TEST STRATEGY FOR ‘’Daruieste Viata‘’ Website Testing

PROJECT NAME: ‘’Daruieste Viata’’ Website Testing

DOCUMENT CONTROL

Version	1.1
Date	16/07/2022
Developed by	Oana-Alexandra Miu

DOCUMENT SIGN-OFF

Version	Status	Date	Approved by	Job Title
1.1	Final	17/07/2022	Maria Grigore	Product Owner

CONTENTS

1 INTRODUCTION

2 PURPOSE

3 SYSTEM OVERVIEW

4 SCOPE OF TESTING

4.1 IN SCOPE

4.2 OUT OF SCOPE

5 APPROACH TO TESTING

5.1 PRINCIPLES & APPLICATION

5.2 TEAM – PLANNED ITERATIVE

5.3 [TEST PHASE 1]

5.3.1 Objective

5.3.2 Scope 14

5.3.3 Test Preparation

5.3.3.1 Entry Criteria

5.3.3.2 Exit Criteria

5.3.4 Test Execution

5.3.4.1 Entry Criteria

5.3.4.2 Exit Criteria

5.4 [TEST PHASE 2]

5.4.1 Objective

5.4.2 Scope 16

5.4.3 Test Preparation

5.4.3.1 Entry Criteria

5.4.3.2 Exit Criteria

5.4.4 Test Execution

5.4.4.1 Entry Criteria

5.4.4.2 Exit Criteria

6 TEST ENVIRONMENTS REQUIREMENTS

6.1 DEVELOPMENT/LOCAL ENVIRONMENT

6.2 TEST ENVIRONMENT

6.3 PRE PRODUCTION ENVIRONMENT

6.6 PRODUCTION ENVIRONMENT

7 TEST DATA REQUIREMENTS

8 TESTING TOOLS & TECHNIQUES

8.1 REQUIREMENTS & USE CASE MANAGEMENT

8.3 TEST MANAGEMENT & DEFECT TRACKING

9 TESTING ROLES & RESPONSIBILITIES

10 TEST MANAGEMENT

11 DEFECT MANAGEMENT

11.1 DEFECT MANAGEMENT PROCESS

12 TEST SCHEDULE

12.1.1 Test Phase 1

12.1.2 Test Phase 2

15 REFERENCED DOCUMENTS

1 INTRODUCTION

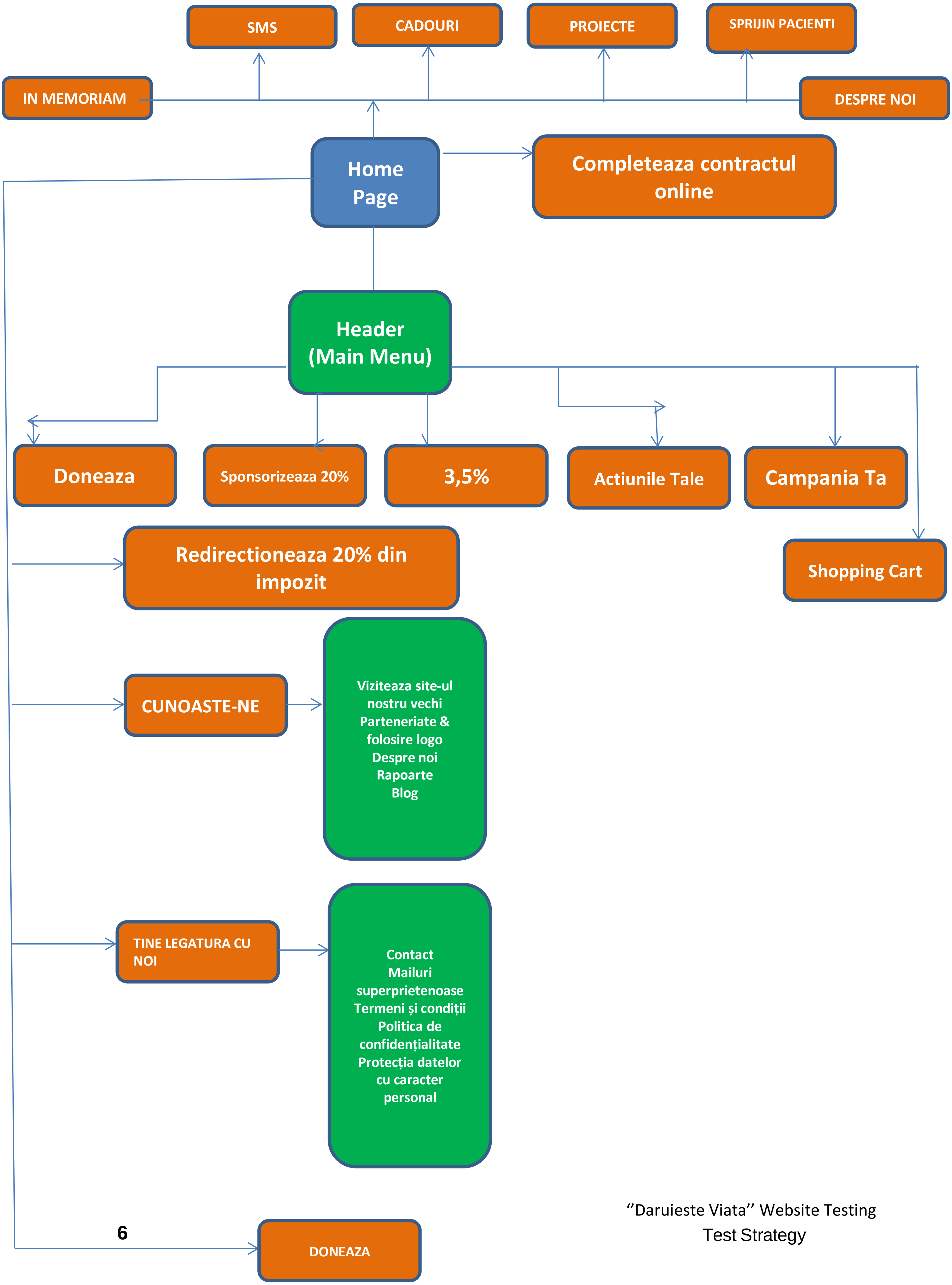
