|  |  |
| --- | --- |
|  | [Document title] |
|  |  |
|  | Ben Royans  [Course title]  [Date] |

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Change Date | By | Description |
| 1.0 | 8/6/2020 | Ben Royans | Updated for sprint 2 |
|  |  |  |  |
|  |  |  |  |

1 Introduction 1

1.1 Scope 1

1.1.1 In Scope 1

1.1.2 Out of Scope 1

1.2 Quality Objective 1

1.3 Roles and Responsibilities 2

1.3.1 Scrum Master: 2

1.3.1.1 Responsibilities 2

1.3.2 Developers 2

1.3.2.1 Responsibilities 2

2 Test Methodology 3

2.1 Overview 3

2.2 Test Levels 3

2.2.1 Use Case Testing 3

2.3 Bug Triage 3

2.3.1 Confirm new bugs 3

2.3.2 Prioritise confirmed bugs 3

2.3.3 Solve inconsistencies 3

2.3.4 Review stale/in-progress bugs 3

2.3.5 Review prioritised bugs 3

2.4 Suspension Criteria and Resumption Requirements 5

2.4.1 Suspension Criteria 5

2.4.2 Resumption Criteria 5

2.5 Test Completeness 5

2.6 Test Deliverables 5

3 Resource & Environment Needs 6

3.1 Testing Tools 6

3.2 Test Environment 6

4 Business Requirements 7

4.1 Business Requirement Identifcation 7

4.2 Requirements Traceability Matrix 7

5 Test Cases 8

5.1 Test Case Results Table 8

5.2 Test Case Screenshots 10

6 Terms/Acronyms 23

# Introduction

This document outlines the procedures, strategies and other details regarding the testing phase of the current sprint. This document will define the scope of the test process, the roles of the team and the hardware/software required.

## Scope

### In Scope

The following items will be included during the testing phase:

* Newsletter subscription functionality
  + Subscribe function
    - First name
    - Last name
    - Email address
    - Monthly newsletter
    - Burst notifcations
  + Modify subscription function
    - Update subscription preferences
  + Request removal of subscription
    - Email is sent to admin
    - Preferences reduced to none
* Admin login functionality
  + Login with credentials
  + Gain access to priveldged administrative pages

### Out of Scope

Anything NOT outlined in the above scope section will NOT be tested.

## Quality Objective

The objective of testing this application is to ensure it satisfies all requirements (both functional and non-functional) and also meets the quality standards as defined by the client. Through rigorous testing patterns any bugs, issues and un-expected behaviour will be identified and corrected. This approach will help us to deliver a quality application as the final product that exceeds customer expectations.

## Roles and Responsibilities

### Scrum Master:

The current scrum master is Benjamin Royans.

#### Responsibilities

* Sprint 2 project plan
* Software testing plan
* Programming
* Client demonstration

### Developers

The current developers are Catherine Burns and Jai Ananda.

#### Responsibilities

* Programming
* Software review plan
* Testing
* Performance report

# Test Methodology

## Overview

This project employs the Rapid Application Development (RAD) methodology. This means testing occurs within every iteration (sprint) of the software development cycle. As this project has been structured to carry out weekly sprints, the testing will also occur as frequently.

## Test Levels

### Use Case Testing

As functionality as implemented into the website, it will be tested against it’s requirements and specifications to ensure it behaving correctly. Use case testing involves tabulating all possible course of actions for a user to achieve a desired result (such as navigate to the search page), recording the expected results and then testing them against real-world tests.

## Bug Triage

The goal of the triage is to provide framework and structure for the process of eliminating bugs or issues within the software as it is being developed.

### Confirm new bugs

Confirm the presence or absence of a newly reported bug. Attempt to re-create the bug in simulated conditions.

### Prioritise confirmed bugs

Prioritise all existing bugs based on their severity level. The chart below will provide a guide as how to classify bugs.

