TEST PLAN

XiaOMI PROJECT

Sakhatarov, Dmitry

2022

Table of Contents

[**1.** **INTRODUCTION** 3](#_Toc100755849)

[**2.** **SCOPE** 3](#_Toc100755850)

[2.1. In Scope: 3](#_Toc100755851)

[2.2. Out of Scope: 3](#_Toc100755852)

[**3.** **QUALITY OBJECTIVES** 3](#_Toc100755853)

[3.1. Primary Objectives: 3](#_Toc100755854)

[3.2. Secondary Objectives: 3](#_Toc100755855)

[**4.** **ROLES AND RESPONSIBILITIES** 4](#_Toc100755856)

[**5.** **TEST APPROACHES** 4](#_Toc100755857)

[5.1. Website Manual Tests Overview 4](#_Toc100755858)

[5.2. Website Automation Tests Overview 5](#_Toc100755859)

[5.3. Website API Tests Overview 5](#_Toc100755860)

[5.4. Website Performance Tests Overview 5](#_Toc100755861)

[**6.** **TEST TYPES** 5](#_Toc100755862)

[**7.** **TEST STRATEGY** 7](#_Toc100755863)

[7.1. QA role in test process: 7](#_Toc100755864)

[7.2. Bug Triage 8](#_Toc100755865)

[7.3. Bug life cycle 9](#_Toc100755866)

[7.3.1. Bug Severity and Priority Definition: 9](#_Toc100755867)

[7.3.2. Bug Priority List: 9](#_Toc100755868)

[**8.** **ENTRY AND EXIT CRITERIA** 10](#_Toc100755869)

[8.1. Entry Criteria 10](#_Toc100755870)

[8.2. Exit Criteria 10](#_Toc100755871)

[**9.** **SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS** 10](#_Toc100755872)

[9.1. Suspension criteria 10](#_Toc100755873)

[9.2. Resumption criteria 10](#_Toc100755874)

[**10.** **RESOURCE AND ENVIRONMENT NEEDS** 11](#_Toc100755875)

[10.1. Testing Tools: 11](#_Toc100755876)

[**11.** **TEST SCHEDULE** 11](#_Toc100755877)

[**12.** **APPROVALS** 11](#_Toc100755878)

[**13.** **TERMS/ACRONYMS** 11](#_Toc100755879)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Version** | **Author** | **Description** | **Reviewer** | **Approver** |
| 12.04.2022 | 1.1 | Sakhatarov D.M. | First draft | Efremov S. |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Document Revision History:**

# **INTRODUCTION**

The Test Plan is designed to prescribe the scope, approach, resources, and schedule of all testing activities of the project Xiaomi website <https://www.mi.com/ru/index.html> . The plan identifies the items to be tested, the features to be tested, the types of testing to be performed, the personnel responsible for testing, the resources and schedule required to complete testing, and the risks associated with the plan. Customers want a perfect website, which has passed the full cycle of manual testing.

# **SCOPE**

## In Scope:

All the features of website Xiaomi which were defined in software requirement specs need to be tested. The document mainly targets to check the Registration form, GUI of the website, Xiaomo Social Media Module, Xiaomi’s API, Website Performance and validates data in report output as per Requirements Specifications provided by Client.

**Functions to be tested:**

* GUI
* Registration form
* Xiaomi Social Media Module
* Xiaomi API Collections
* Website Performance

## Out of Scope:

These features are not to be tested because they are not included in the software requirement specs.

**Functions not to be tested:**

* Hardware Interfaces
* Software Interfaces
* Database logical
* Communications Interfaces
* Website Security

# **QUALITY OBJECTIVES**

The test objectives are to verify the functionality of website Xiaomi. The project should focus on testing the Website’s GUI, Registration form, API, Performance and guarantee all these operations can work normally in a real business environment.

## Primary Objectives:

A primary objective of testing is to: assure that the system meets the full requirements, including quality requirements (functional and non-functional requirements) and fit metrics for each quality requirement and satisfies the use case scenarios and maintain the quality of the product. At the end of the project development cycle, the user should find that the project has met or exceeded all of their expectations as detailed in the requirements. Any changes, additions, or deletions to the requirements document, Functional Specification, or Design Specification will be documented and tested at the highest level of quality allowed within the remaining time of the project and within the ability of the test team.

