SuperWoman

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 02.05.2018 | 1.0 | First Version | Team SuperWoman |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 2

1.1 Purpose 2

1.2 Scope 2

1.3 Intended Audience 2

1.4 Document Terminology and Acronyms 2

1.5 References 2

1.6 Document Structure **Fehler! Textmarke nicht definiert.**

2. Evaluation Mission and Test Motivation 2

2.1 Background 2

2.2 Evaluation Mission 2

2.3 Test Motivators 2

3. Target Test Items 2

4. Outline of Planned Tests 2

4.1 Outline of Test Inclusions 2

4.2 Outline of other candidates for potential inclusion 2

4.3 Outline of Test Exclusions 2

5. Test Approach 2

5.1 Initial Test-Idea Catalogs and other reference sources 2

5.2 Testing Techniques and Types 2

5.2.1 Data and Database Integrity Testing 2

5.2.2 Function Testing 2

5.2.3 Business Cycle Testing 2

5.2.4 User Interface Testing 2

5.2.5 Performance Profiling 2

5.2.6 Load Testing 2

5.2.7 Stress Testing 2

5.2.8 Volume Testing 2

5.2.9 Security and Access Control Testing 2

5.2.10 Failover and Recovery Testing 2

5.2.11 Configuration Testing 2

5.2.12 Installation Testing 2

6. Entry and Exit Criteria 2

6.1 Test Plan 2

6.1.1 Test Plan Entry Criteria 2

6.1.2 Test Plan Exit Criteria 2

6.1.3 Suspension and resumption criteria 2

6.2 Test Cycles 2

6.2.1 Test Cycle Entry Criteria 2

6.2.2 Test Cycle Exit Criteria 2

6.2.3 Test Cycle abnormal termination 2

7. Deliverables 2

7.1 Test Evaluation Summaries 2

7.2 Reporting on Test Coverage 2

7.3 Perceived Quality Reports 2

7.4 Incident Logs and Change Requests 2

7.5 Smoke Test Suite and supporting Test Scripts 2

7.6 Additional work products 2

7.6.1 Detailed Test Results 2

7.6.2 Additional automated functional Test Scripts 2

7.6.3 Test Guidelines 2

7.6.4 Traceability Matrices 2

8. Testing Workflow 2

9. Environmental Needs 2

9.1 Base System Hardware 2

9.2 Base Software Elements in the Test Environment 2

9.3 Productivity and Support Tools 2

9.4 Test Environment Configurations 2

10. Responsibilities, Staffing and Training Needs 2

10.1 People and Roles 2

10.2 Staffing and Training Needs 2

11. Iteration Milestones 2

12. Risks, Dependencies, Assumptions and Constraints 2

13. Management Process and Procedures 2

13.1 Measuring and Assessing the Extent of Testing 2

13.2 Assessing the deliverables of this Test Plan 2

13.3 Problem Reporting, Escalation and Issue Resolution 2

13.4 Managing Test Cycles 2

13.5 Traceability Strategies 2

13.6 Approval and Signoff 2

# Introduction

## Purpose

The purpose of the Iteration Test Plan is to gather all of the information necessary to plan and control the test effort for a given iteration. It describes the approach to testing the software, and is the top-level plan generated and used by managers to direct the test effort.

This *Test Plan* for the SuperWoman project supports the following objectives:

* Outlines the testing approach that will be used
* Identifies the items that should be targeted by the tests
* Identifies the required resources.

## Scope

This document addresses the following types and levels of testing:

* Unit-Tests
* Functionality Testing.

## Intended Audience

This document addresses the project members of SuperWoman and everybody who is interested in the SuperWoman-Project.

## Document Terminology and Acronyms

n/a

## References

* [GitHub](https://github.com/ItalisaS/SuperWoman)
* [Blog](https://superwoman2017blog.wordpress.com/)

# Evaluation Mission and Test Motivation

## Background

By testing our project, we make sure that all changes to the sourcecode do not break the functionality. Furthermore, by integrating the test process in our deployment process, we make sure that only working versions of our project get deployed. This renders possible that the web application is always available.

## Evaluation Mission

Our mission for the evaluation effort of the current iteration is following:

* Find as many bugs as possible as early as possible
* Validation of our sourcecode.

## Test Motivators

Using tests makes possible to validate the correctness of the methods and the functionalities in our sourcecode. Therewith, bugs can be found as early as possible and major effects can be prevented.

# Target Test Items

The listing below identifies those test items⎯software, hardware, and supporting product elements ⎯that have been identified as targets for testing. This list represents what items will be tested:

* Navigation within the website
* UI
* Functionality Testing.

# Outline of Planned Tests

## Outline of Test Inclusions

* JUnit-Testing
* Cucumber UI-Testing.

## Outline of Other Candidates for Potential Inclusion

n/a

## Outline of Test Exclusions

n/a

# Test Approach

## Initial Test-Idea Catalogs and Other Reference Sources

n/a

## Testing Techniques and Types

### Data and Database Integrity Testing

We are still working on implementing a database, thus we are unable to test this section presently.

