**“Team” page update Test Plan**0.1

“Team” page update

Test Plan

Version 1.0

25 July 2024

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Modifier** | **Comment** |
| 2024 | 1.0 | M.A. Ermolenko | Original version |

Contents

[Introduction 4](#_Toc172710812)

[Purpose 4](#_Toc172710813)

[Background 4](#_Toc172710814)

[Scope 4](#_Toc172710815)

[Risks 5](#_Toc172710816)

[Project Identification 7](#_Toc172710817)

[Requirements for Test 7](#_Toc172710818)

[Scope of Testing 7](#_Toc172710819)

[Functional Testing 7](#_Toc172710820)

[Browser Compatibility Testing 9](#_Toc172710821)

[User Interface Testing 10](#_Toc172710822)

[Performance Testing 10](#_Toc172710823)

[Security and Access Control Testing 10](#_Toc172710824)

[Test Strategy 10](#_Toc172710825)

[Objectives 10](#_Toc172710826)

[Test Types and Levels 11](#_Toc172710827)

[Test Environments 12](#_Toc172710828)

[Test Data 12](#_Toc172710829)

[Test Execution and Reporting 12](#_Toc172710830)

[Entry and Exit Criteria 12](#_Toc172710831)

[Tools 12](#_Toc172710832)

[Resources 13](#_Toc172710833)

[Roles 13](#_Toc172710834)

[System 13](#_Toc172710835)

[Deliverables 14](#_Toc172710836)

[Test Cases 14](#_Toc172710837)

[Bug Reports 15](#_Toc172710838)

[Test Evaluation Report 15](#_Toc172710839)

[Test Metrics 16](#_Toc172710840)

[Plan of Action for Requirement Changes 17](#_Toc172710841)

[Objective 17](#_Toc172710842)

[Change Management Process 17](#_Toc172710843)

[Implementation 17](#_Toc172710844)

[Monitoring and Reporting 18](#_Toc172710845)

[Post-Implementation Review 18](#_Toc172710846)

[Summary 18](#_Toc172710847)

[Workflows 19](#_Toc172710848)

[Requirement changes 19](#_Toc172710849)

[Defects and Tasks 20](#_Toc172710850)

# Introduction

## Purpose

*This Test Plan document for the project “Team” page update supports the following objectives:*

* *Identify existing project information and the software components that should be tested*
* *List the recommended Requirements for Test (high level)*
* *Recommend and describe the testing strategies to be employed*
* *Identify the required resources and provide an estimate of the test efforts*
* *List the deliverable elements of the test project*
* *Provide processes workflows*

## Background

*Dataart.team - internal page of the site with information about employees. Currently the functionality has the following disadvantages:*

*1) The need to manually add and remove the data of new and old employees*

*2) The need to edit photos for the website manually*

*The current project is an upgrade of existing functionality in order to eliminate the shortcomings described above*

*To achieve this, the following approach is proposed:*

*1) Use the PM database (http://pm.dataart.com) as a source of employee data*

*2) Use the photos from the PM system without further editing for the DataArt Team website*

## Scope

The overall purpose of the testing procedures is to ensure that the update of “Team” page meets all of requirements. The purpose of this document is to describe the overall test plan and strategies that might be used during testing.

Current project purpose is to add required functionality to the existing resource.

The testing process contains following levels – Unit (by developers), Integration, System, Regression, and Acceptance (by the group of the company employees).

* During the testing, the QA Team should verify the following features
* Displaying personal data on the DataArt Team portal for each employee
* DataArt Team website structure and navigation

(The website has 4 types of pages: home page, Office page, Employee personal page, Search results)

## Risks

**Product risks**

**1. Data Synchronization Issues**

* **Risk:** Delays or failures in synchronizing employee data from the PM system to the DataArt Team website.
* **Impact:** Outdated or incorrect employee information displayed on the website.
* **Mitigation:**
  + Implement and test robust data synchronization mechanisms.
  + Monitor synchronization logs and alerts for failures.
  + Perform regular synchronization tests and manual checks.

**2. Inconsistent Data Formatting**

* **Risk:** Data entered in the PM system not displaying correctly on the website due to formatting issues, especially with HTML content.
* **Impact:** Poor user experience due to improperly formatted employee descriptions, preferences, and hobbies.
* **Mitigation:**
  + Validate and sanitize input data in the PM system.
  + Test different HTML content thoroughly in the PM system and ensure it displays correctly on the website.

**3. Photo Display Issues**

* **Risk:** Employee photos may not display correctly.
* **Impact:** Incomplete appearance of employee profiles.
* **Mitigation:**
  + Ensure photos in the PM system meet the required specifications.
  + Test the photo upload.
  + Implement automatic resizing and formatting of photos if needed.