### Solve inconsistencies

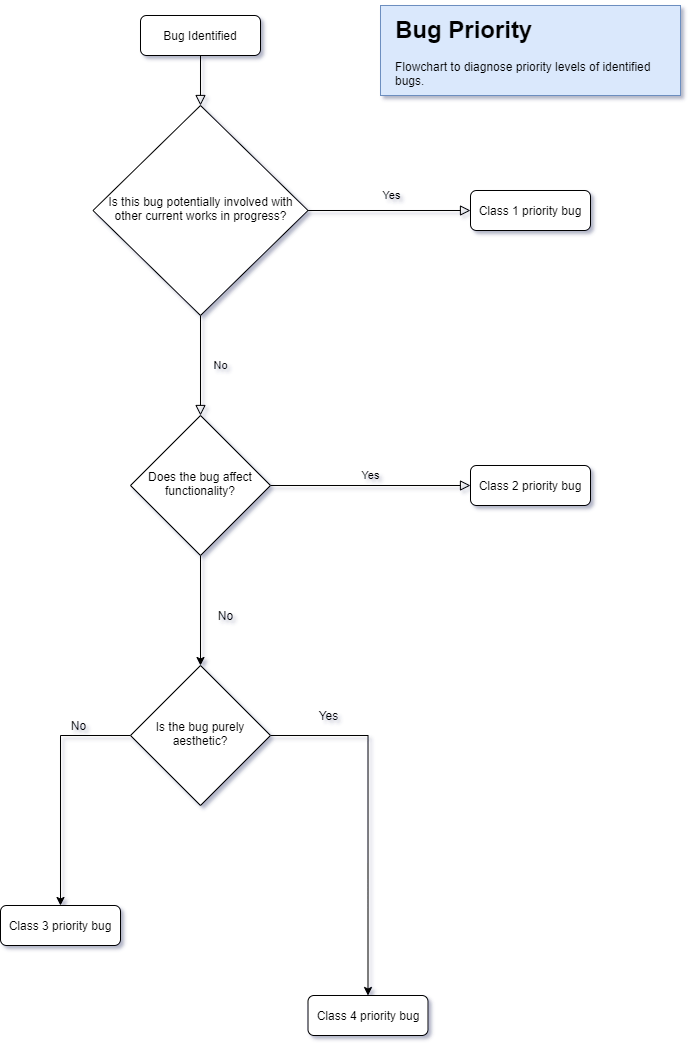
Gather any further required information regarding an existing bug and improve understanding.

### Review stale/in-progress bugs

Review any bug fixes currently in production, prioritising those that have not any recent progression.

### Review prioritised bugs

Assess each individual bug, devising possible solutions and documenting any restrictions.



## Suspension Criteria and Resumption Requirements

The are the criteria for any disruption to the testing process and workflow.

### Suspension Criteria

Testing for a particular functoinality is to be suspended once a bug has been identified. Sub-sequential behaviour can not be deemed reliable for testing once bugged methods are used.

### Resumption Criteria

Testing for particular functionalities can be resumed once any identified bugs have been corrected by the developers.

## Test Completeness

The initial testing process will be considered completed when all methods have been carried out and all data has been recorded. The final testing process will not be considered finished until all functional and non-functional requirement tests have passed.

Bugs will not hinder the testing process providing they do not affect functionality (class 3 or higher bugs are acceptable).

## Test Deliverables

Below is a list of deliverables to be expected upon the completion of the testing process.

* Test Plan (this document)
* Test Cases
* Requirement Traceability Matrix
* Test Metrics
* Customer Sign Off

# Resource & Environment Needs

## Testing Tools

The following tools are required to test the software:

* A modern web browser (Such as Google Chrome or Mozilla Firefox)
* A text editor (Such as VS Code or Notepad++)
* A code cleaner (Such as PHPCodeSniffer)
* GitKraken GUI client (For access to Bugs/Issues board)

## Test Environment

Below outlines the minimum hardware requirements that will be used to test the Application.

