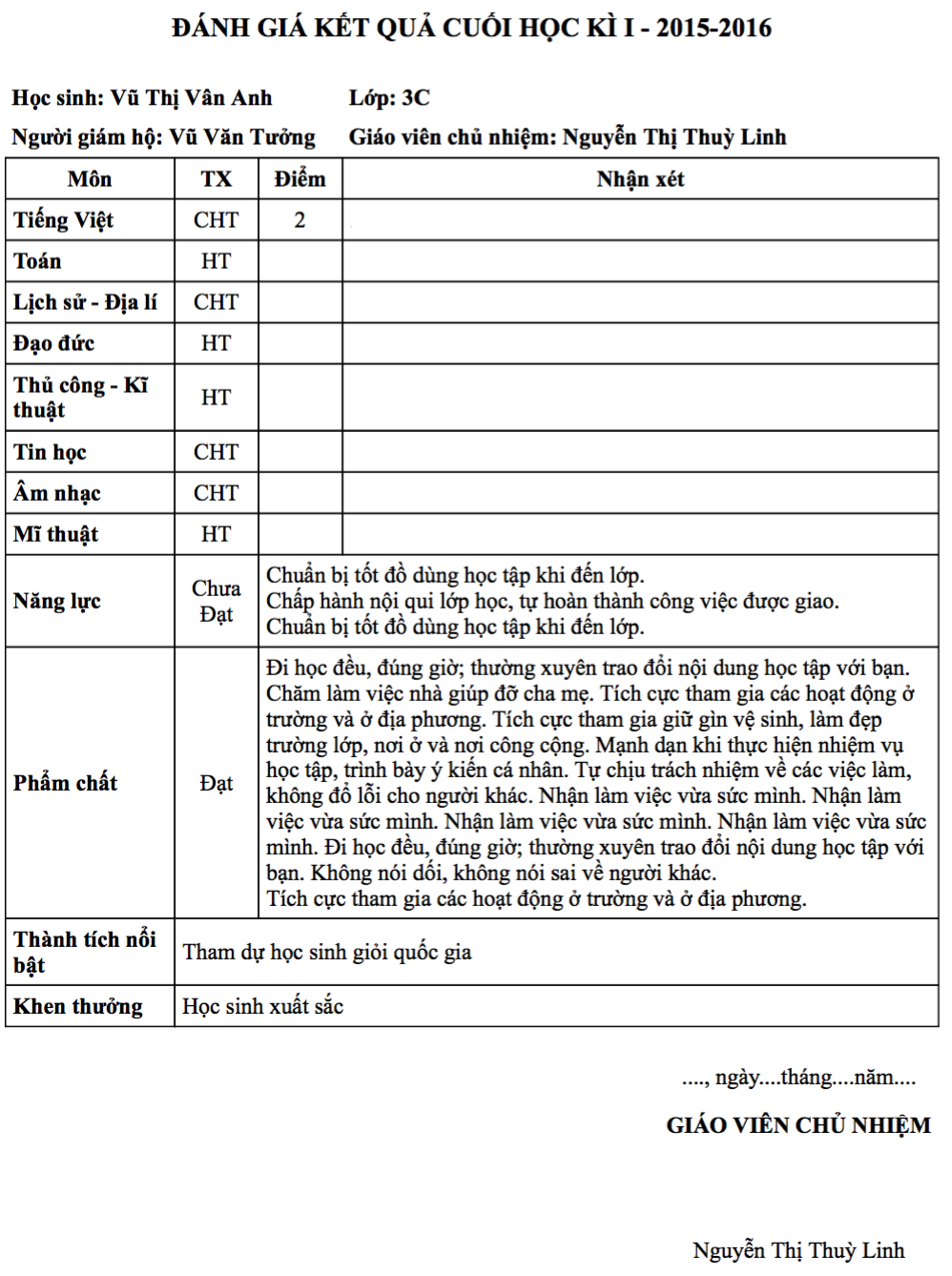


Figure 2‑4-Term end review form

### Form of statistic of education quality

There are two forms, ones for the teacher, another for the headmaster.

For the teacher, only the headteacher can view statistic of education of class which they manager. The form shows 13 main subjects with a number of students done or not done as the figure below show.

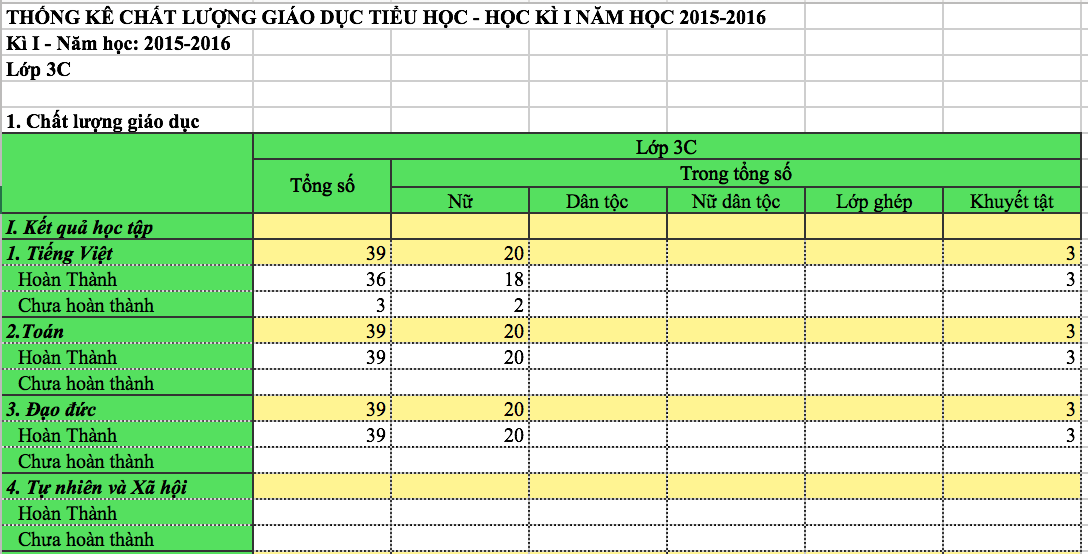


Figure 2‑5-Statistic of education quality in class form

For the headmaster, he/she can view static of education of all grades in the school. Contents of this table are similar contents of the table above.

