TEST PLAN

Software Engineering Project 2014

Project ID: WE-SEP-004

Submitted by:

1. IT12088942 – (K. M. K. N. B. Gamhatha)
2. IT12015122 – (H. M. C. I. Gunathunga)
3. IT12009978 (A. N. M. R. H. S. Athurupana)
4. I T12021512(U. D. R. Piumal)

Submitted to:

…………………………..

(Mr. Thilina Halloluwa)

………………………

Date of submission

Project Details

Project Id	WE-SEP-004
Project Name	eMarketting portal
Project Manager	
Test Plan Prepared By	

REVISION HISTORY

Ver. No	Date of Release	Prepared By	Approved By	List of changes from Previous Version

DOCUMENT APPROVAL

Mr. Thilina Halloluwa has reviewed this document and hereby agree that the contents herein are accurate. Any changes to this document must be communicated in writing and signed off by both parties

Signature	Signature
Date:	Date:
Name:	Name:
Client:	Company :

Table of Contents

1. Introduction

This document defines the strategy and plan to test eMarketting portal. The Quality Assurance test plan will use as the basis for all testing activities. The intended audiences for this document are the developers, project management team and the QA team

This document will address the different standards that will apply to the unit, integration and system testing of the specified application. We will utilize testing criteria under the white box, black box, and system-testing paradigm. This paradigm will include, but is not limited to, the testing criteria, methods, and test cases of the overall design.

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.

1.1. Objectives

1.2. Scope

This document related to the testing carried out for the eMarketting portal project Phase 1 by the QA team. This document can be used for project management information. It is the focal point, which will direct the testing effort for the project.

Review and testing activities conducted by non-QA team members, for example, unit testing are not within the scope of this document.

1.3. Reference Material

Document	Version number
Pressman, Roger S. Software Engineering - A Practitioner's Approach. Fifth edition. The McGraw-Hill companies, Inc	5
Kaner, C., Falk, J., Nguyen, H.-Q. Testing Computer Software. Wiley Computer Publishing, 1999.	1

1.4. Definitions, Acronyms, and Terminology

Item	Description/ Expansion
WAMP	Windows Apache MySQL PHP
XAMP	Any OS Apache MySQL PHP
MVC	Model View Controller

2. Assumptions / Risks

2.1. Assumptions

As this product is a website, it must be tested in almost all the major browsers available in the market.

The product has to be tested on major operating systems of in the market such as Microsoft's windows XP, seven and windows 8 and other operating systems such as Mac OS and Linux based operating systems such as Ubuntu and fedora.

Application has to be tested for all the functionalities identified from the user stories that agreed upon with customer for the delivery.

A complete cycle of Regression test has carried out for each major mile stone deliverables.

Using the scrum software development methodology to complete the system in short iterations

Develop the website and host it in the local Apache server for testing

Using software such as XAMP or WAMP server which has integrated web server, database to develop and test the product locally.

Develop the website using the MVC (Model View Controller) design pattern, for the separation of the code logic from the views.

Use a centralized MySQL database.

Responsive User Interface to support multiple screens.

2.2. Risks

- Since the product requirements are still evolving, functionality and content changes could affect the test schedule and lead to rework.

- Test procedure and execution schedule depends on the availability of the build release schedule and delivery plan on critical milestones

#	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	Lack of knowledge regarding the testing process	High	Delays in testing date	Getting familiar with the testing methodologies and getting help from the experts
3				

3. Execution Plan

The product quality is improved by handling and carrying out various types and series of testing levels test types, test case design methodologies and test methodologies

The code quality is measured in standard measures to check whether it satisfy the requirements and according to the acceptable format.

The following represents the overall flow of the testing process:

1. Identify the requirements to be tested. All test cases shall be derived using the current Program Specification.
2. Identify which particular test(s) will be used to test each module.
3. Review the test data and test cases to ensure that the unit has been thoroughly verified and that the test data and test cases are adequate to verify proper operation of the unit.
4. Identify the expected results for each test.
5. Document the test case configuration, test data, and expected results.
6. Perform the test(s).
7. Document the test data, test cases, and test configuration used during the testing process. This information shall be submitted via the Unit/System Test Report.
8. Successful unit testing is required before the unit is eligible for component integration/system testing.
9. Unsuccessful testing requires a Bug Report Form to be generated. This document shall describe the test case, the problem encountered, its possible cause, and the sequence of events that led to the problem. It shall be used as a basis for later technical analysis.
10. Test documents and reports shall be submitted. Any specifications to be reviewed, revised, or updated shall be handled immediately.

Through this process the defects and problems can be identified easily and early in the process. Therefore early feedbacks can be enabled easily and efficiently.

