

# AUTOMATION TEST PLAN

## 1. Introduction

This Automation Test Plan outlines the strategy, scope, tools, and execution approach for automating critical OrangeHRM web application functionalities using Selenium WebDriver.

---

## 2. Objectives

- Automate high-priority and frequently used test scenarios.
  - Increase coverage for login, user management, Recruitment , Leave Management and Authentication Security
  - Reduce manual testing effort and minimize human error.
- 

## 3. Scope of Automation

### In Scope

- Login
- Authentication security
- Recruitment
- User Role Management
- Leave management

## **Out of Scope**

- Performance testing or load testing
  - API automation
  - Advanced security testing
  - Mobile / Tablet versions
  - Database testing
- 

## **4. Automation Tools & Framework**

- Automation Tool: Selenium WebDriver
  - Language: Java
  - Framework: TestNG
  - Build Tool: Maven
  - Design Pattern: Page Object Model (POM)
  - Reporting: Google docs
  - Version Control: GitHub
- 

## **5. Test Environment**

- Test URL: <https://opensource-demo.orangehrmlive.com>
- Browsers: Chrome

- OS: Windows 10/11
  - Drivers: ChromeDriver
- 

## 6. Test Data

- Valid login credentials
- Invalid login credentials
- Employee creation test data
- User role management data

Stored in: IntelliJ

---

## 7. Automation Strategy

- Use TestNG for annotations and assertions
  - Maintain environment-independent scripts
  - Include positive scenarios
- 

## 8. Entry and Exit Criteria

### Entry

- Application stable for testing
- Manual test cases approved

- Test data prepared
- Environment available

#### **Exit**

- All planned scripts executed
  - Automation report generated
- 

## **9. Timeline**

Phase	Activities	Duration (Days)	Starting Date	End Date
<b>Planning &amp; Test Data</b>	Define automation scope, select test cases, prepare login/leave/role data/Recruitment.	3	14/11/2025	16/11/2025
<b>Framework Setup</b>	Set up Selenium + TestNG + Maven.	2	17/11/2025	18/11/2025
<b>Script Development – Login &amp; Auth</b>	Automate Login.	1	3/12/2025	3/12/2025

<b>Script Development – Recruitment</b>	Automate add vacancy job / Delete / Search.	1	2/12/2025	2/12/2025
<b>Script Development – Admin &amp; Dashboard</b>	Automate User Role Management, Dashboard navigation.	1	1/12/2025	1/12/2025
<b>Execution &amp; Debugging</b>	Run scripts, ensure scripts are stable.	1	3/12/2025	3/12/2025
<b>Reporting &amp; Closure</b>	Generate test execution reports and closure report.	1	3/12/2025	3/12/2025

## 9. Roles and Responsibilities

Owned Feature/Module (Automation Testing Focus)	Owned Feature/Module (API Testing Focus)	Team Member
<b>Login &amp; Leave Management</b>	<b>Leave Management (Leave Request &amp; Status APIs)</b>	<b>Jana Mohamed</b>
<b>User Role Management , Authentication Security</b>	<b>Admin Module (User Management APIs</b>	<b>Sara Haitham</b>

<b>Info Employee</b> (Add/Delete/Search UI Scenarios)	<b>PIM ,Candidate APIs</b>  <b>Spearheaded API Documentation Analysis.</b> (Responsible for obtaining API documentation and guiding the team on writing effective API test cases.)	<b>Afnan Mohamed</b>
<b>Recruitment</b> (Add/Delete/Search UI Scenarios	<b>PIM ,Candidate APIs</b>	<b>Salma Mohamed</b>

---

## 10. Risks

- Demo site instability may cause false failures
  - Changes in UI locators may break scripts
  - Browser compatibility constraints
- 

## 11. Deliverables

- Automated scripts
- Test Execution Report
- Automation Test Closure Report
- Source code on GitHub

# API TEST PLAN

## 1. Introduction

This API Test Plan defines the approach for validating REST APIs exposed by OrangeHRM using Postman.

It covers endpoints for authentication, employee management (PIM), leave management, and admin/job titles.

---

## 2. Objectives

- Validate all API endpoints related to Auth, Employee CRUD, Leave, and Admin modules.
  - Ensure correctness of responses, headers, status codes, and payloads.
  - Verify that functional, negative, and edge-case scenarios behave as expected.
  - Confirm that authentication and authorization are enforced correctly.
- 

## 3. Scope

### In Scope

- Auth APIs (Obtain OAuth2 token, access control validation)
- PIM Module APIs (List, Get, Create, Update, Delete Employees)
- Leave Module APIs (List leave types, Get entitlements, Apply leave, Approve/Reject leave)
- Admin Module APIs (List, Get, Create, Update, Delete Job Titles)
- Error code validation for invalid requests

- JSON schema validation
- Positive & negative testing

## **Out of Scope**

- Performance testing
- Security / API penetration testing
- Integration with external services

## **Timeline**

Phase	Description	Start Date	End Date	Responsible
<b>Test Planning</b>	Define approach, objectives, scope, tools, environment, test data.	16/11/2025	16/11/2025	Team
<b>Test Design</b>	Design test scenarios and test cases for all modules.	17/11/2025	18/11/2025	Team
<b>Test Preparation</b>	Prepare Postman collection, request bodies, expected responses.	19/11/2025	20/11/2025	Team

<b>Test Execution</b>	Execute tests (pending due to token availability).	21/11/2025  (execution failed due to no access to valid token)	23/11/2025	Team
<b>Defect Logging &amp; Review</b>	Log any defects found during execution, review test results.	24/11/2025  (execution failed due to no access to valid token)	24/11/2025	Team
<b>Test Closure</b>	Prepare API Test Closure Report.	02/12/2025	03/12/2025	Team

---

## 4. API Testing Tools

- Postman for manual execution
- 

## 5. Test Environment

- API Base URL (Production Demo)
- Authentication: Basic/Token

- Headers:
    - Content-Type: application/json
- 

## 6. Test Data

- Request bodies (JSON)
  - Valid and invalid payloads
  - Edge-case inputs
  - Token values
- 

## 7. Test Strategy

- **Functional testing:** GET, POST, PUT, DELETE endpoints
  - **Positive & negative testing** (e.g., missing fields, invalid IDs, unauthorized access)
  - **Response validation:** status codes, payload structure, JSON schema
  - **Authentication checks:** validate that protected APIs cannot be accessed without token
  - **Edge-case testing:** invalid dates, non-existent IDs, missing fields
- 

## 8. Entry and Exit Criteria

### Entry

- API documentation available
- Test data completed
- Postman collection prepared

## Exit

- Test cases designed and ready for execution
  - All test scenarios covered (including positive, negative, and edge cases)
  - High-severity defects would be logged if found during execution (*execution not performed due to environment limitations*)
  - API Test Closure Report prepared
- 

## 9. Risks

- Demo environment returns unstable data
  - Token expiration may interrupt tests
  - API documentation may be incomplete
- 

## 10. Deliverables

- Postman Test Cases
- Test Scenarios
- API Responses
- Postman Collections
- API Test Closure Report