## Secondary Objectives:

The secondary objectives of testing will be to: identify and expose all issues and associated risks, communicate all known issues to the project team, and ensure that all issues are addressed in an appropriate manner before release. As an objective, this requires careful and methodical testing of the website to first ensure all areas of the system are scrutinized and, consequently, all issues (bugs) found are dealt with appropriately

# **ROLES AND RESPONSIBILITIES**

The project should use outsource members as the testers also to save the project cost.

|  |  |  |
| --- | --- | --- |
| **№** | **Member** | **Task** |
|  | Test Manager | Manage the whole project Define project directions Acquire appropriate resources |
|  | Testers | Testers Identifying and describing appropriate test techniques / tools / automation. Verify and assess the Test Approach Execute the tests, Log results, Report the defects |
|  | Developer in Test | Test Implement the test cases, test program, test suite etc. |
|  | Test Administrator | Builds up and ensures test environment and assets are managed and maintained Support Tester to use the test environment for test execution |
|  | SQA members | Take in charge of quality assurance Check to confirm whether the testing process is meeting specified requirements |

# **TEST APPROACHES**

In the project Xiaomi, there are 4 types of testing that should be conducted:

* Manual Tests
* Automation Tests
* API Tests
* Performance Tests

The project is using an Agile approach, with weekly iterations (every 14 days). At the end of the second week the requirements identified for that iteration will be delivered to the team and will be tested. Team also must use experience-based testing and error guessing to utilize testers' skills and intuition, along with their experience with similar applications or technologies.

Environment for tests:

* OS: MacOS, iOS, Windows
* Devices: MacBook, iPhone 12 Pro Max, Android 3
* Browsers: Google Chrome, Safari (for MacOS), FireFox (for Windows OS)

## Website Manual Tests Overview

The Xiaomi Website's Module Menu “**Все товары**” on the main page will be tested in this part.

* Verify that all images’ links on Homepage are working correctly according to Business requirements.
* Verify that Logo leads at the website's Homepage after clicking on it.
* Check the ability of user to reach all Xiaomi’s products in a “**Все товары**” section.
* Check the ability of user to reach all Xiaomi’s social medias pages by clicking on the following provided Social Media Icons:
* Vkontakte, Facebook, Telegram, Micommunity.

Check the ability of user sign up to Xiaomi webpage:

* Valid data (valid email) - Positive testing
* Invalid data (invalid email) - Negative testing

Positive/Negative/Ad hoc testing should be used also if necessary.

## Website Automation Tests Overview

Tests which are indicated in the Manual part will be automated.

**Testing Tools:**

* Selenium IDE
* XPath (ChroPath/TruPth)

## Website API Tests Overview

Xiaomi's API Collections will be tested through the Server response in this part.

API Collections for testing:

* Address Collection

**Testing Tools:**

* Postman
* JSON Formatter & Validator

## Website Performance Tests Overview

Tests for evaluation of the website’s speed, responsiveness and stability under a workload will be done in this part. Plan for Performance testing:

1. Check Xiaomi’s website:

* Performance
* Accessibility
* Best practice
* SEO

**Testing Tools:**

* Lighthouse
* GT Metrix
* SpeedLab

# **TEST TYPES**

In the Xiaomi project there are 9 types of testing that should be conducted:

* Exploratory testing
* Smoke Testing
* GUI Testing
* Functional Testing
* Positive testing
* Negative testing
* ADHOC testing
* API testing
* Performance testing

**Exploratory testing:**

Exploratory testing will include a type of software testing where Test cases are not created in advance but QA check system on the fly. QA may note down ideas about what to test before test execution.

**Smoke Testing:**

Smoke Testing is a software testing process that determines whether the deployed software build is stable or not. Smoke testing is a confirmation for the QA team to proceed with further software testing. It consists of a minimal set of tests run on each build to test software functionalities. Smoke testing is also known as “Build Verification Testing” or “Confidence Testing.” In simple terms, we are verifying whether the important features are working and there are no showstoppers in the build that is under testing.

**GUI Testing:**

GUI testing will include testing the UI/UX part of the report. It covers users' Report format, look and feel, error messages, spelling mistakes, GUI guideline violations.

