**TEST PLAN**

Table of Contents

[Introduction 3](#_Toc109387551)

[1.1 Objectives 3](#_Toc109387552)

[1.2 Team Members 3](#_Toc109387553)

[2 Scope 3](#_Toc109387554)

[3 Assumptions / Risks 3](#_Toc109387555)

[3.1 Assumptions 3](#_Toc109387556)

[3.2 Risks 4](#_Toc109387557)

[4 Test tools 4](#_Toc109387560)

[5 Test environment 4](#_Toc109387561)

# Introduction

The Test Plan has been created to communicate the test approach to team members. It includes the objectives, scope, schedule, risks and approach. This document will clearly identify what the test deliverables will be and what is deemed in and out of scope.

## Objectives

Coffee Maker project, is an application developed in java. This application aims to facilitate servicing orders process in Caffes. The project is intended to initiate the orders in terms of (name, amount of (coffee, milk, sugar, and chocolate) and the price. The user will be able to make changes to an existed order, by providing edit and delete options. The test team is responsible for testing the product and ensuring it meets their needs.

Phase 1 of the project will deliver TCCM (Test Case Coffee Maker) with functionality to create and store manual tests. This will allow the test team to start transferring tests over to the new system.

## Team Members

|  |  |
| --- | --- |
| **Resource Name** | **Role** |
| Diana Salameh | Testing Engineer |
|  |  |
|  |  |

# Scope

The initial phase will include all ‘must have’ requirements. These and any other requirements that get included must all be tested. At the end of Phase 1, a tester must be able to:

1. Create (initiate) a new recipe.
2. Set the required values for each recipe.
3. Add the recipe to the recipe book.
4. Retrieve it and have the ability to view it when running the test.
5. Edit / Delete a specific recipe.
6. Compare the equality of two recipes.
7. Retrieve all recipes in the recipe book.

# 

# Assumptions / Risks

## Assumptions

This section lists assumptions that are made specific to this project.

1. Delivery of the values of the recipe ingredients from the user I/O as Strings.

## Risks

The following risks have been identified and the appropriate action identified to mitigate their impact on the project. The impact (or severity) of the risk is based on how the project would be affected if the risk was triggered. The trigger is what milestone or event would cause the risk to become an issue to be dealt with.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Risk | Impact | Trigger | Mitigation Plan |
| 1 | Changes to the functionality may negate the tests already written and we may lose test cases already written | High – to schedule and quality | Loss of all test cases | Export data prior to any upgrade, massage as necessary and re-import after upgrade. |

# Test Approach

Automated unit tests are part of the development process, using JUnit to validate and cover the project functionalities:

* Add a new recipe.
* Set the related recipe values.
* Retrieve the recipe with its properties.
* Add the recipe to the Recipe Book.
* Update (edit / delete) an existed recipe.
* Retrieve all recipes in the Recipe Book.

# Test Tools

Eclipse IDE.

JUnit library.

# Test Environment

Laptop with OS (widows, macOS, Linux, ...).