Software Requirements Specification

Version 1.0

June24, 2020

Virtual Classroom Website

حسن عيد

مصطفى عودة

الياس عويس

Contents

[1.0. Introduction 1](#_Toc43391823)

[1.1. Purpose 1](#_Toc43391824)

[1.2. Scope of Project 1](#_Toc43391825)

[1.3. Glossary 2](#_Toc43391826)

[1.4. References 2](#_Toc43391827)

[1.5. Overview of Document 2](#_Toc43391828)

[2.0. Overall Description 3](#_Toc43391829)

[2.1 System Environment 3](#_Toc43391830)

[2.2 Functional Requirements Specification 4](#_Toc43391831)

[2.2.1 Teacher Use Cases 4](#_Toc43391832)

[2.2.2 Student Use Cases 7](#_Toc43391833)

[2.3 User Characteristics 9](#_Toc43391834)

[2.4 Non-Functional Requirements 9](#_Toc43391835)

[3.0. Requirements Specification 10](#_Toc43391836)

[3.1 External Interface Requirements 10](#_Toc43391837)

[3.2 Functional Requirements 10](#_Toc43391838)

[3.3 Detailed Non-Functional Requirements 10](#_Toc43391839)

# 1.0. Introduction

## 1.1. Purpose

The purpose of this document is to present a detailed description of the virtual classroom website. It will explain the purpose and features of the website, the interfaces of the website, what the website will do, the constraints under which it must operate and how the website will react to external stimuli. This document is intended for both the stakeholders and the developers of the website.

## 1.2. Scope of Project

This software system will be a Virtual Classroom Website for teachers and students who prefer E-Learning. This system will be designed to help teachers and students by providing tools to communicate students with teachers and organize education materials.So teacher can organize course’s content and student can access to the content either real time or at any other time.

More specifically, this system is designed to allow teacher to create course and add lessons and interact with student enrolled the course.

The software will facilitate communication between teacher and student via interaction in the lesson.

The system also contains a NonRelational database (MongoDB) containing data about teachers, students, courses, lessons.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Virtual Classroom | Software that make teacher to be able to present education material and interact with student during explanation of the lessons. |
| Online Teacher | Teacher that has logged in the website. |
| Online Student | Student that has logged in the website. |
| Course Code | After creation of a course the system will generate a unique random string called (course code), this will use when student want to enroll the course. |
| Live Video Stream | A video recorded by teacher. Student can watch it in the real time. |

## 1.4. References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

## 1.5. Overview of Document

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

# 2.0. Overall Description

## 2.1 System Environment

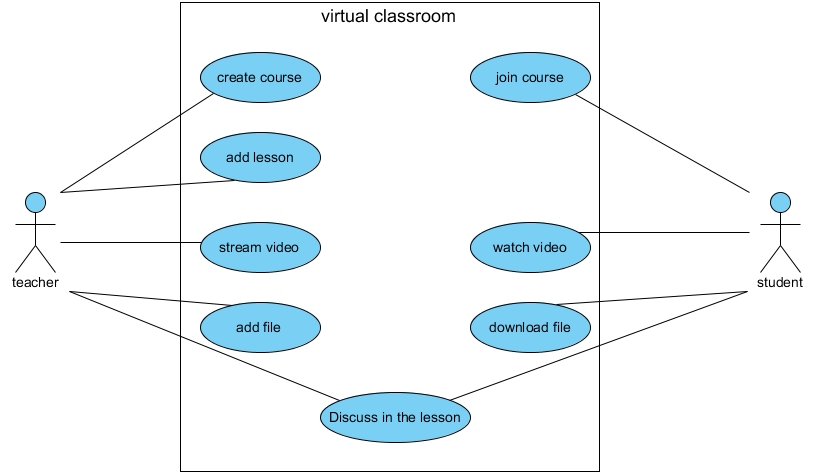


Figure - System Environment

The Virtual Classroom Website has Teacher and many Students.

The Teacher can create course and lessons content.

