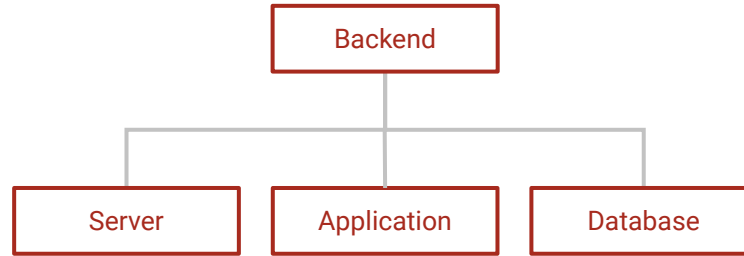


A large red square with a white border, centered on a white background. Inside the square, the text "Back-End Development" is written in white.

Back-End Development

What is the Back-End?



The back end (“server-side”) handles business logic and data storage.

The stuff we **DON'T** see

Cloud Computing Services

“Every company is a tech company”

Challenges of building a backend:

Scaling for Traffic

No backend “management”

Server Failures...

Abstraction from hardware issues



Google Cloud



Why Cloud?

- Build products without managing infrastructure (just write code..)
- Don't need to handle scaling
 - For example: a website suddenly gets a huge spike in traffic

Firebase

Mobile and Web App Development Platform

Firebase Project = Google Cloud Project

- Wrapper around GCP

Site: <https://firebase.google.com/>



Firebase



Google Cloud

Node.js

JavaScript Runtime Environment

- Node.js includes the libraries, environment variables, PATHs, and structures necessary to compile JavaScript code

Mainly for

- “Designed to build scalable network applications”

Popularity

- One of the most popular backend languages alongside Java and Python



Documentation:

<https://nodejs.org/en/docs/>

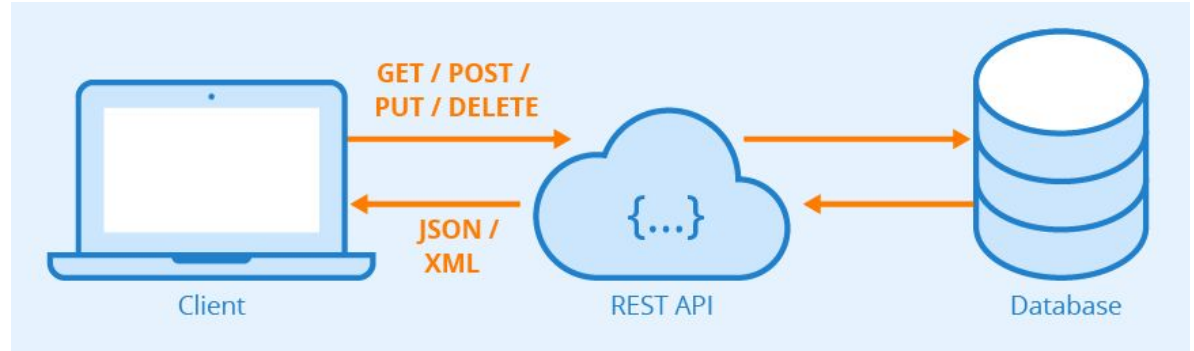
REST (Representational State Transfer)

HTTP Methods - [Official HTTP Method Registry](#)

- GET
- PUT
- POST
- DELETE

Note: Potential Interview Topic

Many other HTTP Methods



CRUD

- POST - **C**reate
- GET - **R**ead
- PUT - **U**ppdate
- DELETE - **D**elelete



Cloud Functions for Firebase

Cloud Functions for Firebase is a serverless framework that lets you automatically run backend code in response to events triggered by Firebase features and HTTP requests.

Serverless Architectures are apps that incorporate “Backend-as-a-Service” (**BaaS**) services

Learn More: <https://firebase.google.com/docs/functions>



HTTP Request (Hypertext Transfer Protocol)

HTTP is the foundation for communication on the web

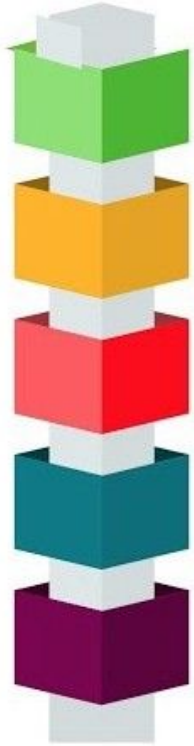
HTTP defines how messages are formatted and communicated

HTTP defines a set of request methods to indicate the desired action to be performed for a given resource.

Typically **JSON** (JavaScript Object Notation)

<https://www.json.org/json-en.html>

HTTP Status Codes



1XX
INFORMATIONAL

2XX
SUCCESS

3XX
REDIRECTION

4XX
CLIENT ERROR

5XX
SERVER ERROR

Standardized form of communication

Learn more:

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>

Cloud Firestore for Firebase

Cloud Firestore is a flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud Platform.

NoSQL cloud database

Querying Language

Different from relational databases like SQL

“Collections”... “Documents”... “Fields”

Alternative: Firebase Realtime Database



Cloud
Firestore

Next Steps:

Next Steps:

1. Set up Firebase
2. Design REST API
3. Code REST API
4. Test REST API

Designing a clean documented API is **VERY** important when working on large back-end applications



1. Set up Firebase

Prerequisites:

- Google Account
- Node.js

1. Set up Firebase (Continued)

Installation

- `npm install -g firebase-tools@latest`
- `npm install express`
- `npm install cors` (we will go into this next week)

Get Key from Firebase Console

Project Settings/Service Accounts, Generate a new Key

Optional

- Install Postman from www.postman.com

Troubleshoot (Windows)

- FAIL: `npm install`
 - ↳ Install via standalone binary
 - ↳ `curl -sL https://firebase.tools | bash`

Troubleshoot (macOS)

- FAIL: `npm install`
 - ↳ Install via standalone binary **OR**
 - ↳ `curl -sL https://firebase.tools | bash`

Official Documentation

- <https://firebase.google.com/docs/cli>

2. Design REST API

Implement 4 basic HTTP verbs for the CRUD operations

- **GET** - get application data
- **POST** - create application data
- **PUT** - modify application data
- **DELETE** - delete an application data

3. Code REST API

firebase login

firebase projects:list

firebase init

4. Test

Run locally

```
firebase serve --only functions
```

Deploy to production server

```
firebase deploy
```

Troubleshoot:

Port Error

- killall node