

Framework Overview

This automation framework is designed to automate the OrangeHRM application using a combination of API and UI testing approaches. It leverages Selenium WebDriver for UI automation and Rest Assured for API testing, organized under a Maven project structure and developed following the Page Object Model (POM) design pattern.

Technologies Used

- Selenium WebDriver for browser automation
- Rest Assured for API testing
- TestNG for test execution and annotations
- Extent Reports for reporting
- Maven for project management
- Java as the programming language

Framework Structure

- `base` package: Contains base classes for WebDriver setup, RestAssured configurations, and shared utilities.
- `page` package: Contains Page Object Model classes that represent different pages of the OrangeHRM application.
- `utils` package: Includes utility classes like API utilities, payload generation, and helper functions.
- `tests` package: Contains TestNG test classes organized by functionality (API tests, Contact Details UI tests, Job Details UI tests, etc.).

Flow of the Tests

1. **Login:**
Selenium is used to perform login and capture the session cookie needed for authenticated API calls.
2. **API Automation:**
 - Add a new employee using the `/pim/employees` API.
 - Update personal details of the employee using the `/pim/employees/{empNumber}/personal-details` API.
3. **UI Automation:**
 - Search and open the newly created employee in the UI.
 - Fill contact details form using Selenium.
 - Upload an attachment to the employee's profile.
 - Fill job details form (joining date, job title, employment status, etc.).
 - Save and verify success messages after each update.

Key Features

- Hybrid framework combining API and UI testing for better speed and coverage.
- Dynamic payload generation with random employee IDs.
- Robust element interaction using dynamic waits.
- Handling of possible network slowness or UI issues with try-catch and retry mechanisms.
- Absolute file paths generated dynamically for upload operations.
- Clear and modular project structure for easy maintenance and scalability.
- Extent Reports integration for detailed reporting.

Challenges Handled

- Occasionally, the OrangeHRM demo server experiences slow responses, leading to API connection timeouts. This was mitigated by configuring higher timeouts in Rest Assured.
- Sometimes the job title dropdown options fail to load properly. A fallback mechanism with try-catch is implemented to allow the test to continue without failure.

Final Notes

- This framework is designed to be flexible, maintainable, and close to real-world standards, ensuring that both API and UI layers are covered, with a clean separation of responsibilities and reusable components.