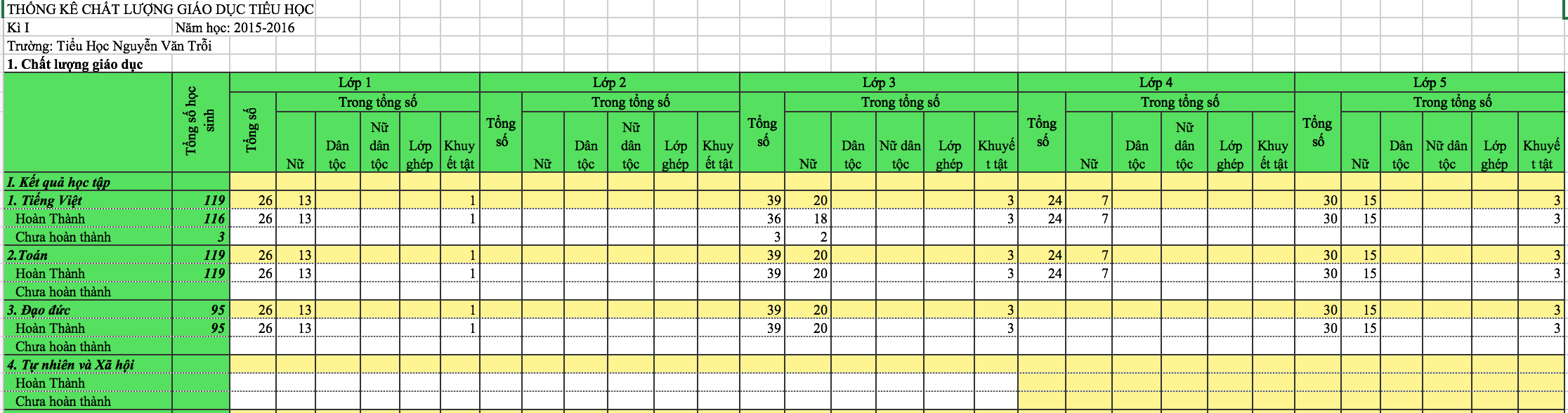


Figure 2‑6-Statistic of education quality for school form

### Form of statistic of score

There are two forms, ones for the teacher, another for the headmaster.

For the teacher, only the headteacher can view statistic of education of class which they manager. The form shows 7 subjects with 6 level score: 10, 9, 8, 7, 6, 5 and below 5 for each subject as the figure below show.

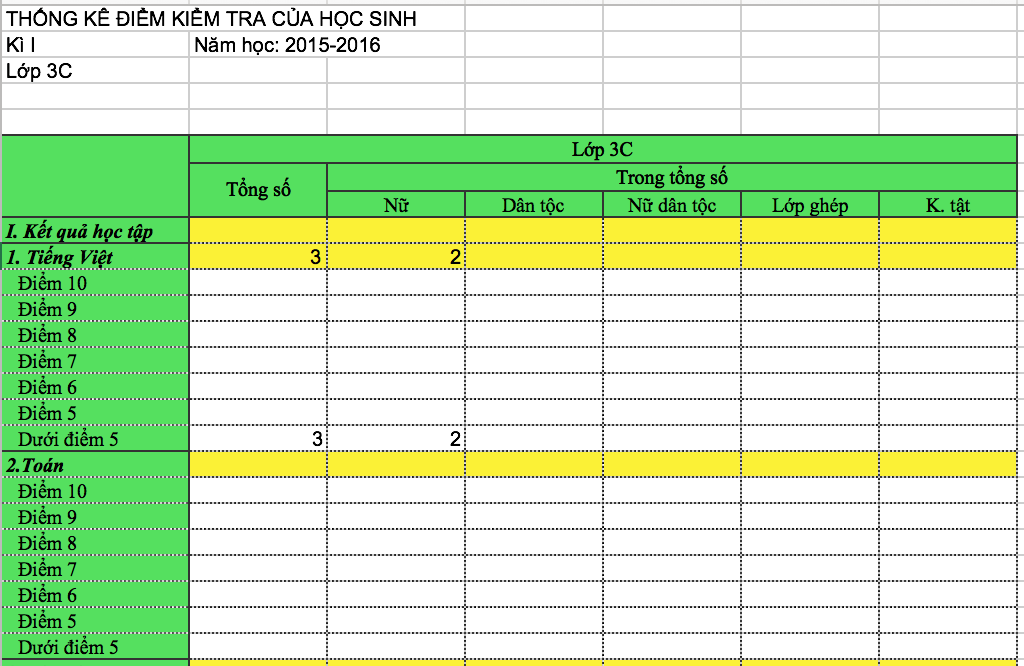


Figure 2‑7-Statistic of score in class form

For the headmaster, he/she can view static of score of all grades in the school. Contents of this table are similar contents of the table above.

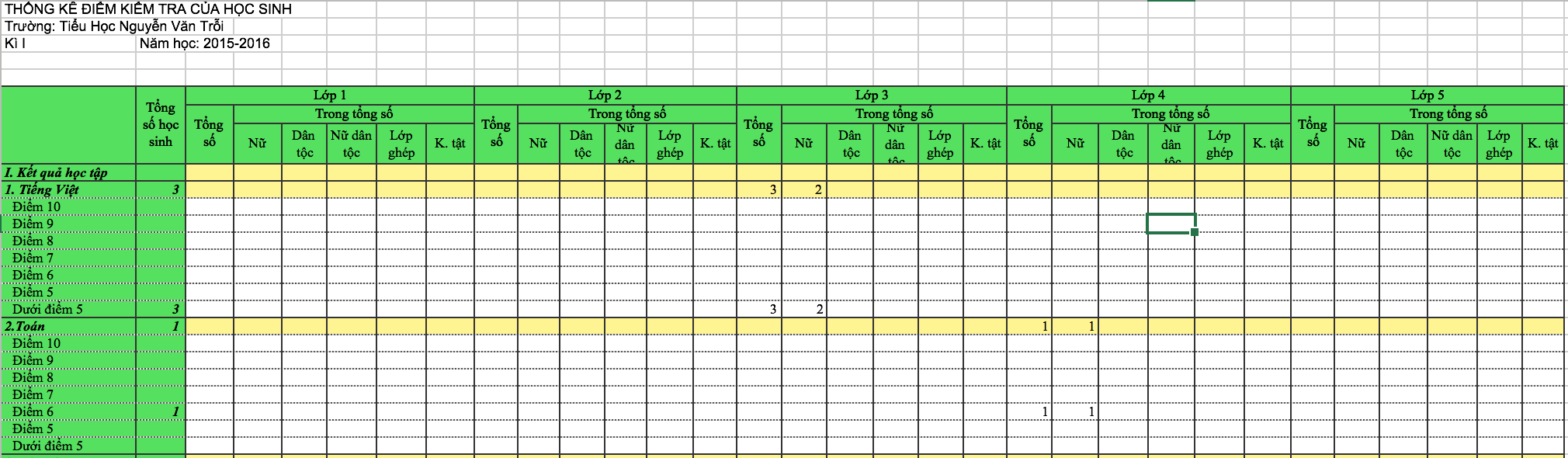


Figure 2‑8-Statistic of score for school form

### Form of EMQS

An EMQS file is exported include 3 sheets: 2 sheets like statistic of education quality for school form and statistic of the score for school form, 1 sheets like the figure below.



Figure 2‑9- File EMQS form

# 

TECHNOLOGIES

After we researched all of the requirements, we found that Ruby on Rails (ROR)[1] is suitable because it is easy to learn and develop fast with the strong library. I will introduce details in the following sections

## Ruby on Rails

### Introduction

Rails is a web application development framework written in the Ruby[2] language make programming web applications easier by making assumptions about what needs to get started. It allows programmers to write less code while accomplishing more languages and frameworks. Experienced Rails developers also report that it make development more fun.

