Deloitte.

Automated API Testing Framework

Automated API Testing Framework

Contents

- Document purpose
- Unit Test Framework Overview
- Unit Test Framework Steps
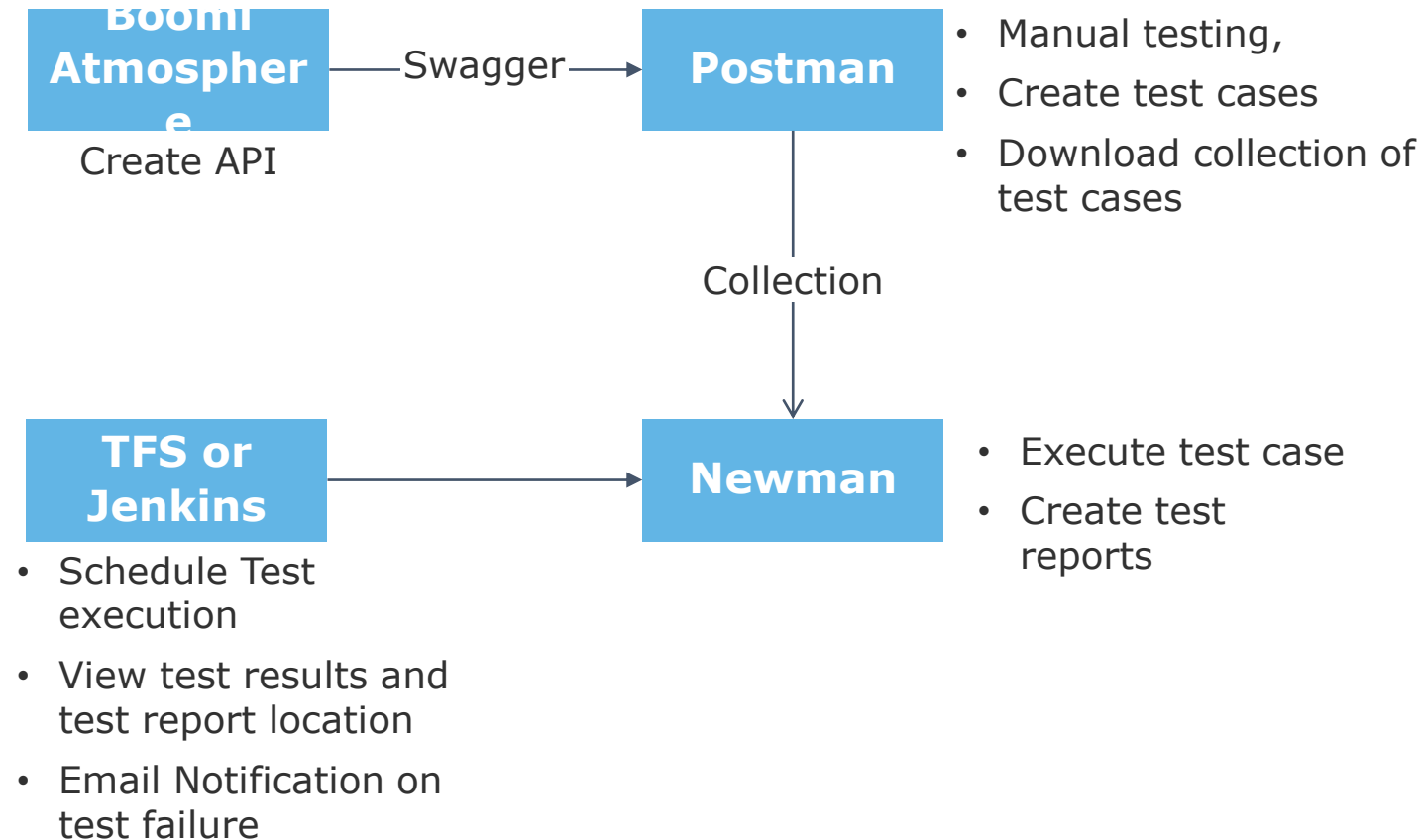- Sample Unit Test Report
- Unit Test Results Template

Purpose of the document

The objectives of this document are:

1. Provide details of the API unit test framework including steps to perform regular health checks.

2. Provide details on the tools to perform test automation in different environment

3. Provide information on the scheduling options of the automated tests

4. Provide details on how the unit test results are captured

Automated API testing will provide early warnings on breakages by minimising test execution efforts. It will also produce detailed test reports and notify user/groups about test failures by email.