**4. Search Functionality Failures**

* **Risk:** Search functionality may not return accurate results or may fail under certain conditions.
* **Impact:** Users unable to find employee information efficiently.
* **Mitigation:**
  + Perform extensive testing of the search functionality with various input scenarios.
  + Ensure the search algorithm handles different languages and special characters correctly.

**5. Browser Compatibility Issues**

* **Risk:** Website may not function correctly across all specified browsers.
* **Impact:** Users may experience inconsistent behaviour or layout issues.
* **Mitigation:**
  + Test the website on all specified browsers (IE 8, 9; Firefox 3, 4; Chrome; Safari; Opera).
  + Use automated tools to check for cross-browser compatibility.
  + Implement fallbacks or polyfills for unsupported features in older browsers.

**6. Performance Bottlenecks**

* **Risk:** Slow performance due to high load or inefficient code.
* **Impact:** Poor user experience and potential downtime.
* **Mitigation:**
  + Conduct performance testing to identify and address bottlenecks.
  + Implement caching mechanisms to improve data retrieval times.

**7. Technical Integration Challenges**

* **Risk:** Integration between the PM system and the DataArt Team website may face technical challenges.
* **Impact:** Delays in project implementation and potential data inconsistencies.
* **Mitigation:**
  + Conduct thorough integration testing.
  + Collaborate closely with development teams to resolve integration issues.

**Project risks**

**1. Involved personnel training and Usability Issues**

* **Risk:** HR managers may not be familiar with the new system or may find it difficult to use.
* **Impact:** Incorrect or incomplete data entry leading to errors on the website.
* **Mitigation:**
  + Provide comprehensive training for HR managers.
  + Develop user-friendly interfaces and documentation.
  + Offer ongoing support and resources for HR managers.

**2. Unclear Requirements or Changes in Requirements**

* **Risk:** Unclear or evolving requirements may lead to misunderstandings or incomplete testing.
* **Impact:** Potential gaps in testing and missed defects.
* **Mitigation:**
  + Maintain clear and detailed documentation of requirements.
  + Communicate regularly with stakeholders to clarify requirements.
  + Implement a change management process to handle requirement changes.

## Project Identification

The table below identifies available documentation and other artefacts used for developing the Test Plan:

|  |  |
| --- | --- |
| **Document** | **Path** |
| Requirements Specification | <https://edu.dataart.com/assets/courseware/v1/>  9ab236521e7319b05bd3d716d98708e8/asset-v1:QA+QAC05+2017+type@asset+block/Technical\_task\_ENG.docx |
| Design Specifications | <https://edu.dataart.com/assets/courseware/v1/>  9ab236521e7319b05bd3d716d98708e8/asset-v1:QA+QAC05+2017+type@asset+block/Technical\_task\_ENG.docx |

# Requirements for Test

The chapters outlined below identify the items (use cases, functional requirements, non-functional requirements) that are identified and approved as the main targets for testing procedures.

This list represents what actually will be tested. Details on each test will be determined later as the Test Cases are identified and the Test Procedures are developed.

## Scope of Testing

**In scope**

The chapters outlined below identify the items (use cases, functional requirements, non-functional requirements) that are identified and approved as the main targets for testing procedures.

This list represents what actually will be tested.

**Out of scope**

Due to the fact that PM is an independent product, this project does not involve security testing, functionality testing, performance testing or other tests except described in this test plan. Also, dataart.team pages will be tested only within the framework of the added functionality, according to the list of tests specified in this Test plan.

This means that load and stress testing are out of the scope assuming that these tests were carried out for platforms whose functionality is extended by this project, as well as accessibility testing, DB integrity testing and mobile platforms compatibility testing since there is no such requirements in the specification.

## Functional Testing

**Editing an employee’s personal data in PM by HR-manager:**

Preconditions:

* authorization under the HR manager account
* access to the employee’s test page in PM

Tests:

* Verify Surname in Russian text field
* Verify First Name: in Russian text field
* employee description in Russian text field with the possibility to enter HTML and preview it
* Verify employee description in English text field with the possibility to enter HTML and preview it
* Verify employee preferences in Russian text field with the possibility to enter HTML and preview it
* Verify employee preferences in English text field with the possibility to enter HTML and preview it
* Verify list of hobbies in Russian text field with the possibility to enter HTML and preview it
* Verify list of hobbies in English text field with the possibility to enter HTML and preview it
* Verify links to profiles in social networks text field with the possibility to enter URL

**Displaying personal data on the DataArt Team portal for each employee:**

Preconditions:

* Create at least 2 test accounts in PM
* Edit or enter data under the HR manager account
* Get access to data from several real accounts in PM from different offices (at least 3)

Tests:

For URL <http://team.dataart.ru>

* Verify Surname, First Name of tracking accounts match corresponding data in Russian in PM
* Verify employees’ photos of tracking accounts match corresponding photos in PM
* Verify that offices where the employees work match corresponding data in Russian in PM
* Verify day and month of birth of tracking accounts match corresponding data in Russian in PM
* Verify employee description of tracking accounts match corresponding data in Russian in PM
* Verify employee preferences of tracking accounts match corresponding data in Russian in PM
* Verify list of hobbies of tracking accounts match corresponding data in Russian in PM
* Verify links to profiles on social networks of tracking accounts match corresponding data in PM

For URL <http://team.dataart.com>

*Similar tests as described above but for data in English, where applicable*

**DataArt Team website structure and navigation:**

Important: Tests below should be performed for both URL’s <http://team.dataart.com> *and* <http://team.dataart.ru>

Home Page:

* Verify that 30 employees are displayed randomly.
* Ensure the design layout matches the specified requirements.

Office Page:

* Confirm that all employees from a specific office are listed correctly.
* Check that the URL includes the office name in English (e.g., <http://team.dataart.ru/offices/voronezh/> or <http://team.dataart.com/offices/voronezh/>).

Employee Personal Page:

* Verify that the full information of the employee is displayed as per the design layout.
* Ensure the URL includes the employee's first name and surname in English (e.g., <http://team.dataart.ru/staff/eugene-goland/> or <http://team.dataart.com/staff/eugene-goland/>)).

Search Results Page:

* Validate the search form functionality with fields for name/surname and birth date.
* Ensure search results display employee photo, name/surname, office, and birth date.
* Test pagination controls to ensure correct navigation through results.

## Browser Compatibility Testing

Important: Tests below should be performed for both URL’s <http://team.dataart.com> *and* <http://team.dataart.ru>

Preconditions:

* Internet Explorer 8, 9; Firefox 3, 4; actual versions of Chrome, Safari, Opera are available

Tests:

* Verify that functionalities (described above), layouts, and styles are consistent across following browsers:
  + Internet Explorer 8, 9
  + Firefox 3, 4
  + Chrome
  + Safari
  + Opera

## User Interface Testing

Preconditions:

* Pixel perfect extension is installed

Tests:

* Using pixel perfect extension verify the positioning of elements on the page matches the layout within 5 pixels
* Verify that the website's overall layout and design match the specified design requirements.
* Ensure consistent font styles, sizes, and colors across the website.
* Check for consistent spacing, padding, and alignment of elements.
* Verify that all images, including employee photos, are displayed correctly and are not distorted.
* Test the website's responsiveness on different browser window sizes

## Performance Testing

Important: Tests below should be performed for both URL’s <http://team.dataart.com> *and* <http://team.dataart.ru>

Tests:

* Verify that source data caching is implemented and effective.
* Ensure that PM remains operational during data updates to the DataArt Team website.
* Test data changes in PM and ensure they reflect on the website within 10 minutes.

## Security and Access Control Testing

Preconditions:

* Access to all roles (positions) accounts in PM

Tests:

* Verify by checking through all rolls (positions) that only HR and owner has access to PM-account personal account data

# Test Strategy

## Objectives

The primary objectives are:

* To ensure that the data synchronization from the PM database to the DataArt Team website is accurate and timely.
* To validate that all employee information is correctly displayed on the website.
* To verify the functionality, usability, and performance of the website across different browsers and devices.
* To identify and resolve defects efficiently through a structured workflow.

## Test Types and Levels

**2.1. Unit Testing**

* Performed by developers to test individual components.
* Focus on ensuring that each component functions correctly in isolation.

**2.2. Integration Testing**

* Verify that data synchronization between the PM system and the DataArt Team website works correctly.
* Test the integration of different modules within the website.

**2.3. Functional Testing**

* Validate all specified functional requirements.
* Ensure accurate data display, search functionality, and page navigation.

**2.4. User Interface (UI) Testing**

* Ensure consistency and correctness of the website's UI elements.
* Verify layout, design, and user interactions.

**2.5. Performance Testing**

* Test website performance meets specification requirements.
* Ensure the website performs well with multiple simultaneous users.

**3.6. Compatibility Testing**

* Test the website on different browsers and devices.
* Ensure consistent behavior and appearance.

**3.7. Security Testing**

* Verify that only account owners and HR have access to personal pages in PM and the added functionality did not create vulnerabilities.

**3.8. Regression Testing**

* Ensure that recent code changes have not adversely affected existing functionality.
* Execute a predefined set of test cases that cover key functionalities of the application.

