Tech Winter GDG On-Campus VIT-AP

Backend HTTP Requests

Let's Start

https://nikhiltanneeru.github.io/My-Portfolio/

nikhiltanneeru4@gmail.com

About Me

Name   : Nikhil Tanneeru

Email   : nikhiltanneeru4@gmail.com

LinkedIn : https://www.linkedin.com/in/nikhil-tanneeru/

GitHub   : https://github.com/NikhilTanneeru

Portfolio : https://nikhiltanneeru.github.io/My-Portfolio/

#OPENTOWORK

Nikhil Tanneeru
Computer Science Student at Vellore
Institute of Technology, Amaravati | Pr.

nikhiltanneeru4@gmail.com

Contents

1	**Introduction To API**	**3**
2	Methods in API	8
3	Implementation of HTTP Requests	15
4	Uses and Advantages	16

The first documented use of the acronym "API" was in 1967 by computer programmer Carl Hewitt.

LET'S DIVE DEEP

GDG On-Campus

VIT-AP

Home

About

Content

Others

Page 04

Introduction to API

Introduction to API

- API stands for Application Programming Interface. It is a set of rules and protocols that allows different software applications to communicate with each other.

- APIs work like messengers: they take your request to a system, get the needed data or action, and deliver the response back to you

- APIs are used in web applications to connect user-facing front ends with back-end functionality and data.
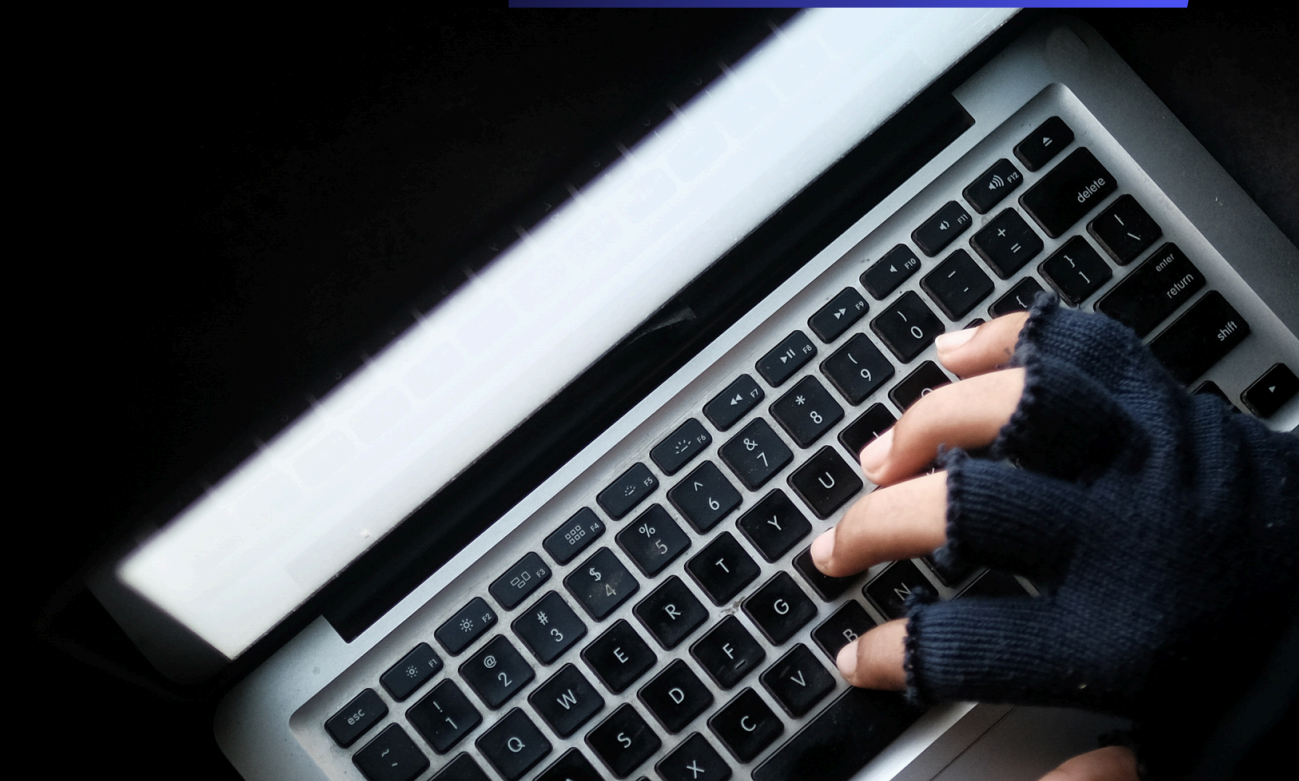
Introduction to API - Types Of API(Based on Architecture)

Web APIs (HTTP APIs):

These APIs allow applications to communicate over the internet using the HTTP/HTTPS protocol.

