**Table of Contents**

1. [INTRODUCTION 2](#_heading=h.gjdgxs)
2. [SCOPE 2](#_heading=h.30j0zll)
3. [QUALITY OBJECTIVES 3](#_heading=h.1fob9te)
   1. [Primary Objectives 3](#_heading=h.3znysh7)
   2. [Secondary Objectives 3](#_heading=h.2et92p0)
4. [TEST APPROACH 3](#_heading=h.tyjcwt)
5. [ROLES AND RESPONSIBILITIES 4](#_heading=h.1t3h5sf)
6. [ENTRY AND EXIT CRITERIA 5](#_heading=h.4d34og8)
   1. [Entry Criteria 5](#_heading=h.2s8eyo1)
   2. [Exit Criteria](#_heading=h.17dp8vu) 5
7. [TEST STRATEGY 6](#_heading=h.35nkun2)
   1. [QA role in test process 6](#_heading=h.1ksv4uv)
   2. [Bug life cycle: 7](#_heading=h.44sinio)
   3. [Testing types 8](#_heading=h.2jxsxqh)
   4. [Bug Severity and Priority Definition 9](#_heading=h.z337ya)

[Severity List](#_heading=h.3j2qqm3) 9

[Priority List 10](#_heading=h.1y810tw)

1. [RESOURCE AND ENVIRONMENT NEEDS 1](#_heading=h.4i7ojhp)0
   1. [Testing Tools 1](#_heading=h.2xcytpi)0
   2. [Test Environment 1](#_heading=h.3whwml4)0
2. [TEST SCHEDULE 1](#_heading=h.2bn6wsx)0

[APPROVALS 1](#_heading=h.qsh70q)1

[TERMS/ACRONYMS 1](#_heading=h.3as4poj)1

**Test Plan**

**Project “Minions”**

Document Revision History

| Date | Version | Description | Author | Reviewer | Approver |
| --- | --- | --- | --- | --- | --- |
| 22.02 | 0.1 | Test plan was created | Minions |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## INTRODUCTION

Customers expect a flawless website that has undergone a thorough cycle of manual testing. Given the site's uniqueness, maintaining the same level of quality and consistency is critical.

The Test Plan was established to help team members communicate more effectively. This document outlines the procedures and methodologies that will be used to test the "Registration and Login of Minions Portal" unit, integration, and system. The objectives, test responsibilities, entry and exit criteria, scope, main schedule milestones, entrance and exit criteria, and approach are all included. This document specifies the test deliverables as well as what is considered in and out of scope.

## SCOPE

The document mainly targets the GUI testing and validating data in report output as per Requirements Specifications provided by Client.

* 1. Functions to be tested.
     + GUI
     + Registration and Login
     + DB Connection
  2. Functions not to be tested.

1. Not other than mentioned above in section 2.1

## QUALITY OBJECTIVES

### Primary Objectives

A fundamental goal of testing is to ensure that the system meets all criteria, including quality requirements (both functional and non-functional) and fit metrics for each quality requirement, as well as to satisfy use case scenarios and maintain product quality. The user should find that the project has met or exceeded all of their expectations as outlined in the requirements at the end of the project development cycle.

Any changes, additions, or deletions to the requirements document, Functional Specification, or Design Specification shall be documented and tested to the highest quality possible given the project's remaining timeframe and the test team's capabilities.

### Secondary Objectives

Secondary testing goals include identifying and exposing all issues and hazards, communicating all known issues to the project team, and ensuring that all concerns are addressed appropriately before release. As a goal, this necessitates thorough and methodical testing of the program to guarantee that all aspects of the system are examined and, as a result, all issues (bugs) discovered are addressed effectively.

## TEST APPROACH

As a result, the approach adopted is analytical, in accordance with a requirements-based strategy, in which an examination of the requirements specification serves as the foundation for planning.

calculating and planning tests During exploratory testing, test cases will be produced. Test Strategy determines all test kinds.

Using testers' abilities and intuition, as well as their expertise with similar applications or technologies, the team must also apply experience-based testing and error guessing.

The project is being carried out in an agile manner, with weekly iterations. The needs established for that iteration will be presented to the team and tested at the end of each week.

## ROLES AND RESPONSIBILITIES

| Role | Staff Member | Responsibilities |
| --- | --- | --- |
| Project Manager | Prof Bill | 1.Acts as a primary contact for development and QA team.  2.Responsible for Project schedule and the overall  success of the project. |
| QA | S.Srilekha | 1. Participation in the project plan creation/update process. 2.Planning and organization of the test process for the release.  3.Report progress on work assignments to the PM  4.Writing and executing Test cases  5.Reviewing Test cases  6.Coordinate with QA Lead for any issues or problems encountered during test preparation/execution/defect handling |
| DEV/UI | Jaswin/Vikram | 1. Understand requirements 2. Front-End development. 3. Back-End code development. 4. DB Connection 5. Bug Review meeting 6. Preparation of Test Data. 7. Coordinate with QA Lead for any issues or problems encountered during test preparation/execution/defect handling. |

## ENTRY AND EXIT CRITERIA

### Entry Criteria

* All test hardware platforms must have been installed, configured, and tested successfully.
* All relevant documentation, design, and requirements information should be available so that testers may operate the system and assess its right behavior.
* All of the standard software tools, including the testing tools, must have been installed and tested successfully.
* There is sufficient test data available.
* The test environment, including the lab, hardware, software, and system administration support, should all be in place.
* The requirements have been thoroughly understood by the QA resources.
* QA personnel are well-versed in functionality.
* Test scenarios and test cases were reviewed.

### Exit Criteria

* A specific level of coverage of requirements has been attained.
* There are no high-priority or serious bugs left unresolved.
* Only minimal residual concerns remain after all high-risk regions have been thoroughly examined.
* The schedule has been achieved

# TEST STRATEGY

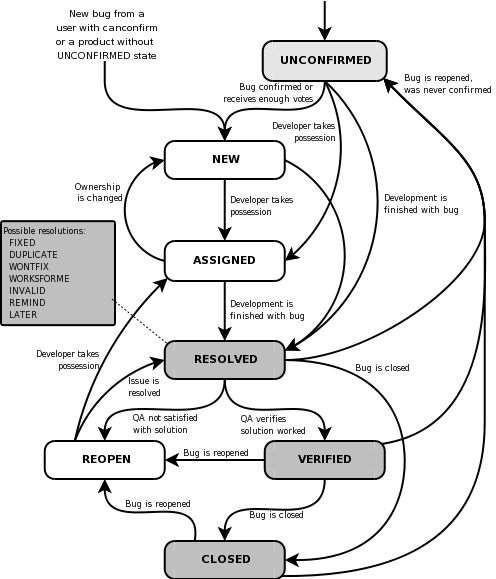
### QA role in test process

* Understanding Requirements:
* Requirement specifications will be sent by client.
* Understanding of requirements will be done by QA
* Preparing Test Cases:
* On the basis of the exploratory testing, QA will prepare test cases. This will cover all requirements scenarios.
* Examining the test cases and the matrix
* QA Lead will conduct peer review for test cases and test matrix.
* Reviewers shall submit any comments or suggestions on test cases and test coverage to the respective Author of Test Case and Test Matrix.
* Author will rework any suggestions or improvements and send them for approval.
* The reviewer will look through the reworked improvements and approve them.
* 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 (Actual Result, Pass/Fail) will updated in test case document Defect Logging and Reporting:
* QA will be logging the defect/bugs, found during execution of test cases. After this, QA will inform the respective developer about the defect/bugs.
* 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:
* After all reported issues/defects have been resolved and no new bugs have been discovered, the report will be deployed to the client's test site by PM.
* If necessary, QA will do one round of testing on the client's test site. The report will be emailed to the appropriate lead and Report group, along with a sample output.
* The filled hard copy of the delivery slip will be submitted to the appropriate developer by QA.
* Lead will send the report delivery email to the client after he receives the hard copy of the delivery slip filled out by QA and development.

### Bug life cycle:

All the issues found while testing will be logged into Word documents.