Rails is optinionated software. It makes the assumption that there is the “best” way to do things, and it’s designed to encourage that way – and in some cases to discourage alternatives. If programmers learn “The Rails Way” programmers will probably discover a tremendous increase in productivity.

The Rails philosophy include two major guiding principles:

**Don’t Repeat Yourself:** DRY is a principle of software development which states that “Every picec of knowledge must have a single, unambigous, authoritative representation withing a system.” By not writing the same information over and over again, out code is more maintainable, more extensible, and less buggy.

**Convention Over Configuration:** Rails has opinions about the best way to do many things in a web application, and defaults to this set of conventions, rather than require that you specify every minutiae through endless configuration files.

### Gem

After creating a new Rails application, the next step is to use Bundler to install and include the gems needed by the app. Gems, are libraries, are declared in Gemfile in projects as a picture below describes. Bundler is run automatically (via bundle install) by the rails command to load and install gems to projects. In my thesis, I focus on gem “spreadsheet”. It helps to read and write Excel file with easy way. After is some example using gem spreadsheet to show functions.

Opening a workbook,

book = Spreadsheet.open '/path/to/an/excel-file.xls'

Accessing a worksheet in a work book by index or name,

sheet = book.worksheet ‘0’

sheet = book.worksheet ‘NameSheet’

Accessing rows directly by their index,

row = sheet.row(3)

Filling a string to a cell,

row.push ‘content cell

### Model-View-Controller

Rails is a web-application framework that includes everything needed to create database-backed web applications according to the Model-View-Controller (MVC)[3] pattern.

MVC divides an application into three layers, each with a specific responsibility.

**View:** The View layer is composed of “templates” that are responsible for providing appropriate representations of your application’s resources. Templates can come in a variety of formats, but most view templates are HTML with embedded Ruby code (.erb files).

**Model:** The Model layer represents application’s domain model (such as Account, Product, Person, etc.) and encapsulates the business logic that is specific to the application. In Rails, database-backed model classes are derived from ActiveRecord::Base. Active Record allows the programmer to present the data from database rows as objects and embellish these data objects with business logic methods. Although most Rails models are backed by a database, models can also be ordinary Ruby classes, that implement a set of interfaces as provide by the ActiveModel module.

**Controller:** The Controller layer is responsible for handling incoming HTTP request and providing a suitable response. Usually, this means returning HTML, but Rails controllers can also generate XML, JSON, PDFs, mobile-specific views, and more. Controllers manipulate models and render view templates in order to generate the appropriate HTTP response

In Rails applications, the standard Rails application structure has an application directory called app/ with three subdirectories: models, views, controllers. This is a hint that Rails follows the model-view-controller (MVC) architectural pattern, which enforces a separation between “domain logic” (also called “business logic”) from the input and presentation logic associated with a graphical user interface (GUI)[4]. In the case of web applications, the “domain logic” typically consists of data model for things like user, articles, and products, and the GUI is just a web page in a web browser

When interacting with a Rails application, a browser sends a request, which is received by a web server and passed on to a Rails controller, which is in charge of what to do next. In some cases, the controller will immediately render a view, which is a template that gets converted to HTML and sent back to the browser. More commonly for dynamic sites, the controller interacts with a model, which is a Ruby object that represents an element of the site (such as a user) and is in charge of communication with the database. After invoking the model, the controller then renders the view and returns the complete web page to the browser as HTML

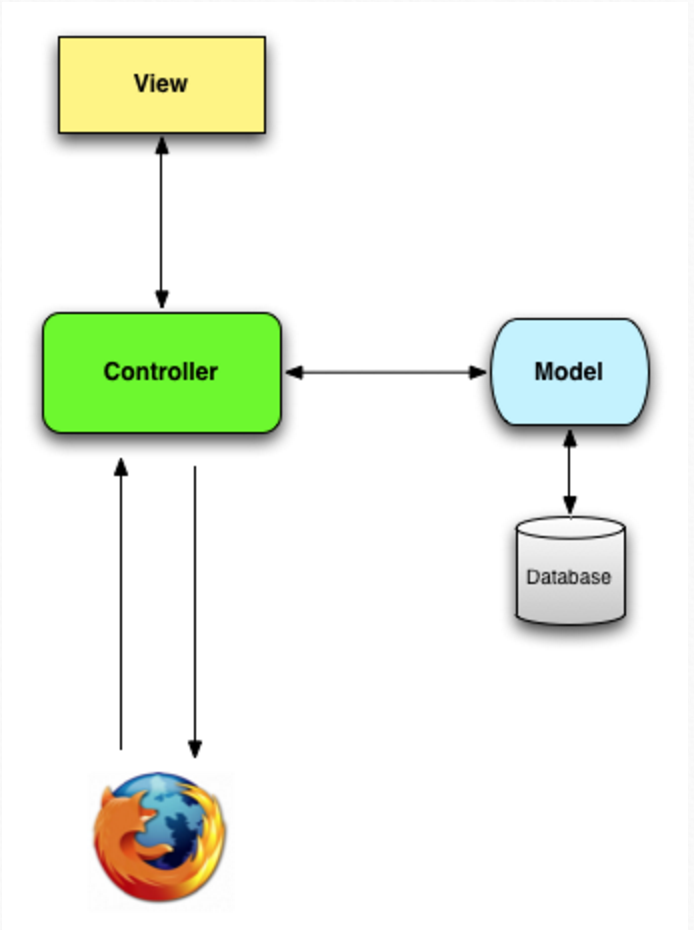


Figure 3‑1-The model-view-controller (MVC) architecture

## PostgreSQL

### Introduction

PostgreSQL is a powerful, open source object-relational database system. It has more than 15 years of active development and a proven architecture that has earned it a strong reputation for reliability, data integrity, and correctness. It runs on all major operating systems, including Linux, UNIX, and Windows. It is ACID complaint, has full support for foreign keys, joins, views, triggers, and stored procedures (in multiple languages). It includes most SQL:2008 data types, including INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, and TIMESTAMP. It also supports storage of binary large objects, including pictures, sounds, or video. It has native programming interfaces for C/C++, Java, .NET, Ruby, etc.

### Advantages of PostgreSQL

* An open-source SQL standard compliant RDBMS:

PostgreSQL is open-source and free, yet a very powerful relational database management system.

* Strong third-party support:

Regardless of the extremely advanced features, PostgreSQL is adorned with many great and open-source third-party tools for designing, managing and using the management system

* Extensible:

It is possible to extend PostgreSQL programmatically with stored procedures, like an advanced RDBMS should be.

* Objective:

PostgreSQL is not just a relational database management system but an objective one – with support for nesting, and more

# 

SYSTEM DESIGN

## Database design

Database designing is one of the most important and difficult parts in developing the system. we optimize database the best we do because we have to meet the system requirements and save a larger number of users, reviews, etc. Sometimes, we need rebuild database when the old design no longer meets new requirements. The figure below show the database of the current system. It is a stable version and meet all current requirements.