4. Test Approach

We will create a lightweight set of testing guide lines and try to think of weaknesses of the system and the ways that the system can produce errors or unexpected outputs. And we will try to find as many defects as possible during test execution of the functions of our system.

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 main role in testing process and since the team has never used this type of tool, there will be a learning period for them. Tests for planned functionality will be created.

4.1. Test Coverage

Type of Testing	Yes/No
Installation / Smoke Testing	Yes
Functional Testing	Yes
Regression Testing	Yes
UI Testing	Yes
Integration Testing	Yes
Test Automation	No
Performance Testing	No
Stress Testing	Yes

4.2. Test Automation
Automated testing is not planned for this system.

4.3. Non-functional Requirements

Non Functional Requirement	Criticality
Accessible from different devices	Medium
Response times. Speed of page loads and calculations	Medium
Consistent and dependable quality of service	High
Online assets need to be protected	High
Easy to use by target users. Both humans and web crawlers	High
Are there any associated safety risks	High

4.4. Out of Scope

External product impacts are not considered. Not responsible for the bugs that are found in other soft wares that are not developed by our group.

Backend Data Maintenance: WAMP Server/ MySQL backup and recovery.

Undocumented Features: Not responsible for bugs appearing in undocumented or non-maintained areas of the specification

The performance testing is not fallen within scope of testing.

The usability testing and security testing are not fallen within the scope of testing.

4.5. Test Environment

No.	Machine	Purpose	Software Requirements	Hardware Configuration	Quantity
1	Windows Seven	Installation of the web application Database to support the web application	Apache web server 2.4.4 php 5.4 MySQL 5.6 Web browser	2.2Ghz Intel ore i7 Processor RAM 6GB 750GB Hard Disk	1
2	Windows 8	Installation of the web application Database to support the web application	Apache web server 2.4.4 php 5.4 MySQL 5.6 Web browser	2.2Ghz Intel core i7 Processor RAM 6GB 650GB Hard Disk	1

4.6. Test Deliverables

Deliverable	Description
Test Case Doc	Test cases used for testing the functional and non-functional requirements.
Test Report	This summarizes the result of test case execution. It is use to understand progress of testing and the quality of the project.

4.7. Testing Methodology

The following outlines the types of testing that will be done for unit, integration, and system testing. While it includes what will be tested, the specific use cases that determine how the testing is done will be detailed in the Test Design Document.

Test types
- ❖ Functional Testing
    - o Unit Testing
    - o Regression testing
    - o System testing
    - o Stress testing

- ❖ Non-Functional Testing
    - o UI Testing
    - o Integration Testing

Functional Testing

Unit testing

Unit Testing is done at the source or code level for language-specific programming errors such as bad syntax, logic errors, or to test particular functions or code modules. The unit test cases shall be designed to test the validity of the programs correctness.

Entry criteria - The particular unit should be implemented. All test cases completed

Exit criteria - A specified percentage of test cases completed with a percentage containing some number of minor defects

Regression testing

Entry criteria - Unit testing has been done to the particular models. Integration testing has been done to the particular model

Exit criteria - A specified percentage of the particular model completed with a percentage containing some number of minor defects.

System testing

Entry criteria - All the components are implemented and regression testing has completed

Exit criteria - A specified percentage of the system completed with a percentage containing some number of minor defects.

Stress testing

Entry criteria - System should be tested

Exit criteria - A specified percentage of the system completed with a percentage containing some number of minor defects.

4.8. Test Equipment and Tools

No.	Purpose	Tool
1	Configuration management	VSS
2	Defect management	Bug Tracker
3	Project Management	Microsoft Project 2000

4.9. Test Data Requirements

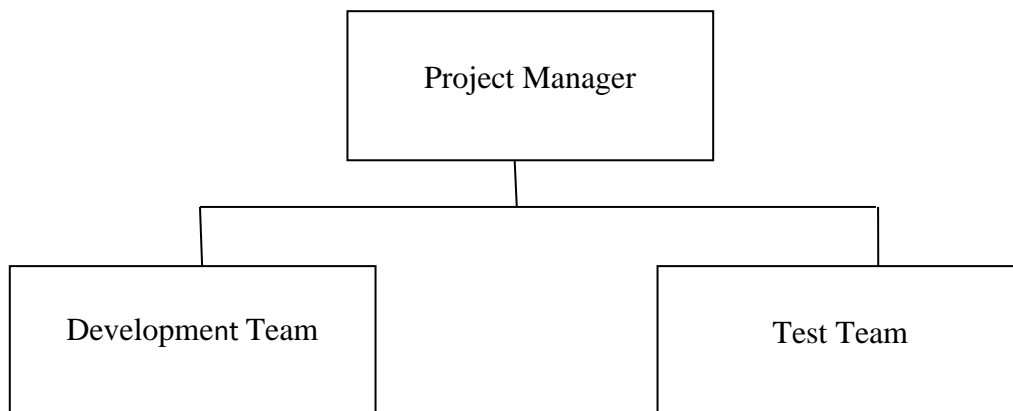Most of the time data is gathered by reading a file.

