**TEST PLAN**

for

OrangeHRM Application

**Document Version**: 1.0  
**Prepared By**: Yaman Maharjan  
**Reviewed By**: [Reviewer Name]  
**Date**: 2024-10-22

**1. Introduction**

This test plan outlines the strategy and approach for testing the OrangeHRM application. The goal is to ensure that all modules function as intended, meet specified requirements, and maintain performance, usability, and security standards. The test plan covers functional, non-functional, integration, and regression testing.

### 2. ****Scope****

The testing will cover the following key modules of OrangeHRM:

* **Employee Management**
* **Leave Management**
* **Time and Attendance**
* **Recruitment Management**
* **Performance Management**
* **Benefits and Compensation**
* **Reporting and Analytics**

**3. Objectives**

The primary objectives of this test plan are:

1. Ensure that all OrangeHRM modules function according to the specified requirements.
2. Identify and log defects, inconsistencies, or issues.
3. Verify that the system meets user and business needs.
4. Test the performance, security, and usability of the application.
5. Ensure smooth integration between various modules and with third-party applications.
6. Validate that new updates or patches do not break existing functionality (regression testing).

**4. Test Strategy**

The test strategy includes the following types of testing:

* **Functional Testing**: Ensure that all functions and features in each module work as expected.
* **Exploratory Testing**: A hands-on approach that complements scripted testing by focusing on the tester's ability to explore the application without predefined steps. Testers use their experience, creativity, and intuition to uncover issues that may not be covered by test cases. The focus is on simultaneous learning, test design, and test execution.
* **Integration Testing**: Verify that different modules interact with each other correctly.
* **Performance Testing**: Test system response time, throughput, and overall performance under various loads.
* **Security Testing**: Test for security vulnerabilities, user roles, permissions, and data privacy.
* **Usability Testing**: Ensure the user interface is intuitive, accessible, and easy to navigate.
* **Regression Testing**: Re-examine modules after changes or updates to ensure existing functionality remains intact.
* **Compatibility Testing**: Ensure the system works across different browsers, devices, and platforms.

**5. Test Items**

The following modules and functionalities will be tested:

* **Employee Management**: Adding, editing, deleting, and searching employee records.
* **Leave Management**: Applying for leave, approval workflows, leave balances, leave types.
* **Time and Attendance**: Logging hours, clock-in/clock-out functionality, shift management, attendance reports.
* **Recruitment**: Job posting, applicant tracking, interview scheduling, and hiring.
* **Performance**: Setting performance goals, appraisal management, feedback, and reviews.
* **Benefits & Compensation**: Managing employee salaries, bonuses, and other benefits.
* **Reporting**: Generation and export of HR reports.

**6. Test Deliverables**

The deliverables for the testing process include:

1. **Test Plan**: This document outlining the test strategy.
2. **Test Cases**: Detailed test cases covering various functional and non-functional scenarios.
3. **Test Scripts**: Automated scripts for regression and performance testing (if applicable).
4. **Defect Reports**: Log of identified defects, bugs, or issues during testing.
5. **Test Execution Report**: A summary of test results including pass/fail status of executed test cases.
6. **Final Test Report**: A summary of the overall test process, results, issues, and recommendations.

**7. Test Environment**

The following environment will be used for testing:

* **Operating System**: Windows 10
* **Browsers**: Chrome, Firefox, Edge, Safari
* **Devices**: Desktop, Laptop, Mobile (for responsive testing)
* **Database**:
* **Server**:

### 8. ****Test Schedule****

|  |  |  |  |
| --- | --- | --- | --- |
| **Phase** | **Start Date** | **End Date** | **Responsible** |
| Test Planning | 2024-10-22 | 2024-10-22 | Yaman Maharjan |
| Test Case Preparation | 2024-10-22 | 2024-10-22 | Yaman Maharjan |
| Test Environment Setup | [Start Date] | [End Date] | [Name] |
| Functional Testing | [Start Date] | [End Date] | [Name] |
| Integration Testing | [Start Date] | [End Date] | [Name] |
| Performance Testing | [Start Date] | [End Date] | [Name] |
| Security Testing | [Start Date] | [End Date] | [Name] |
| Usability Testing | [Start Date] | [End Date] | [Name] |
| Regression Testing | [Start Date] | [End Date] | [Name] |
| Test Execution Completion | [Start Date] | [End Date] | [Name] |
| Final Test Report | [Start Date] | [End Date] | [Name] |

**9. Test Resources**

* **Test Manager**:
* **Test Analysts**: Yaman Maharjan
* **Automation Engineers**: Yaman Maharjan
* **Developers**:
* **Business Analysts**:

### 10. ****Entry & Exit Criteria****

#### Entry Criteria:

* Functional specifications are finalized.
* Application build is stable and available.
* Test cases are prepared and reviewed.

#### Exit Criteria:

* All planned test cases are executed.
* Critical defects are resolved.
* Test coverage is above the defined threshold (e.g., 95% of test cases passed).
* Final test report is reviewed and approved.

### 11. **Risk and Mitigation**

|  |  |
| --- | --- |
| **Risk** | **Mitigation** |
| Tight deadlines for testing | Prioritize test cases based on criticality and usage. |
| Incomplete or ambiguous requirements | Clarify requirements with business analysts or stakeholders. |
| Limited test resources (hardware, personnel) | Schedule testing in phases and allocate resources effectively. |
| Unstable test environment | Ensure proper configuration and regular monitoring. |

### 12. **Approval**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Signature** | **Date** |
| [Test Manager] | Test Manager |  |  |
| [Project Manager] | Project Manager |  |  |
| [QA Lead] | QA Lead |  |  |