***Learning Management System***

Software Project Management Plan forLearning Management System

**Responsible Party**: Oscar Garcia, Moises Morales, Nabil Mehari, Florine Nswal

**Date**: 9/19/2019

*Table of Contents*

1. ***Introduction***
2. ***Project Organization***
3. ***Managerial Process***
4. ***Technical Process***
5. ***Work Packages, Schedule, & Budget***
6. ***Introduction***

This final project is to implement a learning management System that deals with all kind of student details such as the academic related reports, college details, course details, curriculum, batch details and other resource related details as well.

* 1. **Project Overview**

This project tracks all the details of a student from the day one to the end of his course which can be used for all reporting purpose, tracking of attendance, progress in the course, completed semesters years, coming semester year curriculum details, exam details, project or any other assignment details, final exam result, etc.

* 1. **Evolution of the SPMP**

We plan to use C# under the direction of Oscar Garcia to accomplish our goals for this program. We work on the project at least twice a week after lectures, and once a week outside of the classroom to ensure the overall progress on all aspects of this project.

* 1. **Definitions and Acronyms**

**UML (Unified Modeling Language)**: This is a modeling language that helps visualize how a system is designed. This is NOT a programming language.

**SQL (Structured Query Language)**: This language helps communicate with a database & manage any data present in said database.

**C#**: A programming language created by Microsoft that uses the .NET Framework. It shares many similarities with C++, but also has its fair share of differences.

**GUI (Graphical User Interface)**: This allows the user to interact with the program with visual & audio indicators.

**SPMP (Software Project Management Plan)**: A document created by the project managers upon the completion of the project’s design. This lays out exactly what will be done in the project & how they’ll be done.

# *Project Organization*

For this project, we used the democratic model where we all voted on how we wanted to proceed for the best result of the project. We used GitHub for communication, C# and Visual Studio for the code implementation.

2.1 **Process Model**

Rapid prototyping

2.2 **Organizational Structure**

We worked on this project twice a week in class after lecture and we met once a week outside of the classroom to work on the project. Also, we used GroupMe app and GitHub for communication.

This project helps to understand how the process of checking our classes & grades actually works from the technical side of things.

] 2.3 **Organizational Interfaces**

|  |  |  |
| --- | --- | --- |
| ORGANIZATION | LIASISON | CONTACT INFORMATION |
| DR. Chang | CS 3321 | 800-000-0000 |

2.4 **Project Responsibilities**

We basically did not assigned any responsibility to each individual we all met once a week outside of the class room to work in specific material that we wanted to work on that specific day, but Oscar and Moise had huge impact for the success of this specific project because of their expertise with C#.

|  |  |
| --- | --- |
| Oscar Garcia | GUI, DATA BASE, POWERPOINT |
| Moises Morales | GUI, UML, CASE DESIGN, POWERPOINT |
| Nabil Mehari | GUI, SPMP, POWERPOINT |
| Florine Nswal | GUI, SPMP, POWERPOINT |

# *Managerial Process*

The objective of this project was to build a learning management system that allows the students to see their personal information such as name, course, grade, and GPA for their current semester. Also, for the administrator to access student’s information and Edit if they had too.

3.1 **Management Objectives and Priorities**

The goal of this project was to learn how to create a learning management system that is similar to blackboard. Our first priority was to get the GUI done first followed by the database, the UML and the case design.

For this project we did not have any budget.

3.2 **Assumptions, Dependencies, and Constraints**

This project is based on the software engineering course, for the success of this project we had to meet at least once a week outside of the classroom for about two hours to discuss different aspects of the projects. We did not have any budget for this project

* **Project Assumptions**

1. Team of 4 students
2. Not budget
3. Software availability

* **Project Dependencies**

1. Administrator and student logins have different functionalities
2. Student cannot edit some information such as grade or GPA while the administrator can.

* **Project constraints**

1. Time consuming
2. Availability of existing software

3.3 **Monitoring and Controlling Mechanisms**

we meet once a week outside of the classroom for about two hours and twice a week during class section for 15 to 30 minutes, and we shared our update on GitHub.

* 1. **Staffing Approach**

For the success of this project we needed to be familiar with different things include C#, visual studio, GitHub and Microsoft office.

# *Technical Process*

This section specifies the technical methods, tools, and techniques to be used on the project. It also includes identification of the work products and reviews to be held and the plans for the support group activities in user documentation, training, software quality assurance, and configuration management.

4.1 **Methods, Tools, and Techniques**

We used the following programs to achieve the results needed for this project.

* **Program Creation Application**: Visual Studio 2017 & 2019
* **Programming Language**: C#
* **Database Type**: SQL

4.2 **Software Design Description (SDD)**

This product is capable of log-in input, with the ability to reset the password if you forgot it. Likewise, the product can discern between students & administrators. The former can look at their classes & grades as well as update their own personal information, while the latter can edit the students’ classes & grades.

4.3 **User Documentation**

We mostly used GitHub for all user documentation, including the program itself & the SPMP file. And while it’s not actually part of the product, the presentation for the product is in PowerPoint form, which was saved to Google Drive rather than GitHub.

# *Work Packages, Schedule, & Budget*

Specify the work packages, dependency relationships, resource requirements, allocation of budget and resources to work packages, and a project schedule. Much of the content may be in appendices that are living documents, updated as the work proceeds.

5.1 **Resource Requirements**

The main resources we needed were time & a place to work. Most of the time we used was the class time allotted to us. But within the last month of the time we had, we began to use the library’s resources to work in their rooms to get the product finished.

5.2 **Budget and Resource Allocation**

Due to the lack of money necessary for the project, we didn’t have a budget for this project. Most of the resources necessary to complete the program were either already provided to us by the school or were free to begin with.

5.3 **Schedule**

We didn’t really have a set schedule for the project. As mentioned in Section 5.1, we mainly worked in class or in the reserved library rooms over the course of the semester. We used GitHub as well as a group chat to keep in contact as well as documenting our progress.