

# **Software Project Management Plan**

**The Linux School  
Website**

**VERSION 1.3**

**Jeffrey Ampeh**

## TABLE OF CONTENTS

<b>TABLE OF CONTENTS</b> .....	<b>2</b>
<b>CHANGE HISTORY</b> .....	<b>3</b>
<b>INTRODUCTION</b> .....	<b>4</b>
PROJECT OVERVIEW .....	4
EVOLUTION OF THE SPMP .....	4
REFERENCE MATERIALS .....	4
<b>PROJECT ORGANIZATION</b> .....	<b>5</b>
PROCESS MODEL .....	5
PROJECT STRUCTURE .....	5
PROJECT RESPONSIBILITIES .....	6
<b>MANAGERIAL PROCESS</b> .....	<b>6</b>
OBJECTIVES .....	6
PRIORITIES .....	6
SCHEDULE .....	6
TENTATIVE BUDGET .....	6
PROJECT ANTICIPATIONS .....	7
RISK MANAGEMENT.....	7
MONITORING AND CONTROLLING MECHANISM .....	8
<b>TECHNICAL PROCESS</b> .....	<b>8</b>
DEVELOPMENT TOOLS .....	8
DOCUMENTATION PLAN.....	9
<b>WORK BREAKDOWN STRUCTURE</b> .....	<b>10</b>

## CHANGE HISTORY

SPMP VERSION	RELEASE DATE	RELEASE NOTES
1.0	10/01/2020	Baseline version
1.1	10/08/2020	Changes to phrasing
1.2	10/13/2020	<ul style="list-style-type: none"><li>• Budget included</li><li>• Development tools upgraded</li><li>• Process model updated</li><li>• Work breakdown structure updated</li></ul>
1.3	10/14/2020	<ul style="list-style-type: none"><li>• Changes to phrasing</li><li>• Updated schedule and budget</li></ul>

# INTRODUCTION

## PROJECT OVERVIEW

The project involves the creation of a functional website for a Linux training school. The website would not only serve as an online presence and marketing tool for the school but also consist of a registration and payment platform for users interested in enrolling in the school. The website would have a blog to inform users, allow alumni and professionals to share their ideas, and answer any questions new users and current students might have.

The website would also have a newsfeed that would inform users daily on the latest and groundbreaking trends in IT and Linux in particular.

The website would provide users with prospective careers in Linux, industries they can work in and the average salaries they could earn.

## EVOLUTION OF THE SPMP

As the project is developed, new information and details may be discovered that requires changing the structure or timelines established in this document.

To ensure these changes can be quickly and easily made and communicated, changes would be discussed in future versions of this document to keep the client informed and provide them a chance to process, discuss and decide whether they approve of the changes or not.

If significant changes need to be made to this document or any other document, that might alter the project schedule and structure, the new version will state them as well.

These details would help with change management and control.

## REFERENCE MATERIALS

This section lists all documents referenced and used to develop this project and its ideas.

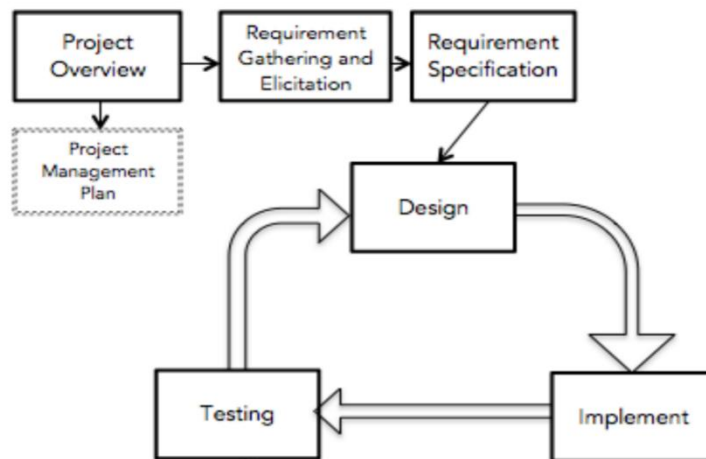
- "CMMI for Development, Version 1.3". *CMMI-DEV (Version 1.3, November 2010)*. Carnegie Mellon University Software Engineering Institute. 2010.
- IEEE Standard for Software Project Management Plans," in *IEEE Std 1058-1998*, vol., no., pp.1-28, 22 Dec. 1998, DOI: 10.1109/IEEESTD.1998.88822.
- IEEE Guide for Software Quality Assurance Planning," in *IEEE Std 730.1-1995*, vol., no., pp.1-20, 10 April 1996, DOI: 10.1109/IEEESTD.1996.80817.
- "Standard CMMI Appraisal Method for Process Improvement (SCAMPI) A Version 1.2: Method Definition Document" (doc). Carnegie Mellon University Software Engineering Institute. 2006.

## PROJECT ORGANIZATION

This section explains the process model, organizational structure and project responsibilities.

