**What is a Test Plan ??**

Test Plan Template is a detailed document that describes the test strategy, objectives, schedule, estimation and deliverables, and resources required for testing.

**Why do we need it ??**

* Test Plan helps us determine the effort needed to validate the quality of the application under test.
* Easy to monitor and control by the test manager.

**TEST PLAN**

**Table of Contents**

[**INTRODUCTION 3**](#_heading=h.30j0zll)

[1.1](#_heading=h.1fob9te) Objectives 3

[1.2](#_heading=h.3znysh7) Team Members 3

[**2**](#_heading=h.2jxsxqh) **SCOPE 3**

[**3**](#_heading=h.z337ya) **ASSUMPTIONS / RISKS 4**

[3.1](#_heading=h.1t3h5sf) Assumptions 4

[3.2](#_heading=h.17dp8vu) Risks 4

[**4**](#_heading=h.3rdcrjn) **TEST APPROACH 4**

[4.1](#_heading=h.26in1rg) Test Automation 4

[**5**](#_heading=h.3j2qqm3) **TEST ENVIRONMENT 5**

[**6**](#_heading=h.35nkun2) **MILESTONES / DELIVERABLES 5**

[6.1](#_heading=h.1y810tw) Test Schedule 5

[6.2](#_heading=h.44sinio) Deliverables 5

**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.

* 1. Objectives

Test Case Tamer is a web-based Test Management tool used to create and store tests as well as the results of running those tests. This tool is a new product written with Ruby on Rails using a ‘mysql’ database. The test team is responsible for testing the product and ensuring it meets their needs. The test team is both the customer and the tester in this project.

Phase 1 of the project will deliver TCT (Test Case Tamer) with functionality to create and store manual tests. This will allow the test team to start transferring tests over to the new system. Must have functionality is considered more important than the delivery date in this project.

* 1. Team Members

| **Resource Name** | **Role** |
| --- | --- |
|  |  |
|  |  |
|  |  |

1. **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 a manual test with as many steps as necessary
2. Save it
3. Retrieve it and have the ability to view it when running the test
4. Enter results and appropriate comments
5. View results

As the team works with the product they will define the needs for the second phase.

Load testing will not be considered part of this project since the user base is known and not an issue.

Rewriting, moving or porting existing test cases from the existing Word documents is not considered part of this project.

1. **Assumptions / Risks**
   1. Assumptions

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

1. Delivery of the product is in format that the test team can check it into CVS.
   1. 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 | Scope Creep – as testers become more familiar with the tool, they will want more functionality | High | Delays in implementation date | Each iteration, functionality will be closely monitored. Priorities will be set and discussed by stakeholders. Since the driver is functionality and not time, it may be necessary to push the date out. |
| 2 | 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. |
| 3 | Weekly delivery is not possible because the developer works off site | Medium | Product did not get delivered on schedule |  |
| 4 |  |  |  |  |

1. **Test Approach**

The project is using an agile approach, with weekly iterations. At the end of each week the requirements identified for that iteration will be delivered to the team and will be tested.

Exploratory testing will play a large part of the testing as the team has never used this type of tool and will be learning as they go. Tests for planned functionality will be created and added to TCT as we get iterations of the product.

* 1. Test Automation

Automated unit tests are part of the development process, but no automated functional tests are planned at this time.

1. **Test Environment**

A new server is required for the web server, the application and the database.

1. **Milestones / Deliverables**

* 1. Test Schedule

The initial test schedule follows……….

| **Task Name** | **Start** | **Finish** | **Effort** | **Comments** |
| --- | --- | --- | --- | --- |
| Test Planning |  |  |  |  |
| Review Requirements documents |  |  | 2 d |  |
| Create initial test estimates |  |  | 1 d |  |
| Staff and train new test resources |  |  |  |  |
| First deploy to QA test environment |  |  |  |  |
| Functional testing – Iteration 1 |  |  |  |  |
| Iteration 2 deploy to QA test environment |  |  |  |  |
| Functional testing – Iteration 2 |  |  |  |  |
| System testing |  |  |  |  |
| Regression testing |  |  |  |  |
| UAT |  |  |  |  |
| Resolution of final defects and final build testing |  |  |  |  |
| Deploy to Staging environment |  |  |  |  |
| Performance testing |  |  |  |  |
| Release to Production |  |  |  |  |

* 1. Deliverables

| **Deliverable** | **For** | **Date / Milestone** |
| --- | --- | --- |
| Test Plan | Project Manager; QA Director; Test Team |  |
| Traceability Matrix | Project Manager; QA Director |  |
| Test Results | Project Manager |  |
| Test Status report | QA Manager, QA Director |  |
|  |  |  |
| Metrics | All team members |  |
|  |  |  |