Database includes tables as showed below

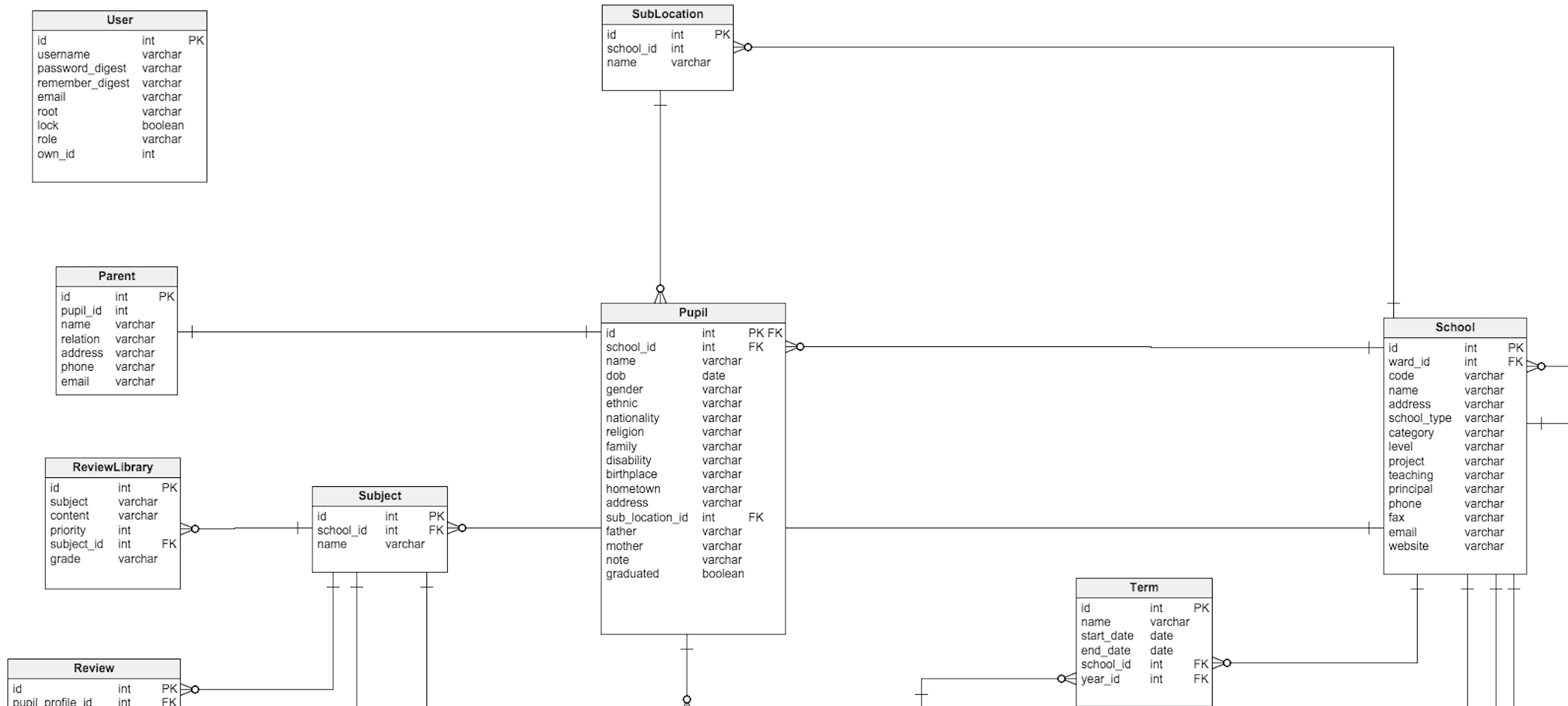


Figure 4‑1-Database design (part 1)

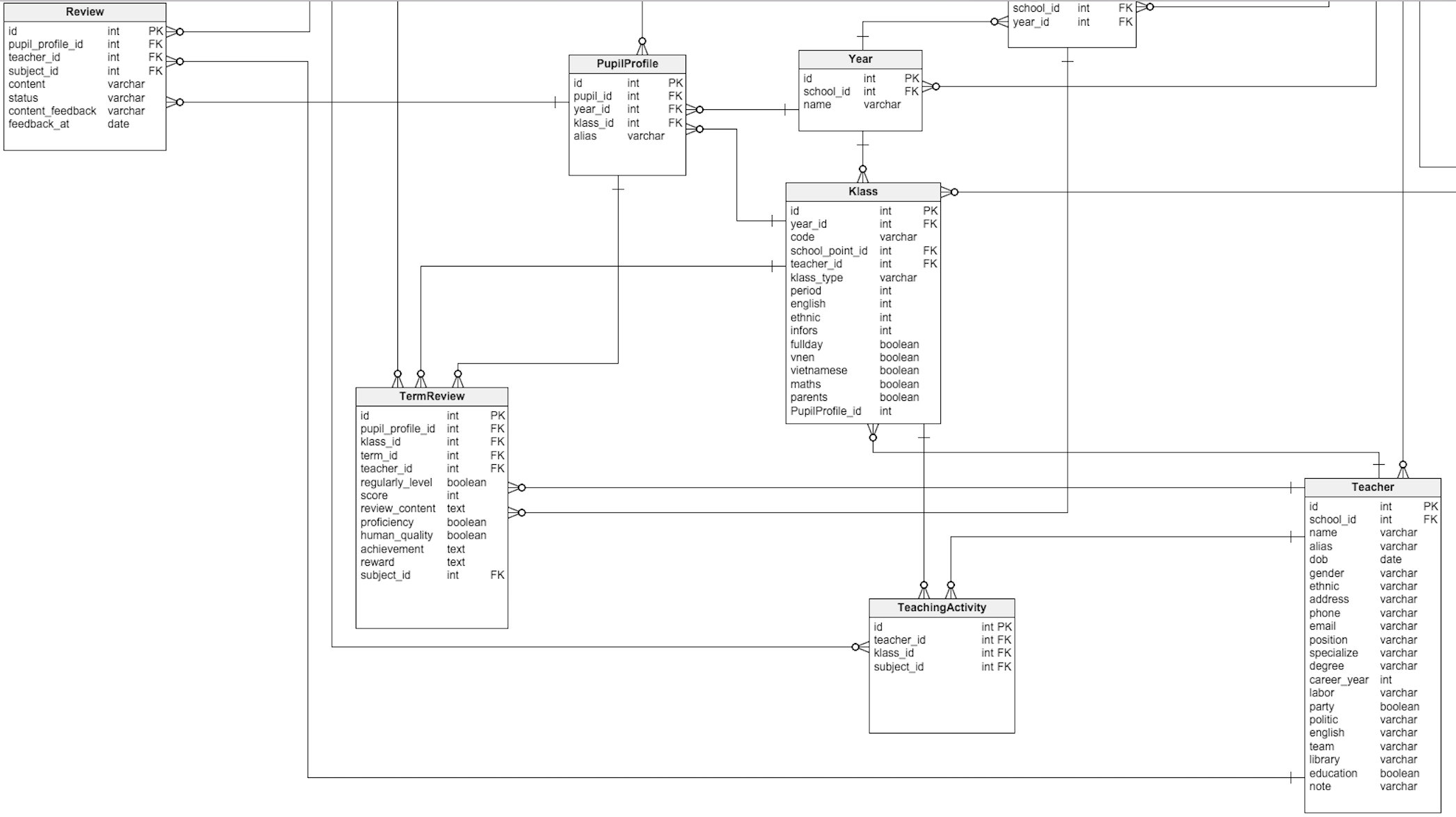


Figure 4‑2-Database design (part 2)

**User:** Is an account in the system. The system bases on role attribute in the table to know what kind of account. It can be SchoolManager, Teacher, Parent, Pupil.

