Software Requirements Specification

for

Charles Xavier's School for Gifted Youngsters Learning Management System

Version 1.0 approved

Prepared by Luis Gonzalez, Donald Mathews, Tan Le, Hunter Newton, Brian Cobo

Software Engineering 3321

February 1, 2019

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

2. Overall Description 1

2.1 Product Perspective 1

2.2 Product Functions 1

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

3. External Interface Requirements 2

3.1 User Interfaces 2

3.2 Software Interfaces 2

3.3 Communications Interfaces 2

# Introduction

## Purpose

The purpose is to develop a learning management system to help Charles Xavier's School for Gifted Youngsters with their activities and improve their services, and for the management to track a student's basic information.

## Document Conventions

The higher level requirements are assumed to be inherited by detailed requirements. Such requirements include being able to store and retrieve student information. It will go into more detail in section 4.1.

## Intended Audience and Reading Suggestions

This document is intended for project managers, marketing staff, developers, and users. It is intended to provide an overview of the project in a way that everyone will be able to understand the purpose and features of the Learning Management System.

## Product Scope

The goal of the project is to be able to assist the administrators of Charles Xavier's School for Gifted Youngsters with managing student information. More specifically, being able to store and retrieve information about the student such as their name, ID, courses, and grades per semester. This will increase with the efficiency and organization of the school's students and pertinent information.

# Overall Description

## Product Perspective

The school is rather old and have their files on paper. While the system works, the new system will increase efficiency and allow them to maintain larger amounts of students.

## Product Functions

Some of the functions that will be included are:

* Adding and removing students
* Adding and removing administrators
* Inputting grades into students files
* Changing grades in students files
* Provide updated grades for students

## User Classes and Characteristics

The classes include each of the main entities such as Administrators, Professors, Students, and Classes. They each contain attributes such as their names, login info, and their different functions they can perform, such as administrators being able to add and remove students, professors, classes. Professors being able to add and remove student grades. Classes are able to contain exam grades. Students being able to view their courses and grades.

## Operating Environment

The software will operate as a web application created by Flask, and the database hosted on AWS.

## Assumptions and Dependencies

The project is dependent on internet access and is also dependent on the hosting service. However, an offline version could be made where the database can be stored locally.

# External Interface Requirements

## User Interfaces

The user interface will be similar to that of a website. The user will log in and a dashboard of their user functions will show up. There'll be different buttons where the user can choose if they want to view different attributes, make changes to the database. Some of the error messages may include not being able log in, corrupted database, function not being able to be performed.

## Software Interfaces

The software is flask python application which is used online. We have the backend database run by MySQL which is hosted on AWS. The flask application communicates with the database to store and retrieve information.