**3.9 Acceptance testing**

* Organize trial testing of the updated version of the site among company employees with feedback

## Test Data

* Use real data and test profiles from the PM system for integration and functional testing.
* Create mock data for specific test scenarios to ensure comprehensive coverage.

## Test Execution and Reporting

* **Test Planning**: Define test cases based on requirements and functional specifications.
* **Test Execution**: Execute test cases in the defined test environments.
* **Defect Reporting**: Log defects in the defect tracking system with detailed information.
* **Daily Stand-ups**: Regular meetings to discuss progress, issues, and blockers.
* **Test Reports**: Generate test reports to summarize testing activities, defects, and status.

## Entry and Exit Criteria

**Entry Criteria:**

* All development tasks are completed.
* Test environments are set up and configured.
* Test data is prepared.

**Exit Criteria:**

* All test cases have been executed.
* Critical and high-severity defects are resolved.
* Test coverage meets the defined goals.
* Acceptance criteria are met.

## Tools

The following tools will be employed for this project:

|  |  |
| --- | --- |
|  | **Tool** |
| Test Management | TestRail |
| Defect Tracking | JIRA |
| Tool for browsers-compatibility testing | BrowserStack or Sauce Labs |
| Tool for CD/CI | Appveyor or Jenkins |

# Resources

*This section presents the recommended resources for the “Team” page update project.*

## Roles

This table shows the staffing assumptions for the test effort:

|  |  |  |
| --- | --- | --- |
| **Human Resources** | | |
| **Role** | **Minimum Resources Recommended**  **(number of full-time roles allocated)** | **Specific Responsibilities or Comments** |
| QA Lead | 1 | Provides management oversight.  Responsibilities include:   * planning and logistics * agree mission * identify motivators * acquire appropriate resources * present management reporting * advocate the interests of test * evaluate effectiveness of test effort * identify test ideas * define test details * determine test results * document change requests * evaluate product quality |
| QA-engineer | 2 | Implements and executes the tests.  Responsibilities include:   * implement tests and test suites * execute test suites * log results * analyze and recover from test failures * document incidents |

## System

The following table sets forth the system resources for the testing project.

|  |  |
| --- | --- |
| **Testing resources** | |
| **Resource** | **Path** |
| Development server | *…* |
| Test server | *…* |
| Production server | <http://team.dataart.ru>, <http://team.dataart.>com |
| Test repository | *…* |
| Issue tracking system | *Link to the project in Jira* |
|  |  |

**Project Milestones**

The testing of the “Team” page update project should incorporate the test activities for each test effort that is identified in the previous sections.

The separate project milestones should be identified to communicate the project’s status accomplishments.

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone task** | **Effort** | **Start date** | **End date** |
| Plan test |  |  |  |
| Design test |  |  |  |
| Implement test |  |  |  |
| Execute test |  |  |  |
| Evaluate test |  |  |  |

# Deliverables

*In this section, list various documents, tools, and reports that will be created, and provide information by whom, to whom, and when delivered.*

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **Owner** | **Due date** |
| Plan test | QA-lead |  |
| Test Cases | QA-engineers |  |
| Bug Reports | QA-engineers |  |
| Test Evaluation Report | QA-lead |  |

## Test Cases

Each test case has to contain the following fields:

* Test Case ID;
* Test Case Description;
* Preconditions;
* Test Data;
* Test Steps;
* Expected Result;
* Environment Information;
* Assign to;
* Comments.

## Bug Reports

Jira is going to be used for the logging and tracking of individual defects. In case of each certain Bug Report the listed below fields are have to be filled:

* Project;
* Issue Type;
* Summary;
* Priority;
* Components;
* Affect Versions;
* Fix Versions;
* Assignee;
* Reporter;
* Environment;
* Description (Data, Steps to Reproduce, Actual Result, Expected Result);
* Attachment (optional);
* Labels (optional)
* Epic link;
* Sprint.

## Test Evaluation Report

Below is recommended structure for the Test evaluation report:

* Project Information
* Introduction
  + Purpose
  + Scope
  + Definitions
  + Overview
* Test Results
* Test Coverage
* Recommendations
* Additional information (optional)

## Test Metrics

* Requirement coverage.
* Defect Distribution.

# Plan of Action for Requirement Changes

## Objective

To establish a structured approach for managing and implementing changes to requirements during the testing phase of the DataArt Team website project. This plan ensures that changes are effectively communicated, assessed, and integrated without compromising the quality or timeline of the project.

## Change Management Process

**2.1. Requesting a Change**

* **Initiation**: Any change to the requirements should be formally initiated by stakeholders through a Change Request (CR) form.
* **Submission**: The CR form should be submitted to the Project Manager (PM) and documented in the project management system (e.g., JIRA, Confluence).

