ARGO Testing Scanner

Project Management Plan

Team 5

Developers

Amanda Bosson

Matthew Freese

Jessica Jennings

Mazen Lawand

Hisham Nabi

Abstract

ARGO has assigned Team 5 the Testing Scanner application project. The main goals for this project are to detect events, determine the source users tied to the event, notify the user, and capture all of the trace and log data specific to the event recorded on the log. The point of this application is to help testers track any unwanted events that might not be immediately evident during their black box testing.

Table of Contents

[**1. Introduction**](#_ywpnq8m9u7o) **3**

[1.1 Introduction to Project Management Plan](#_upkd5stzzqgn) 3

[1.2 Purpose of Project Management Plan](#_ba3lw23so28c) 3

[1.3 Product Overview](#_daf2ao8jdli4) 3

[1.4 Plan Structure Description](#_8rhm24n63804) 3

[**2. Project Organization**](#_3fejdyrn6his) **3**

[2.1 Development Team Organization and Roles](#_sbc0acdsxp5o) 3

[**3. Lifecycle Model**](#_6n3p8ljebktp) **4**

[3.1 Lifecycle Model](#_93iij5n60yn8) 4

[**4. Risk Analysis**](#_p73vm3q1yx3s) **5**

[4.1 Project Risks, Impact, and Likelihood](#_zcwzb75uhvfz) 5

[4.2 Risk Reduction Strategy](#_pxy9qtwty9s2) 5

[**5. Hardware & Software Resources Requirements**](#_sayqrss2mrdy) **6**

[5.1 Software Resources](#_nn83qi1ufebk) 6

[**6. Deliverables and Schedule**](#_wq13y5do2jlq) **6**

[6.1 Schedule](#_hbx49mnpz69q) 6

[**7. Monitoring, Reporting, and Controlling Mechanisms**](#_m2dm841ja3ci) **6**

[**8. Professional Standards**](#_agp77mlqedm5) **7**

[8.1 Team Expectations](#_4iosjevuhq7z) 7

[8.1.1 Schedule & Deliverables](#_d802jd5x78tj) 7

[8.1.2 Work Quality](#_qa22igl57dpf) 7

[8.1.3 Scholastic Dishonesty](#_3az7r7h2mp5k) 7

[8.1.4 Professional Misconduct](#_2pu7ekf3bv06) 7

[**9. Configuration Management Evidence**](#_yhi3yypzbsc8) **8**

[9.1 Configuration Management Tool](#_y2ycgcwzrn34) 8

[9.2 Evidence](#_yqbz7kdt3rx2) 8

[**10. Reference**](#_69uybdlwwa2l) **8**

# 1. Introduction

## 1.1 Introduction to Project Management Plan

The ARGO Testing Scanner is an application that detects and notifies testers of errors and defects. This document provides a brief overview of the process for planning the ARGO Testing Scanner.

## 1.2 Purpose of Project Management Plan

This plan will be used to describe the technical and social specification for the ARGO Testing Scanner project including project organization, schedules, and deliverables. This document may also be used by future teams as a reference for future implementations and improvements.

## 1.3 Product Overview

The ARGO testing scanner is an application that consists of four major components that will:

* Detect events
* Determine the user responsible for an event
* Alert the user responsible for the event that occurred
* Provide the user related log and trace files

Users will be ARGO testers. The application will be run when the user is doing testing activities.

## 1.4 Plan Structure Description

This document will cover the organization of the team, the lifecycle of the project, risk analysis, deliverables and deadlines, monitoring and reporting mechanisms, and professional standards.

# 2. Project Organization

## 2.1 Development Team Organization and Roles

Project Manager - Mazen Lawand

The role of project manager includes ensuring project deadlines are met, turning in deliverables, and coordinating meetings

Documentation Lead - Amanda Bosson

The documentation lead will ensure that documents are kept up to date and have consistent formatting.

Architect - Matthew Freese

The architect will create the architecture of the application and will ensure that individual team members are informed about the application’s architecture and ensure adherence to the architecture.

UI Lead - Jessica Jennings

UI lead will design the presentation of the program. Responsibilities will be to ensure notifications are passed correctly to the users with an appropriate design.

Testing - Hisham Nabi

The tester will find and report any bugs found within the code. The tester will also test the program for performance issues and operability.

Roles were divided based on the following rationale. Each member of the team took ownership of a role based on past experience and passion.

# 3. Lifecycle Model

## 3.1 Lifecycle Model

The ARGO Testing Scanner will be developed using the spiral model. The spiral model is defined as “a systems development lifecycle (SDLC) model used in information technology (IT). This model of development combines the features of the prototyping model and the waterfall model” [1].

The spiral lifecycle was chosen because of the many advantages it has over other lifecycle models such as the waterfall method. Some of the advantages that were determined to be important to the project include:

* Early software production
* Early risk analysis
* Ability to add additional functionality

# 

# 

# 4. Risk Analysis

Due to the size and complexity of the project, there are several factors that may lead to project failure. This section details what possible risks may exist and what risk analysis strategies will be used.