The objective of ‘’Daruieste Viata’’ Website Testing project is to present the whole process of testing a software product, including the documentation that contains the important aspects of this process. Quality Assurance (QA) presents not only the technical side, but also the administrative side, which helps the team to work more efficiently. This actually ensures the improvement of the future experience.

2 PURPOSE

This Test Strategy will provide a high-level view of how testing will be completed for the ‘’Daruieste Viata’’ Website Testing project. There are many sections to this document, but the focus is on the overall approach to testing for the different test phases that are to be completed.

3 SYSTEM OVERVIEW

The System Under Test (SUT) consists of the following components.


SMS
CADOURI
PROIECTE
SPRIJIN PACIENTI
IN MEMORIAM
DESPRE NOI
Home Page
Completeaza contractul online
Header (Main Menu)
Doneaza
Sponsorizeaza 20%
3,5%
Actiunile Tale
Campania Ta
Shopping Cart
Redirectioneaza 20% din impozit
CUNOASTE-NE
Viziteaza site-ul nostru vechi
Parteneriate & folosire logo
Despre noi
Rapoarte
Blog
TINE LEGATURA CU NOI
Contact
Mailuri superprietenoase
Termeni și condiții
Politica de confidențialitate
Protecția datelor cu caracter personal
DONEAZA

4 SCOPE OF TESTING

As the diagram in section 3 shows the scope of the ‘’Daruieste Viata’’ Website Testing project testing is to show whether the customer's user experience requirements are met. The exploratory test conducted will identify (if applicable) the main impediments to user-friendly and qualitative delivery on the platform called ‘’Daruieste Viata’’.