Automated Unit Test Framework Overview

Boomi Atmosphere

Create API

— Swagger →

Postman

- Manual testing,
- Create test cases
- Download collection of test cases

Collection ↓

TFS or Jenkins

- Schedule Test execution
- View test results and test report location
- Email Notification on test failure

→

Newman

- Execute test case
- Create test reports

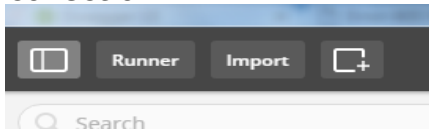# Step 1 – Develop API and unit test

1. Create API using Boomi Atomsphere

2. Manually test API using Postman

Note: The collection scripts are JSON files exported from Postman.
Subsequent slides explain in detail how to export collection.
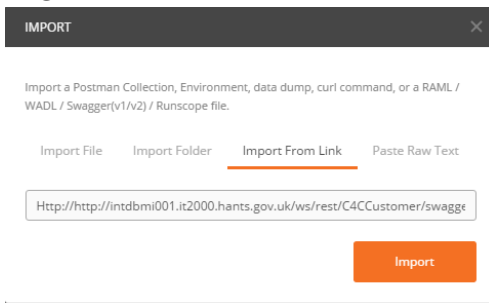
Step 2 - Create Postman Collection

Create collection script for test execution. The collection scripts are JSON files exported from Postman. Subsequent slides explain in detail how to export collection.
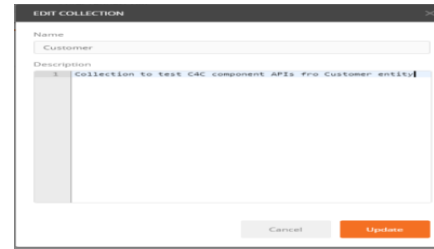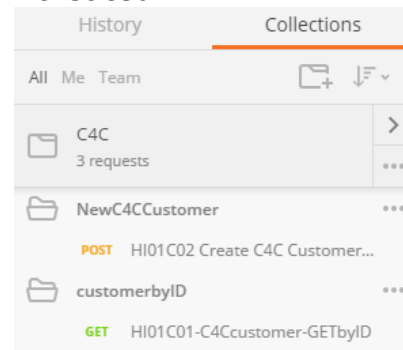
a. Click to import swagger to create collection

c. Add suitable description for collection

b. Use Import from Link, and paste the swagger.json location of swagger file

d. Verify that collection with folder and requests for all endpoints in Swagger will be created

Step 3 - Create Environments in Postman (1/2)

Create Dev and Test environments and in each environment, add key URL and provide value for it

Step 3 - Create Environments in Postman (2/2)

Update requests to use environment keys

HI01C02 Create C4C Custo | + | Dev

HI01C02 Create C4C Customer v0.1 ▼

| POST ▼ | https://intdbmi001.it2000.hants.gov.uk:80/ws/rest/C4CCustomer/NewC4CCustomer | Params | **Send** ▼ | Save ▼ |

