# Software Project Management Plan

# The Linux School Website

# VERSION 1.1

JEFFREY AMPEH

# TABLE OF CONTENTS

# INTRODUCTION

## PROJECT OVERVIEW

The project involves the creation of a functional website for a Linux training school. The website would not only serve as online presence and marketing but consist of a registration and payment platform for users interested in enrolling in the school. The website would have a blog to inform users, alumni and professionals share their ideas and answer any questions they might have. The website would also have a newsfeed that would inform users daily on 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 average salaries they could earn.

## PROJECT DELIVERABLES

| WEEK | DESCRIPTION | DATE |
|------|-------------|------|
| 1 | Design Review Meeting | TBA |
| 2 | Weekly Check-ins | 10/13/2020 |
| 3 | Weekly Check-ins | 10/20/2020 |
| 4 | Weekly Check-ins | 10/27/2020 |
| 5 | Weekly Check-ins | 11/03/2020 |
| 6 | Weekly Check-ins | 11/10/2020 |
| 7 | Working Prototype | 11/17/2020 |
| 8 | Presentation | 12/08/2020 |

## EVOLUTION OF THE SPMP

As I work on the project, 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 them or not.

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

The details would help in change management and control.

## REFERENCE MATERIALS

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

- "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, interface 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 is efficiently and correctly implemented and thoroughly tested as we constantly receive feedback from the client. Also, this model enables flexibility.
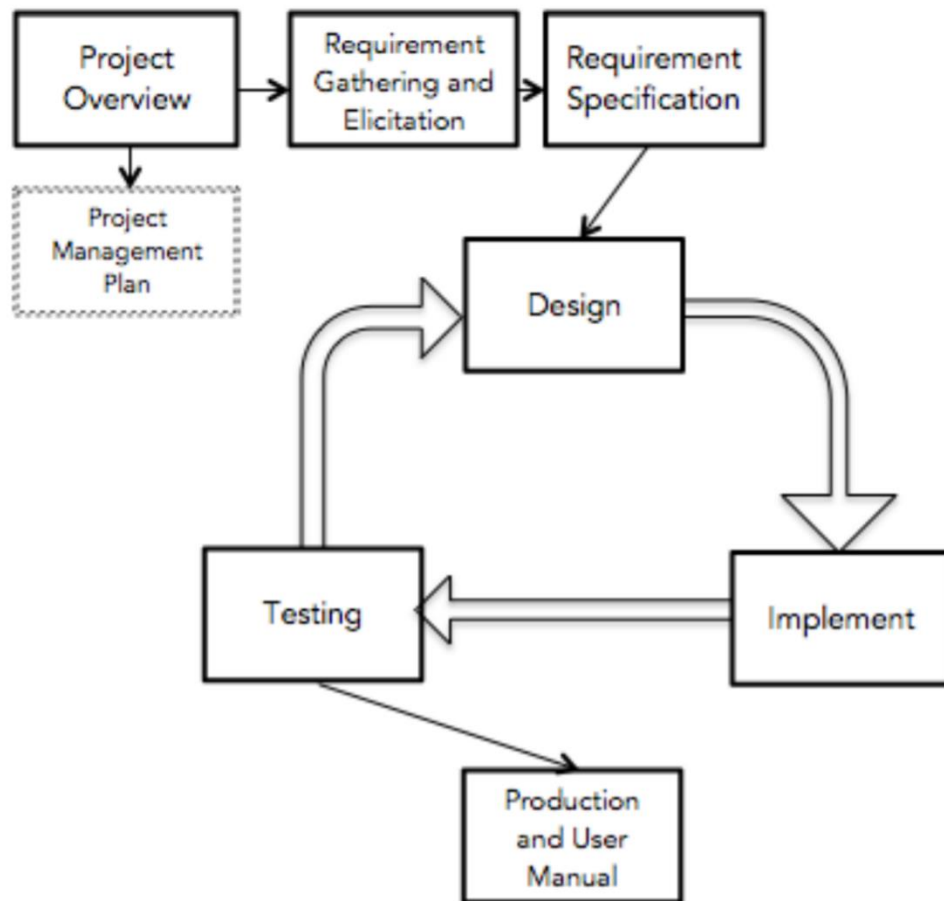
## ORGANIZATIONAL STRUCTURE

