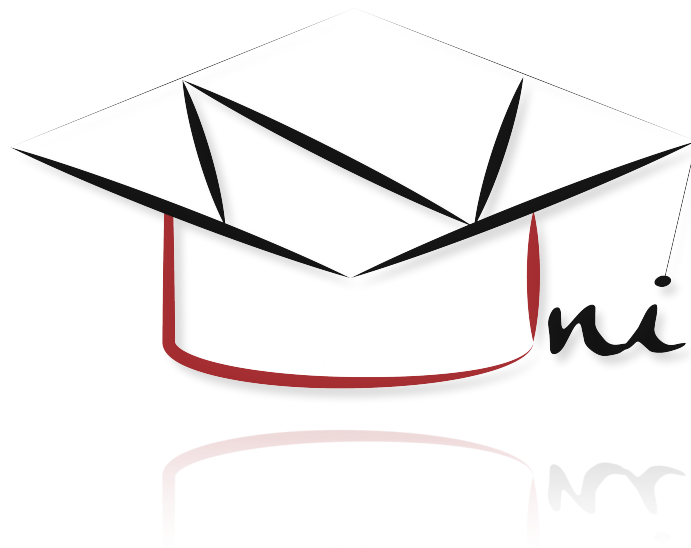


# VnUni

*Object Oriented Analysis and Design*



Instructor: Dr. Truong Ninh Thuan

Author: Team OOAD K57CA

March 23, 2015

# Course project

## Instructor

Dr. Truong Ninh Thuan - University of Engineering and Technology - VNU

## Team members

- |                             |       |                      |
|-----------------------------|-------|----------------------|
| • Truong Quoc Tuan (Leader) | K57CA | Student ID: 12020416 |
| • Nguyen Thac Thong         | K57CA | Student ID: 12020624 |
| • Le Van Giap               | K57CA | Student ID: 12020493 |

## Requirement

Give the analysis and design documentation of the software.

## Overview

VnUni is a website for high school students to get information about Vietnamese universities' admission. This report is the documentation for our website including three main parts: software's requirement, analysis and design.

## Acknowledgment

We would like to express our special thanks to Dr. Truong Ninh Thuan who gave us the enthusiastic instruction and support.

We would like to thank Team OOAD K56CA for their "YouTube Video Player" report that we used as reference and sample document.

# Table Of Contents

<b>1. Requirements</b>	<b>5</b>
1.1. Problem statement .....	5
1.2. Glossary .....	6
1.3. Supplementary Specification .....	7
1.4. Use-case model .....	9
1.4.1. Use application.....	9
1.4.2. Manage database.....	9
<b>2. Analysis</b>	<b>9</b>
2.1. Architectural Analysis .....	9
2.1.1. High-level Component.....	9
2.1.2. Key Abstractions .....	9
2.2. Use-case Analysis .....	9
2.2.1. Interaction diagrams .....	9
2.2.2. Use-case Realization View of Participating Class (VOPCs) .....	9
2.2.3. Analysis mechanism .....	9
2.2.4. Unify analysis classes .....	9
<b>3. Design</b>	<b>9</b>
3.1. Identify Design Elements.....	10
3.1.1. Subsystem Context Diagram .....	10
3.1.2. Analysis Class to Design Element map .....	10
3.1.3. Design Element to Package map.....	10
3.1.4. Architectural Components .....	10
3.2. Describe the Run-time Architecture .....	10
3.3. Describe Distribution.....	10
3.4. Use-case Design .....	10

3.5. Subsystem Design .....	10
3.6. Class Design.....	10
3.6.1. Describe each class, interface and relation in each package .....	10
3.6.2. Class diagram in total.....	10
<b>Appendix A - Figures</b>	<b>10</b>

# 1. Requirements

## 1.1. Problem statement

There are millions of high school students constantly taking university entrance examination every year in Vietnam. Due to the great number of enquiries about universities' and colleges' information on majors, quality teaching, admission marks, etc from students and their parents, many kinds of handbooks and websites are being published with diverse and up-to-date data to meet this demand.

However, these sources are sometimes not reliable; the interface is not friendly and unscientific – which make users confused and worried. To tackle this issue, we want a useful website called VnUni for students to help them approach easily the necessary information about universities and colleges by a brand-new way, which would hopefully become a trustful connection between students and schools in near future.

This website provides an information search engine based on different filters: location, majors, recently admission marks, etc. Moreover, users could have an objective and comprehensive view with “Comparison and Ranking” function that compares and rates schools according to various criteria.