**2.2. Assessing the Change**

* **Initial Review**: Project manager with QA-lead perform an initial review to determine the validity and necessity of the change.

**2.3 Analysis stage**

* **Impact Analysis**: Conduct a detailed impact analysis to assess the implications of the change on the following:
  + Scope of the project
  + Test plan and strategy
  + Test cases and test data
  + Schedule and timelines
  + Resources and budget
* **Feasibility Assessment**: The development and test teams will evaluate the technical feasibility of the change.

**2.4. Approval Process**

* **Review Meeting**: Convene a meeting with key stakeholders, including the PM, Test Manager, development leads, and relevant business stakeholders, to discuss the impact analysis and feasibility assessment.
* **Decision Making**: Stakeholders will decide whether to approve, reject, or request modifications to the change request.
* **Documentation**: Document the decision and update the project management system accordingly.

## Implementation

**3.1. Update Documentation**

* **Requirements Document**: Update the requirements to reflect the approved changes.
* **Test Plan**: Revise the test plan and strategy to incorporate changes.
* **Test Cases and Scripts**: Update existing test cases and scripts, and create new ones if necessary.

**3.2. Communicate Changes**

* **Team Meeting**: Conduct a briefing session with the test team and other relevant parties to explain the changes and their impact.
* **Change Log**: Maintain a change log to track all changes made to the requirements and their status.

**3.3. Execute Testing**

* **Re-Testing**: Execute tests for the changed requirements.
* **Regression Testing**: Perform regression testing to ensure that the changes have not adversely affected existing functionalities.
* **Performance and Security Testing**: Re-execute performance and security tests if the changes impact these areas.

## Monitoring and Reporting

* **Progress Tracking**: Regularly track and monitor the progress of implementing the changes.
* **Status Reports**: Provide status reports to stakeholders, highlighting the impact of the changes, progress, and any issues encountered.
* **Defect Logging**: Log any defects identified during re-testing and regression testing in the defect tracking system.
* **Defect Resolution**: Address and resolve defects through the established defect management workflow.
* **Verification**: Verify the fixes for the defects and ensure they meet the requirements

## Post-Implementation Review

* **Evaluation**: Conduct a post-implementation review to evaluate the success of the change implementation process.
* **Feedback**: Gather feedback from the test team and stakeholders to identify any areas for improvement in the change management process.
* **Documentation**: Provide Test evaluation report and update the change management process as necessary.

## Summary

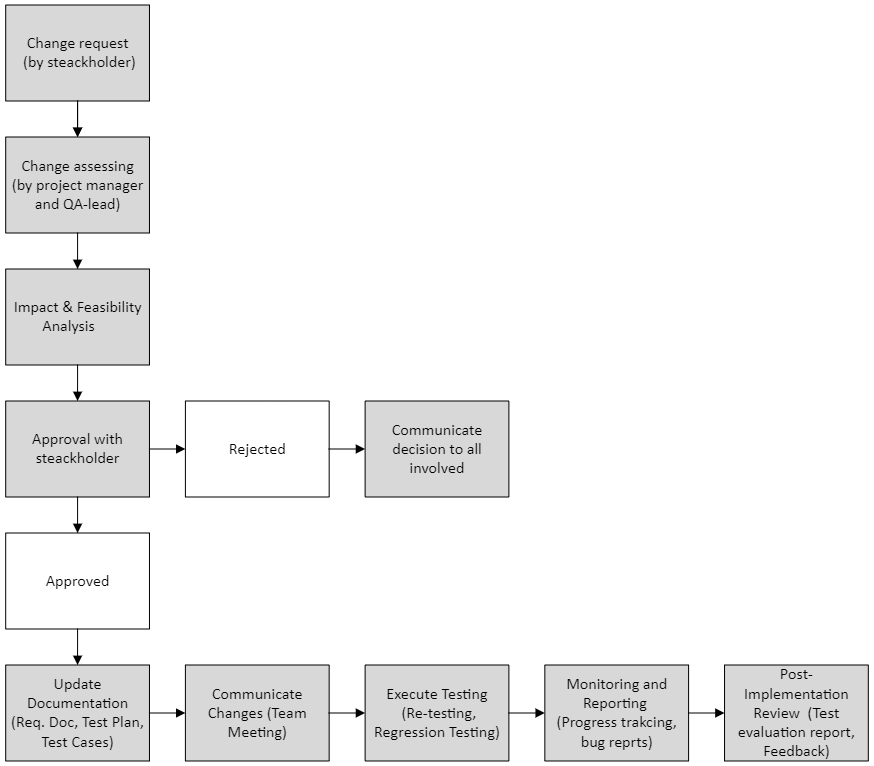
This action plan aims to provide a clear and structured approach to managing requirements changes during the testing phase.