Authorization Headers Body Pre-request Script Tests   Generate Code

Type   No Auth   ▼
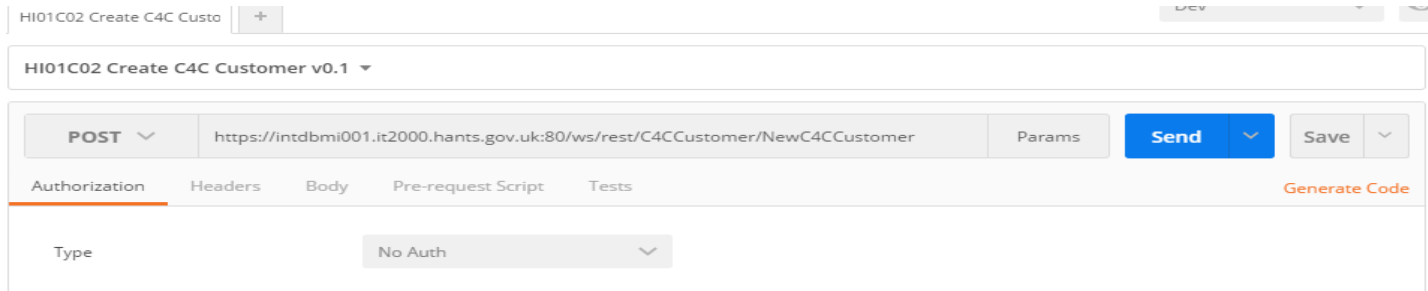
Change the requests by replacing https://intdbmi001.it2000.hants.gov.uk:80 with {{url}} as shown below

HI01C02 Create C4C Customer v0.1 ▼

| POST ▼ | {{url}}/ws/rest/C4CCustomer/NewC4CCustomer | Params | **Send** ▼ |

Authorization Headers Body Pre-request Script Tests   G
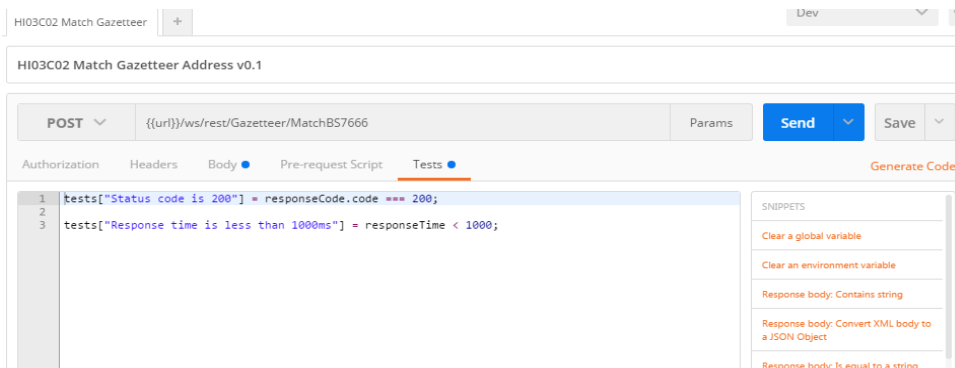
Type   No Auth   ▼
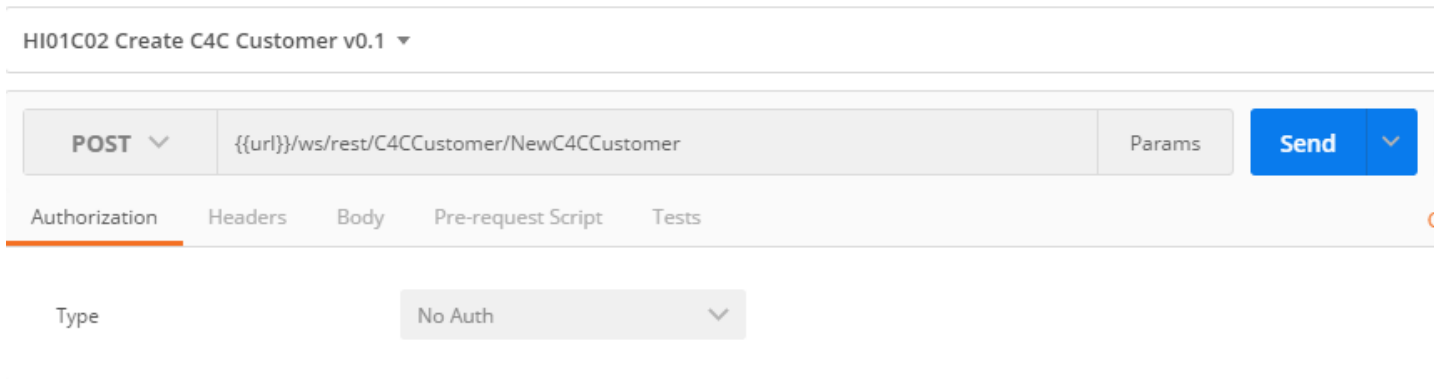
Step 4 - Create Unit Test Cases