Each university or college is supplied with a separate account to log in the system so that they could modify their data very conveniently.

This website must have an administrator that manages all operations of the system: add/remove a school, add/remove a user, modify database. Anonymous users must not have the privilege to register a new school account. It is only provided by administrator.

Users could connect to VnUni via client devices such as laptops, PC, tablets or smartphones.

## 1.2. Glossary

### **Brief Description**

The glossary contains the working definitions for all classes in the VnUni System.

### **Term**

#### *University*

An education organization in Vietnam.

#### *University Admission*

An activity of universities to admit students.

#### *Entrance exam*

An exam of Vietnam universities for graduated high-school students who want to get higher education.

#### *Faculty*

A group of university departments concerned with a major division of knowledge.

#### *Admission Mark*

A level of mark for students to take to the university.

#### *Admission Stuff*

An user of the university who has the permission to edit the information page of this university.

#### *Filter Catalog*

The fields catalog of filter system, helping user to filter out suitable universities quickly.

#### *User*

A person who view the website in their browser.

#### *Admin*

A person who develop, maintain the system and update database.

## 1.3. Supplementary Specification

### **Objectives**

The purpose of this document is to define requirements of the VnUni System. This Supplementary Specification lists the requirements that are not readily captured in the use cases of the use-case model. The Supplementary Specifications and the use-case model together capture a complete set of requirements on the system.

### **Scope**

This Supplementary Specification applies to the VnUni System which will be developed by the VnUni Team – Computer Science Department – University of Technology and Engineering, Vietnam National University.

The VnUni System will enable users to search and view the admission information of universities and colleges in Vietnam. This system also allows admission staffs to provide or modify the data of their schools.

This specification defines the non-functional requirements of the system; such as reliability, usability, performance, and supportability as well as functional requirements that are common across a number of use cases. (The functional requirements are defined in the Use Case Specifications.

### **References**

Applicable references are:

- IBM Rational Software Documentation (Version 2004).
- Document of K55CA OOAD project - Restaurant Management System.
- Document of K56CA OOAD project - YouTube Video Player.

### **Functionality**

- Multiple users must be able to perform their works concurrently.
- All functionality shall be available remotely through an internet connection. This may require applications or controllers running on the remote computers.

## Usability

This section lists all of those requirements that relate to, or affect, the usability of the system.

- *Browser Compliance*

The user-interface should run smoothly on all browsers, for instant: Chrome, Firefox, and Internet Explorer, etc.

- *Design for Ease-of-Use*

The user interface of the VnUni System shall be designed for ease-of-use and shall be appropriate for a computer-literate user community with no additional training on the System. The user interface has to be nice and clear.

## Reliability

The VnUni System shall be available 24 hours a day, 7 days a week. There shall be no more than 4% down time.

## Performance

The performance characteristics of the system are outlined in this section.

- *Simultaneous Users*

The system shall support up to 10000 simultaneous users against the central database at any given time, and up to 5000 simultaneous users against the local servers at any one time.

- *Database Access Response Time*

The system shall provide access to the VnUni database with no more than a 5 second latency.

## Supportability

None.

## Security

The system must prevent anonymous users to modify the database. Almost changes of the system databases can only be done by managers. Require confirm account before submit the changes.



## **Design Constraints**

This system will provide only for any browsers supporting JavaScript.

### 1.4. Use-case model

#### 1.4.1. Use application

#### 1.4.2. Manage database

## **2. Analysis**

### 2.1. Architectural Analysis

#### 2.1.1. High-level Component

#### 2.1.2. Key Abstractions

### 2.2. Use-case Analysis

#### 2.2.1. Interaction diagrams

#### 2.2.2. Use-case Realization View of Participating Class (VOPCs)

#### 2.2.3. Analysis mechanism

#### 2.2.4. Unify analysis classes

## **3. Design**

- 3.1. Identify Design Elements
  - 3.1.1. Subsystem Context Diagram
  - 3.1.2. Analysis Class to Design Element map
  - 3.1.3. Design Element to Package map
  - 3.1.4. Architectural Components
- 3.2. Describe the Run-time Architecture
- 3.3. Describe Distribution
- 3.4. Use-case Design
- 3.5. Subsystem Design
- 3.6. Class Design
  - 3.6.1. Describe each class, interface and relation in each package
  - 3.6.2. Class diagram in total

## **Appendix A - Figures**