DOKUMEN TEST PLAN WEBSITE ORANGEHRM

YUDHISTIRA KUSUMA

(202310067)

PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS INFORMATIKA DAN PARIWISATA

INSTITUT BISNIS DAN INFORMATIKA KESATUAN

BOGOR

2023

Table of Contents

Test Plan

Website “OrangeHRM”

Document Revision History

Date	Version	Description	Author	Reviewer	Approver
04.07	0.1	Test plan was created	Yudhistira K.		

1 INTRODUCTION

Customers naturally want the perfect website that works well and meets their needs. To fulfill this desire, a software tester is required to check and carry out tests on the website before being sent to customers. Testing can be done manually or automatically, and the website being tested must pass this test flawlessly. In order to improve their skills, a tester must constantly practice to excel in testing, and one way to achieve this is by using the OrangeHRM website.

To ensure consistent quality across websites, it is very important to adhere to the Test Plan. This document serves as an effective means of communication among team members and outlines the approach and methodology to be applied for unit, integration, and system testing "https://opensource-demo.orangehrmlive.com/". It covers various aspects such as objectives, test responsibilities, entry and exit criteria, scope, schedule, major milestones and overall approach. The Test Plan also provides a clear identification of the expected test results and defines what is included in the scope of the test and what is considered out of scope.

2 SCOPE

The document mainly focuses on testing the GUI and checking if the system successfully validates the entered data.

2.1 Modules to be tested

- Admin
- PIM
- Leave
- Time
- Recruitment
- My Info
- Performance
- Dashboard
- Directory
- Buzz

2.2 Modules not to be tested

1. Not other than mentioned above in section 2.1

3 QUALITY OBJECTIVES

3.1 Primary Objectives

The main purpose of testing is to ensure that the system fully meets all requirements, including quality requirements (both functional and non-functional) and meets the metrics specified for each quality requirement. Additionally, testing aims to validate that the system meets use case scenarios and maintains overall product quality. At the end of the project development cycle, users should be able to ensure that the project not only meets but exceeds all of their expectations as outlined in the requirements.

If there are modifications, additions, or deletions of the requirements documents, Functional Specifications, or Design Specifications, they will be documented and tested at the highest possible level of quality within the remaining project time and ability of the testing team.

3.2 Secondary Objectives

The secondary objectives of testing are to: identify and disclose all relevant problems and risks and document them in a test case document. As a goal, it requires careful and methodical

application testing to ensure that all areas of the system are thoroughly checked and, therefore, any issues (bugs) found are appropriately addressed.

4 TEST APPROACH

The approach used is Analytical because it corresponds to a requirements-based strategy, where analysis of requirements specifications forms the basis for planning, estimating and designing tests. Test cases will be created during exploratory testing. All the type of test is specified in the Test Strategy.

Teams should also use experience-based testing and fault guessing by leveraging the testers' skills and intuition, along with their experience with similar applications or technologies.

4.1 Test Automation

Automated unit testing is part of the development process, and UI smoke testing is also done to reject apps that break early in development.

5 ROLES AND RESPONSIBILITIES

Role	Staff Member	Responsibilities
QA		1. Understand requirements 2. Writing and executing Test cases 3. Preparing RTM 4. Reviewing Test cases, RTM 5. Defect reporting and tracking 6. Retesting and regression testing 7. Preparation of Test Data

6 ENTRY AND EXIT CRITERIA

6.1 Entry Criteria

- All necessary documentation, design, and requirements information shall be available to enable testers to operate the system and assess correct behavior.
- All standard software tools including the test kit should be successfully installed and functioning properly.

- Precise test data available.
- QA resources have fully understood the requirements.
- QA resources have good functionality knowledge.
- Review test scenarios, test cases, and RTMs.

6.2 Exit Criteria

- The specified coverage requirement level has been achieved.
- No high priority or severe bugs are left outstanding.
- All high risk areas have been fully tested, with only minor risks remaining.
- Schedule has been reached.

7 SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

7.1 Suspension criteria

- The build contains many serious defects which seriously or limit testing progress.
- Software/Hardware Problems.
- Assigned resources are not available when needed by the testing team.

7.2 Resumption criteria

Resumption will only occur if the problem(s) that caused the suspension is resolved.

8 TEST STRATEGY

8.1 QA role in test process

- Understanding Requirements:
  - Understanding of requirements will be done by QA
- Preparing Test Cases:

  QA will prepare test cases based on exploratory testing. This will cover all scenarios for requirements.
- Preparing Test Matrix:

  QA will prepare a test matrix that maps test cases to their respective requirements. This will ensure coverage for the requirements.

