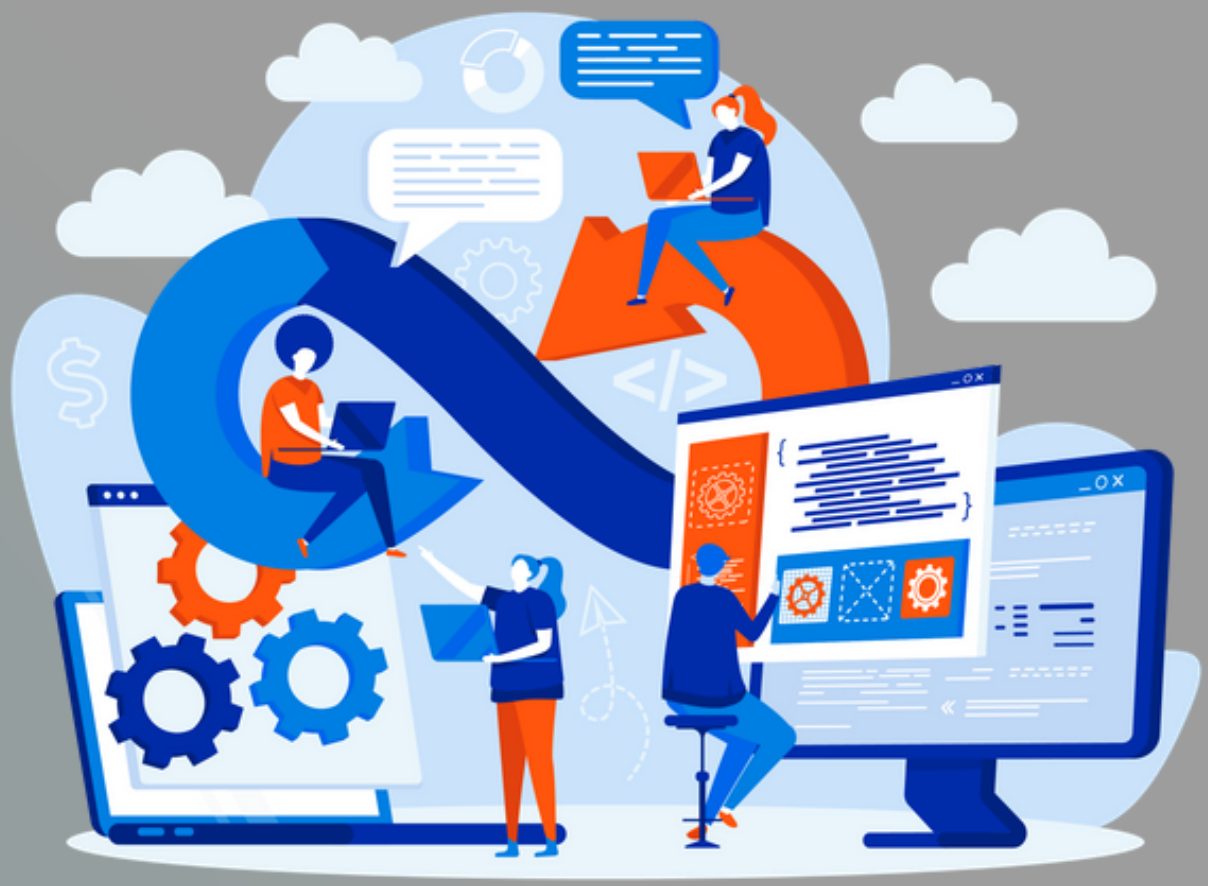
API development best practices >>>> that you should follow.



Rithikesh Surthi

SWIPE

➤ Pagination and filtering

Some endpoints may return substantial amounts of data that slow down a system.

Pagination and filtering avoid this by returning only a certain number of results at a time. This improves performance and reduces the usage of server resources.

➤ Use JSON

JSON is the standard format for transferring data. JavaScript has built-in methods to parse JSON quickly, and it is supported in almost all programming languages.

For simplicity, APIs should accept JSON payloads, and endpoints should return JSON as a response.

➤ Do not use verbs in endpoint names

Endpoint paths should always be named in reference to the entity they represent, not the action being carried out.

The HTTP method is sufficient to describe the action.

For example:

❌ POST / createArticle

✅ POST / article

Rithikesh Surthi

SWIPE

➤ Use plural nouns for resource collections

Be precise about naming data collections by using plurals.

For example, a collection of users should be ‘/users’ and never ‘/user’.

➤ Use resource nesting

Applying logical resource nesting is important to construct a hierarchy of your resources.

For example: example. com / blogs / blogId / blogAuthor

Keep in mind that deeper resource nesting (more than 3 levels) should be avoided.

➤ Implement timeouts

Timeouts cause a request to fail after a specified amount of time.

This is useful when there is a network issue, and the request cannot be completed, or a user sends too much data.

This connection is closed instead of remaining open.

➤ Use caching

Using Cache-Control headers will allow users to make effective use of cached data.

Caching allows users to access data faster because it is stored locally, meaning another request to the server to retrieve it is not needed.

➤ Use status codes in error handling

HTTP status codes convey information about the result of a client's request.

This assists developers and users with debugging.

➤ Maintain good security practices

As with any data transfers over the internet, SSL/TLS security certificates are a must if data is to be kept encrypted and safe.

Without it, data is at risk of being exposed and is susceptible to MITM (Man-in-the-middle) attacks, and more.

Rithikesh Surthi

SWIPE

➤ Versioning

Versioning provides you with a backup if you make breaking changes to your API.

It also lets you phase out old endpoints and introduce new ones with new features.

There are various methods of versioning, such as URI path, query parameters, and custom headers.

Rithikesh Surthi

SWIPE

➤ Document your API

Documentation is vital to communicate to other developers how to use your API and all of its features.

API docs should contain info on your API's endpoints, code snippet examples in various programming languages, tutorials, debugging info, and more.

WERE YOU ABLE TO FIND ANY USEFUL INFORMATION IN THIS POST?

FOLLOW ME FOR
MORE SUCH CONTENT

Rithikesh Surthi