4.1 IN SCOPE

The types of testing that are in-scope are:

- Static Testing
  - Requirements
  - Architecture
  - Code Reviews
- Functional Testing
- Non-functional Testing
  - Performance (Load, Stress)
  - Browser compatibility

4.2 OUT OF SCOPE

The types of testing that are out of scope are:

- User Acceptance Testing (UAT) - this will be performed in the development's environment butwill be planned and executed by the client;
- Operational Acceptance Testing (OAT) – this test is usually performed right before releasing the product on the production environment
- Unit Testing is the type of testing within the smallest pieces of code are tested in the development's environment and it is usually performed by those who developed the code;
- Regression Testing which is the type of testing that is telling us wheter a software application still functions in the right parameters after any code changes, updates, or improvements;

5 APPROACH TO TESTING

The following section describes the high-level approach to testing.

5.1 PRINCIPLES & APPLICATION

5.1.1.1 Principle

It is important to understand the risks to the business with regards to any system that is developed. Some parts of the system will be critical to the running of the business and some will not. By understanding the risk profile, we can tune the amount & types of testing we complete. This can then give a realistic scale of testing for each project and therefore we can support the estimates we put forward.

5.1.1.2 Application

Only by applying a few of the 7 essential principles of testing to this project we have ensured the most effective process possible, both from a technical and business point of view.

- Exhaustive testing was avoided by applying minimal technical knowledge (e.g. viewing html code), so time was streamlined. We also focused on the main functionalities that the user may encounter while browsing the site.
- Although the product appears to be without defect at first inspection, the use of the main features has revealed easily visible defects (although with a small degree of severity).
- Most of the defects are in the area of user experience through translation errors and defective design. So, we can conclude that the defects cluster together.
- The product has been tested through black box testing, from the perspective of a user who wants the site to provide the convenience needed to perform the main actions (donate, get involved, purchase, inform). Therefore, the testing carried out is totally dependent on this context.

5.2 TEAM – PLANNED ITERATIVE

Testing is a core discipline within Development Company's development framework (TEAM – The Development Company Adaptive Model), with engagement at each of the key stages of development as shown in the diagram below.

TEAM defines six dimensions, the core dimensions of: Process, Team and Tools and the supporting dimensions of: Communication, Culture and People. Each of these is addressed specifically in the testing discipline within the methodology.

There are two distinct software development approaches described by TEAM:

- Agile

- Planned Iterative

This project will be delivered using the Planned Iterative methodology.

Planned Iterative Principles impacting testing include:

- Testing is completed by an Independent Testing Group
- Test throughout the Project
- Focus on finding defects early in the SDLC.
- Formal Testing Scope within each iteration (Continuously verifying quality)
- Focus on testing the Architecture first.
- Test Cases derived from Use Cases.
- Tailor process and documentation for the size and complexity of the project

5.3 TEST PHASE 1 - FIRST SPRINT

5.3.1 Objective

The objective of Test Phase 1 is to test the secondary set of features next to those in the main menu that aim to improve the user experience on the tested web platform.

5.3.2 Scope

The scope of the first phase is to carry out a set of tests on a set of functionalities, with an importance similar to that of the main menu options. Their functioning is also a priority.

We performed such tests as:

- Functional Testing
- Static Testing
- Black box testing

5.3.3 Test Preparation

This project presents a step-by-step testing, which is an efficient and organized method of the testing process. Thus, the tests were performed after following and evaluating the customer requirements, structured as user stories.

We made sure that testing is possible by addressing questions such as: ''Who will use the application?'', ''Where will certain functionalities be accessed?'', ''What do we expect to get at the end of this operation?'', ''What needs to happen next?''.

We have also made sure that we are provided with the necessary information about the product, the necessary tools have been established, as well as the corresponding documents (Test Plan).

5.3.3.1 Entry Criteria

Entry Criteria is a set of conditions and requirements that are made in order to provide a favourable testing environment, before starting the testing process.

In this project, we approached entry criteria such as:

- The software product is developed and ready for testing;
- Customer requirements have been established and analysed;
- Test environment is prepared ;
- Material and human resources have been secured for the test;
- Test Cases have been written and ready to run.

5.3.3.2 Exit Criteria

Exit Criteria is a set of conditions and requirements that must be met before declaring the project or process complete. Providing these exit criteria helps us to complete the testing process as completely and correctly as possible by offering a possibility to evaluate the effectiveness of the testing performed.

In this project, we approached exit criteria such as:

- Ensuring the execution of all test cases;

- Meeting a deadline;

- All identified defects have been reported.

5.3.4 Test Execution

All functionalities assigned in this phase have been tested and ensured that all customer requirements are met and that a good user experience is ensured. Test Cases were executed and all defects were reported.

5.4 [Test Phase 2] - second sprint

5.4.1 Objective

The objective of the second phase is similar to that of the first phase, but dividing them into two clusters makes the testing process more efficient by making it easier to track the performance of the testing team in the process.

This phase approaches for testing the most important set of customer requirements: main menu . Through the sections addressed in the main menu, the user becomes aware of what the client offers and addresses, the purpose of that site's existence.

5.4.2 Scope

The scope of testing for Test Phase 2 is ensuring the functionality of the main menu options that allow the user to have contact and be informed about the organisation's actions.

We performed such tests as:

- Functional Testing
- Static Testing
- Black box testing

5.4.3 Test Preparation

The process, being similar to the first phase, involves the same points to be reached for the test process. The same as the ones mentioned at 5.3.3.

5.4.3.1 Entry Criteria

- The software product is developed and ready for testing;
- Customer requirements have been established and analysed;
- Test environment is prepared ;
- Material and human resources have been secured for the test;
- Test Cases have been written and ready to run.

5.4.3.2 Exit Criteria

- Ensuring the execution of all test cases;

- Meeting a deadline;

- All identified defects have been reported.

5.4.4 Test Execution

All functionalities assigned in this phase have been tested and ensured that all customer requirements are met and that a good user experience is ensured. Test Cases were executed and all defects were reported.

6 TEST ENVIRONMENTS REQUIREMENTS

Test environments are a set, hardware and software configuration that allow test cases to be executed according to the test context.

There are different types of testing environments that help analyze various elements of a software program such as performance, user experience, security and more. The main test environments we can exemplify are:

- Component/ Unit Testing
- Integration Testing -Unit Integration
  - System Integration

-System Testing

- Acceptance Testing - Operational AC
  - User AC
  - Alpha-Beta Testing
  - A-B Testing

6.1 LOCAL ENVIRONMENT

This is the first environment a software product passes through. It serves as the workspace for the developer of the product. Here, changes can be made without impacting the user or customer experience. Typically, this environment is individual and patronized by the developer.

6.2 TEST ENVIRONMENT

This is the environment in which the quality and functionality of the product is checked. It is in this environment where the testing process takes place. This is where we check the behaviour of the code written by the developers, report bugs and test subsequent changes. On this environment, access is more permissive than on the local environment. Access is allowed to the test team, developers, non-technical team, customers, and others.

6.3 PRE-PRODUCTION TEST ENVIRONMENT

This is the closest development environment to the production environment. In this environment, the product is at a stage almost similar to that in the production environment and allows identification of defects that may impact the user experience.

6.4 Production Environment

The production environment is the final phase through which the developed product may be. It is basically the environment where the user interacts and experiences the final product. Although it is a final stage in which the product must comply as much as possible with the customer's requirements and provide a satisfying experience, a complete test cannot be achieved without the user's help. Feedback on the developed product can only come when the user is provided with what has been worked on.

7 TEST DATA REQUIREMENTS

The data used to run the test cases were generated manually. The simplistic utility of this site did not require the use of complex data, so it was used:

- Valid Data - data such as name, surname, password, amount of money to verify the functionality of the product according to the requirements;
- Invalid Data - set of invalid data to check the behaviour of the application when using data that cannot be supported from a business point of view;
- Boundary Condition Data Set - to test the behaviour and messages displayed when using values that do or do not respect a range suggested by the merchant;
- **Decision Data Table Set** - sets of options to test the behaviour and how the application reacts by addressing pre-defined scenarios (such as choosing a certain languagy or a certain currency for a transaction) .

8 TESTING TOOLS & TECHNIQUES

The tools inside the testing discipline are in a permanent process of evaluation and customization so that they can offer the best solution in the right context; however the tools described in the following sections are proposed for use in this project.

8.1 Requirements & Use Case Management

For this project we used Jira, a management tool that approaches the Agile methodology. Jira approaches projects by using transparency, efficiently organizing user stories by dividing them into sprints, and assigning certain teams or members. It also allows for the estimation of US by assigning story points.
The flexibility of creating a work board allows real-time visualization of the project status by easily

swapping USs in the appropriate columns.
As a list of attributes of main interest, Jira is characterized by the following functionalities:

- Issue tracking
- Customizable workflows
- Estimation & work logging
- Progress reporting
- Scrum boards
- Kanban boards
- Project-level permissions
- Project backlogs
- Email notifications
- Roadmaps

For this project, the Scrum framework was used.

8.2 Test Management & Defect Tracking

For this current project, we used Test Case Lab, a test management tool where the user can:

-Create test cases with automatic generation of an ID
- Run the test cases and give the status of passed/failed/blocked
- Create test runs based on existing test plans or add test cases manually
- Assign certain TCs to other members
-Create new projects within one account
-Export your test cases database to CSV file
-Import your test cases database to existing project from CSV file
-Copy the link of the TC and attaching it to your projects

The bolded functionalities have been used in this project, using Test Case Lab.

9 Testing Roles & Responsibilities

The following table shows the testing roles for the project, together with the individuals involved in the testing effort.

Activity	Responsibility/Ownership	Name
Test Plan Creation	[Test Manager]	Oana Alexandra Miu
Test Phase Plan Creation	[Test Lead]	Alex Palade
Test Management	[Development Company Test Manager & System Test Lead]	Oana Alexandra Miu, Alex Palade
[Test Analysis and Design]	[Development Company Test Engineers]	Diana Bulai, Madalin Coman
[Test Preparation, Execution & Results]	[Development Company Test Engineers]	Milut Alina, Alina Silochi, Madalin Coman, Madalin Duplea, Diana Bulai
[Test Defect Submission]	[Development Company Test Engineers]	Alina Silochi, Madalin Duplea
[Test Summary Reporting]	[Development Company Test Manager & Test Lead]	Oana Alexandra Mau, Alex Palade
[Test Completion Reporting]	[Development Company Test Manager & Test Lead]	Diana Bulai, Alex Palade

10 TEST MANAGEMENT

Overall responsibility for the Testing Project will be with the Development Company Test Manager Oana Alexandra Miu. Day-to-day Test Management will be the responsibility of the Development Company Test Lead Alex Palade.

One of the key roles for the Test Management Team is to work closely with the Project Manager, Technical Lead & Lead Business Analyst to ensure that testing is integrated and executed efficiently. On that basis the Test Management Team will attend the weekly Project Managers meeting with the