A schematic representation of the main stages of this plan is presented in the corresponding section of the Workflow section

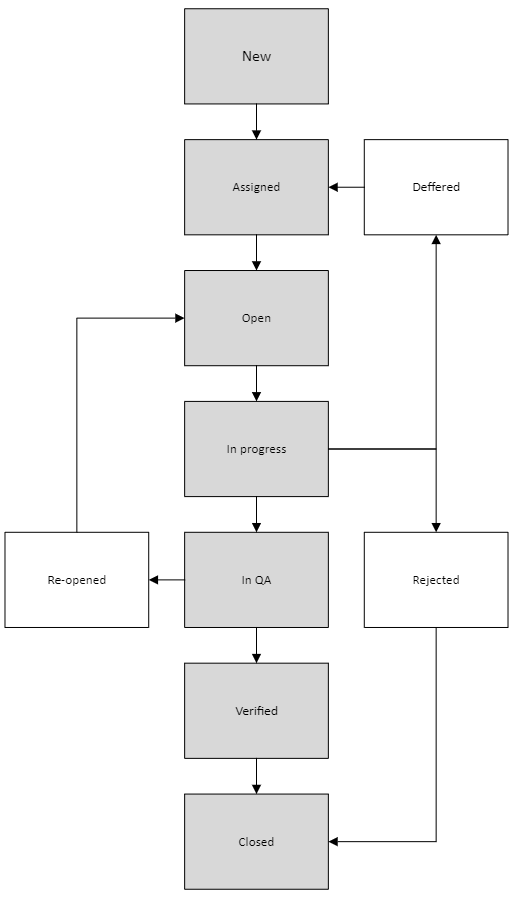
# Workflows

In this section presented diagrams of the main workflows

## Requirement changes



## Defects and Tasks



INTERNAL INFORMATION

Document info

|  |  |
| --- | --- |
| Document name: | “Team” page update |
| Document version: | *1.0* |
| Author: | *Maxim Ermolenko* |
| Author e-mail: |  |
| Support e-mail: |  |
| Date of last review: | *25 July 2024* |

Workflow for change request

(Untitled project)

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Modifier** | **Comment** |
| 25.07.2024 | 1.0 | M.A. Ermolenko | Original version |

Contents

[Purpose and scope 4](#_Toc172732929)

[Roles and responsibilities 4](#_Toc172732930)

[Change request status 5](#_Toc172732931)

[Entry criteria 6](#_Toc172732932)

[Tasks 6](#_Toc172732933)

[Evaluate change request 6](#_Toc172732934)

[Make change decision 6](#_Toc172732935)

[Implement the change 6](#_Toc172732936)

[Verify the change 6](#_Toc172732937)

[Exit criteria 6](#_Toc172732938)

[Change control status reporting 7](#_Toc172732939)

# Purpose and scope

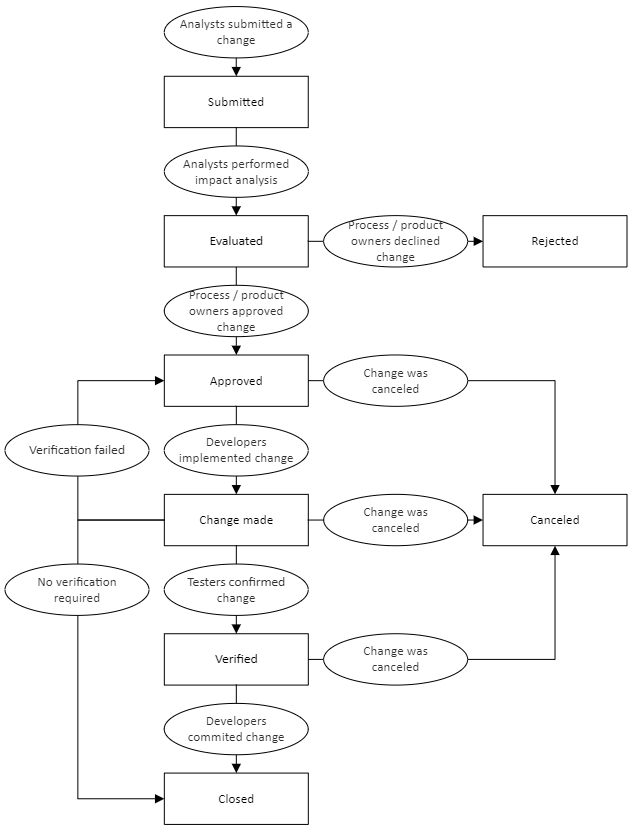
The purpose of this workflow is to provide a structured approach for handling change requests (CRs) in the project. It ensures that all changes are thoroughly evaluated, approved, implemented, and verified, integrating the efforts of analysts, developers, and testers. The scope includes all change requests related to requirements, features, and defects.