## 4.1 Project Risks, Impact, and Likelihood

The table below illustrates possible risks, impacts, and likelihood.

|  |  |  |
| --- | --- | --- |
| **Risk** | **Impact** | **Likelihood** |
| **Changing Requirements** | **Medium** | **High** |
| **Insufficient Documentation** | **Low** | **Medium** |
| **Miscommunication between developers and sponsors** | **High** | **Medium** |
| **Differences between product and vision** | **High** | **Low** |

## 4.2 Risk Reduction Strategy

The project management plan proposes a five step plan for risk reduction:

1. Identify all possible project risks

2. Evaluate the probability of each risk

3. Determine the impact if a risk was to occur

4. Prioritize all risks based on probability and impact

5. Control and mitigate risks

# 

# 5. Hardware & Software Resources Requirements

## 5.1 Software Resources

The team will use Java to program the application. Java was chosen based on the reasoning that there will be a large number of xml files that need to be parsed and searched through and Java has good built in xml support, the application will be more valuable if it can easily be run on multiple operating systems, and the development team is most comfortable with Java.

The team will also need access to sample log and trace files from ARGO.

# 6. Deliverables and Schedule

## 6.1 Schedule

The deadline for all assignments can be seen below. Because of the intricate nature of software development, future assignments are dependent on the completion of prior assignments. All team members will be responsible for completing their portion of the assignment.

|  |  |
| --- | --- |
| * Project Management Plan * Requirements Documentation * Architecture Documentation * Detailed Design Documentation * Testing Plan * Final Project Report | Due: 01/27/2017  Due: 02/10/2017  Due: 02/24/2017  Due: 03/17/2017  Due: 04/07/2017  Due: 04/28/2017 |

# 7. Monitoring, Reporting, and Controlling Mechanisms

For every meeting, we will scribe that will write down what has been talked about and decided on. The end of the document will have what each team member(s) is responsible of doing for the week. We have a document on the Google Drive that we will use in order to write down any questions we have for our sponsor. Meetings will be held weekly on Wednesdays at 4pm.

Team members are communicating and collaborating via Slack. All documents are stored on the Google Drive. Communication with the TAs and the professors will include all team members via the “reply all” option. Team mates schedules and meeting times are agreed on what has been recorded on our Doodle poll.

# 8. Professional Standards

## 8.1 Team Expectations

Every team member is responsible for taking care of their specific responsibility and collaborating with the rest of the team to get their assigned goal done in a timely fashion. Team members are also expected to work with their peers in order to help them achieve a goal based on their needs. Team members are also expected to participate in team meetings along with corporate sponsor meetings.

## 8.1.1 Schedule & Deliverables

All team members are expected to submit their portion of the assignments by the deadline set by the team leader. Assignments are to be turned in by the deadlines set by the instructor.

If a Team Member is unable to complete his/her work before the deadline, they are required to notify the Team Leader at least 3 days before the deadline. Team members who do not send notice will be subjected to the professional misconduct protocol.

## 8.1.2 Work Quality

To ensure that all product goals and objectives are met, team members must produce quality work. Work quality is gauged on both meeting deliverables schedule and also meeting all requirements given. Team members who do not contribute quality work on time may be subjected to professional misconduct review.

## 8.1.3 Scholastic Dishonesty

Students are expected to follow the University of Texas at Dallas’s guidelines concerning scholastic dishonesty[utdallas.edu/conduct/dishonesty]. Any team member suspected of scholastic dishonesty will be subjected to the professional misconduct protocol found in section 8.1.4.

## 8.1.4 Professional Misconduct

Examples of misconduct include missed deadlines and team meetings without prior notice, poor work quality, scholastic dishonesty, and disrespectful behavior. Any misconduct will be

handled as follow:

Offense #1: The team leader will be notified of the unacceptable behavior. The team leader will determine the circumstances surrounding the misconduct and resolve the problem.

Offense #2: The instructor will be notified of the unacceptable behavior. A meeting will be set up to resolve the problem.

Offense #3: The instructor will again be notified of the unacceptable behavior. A meeting will be set up to resolve the problem. At this time,

the team will have the option of voting to remove the offending team member.

# 9. Configuration Management Evidence

## 9.1 Configuration Management Tool

This project will use GitHub as a version control tool in order to keep track of configuration management.

## 9.2 Evidence

GitHub keeps track of the changes to the software system by logging changes and when they were pushed to the main repository for the project. A link to this repository can be provided after it is created.

# 

# 10. Reference

[1]Rouse, Margaret. "What Is Spiral Model (spiral Lifecycle Model) ? - Definition from WhatIs.com." SearchSoftwareQuality. TechTarget, Mar. 2007. Web. 25 Jan. 2017.