# Test Plan

Prepared By:

Satish Chilkaka

Date: 10-02-2023

Test plan number: Test\_Plan \_10-02-23

Table of content

1. Introduction
2. Product Analysis
3. Test Strategy
4. Test Criteria
5. Resource Allocation
6. Test Environment
7. Test schedule and Estimation
8. Test Deliverables

## **Introduction**

* 1. **Purpose**

This test plan describes the API testing approach for REST-API ready to use application.

* 1. **Project Overview**

REST-API ready to use application is an open source REST-API application which handles HTTP requests. Mainly this project can be used for demo purposes to educate, learn, test and work. When a user hits endpoints it is stored and created in a real database, the date won't be lost. However, due to the cost and server, database bandwidth in a day can send/ work with only 100 requests. In addition, rest-api applications are flexible to create your own custom objects which can store different attributes of different types as part of a "data" field that represents a customizable JSON object.

## Product analysis

The Real REST API which handles different types http requests, It's an open source product, can be accessed anytime, and it's free of cost. The primary purpose of the product is to educate the students, QA, developers, and learn from this application. This application supports GET, POST, PUT, DELETE and PATCH. This application uses a real database which means every testing purpose request will store the database. At the moment not able to look at the application backend base, However, every request will be stored.

This product does not have strict security validation, it can be accessed without an authorization token. To access this application use postman, chrome browser.

## Test Strategy

* 1. **Test Objectives**

The objective of the test is to verify that the multiple endpoints in REST-API Ready to use application, the test will execute and verify the test scripts, identify, fix and retest all high and medium severity issues, and low priority issues can be added into future backlog. After final testing the product outcome would be A production-ready software and a set of stable test scripts, which can be run in future deployments.

* 1. **Test assumptions**
* Test data can be valid or invalid
* Performance testing is not considered for this estimation
* Identified defects should added to backlog with evidence (screenshot, video)
* Test case design activities will be performed by QA
* The defects will be tracked through Linear board -> backlog
* Project Manager/ owner will review and sign-off all QA test deliverables
* Santy testing
* Functional Testing
* Regression Testing
* Automation testing
* During the functional testing QA will verify all endpoints with happy and negative path and validate the response
* In functional testing QA will perform smoke test and provide feedback in iteration based approach
* Product can be tested by multiple tools such as, JavaScript, nodeJs, cypress, Java, python

## Test Criteria

REST-API ready to use application test criteria refers to standards of all activities in a testing project. The two main test criteria are:

* **Suspension Criteria**: When QA finds 50% or more test cases have failed then all test activities should be suspended in REST-API ready to use application. Provide accurate steps to reproduce the issue and make documentation
* **Exit Criteria**: Defines the benchmarks that signify the successful completion of a test phase or project. The exit criteria are the expected results of tests and must be met before moving on to the next stage of development. For example, 80% of all test cases must be marked successful before a feature or portion of the software can be considered suitable for public use, And there shouldn’t be priority 1 issues.

| **Exit Criteria** | Test  Team | Technical  Team | Notes |
| --- | --- | --- | --- |
| 100% Automation Test Scripts executed |  |  |  |
| 95% pass rate of Test Scripts |  |  |  |
| Every endpoint works as expected |  |  |  |
| There is no remaining P1 defects |  |  |  |
| GET, POST, PUT, DELETE, PATCH endpoints should work as expected |  |  |  |

Issues Severity:

1. **Critical**: This type of defect is critical. It includes endpoint crash, not able to get any data when using the GET request, Not able to manipulate (POST, PUT, DELETE) operations

## 5. Resource Allocation

The recommended resources for testing REST-API ready to use application

| **Name** | **Role** | **Responsibilities** |
| --- | --- | --- |
| Satish Chilkaka | QA | * Test all (GET, POST, PUT, DELETE, PATCH) available rest-api endpoints * Provide Test status * Write and execute manual test cases * Write and execute automation test case |

## 6. Test Environment

**REST-API** hosted by <https://restful-api.dev/>

A chrome browser

POSTMAN

NodeJs - Axios

NodeJs

Supertest

With mocha and chai assertions

## 7. Test Schedule and Estimation

Test will be conducted in Agile environment, there are multiple ways to estimate, some of them are:

Planning Poker

Big/Uncertain/Small

T-Shirt Sizes

## 8. Establish Test Deliverables

List task required before testing:

Test Plan

Test Design

Test Data

Test scripts

After test execution list of deliverables:

Test execution status

Test report