

Test Plan:

Orange HRM Management

Prepared by:

Ayesha Sajid
1-03-2024



1. INTRODUCTION.....	9
1.1 TECHNOLOGY STACK.....	9
2. OBJECTIVES AND TASKS	10
2.1 OBJECTIVES	10
2.2 TASKS	10
2.2.1 TESTING TASKS	10
2.2.2 POST-TESTING TASKS.....	11
2.2.3 PROBLEM REPORTING	12
2.3 FEATURES TO BE TESTED:	12
3. SCOPE	14
3.1 GENERAL	14
3.2 TACTICS.....	14
4. TESTING STRATEGY	15
4.1 UNIT TESTING	16
4.2 SYSTEM AND INTEGRATION TESTING.....	16
TEAM MEMBERS	16
4.3 PERFORMANCE AND STRESS TESTING.....	17
TEAM MEMBERS	17
4.4 USER ACCEPTANCE TESTING	18
TEAM MEMBERS	18
4.5 BATCH TESTING.....	18
TEAM MEMBERS	19
4.6 AUTOMATED REGRESSION TESTING	19
TEAM MEMBERS	19
4.7 BETA TESTING	20
TEAM MEMBERS	20

5.	HARDWARE REQUIREMENTS.....	20
5.1	SERVER REQUIREMENTS	20
5.2	TESTING WORKSTATIONS	21
5.3	OTHER EQUIPMENT.....	22
6.	ENVIRONMENT REQUIREMENTS	22
6.1	MAIN FRAME	22
7.	TEST SCHEDULE	24
	Estimated Time and Schedule for Testing Tasks	24
	Usage Periods for Testing Resources	25
8.	CONTROL PROCEDURES.....	25
8.1	PROBLEM REPORTING	25
8.2	CHANGE REQUESTS	26
	Criteria for Including Changes	26
8.3	MODULES AFFECTED BY CHANGES	27
9.	FEATURES TO BE TESTED	27
9.1	LOGIN PAGE.....	27
9.2	DASHBOARD PAGE.....	27
9.3	MY INFO PAGE.....	28
10.	FEATURES NOT TO BE TESTED.....	28
10.1	LOCALIZATION AND INTERNATIONALIZATION	28
10.2	THIRD-PARTY INTEGRATIONS	28
10.3	LEGACY SYSTEM COMPATIBILITY	28
10.4	MOBILE APPLICATION TESTING.....	29
10.5	LEAVE MANAGEMENT	29
10.6	TIME AND ATTENDANCE.....	29
10.7	RECRUITMENT	29
10.8	PERFORMANCE MANAGEMENT	29
10.9	REPORTING	30
10.10	SYSTEM ADMINISTRATION.....	30
11.	RESOURCES/ROLES & RESPONSIBILITIES.....	30
11.1	PROJECT MANAGER.....	30
11.2	TEST MANAGER.....	31
11.3	TEST LEAD	31
11.4	TESTERS	31
11.5	AUTOMATION TEST ENGINEER	32
11.6	DEVELOPERS.....	32
11.7	OPERATIONS STAFF.....	32
11.8	BUSINESS ANALYST	33
11.9	QUALITY ASSURANCE DIRECTOR	33
11.10	STAKEHOLDERS	33

12.	SCHEDULES.....	34
12.1	TEST SCHEDULE.....	34
12.2	DELIVERABLES	34
13.	SIGNIFICANTLY IMPACTED DEPARTMENTS (SIDS).....	35
	Descriptions:	35
14.	DEPENDENCIES:.....	36
14.1	TEST-ITEM AVAILABILITY:	36
14.2	TESTING-RESOURCE AVAILABILITY:	36
14.3	DEADLINES	37
14.4	OTHER CONSTRAINTS	37
14.5	MITIGATION STRATEGIES.....	37
15	RISKS/ASSUMPTIONS.....	38
15.1	ASSUMPTIONS:.....	38
15.2	RISKS.....	39
16	TOOLS.....	40
16.1	AUTOMATION TOOLS	40
16.2	BUG TRACKING TOOLS	40
16.3	OTHER TOOLS.....	40
17	ENTRY AND EXIT CRITERIA	41
17.1	REQUIREMENT ANALYSIS	41
17.2	TEST PLANNING	41
17.3	TEST CASE DESIGN.....	42
17.4	TEST ENVIRONMENT SETUP.....	42
17.5	TEST EXECUTION	42
17.6	DEFECT REPORTING AND TRACKING	43
17.7	TEST CLOSURE.....	43
18.	APPROVALS.....	43

1. INTRODUCTION

In this document of the Test Plan for the Orange HRM application, our objective is to ensure the system's robustness, reliability, and performance across various functionalities related to human resource management. The Orange HRM application provides a comprehensive suite of HR tools to streamline and automate HR processes, enhance employee engagement, and optimize organizational workflows.

1.1 Technology Stack

The Orange HRM application utilizes the following technologies:

- **Frontend Framework:** React 17.0.2
- **JavaScript Library:** jQuery 3.5.1
- **Programming Language:** JavaScript
- **Database:** MySQL
- **Web Server:** Apache
- **Backend Framework:** PHP 7.4



Login

Username : Admin

Password : admin123

Username

Username

Password

Password

Login

Forgot your password?

OrangeHRM OS 5.6.1

© 2005 - 2024 OrangeHRM, Inc. All rights reserved.