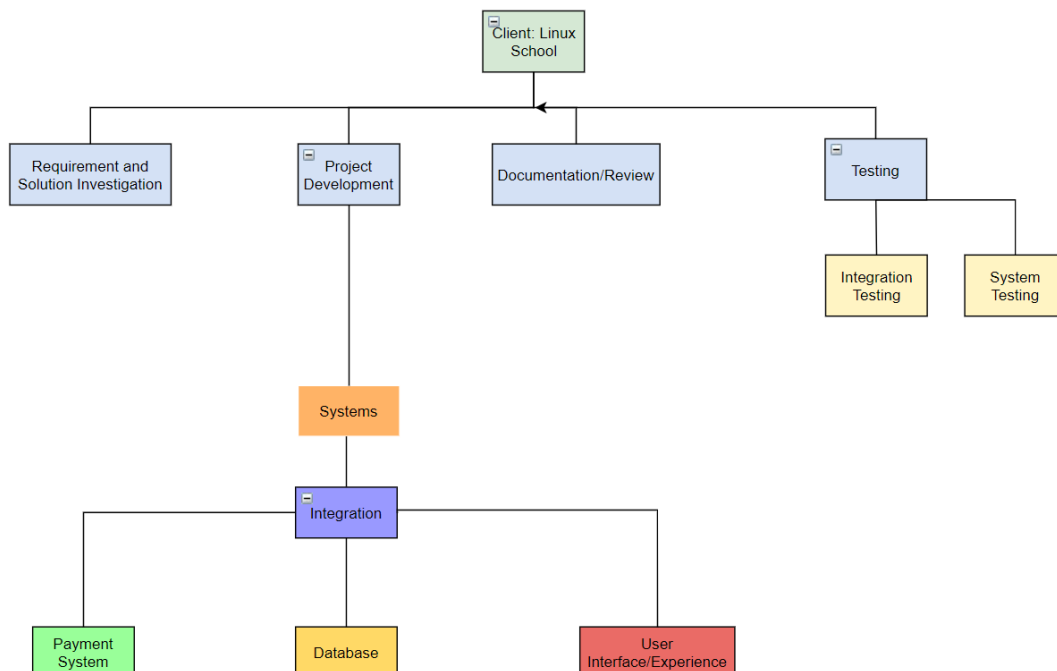
### PROCESS MODEL

The project involves the discovery and statement of requirements. After the requirements have been well defined, the project enters an iterative cycle during design and implementation. This model is adopted to ensure all processes and functionality are correctly implemented and thoroughly tested as feedback is constantly received from the client.



### PROJECT STRUCTURE

The development of the project will follow the hierarchical structure outlined in the chart below.



## PROJECT RESPONSIBILITIES

The project will be led and managed by a one-man team receiving advising and consultation from Rebecca Broadwater and the client.

The one-man team would be responsible for:

- Software Quality Assurance
- Efficient user interaction and user experience
- Systems integration
- Software Testing and Management

## MANAGERIAL PROCESS

The section specifies the management process of this project.

## OBJECTIVES

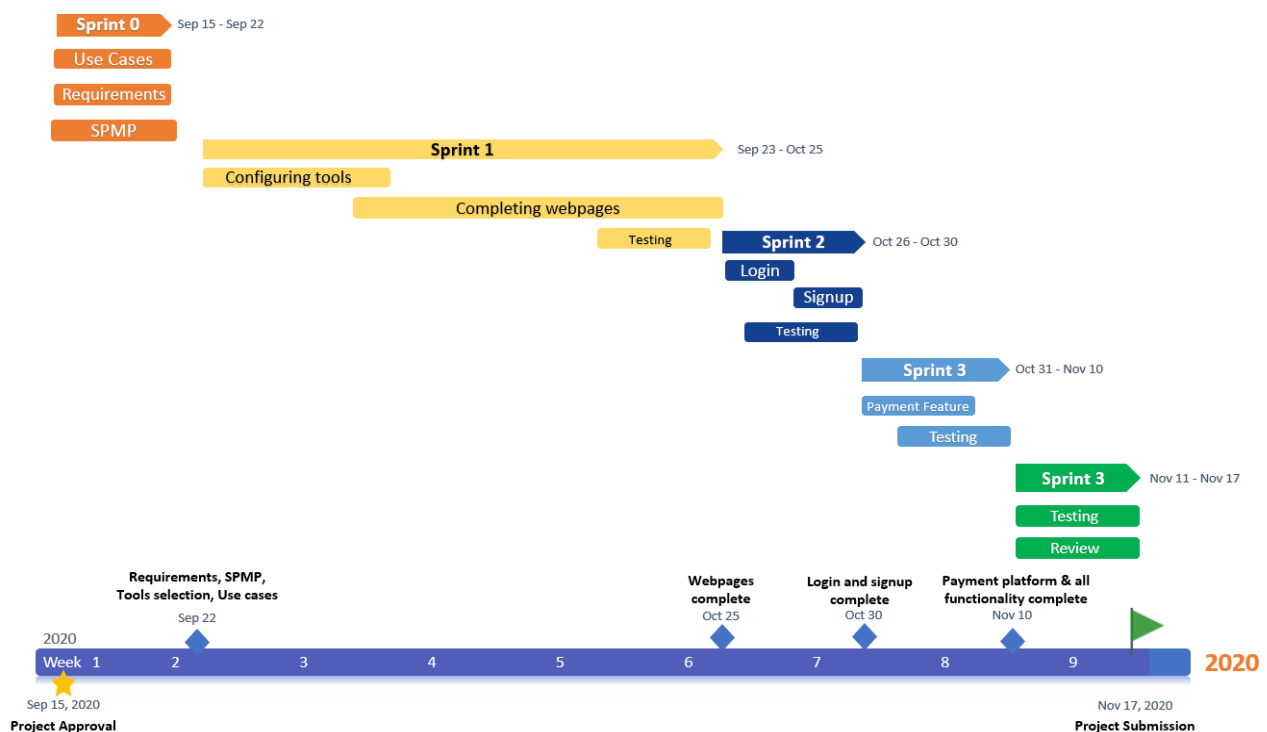
The vision of this project has been well laid in the project overview. The key to the success of this vision is making the consulting and implementation of the website as clear and precise to the client and prospective users of the website.