The Student can access to content of enrolled courses.

Teacher and Students can discuss via Discussion Section in lesson page.

## 

## 2.2 Functional Requirements Specification

This section outlines the use cases for each of the teachers and students.

### 2.2.1 Teacher Use Cases

#### Use case: Create Course

**Diagram:**

Teacher

Create Course

**Brief Description**

The Teacher can create course in the website after signing up and logging in.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Teacher has already accessed the Virtual Classroom Website.

1. The Teacher sign up in the website to have an account then log in.
2. The Teacher enter the name of new course that will created.
3. The Teacher can add course image else the system will use the default course image.
4. The System generate Course Code to this course.
5. The Teacher can give Course Code to Student that wanted to join the course.

#### Use case: Add Lesson

**Diagram:**

Teacher

Add Lesson

**Brief Description**

The Teacher can add lesson to the course after create it.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Teacher created a course.

1. The Teacher clickAdd Lesson Button to open Add Lesson Form.
2. The Teacher enter the name of new lesson.
3. The System determine that as lesson created date.
4. The System links the lesson data with the course.

#### Use case: Stream Video

**Diagram:**

Teacher

Stream Video

**Brief Description**

The Teacher can stream one live video for each lesson.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Teacher added a lesson.

1. The Teacher click live video button.
2. The Teacher can choose to stream either whole screen or an app window.
3. Any online student enrolled this course can watch the video in realtime.
4. The System will save video automatically so student can rewatch it later.

#### Use case: Add File

**Diagram:**

Teacher

Add File

**Brief Description**

The Teacher can add files to lesson.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Teacher added a lesson.

1. The Teacher click add file button to open the file upload form.
2. The Teacher choose file to upload and press upload button.
3. The System save the file in the file system and save the file name in the DB to access it.
4. The Teacher can download and remove the files that uploaded.

#### Use case: Discuss in the Lesson

**Diagram:**

Teacher

Discuss in the Lesson

**Brief Description**

The Teacher can send messages like a chat in the lesson to communicate with student.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Teacher added a lesson.

1. The Teacher write the message in the Discussion Section and click send button.
2. The System will save the message in the DB.
3. The System send a copy of it to all online students in the course in real time.

### 2.2.2 Student Use Cases

#### Use case: Join Course

Student

Join Course

**Diagram:**

**Brief Description**

The Student can join a course by entering the Course Code that get from The Teacher.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Student must sign up and log in the website.

1. The Teacher give the Course Code to Student outside the website.
2. The Student click join course button to open join course form.
3. The Student Enter the Course Code and click join.
4. The System check if there is a course with this code and if the course is existed The System join this Student to this Course.
5. The Student can discover the content of the course.

#### Use case: Watch Video

Student

Watch Video

**Diagram:**

**Brief Description**

The Student can watch live video streamed by teacher or watch lesson video saved on the server after streamed by teacher.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Student must join the course.

1. The Student choose a lesson and go to lesson page.
2. If The Teacher start a live video stream and don’t stop it yet, The Student can watch the live stream in real time.
3. If The Teacher stop the stream, The Student can watch the saved video at any time.

#### Use case: Download File

Student

Download File

**Diagram:**

**Brief Description**

The Student can Download the Lesson Files added by The Teacher.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Student must join the course.

1. The Student choose a lesson and go to lesson page.
2. The Student on file name that he wants to download it.
3. The Student select the location to save downloaded file then download starts.

#### Use case: Discuss in the Lesson

**Diagram:**

Student

Discuss in the Lesson

**Brief Description**

The Student can send messages like a chat in the lesson to communicate with teacher andother students.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Studentmust join a course.

1. The Student write the message in the Discussion Section and click send button.
2. The System will save the message in the DB.
3. The System send a copy of it to online teacher and all online students in the course in real time.

## 2.3 User Characteristics

The Reader is expected to be able to use web browsers.

The main screen of the Virtual Classroom Website will have the courses Created by The Teacher or the courses Enrolled by The Student.