# Roles and responsibilities

This table shows the staffing assumptions for the test effort:

|  |  |  |
| --- | --- | --- |
| **Human Resources** | | |
| **Role** | **Minimum Resources Recommended**  **(number of full-time roles allocated)** | **Responsibilities** |
| Developer | 10 | * Product development in accordance with requirements and specifications * Conduct unit testing and E2E testing withing CI/CD * Implement approved changes |
| Analyst | 3 | * Change and supplement the requirement * Initiate and document change requests * Conduct initial impact analysis |
| Tester (1st team) | 3 | * Implements and executes the tests. * Document and report test results * Verify fixes for defects found by both testing teams |
| Tester (2nd team) | 3 | * Implements and executes the tests. * Document and report test results * verify fixes of the found by the team defects, after the first team’s confirmation |

# Change request status

\

# Entry criteria

* Change request form is completed and submitted by an analyst.
* Change request includes necessary details: description, rationale, impact analysis, and initial feasibility.

# Tasks

## Evaluate change request

* **Action**: Analysts review and document the change request, conduct an initial impact analysis.
* **Responsible**: Analysts
* **Outcome**: Detailed change request document.

## Make change decision

* **Action**: Conduct a review meeting with stakeholders to assess feasibility and impact.
* **Responsible**: Analytics
* **Outcome**: Approval or rejection of the change request.

## Implement the change

* **Action**: Assign tasks to developers, implement changes, conduct unit and integration testing.
* **Responsible**: Development team
* **Outcome**: Change implemented.

## Verify the change

* **Action**: Conduct system testing and verification by testing teams.
  + **First Testing Team**: Verifies fixes for defects found by both testing teams.
  + **Second Testing Team**: Conducts independent testing and verifies fixes of the defects, founded by the second team, after the first team's confirmation.
* **Responsible**: First Testing Team, Second Testing Team
* **Outcome**: Change verified and confirmed by relevant testing teams.

# Exit criteria

* All change request tasks are completed.
* Change is verified by the first and/or second testing team as required.
* Change request is marked as closed in the tracking system.

# Change control status reporting

* **Status Report**: Provide status report to stakeholders, highlighting the impact of the changes, progress, and any issues encountered
* **Stakeholder Meeting**: Meeting with stakeholders to review change control status and address any issues or delays.
* **Dashboard Update**: Real-time update on the project dashboard to track change request progress and status (if applicable).

Workflow for defects handling

(Untitled project)

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Modifier** | **Comment** |
| 25.07.2024 | 1.0 | M.A. Ermolenko | Original version |

Contents

[Purpose and scope 4](#_Toc172732929)

[Roles and responsibilities 4](#_Toc172732930)

[Change request status 5](#_Toc172732931)

[Entry criteria 6](#_Toc172732932)

[Tasks 6](#_Toc172732933)

[Evaluate change request 6](#_Toc172732934)

[Make change decision 6](#_Toc172732935)

[Implement the change 6](#_Toc172732936)

[Verify the change 6](#_Toc172732937)

[Exit criteria 6](#_Toc172732938)

[Change control status reporting 7](#_Toc172732939)