**Functional Testing:**

Functional testing is carried out in order to find out unexpected behavior of the report. The characteristics of functional testing are to provide correctness, reliability, testability, and accuracy of the report output/data.

**Positive testing:**

Positive testing will include the type of testing that can be performed on the system by providing the valid data as input. It checks whether an application behaves as expected with positive inputs.

**Negative testing:**

Negative testing will include a method of testing an application or system that ensures that the application is according to the requirements and can handle the unwanted input and user behavior. Invalid data is inserted to compare the output against the given input. Negative testing is also known as failure testing or error path testing. When performing negative testing errors messages are expected.

**ADHOC testing:**

ADHOC testing will include an informal testing type with an aim to “break” the system

**API testing:**

API testing is a type of software testing that analyzes an application program interface (API) to verify it fulfills its expected functionality, security, performance, and reliability. An API test is generally performed by making requests to one or more API endpoints and comparing the response with expected results.

**Performance testing:**

Performance testing is the practice of evaluating how a system performs in terms of responsiveness and stability under a particular workload. Performance tests are typically executed to examine speed, robustness, reliability, and application size

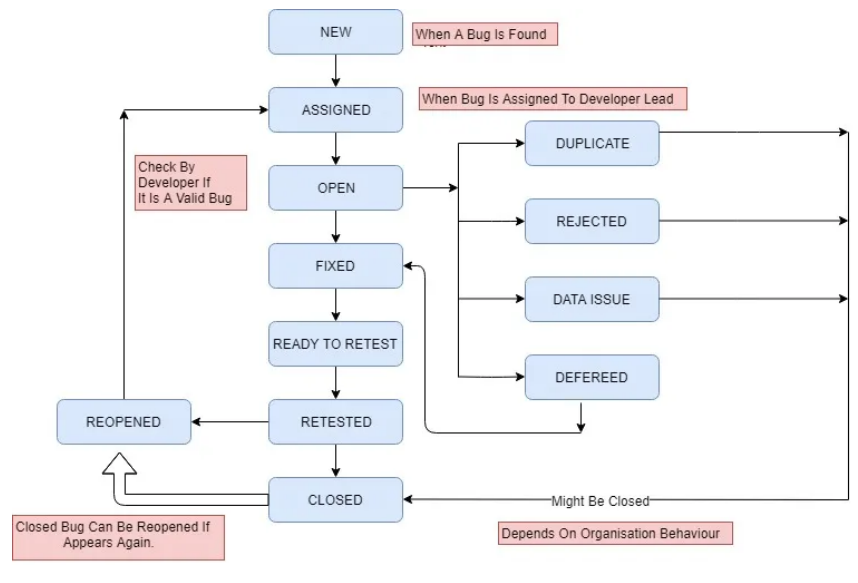
# **TEST STRATEGY**

## QA role in test process:

* Understanding Requirements.
* Requirement specifications will be sent by client.
* Understanding of requirements will be done by QA.
* Preparing **Test Cases** - QA will be preparing test cases based on the exploratory testing. This will cover all scenarios for requirements.
* Preparing **Test Matrix** - QA will prepare an RTM which maps test cases to respective requirements. This will ensure the coverage for requirements.
* Reviewing test cases and matrix.
* Peer review will be conducted for test cases and test matrix by QA Lead.
* Any comments or suggestions on test cases and test coverage will be provided by the reviewer respective Author of Test Case and Test Matrix.
* Suggestions or improvements will be re-worked by the author and will be sent for approval.
* Re-worked improvements will be reviewed and approved by the reviewer.
* Creating **Test Data** - Test data will be created by respective QA on client's developments/test site based on scenarios and Test cases
* Executing **Test Cases** - Test cases will be executed by respective QA on the client's development/test site based on designed scenarios, test cases and Test data.
* Test result (Pass/Fail) will be updated in test case document Defect Logging and Reporting. QA will be logging the defect/bugs in Word document and JIRA, found during execution of test cases.
* QA will inform the respective developer about the defect/bugs.
* QA will perform Retesting and Regression Testing. Retesting for fixed bugs will be done by respective QA once it is resolved by the respective developer and bug/defect status will be updated accordingly. In certain cases, regression testing will be done if required
* **Deployment/Delivery** - Once all bugs/defects reported after complete testing are fixed and no other bugs are found, the report will be deployed to the client's test site. Once a round of testing will be done by QA on the client's test site if required Report will be delivered along with sample output by email to the respective lead and Report group.