2. OBJECTIVES AND TASKS

2.1 Objectives

1. **Ensure Functional Accuracy:** Validate that all features and functionalities of the Orange HRM application operate as intended, meeting specified requirements and providing a seamless user experience.
2. **Identify and Resolve Defects:** Detect any bugs or issues within the application, document them clearly, and work towards their resolution to ensure the stability and reliability of the system.
3. **Validate Integration and Interoperability:** Confirm that all modules within the Orange HRM application integrate smoothly with each other and with any third-party systems, ensuring consistent data flow and functionality.
4. **Assess Performance:** Evaluate the application's performance under various conditions to ensure it can handle expected load and usage without degradation in response times or user experience.
5. **Verify Security:** Ensure that the application is secure from vulnerabilities and threats, protecting sensitive employee and organizational data.
6. **Enhance Usability:** Assess the user interface and user experience to ensure that the application is intuitive, easy to navigate, and meets the needs of its users.

2.2 Tasks

2.2.1 Testing Tasks

1. **Requirement Analysis**
 - Review functional and non-functional requirements.
 - Understand business logic and user workflows.
2. **Test Planning**
 - Develop the test plan, including scope, objectives, resources, schedule, and deliverables.
3. **Test Case Development**
 - Create detailed test cases and scenarios.
 - Include both positive and negative test cases.

4. Test Environment Setup

- Configure necessary servers, databases, and other test environments.
- Prepare test data for various scenarios.

5. Manual Test Execution

- Execute test cases manually.
- Document test results and observed behavior.

6. Regression Testing

- Re-test the application after bug fixes or new features are added.
- Ensure existing functionality is not affected.

7. Performance Testing

- Assess application performance under various load conditions.
- Identify any performance bottlenecks.

8. Security Testing

- Conduct security assessments to identify vulnerabilities.
- Test for common security threats like SQL injection, XSS, etc.

9. Usability Testing

- Evaluate the user interface and user experience.
- Ensure the application is intuitive and user-friendly.

2.2.2 Post-Testing Tasks

1. Defect Reporting and Tracking

- Log identified defects in a tracking system.
- Prioritize defects based on severity and impact.
- Monitor defect resolution status.

2. Test Reporting

- Compile test reports summarizing activities, results, and findings.
- Highlight key metrics and statistics from testing.

3. Final Validation

- Conduct a final round of testing to ensure all issues are resolved.
- Perform a thorough review before deployment.

4. **Stakeholder Sign-off**

- Obtain formal approval from stakeholders after successful testing.
- Ensure all critical issues are addressed and resolved.

2.2.3 **Problem Reporting**

1. **Defect Logging**

- Document defects with clear descriptions, steps to reproduce, and screenshots if necessary.
- Assign defects to the appropriate team members for resolution.

2. **Issue Tracking**

- Use a defect tracking system to monitor the status of reported issues.
- Update defect status as they are resolved, retested, and closed.

3. **Communication**

- Regularly communicate with the development team and stakeholders about defect status.
- Discuss critical issues in stand-up meetings or defect triage sessions.

2.3 **Features to be Tested:**

1. **Login:** Ensure the login functionality is working correctly, including validation of credentials and handling of login errors.
2. **Dashboard Page:** Verify that the dashboard page displays accurate and relevant information and functions as intended.
3. **Create Account:** Test the account creation process to confirm users can successfully register new accounts and that all related processes are functional.



Login

Username : Admin
Password : admin123

Username

Password

Login

[Forgot your password?](#)



Dashboard



Mojisola Otusheso

Search

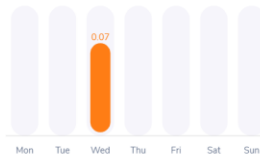
- Admin
- PIM
- Leave
- Time
- Recruitment
- My Info
- Performance
- Dashboard**
- Directory
- Maintenance
- Claim

Time at Work

Punched Out
Punched Out: Today at 06:37 PM (GMT 5.5)

0h 4m Today

This Week
Jul 22 - Jul 28



Quick Launch

My Actions

- (1) Timesheet to Approve
- (1) Pending Self Review
- (1) Candidate to Interview

Buzz Latest Posts



PIM



Mojisola Otusheso

Search

- Admin
- PIM
- Leave
- Time
- Recruitment
- My Info**
- Performance
- Dashboard
- Directory
- Maintenance
- Claim

Mojisola Otusheso

Personal Details

Contact Details

Emergency Contacts

Dependents

Immigration

Job

Salary

Personal Details

Employee Full Name*

Mojisola

Olushola

Otusheso

Employee Id

2428

Other Id

222

Driver's License Number

12345

License Expiry Date

2023-18-10

Nationality

Nigerian

Marital Status

Single

Date of Birth

2023-21-10

Gender

☐ Male

☒ Female

3. SCOPE

3.1 General

The initial phase will include all 'must have' requirements for the OrangeHRM management system. These and any additional requirements that are included must all be tested. At the end of Phase 1, a tester must be able to:

1. Successfully log in using valid credentials.
2. Access the dashboard page and navigate through its features.
3. View, edit, and save personal information on the My Info page.
4. Validate the security and privacy measures on each page.
5. Ensure the functionality of input fields, buttons, and links across the login, dashboard, and My Info pages.

As the team works with the product, they will define the needs for subsequent phases. Load testing will not be considered part of this project since the user base is known and not an issue. Rewriting, moving, or porting existing test cases from existing documents is not considered part of this project.

3.2 Tactics

The following strategies will be employed to achieve the objectives outlined in the Scope section:

- **Requirement Analysis and Documentation**
 - Review the detailed requirements for the login page, dashboard page, and My Info page.
 - Document the expected behaviors and outcomes for each feature.
- **Test Case Development**
 - Create comprehensive test cases covering all functionalities of the login page, dashboard page, and My Info page.
 - Include both positive and negative test cases to ensure robust validation.
- **Test Environment Setup**
 - Configure the testing environment to mirror the production setup.