### Function Testing

|  |  |
| --- | --- |
| Technique Objective: | Testing if the UI fits the expectations. |
| Technique: | We try to cover all use cases within one test by executing them one after another. |
| Oracles: | We are relying on the outcome of our testing tool. |
| Required Tools: | Jenkins |
| Success Criteria: | The tests pass without error messages. |
| Special Considerations: | tbd |

### Business Cycle Testing

n/a

### User Interface Testing

|  |  |
| --- | --- |
| Technique Objective: | Testing of the User Interface (test of the usability of our website) |
| Technique: | Usability Tests: Users should navigate through the website and play the game |
| Oracles: | The user likes the website. The website is easy to use and the menu navigation is simple and intuitive. |
| Required Tools: | User, Laptop with internet access |
| Success Criteria: | Website has a good usability and the users like our game. |

### Performance Profiling

n/a

### Load Testing

n/a

### Stress Testing

n/a

### Volume Testing

n/a

### Security and Access Control Testing

n/a

### Failover and Recovery Testing

n/a

### Configuration Testing

n/a

### Installation Testing

n/a

# Entry and Exit Criteria

## Test Plan

### Test Plan Entry Criteria

The execution of the test plan can begin as soon as the first unit test is written.

### Test Plan Exit Criteria

The execution of the test plan is complete when the project is finished

### Suspension and Resumption Criteria

tbd

## Test Cycles

### Test Cycle Entry Criteria

A test cycle begins when the build of a new version is initiated.

### Test Cycle Exit Criteria

tbd

### Test Cycle Abnormal Termination

A test cycle ends abnormally if an error occurs during the build.

# Deliverables

## Test Evaluation Summaries

Test evaluation is done by hand directly after the test has been executed.

## Reporting on Test Coverage

tbd

## Perceived Quality Reports

tbd

## Incident Logs and Change Requests

tbd

## Smoke Test Suite and Supporting Test Scripts

tbd

## Additional Work Product

tbd

### Detailed Test Results

tbd

### Additional Automated Functional Test Scripts

tbd

### Traceability Matrices

tbd

# Testing Workflow

Unit tests run automatically in the IDE, we start the functional tests after every commit.

# Environmental Needs

## Base System Hardware

The following table sets forth the system resources for the test effort presented in this *Test Plan*.

| **System Resources** | | |
| --- | --- | --- |
| **Resource** | **Quantity** | **Name and Type** |
| Database Server |  |  |
| —Network or Subnet |  | TBD |
| —Server Name |  | TBD |
| —Database Name |  | Postgres |
| Client Test PCs |  |  |
| —Include special configuration requirements |  | TBD |
| Test Development PCs |  | Özen, Lea, Isabella |

## Base Software Elements in the Test Environment

The following base software elements are required in the test environment for this *Test Plan*.

| **Software Element Name** | **Version** | **Type and Other Notes** |
| --- | --- | --- |
| Windows | Win 8 or higher | Operating System |
| Google Chrome |  | Internet Browser |
| Eclipse |  | IDE |
| JUnit | latest |  |
| Selenium | latest |  |

## Productivity and Support Tools

The following tools will be employed to support the test process for this *Test Plan*.

| **Tool Category or Type** | **Tool Brand Name** | **Vendor or In-house** | **Version** |
| --- | --- | --- | --- |
| Project Management | JIRA |  |  |

## Test Environment Configurations

The following Test Environment Configurations needs to be provided and supported for this project.

| **Configuration Name** | **Description** | **Implemented in Physical Configuration** |
| --- | --- | --- |

# Responsibilities, Staffing, and Training Needs

## People and Roles

This table shows the staffing assumptions for the test effort.

* Requirement Specifier: Özen, Lea & Isabella
* Designer: Özen
* Implementer: Özen, Lea & Isabella
* Tester: Özen, Lea & Isabella
* Project Manager [Deadline Manager]: Isabella
* Implementer [Code Cleaner]: Lea

# Iteration Milestones

[Identify the key schedule milestones that set the context for the Testing effort. Avoid repeating too much detail that is documented elsewhere in plans that address the entire project.]

| **Milestone** | **Planned Start Date** | **Actual Start Date** | **Planned End Date** | **Actual End Date** |
| --- | --- | --- | --- | --- |
| Iteration Plan agreed |  |  |  |  |
| Iteration starts |  |  |  |  |
| Requirements baselined |  |  |  |  |
| Architecture baselined |  |  |  |  |
| User Interface baselined |  |  |  |  |
| First Build delivered to test |  |  |  |  |
| First Build accepted into test |  |  |  |  |
| First Build test cycle finishes |  |  |  |  |
| [Build Two will not be tested] |  |  |  |  |
| Third Build delivered to test |  |  |  |  |
| Third Build accepted into test |  |  |  |  |
| Third Build test cycle finishes |  |  |  |  |
| Fourth Build delivered to test |  |  |  |  |
| Fourth Build accepted into test |  |  |  |  |
| Iteration Assessment review |  |  |  |  |
| Iteration ends |  |  |  |  |

# Risks, Dependencies, Assumptions, and Constraints

The risk for our project are outlined in our Risk Management Table. 🡪LINK

There are no additional risks for testing.

# Management Process and Procedures

## Measuring and Assessing the Extent of Testing

tbd

## Assessing the Deliverables of this Test Plan

The evaluation is carried out by monitoring the success of each test cycle. Errors can be assessed by reviewing the detailed test reports.

## Problem Reporting, Escalation, and Issue Resolution

Smaller problems are solved by the implementer of the corresponding task. Larger problems escalate and need an extra task in Jira.

## Managing Test Cycles

tbd

## Traceability Strategies

tbd

## Approval and Signoff

This test plan has to be approved by the Team.