➢ REST APIs (Representational State Transfer):Used for simple, lightweight communication.

▪ Example: A weather app fetching the current temperature from a server.

➢ SOAP APIs (Simple Object Access Protocol):Used for more complex systems requiring high security and structure. ▪ Example: Payment processing in banking systems.

Introduction to API - REST API

- REST is a lightweight, web-friendly architecture for building APIs. It uses standard HTTP methods (GET, POST, PUT, DELETE) and is often implemented with JSON or XML for data exchange.
- Key Features:
  - Stateless: Each request is independent and contains all the information needed.
  - Uses URLs to access resources.
  - Simple and scalable.
  - Commonly uses JSON for data transfer.

Example:

GET https://api.example.com/users   ->   Returns a list of users in JSON format.

Introduction to API - Types Of API(Based on Accessibility)

- Open API (Public):

    External users can access services.

    Example: OpenWeather API , Google Maps API.

- Internal API (Private):

    Used by employees for internal tools.

    Example: HR system API for employee management and payroll.

- Partner API:

    Shared with specific external partners.

    Example: PayPal API for e-commerce payment integration.

- Composite API:

    Combines multiple API calls into one request.

    Example: Fetching customer details and order history in one call.

HTTP Request Methods

- HTTP request methods are actions used to communicate with a server, such as retrieving, creating, updating, or deleting resources.
- Some of the major methods we use are:
- GET, POST, PUT, DELETE
- These methods are used for CRUD operations.

CREATE	**POST**
READ	GET
UPDATE	PUT
DELETE	DELETE

HTTP Request Methods - Data

Consider the JSON as a database

```json
[
 {
  "id": 1,
  "name": "Nikhil Tanneeru",
  "email": "nikhiltanneeru4@gmail.com",
  "age": 20,
  "role": "admin"
 },
 {
  "id": 2,
  "name": "Rahul",
  "email": "rahul@email.com",
  "age": 21,
  "role": "editor"
 }
]
```

HTTP Request Methods - GET

- The GET method is used to request data from a server without making any changes to the data. It's read-only and does not affect the resource.
- Example:

Fetch user data → GET /users/1

- This will retrieve the details of the user with ID 1.

HTTP Request Methods - POST

- The POST method is used to send new data to the server to create a resource. It is often used in forms where you submit data.

- Example:

  Create a new user → POST /users with user details like name and email.

- This will add a new user to the database.

HTTP Request Methods - PUT

- The PUT method is used to update or replace an entire existing resource. If the resource doesn't exist, it can create it in some cases.

- Example:

- Update user info → PUT /users/1 with new details like name and age.

- This can replace the user's information completely.

HTTP Request Methods - DELETE

- The DELETE method is used to remove a resource from the server.

  Once deleted, the resource cannot be retrieved.

  Example:

    Delete a user → DELETE /users/1

- This will remove the user with ID 1 from the database.

HTTP Request Methods - Implementation

Let's explore these methods by practical hands-on implementation

Advantages of using HTTP Requests

1. Standardized Communication:
   - Follow a universal protocol ensuring interoperability across platforms and systems.
2. Clear Intentions:
   - Each method specifies its purpose (e.g., GET for retrieving, POST for sending data), simplifying server processing.
3. Efficient Resource Management:
   - Different methods enable efficient handling of resources on the server, reducing unnecessary load (e.g., HEAD retrieves only headers without a full response).

Thynk
Unlimited.

Home

About

Content

Others

Page 17

Nikhil Tanneeru
Computer Science Student at Vellore
Institute of Technology, Amaravati | Pr...

https://nikhiltanneeru.github.io/My-Portfolio/

nikhiltanneeru4@gmail.com

Thynk
Unlimited.

Home

About

Content

Others

Page 18

Thank You