- Reviewing test cases and matrix:
    - Any comments or suggestions on test cases and test coverage will be provided by reviewer respective Author of Test Case and Test Matrix
    - Suggestions or improvements will be re-worked by author and will be send for approval
    - Re-worked improvements will be reviewed and approved by 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 run by respective QA on client's development/test site based on designed scenarios, test cases and Test data.
    - Test results (Actual Result, Pass/Fail) will be updated in the Defect Recording and Reporting test case document:

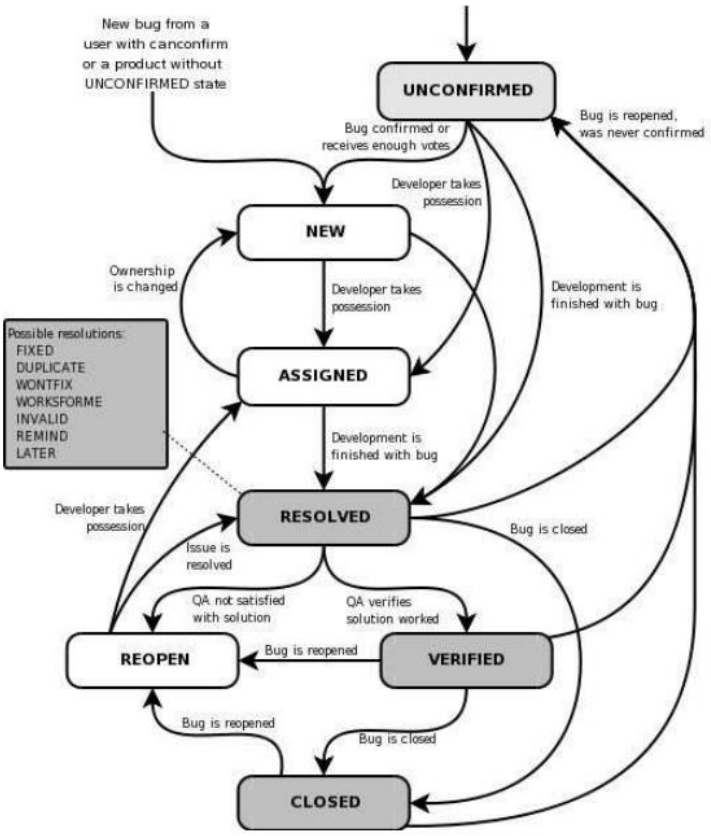
      QA will log the defects/bugs found during the execution of the test cases.
- Retesting and Regression Testing:

  Retesting for fixed bugs will be done by respective QA once it is resolved by respective developer and bug/defect status will be updated accordingly. In certain cases, regression testing will be done if required.

8.2 Bug life cycle

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

Bug life cycle for this project is as follows:



8.3 Testing types

Black box testing:

It is some time 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 that will fully exercise all functional requirements for a program.

GUI Testing:

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

Integration Testing:

Integration testing is systematic technique for constructing the program structure while conducting test 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 characteristic 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

8.4 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 nonrecoverable 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 component 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 to the user, prevents 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	Severity 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.
3	Fix When Have Time	The problem should be fixed within the time available. If the bug does not delay shipping date, then fix it.
4	Low Priority	It is not important (at this time) that these bugs be addressed. Fix these bugs after all other bugs have been fixed. Enhancements/ Good to have features incorporated just are out of the current scope.

9 RESOURCE AND ENVIRONMENT NEEDS

9.1 Testing Tools

Process	Tool
Test case creation	Microsoft Excel
Test case tracking	Microsoft Excel
Test case execution	Manual, Selenium
Test case management	Microsoft Excel

9.2 Configuration Management

- Code CM: Git

9.3 Test Environment

- Support level 1 (browsers):
    - Windows 11: Chrome (latest)

10 TEST SCHEDULE

Task Name	Start	Finish	Effort	Comments
Read Administrator's User Manual for OrangeHRM Version 3.0	05.06	06.06		
Test Planning	06.06	09.06		
Manual Testing	10.06	12.06		
Automation Testing	13.06	20.06		

APPROVALS

	Software Testing Lecturer
Name	
Signature	

TERMS/ACRONYMS

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

TERM/ACRONYM	DEFINITION
GUI	Graphical user interface
CM	Configuration Management
QA	Quality Assurance
RTM	Requirements Traceability Matrix