# API Test Plan Template

## 1. Introduction

- Purpose of testing the API: To validate and verify the User Management APIs, ensuring they function correctly, handle data accurately, maintain security, and operate reliably.

- Scope: The following APIs will be tested:

1. CreateUser API – create a new user
2. UpdateUser API – update user information
3. DeleteUser API – delete a user
4. GetUser API – retrieve user information
5. ETE Flow – end-to-end flow:
   1. Create → update → delete user
   2. Get exist user → Update

- Out of scope: all APIs that are not listed in scope

- References: https://reqres.in/

## 2. Test Objectives

- Verify API functionality against requirements.

- Validate request/response correctness (status codes, headers, body).

- Ensure security, performance, and reliability of APIs.

- Confirm backward compatibility and integration with other services.

## 3. Test Scope

- In-Scope: Functional testing of endpoints, Data validation (input/output), Authentication & authorization, Error handling and negative cases, Performance (basic load/stress tests)

- Out-of-Scope: UI testing, Non-documented third-party APIs, Hardware or device-specific tests

## 4. Test Approach

- Tools: Playwright

- Test Levels: API functional testing, Integration testing

- Types of Testing: Positive & negative scenarios, Boundary testing, Security testing (auth, tokens, encryption), Data-driven testing (CSV/JSON test data)

## 5. Test Environment

- Base URL(s): DEV, STAGING

- Authentication method (API key)

- Test Data setup: from https://reqres.in/

## 6. Test Cases & Coverage

- Endpoint list (GET /users, POST /orders, etc.)

- Parameters and payloads

- Expected responses (200, 400, 401, 500, etc.)

- Sample Test Case Table:

## 7. Entry & Exit Criteria

- Entry: API specs ready and stable, Test environment set up, Access tokens/credentials available

- Exit: All planned test cases executed, Critical defects resolved, Regression testing passed

## 8. Risks & Mitigation

- API documentation outdated → Mitigate by sync with dev team

- Environment instability → Use mock/stub services

- Data dependencies → Prepare test data in advance

## 9. Deliverables

- Test cases ( automated scripts)

- Test execution report

- Bug/defect reports in tracking tool

- Final test summary

## 10. Schedule & Resources

- TBU

## 11. Approval

- TBU

### Sample Test Case Table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Endpoint | Method | Input | Expected Status | Expected Response | Remarks |
| TC001 | /api/users | POST | {email:"eve.holt@reqres.in", password:"cityslicka"} | 200 | {id, token} | Create User Successfully |