- Ensure all necessary data and user accounts are prepared for testing scenarios.
- **Manual Test Execution**
 - Execute the developed test cases manually.
 - Record the outcomes and any deviations from expected results.
- **Defect Reporting and Tracking**
 - Log any identified defects in a tracking system with clear descriptions and steps to reproduce.
 - Prioritize and assign defects to the relevant development team members.
- **Regression Testing**
 - Conduct regression testing to ensure that fixes do not introduce new issues.
 - Revalidate all functionalities after bug fixes or updates.
- **Stakeholder Communication**
 - Notify key stakeholders and representatives from each functional area about the testing schedule and requirements.
 - Allot time in their schedules to assist with testing activities, provide necessary information, and address any issues that arise.
 - Use meetings, emails, and project management tools to ensure clear communication and coordination.
 - Provide regular updates on testing progress and any critical issues.
- **Post-Testing Review**
 - Conduct a review of the testing process and outcomes.
 - Document lessons learned and suggest improvements for future testing cycles.

4. TESTING STRATEGY

The overall approach to testing for the OrangeHRM management system includes a comprehensive testing strategy that ensures all functionalities of the login page, dashboard page, and My Info page are adequately tested. This will involve unit testing, system and integration testing, performance and stress testing, user acceptance testing,

batch testing, and automated regression testing. Each phase will include specific activities, techniques, and tools to achieve thorough validation.

The project is using an agile approach with weekly iterations. At the end of each week, the requirements identified for that iteration will be delivered to the team and tested. Exploratory testing will play a large part of the testing as the team is new to this type of tool and will be learning as they go. Tests for planned functionality will be created and added to TCT as we receive iterations of the product.

4.1 Unit Testing

Unit testing will focus on individual components of the OrangeHRM management system to ensure each part functions correctly in isolation. This will include testing the login form, input fields, buttons, and error messages.

Participants:

- Development Team
- QA Engineers

Methodology:

- Developers will write the initial unit test scripts as part of the development process.
- QA Engineers will review and enhance these scripts to cover all possible scenarios.
- Each unit test will be executed to verify that every component meets the defined requirements.
- The comprehensiveness of unit testing will be measured using code coverage tools to ensure all statements are executed at least once.

4.2 System and Integration Testing

System and integration testing will ensure that all modules and components of the OrangeHRM system work together seamlessly. This includes testing interactions between the login page, dashboard page, and My Info page.

Participants:

- QA Engineers
- System Integrators

Team Members

Participant Name	Role
NA	QA Engineer
NA	System Integrator

Methodology:

- QA Engineers will develop integration test scripts to cover interactions between different system components.
- System integrators will ensure all modules are correctly connected and data flows smoothly across the system.
- The testing will follow a predefined sequence of events, starting from logging in, navigating the dashboard, and accessing My Info.
- Any issues identified during integration testing will be documented and addressed promptly.

4.3 Performance and Stress Testing

Performance testing will assess the responsiveness and stability of the OrangeHRM system under normal and peak conditions. Stress testing will push the system beyond its operational capacity to identify breaking points.

Participants:

- Performance Test Engineers
- QA Engineers

Team Members

Participant Name	Role
NA	QA Engineer
NA	Performance Test Engineer

Methodology:

- Performance Test Engineers will develop scripts to simulate normal and peak user loads.
- Stress tests will involve gradually increasing the load until the system fails to understand its limits.
- Tools like JMeter or LoadRunner will be used to execute these tests.
- Test results will be analyzed to identify performance bottlenecks and areas for improvement.

4.4 User Acceptance Testing

User acceptance testing (UAT) will confirm that the OrangeHRM system is ready for operational use by comparing the system to its initial requirements.

Participants:

- End-Users (Customers)
- QA Team

Team Members

Participant Name	Role
NA	QA Engineer
NA	End-User

Methodology:

- End-users will execute predefined test scripts to validate the system's functionality against their requirements.
- The QA team will facilitate the testing process, provide necessary support, and collect feedback.
- Any issues or discrepancies identified during UAT will be logged and addressed before final deployment.

4.5 Batch Testing

Batch testing will validate the execution of batch processes and scheduled jobs within the OrangeHRM system.

Participants:

- QA Engineers
- System Administrators

Team Members

Participant Name	Role
NA	QA Engineer
NA	System Administrator

Methodology:

- QA Engineers will create test scripts to simulate batch processes.
- System administrators will ensure scheduled jobs are executed as expected.
- The results of batch processes will be validated to ensure accuracy and completeness.

4.6 Automated Regression Testing

Regression testing is the selective retesting of the OrangeHRM system to verify that modifications have not caused unintended effects and that the system still works as specified.

Participants:

- QA Engineers

Team Members

Participant Name	Role
NA	QA Engineer

Methodology:

- Automated test scripts will be developed to cover critical functionalities.
- These scripts will be executed after each code change to ensure no new issues are introduced.

- Tools like Selenium or QTP will be used for automated regression testing.

4.7 Beta Testing

Beta testing involves releasing the OrangeHRM system to a limited group of end-users to identify any final issues before the official launch.

Participants:

- Selected End-Users
- QA Engineers

Team Members

Participant Name	Role
NA	QA Engineer
NA	Selected End-User

Methodology:

- A beta version of the system will be deployed to selected end-users.
- QA Engineers will collect feedback and identify any issues reported by users.
- Necessary adjustments will be made based on user feedback before the final release.

