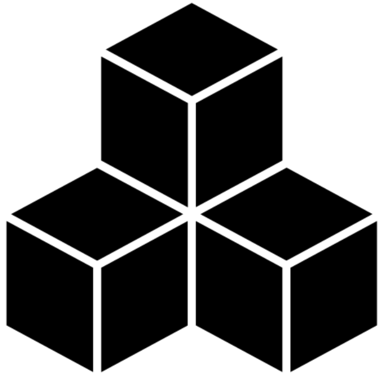


Hands on Workshop

On PostMan

Rasel Hossain

Software Developer at Stack Learner

- **Documentation Deficiency:** Their APIs lack proper documentation. For instance, details on how their partners should call API endpoints from different clients, the necessary payload, headers required for sending requests, etc., are missing.
- **Performance Analysis Dilemma:** They lack an easy solution to analyze and identify performance issues, errors, or any unexpected behavior in their APIs, hampering improvements.
- **Collaboration Challenges:** There is a problem with collaboration and easily sharing APIs across the development team.

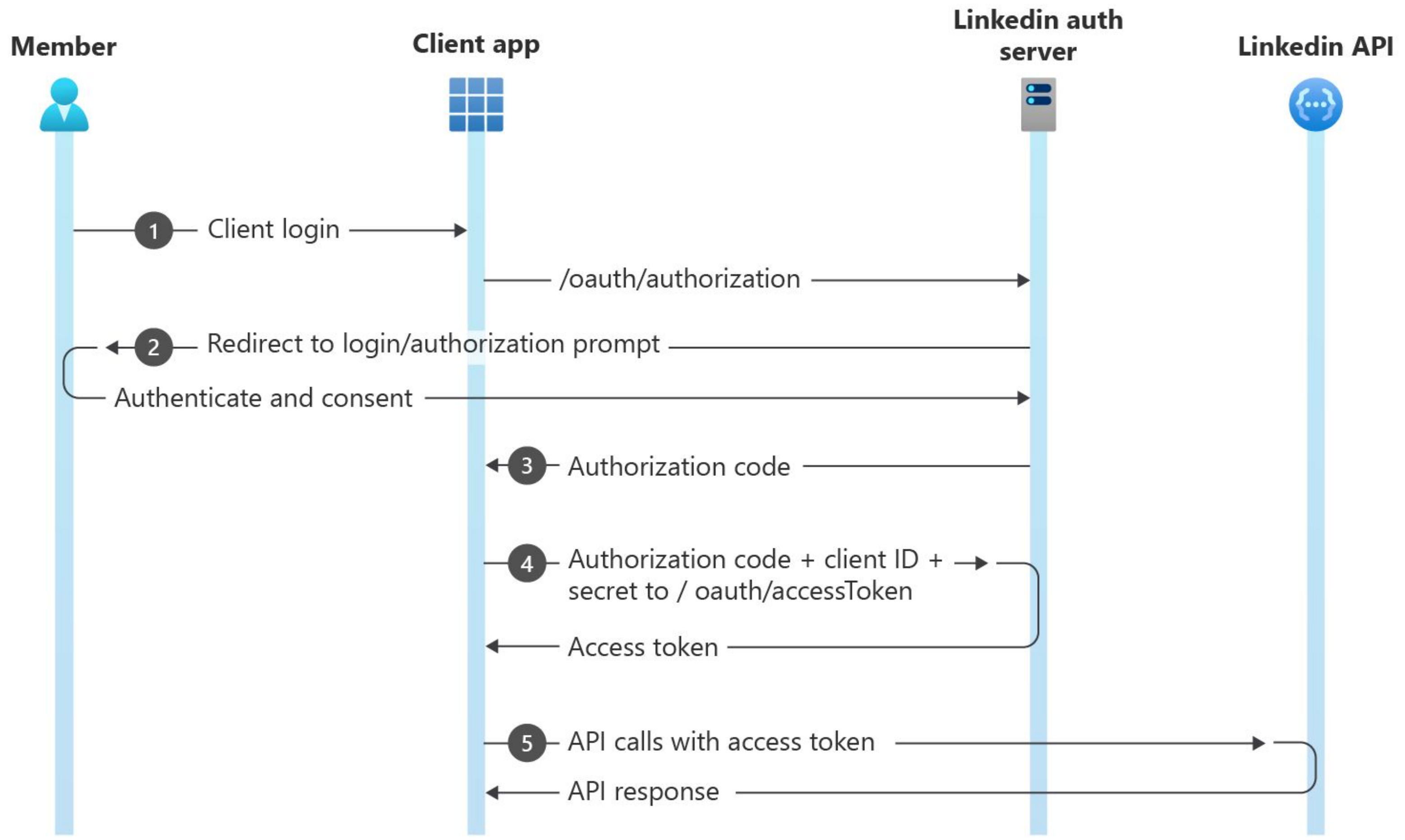
Focus on The Following Problems

- **Frontend-Backend Coordination Issues:** They regularly observe that their frontend teams sometimes wait for the backend team to complete APIs, causing delays in completing projects on time.
- **Automated Testing Goals:** They also aim to automate their API testing process, generating reports automatically.
- **API Data Flow:** Their backend team desires a system where they can directly manipulate APIs and observe the data flowing between them without creating a new application. This helps them create a mind map of their API functionality.

Authorization Code Flow

The Authorization Code Flow is an OAuth 2.0 authentication flow used by applications to obtain authorization to access resources on behalf of a user. It is commonly used in web applications where the client-side code runs in a web browser.

The Authorization Code Flow provides a secure and efficient way for applications to obtain access to protected resources on behalf of users without exposing sensitive credentials. It ensures that only authorized applications can access user data and that users have control over which resources the application can access.