* [No minimum requirements]

The test environment is required to have one of the following operating systems installed:

1. Windows XP Service Pack 2 and above
2. Android Jellybean
3. Mac OS X

# Business Requirements

## Business Requirement Identifcation

The below table defines the business requirements for the current sprint.

|  |  |
| --- | --- |
| **Code** | **Requirement Description** |
| BR1 | The website must contain a member’s signup page which adds members to the database. |
| BR2 | Selection of monthly notifications. |
| BR3 | Selection of burst notifications. |
| BR4 | Have a remove subscription option. |
| BR5 | This function should email the administrator requesting the removal of the user’s database entry. |
| BR6 | Member’s full name and email address should be validated before entry. |

## Requirements Traceability Matrix

Below is the requirements traceability matrix outlining the cases responsible for the testing of the individual business requirements.

|  |  |  |  |
| --- | --- | --- | --- |
| **Business Requirement** | **Test Case #** | **Defects** | **Status** |
| BR1 | 1 | - | Satisfied |
| BR2 | 3, 4, 5, 8 | - | Satisfied |
| BR3 | 3, 4, 6, 8 | - | Satisfied |
| BR4 | 9, 10 | - | Satisfied |
| BR5 | 9, 10 | - | Satisfied |
| BR6 | 2, 3, 4, 5, 6, 7, 8 | - | Satisfied |

# Test Cases

## Test Case Results Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Functionality** | **Data used / Input** | **Expected Result** | **Pass / Fail** |
| 1 | Subscribe | Clicked ‘Subscribe+’ button | Navigates to subscription form. | PASS |
| 2 | Subscribe | Subscribe with:   * Invalid first name * Valid last name * Valid email address * Monthly box ticked * Burst box ticked | Error showing invalid name. Subscription does not continue. | PASS |
| 3 | Subscribe | Subscribe with:   * Valid first name * Invalid last name * Valid email address * Monthly box ticked * Burst box ticked | Error showing invalid name. Subscription does not continue. | PASS |
| 4 | Subscribe | Subscribe with:   * Valid first name * Valid last name * Invalid email address * Monthly box ticked * Burst box ticked | Error showing invalid email address. Subscription does not continue. | PASS |
| 5 | Subscribe | Subscribe with:   * Valid first name * Valid last name * Valid email address * Monthly box ticked * Burst box un-ticked | Notification stating subscription has been successful. User is added to the database table ‘subscriptions’, showing 1 for monthly, 0 for burst. | PASS |
| 6 | Subscribe | Subscribe with:   * Valid first name * Valid last name * Valid email address * Monthly box un-ticked * Burst box ticked | Notification stating subscription has been successful. User is added to the database table ‘subscriptions’, showing 0 for monthly, 1 for burst. | PASS |
| 7 | Subscribe | Subscribe with:   * Valid first name * Valid last name * Valid email address * Monthly box un-ticked * Burst box un-ticked | Notification stating subscription has been successful. User is added to the database table ‘subscriptions’, showing 0 for monthly, 0 for burst. | PASS |
| 8 | Subscribe | Subscribe with:   * Valid first name * Valid last name * Valid email address * Monthly box ticked * Burst box ticked | Notification stating subscription has been successful. User is added to the database table ‘subscriptions’, showing 1 for monthly, 1 for burst. | PASS |
| 9 | Unsubscribe | Unsubscribe with:   * Email address not on subscription list | Notification saying request has been sent to administration. | PASS |
| 10 | Unsubscribe | Unsubscribe with:   * Email address on subscription list | Notification saying request has been sent to administration. | PASS |
| 11 | Subscribe | Subscribe with:   * Given Name = “ “ (only white spaces as name) | Error showing invalid name. Subscription does not continue. | PASS |
| 12 | Subscribe | Subscribe with:   * Surname = “ “ (only white spaces as name) | Error showing invalid name. Subscription does not continue. | PASS |
| 13 | Subscribe | Subscribe with:   * Email address = “ “ (only white spaces as name) | Error showing invalid name. Subscription does not continue. | PASS |

## Test Case Screenshots

|  |  |
| --- | --- |
| **#** | **Screenshot** |
| 1 |  |
| 2 |  |

|  |  |
| --- | --- |
| 3 |  |

|  |  |
| --- | --- |
| 4 |  |

|  |  |
| --- | --- |
| 5 |  |

|  |  |
| --- | --- |
| 6 |  |
| 7 |  |

|  |  |
| --- | --- |
| 8 |  |
| 9 |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |

# Terms/Acronyms

| TERM/ACRONYM | DEFINITION |
| --- | --- |
| API | Application Program Interface |
| AUT | Application Under Test |
| RAD | Rapid Application Development |
| GUI | Graphical User Interface |
| RTM | Requirement Traceability Matrix |