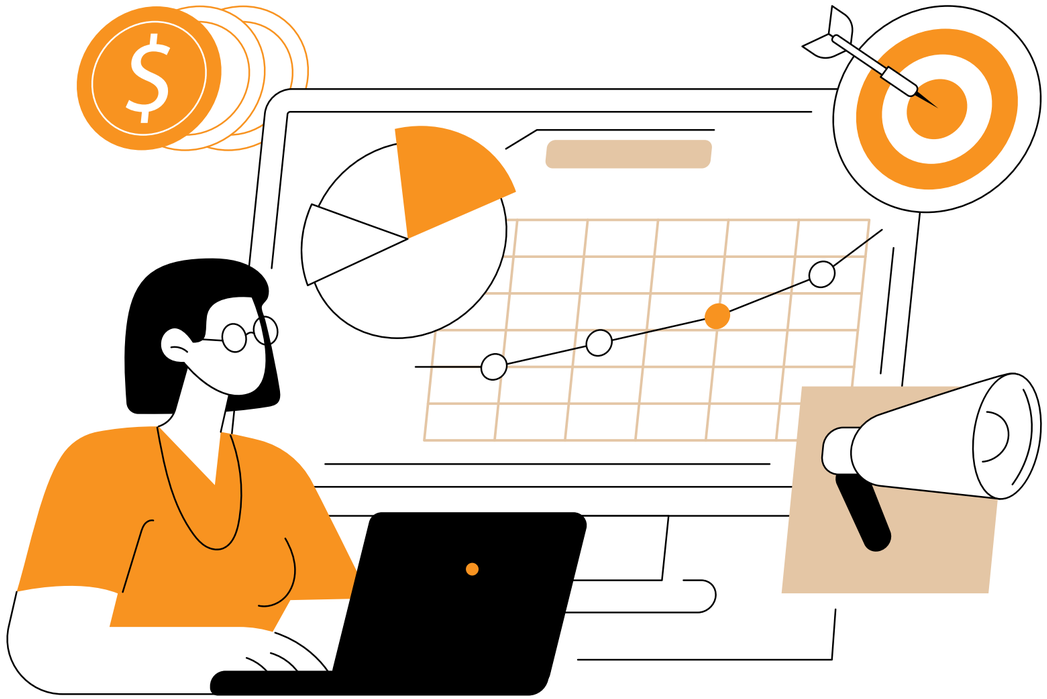
Project Workflow Overview

Testing &
QA Lifecycle

A full overview of how the QA team gathers requirements, performs static testing, writes test cases, executes tests, reports bugs, and builds automation.