The development of the project will follow the hierarchical structure outlined by the chart below.
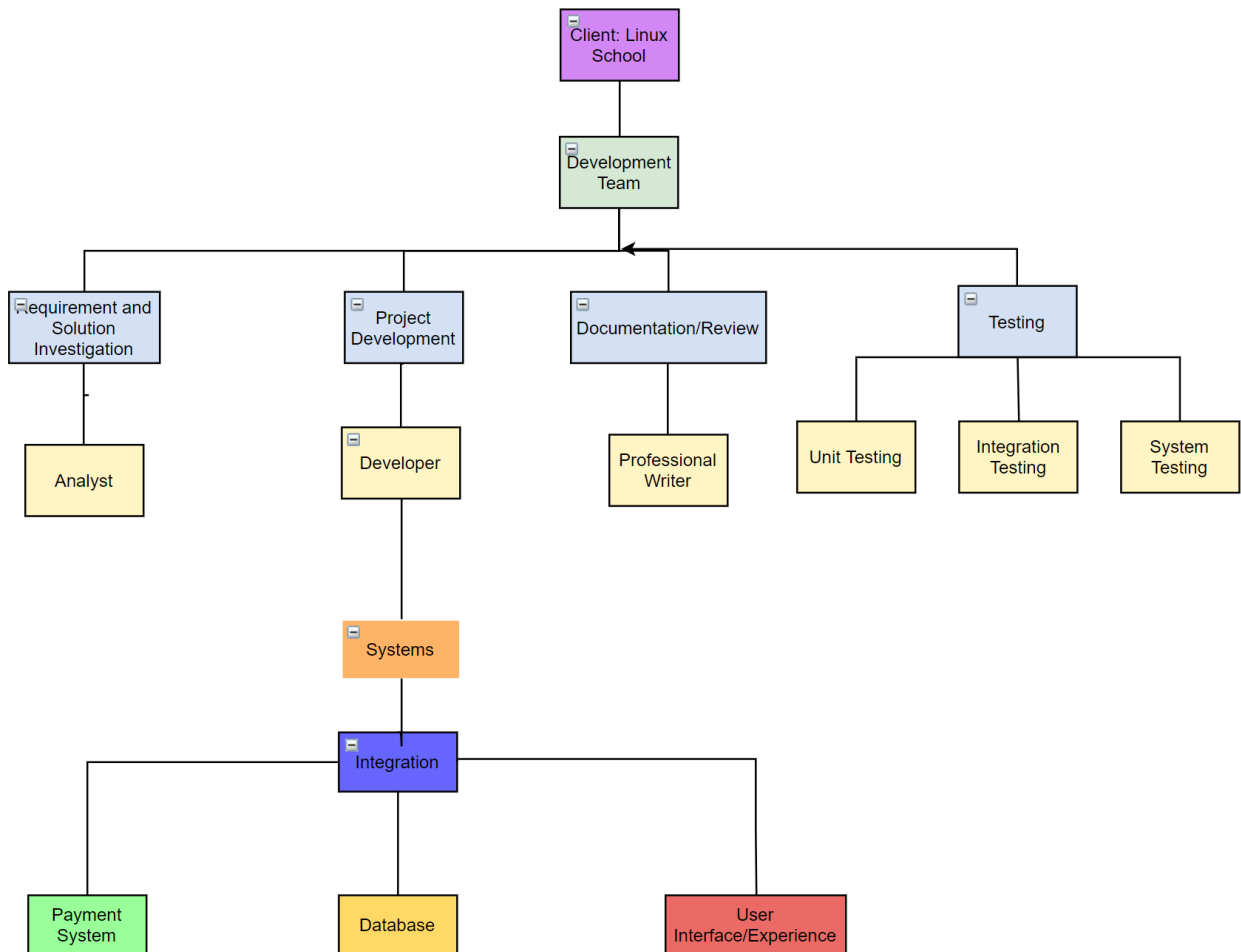
## ORGANIZATIONAL INTERFACES

The website will consist of various aspects integrated into a content management system (CMS). WordPress would be used as the website's CMS. It will integrate the user interface, blog, user registration and payment system to users. The website would be hosted by a third-party hosting platform to allow users access it through the internet.