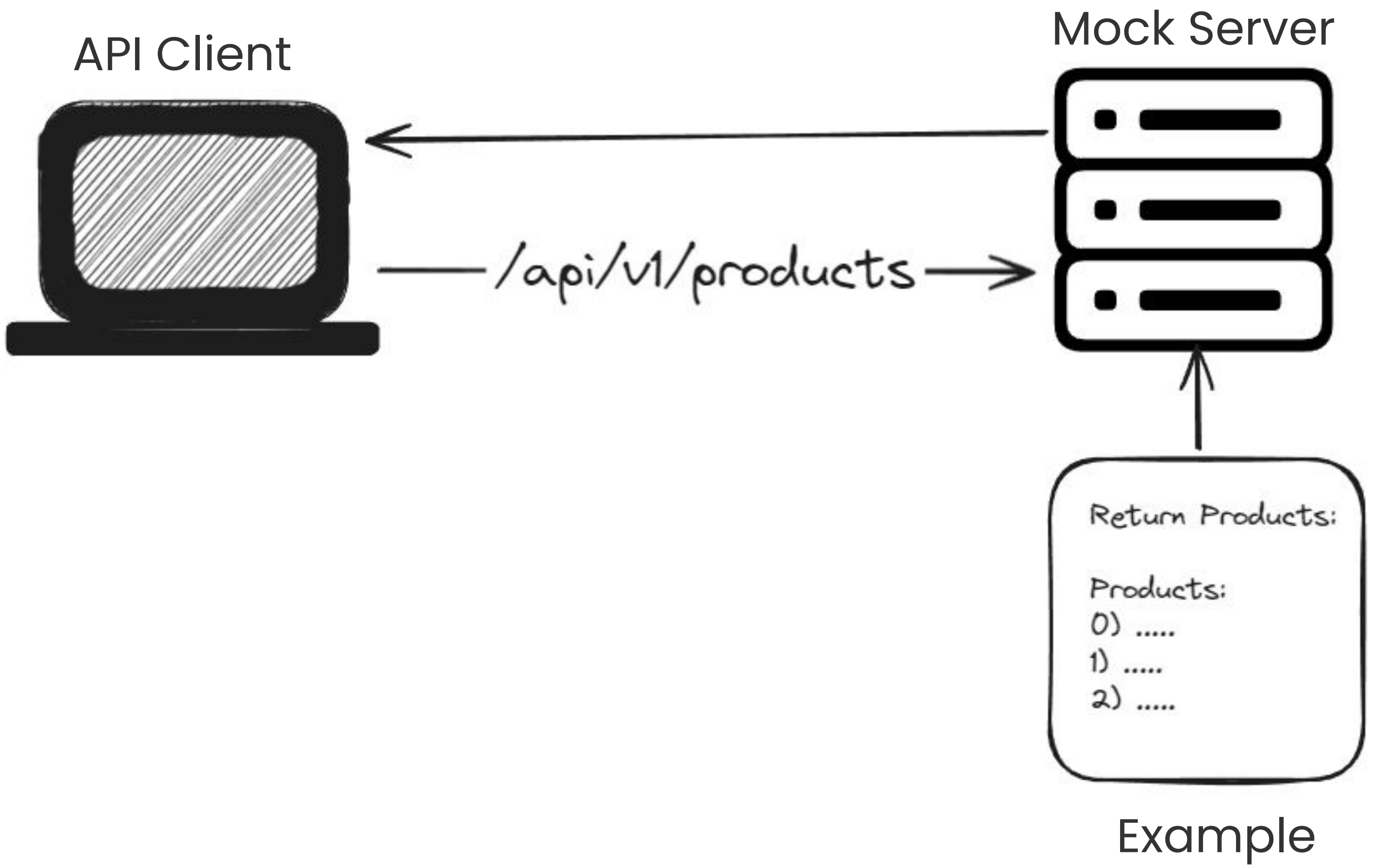

Member
Client app
Linkedin auth server
Linkedin API
1
Client login
/oauth/authorization
2
Redirect to login/authorization prompt
Authenticate and consent
3
Authorization code
4
Authorization code + client ID + secret to / oauth/accessToken
Access token
5
API calls with access token
API response

Frontend-Backend Coordination Issues:

Postman Mock Server

Postman enables you to create mock servers to assist with API development and testing. A mock server simulates the behavior of a real API server by accepting requests and returning responses. By adding a mock server to your collection and adding examples to your requests, you can simulate the behavior of a real API.

When you send a request to a mock server, Postman matches the request to a saved example in your collection. Postman then responds with the data you added to the example.


API Client
Mock Server
/api/v1/products
Return Products:
Products:
0) .....
1) .....
2) .....
Example

Let’s create a Mock Server

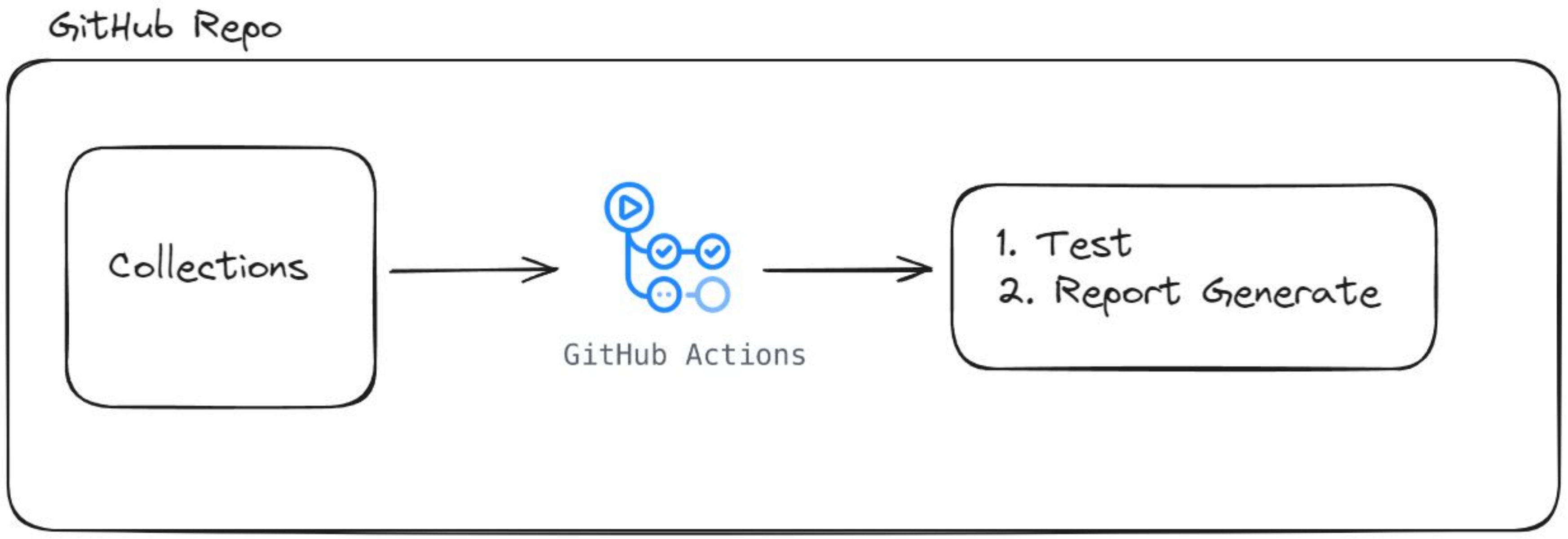
Automated Testing Goals:

Postman CLI

Postman CLI, created and maintained by Postman, provides a command-line interface for executing Postman collections. It supports collection runs and automatically sends run results to Postman by default. It's distributed as a downloadable package, signed by Postman, and can be downloaded programmatically. Postman CLI also supports sign in and sign out.

Newman

Newman, also created by Postman but maintained as an open-source project with community contributions, is a command-line collection runner for Postman. It supports executing Postman collections. Newman is distributed on npm and can be downloaded programmatically. Unlike Postman CLI, Newman doesn't support sign in and sign out. It is very handy to integrate with your CI pipeline


GitHub Repo
Collections
GitHub Actions
1. Test
2. Report Generate

LET'S DO THIS

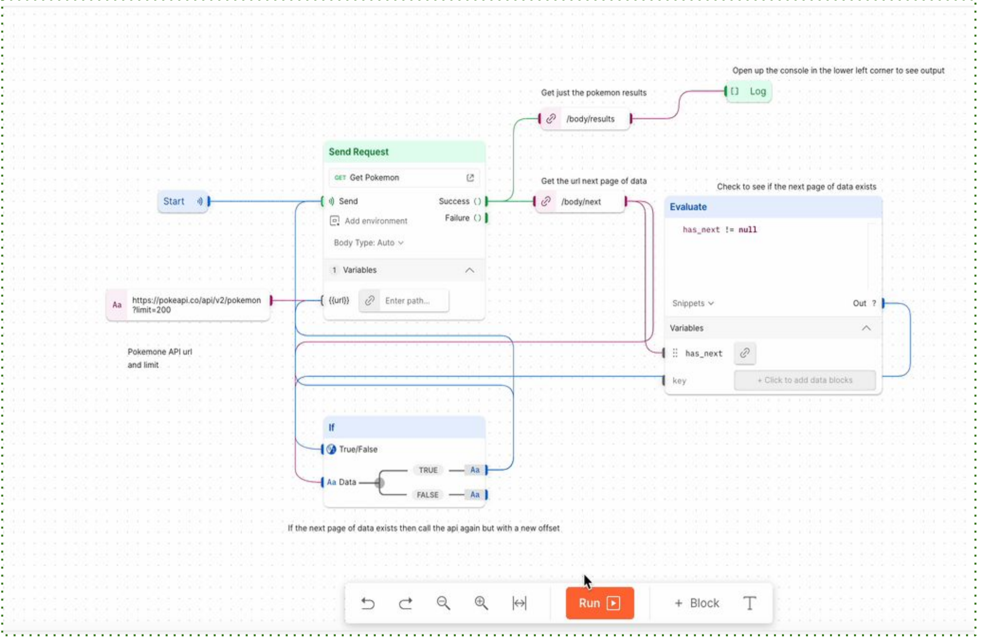
API Data Flow:

Postman Flows

Flows is a visual editor that lets you directly manipulate APIs and observe the data flowing between them. You can use Flows to chain requests, handle data, and create real-world workflows in your Postman workspace.

Blocks:

1) Task 2) Logic
3) Looping 4) Output



Open up the console in the lower left corner to see output

Get just the pokemon results

Get the url next page of data

Check to see if the next page of data exists

Pokemone API url and limit

If the next page of data exists then call the api again but with a new offset

```
has_next != null
```

Let’s explorer the flows

Congratulations 🎉

We Solved

- **Documentation Deficiency:** Their APIs lack proper documentation. For instance, details on how their partners should call API endpoints from different clients, the necessary payload, headers required for sending requests, etc., are missing.
- **Validation and Testing Gaps:** There are no solutions in place to validate or test their API responses.
- **Collaboration Challenges:** There is a problem with collaboration and easily sharing APIs across the development team.
- **Environment Mix-up:** They cannot easily manage their APIs in different environments (e.g., development, testing, production).

- ❑ **Performance Analysis Dilemma:** They lack an easy solution to analyze and identify performance issues, errors, or any unexpected behavior in their APIs, hampering improvements.
- ❑ **Frontend-Backend Coordination Issues:** They regularly observe that their frontend teams sometimes wait for the backend team to complete APIs, causing delays in completing projects on time.
- ❑ **Unclear authorization process:** Toto Company uses OAuth2 and OpenID Connect to secure their APIs. However, their backend team often receives questions from the frontend team, such as "Frontend team is unable to authorize and obtain a token from the authorization server." This is common for the frontend team because the backend team cannot demonstrate simple steps to work with their Auth server.

- ❑ **Data-Driven Task Hurdles:** Sometimes they want to perform Data-Driven testing.
- ❑ **Automated Testing Goals:** They also aim to automate their API testing process, generating reports automatically.
- ❑ **API Data Flows:** Their backend team desires a system where they can directly manipulate APIs and observe the data flowing between them without creating a new application. This helps them create a mind map of their API functionality.

Thank You