**Software Project Management Plan (SPMP)**

The basic template to be used is derived from IEEE Std 1058-1998, IEEE Standard for Software Project Management Plans. The following is a template for the SPMP. It begins with a cover page that contains the version control and release information. Each section has a description of the information contained within.

Software Project Management Plan

for

<BIRDBOX>

<OMER MIR, FRANCISCO MOKUY, MONIKA SHIRMA, MIN THIA MYO, MARBIN GONZALEZ>

<3/15/19>

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Description of Versions / Changes** | **Responsible Party** | **Date** |
| 1.0 | Initial version of SPMP for comment by team | BIRDBOX | 2/20/19 |
| 1.1 | First revised draft | BIRDBOX | 3/20/19 |
| 1.2 | Second revised draft | BIRDBOX | 4/5/19 |
| 1.3 | Incorporated feedback from Team | BIRDBOX | 4/20/19 |
| 1.4 | Additional feedback from team incorporated prior to submission to chief programmer for review | BIRDBOX | 4/24/19 |

## Table of Contents

Introduction 3

1.1 Overview and Objectives 3

1.2 Deliverables and Internal Project Documents 3

1.3 Evolution of the SPMP 3

1.4 Reference Materials 3

1.5 Definitions and Acronyms 4

2 Project Organization 4

2.1 Process Model 4

2.2 Organizational Structure/Roles 5

3 Managerial Process 5

3.1 Risk Management 5

4 Technical Processes 6

4.1 Methods, Tools, and Techniques 6

4.2 Class Diagram 7

4.3 Testing 8

5 Refrences 9

# 1 Introduction

## 1.1 Overview and Objectives

The BIRDBOX Project is focusing on developing a Learning Management System deals with all kind of student details, academic related reports, college details, course details, curriculum, batch details and other resource related details too. It 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.

The *primary* objective of Project is to have 2 different information management systems, one for user and one for the administrator

The project will be executed over one semester, and the project team will consist of the following students – Marbin Gonzalez, Omer Mir, Monika Sharma, Francisco Mokuy, Min Thiha Myo. and Professor Chang will act as the primary customer for the project.

## 1.2 Deliverables and Internal Project Documents

The following deliverables will be provided for the project (to all stakeholders):

* Plans
  + Software Project Management Plan (SPMP)
  + Software Test Plan (STP)
  + Software Quality Assurance Plan (SQAP)
* Technical Documents and Software
  + Software Requirements Specification (SRS)
  + Software Architecture Document (SAD)
  + Software User Documentation
  + Source Code (including installation and configuration instructions).

## 1.3 Evolution of the SPMP

This initial draft of the Software Project Management Plan (SPMP) will be placed under change control. During each subsequent phase of the project, the project team will meet periodically to review any proposed changes to the plan, assess the impact of making those changes and recommend accordingly (approve, reject, defer). If necessary, an updated version of the SPMP will be published at the conclusion of each phase of the project.

## 1.4 Reference Materials

This Software Project Management Plan (SPMP) references the following existing project documentation:

* [Project Outlines](file:///C:\Users\computer\Downloads\(https:\d2l.pdx.edu\d2l\lms\content\viewer\main_frame.d2l%3fou=114468&tId=899550)
* [2012 Practicum Project Proposal](https://projects.cecs.pdx.edu/attachments/998/ProjectProposal.docx)
* [Distributed Software Engineering Process Specification (DSEPS)](https://projects.cecs.pdx.edu/~omsetrac/OMSE-555-556-2010-Project-A/index.cgi/browser/FormalDocumentation/trunk/Distributed%20Software%20Engineering%20Process%20Specification.doc)
* [DSEPS Process to SDLC Mapping (PSM)](https://projects.cecs.pdx.edu/~omsetrac/OMSE-555-556-2010-Project-A/index.cgi/browser/FormalDocumentation/trunk/Process%20to%20SDLC%20Mapping%20(PSM).docx)
* [SPMP](https://projects.cecs.pdx.edu:8443/~omsetrac/OMSE-555-556-2010-Project-A/) Format

## 1.5 Definitions and Acronyms

LMS- Learning Management System

GUI- Graphical User Interface

SPMP- Software Project Management Plan

IEEE- Institute of Electrical and Electronics Engineers

STP- Software Test Plan

SQAP-Software Quality Assurance Plan

SRS- Software Requirements Specification

SAD-Software Architecture Document

DSEPS-Distributed Software Engineering Process Specification

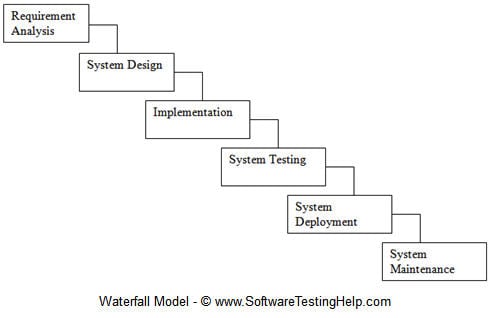
PSM- Process to SDLC Mapping

# 2 Project Organization

## 2.1 Process model

The project will use the Waterfall life cycle model with a democratic approach. This will allow the project team to:

* Focus on understanding the core desired functionality for the LMS.
* Conservatively manage project scope based on team member availability
* Establish a working relationship within the project team, and clarify roles and responsibilities



## 2.2 Organizational Structure/Roles

Min Thiha Myo is responsible for the development of the Graphical User Interface (GUI), Francisco Mokuy is overseeing development of the GUI as well as documentation such as the Class, Collaboration, and Sequence Diagram. Monika Sharma is in charge of all documentation as well overall team organization and managing testing and quality assurance. Marbin Gonzalez is responsible for the development of the Software Project Management Plan (SPMP), test cases, and any integration between the GUI and the database. Omer Mir is the primary engineer overseeing the development of the database as the data access tools.

Software Developer

Marvin Gonzalez

Chief

Programmer

Monika

Sharma

CUSTOMER

Professor

Chang

Software Developer

Min Thiha Myo

Software Developer

Omer

Mir

Software Developer

Francisco Mokuy

## 3 Managerial Process

## 3.1 Risk Management

The project team will maintain a risk management plan that lays out both the risks identified for the project, as well as the mitigation plan for each of those risks. Each risk will be classified in terms of probability of occurrence, and the degree of impact to the project schedule, on a scale of low (L), medium (M) or high (H). Any risk identified as having a probability of occurrence or degree of impact greater than low will be reviewed during a periodic review meeting with the primary stakeholder.

For the currently identified risks, the mitigation plan is as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Probability**  **(L/M/H)** | **Impact**  **(L/M/H)** | **Mitigation Plan** |
| Limited resources | H | M | Project team will manage scope accordingly, and each team member will assume multiple project roles (see section 2.2 above) |
| Unavailability of main stakeholder | M | L | Project team will assume the role if main stakeholder is unavailable, and document any assumed requirements for review at a later meeting |
| Confusion around practicum project process | H | H | Addressed by week 4 recalibration meeting |
| New development process / project team | M | M | Project team will review DSEPS |

## 4 Technical Processes

## 4.1 Methods, Tools, and Techniques

Based on the outlined system and the project team’s technical expertise, it is proposed that the following software development environment (tools, languages, operating systems and utilities) will be adopted for this project:

* IDE used: NetBeans
* Programming language: Java
* Using XXAMP for database and storage
* ArgoUML application for UML diagram
* JUnit 5 for testing

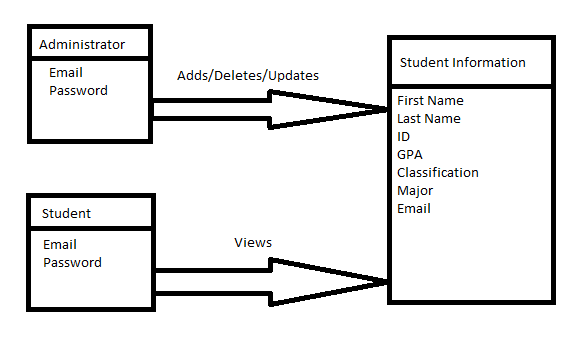


We used the Net Beans IDE to create the Program while using the Programming Language JAVA. We decided to go with NetBeans because It highlights source code syntactically and semantically, lets you easily refactor code, with a range of handy and powerful tools. It also cross platform compatible so it benefited us in using it. NetBeans provides editors, wizards, and templates to help you create applications in Java, PHP and many other languages and since, we used Java this was the optimal way to go.



XXAMP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible. This Platform was the best choice to use as our database since it was free and completed the job.

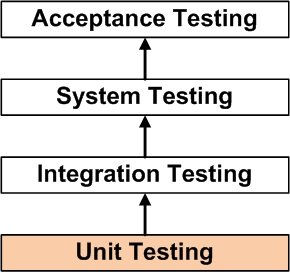
## 4.2 Class Diagram



This is the format we implemented in our program. We wanted two classes the administrator and the Student. We wanted the administrator to have the power to add, delete, or update any information that is available to them and the student. The student however would only be able to view their information when they log in

## 4.3 Testing

` For testing we used the following model



UNIT TESTING is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

INTEGRATION TESTING is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

SYSTEM TESTING is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system’s compliance with the specified requirements.

And finally, ACCEPTANCE TESTING is a level of software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system’s compliance with the business requirements and assess whether it is acceptable for delivery.

## 5 References

* [http://softwaretestingfundamentals.com](http://softwaretestingfundamentals.com/)
* <https://en.wikipedia.org/wiki/XAMPP>
* <https://netbeans.apache.org/>
* <https://cs.uwaterloo.ca/~apidduck/se362/Assignments/A2/spmp>
* <http://softwaretestingfundamentals.com/unit-testing/>