## 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 team would be responsible for:

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

# MANAGERIAL PROCESS

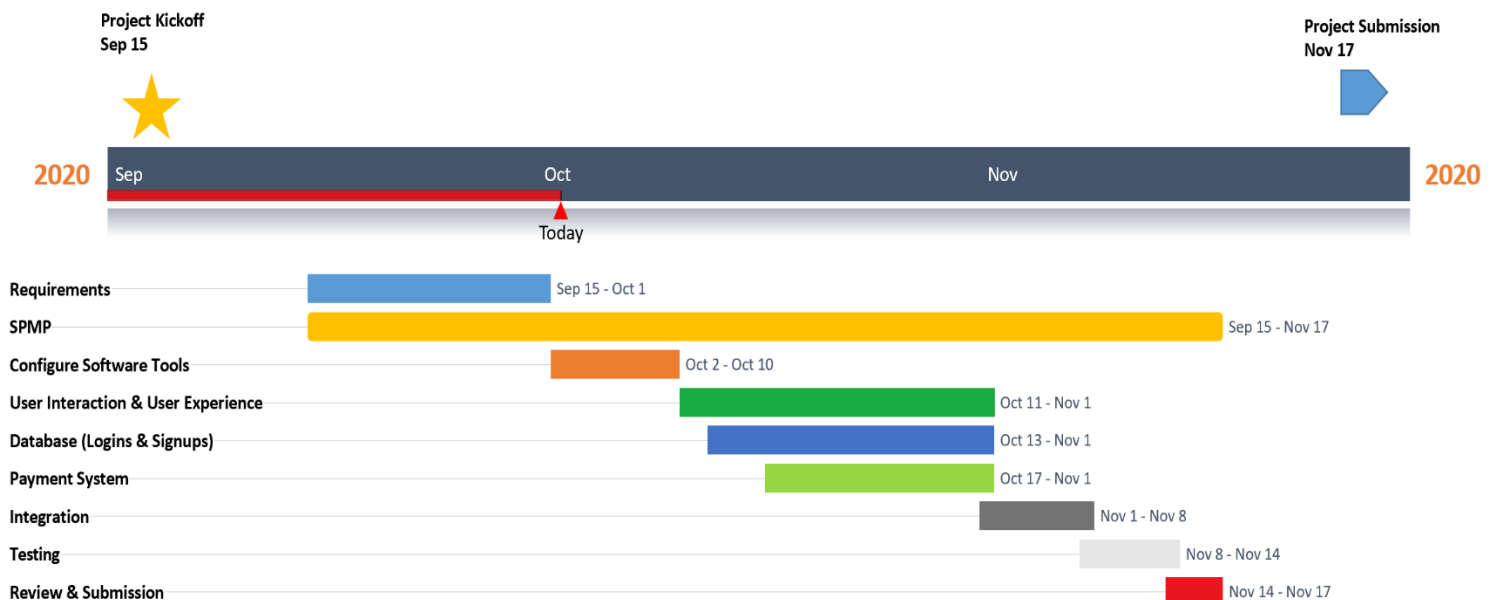The section specifies the management process of this project.

## 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 as possible to the client, prospective users of the website an reduce as much as possible to use of technical jargons and ambiguity.

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

Project Kickoff
Sep 15

Project Submission
Nov 17

| | | | |
|---|---|---|---|
| **2020** | Sep | Oct | Nov | **2020** |

Today

| Requirements | Sep 15 - Oct 1 |
| SPMP | Sep 15 - Nov 17 |
| Configure Software Tools | Oct 2 - Oct 10 |
| User Interaction & User Experience | Oct 11 - Nov 1 |
| Database (Logins & Signups) | Oct 13 - Nov 1 |
| Payment System | Oct 17 - Nov 1 |
| Integration | Nov 1 - Nov 8 |
| Testing | Nov 8 - Nov 14 |
| Review & Submission | Nov 14 - Nov 17 |

## 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
- Full encryption on webpages that does not slow performance
- 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.
- Virtual collaboration between client, testers and developer
- 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 necessary guidelines to handle all changes and feedback made by the client in order to update the timeline and project expenses when the need arises.

  Also, the client would be clearly 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 time

- ### Unbearable increase in expenses
  Solution: As the project is being built other technologies would be constantly monitored to ensure that the client is overspending on the project for features and services that could get at an affordable price.

- ### 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 left, the client would be consulted as to which extra features they would 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.

# 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.

## METHODOLOGY

This project will use the spiral methodology. This methodology is intended to help manage risk. In spiral model the entire system is no defined in detailed at first. The developers should only define the highest priority features. This type of development relies on developing prototypes and then giving them back to the user for trial. With this feedback the next prototype is created. Define and implement this, then get feedback from users. With this knowledge, it is easy to define and implement more features in smaller chunks, until an acceptable system is delivered.

## TOOLS

- **Wordpress.org**
- **Apple Pay online payment system**
- **GitHub**
- **MySQL Workbench**

## DOCUMENTATION PLAN

**All modifications to our project will be reflected in the SPMP and other relevant documents will be made available and accessible in the GitHub repository.**

## WEBSITE SUPPORT FUNCTIONS

- The support desk contact details for all tools and features used for the project will be kept in case help and advising is needed from these vendors.
- Monitoring tools and SEOs will be connected to the website to track traffic.
- All monitoring tools on the website will be routinely and frequently checked to resolve issues found and those raised by the client or users.

## 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 clear as possible to client.

## CONFIGURATION MANAGEMENT PLAN

- **All documentations relating to tools and features used and the ones made for the project will be safely kept.**
- **The use of different deployment environments for software that has already been tested and for one that has not.**
- **All updates made will be stated in documentations**


## 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 clearly documented to include issues and fixes to be made.

# WORK BREAKDWN STRUCTURE