## Bug Triage

All the issues found while testing will be logged into JIRA



## Bug life cycle

### Bug Severity and Priority Definition:

Bug Severity and Priority fields are both very important for categorizing bugs and prioritizing if and when the bugs will be fixed. The bug Severity and Priority levels will be defined as outlined in the following tables below. Testing will assign a severity level to all bugs. The Test Lead will be responsible to see that a correct severity level is assigned to each bug. The QA Lead, Development Lead and Project Manager will participate in bug review meetings to assign the priority of all currently active bugs. This meeting will be known as “Bug Triage Meetings”. The QA Lead is responsible for setting up these meetings.

|  |  |  |
| --- | --- | --- |
| **Severity ID** | **Severity** | **Description** |
|  | Highest | The module/product crashes or the bug causes nonrecoverable conditions. System crashes or database or file corruption, or potential data loss, program hangs requiring reboot are all examples of a Severity 1 bug. |
|  | High | Major system components unusable due to failure or incorrect functionality. Severity2 bugs cause serious problems such as a lack of functionality, or insufficient or unclear error messages that can have a major impact to the user, prevents other areas of the app from being tested, etc. Severity 2 bugs can have a work around, but the work around is inconvenient or difficult. |
|  | Medium | Incorrect functionality of component or process. There is a simple work around for the bug if it is Severity 3. |
|  | Low | Documentation errors or signed off Severity 4 bugs. Low severity bug occurs when there is almost no impact on the functionality, but it is still a valid defect that should be corrected |

### Bug Priority List:

|  |  |  |
| --- | --- | --- |
| **Priority ID** | **Priority** | **Description** |
|  | Highest | This bug must be fixed immediately; the product cannot ship with this bug. |
|  | High | These are important problems that should be fixed as soon as possible. It would be an embarrassment to the company if this bug shipped. |
|  | Medium | The problem should be fixed within the time available. If the bug does not delay the shipping date, then fix it. |
|  | Low | It is not important (at this time) that these bugs be addressed. Fix these bugs after all other bugs have been fixed. |
|  | Lowest | Documentation errors. |

# **ENTRY AND EXIT CRITERIA**

## Entry Criteria

* All test hardware platforms must have been successfully installed, configured, and functioning properly.
* All the necessary documentation, design, and requirements information should be available that will allow testers to operate the system and judge the correct behavior.
* All the standard software tools including the testing tools must have been successfully installed and functioning properly.
* Proper test data is available.
* The test environment such as lab, hardware, software, and system administration support should be ready.
* QA resources have completely understood the requirements.
* QA resources have strong knowledge of functionality.
* Reviewed test scenarios, test cases and RTM

## Exit Criteria

* A certain level of requirements coverage has been achieved.
* No high priority or severe bugs are left outstanding.
* All high-risk areas have been fully tested, with only minor residual risks left outstanding.
* Cost – when the budget has been spent.
* The schedule has been achieved.

# **SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS**

## Suspension criteria

* The build contains many serious defects which seriously or limit testing progress.
* Significant change in requirements suggested by client.
* Software/Hardware problems.
* Assigned resources are not available when needed by test team

## Resumption criteria

* Resumption will only occur when the problem(s) that caused the suspension have been resolved.

# **RESOURCE AND ENVIRONMENT NEEDS**

## Testing Tools:

|  |  |
| --- | --- |
| **Process** | **Tools** |
| Test |  |
|  |  |
|  |  |
|  |  |
|  |  |

# **TEST SCHEDULE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task Name** | **Start** | **Finish** | **Effort** | **Comments** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# **APPROVALS**

|  |  |  |
| --- | --- | --- |
|  | **Project Manager** | **QA Lead** |
| Name |  |  |
| Signature |  |  |

# **TERMS/ACRONYMS**

The below terms are used as examples, please add/remove any terms relevant to the document.

|  |  |
| --- | --- |
| **Term/Acronym** | **Definition** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |