

Software Project Management Plan

Team 4

December 20, 2019

Team Members

Bonizzato Alessandro

Isoni Davide

Sandrini Mattia

Document Control

Change History

Revision	Change Date	Description of changes
V1.0	24/10/20	Initial release

Document Storage

This document is stored in the project's Github repository at:

https://github.com/aleboniz/MarconiBooks5AiTeam4/blob/master/docs/Software_Project_Management_Plan.md

Document Owner

Alessandro Bonizzato is responsible for developing and maintaining this document.

Table of Contents

1 OVERVIEW

1.1 Purpose and Scope

1.2 Goals and Objectives

1.3 Project Deliverables

1.4 Assumptions and Constraints

1.4.1 Assumptions

1.4.2 Definitions

2 STARTUP PLAN

2.1 Team Organization

2.2 Project Communications

2.3 Technical Process

2.4 Tools

3 WORK PLAN

3.1 Release Plan

4 CONTROL PLAN

4.1 Configuration Management Plan

5 SUPPORTING PROCESS PLANS

5.1 Risk Management Plan

5.2 Test Plan

5.3 Product Acceptance Plan

1 Overview

1.1 Purpose and Scope

Team 4 is interested in creating a mobile application that will be useful to students and give Marconi visibility. The Marconi Books platform accomplishes this by giving Marconi's students a way to keep track of school books information.

The Marconi Books website will provide users the ability to log in to their account via the Marconi Books website. Two features will allow users to see a list of disponible books and to add their books.

The user interface will be intuitive and simple to navigate. A user guide and system administration manual will be provided.

1.2 Goals and Objectives

The overall objective is to give students an alternative way to buy and sell school books.

Project Goals:

Create a website that functions as expected, looks great.

Learn about software engineering and creating a website.

Project Objectives:

Create a website interface to give students access to their account information.

Create a website that functions in a simple and intuitive manner.

1.3 Project Deliverables

24/10/2019 - Project Charter Approved
24/10/2019 - Preliminary Requirements Complete
31/10/2019 - Activity planning
31/10/2019 - Create Initial Design of Website
14/11/2019 - Create Database to store users account and books
14/11/2019 - Develop login and logout feature
21/11/2019 - Develop Add Section
28/11/2019 - Develop Search Section
05/12/2019 - Develop User Section
12/12/2019 - System Test Complete
19/12/2019 - Product Released

1.4 Assumptions and Constraints

1.4.1 Assumptions

Marconi Books is available and function when we need them.

1.5 Definitions

Term	Definition
Actor	user or other software system that receives value from a user case.
Baselined	the work product has undergone a formal review and can only be changed through the prescribed change control procedures
Client or Customer	the person or organization for which this Roo Balance application is being built.
Developer	the person or organization developing the system, also sometimes called the supplier.
Project	activities that will lead to the production of the Roo Balance application. ** **
Roo Balance Application	the product that is being described here; the software system specified in this document.
Scenario	one path through a user case
Stakeholder	anyone with an interest in the project and its outcomes. This includes clients, customers, users, developers, testers, managers and executives.
User	the person or persons who will actually interact with the Roo Balance application.
Use case	describes a goal-oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.

2 Startup Plan

2.1 Team Organization

Role	Actor(s)	Responsibility
Project Manager	Davide	Call team meetings, coordinate communications within group, coordinate communications outside group, break out tasks, assign them to teammates
Developer	Davide, Alessandro, Mattia	Develop software based on requirement and architect specifications
Programmer	Davide, Alessandro, Mattia	Program to requirement and architect specifications
Tester	Davide	Write test cases, perform unit testing of test cases against incremental release of code, perform integrated testing of test cases against incremental release of code, report issues
Architect	Alessandro	Specify overall internal workings of application
Requirement Engineer	Mattia	Outline and document project dependencies and requirements. This includes internal and external dependencies.

2.2 Project Communications

Event	Information	Audience	Format	Frequency
Team Meeting	Task status: completed since last meeting & planned for next; obstacles encountered; change requests in process	All team members	Informal meetings following class; Formal meetings as needed; E-mail status updates & problems as they occur	As needed
Project Status Report	Review finished items, status of prototype; review any problems, schedule slippage, programming issues	All team members, customer	E-mail with information or In-person as customer sees fit	Iteration Closeout

2.3 Technical Process

An iterative and incremental development process is planned. Feedback will be used from each iteration to improve the next. The first iteration will focus on basic functionality of the application. Subsequent iterations will build upon that and incorporate more features as time allows.

2.4 Tools

- Programming & Markup Languages – HTML, PHP, MySQL
- Operating System – Windows, Ubuntu
- Version Control – all work products will be stored in an GitHub repository

3 Work Plan

3.1 Release Plan

Iteration #1

Objective: The purpose of this iteration is to understand system requirements well enough to schedule requirements and timing for the project.