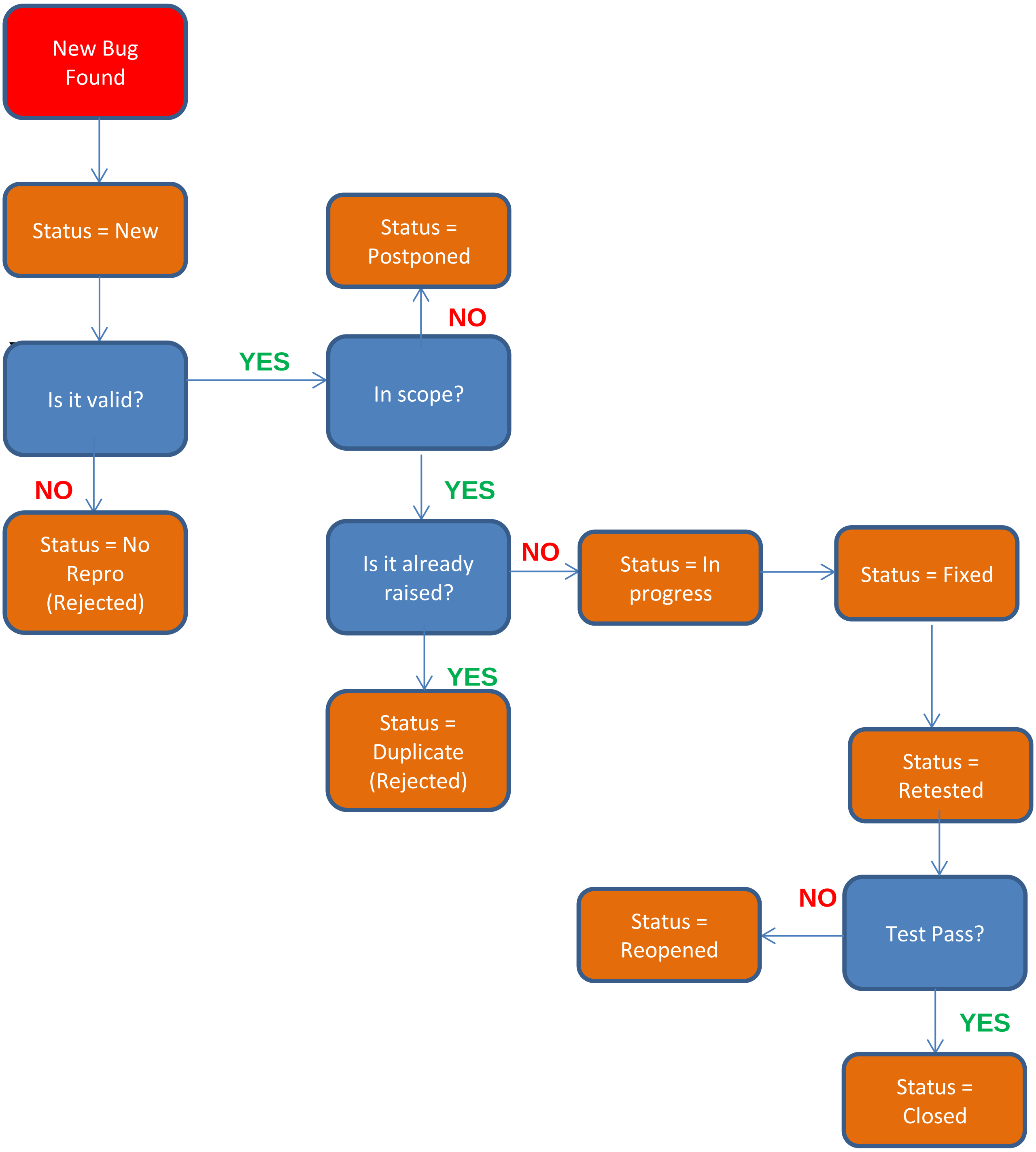
Technical Team Lead and the Lead Business Analyst where all actions, issues and risks are reviewed and progressed.

In addition, daily testing update meetings will be undertaken each morning, whereby each tester will be required to state their progress (what they did yesterday, what they intend to do today and any issues they have preventing them from progressing)

At the end of every Testing Phase, a Testing Review Board Meeting will be arranged. The Testing Review Board (TRB) is a review board working within a project environment assembled from the key stakeholders and decision makers for the Project. It is designed as a mechanism to review readiness for testing, testing status, and make formal strategic project decisions on testing and portfolio management.

The Project Team (Test Management, Project Management, Business Analysis and Development) will also attend a daily Defect Review Board Meeting. This is to be held at the end of each day, whilst Test Execution is in progress. The Defect Review Board (DRB) will assess errors and issues that arise during testing. The key purpose of the DRB is to prevent the testing becoming distracted by low priority issues that take them off the critical path. The DRB will carefully assess functional, technical and data implications, as well as business impact.