Add request data in body, header, params as applicable in request.
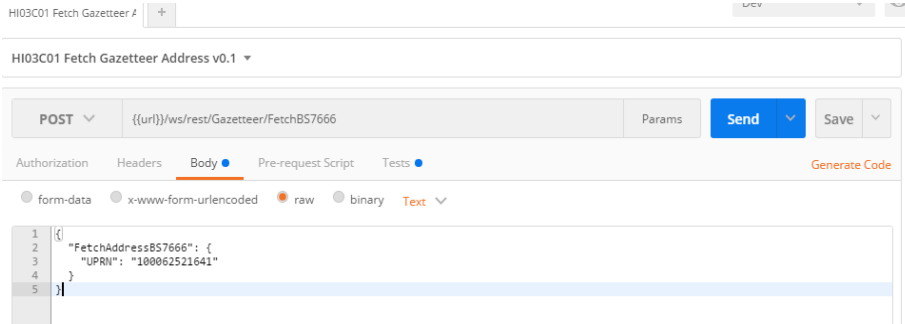
In test tab add tests by clicking suitable orange texts on right side and editing it later, tests are all JavaScript , which you can edit and explore as per tests.

Step 5- Execute Unit Test Case

Click 'Send'

Body	Cookies	Headers (8)	Tests (1/2)

All   Passed   Failed

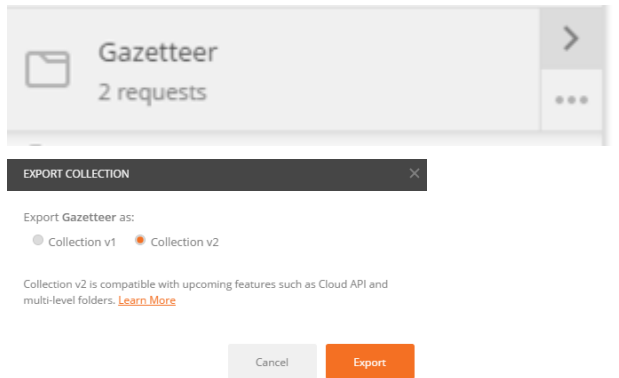PASS   Status code is 200
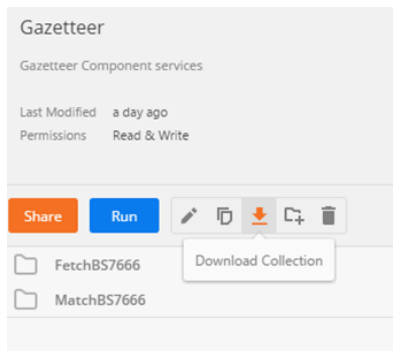
FAIL   Response time is less than 1000ms

Step 6 -Download Collection

After creating all test cases , download the collection.

Click > next to name of collection

Click download arrow to download collection

Newman Command Examples

Sample commands used in Jenkins build step

```
"C:\Users\xcncopq\AppData\Roaming\npm\newman" -c
"C:\collections\Gazetteer.postman_collection.json" -e
"C:\collections\Dev.postman_environment.json" -H
"C:\collections\Gazetteer%RANDOM%.html" --exitCode 1
```

Sample Automated Unit Test Report

Newman test report

Collection: Gazetteer
Iterations: 1
Description: Gazetteer Component services

Requests:

HI03C01 Fetch Gazetteer Address v0.1

Description: undefined
Mean time per request: 1156 ms
Total tests that passed: 1
Total tests that failed: 0

Individual tests:

Test name	Pass count	Fail count
Status code is 200	1	0

HI03C02 Match Gazetteer Address v0.1

Description: undefined
Mean time per request: 1467 ms
Total tests that passed: 1
Total tests that failed: 1

Individual tests:

Test name	Pass count	Fail count
Status code is 200	1	0
Response time is less than 1000ms	0	1