5. HARDWARE REQUIREMENTS

To ensure a smooth testing process for the OrangeHRM management system, the following hardware requirements must be met. These requirements cover the servers, testing workstations, and other necessary equipment to support the testing activities.

5.1 Server Requirements

1. Application Server:

- **Processor:** Quad-core 2.0 GHz or higher
- **RAM:** 16 GB or higher
- **Storage:** 500 GB SSD or higher

- **Operating System:** Linux (preferred) or Windows Server 2016/2019
- **Network:** 1 Gbps Ethernet

2. Database Server:

- **Processor:** Quad-core 2.0 GHz or higher
- **RAM:** 32 GB or higher
- **Storage:** 1 TB SSD or higher
- **Operating System:** Linux (preferred) or Windows Server 2016/2019
- **Database:** PostgreSQL (preferred) or MySQL
- **Network:** 1 Gbps Ethernet

3. Web Server:

- **Processor:** Dual-core 2.0 GHz or higher
- **RAM:** 8 GB or higher
- **Storage:** 250 GB SSD or higher
- **Operating System:** Linux (Apache suggested) or Nginx
- **Network:** 1 Gbps Ethernet

5.2 Testing Workstations

1. QA Workstations:

- **Processor:** Quad-core 2.0 GHz or higher
- **RAM:** 16 GB or higher
- **Storage:** 256 GB SSD or higher
- **Operating System:** Windows 10 or macOS
- **Display:** 1920x1080 resolution or higher
- **Network:** 100 Mbps Ethernet or Wi-Fi

2. Developer Workstations:

- **Processor:** Quad-core 2.5 GHz or higher
- **RAM:** 32 GB or higher
- **Storage:** 512 GB SSD or higher

- **Operating System:** Windows 10 or macOS
- **Display:** 1920x1080 resolution or higher
- **Network:** 100 Mbps Ethernet or Wi-Fi

5.3 Other Equipment

1. Backup and Recovery:

- **Backup Storage:** 2 TB NAS or external hard drive
- **Backup Software:** Compatible with the operating systems and databases used

2. Network Equipment:

- **Router/Switch:** 1 Gbps throughput
- **Firewall:** Enterprise-grade firewall with intrusion detection and prevention capabilities

3. Peripherals:

- **Printers/Scanners:** Network-enabled multifunction devices
- **Uninterruptible Power Supply (UPS):** For critical servers and network equipment

4. Mobile Devices (for testing mobile access):

- **Smartphones/Tablets:** iOS and Android devices with up-to-date operating systems

6. ENVIRONMENT REQUIREMENTS

6.1 Main Frame

For the testing of the OrangeHRM management system, the following environment requirements must be met:

Necessary Properties:

- **Hardware:**
 - Servers: Minimum 2 servers for load balancing and redundancy
 - Client Machines: Minimum 5 client machines with standard configurations

- Network: High-speed LAN with backup connectivity options
- **Communications:**
 - Secure VPN access for remote testing
 - Reliable internet connection with a minimum bandwidth of 100 Mbps
- **System Software:**
 - Operating Systems: Windows Server 2016/2019, Linux (Ubuntu/CentOS)
 - Database: MySQL 5.7 or higher
 - Web Server: Apache 2.4 or Nginx 1.14
 - Application Server: Tomcat 8.5 or higher
- **Mode of Usage:**
 - Stand-alone and networked configurations to simulate different user environments
- **Additional Software and Supplies:**
 - Browsers: Latest versions of Chrome, Firefox, Edge
 - Development Tools: Eclipse, Visual Studio Code
 - Testing Tools: Selenium, JUnit, TestNG
 - Security: Antivirus software, firewall configurations, secure storage for test data

Desired Properties:

- **Advanced Hardware:**
 - High-performance SSDs for faster data access and retrieval
 - Additional servers for dedicated performance and stress testing
- **Enhanced Network:**
 - Redundant internet connections to avoid downtime
 - Advanced network monitoring tools
- **Advanced System Software:**
 - Containerization tools like Docker for isolated testing environments
 - Cloud-based infrastructure for scalability

- **Security:**
 - Two-factor authentication for accessing test environments
 - Encrypted data storage for sensitive test data
 - Hardware: Secure storage and disposal methods for any sensitive hardware components

Special Test Tools:

- Automated testing tools: Selenium WebDriver, JMeter for performance testing
- Code analysis tools: SonarQube
- Bug tracking tools: JIRA, Bugzilla

Source of Needs:

- **Internal Sources:**
 - Existing company resources like servers, network infrastructure, and software licenses
- **External Sources:**
 - Third-party providers for additional hardware and software needs
 - Cloud service providers for scalable testing environments

7. TEST SCHEDULE

Estimated Time and Schedule for Testing Tasks		
<i>Testing Task</i>	<i>Estimated Time</i>	<i>Schedule</i>
<i>Test Planning</i>	3 days	Week 1
Review Requirements Documents	2 days	Week 1
Create Initial Test Estimates	1 day	Week 1
Staff and Train New Test Resources	5 days	Week 1-2
First Deploy to QA Test Environment	1 day	Week 2
Functional Testing - Iteration 1	5 days	Week 2-3

Iteration 2 Deploy to QA Environment	1 day	Week 3
Functional Testing - Iteration 2	5 days	Week 3-4
System Testing	5 days	Week 4-5
Regression Testing	5 days	Week 5-6
User Acceptance Testing (UAT)	5 days	Week 6-7
Resolution of Final Defects	3 days	Week 7
Deploy to Staging Environment	1 day	Week 7
Performance Testing	5 days	Week 7-8
Release to Production	1 day	Week 8
Usage Periods for Testing Resources		
Resource	Period of Use	
QA Environment	Week 2 - Week 8	
Testing Tools (Selenium, JMeter)	Week 2 - Week 8	
Staff	Week 1 - Week 8	