The Course Page will have the Lessons added by The Teacher.

The Lesson Page will have 3 Sections   
(video section, files section, discussion section).

The detailed look of these pages is discussed in section 3.2 below.

## 2.4 Non-Functional Requirements

The Virtual Classroom Website will be on a server with high speed Internet capability. The software developed here assumes the use of a tool to connect with database and a tool for Real Time communication.

# 3.0. Requirements Specification

## 3.1 External Interface Requirements

This system needs to interact with any web browser on any operating system to enable the user to interact with the website.

In addition, it requires communication with MongoDB database in order to store and retrieve data for the professor, students, courses, lessons and modification thereof.

## 3.2 Functional Requirements

3.2.1 Create Course

|  |  |
| --- | --- |
| **Use Case Name** | Create Course |
| **Trigger** | The teacher select to create a new course . |
| **Precondition** | The teacher is inside *Courses* page on the website . |
| **Basic Path** | 1. The teacher click on *Create Course* button . 2. The system displays windows which contains *Create Course* Form to enter course information . 3. The teacher enters the information and submits the form . 4. The system checks that required fields are not blank . 5. The system store this course in database and generates *course code* . 6. The system will create a new page for this course and then move to it . 7. The teacher can give *course code* that is existed in course page to his students that wanted to join the course. |
| **Alternative Paths** | In step 3, if any required field is blank , the teacher is instructed to add an entry . |
| **Postcondition** | The course has been stored in database . |
| **Exception Paths** | The teacher may abandon the operation at any time . |
| **Other** | None |

3.2.2 Add Lesson

|  |  |
| --- | --- |
| **Use Case Name** | Add Lesson |
| **Trigger** | The teacher select to add a new lesson for specific course . |
| **Precondition** | The teacher is on *Courses* page inside lessons section . |
| **Basic Path** | 1. The teacher click on *Add new Lesson* button . 2. The system displays windows which contains *Add Lesson* Form to enter lesson information . 3. The teacher enters the information and submits the form . 4. The system checks that required fields are not blank . 5. The system store this lesson in database . 6. The system will move to the page for this lesson through which the teacher can upload files and attachments to the lesson . |
| **Alternative Paths** | In step 3, if any required field is blank , the teacher is instructed to add an entry . |
| **Postcondition** | The lesson has been stored in database . |
| **Exception Paths** | The teacher may abandon the operation at any time . |
| **Other** | None |

3.2.3 Stream Video

|  |  |
| --- | --- |
| **Use Case Name** | Stream Video |
| **Trigger** | The teacher select to create a live video . |
| **Precondition** | The teacher is inside lesson page . |
| **Basic Path** | 1. The teacher click on *Live Video* button . 2. The teacher chooses the way the video will be broadcast, either it is full screen or is limited to the application window. 3. The system will send notifications to all online students when live video starts . 4. Any online student enrolled this course can watch the video in realtime . 5. The system will save video automatically so student can rewatch it later . |
| **Alternative Paths** | None |
| **Postcondition** | The video broadcast has ended and the video has been saved to the database . |
| **Exception Paths** | If an error occurs, the teacher stops broadcasting and uploads a recorded video for students to view later . |
| **Other** | None |

3.2.4 Add File

|  |  |
| --- | --- |
| **Use Case Name** | Add File |
| **Trigger** | The teacher select to upload new file . |
| **Precondition** | The teacher is inside *lesson* page . |
| **Basic Path** | 1. The teacher click  *Add File*  button to open the file upload form. 2. The teacher choose file to upload and press upload button. 3. The system save the file in the file system and save the file name in the DB to access it. 4. The teacher can download and remove the files that uploaded. |
| **Alternative Paths** | None |
| **Postcondition** | The file has been stored in file system and its name saved in database . |
| **Exception Paths** | The teacher may abandon the operation at any time . |
| **Other** | None |

## 3.3 Detailed Non-Functional Requirements