# Brief Documentation – RedBlox QA Practical Assignment

This document provides a concise overview of the work completed for the RedBlox QA Practical Assignment, covering both API and UI automation using Cypress.

## 1. Overview

The assignment consists of two components:

### Part 1 — API Automation (DummyJSON API)

Automated key authentication and user workflows using Cypress cy.request(): - Generate OTP / Login (positive & negative cases) - Verify OTP with invalid and missing fields - Register user (response validation + negative test) - Login flow (success & invalid credentials) - Get User Info with permission simulation

DummyJSON is a mock API, and newly created users may not always persist. The test suite handles both status 200 and 404.

### Part 2 — UI Automation (OrangeHRM Demo Site)

UI workflows automated using Cypress against the public OrangeHRM demo: - Login with Admin credentials - Assert successful login and dashboard visibility - Navigate to **Admin → User Management → Users** - Validate table rendering, headers, and rows - Role-based access menu visibility (Admin vs non-admin scenario)

Selectors were designed to be stable and adaptive, accounting for OrangeHRM's dynamic DOM.

## 2. How to Install & Run

### Install dependencies

npm install

### Run tests (interactive mode)

npx cypress open

### Run tests (headless Chrome)

npx cypress run --browser chrome

## 3. Project Structure

```
cypress/
  e2e/                → Automated test scripts
    api_tests.cy.js
    ui_login.cy.js
    user_management.cy.js
    role_access.cy.js
  fixtures/           → Test data
  support/            → Commands & hooks
videos/               → Auto-generated on test runs
README.md             → Setup instructions
```

## 4. Additional Notes

- DummyJSON API behavior is inconsistent with data persistence; tests are built to handle this.
- OrangeHRM UI changes occasionally; fallback selectors ensure stability.
- Negative test coverage included for both API and UI flows.
- Screenshots and videos are generated automatically by Cypress but kept local as per best practices.