



FINAL REPORT

PROJECT API TESTING
REQRES.IN

Prepared by:
Irma Suryani

LAPORAN AKHIR PROJECT API TESTING

1. Gambaran Umum Project:

Project ini bertujuan untuk melakukan API Testing pada dummy API Reqres.in dengan memvalidasi respon API melalui pengujian manual, automation, dan performance testing. Proses testing dimulai dari pembuatan Test Plan, perancangan Test Case, eksekusi manual via Postman dengan Report Newman + HTML Extra, automation menggunakan Node.js, node fetch, framework mocha dengan Report Mochawesome, serta Load & Stress Testing menggunakan JMeter.

2. Temuan Penting:

a. Perbedaan Perilaku pada Metode PATCH:

Saat pengujian manual, ditemukan bahwa request PATCH di Reqres.in tidak berperilaku seperti API pada umumnya. Biasanya, API akan mengembalikan data yang telah diperbarui sambil mempertahankan field lain yang tidak diubah. Namun, API Reqres hanya mengembalikan data yang diubah saja dan menghapus field yang tidak diperbarui. Hal ini bukan termasuk bug, melainkan gap (perbedaan behavior antara API dummy dengan ekspektasi API pada umumnya). Sebagai perbandingan, dummy API jsonplaceholder.typicode.com tetap mempertahankan semua data dan hanya memperbarui field yang di-modify.

b. Hasil Automation Testing:

Pengujian automation menggunakan Newman (Node.js + Mocha + node-fetch + Mochawesome Report) menghasilkan output yang konsisten dengan pengujian manual. Pada PATCH request, API hanya mengembalikan field yang dimodifikasi saja, sedangkan field lain yang tidak ikut di-update justru hilang dari response. Hal ini menunjukkan bahwa behavior automation dan manual test tetap sama sesuai karakteristik dummy API Reqres.in.

c. Hasil Performance Testing:

Load & Stress Testing dilakukan menggunakan Apache JMeter. Ditemukan bahwa batas optimal jumlah user concurrent berada di angka 30-32 user. Melebihi jumlah tersebut mulai muncul tanda-tanda stress pada server (response time meningkat & terjadi request failure). Walaupun jumlah user ini tergolong kecil, server Reqres.in memang memiliki batasan user, sehingga masuk ke kondisi stress lebih cepat. Untuk menjaga kestabilan server, setiap skenario test diberi jeda 5 menit.

3. Kendala Timeline Project:

Project ini mengalami keterlambatan 10 hari dari estimasi awal. Faktor penyebabnya antara lain tidak memperhitungkan hari weekend yakni 4 hari sabtu-minggu (tidak ada progress saat weekend), kendala kuota internet selama 3 hari, dan mengalami stuck pada beberapa skenario selama 3 hari.

4. Saran & Insight yang Didapat Selama Project:

a. Enhanced Reporting (Sudah Diterapkan):

Dari masukan yang diterima selama project ini, akhirnya report API Collection Postman diterapkan dengan menggunakan Newman + HTML Extra agar hasil laporan lebih rapi dan mudah dibagikan.

- [GitHub Repository](#).
- [Live HTML Extra Report](#)

b. Dynamic Data Chaining (Untuk Project Berikutnya):

Terdapat saran untuk mencoba penerapan dynamic data chaining antar request API (contohnya, menggunakan order ID yang di-generate dari API pertama untuk request berikutnya). Saran ini belum diterapkan di project sekarang, tapi akan dieksplor lebih lanjut di project berikutnya.

c. Test Case Management Tools (Next Exploration):

Saran lainnya adalah menggunakan tools Test Case Management seperti [Qase.io](#) untuk dokumentasi dan tracking test case agar lebih terstruktur. Hal ini juga akan dieksplor di project selanjutnya.

3. Kesimpulan:

Project API Testing ini berhasil meliputi fase manual, automation, dan performance testing, dengan hasil yang sesuai dengan ekspektasi perilaku dummy API Reqres.in. Hasil automation pun konsisten dengan manual testing. Batasan performa API berhasil diidentifikasi menggunakan JMeter. Walaupun mengalami beberapa kendala timeline, project ini memberikan insight penting dalam peningkatan kualitas reporting dan pengelolaan data pengujian untuk project berikutnya.

4. Referensi & Dokumentasi Link:

Berikut adalah beberapa link yang mendukung project API Testing ini, mulai dari dokumentasi postingan linkedin demo task tracking, test plan sampai hasil report yang sudah di-generate:

[!\[\]\(2bdfe261b986065ee0ac76460d6528c9_img.jpg\) Trello Board \(Task Tracking & Timeline\)](#)

[!\[\]\(dfbd6b3763a6d1d9afaa974f64e2e4b5_img.jpg\) Test Plan Document](#)

[!\[\]\(e78f798d4ea5c530c9db49e7d26e6b95_img.jpg\) Test Case Documentation](#)

[!\[\]\(23d9fc146e83b5c3013cfa32c784f8d5_img.jpg\) Postman Test](#)

[!\[\]\(c694a3ff3b077d76910920a6a1593ab4_img.jpg\) Automation Test](#)

[!\[\]\(ec9132f1d27c8919987d92907322654d_img.jpg\) Performance Testing Test](#)

[!\[\]\(05be7c7a8995decd503647c99211f7c2_img.jpg\) Report Newman + HTML Extra Post](#)

[!\[\]\(aa53ad6fea213b8b2226d3077e30533a_img.jpg\) GitHub Repository \(Project Files\):](#)