# Purpose and scope

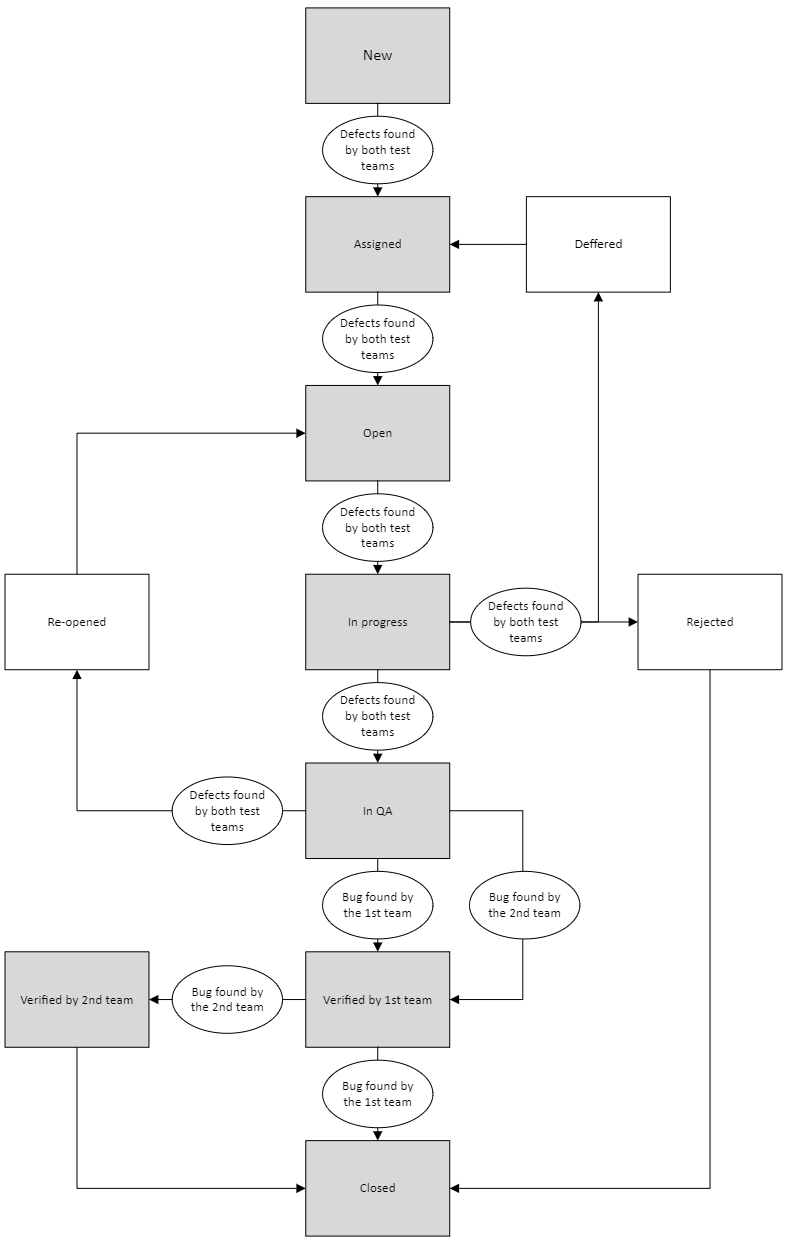
# The purpose of this workflow is to manage defects and ensure that all defects are properly, prioritized, fixed, and verified. The scope includes handling defects found during system testing by both testing teams and ensuring coordination between developers and testers.

# Roles and responsibilities

This table shows the staffing assumptions for the test effort:

|  |  |  |
| --- | --- | --- |
| **Human Resources** | | |
| **Role** | **Minimum Resources Recommended**  **(number of full-time roles allocated)** | **Responsibilities** |
| Developer | 10 | * Product development in accordance with requirements and specifications * Conduct unit testing and E2E testing withing CI/CD * Implement approved changes |
| Analyst | 3 | * Change and supplement the requirement * Initiate and document change requests * Conduct initial impact analysis |
| Tester (1st team) | 3 | * Implements and executes the tests. * Document and report test results * Verify fixes for defects found by both testing teams |
| Tester (2nd team) | 3 | * Implements and executes the tests. * Document and report test results * verify fixes of the found by the team defects, after the first team’s confirmation |

# Defect Workflow diagram



# Entry criteria

# Defects are identified during system testing by the first or second testing teams.

# Defects are documented with necessary details: description, steps to reproduce, expected and actual results, severity, and priority.

# Tasks

## Evaluate change request

* **Action**: Analysts review and document the change request, conduct an initial impact analysis.
* **Responsible**: Analysts
* **Outcome**: Detailed change request document.

## Make change decision

* **Action**: Conduct a review meeting with stakeholders to assess feasibility and impact.
* **Responsible**: Analytics
* **Outcome**: Approval or rejection of the change request.

## Implement the change

* **Action**: Assign tasks to developers, implement changes, conduct unit and integration testing.
* **Responsible**: Development team
* **Outcome**: Change implemented.

## Verify the change

* **Action**: Conduct system testing and verification by testing teams.
  + **First Testing Team**: Verifies fixes for defects found by both testing teams.
  + **Second Testing Team**: Conducts independent testing and verifies fixes of the defects, founded by the second team, after the first team's confirmation.
* **Responsible**: First Testing Team, Second Testing Team
* **Outcome**: Change verified and confirmed by relevant testing teams.

# Exit criteria

* All defects are resolved and verified.
* Defects are closed in the defect tracking tool.
* Verification is confirmed by the relevant testing team.

# Defect Control Status Reporting

* **Dashboard Update**: Real-time update on the project dashboard to track change request progress and status (if applicable).
* **Bug Reports:** Detailed reports on critical and high-priority defects, including steps to reproduce, screenshots, logs, and impact analysis.