8. CONTROL PROCEDURES

8.1 Problem Reporting

During the testing process, incidents will be documented to ensure a clear and structured approach to problem identification and resolution. The following steps outline the problem reporting procedure:

1. **Identify the Incident:** When an issue is encountered, the tester will document the details of the incident, including the steps to reproduce, expected results, actual results, and any error messages.
2. **Log the Incident:** Using the automated incident logging system (e.g., JIRA, Bugzilla), the tester will log the incident with all relevant information. Each incident will be assigned a unique identifier.
3. **Categorize the Incident:** Incidents will be categorized based on their severity (e.g., critical, major, minor) and type (e.g., functional, performance, UI).

4. **Notify Relevant Personnel:** The tester will notify the relevant team members (developers, QA lead) about the incident. Notification will include a link to the logged incident and any immediate actions required.
5. **Track the Incident:** The incident will be tracked through the automated system, with updates made as the issue is investigated and resolved. The tester will verify the fix and close the incident once resolved.
6. **Report and Review:** Regular incident reports will be generated to review the status and trends of logged incidents. This will help in identifying recurring issues and areas needing improvement.

8.2 Change Requests

To manage modifications to the OrangeHRM management software, the following change request procedure will be implemented:

1. **Submit Change Request:** Any stakeholder (e.g., tester, developer, end-user) can submit a change request using the designated change request form or system.
2. **Review Change Request:** The change request will be reviewed by the QA lead, project manager, and relevant team members to assess its impact on the current product. This review includes evaluating the necessity, benefits, and potential risks of the proposed change.
3. **Approve/Reject Change Request:** The change request will be approved or rejected based on predefined criteria such as alignment with project goals, resource availability, and potential impact on existing functionality.
4. **Implement Approved Changes:** Approved changes will be assigned to the development team for implementation. A timeline for the implementation will be established, and the change will be documented in the project tracking system.
5. **Test Changes:** Once the changes are implemented, they will be tested according to the defined testing strategy. This includes regression testing to ensure that the changes do not adversely affect existing functionality.
6. **Sign-off:** After successful testing, the changes will be reviewed and signed off by the QA lead and project manager. Any documentation affected by the change will be updated accordingly.
7. **Communicate Changes:** All stakeholders will be informed of the implemented changes, including any impacts on their workflows or usage of the system. Training or support will be provided if necessary.

Criteria for Including Changes

- **Urgency:** The change addresses a critical issue that affects the core functionality of the system.

- **Benefit:** The change provides significant improvements in usability, performance, or user satisfaction.
- **Feasibility:** The change can be implemented within the available resources and timeline.
- **Impact:** The change has minimal negative impact on existing functionality and does not introduce new risks.

8.3 Modules Affected by Changes

Changes that affect existing modules will be identified during the review process. These modules will be highlighted in the change request documentation, and specific regression tests will be conducted to ensure their continued functionality. Any dependencies or integrations with other modules will also be considered and tested accordingly.

9. FEATURES TO BE TESTED

The OrangeHRM Management system comprises various features that need to be thoroughly tested to ensure their functionality, performance, and reliability. The following is a comprehensive list of software features and combinations of features that will be tested:

9.1 Login Page

- **User Authentication:** Ensure users can log in with valid credentials.
- **Validation Messages:** Verify appropriate error messages for invalid credentials, empty fields, and incorrect formats.
- **Forgot Password:** Check functionality for password reset requests and email verification.
- **Login with Link:** Verify that users can log in using a secure link sent to their registered email.

9.2 Dashboard Page

1. **Navigation:** Ensure all menu items and links navigate to the correct pages.
2. **User Information Display:** Verify that user-specific information (e.g., name, role) is displayed accurately.
3. **Widgets and Reports:** Test the display and functionality of various widgets and reports on the dashboard.

4. **Notifications:** Ensure that notifications are displayed and can be marked as read or deleted.

9.3 My Info Page

- **Personal Details:** Verify the ability to view and update personal details, such as name, address, and contact information.
- **Attachments:** Test uploading, viewing, and deleting attachments (e.g., documents, images).
- **Emergency Contacts:** Ensure users can add, update, and delete emergency contact information.
- **Dependents:** Verify functionality for managing dependents' information.

10. FEATURES NOT TO BE TESTED

10.1 Localization and Internationalization

- **Multi-language Support:** Testing for language translations and localization features will not be conducted. The focus will be on the English language interface only.
- **Regional Settings:** Time zones, date formats, and other region-specific settings will not be part of this testing phase.

10.2 Third-party Integrations

- **External System Integrations:** Integration with external systems such as payroll, ERP, or other third-party software will not be tested.
- **API Testing:** Testing the API endpoints for external integrations will not be included.

10.3 Legacy System Compatibility

- **Backward Compatibility:** Ensuring compatibility with older versions of OrangeHRM or legacy systems is not within the scope of this test plan.
- **Data Migration:** Testing the migration of data from previous versions or other systems is excluded.

10.4 Mobile Application Testing

- **Mobile App Functionality:** Testing the functionalities of OrangeHRM mobile applications (iOS and Android) will not be part of this test plan. The focus will be solely on the web-based application.
- **Responsive Design:** Ensuring the web application's responsiveness on various mobile devices and screen sizes will not be included.

