

# **Software Project Management Plan**

Learning Management System

Version 1.0

Team: Jacquelyn Johnson, Andrew Samuel, Julio Quintero, Andrew Tomich, Ary  
Hernandez

Due Date: December 1, 2019

## Table of Contents

Project Overview.....	3
Project Organization .....	3

# **1 Introduction**

## **1.1 Project Overview**

The purpose of this project is to develop a LMS that is used to store important student data such as grades, student ID, registered courses, and GPA. It will contain two access modes, one for administrators and one for students. The project will be developed using Java and SQL programming languages.

## **1.2 Project Deliverables**

The team will produce a working LMS that will be delivered by the due date of December 1, 2019. The project must have a functioning GUI and make use of a UML diagram during the development phase. Along with the complete product, an SPMP documenting the development of the project will be provided.

## **1.3 Evolution of the SPMP**

All changes to the project will be documented in this SPMP. The Project management plan will be updated as development of the project continues.

## **1.4 Reference Materials**

*IEEE Std 1058-1998, IEEE Standard for Software Project Management Plans, IEEE 1998*

## **1.5 Definitions and Acronyms**

- UML – Unified Modeling Language
- LMS – Learning Management System
- SPMP – Software Project Management Plan
- GUI – Graphical User Interface
- Java – A general purpose, class based, and object-oriented programming language
- SQL – “sequel”, a programming language used for database systems and storing data

# **2 Project Organization**

This section of the SPMP will outline the organizational structure of the project including roles and responsibilities.

## **2.1 Process Model**

The project will make use a democratic team approach. Every team member is responsible for the project, and decisions will made in a democratic fashion. Java was chosen as the main programming language before the code was implemented. All major changes to the project will be subject to team approval.

## 2.2 Organizational Structure

### Team Members

- I. Andrew Tomich
- II. Julio Quintero
- III. Ary Hernandez
- IV. Jacquelyn Johnson
- V. Andrew Samuel

## 2.3 Organizational Interfaces

Describe the administrative and managerial interfaces between the project and the primary entities with which it interacts. A table may be a useful way to represent this information.

Organization	Liaison	Contact Information
Customer: <name>	<name>	<phone, email, etc.>
Subcontractor: <name>		
Software Quality Assurance		
Software Configuration Management		
<etc.>		

## 2.4 Project Responsibilities

Ultimately, all members are responsible for a working product.

1. Software Project Management Plan
2. Unified Modeling Language
3. Analysis
4. Source Code
5. Final Deliverable

## 3. Managerial Process

This section of the SPMP specifies the management process for this project.

### 3.1 Management Objectives and Priorities

The management objective is to deliver a working product that satisfies all requirements by the due date. Additionally, the product must be easy to use and intuitive for both access modes. Progress of the project will be monitored and outlined in this SPMP.

### 3.2 Assumptions, Dependencies, and Constraints

- The team consists of 5 members
- The deadline of the project is December 1, 2019
- Presentation displaying progress will be on September 26, 2019

- No budget

### 3.3 Risk Management

Describe the process to be used to identify, analyze, and manage the risk factors associated with the project. Describe mechanisms for tracking the various risk factors and implementing contingency plans. Risk factors that should be considered include contractual risks, technological risks, risks due to size and complexity of the product, risks in personnel acquisition and retention, and risks in achieving customer acceptance of the product.

The specific risks for this project and the methods for managing them may be documented here or in another document included as an appendix or by reference.

Risk	Possibility	Impact	Description	Solution

### 3.4 Monitoring and Controlling Mechanisms

The team will monitor the progress of the project by use of

- Shared repository
- Weekly Team meeting

### 3.5 Staffing Approach.

Describe the types of skills required for the project, how appropriate personnel will be recruited, and any training required for project team members.

## 4. Technical Process

### 4.1 Methods, Tools, and Techniques

- The team will use Java to implement the project.
- SQL will be used for the database system of records.

### 4.2 Software Documentation

- Unified Modeling Language
- Project Documents in Microsoft Office

#### 4.2.1 Software Requirements Specification (SRS)

The SRS clearly and precisely describes each of the essential requirements (functions, performances, design constraints, and attributes) of the software and the external interfaces. Each requirement is defined such that its achievement is capable of being objectively verified and validated by a prescribed method, for example, inspection, analysis, demonstration, or test.

#### **4.2.2 Software Design Description (SDD)**

The SDD describes the major components of the software design including databases and internal interfaces.

#### **4.2.3 Software Test Plan**

The Software Test Plan describes the methods to be used for testing at all levels of development and integration: requirements as expressed in the SRS, designs as expressed in the SDD, code as expressed in the implemented product. The test plan also describes the test procedures, test cases, and test results that are created during testing activities.

#### **4.3 User Documentation**

Describe how the user documentation will be planned and developed. (This may be just a reference to a plan being built by someone else.) Include work planned for online as well as paper documentation, online help, network accessible files and support facilities.

#### **4.4 Project Support Functions**

Provide either directly or by reference, plans for the supporting functions for the software project. These functions may include, but are not limited to, configuration management, software quality assurance, and verification and validation. Plans for project support functions are developed to a level of detail consistent with the other sections of the SPMP. In particular, the responsibilities, resource requirements, schedules and budgets for each supporting function must be specified. The nature and type of support functions required will vary from project to project. The absence of a software quality assurance, configuration management, or verification and validation plan, however, must be explicitly justified in project plans that do not include them.

## **5. Work Packages, Schedule, and 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 Work Packages**

Specify the work packages for the activities and tasks that must be completed in order to satisfy the project agreement. Each work package is uniquely identified. A diagram depicting the breakdown of project activities and tasks (a work breakdown structure) may be used to depict hierarchical relationships among work packages.

### **5.2 Dependencies**

Specify the ordering relations among work packages to account for interdependencies among them and dependencies on external events.

Techniques such as dependency lists, activity networks, and the critical path method may be used to depict dependencies among work packages.

### **5.3 Resource Requirements**

Provide, as a function of time, estimates of the total resources required to complete the project. Numbers and types of personnel, computer time, support software, computer hardware, office and laboratory facilities, travel, and maintenance requirements for the project resources are typical resources that should be specified.

### **5.4 Budget and Resource Allocation**

Specify the allocation of budget and resources to the various project functions, activities, and tasks.

### **5.5 Schedule**

Provide the schedule for the various project functions, activities, and tasks, taking into account the precedence relations and the required milestone dates. Schedules may be expressed in absolute calendar time or in increments relative to a key project milestone.

## **6. Additional Components**

Certain additional components may be required and may be appended as additional sections or subsections to the SPMP. Additional items of importance on any particular project may include subcontractor management plans, security plans, independent verification and validation plans, training plans, hardware procurement plans, facilities plans, installation plans, data conversion plans, system transition plans, or the product maintenance plan.

### **6.1 Index.**

An index to the key terms and acronyms used throughout the SPMP is optional, but recommended to improve usability of the SPMP.

### **6.2 Appendices**

Appendices may be included, either directly or by reference, to provide supporting details that could detract from the SPMP if included in the body of the SPMP. Suggested appendices include:

- A. Current Top 10 Risk Chart
- B. Current Project Work Breakdown Structure
- C. Current Detailed Project Schedule