Bug life cycle for this project is as follows:



### Testing types:

Black box testing:

It is sometimes called behavioral testing or Partition testing. This kind of testing focuses on the functional requirements of the software. It enables one to derive sets of input conditions that will fully exercise all functional requirements for a program.

GUI Testing:

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

Integration Testing:

Integration testing is a systematic technique for constructing the program structure while conducting tests to uncover errors associated with interacting. In Report, integration testing includes the testing Report from respective location(s).

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.

System Testing:

System testing of software is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.

Performance Testing:

* Check the optimal time the page is loaded

- Check the operation of the system under load User acceptance testing:

The purpose behind user acceptance testing is to confirm that the system is developed according to the specified user requirements and is ready for operational use. Acceptance testing is carried out at two levels - Alpha and Beta Testing. User acceptance testing (UAT) will be done at the Client.

Alpha testing:

The alpha test is conducted at the developer's site by client.

### 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 on a routine basis to address the current set of new and existing but unresolved bugs.

### Severity List

The tester entering a bug into GForge is also responsible for entering the bug Severity.

| **Severity ID** | **Severity** | **Severity Description** |
| --- | --- | --- |
| 1 | Critical | The module/product crashes or the bug causes non- recoverable conditions. System crashes, GP Faults, or database or file corruption, or potential data loss, program  hangs requiring reboot are all examples of a Sev. 1 bug. |
| 2 | High | Major system components unusable due to failure or incorrect functionality. Sev. 2 bugs cause serious problems such as a lack of functionality, or insufficient or unclear error messages that can have a major impact on the user, prevent other areas of the app from being tested, etc. Sev. 2 bugs can have a  work around, but the work around is inconvenient or difficult. |
| 3 | Medium | Incorrect functionality of component or process. There is a  simple work around for the bug if it is Sev. 3. |
| 4 | Minor | Documentation errors or signed off severity 3 bugs. |

Priority List

| **Priority** | **Priority Level** | **Priority Description** |
| --- | --- | --- |
| 1 | Must Fix | This bug must be fixed immediately; the product cannot  ship with this bug. |
| 2 | Should Fix | These are important problems that should be fixed as soon as possible. It would be an embarrassment to the company  if this bug shipped. |

## RESOURCE AND ENVIRONMENT NEEDS

### Testing Tools

| Process | Tool |
| --- | --- |
| Test case creation | Microsoft Excel |
| Test case tracking | Microsoft Excel |
| Test case execution | Manual |
| Test case management | Microsoft Excel |
| Defect management | Microsoft Word |
| Test reporting | Microsoft Word |

### Test Environment

* Support level 1 (browsers):
  + Windows 10: Edge, Chrome (latest), Firefox (latest), Safari (latest)

## TEST SCHEDULE

| **Task Name** | **Start** | **Finish** | **Effort** | **Comments** |
| --- | --- | --- | --- | --- |
| Test Planning | 2/14 | 2/17 |  | Support! |
| Review Requirements  documents | 2/14 | 2/18 |  |  |
| Create test basis | 2/15 | 2/19 |  |  |
| Staff and train new test  resources | 2/16 | 2/19 |  |  |
| First deploy to QA test  environment | 2/20 | |  |  |
| Functional testing –  Iteration 1 | 2/20 | 2/20 |  |  |
| Regression testing | 2/21 | 2/21 |  |  |
| UAT | 2/21 | 2/21 |  |  |
| Resolution of final defects  and final build testing | 2/22 | 2/22 |  |  |
| Release to Production | 2/22 | 2/22 |  |  |

# APPROVALS:

|  | **Project Manager** | **QA Lead** |
| --- | --- | --- |
| **Name** | Prof Bill | S.Srilekha |
| **Signature** |  |  |

## TERMS/ACRONYMS

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

| **TERM/ACRONYM** | **DEFINITION** |
| --- | --- |
| API | Application Program Interface |
| GUI | Graphical user interface |
| PM | Project manager |
| UAT | User acceptance testing |
| CM | Configuration Management |
| QA | Quality Assurance |