10.5 Leave Management

- **Apply for Leave:** Ensure employees can apply for leave and that requests are submitted correctly.
- **Leave Approval:** Verify the process for leave approval/rejection by managers.
- **Leave Balance:** Test the display and calculation of leave balances.
- **Leave Types:** Ensure all types of leave (e.g., annual, sick) are available and functioning correctly.

10.6 Time and Attendance

- **Punch In/Out:** Test the ability to punch in and out, and ensure accurate recording of attendance.
- **Timesheets:** Verify the submission, approval, and management of employee timesheets.
- **Overtime Calculation:** Ensure overtime hours are calculated and displayed correctly.

10.7 Recruitment

- **Job Vacancies:** Verify the creation and management of job vacancies.
- **Candidate Applications:** Test the process for submitting and reviewing candidate applications.
- **Interview Scheduling:** Ensure interviews can be scheduled and managed within the system.
- **Offer Letters:** Verify the generation and management of offer letters.

10.8 Performance Management

- **Performance Reviews:** Test the creation, management, and completion of performance reviews.

- **Goals and Objectives:** Ensure employees can set, view, and update goals and objectives.
- **Feedback and Ratings:** Verify the functionality for providing and viewing feedback and ratings.

10.9 Reporting

- **Standard Reports:** Test the generation and accuracy of standard reports.
- **Custom Reports:** Verify the ability to create and run custom reports based on specific criteria.
- **Export Functionality:** Ensure reports can be exported in various formats (e.g., PDF, Excel).

10.10 System Administration

- **User Roles and Permissions:** Verify the creation and management of user roles and permissions.
- **System Settings:** Test the configuration and update of system settings.
- **Audit Logs:** Ensure audit logs are generated for key actions and can be reviewed by administrators.

11. RESOURCES/ROLES & RESPONSIBILITIES

The following staff members are involved in the test project for the OrangeHRM Management system, with their respective roles and responsibilities outlined:

11.1 Project Manager

- **NAME:** NA

<ul style="list-style-type: none"> • RESPONSIBILITIES:
<ul style="list-style-type: none"> ○ OVERSEE THE ENTIRE TEST PROJECT
<ul style="list-style-type: none"> ○ COORDINATE BETWEEN DIFFERENT TEAMS AND STAKEHOLDERS
<ul style="list-style-type: none"> ○ ENSURE THE TEST PLAN IS EXECUTED AS SCHEDULED
<ul style="list-style-type: none"> ○ APPROVE CHANGES AND HANDLE ESCALATIONS

11.2 Test Manager

- NAME: NA

- **RESPONSIBILITIES:**

- DEVELOP AND MAINTAIN THE TEST PLAN
- MANAGE THE TESTING TEAM
- ASSIGN TASKS AND MONITOR PROGRESS
- ENSURE ALL TEST CASES AND TEST SCRIPTS ARE PREPARED AND REVIEWED
- REPORT TEST STATUS TO THE PROJECT MANAGER

11.3 Test Lead

- NAME: NA

- **RESPONSIBILITIES:**

- LEAD THE TEST EXECUTION EFFORTS
- COORDINATE WITH TESTERS AND DEVELOPERS FOR ISSUE RESOLUTION
- MONITOR AND ENSURE THE QUALITY OF TEST DELIVERABLES
- CONDUCT TEST CASE REVIEWS AND PROVIDE FEEDBACK

11.4 Testers

- NAMES:

- NA
- NA
- NA

- **RESPONSIBILITIES:**

- DESIGN AND EXECUTE TEST CASES
- RECORD AND REPORT DEFECTS
- VERIFY DEFECT FIXES

- PERFORM REGRESSION TESTING
- PROVIDE TEST STATUS UPDATES TO THE TEST LEAD

11.5 Automation Test Engineer

- NAME: NA

• RESPONSIBILITIES:

- DEVELOP AND MAINTAIN AUTOMATED TEST SCRIPTS
- EXECUTE AUTOMATED TEST CASES
- INTEGRATE AUTOMATED TESTING INTO THE CI/CD PIPELINE
- REPORT AND ANALYZE TEST AUTOMATION RESULTS

11.6 Developers

- NAMES:

- NA
- NA

• RESPONSIBILITIES:

- FIX DEFECTS REPORTED BY TESTERS
- PROVIDE TECHNICAL SUPPORT DURING TESTING
- PARTICIPATE IN CODE REVIEWS
- ASSIST WITH UNIT TESTING AND INTEGRATION TESTING

11.7 Operations Staff

- NAMES:

- NA
- NA

• RESPONSIBILITIES:

- SET UP AND MAINTAIN THE TEST ENVIRONMENT
- ENSURE THE AVAILABILITY OF TEST DATA

- PROVIDE SUPPORT FOR DEPLOYMENT AND CONFIGURATION OF THE APPLICATION

11.8 Business Analyst

- NAME: NA

- **RESPONSIBILITIES:**

- PROVIDE REQUIREMENTS CLARIFICATION
- ASSIST IN DEVELOPING ACCEPTANCE CRITERIA
- PARTICIPATE IN USER ACCEPTANCE TESTING
- VALIDATE THAT THE APPLICATION MEETS BUSINESS NEEDS

11.9 Quality Assurance Director

- NAME: NA

- **RESPONSIBILITIES:**

- ENSURE ADHERENCE TO QUALITY STANDARDS
- REVIEW AND APPROVE TEST PLANS AND TEST CASES
- PROVIDE GUIDANCE AND SUPPORT TO THE TESTING TEAM
- OVERSEE THE IMPLEMENTATION OF QUALITY ASSURANCE PROCESSES