**Pupil:** Who is learn in elementary schools.

**PupilProfile:** Provide information about what class pupil in.

**Parent:** Is one kind of user. Who follows the process of their children’s studying.

**Teacher:** Is one kind of user. Who teaches, makes reviews and reports.

**SchoolManager:** Is one kind of user. Who manages the school.

**Subject:** Save information about subjects in school.

**Klass:** Save information about classes in a school.

**School:** Save information about schools in the system.

**Term:** Save information about the semester in a school.

**Year:** save information about years in school.

**Reviews:** Save all reviews which teacher writes in the system

**TermReview:** Evaluation and score will save in this table.

**TeachingActivity:** save information about what subject and class that teacher teaches.

**ReviewLibrary:** save the review form of a teacher. It makes the teacher feel easy and convenient in writing the review.

## User-Case Model

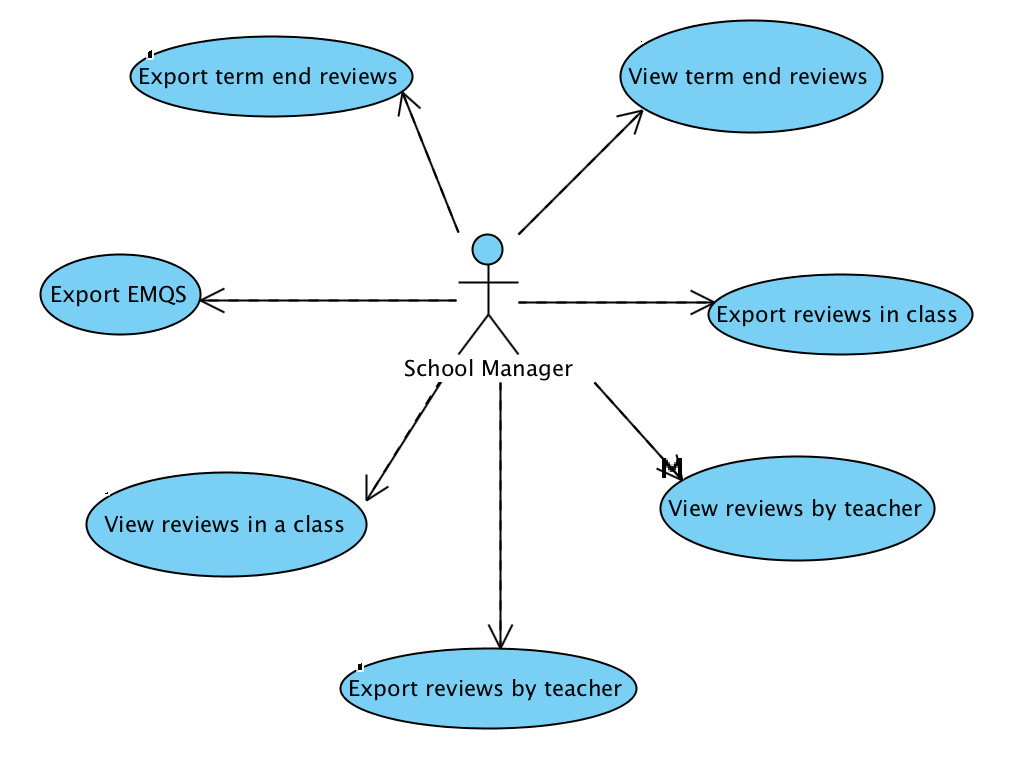


Figure 4‑3-The user-case model in school manager view

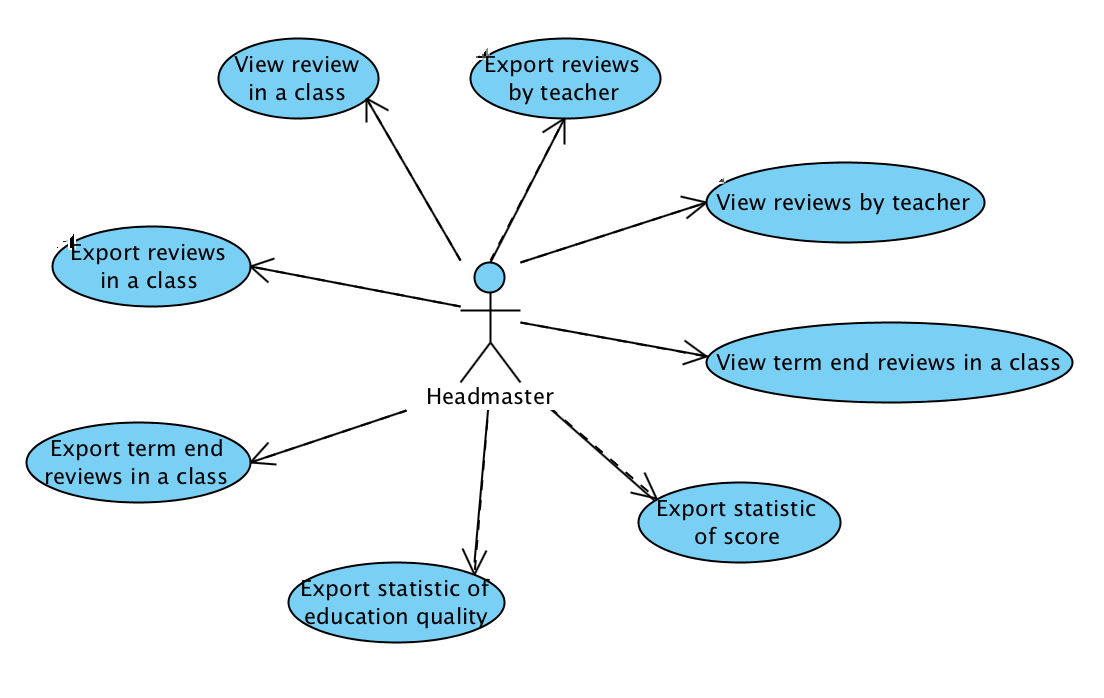


Figure 4‑4-The user-case model in headmaster view

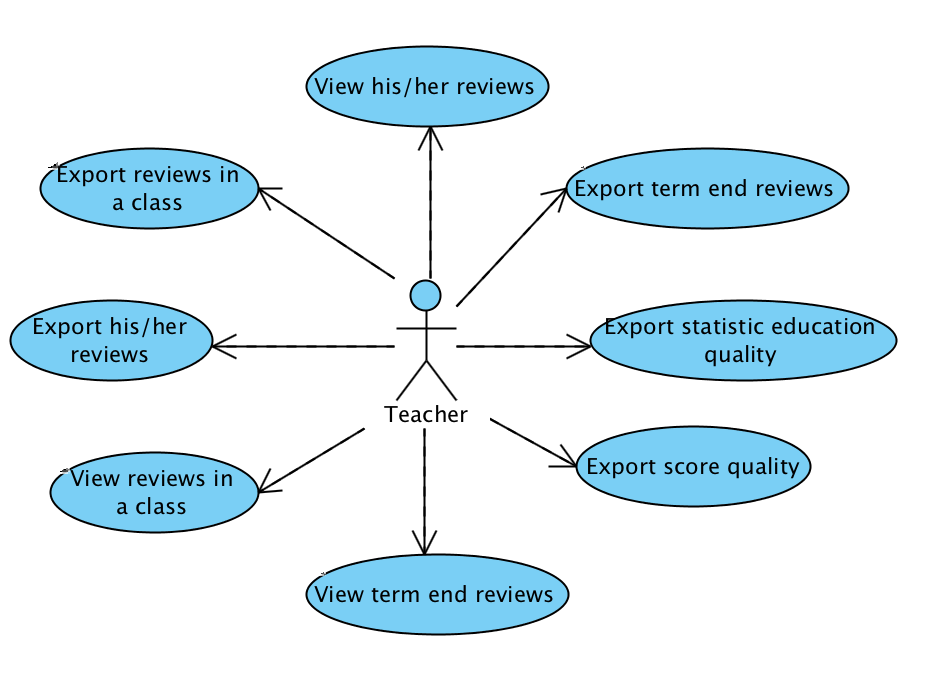


Figure 4‑5-The user-case model in teacher view

### View reviews by classes

**Brief Description**

This use case allows school managers and headmasters see all reviews of students in a specific class in school. It also allows teachers view all reviews of students in a specific class which they are teaching.

**Flow of events**

Basic Flow

This use case starts when an actor wants to see reviews of students in a class

1. The user will choose “In nhận xét lớp
2. Select a period of time which reviews are written with 2 boxes named “Từ ngày” and “Đến ngày”.
3. Choose a specific class that school manager or headmaster wants to see reviews of.
4. Select which student that the actor want to see review of
5. Click “Xem trước” button to show reviews of students

Alternative Flows

If the actor wants to see other reviews of other students in other class, he/she can back to the step 2 in the Basic Flow again and again

**Special Requirements**

If the actor is school manager or headmaster, the system will display all teachers’ name in school in the select box.

**Pre-Conditions**

The system is in the login state and the actor must be school manager or headmaster or teacher

**Post-Conditions**

None.

**Extension Points**

None.

### Export review by classes

**Brief Description**

This use case allows school managers and headmasters export all reviews of students in a specific class in school. It also allows teachers export all reviews of students in a specific class which they are teaching.

**Flow of events**

Basic Flow

This use case starts when an actor wants to see reviews of students in a class.

1. The user will choose “In nhận xét lớp”.
2. Select a period of time which reviews are written with 2 boxes named “Từ ngày” and “Đến ngày”.
3. Choose a specific class that school manager or headmaster wants to see the review of.
4. Select which student that the actor want to see review of.
5. Click “Xem trước” button to show reviews of students.
6. Click “In nhận xét” button to export reviews.

Alternative Flows

If the actor wants to see other reviews of other students in other class, he/she can back to the step 2 in the Basic Flow again and again

If the actor wants to add feedback blank lines into report form to parents can write into, he/she can check to “Có ý kiến phản hồi” checkbox.

**Special Requirements**

If the actor is school manager or headmaster, the system will display all teachers’ name in school in the select box.

**Pre-Conditions**

The system is in the login state and the actor must be school manager or headmaster or teacher.

**Post-Conditions**

None.

**Extension Points**

None.

### View result of evaluation at the end semester

**Brief Description**

Term end review is created by the teacher at the end of each semester after students finish all subjects test. It is different reviews day by day is mentioned above.

This use case allows school managers and headmasters view all term end reviews of students in a specific class in school. It also allows teachers view all reviews of students in a specific class which they are teaching.

**Flow of events**

Basic Flow

This use case starts when an actor wants to see reviews of students in a class.

1. The user will choose “In nhận xét cuối kì”.
2. Select a specific semester with the select box, named “Học kì” .
3. Choose a specific class that school manager or headmaster want to see the reviews of.
4. Select which student that the actor want to see review of.
5. Click “Xem trước” button to show reviews of students.

Alternative Flows

If the actor wants to see other reviews of other students in other class, he/she can back to the step 2 in the Basic Flow again and again.

If the actor is school manager or headmaster, the system will display all teachers’ name in school in the select box.

**Pre-Conditions**

The system is in the login state and the actor must be school manager or headmaster or teacher.

**Post-Conditions**

None.

**Extension Points**

None.

### Export result of evaluation at the end semester

**Brief Description**

This use case allows school managers and headmasters export all term end reviews of students in a specific class in school. It also allows teachers export all reviews of students in a specific class which they are teaching.

**Flow of events**

Basic Flow

This use case starts when an actor wants to see reviews of students in a class

1. The user will choose “In nhận xét cuối kì”.
2. Select a specific semester with the select box, named “Học kì”.
3. Choose a specific class that school manager or headmaster want to see reviews of.
4. Select which student that the actor wants to see the review of.
5. Click “Xem trước” button to show reviews of students.
6. Click “In nhận xét” button to export reviews.

Alternative Flows

None.

**Special Requirements**

If the actor is school manager or headmaster, the system will display all teachers’ name in school in the select box.

**Pre-Conditions**

The system is in the login state and the actor must be school manager or headmaster or teacher

**Post-Conditions**

None.

**Extension Points**

None.

### View statistics of education quality

**Brief Description**

This use case allows headmasters and teacher view statistic of education quality

**Flow of events**

Basic Flow

This use case starts when an actor wants to see statistic form of education quality

1. The user chooses “Đánh giá cuối kì” tab.
2. The user chooses “Chất lượng giáo dục” tab.
3. The user selects a specific semester.
4. The user select a specific class.

Alternative Flows

If user wants to see another statistic forms of education quality of other semester and other class. They can change semester select box or class select box and back to the step 3 in the Basic Flow

**Special Requirements**

If the actor is headmaster, the system will show statistic form of education quality following by grade. If the actor is a teacher, the system will show the statistic form of education quality following by class which they are teaching.

**Pre-Conditions**

The system is in the login state and the actor must be headmaster or teacher.

**Post-Conditions**

None.

**Extension Points**

None.

### Export statistics of education quality

**Brief Description**

This use case allows headmasters and teachers export statistic of education quality with Excel format

**Flow of events**

Basic Flow

This use case starts when an actor wants to see statistic form of education quality

1. The user chooses “Đánh giá cuối kì” tab.
2. The user chooses “Chất lượng giáo dục” tab.
3. The user selects a specific semester.
4. The user selects a specific class.
5. The user clicks “Xuất Excel” select box and chooses .xls or .xlsx file format.

Alternative Flows

If user wants to see another statistic forms of education quality of other semester and other class. They can change semester select box or class select box and back to the step 3 in the Basic Flow

**Special Requirements**

If the actor is headmaster, the system will show the statistic form of education quality following by grade. If the actor is a teacher, the system will show the statistic form of education quality following by class which they are teaching.

**Pre-Conditions**

The system is in the login state and the actor must be headmaster or teacher

**Post-Conditions**

None.

# 

IMPLEMENTATION AND DEPLOYMENT

In this chapter, I will show the way build report module. However, there are many reports in the system so I just choose use case “**Export statistic of education quality**”.

## Implementation

ROR project is build base on the characteristic of MVC architecture so most of the logic coding is implement in the controller. Beside, ROR projects include helps folder to help logic coding in controller folder. In this part, I will show and explant details how to Rails work and how to build export report Excel file.

### Ideas

We need an array variable with 2 dimentions like arr[a][b] = x. Inside, arr is name array; a is index of subjects; b is values: totals, results\_1, results\_2; and x is value array of content table.

For example, as the figure below show. “Tiếng Việt” has index 1 so we have

arr[1][total] = [39, 20, 0, 0, 0, 3]

arr[1][result\_1] = [36, 18, 0, 0, 0, 3]

arr[1][result\_0] = [3, 2, 0, 0, 0, 0]

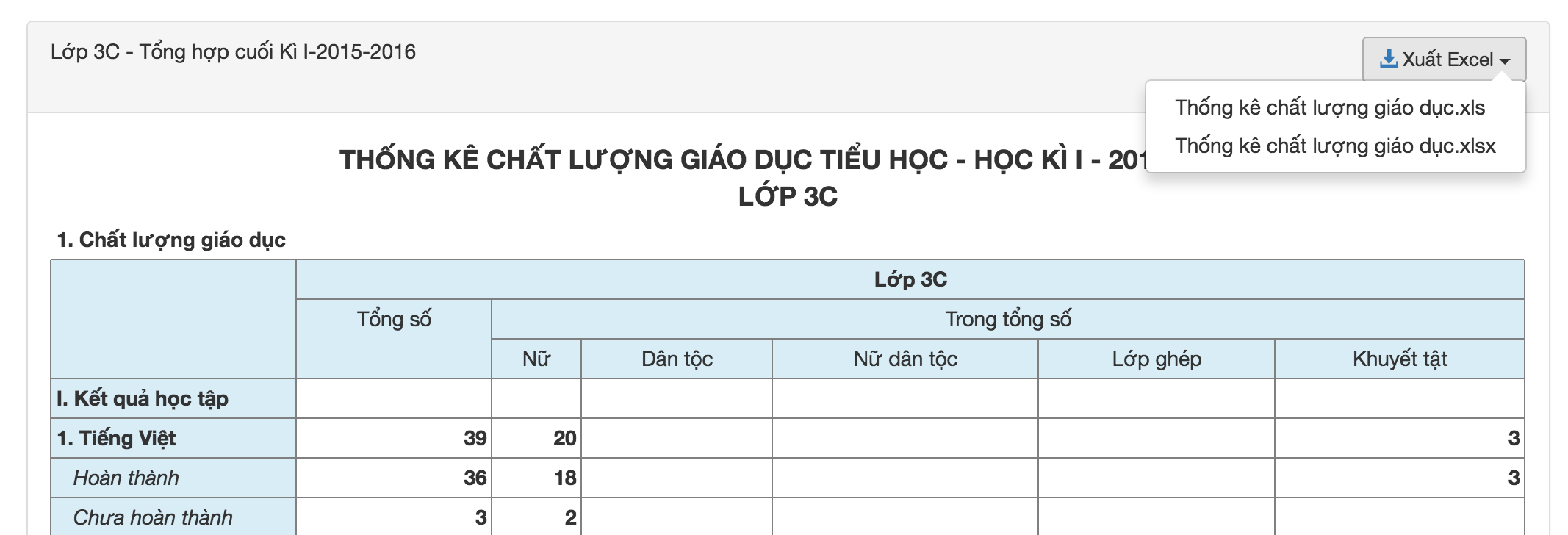


Figure 5‑1-Export Excel interface

### Implements

When a user clicks button”Xuất Excel”, 2 chooses will show as figure 5-1

Each chooses “Thống kê chất lượng giáo dục.xls” and “Thống kê chất lượng giáo dục.xlsx” corresponding with URLs. ROR will go to **routes.rb** file to find controller’s functions corresponding with URLs which calls

post "end\_term\_statistics/class\_education\_quality/download/:term\_id/:klass\_id/:format"

=>"end\_term\_statistics#download\_class\_education\_quality"

ROR will implement **download\_class\_education\_quality** function in **EndTermStatisticsController** class. In **download\_class\_education\_quality** function, I will do 2 tasks: one is creating a variable contain all value of table, another is creating Excel file contain value and send to user download.

1. Creating a variable contain all value of table

I create @class\_education\_quality variable. For each subject, I will find the number of students not done and done. After that I will plus the number of students not done and the number of students done. The result after calculating will assign to @class\_education\_quality[‘total’].

I will query database to find number of student not done and done.

pupil\_profiles\_ids = TermReview.total\_by\_result(result, klasses\_ids, subject\_id, term\_id).pluck(:pupil\_profile\_id)

pupil\_profiles = PupilProfile.where("id IN (?)", pupil\_profiles\_ids)

result = 1 if I want to query student done

result = 1 if I want to query student not done

1. Creating Excel file contain value and sent to user download.

From @class\_education\_quality variable I find, I will pass it on write\_class\_education\_quality function to write content to Excel file.