Features / Activities / Deliverables	Estimated Effort	Actual Effort
System use cases in summary form	10 hrs	12 hrs
Welcome screen	4 hrs	3 hrs
UI prototype	4 hrs	4 hrs
Validate prototype with users	4 hrs	2 hrs
Totals:	22 hrs	21 hrs

Iteration #2

Objective: The purpose of this iteration is to gain a detailed understanding of critical requirements, mitigate major architectural risks and create the website design.

Features / Activities / Deliverables	EstimatedEffort	Actual Effort
Website design	2 hrs	3 hrs
Database design	2 hrs	1 hrs
Baseline system architecture	2 hrs	3 hrs
Detailed descriptions of critical use cases	5 hrs	6 hrs
Totals:	11 hrs	13 hrs

Iteration #3

Objective: The purpose of this iteration is to implement basic server-side functionality in order to validate architecture. It is necessary to learn PHP language.

Features / Activities / Deliverables	EstimatedEffort	Actual Effort
Login screen to access the website.	1 hrs	3 hrs

Register screen to create new account. Features / Activities / Deliverables	EstimatedEffort	Actual Effort
Logout feature to go out.	1 hrs	1 hrs
Total:	3 hrs	5 hrs

Iteration #4

Objective: The purpose of this iteration is to implement basic client-side functionality in order to allow books search and books add. It is necessary to use PHP language.

Features / Activities / Deliverables	EstimatedEffort	Actual Effort
Basic search screen. It provides a form that prompts for books search criteria. In this initial version of the form, the user will be able to specify only class and subject. Upon entering class and subject a list of books matching the search criteria will be displayed by clicking a button. Before the form completed all books are showed.	4 hrs	6 hrs
Basic add screen. It provides a form that prompts for books add criteria. The user must specific the class, the subject, the name, the code and the price of the book.By pressing the submit button the book will be added.	4 hrs	3 hrs
Total:	8 hrs	9 hrs

Iteration #5

Objective: The purpose of this iteration is to implement basic client-side functionality in order to allow books search and books add. It is necessary to use PHP language.

Features / Activities / Deliverables	EstimatedEffort	Actual Effort
Basic user screen. Ab personal area dedicated to the logged user that show user data and books. It is used to show personal data and books of other users.	4 hrs	3 hrs
Total:	4 hrs	3 hrs

Iteration #6

Objective: The purpose of this iteration is to defect repair and to stabilize the project running.

Features / Activities / Deliverables	EstimatedEffort	Actual Effort
Resolve problems.	2 hrs	4 hrs
Total:	2 hrs	4 hrs

Iteration #7

Objective: The purpose of this iteration is to definitively check and test the product.

Features / Activities / Deliverables	Estimated Effort	Actual Effort
Test project.	1 hrs	1 hrs
Total:	1 hrs	1 hrs

Features not scheduled but under consideration

Feature	Effort Estimate
Online payment. Enable payment feature to allow user to buy directly books on the website.	10 hrs
Chat. Implement an online Marconi books' chat between users. It could guarantee more secure maintaining private phone number and mail.	15 hrs
Email. Implement an online system to send email to the administrators in the Contact page.	5 hrs

4 Control Plan

4.1 Configuration Management Plan

The following procedure is to be used when making changes to all baselined work products:

- All project work products will be stored in a GitHub repository running on a central server.
- All baselined documents will have a history to track initialization and subsequent changes.
- All project work products (documents, source code, test cases, program data, test data, etc) will be stored in the GitHub repository but not all will be under change control (subject to formal change control procedures.) Only the system requirements, project plan and source code will be baselined and under configuration control.
- Items that are subject to change control will be considered baselined after a group review at the end of the initial document creation.

The change control procedure once a product is baselined is:

- anyone wanting to make a change to a baselined item sends an email to the rest of the team and project sponsor describing the change, reason for the change, expected schedule impact, and timeline for integrating the change.
- if no one responds to the group within 2 days with a reason for why the change request shouldn't be permitted, it will be considered accepted and the person proposing the change may proceed with the change.
- if anyone does object to the change, the reason for objecting will be discussed at a meeting where

everyone is invited to attend and voice their opinion. At the end of the meeting a democratic vote will be held to decide whether or not the change should be allowed.

- if a change takes place, the initiator must collaborate with the project manager to update the schedule

5 Supporting Process Plans

5.1 Risk Management Plan

Rank	Risk	Probability of Loss	Size of Loss	Risk Exposure	Response
1	Schedule / time line delivery	Likely	Major	High	Mitigate: Stick to the schedule.
2	Logistic problems	Likely	Moderate	Moderate	Follow the program and organize materials and tools.
3	Learning curve for new tools and technologies longer than expected	Unlikely	Moderate	Moderate	Buy Information: Begin working on a basic prototype early to test out fundamental programming concepts & knowledge

5.2 Test Plan

The test plan defines the items that will be tested, methods for testing, and a schedule detailing the tasks, owners, and timeline.

5.3 Product Acceptance Plan

At the conclusion of each iteration, the prototype created will be tested to ensure it meets the requirements of that iteration. An environment such as the iPhone simulator may be used to test functionality in lieu of the app being loaded on an actual smartphone.

For the final iteration, product acceptance testing will ensure that the prototype functions as expected with a user's data.