11.10 Stakeholders

- NAMES: NA

- VARIOUS DEPARTMENT HEADS AND KEY USERS

- **RESPONSIBILITIES:**

- PROVIDE INPUT ON BUSINESS REQUIREMENTS
-
- PARTICIPATE IN USER ACCEPTANCE TESTING
- REVIEW AND APPROVE FINAL DELIVERABLES

12. SCHEDULES

12.1 Test Schedule

The initial test schedule for the OrangeHRM management system testing follows:

Task Name	Start	Finish	Effort	Comments
Test Planning				
Review Requirements documents				
Create initial test estimates				
Staff and train new test resources				
First deploy to QA test environment				
Functional testing – Iteration 1				
Iteration 2 deploy to QA test environment				
Functional testing – Iteration 2				
System testing				
Regression testing				
UAT				
Resolution of final defects and final build testing				
Deploy to Staging environment				
Performance testing				
Release to Production				

The project is using an agile approach, with weekly iterations. At the end of each week, the requirements identified for that iteration will be delivered to the team and will be tested. Exploratory testing will play a large part in the testing as the team has never used this type of tool and will be learning as they go. Tests for planned functionality will be created and added to TCT as we get iterations of the product.

12.2 Deliverables

Deliverable	For	Date / Milestone
Test Plan	Project Manager; QA Director; Test Team	
Traceability Matrix	Project Manager; QA Director	
Test Results	Project Manager	
Test Status report	QA Manager, QA Director	
Metrics	All team members	
Test Cases	QA Team	
Test Incident Reports	QA Team	
Test Summary Reports	QA Manager, QA Director	

13. SIGNIFICANTLY IMPACTED DEPARTMENTS (SIDs)

Department/Business Area	Business Manager	Tester(s)
Human Resources	NA	NA
IT Department	NA	NA
Payroll	NA	NA
Recruitment	NA	NA
Performance Management	NA	NA
Reporting and Analytics	NA	NA
System Administration	NA	NA

Descriptions:

- **Human Resources:** Oversees employee data management, leave, and attendance.
 - **Business Manager:** NA
 - **Tester(s):** NA
- **IT Department:** Manages system maintenance, security, and technical support.
 - **Business Manager:** NA
 - **Tester(s):** NA
- **Payroll:** Handles payroll processing and financial transactions.
 - **Business Manager:** NA
 - **Tester(s):** NA
- **Recruitment:** Manages job vacancies, candidate applications, and interview processes.
 - **Business Manager:** NA
 - **Tester(s):** NA

- **Performance Management:** Manages performance reviews, goal setting, and feedback.
 - **Business Manager:** NA
 - **Tester(s):** NA
- **Reporting and Analytics:** Generates standard and custom reports, data analysis.
 - **Business Manager:** NA
 - **Tester(s):** NA
- **System Administration:** Oversees user roles, permissions, system settings, and audit logs.
 - **Business Manager:** NA
 - **Tester(s):** NA

14. DEPENDENCIES:

In the OrangeHRM Management System test plan, various dependencies need to be considered to ensure the testing process runs smoothly and efficiently. These dependencies include constraints related to test-item availability, testing-resource availability, and deadlines. Identifying and managing these dependencies is crucial for the success of the testing process.

14.1 Test-Item Availability:

1. **Software Modules:**
 - Availability of the Login Page, Dashboard, and My Info Page for testing.
 - Timely delivery of software builds from the development team.
2. **Test Data:**
 - Access to realistic and sufficient test data, including user accounts, employee records, and other relevant data sets.

14.2 Testing-Resource Availability:

1. **Human Resources:**
 - Availability of designated testers for conducting various tests (functional, integration, performance, etc.).

- Availability of business managers and subject matter experts for providing insights and validating test results.

2. Technical Resources:

- Access to required hardware and software environments, including servers, workstations, and testing tools.
- Availability of network resources and internet connectivity for cloud-based testing scenarios.

14.3 Deadlines

1. Project Milestones:

- Adherence to the project schedule and deadlines for each phase of testing.
- Timely completion of test planning, test case development, test execution, and defect resolution activities.

2. Release Schedules:

- Coordination with the development team to align testing activities with software release schedules.
- Meeting deadlines for delivering test results and feedback to the development team for timely fixes and enhancements.

14.4 Other Constraints

1. Regulatory Compliance:

- Ensuring that testing activities comply with relevant industry standards and regulations.

2. Security Considerations:

- Maintaining the confidentiality and integrity of test data and ensuring that testing activities do not compromise system security.

3. Third-Party Integrations:

- Dependency on third-party services and APIs integrated with the OrangeHRM Management System, which may affect the testing schedule and outcomes.

14.5 Mitigation Strategies

To mitigate the impact of these dependencies, the following strategies will be implemented:

1. Regular Communication:

- Establishing clear communication channels between the testing team, development team, and other stakeholders.
- Regular status meetings to discuss progress, challenges, and upcoming tasks.

2. Resource Planning:

- Proper planning and allocation of resources to ensure availability throughout the testing process.
- Backup plans for critical resources to avoid delays due to unavailability.

3. Flexible Scheduling:

- Allowing some flexibility in the testing schedule to accommodate unexpected delays or issues.
- Prioritizing critical test cases and functionalities to ensure essential features are thoroughly tested first.

15 RISKS/ASSUMPTIONS

15.1 ASSUMPTIONS:

This section lists assumptions specific to the OrangeHRM Management System project:

1. Delivery of the product is in a format that the test team can check it into the version control system (e.g., Git).
2. Required test data and user accounts will be provided in a timely manner.
3. Development and testing environments will be stable and available throughout the testing phase.
4. Stakeholders will be available for consultations and validations when needed.
5. All necessary documentation and requirements will be provided before the testing phase begins.
6. Test environments will mirror the production environment closely to ensure accurate results.