11 DEFECT MANAGEMENT

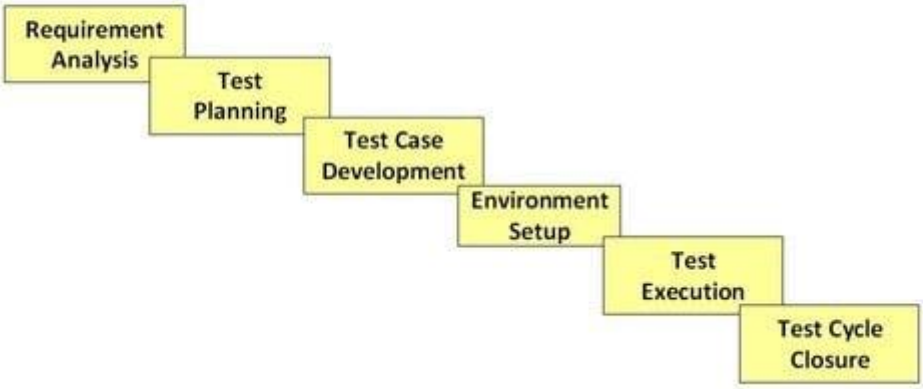


11.1 DEFECT MANAGEMENT PROCESS

- Subject must be concise, having a max amount of information with a minimum number of words.
- Defect comments must contain repro, expected and actual results.
- A good practice for a Tester is always to try to reduce the repro steps to the minimal number of steps; this is extremely helpful for the programmer who must find the defec

- Remember that the only person who can close a defect is the person who opened it. Anyone can resolve it, but only the person who detected the defect can really be sure that what he found is fixed.
- No Repro means that nobody could ever reproduce the defect. Programmers often use this when the defect report is missing the repro steps.
- Every build of the software that is given to testing should have a build ID number with revision number. A fixed defect report for the build should be made available to testers through a change log provided by PM, where environment, build number and list of changes is presented. In this case, the tester is informed which non-conformities are fixed in this build and does not have to retest the defect on a version of the software where it was not even supposed to be fixed.
- If you find bug reproduction very difficult to explain, having more steps, interaction with many systems provide more visual evidence by uploading pictures and videos.
- It is not a good practice to keep track of the non-conformities both in Bugzilla and in spreadsheets or word documents. Still if you must, then make sure to include the IDs of the non-conformities from Bugzilla and always use Bugzilla as a data source when creating a word document or a spreadsheet.
- Any Change request will have the "Enhancement".

12 TEST SCHEDULE



12.1.1 Test Phase 1

The following table shows the high level testing milestones for this phase of the project.

Ref No	Stage	Project Milestone	Due date
TM101	Test Preparation Stage	Phase Test Plan document completed.	20/06/2022
TM102	Test Preparation Stage	Test analysis completed on the detailed requirements and technical documentation.	21/06/2022
TM103	Test Preparation Stage	Test Conditions/Cases/Scripts completed and signed off.	26/06/2022

TM104	Test Execution Stage	Execution of Test scripts completed.	27/06/2022

12.1.2 Test Phase 2

The following table shows the high-level testing milestones for this phase of the project.

Ref No	Stage	Project Milestone	Due date
TM201	Test Preparation Stage	Phase Test Plan document completed.	28/06/2022
TM202	Test Preparation Stage	Test analysis completed on the detailed requirements and technical documentation.	05/07/2022
TM203	Test Preparation Stage	Test Conditions/Cases/Scripts completed and signed off.	07/07/2022
TM204	Test Execution Stage	Execution of Test scripts completed.	15/07/2022

13 REFERENCED DOCUMENTS

The following table identifies the documentation used for developing this Test Plan:

#	Document	Author	Description
1	Test Plan	Oana-Alexandra Miu	This document provides information regarding what specific testing will be completed on the Project.