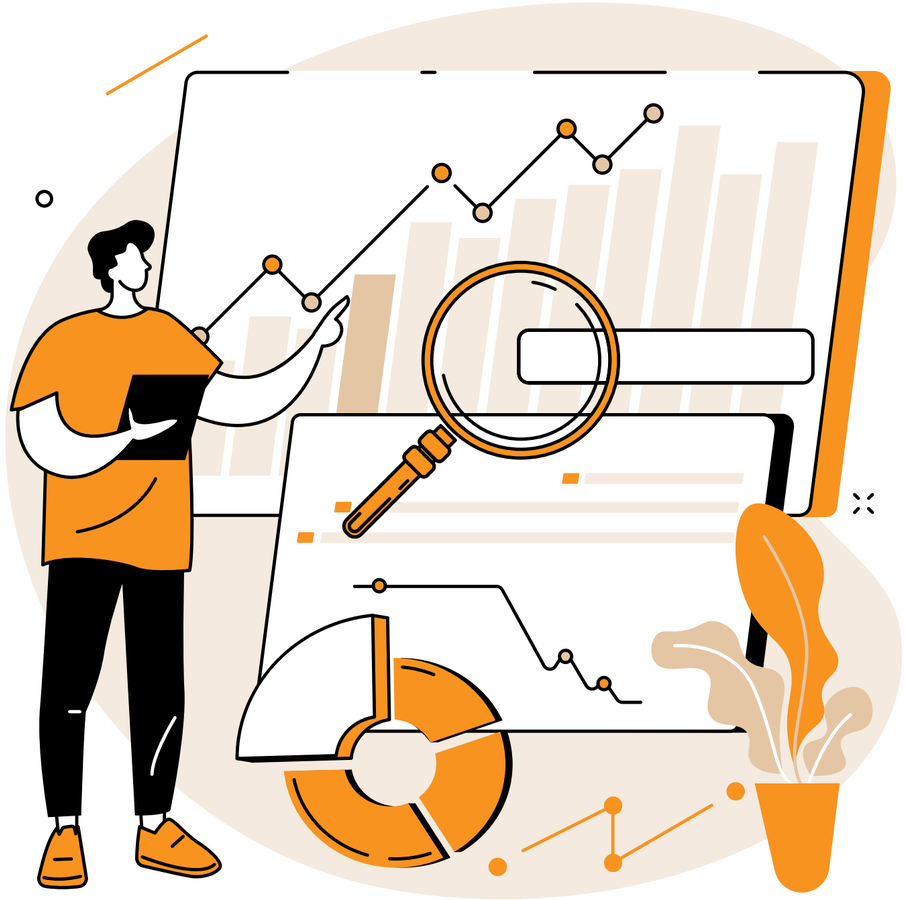
Agenda

- Introduction
- Why ?
- Requirments gathring
- Test Planning
- Testing Analysis
- Testing Design
- Test Execution
- Test Reporting

Introduction

Testing Approach

- Testing is a core phase in SDLC
- Testing started early from requirement analysis to reduce risks.
- structured Software Testing Life Cycle (STLC).
- Focused on understanding real business logic.



Coverage

Avoided

- Ad-hoc testing
- Random retesting
- Expected vs actual mismatch
- Assumption-based testing

Covered

- Core business workflows
- Happy, negative & edge case
- Role-based access control
- Data validation & integrity
- Requirements traceability
- Manual testing (UI/UX)
- Automation (regression & smoke)

Why OrangeHRM

is a good demo website for testing

- Real HR management system(looks and behaves like a real company system)
  It supports the business by organizing people and processes
  It includes modules like
  - Admin
  - PIM (Employee management)
  - Leave
  - Time
  - Recruitment

  This helps you practice on a system that is close to what you'll see in real jobs.

- Multiple user roles
  - Admin
  - ESS (Employee Self-Service)
- Covers many testing types
- Supports automation well

01-Sep-2025 → 15-Sep-2025

1.Requirements Gathering

Requirements Analysis & Static Testing

Gathering Requirements

Teams collect Epics, User Stories, and Acceptance Criteria individually.

Getting APIs documentation

Searching for Apis Documentation so we can do testing

Deliverables from Requirements Phase

01 Epics, User Stories and Acceptance criteria

02 APIs Documentations

15-Sep-2025 → 25-Sep-2025

2.Test Planning

Key Activities:

- Defined scope (UI + API)
- Prepared Test Plan v1
- Test Plan
- Set up test tools (Postman, Selenium, Jira, GitHub)
- Defined responsibilities
- Final approval of Test Plan

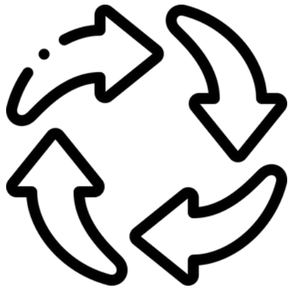
Deliverables from Planning Phase

01 Test Plan

02 Automation Test Plan

03 Ganette chart

Static Testing



How We Performed Static Testing:

- Management rotated requirements among team members
- Each member reviewed requirements they didn’t write
- Focused heavily on Acceptance Criteria (ACs) for system behavior
- Identified missing flows, unclear AC, conflicts, or ambiguous cases

Outcome:

Updated and validated requirements before test case design.