15.2 Risks

The following risks have been identified along with appropriate actions to mitigate their impact on the project. The impact (or severity) of the risk is based on how the project would be affected if the risk was triggered. The trigger is the milestone or event that would cause the risk to become an issue to be dealt with.

#	Risk	Impact	Trigger	Mitigation Plan
1	Scope Creep – as testers become more familiar with the tool, they will want more functionality	High	Delays in implementation date	Each iteration, functionality will be closely monitored. Priorities will be set and discussed by stakeholders. Since the driver is functionality and not time, it may be necessary to push the date out.
2	Changes to the functionality may negate the tests already written, and we may lose test cases already written	High – to schedule and quality	Loss of all test cases	Export data prior to any upgrade, massage as necessary, and re-import after upgrade.
3	Weekly delivery is not possible because the developer works off-site	Medium	Product did not get delivered on schedule	Implement a clear and regular communication plan with the developer. Set up contingency plans to handle delays.
4	Incomplete or delayed delivery of test data or documentation	Medium	Lack of required information	Establish a timeline for data and documentation delivery. Escalate issues promptly to project managers.
5	Test environment instability	High	Frequent downtimes or unexpected behavior	Ensure test environment maintenance is scheduled. Use backup environments if necessary.

6	Inadequate stakeholder availability for consultations	Medium	Delays in validation and decision-making	Schedule regular check-ins with stakeholders. Have a backup plan for delayed responses.
---	---	--------	--	---

16 TOOLS

For the testing of the OrangeHRM Management System, the following tools will be utilized:

16.1 Automation Tools

1. **Selenium WebDriver:** For automating web application testing and executing test scripts.
2. **TestNG:** For designing and running test cases in Java, enabling parallel test execution.
3. **Jenkins:** For continuous integration and continuous delivery (CI/CD) to automate the testing process.
4. **Postman:** For API testing, ensuring that all endpoints work as expected.

16.2 Bug Tracking Tools

1. **JIRA:** For tracking and managing bugs, features, and tasks. It allows for detailed reporting and tracking of issues from discovery to resolution.
2. **Bugzilla:** As an alternative bug tracking system to manage defect lifecycle effectively.
3. **MantisBT:** For additional bug tracking and project management needs, providing an open-source solution for tracking issues.

16.3 Other Tools

1. **Apache JMeter:** For performance and load testing of web applications.
2. **MySQL Workbench:** For database management and to run SQL queries for verifying database-related test cases.
3. **Git/GitHub:** For version control and collaboration on test scripts and related documents.
4. **Slack:** For team communication and collaboration.

5. **Confluence:** For documentation and sharing knowledge within the testing team.

17 Entry and Exit Criteria

The entry and exit criteria for each phase of the Software Testing Life Cycle (STLC) ensure that the testing process is systematic and thorough. Below are the criteria specific to each phase:

17.1 Requirement Analysis

Entry Criteria:

- The Testing Team receives the complete and finalized Requirements Documents or details about the Project.
- Project stakeholders have provided necessary clarifications on the requirements.
- Requirements are documented in a format accessible to the Testing Team.

Exit Criteria:

- The list of requirements is explored and fully understood by the Testing Team.
- All ambiguities and doubts regarding requirements are resolved.
- A Requirements Traceability Matrix (RTM) is created to map test cases to requirements.
- Testing scope and objectives are defined based on the requirements.

17.2 Test Planning

Entry Criteria:

- Requirements have been analyzed and understood.
- Resources, tools, and environment for testing have been identified.

Exit Criteria:

- A comprehensive Test Plan document is created and approved.
- Test strategy, scope, objectives, schedule, and resource allocation are defined.
- Test cases are drafted and reviewed.

17.3 Test Case Design

Entry Criteria:

- Requirements and Test Plan are available and approved.
- Test scenarios and objectives are defined.

Exit Criteria:

- Test cases are written, reviewed, and approved.
- Test cases cover all aspects of the requirements and are mapped to the RTM.
- Test data requirements are identified and prepared.

17.4 Test Environment Setup

Entry Criteria:

- Test Plan and Test Cases are finalized.
- Test environment specifications are available.

Exit Criteria:

- Test environment is set up and configured as per requirements.
- Test environment is validated to ensure it meets the criteria for testing.
- Test data is loaded and available.

17.5 Test Execution

Entry Criteria:

- Test cases are finalized and approved.
- Test environment is set up and stable.
- Test data is prepared.

Exit Criteria:

- All planned test cases are executed.
- Test results are recorded, and defects are logged.
- Test execution report is generated.

17.6 Defect Reporting and Tracking

Entry Criteria:

- Test cases are executed, and defects are identified.

Exit Criteria:

- All defects are logged, categorized, and assigned to the relevant teams.
- Defects are resolved, retested, and closed.
- Defect report is reviewed and approved.

17.7 Test Closure

Entry Criteria:

- All test cases have been executed, and defects have been addressed.
- Test results have been reviewed and approved.

Exit Criteria:

- Test summary report is created and reviewed.
- Lessons learned are documented.
- Test artifacts are archived.
- Testing phase is formally closed, and all documentation is finalized.

18. Approvals

The following individuals are responsible for approving this Test Plan. Please provide your signature and the date upon approval.

Name (In Capital Letters)	Signature	Date
Project Manager		
QA Director		
Development Manager		
Business Analyst		
Client Representative		