Also, the schedule would be respected and closely followed without the cutting of corners and more time requested from the client if the time initially given is short to ensure a functional and optimum website is created.

## PRIORITIES

User interaction and core functionality will be of most priority. User registration and logins will be the next priority. After this, the website will then be thoroughly tested before the integration of the payment system.

## SCHEDULE



Domain registration	10/yr
Plugins	50
WordPress Template	20
Hosting	50/yr
TOTAL	130

## PROJECT ANTICIPATIONS

This section describes the various assumptions, dependencies and constraints that are anticipated to occur during the development of this project.

### ASSUMPTIONS

- Timely deliverables
- No significant changes from client feedback
- Dependable hosting platform
- Smooth integration between the CMS and custom web features
- Minimal support for legacy browsers
- Accessibility from any type of mobile device.

### DEPENDENCIES

- MYSQL Database
- Payment System
- Customizable template from WordPress
- SSL encryption for all webpages

### CONSTRAINTS

- Very limited time for project submission.
- Communication of the design and implementation of the website between the developer and non-IT client

## RISK MANAGEMENT

In this section, various risks to the project will be identified and solutions discussed to prevent the project's delay and overspending.

- Significant changes to the project.

Solution: The SPMP and the change management document will define the necessary guidelines to handle all changes and feedback made by the client to update the timeline and project expenses when the need arises.

Also, the client would be informed of how changes made would affect the project delivery.

- **Missed deadlines**

Solution: The development team would strive to be on time with submissions and deliverables but in cases of emergencies or when a deliverable cannot be delivered on time, a notice would be sent to the client and respective shareholders at least 24 hours before the due submission date.

- **Unbearable increase in expenses**

Solution: As the project is being built, other technologies would be constantly monitored to ensure that the client is not overspending for features and services they could get at an affordable price.

- **The scope of the project becomes larger than planned**

The core functionality and requirements will always be prioritized first and if there is more time and resources, the client would be consulted as to which extra features they want added.

## MONITORING AND CONTROLLING MECHANISM

GitHub would be used to help with monitoring change control and monitoring of software and features added to the system. Git enables the tracking of files and code for specific changes. Hence enabling the easy detection of problems with code since it provides tracks all modifications made that might have resulted in error(s).

## TECHNICAL PROCESS

This section specifies the technical methods, tools, and techniques to be used on the project. It also includes software quality assurance, and configuration management.

### DEVELOPMENT TOOLS

#### Operating System

- Linux

#### Platforms



- Database - MySQL
- Web Server - Apache

### Development

- Languages – PHP, HTML, JavaScript

### Testing

- Integration Testing – TBD
- Systems Testing - TBD

### Configuration Management

- Source Code Management (SCM) – Git
- Build Automation – TBD
- Continuous Integration – TBD
- Defect Tracking – Redmine
- Code Review – TBD

## DOCUMENTATION PLAN

All modifications to the project will be reflected in the SPMP and other relevant documents. These documents will be available in the GitHub repository.

## QUALITY ASSURANCE

- Ensure that all tools and software are up-to-date and are from trusted vendors.
- A thorough software testing approach.
- Bugs and security flaws will be fixed and patched as the application is developed.
- Ensure information is communicated as clearly as possible to the client.

## CONFIGURATION MANAGEMENT PLAN

- All documentation relating to tools and features used and the ones made for the project will be safely kept enabling easy troubleshooting.
- The use of sandboxing to manage the production and test runs of the website.
- All updates made will be stated in documentations
- Proper documentation of various changes requested by the client through feedbacks to enhance change management.
- Designing a plan for how problems would be resolved when they arise.
- Management of configuration settings of various tools used.
- Document project lifecycle.

## VERIFICATION & VALIDATION PLAN

- Testing will be implemented at every step of the design and implementation of the project.
- Appropriate time will be specifically allocated for only testing and this period will be prioritized.
- All tests will be documented to include issues and fixes to be made.

## WORK BREAKDOWN STRUCTURE