Phase 2: 26-Sep → 24-Oct

3.Testing Analysis

Tasks Completed:

- Design and analysis UI test cases for Login, Admin, PIM, Leave, Recruitment, etc.
- Prepared system test cases
- Mapped all test cases to Acceptance Criteria

Deliverables from Testing Analysis

01 Test scenarios

02 Test conditions

Phase 2: 26-Sep → 24-Oct

3.Testing Design

Tasks Completed:

- Design and analysis UI test cases for Login, Admin, PIM, Leave, Recruitment, etc.
- Prepared system test cases
- Mapped all test cases to Acceptance Criteria

Deliverables from Testing Design

01 Test Cases

02 Automation Test cases

03 API Test cases

Phase 3: 24-Oct → 30-Oct

Test Execution & Exploring the system

Execution Steps:

- Ran all manual UI test cases
- Documented bugs
- Took screenshots
- Updated Excel sheets & Jira tasks
- Exploring the system for UI/UX and Usability Testing

Focus:

Since system had few functional bugs → we emphasized UI/UX and usability issues.

Bug Reporting

Execution Steps:

- Included steps to reproduce, severity, screenshots
- Linked each defect to its test case
- Prioritized bugs (Critical/High/Medium/Low)
- Four types of bugs (functional, UI/UX, Usability and exploratory)

Deliverables from Test Execution Phase

01 UI/UX Usability and functional bug report

Phase 5: 1-Dec → 5-Dec

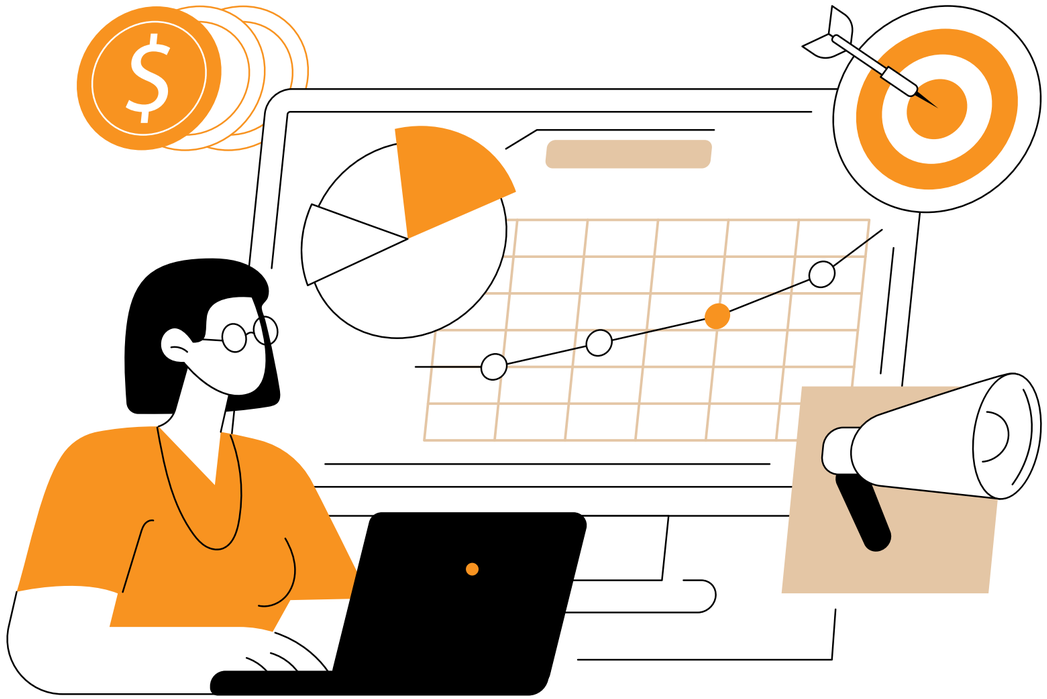
Test Reporting

Deliverables:

- Manual Clouser Report
- Final presentation

Project Workflow Overview

API Testing



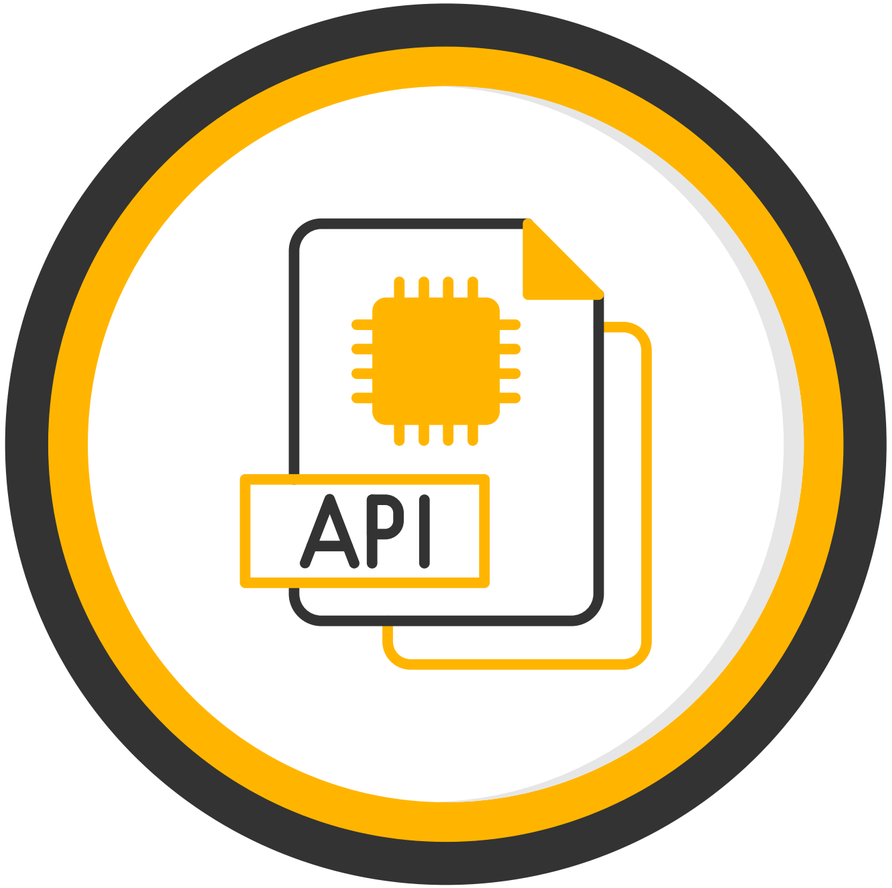
API Manual Testing

Tools Used:

Excel, Postman

Performed:

- Manual API testing (GET/POST/PUT/DELETE)
- Validating response payload, structure, status codes
- Positive & negative API scenarios
- Authentication testing
- Error handling testing



API Automation

Implemented during Test Execution & Automation Phase

We Built:

- Postman collections
- Automated workflows
- Newman CLI integration

Outcome:

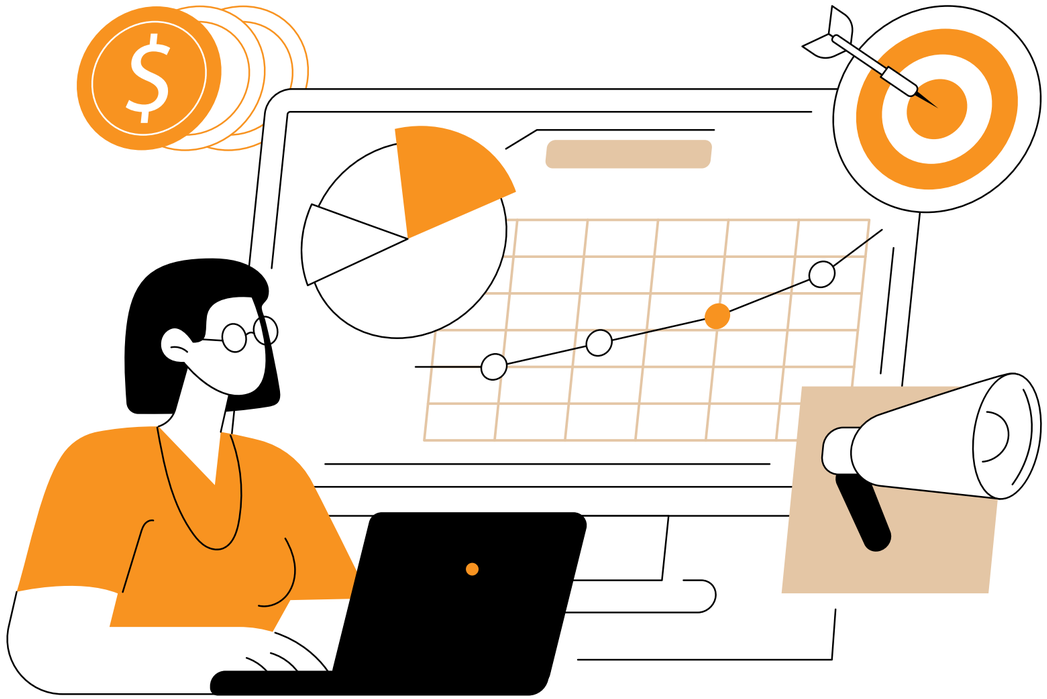
Fast API regression cycles anytime needed.

Deliverables from API Testing Phase

01 API Test Cases

Project Workflow Overview

Automation Testing

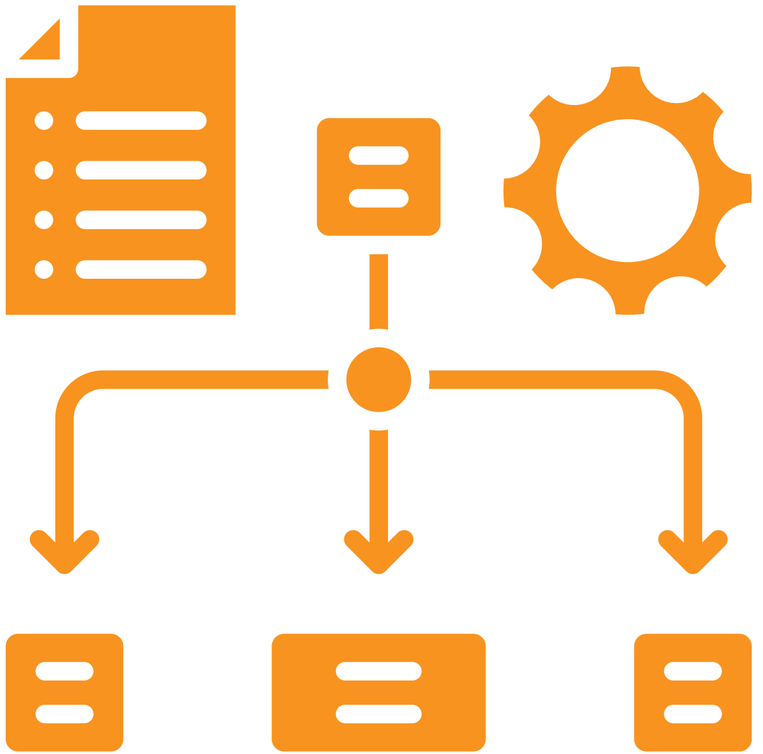


Phase 4: Test Automation (1-Nov → 30-Nov)

Automation Planning

Planning Steps:

- Selected test cases with highest ROI for automation
- Defined framework structure (Selenium + TestNG)
- Prepared GitHub repository
- Created automation roadmap



Test Automation Strategy

Tools Used:

Selenium, Java, TestNG, Postman

We created:

- UI automated scripts for critical scenarios
- Page Object Model (POM) structure
- Reusable libraries
- API automated scripts in Postman

Deliverables from Automation Phase

01 Automation framework

02 Automated test scripts

03 Smoke suite

04 Regression suite

05 Execution reports

06 Defect reports

Thank You!

For more information about the QA Workflow, feel free to reach out.