**Software Project Management Plan**

Learning Management System Project – Codenamed: Xmen

CS 3321

Introduction to Software Engineering

Spring, 2019

**Team Members**

Tan Le

Brian Cobo

Donald Mathews

Hunter Newton

Luis Gonzalez

Document Control

**Change History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Change Date** | **Description of changes** | **Initials** |
| V1.0 | 02/28/19 | Initial release | LG |
|  | 04/24/19 | Latest | LG |

Table of Contents

[1.1 Purpose and Scope 3](#_Toc7003885)

[1.2 Goals and Objectives 3](#_Toc7003886)

[1.3 Project Deliverables 4](#_Toc7003887)

[1.4 Assumptions and Constraints 4](#_Toc7003888)

[1.4.1 Assumptions 4](#_Toc7003889)

[1.4.2 Constraints 4](#_Toc7003890)

[1.5 Schedule Summary 5](#_Toc7003891)

[1.5.1 Cost Estimate 5](#_Toc7003892)

[1.5.2 Schedule Summary 5](#_Toc7003893)

[1.6 Success Criteria 6](#_Toc7003894)

[1.7 Definitions 6](#_Toc7003895)

[2 Startup Plan 6](#_Toc7003896)

[2.1 Team Organization 6](#_Toc7003897)

[2.2 Project Communications 7](#_Toc7003898)

[2.3 Technical Process 7](#_Toc7003899)

[2.4 Tools 7](#_Toc7003900)

[3 Work Plan 7](#_Toc7003901)

[3.1 Release Plan 7](#_Toc7003902)

[3.1.1 Final Product 7](#_Toc7003903)

[4 Control Plan 7](#_Toc7003904)

[4.1 Monitoring and Control 7](#_Toc7003905)

[4.2 Configuration Management Plan 8](#_Toc7003906)

[5 Supporting Process Plans 8](#_Toc7003907)

[5.1 Test Plan 8](#_Toc7003908)

[5.2 Product Acceptance Plan 8](#_Toc7003909)

Overview

## Purpose and Scope

This project develops a learning management system (LMS) to help a university IT department their activities and improve their services, and for the management to track student’s basic information. Typical LMS includes Blackboard or Moodle. This class project is intended to be done by a team of 4-5 undergraduate students during an academic semester, in conjunction with lectures and other class activities. It is similar to Blackboard or Moodle LMS system.

The project goal is to create a development build that will satisfy the project requirements. The development build consists of running the application on a machine and accessing the management system as a user to perform available functions.

The application will be created using Python’s Flask framework and will provide the ability for a user to login and either review system information (read-only) or update information based on user role. The data is stored in AWS RDS MySQL database. The profile types are student and school administrator.

## Goals and Objectives

Since a user can assume three distinct roles, the overall objective is to allow an admin role user to create and manage courses, enrollments and for a teacher role to manage grades in their courses. A student role will be able to review information for classes they are enrolled in.

Project Goals:

1. Create a working prototype of LMS using FLASK framework for class project.
2. Create Documentation to assist in development of the application
3. Learn Software Development lifecycle.
4. Use Git

Project Objectives:

1. Create Application structure in flask using MVC design pattern
2. Create Database for persistent storage in AMAZON WebServices.
3. Use Python as language of choice
4. Use Template engine to generate HTML and CSS for application views

## Project Deliverables

|  |  |
| --- | --- |
| **Date** | **Deliverable** |
| 04/22/10 | Development Artifacts |
| 04/24/10 | Documentation |
| 04/24/10 | Presentation |
| 04/28/10 | Project Submission |

## Assumptions and Constraints

### Assumptions

1. Application can be run locally in each of developers machines
2. All developers will use python and mysql db
3. All work will be committed to GIT
4. This is not a full-blown LMS system, team will explore construction of basic features, with room for rapid extension and addition of features.

### Constraints

Constraints:

1. Team availability. Can only meet on Monday. Text messaging is the preferred method of communication
2. Programming Methodology experience. We need to learn Flask
3. Experience producing support documentation.

## Schedule Summary

### Cost Estimate

We are working for free. The time estimate is developers will spend 2-16 hours weekly.

### Schedule Summary

**February 25 – March22 2019**

Assign Roles

Discuss technology to use

Create Database design

Training on how to use Flask and submodules

**March 25 – March29 2019**

Create Database on remote location (AWS)

Generate Database structure, tables, columns

Create Flask Application prototype (starter pack)

Discuss Flask routes

Assign Documentation to Produce.

**April 1 – April 12 2019**

Create Class Design

Discuss Use Case for each role

Create UI for Application

* Forms
* Style
* Info Tables

Create SQL Queries

Discuss python classes

Discuss Testing

**April 15 – April 24 2019**

End to End testing

Testing and Integration

UI corrections

## Success Criteria

The application runs and performs the actions described in the requirements.

## Definitions

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **Actor** | user or role in the LMS system |
| **Developer** | A team member |
| **Project** | Activities performed to have a running application that meets requirements |
| **XMEN LMS Application** | The codename for the project |
| **User** | The person logged in the application |
| **Use case** | Describes a goal-oriented interaction between the system and a user depending on its role |

# Startup Plan

## Team Organization

|  |  |  |
| --- | --- | --- |
| **Role** | **Actor(s)** | **Responsibility** |
| Project Manager / Main Developer | Hunter | Leader and main developer. Writes Code. Coordinates and assigns tasks for team. Resolves technical issues. |
| Advisor / Architect / UI | Luis | Provides guidance on overall system. Writes Code. Resolves technical issues. UI Design. |
| Architect /Requirements Engineer | Brian | Provides guidance on overall system. Requirements Expert. |
| Database Expert | Donald | Designs DB |
| Quality Assurance / Developer | Tan | Quality Assurance Expert. Writes Code. |

## Project Communications

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Event** | **Information** | **Audience** | **Format** | **Frequency** |
| Team Meeting | Task Status and Questions | All team members | Informal meetings during class; | Wednesday |
| Project Messaging | Finished Items, Questions, Assignments | All team members | Text Messaging | As needed |

## Technical Process

An iterative and incremental development process is planned. Feedback will be used from each iteration to improve the next. The first iteration will focus on basic functionality of the application.

## Tools

* Programming & Markup Languages – Python, SQL, HTML, CSS, JAVASCRIPT
* Operating System – all
* Version Control – Github <https://github.com/ehunternewton/SWEproject>
* Development Tools – PyCharm, MySQLWorkbench

# Work Plan

## Release Plan

A working application will be presented 04/24/2019

### Final Product

Final product will be a Developmental version of system and will be showcased from personal machine.

# Control Plan

## Monitoring and Control

We are using GITHUB to keep track of application versioning.

The database used have periodic backup available in case the database its corrupted.

## Configuration Management Plan

For the application to run a Flask server needs to be installed in local machine. This requires the installation of python modules and verification by each of the team members.

An AWS database is created for the team to use a single source of data during development.

The following procedure is to be used when making changes to python and html source:

1. Before making any change pull latest version from GITHUB
2. Check for breaking changes
3. Make change
4. Test that nothing was broken.
5. Push to GITHUB before closing development session.

# Supporting Process Plans

## 

## Test Plan

The test design specification was created and can be accessed here <https://github.com/ehunternewton/SWEproject/blob/master/Test%20Design%20SWE.docx>

## Product Acceptance Plan

Acceptance is achieved when agreed functionality is reached with minimal errors.