Test data is prepared and collected from the similar scenarios and done by the testing team.

5. Team structure

5.1. Team Organization

The team will be organized under the project manager. There are two different teams known as development team and the test team. For this project with only four members, both the test team and the development team comprised with same for members.

The test team will be led by the QA Lead and QA Lead has the responsibility to coordinate and monitor the testing process

```
            ┌─────────────────────┐
            │                     │
            │   Project Manager   │
            │                     │
            └─────────────────────┘
                       │
          ┌────────────┴────────────┐
┌──────────────────┐        ┌──────────────────┐
│                  │        │                  │
│ Development Team │        │    Test Team     │
│                  │        │                  │
└──────────────────┘        └──────────────────┘
```

5.2. Roles and Responsibilities

The Test Team will work closely with the Development Team to achieve a high quality design and user interface specifications based on customer requirements. The Test Team is responsible for visualizing test cases and raising quality issues and concerns during meetings to address issues early enough in the development cycle.

The Test Team will work closely with Development Team to determine whether or not the application meets standards for completeness. If an area is not acceptable for testing, the code complete date will be pushed out, giving the developers additional time to stabilize the area.

Since the application interacts with a back-end system component, the Test Team will need to include a plan for integration testing. Integration testing must be executed successfully prior to system testing.

Role	Name(s)	Responsibilities
QA Lead	A. N. M. R. H. S. Athurupana H. M. C. I. Gunathunga	Set realistic expectations on timelines /deliveries Review all QA deliverables for conformity to quality and standards Collect, analyze, monitor and present QA metrics Ensure availability of Test Environment Defect management
Development Team	K. M. K. N. B. Gamhatha U. D. R. Piumal	Code development Finding errors Testing modules that have been developed

6. Test Execution

6.1. Test Metrics

Test execution involves running test cases developed for the system and reporting test results. The first step in test execution is generally to validate the infrastructure needed for running tests in the first place.

Metric Name	Description
Severity Analysis	Distribution of severity of defects found
Severity Index	Weighted average of defects logged
Defects to Rejected Ratio	Number of defects against the number of rejects
Test Case Execution Productivity	Number of test cases executed per day per person.
Percentage of test Coverage	Calculated at the end of each milestone
Defect Cause Distribution Chart	This chart gives information on the cause of defects.
Number of Defects	The total number of defects found in a given time period/phase/test type that resulted in software or documentation modifications. Only accepted defects that resulted in modifying the software or the documentation are counted
Time to Fix a Defect	Effort required to resolve a defect (diagnosis and correction).Provides an indication of the maintainability of the product and can be used to estimate projected maintenance costs.
Test Case Execution Productivity	Number of test cases executed per day per person
Test Case Execution Statistics	This metric provides an overall summary of test execution activities. This can be categorized by build or release, module, by platform (OS, browser, locale etc.)

6.2. Test Reporting

Report	Objective/Content	Owner	Audience	Frequency
Test Report	Presents the test results	QA Lead	Project Manager	At every major deliverables stage
Test Status Report	Presents the progress of the test a activities	QA Lead	Project Management Test Team	Status report generates at regular intervals (Daily/Weekly/Monthly etc)
Usability Test Report	Presents the test results	QA Lead	Project Management Test Team	At every major deliverables stage
Deviation Test Report	Describes all the events that occur during a test execution	QA Lead	Project Management Test Team	At every major deliverables stage
Test Final Report	Summarizes the test activities associated with test design specification	QA Lead	Project Manager	At every major deliverables stage

6.3. Defect Management

A defect management system is made up of a combination of defect management tools or tool and a defect management process. In addition, the effectiveness of a defect management system is influenced by the organizational culture it operates within.

Following are the core features of a defect management tool:

- ❖ Provides a centralized repository for tracking defects across projects.
- ❖ Provides automated notifications of resource assignments.
- ❖ Ability to define defect resolution status in order to map back to your defect management process.
- ❖ Ability to provide management reporting, like the number of open defects grouped by
- ❖ various criteria such as open defects by project, severity, and priority.

Bug Tracker used as a defect management system.

Test engineer should have to report all initially found issues as defects in the defect management system.

All defects should track through the defect management system by development team and QA team.