Software Requirements Specification

for

Teachy

Version 1.0 approved

Prepared by Team 2

Hanoi, 20/10/2016

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Teachy v1.0	20/10/2016	This is the very first version	1.0

1. Introduction

1.1.1 Purpose

This document provides detailed requirements of Teachy v1.0, both functional and non-functional.

1.2 Document Conventions

SRS design (Software requirements specification IEEE 830) is applied in this document.

1.3 Intended Audience and Reading Suggestions

Developers, project manage, tester and users shoud read this for details of the system.

For a overview of the document, please take a look at the table of content.

1.4 References

https://en.wikipedia.org/wiki/Software_requirements_specification

2. Overall Description

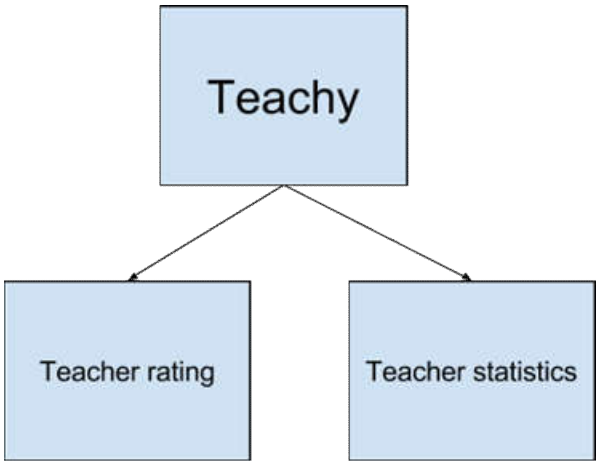
2.1 Product Perspective

Teachy is a new, self-contained web app which is developed by a group of young enthusiastic CS students.

2.2 Product Functions

Teachy is quite simple with two main functions:

- Teacher rating
- Teacher statistics



2.3 User Classes and Characteristics

Users of Teachy should be

- Students
- Staffs of department of studies
- Others who have interest in teaching performance.

2.4 Operating Environment

As a user, Teachy can be run on ordinary browsers such as Chrome, Firefox and Internet Explore. Users with a normal PC or laptop shall access and use the application.

2.5 Design and Implementation Constraints

The product is designed in MVC pattern and implemented in waterfall model.

2.6 User Documentation

Teachy is very easy to use that users can do it well without a tutorial.

2.7 Assumptions and Dependencies

Teachy servers run on Amazone web service with a very small chance to be affected by unexpected problems.

3. External Interface Requirements

3.1 User Interfaces

Teachy interfaces follow W3C standards, which can be found at https://www.w3.org/standards/.

3.2 Hardware Interfaces

Teachy has no interaction with hardware components.

3.3 Software Interfaces

Teachy consists of two components:

- Front-end: using Angular JS
- Back-end: using Laravel 5.3 (a PHP framework) with database MySql 14.14

3.4 Communications Interfaces

HTTP request with content-type application/json is used to communicate between the two components.

4. System Features

4.1 Teacher Rating

4.1.1 Description and Priority

Allows students to rate their teachers by various properties

4.1.2 Stimulus/Response Sequences

A student's report is recorded and later shown in statistics.

4.1.3 Functional Requirements

REQ-1: A user shall be able to search for all available teacher and class that they want to rate.

REQ-2: The software shall record correctly what students rated.
REQ-3: All teacher and class data shall be updated every semeter.

4.2 Teacher Statistics

4.2.1 Description and Priority

Allows students to see teachers rated by students who took their classes. A student can then chose a class he/she finds most suitable.

4.2.2 Stimulus/Response Sequences

After choosing a teacher, a new page about statistics of this teacher is shown to users.

4.3.3 Functional Requirements

REQ-1: Users shall be able to search for all available teachers.
REQ-2: The software shall show correctly the performance of the teacher in term of various properties based on rating by their students.
REQ-2: The software shall rank teachers by one or more properties or compare two teachers

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The system should handle a total of 10.000 users and upto 1000 users simultaneously.

Any request should be handled by the server in 1-2s and return to the client.

5.2 Safety Requirements

DDoS and other network attacks can be harmful to the product.

5.3 Security Requirements

The application accesses student's study schedule and private rating for their teachers.
This information should not be shown to other parties such as teachers or other students.
A student is advised to keep his/her student password for his/herself.

5.4 Software Quality Attributes

Adapts quickly to UET or any other university environment. Flexible to changes or addition of classes, teachers, students from different school years.

5.5 Business Rules

The school department of study is resposible for running, maintenent and any changes to the databases.
Students are responsible for their report and take full responsibility for spamming or impolite, inappropriate comments.
The school department of studies is also responsible for any queries/ requests from teacher about their ratings.