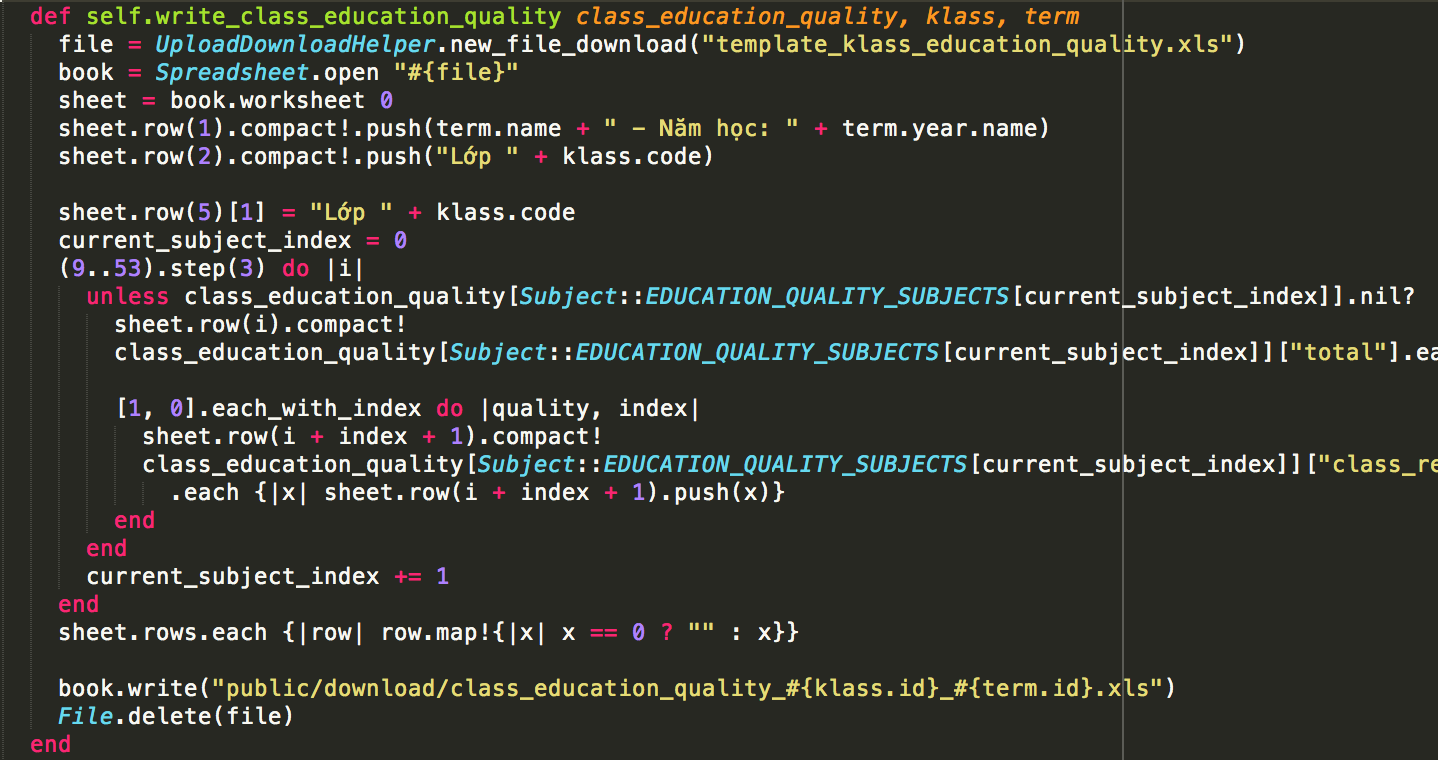


Figure 5‑2-Code for write Excel file

After filling Excel form, server will send Excel file to user.

## Deployment

After a long time building the system, we deployed the system at address <http://truongnha.com/>. The current system has essential functions to serve the headmaster, teacher, and parents. There are positive feedbacks in our system as well as the contributions needed to develop the system become better in the future.

Now, Truong Nha is being deployed by Sao Khue Company. Until now, Truong Nha is during trail time. It is used in 180 primary schools. Most schools are in Nam Dinh, Dong Nai, Ha Noi, Dac Lac, Hai Duong province

The figure below shows positive results after deployment time.

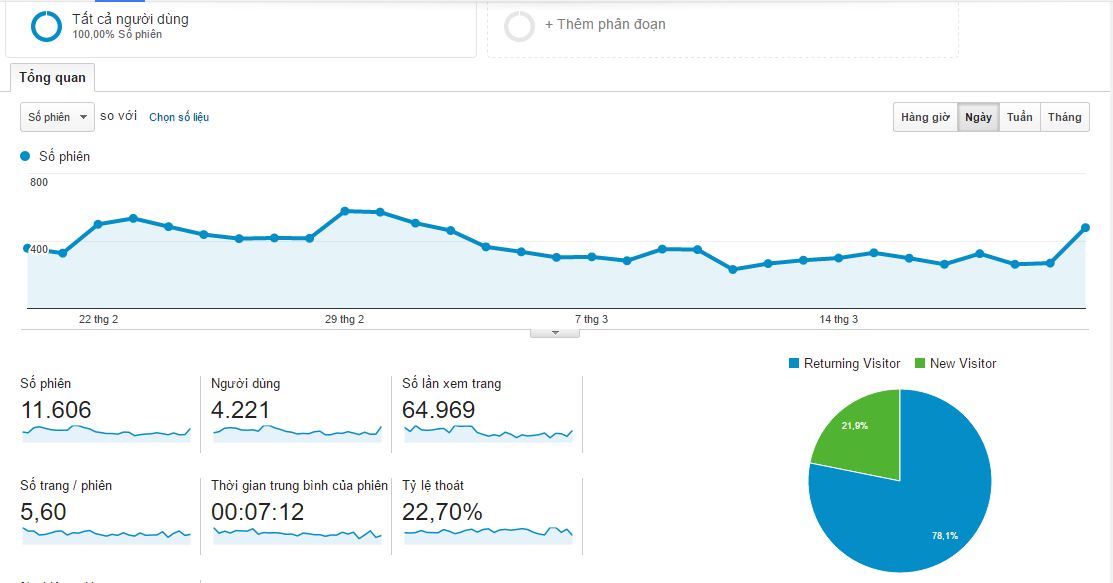


Figure 5‑3-Statistical data via Google Analistics

The number of user account in Truong Nha is more than 50000 accounts, with 1000 regular users,

The number of reviews of today is more than 100000 reviews and 5000 feedbacks from the parent.

After making a survey about the satisfaction of users, we received positive feedback on the system when most of the results are satisfy. Besides, some teachers recommend advice for improving weak points of the system.

During deploy time, we also find some difficulties:

Although web version brings many benefits for the teacher, some old teachers prefer to write reviews rather than use our system.

The number of users, reviews increase day by day. So the performance and security are our challenges.

# 

DISCUSTIONS AND CONCLUSIONS

## Discussions

Developing Truong Nha is an interesting experience for me. I not only consolidating knowledge at school but also learn new knowledge about technology and the way to deploy an actual product.

The knowledge which I consolidating:

* It relates to Software Engineering, Object-Oriented Programming, Object-Oriented Analysis Design, Service Oriented Architecture, Database subjects.

Technology knowledge which I have learned:

* Learning Ruby language, Ruby on Rails framework, HTML, CSS, Ajax, Jquery.
* Using Git to manage software versions

Besides, I have learned something not teaching in school:

* How to communicate to customers effectively and understand what they need to solve difficulties they face when using the system. All this work is done by our forum or direct phone.
* Because the system is deployed in many schools, so we need fix problems of the system as fast as we can.
* Database is very important, so we have to backup database work daily to protect the risks

## Conclusions

This thesis provides an overview of Truong Nha as well as shows the way building report module for elementary school manager system. In fact, many reports are generated in the whole semester lead to make headmaster and teacher feel inconvenient and uncomfortable. Building reports module in the system is necessary to solve all problems which the teacher faces.

## Future Works

Truong Nha is not only the application which I build for my thesis but also an important part to contribute to national education. In the future, we want to make Truong Nha become stronger with many new functions which the teacher need. There are some greate ideas to develop Truong Nha in the near future:

* Enter reviews by voice processing.
* Building a chatting window for user exchange information easily.
* Extending this system to use for secondary and high schools

References

1. http://rubyonrails.org/
2. https://www.ruby-lang.org/en/
3. M. Hartl, "RUBY ON RAILS TUTORIAL (3RD ED.)," 2014, pp. 30-32.
4. S. Levy, "Graphical user interface (GUI)," 18 12 2014.
5. http://www.postgresql.org/docs/9.5/static/release-9-5.html Online: