

Backend Development For Beginners

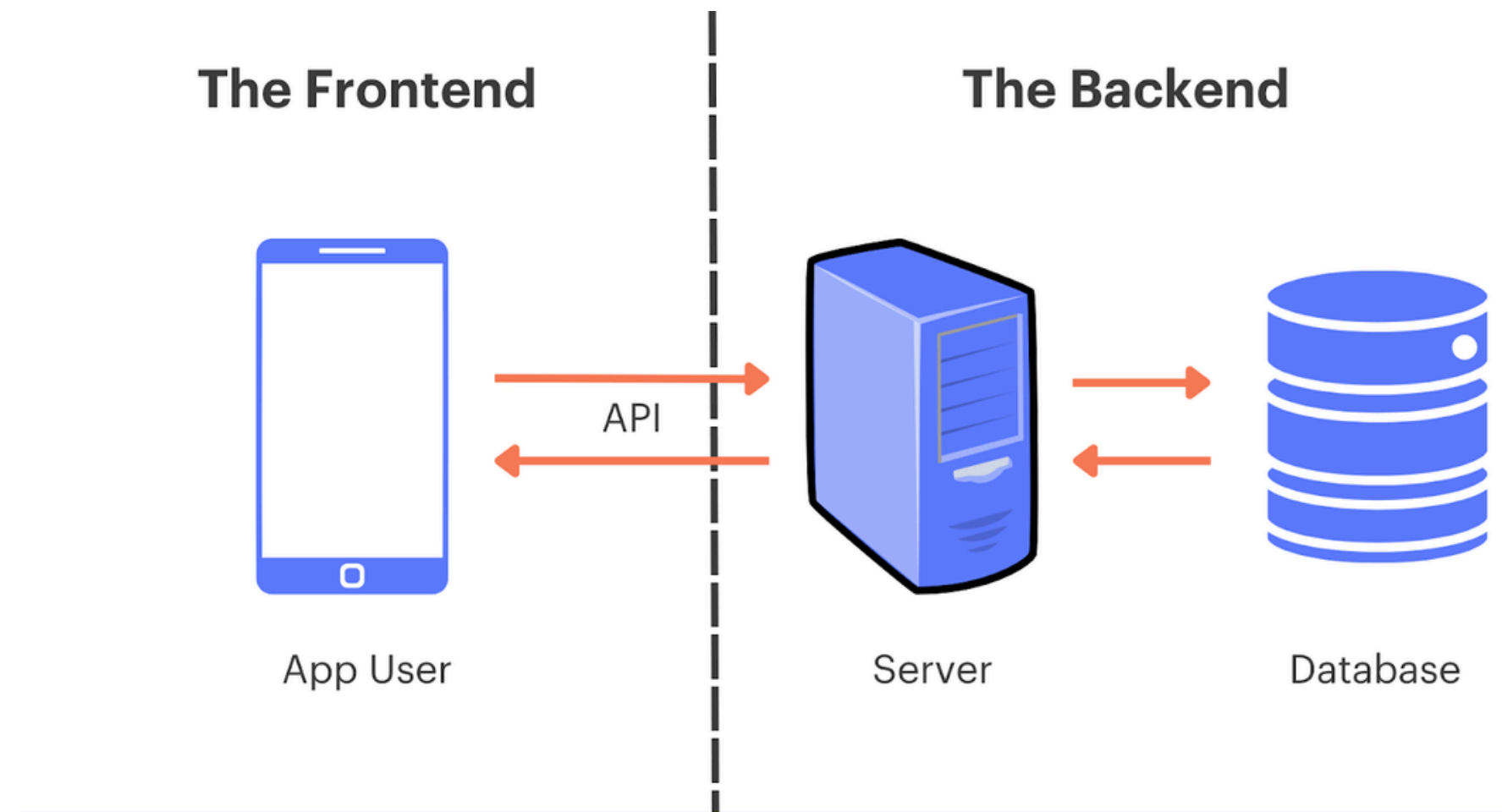
Presenter

- Suraj Pandey
- Backend Developer

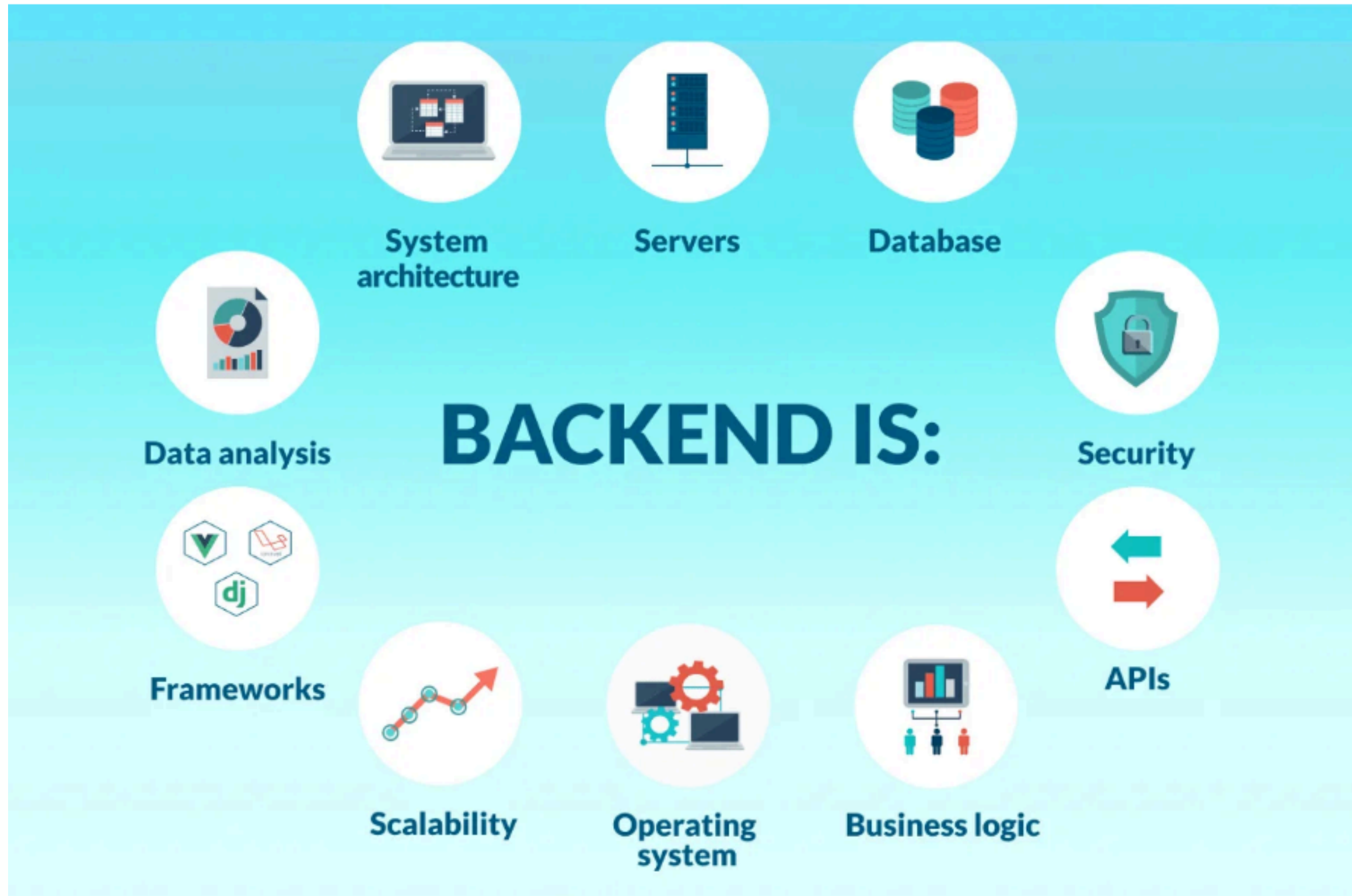


INTRODUCTION TO BACKEND DEVELOPMENT

- **Behind-the-scenes** magic that makes websites and apps work.
- Powers the **database, server, and application** — basically, everything that happens on the server side.
- As a back-end web developer, you're the one who **bring a website to life**. Without you, the website wouldn't function.



BACKEND DEVELOPMENT INCLUDES



PROGRAMMING LANGUAGES AND FRAMEWORKS

Programming Languages

- Python
- Java
- **JavaScript**
- Ruby
- SQL
- Golang

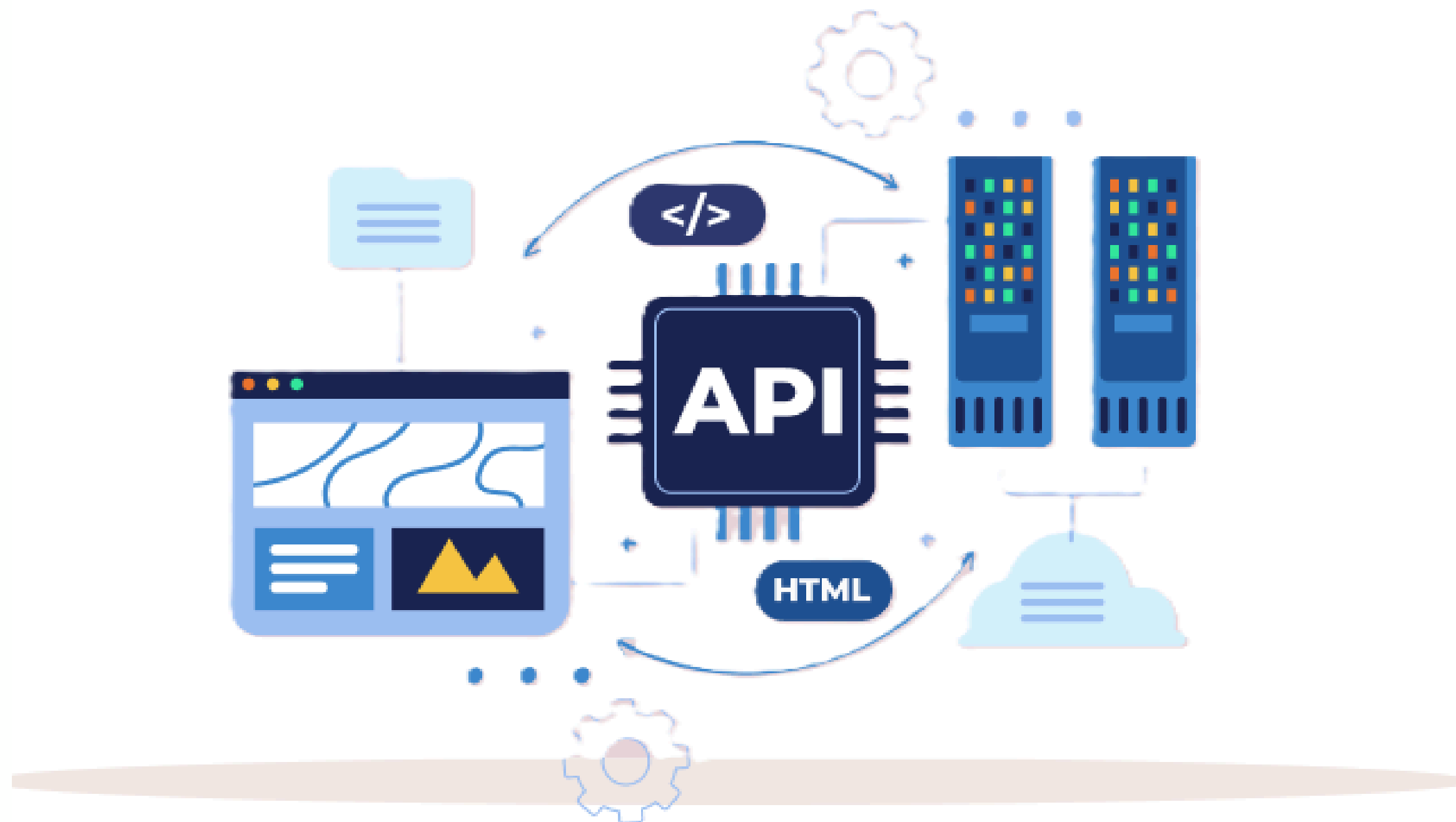


Frameworks

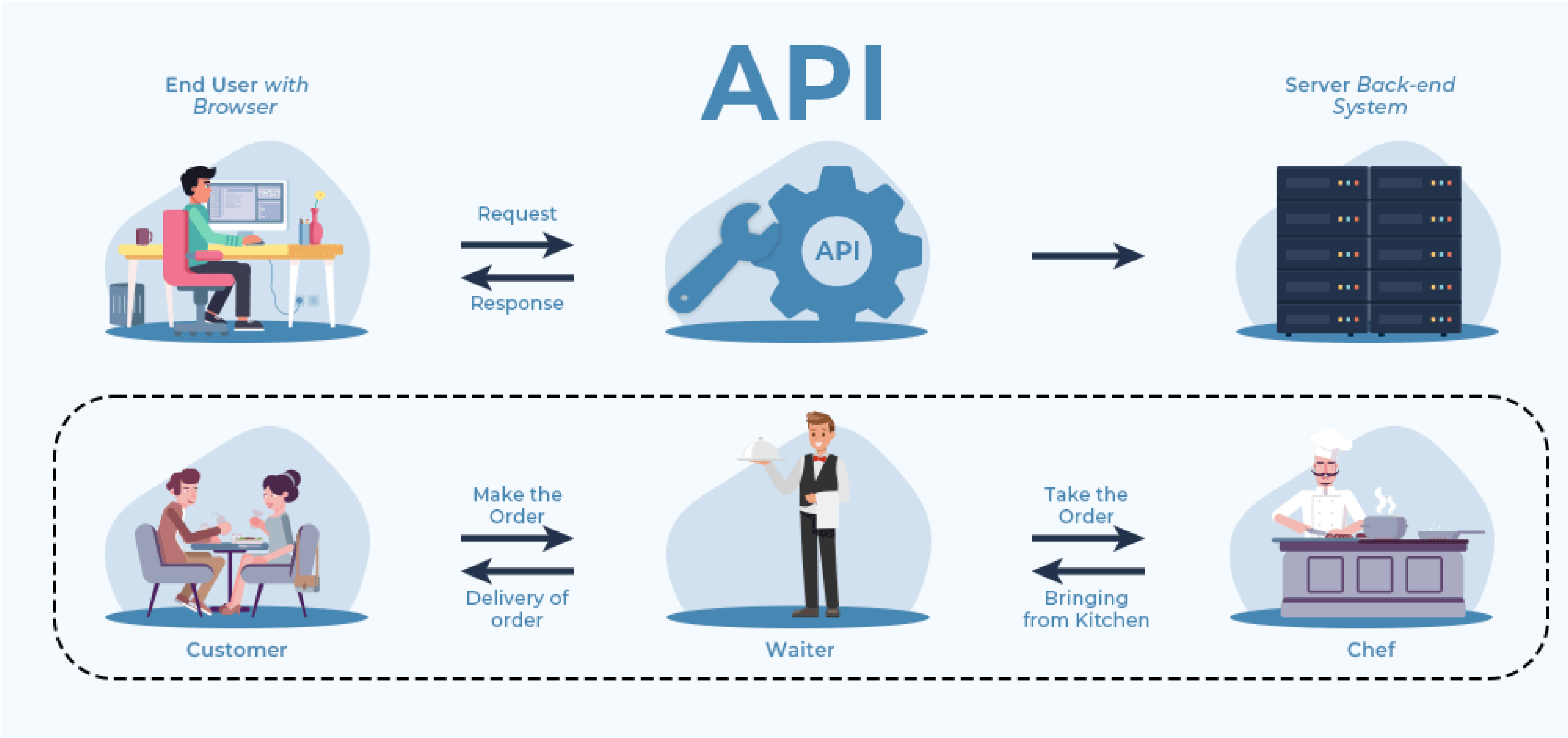
- Laravel
- **Express(Nodejs)**
- Flask
- Ruby on Rails

API(application programming interfaces)

- Allows backend to **communicate** with the front end.

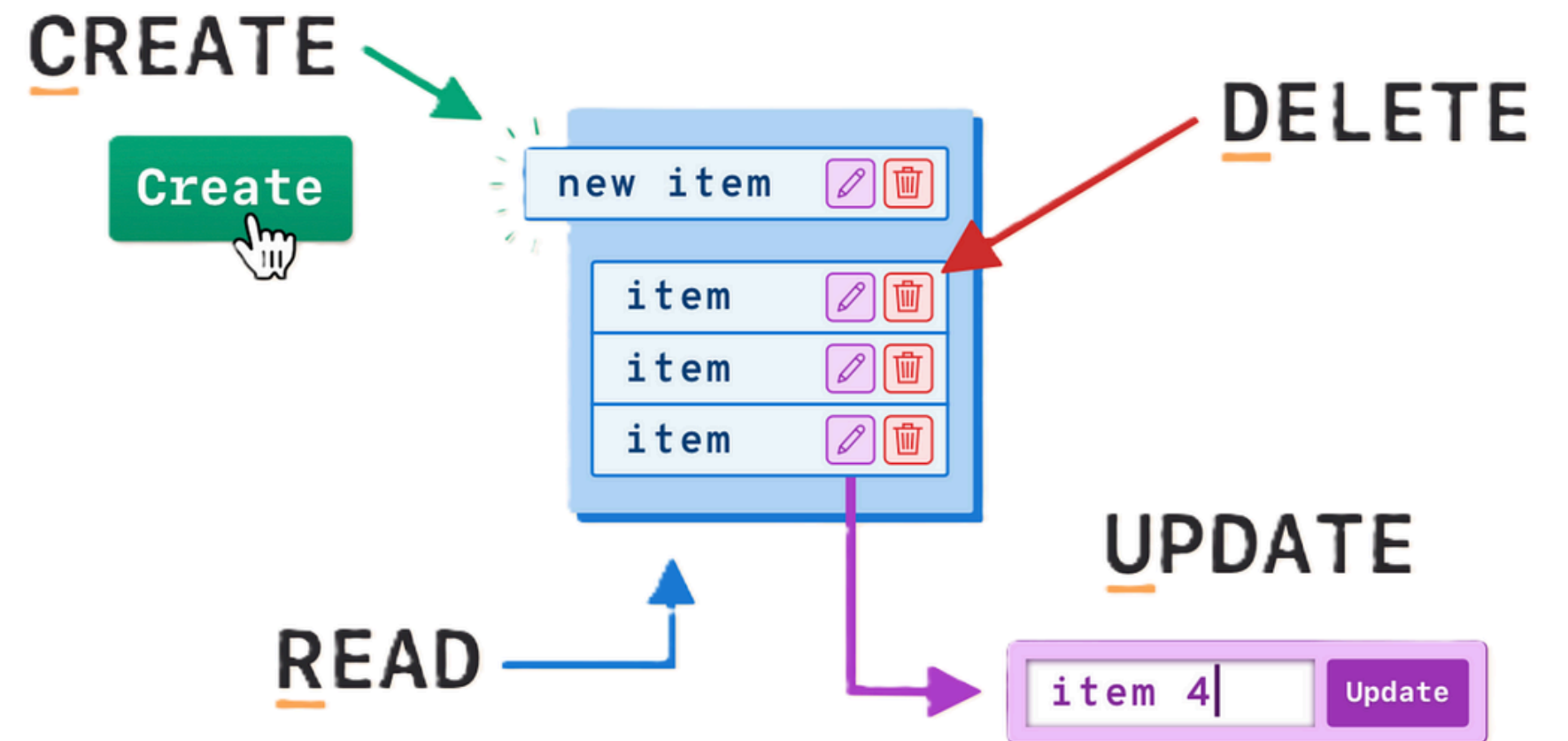


API working mechanism

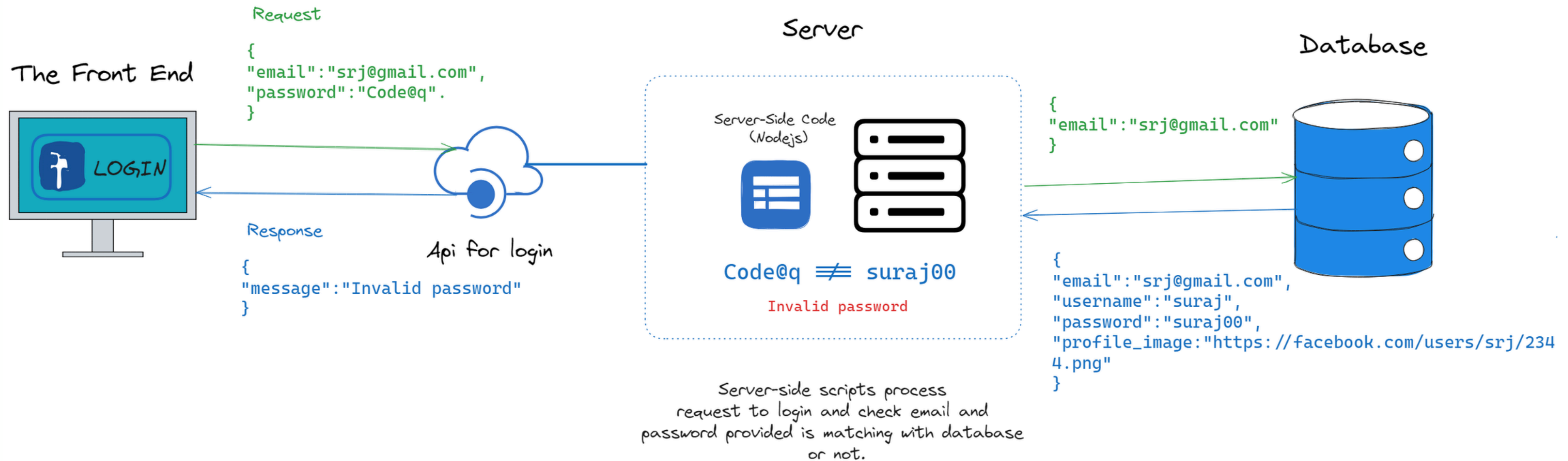


Crud Operation

- **CRUD** (Create, Read, Update, and Delete)
- **API's enables users to:**
 - create new records on the databases,
 - **fetch** existing data,
 - **update** specific entries,
 - or **delete** certain or all records.



WORKFLOW OF LOGIN IN FACEBOOK

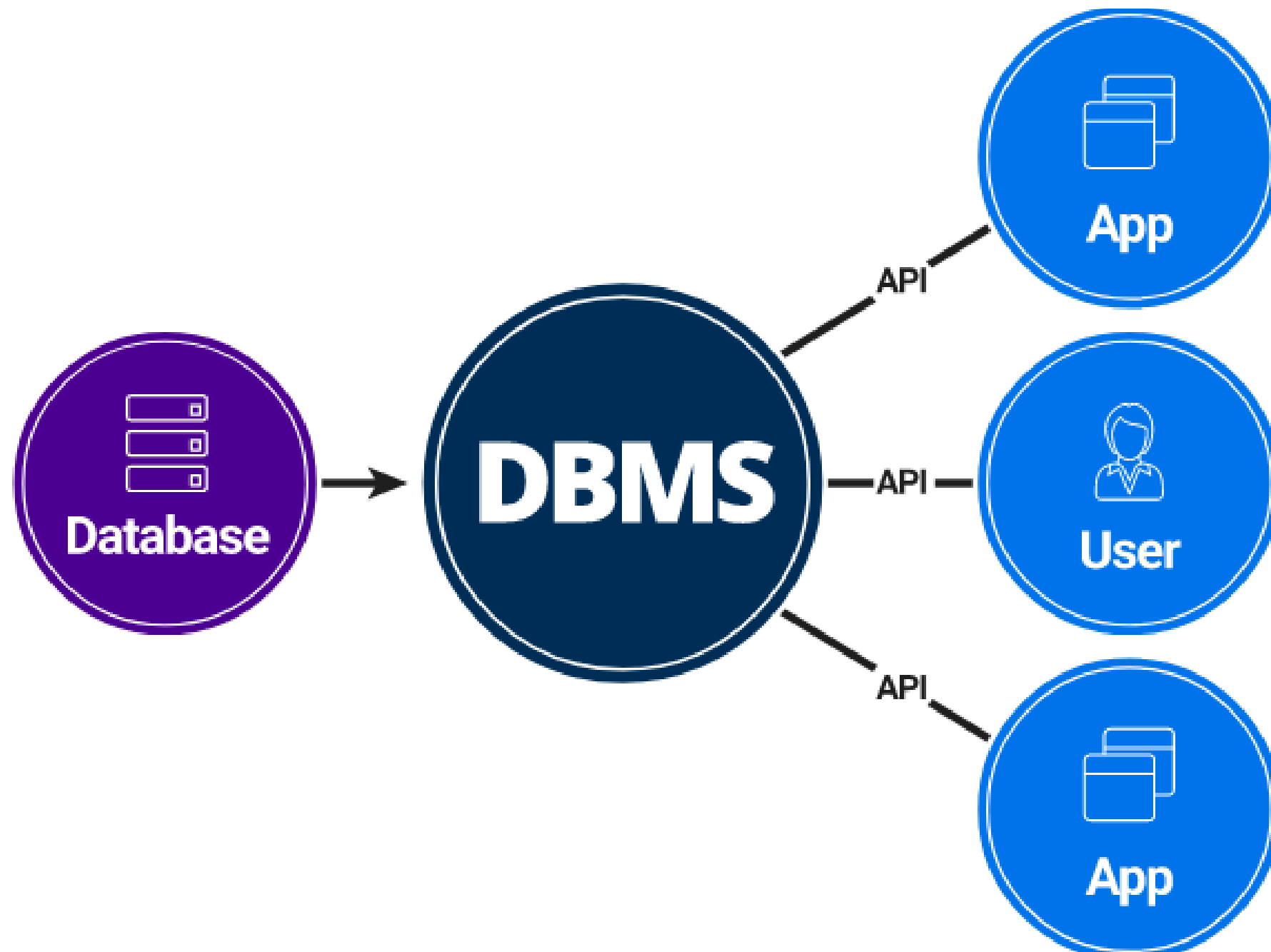


DATABASE AND DBMS(DATABASE MANAGEMENT SYSTEM)

- **Database:** organized **collection** of structured information or **data**, typically stored electronically in a computer system.
- The data within a database is often **managed** and **accessed** using a Database Management System (DBMS)



DATABASE AND DBMS EXAMPLE

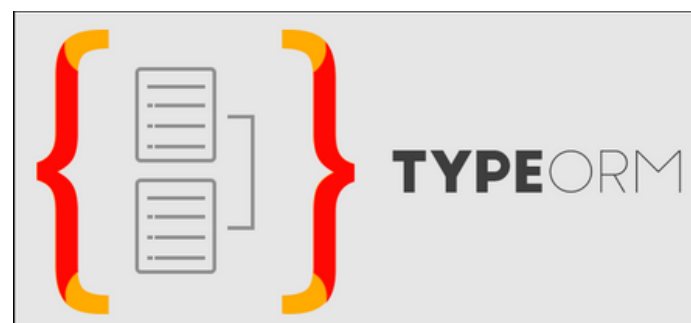


ORM(OBJECT RELATIONAL MAPPING)

Tool that helps you **interact** with a database using your **programming language** without writing **complex** SQL queries.

Advantages:

- Significantly reduce boilerplate code,
- Make working with databases a lot easier,
- Helps to maintain and update code more easily.



RAW SQL VS PRISMA CODE

SQL query to create table

```
-- Create User table
CREATE TABLE users (
  id SERIAL PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  email VARCHAR(100) UNIQUE NOT NULL,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Create Post table
CREATE TABLE posts (
  id SERIAL PRIMARY KEY,
  user_id INT NOT NULL,
  title VARCHAR(255) NOT NULL,
  content TEXT NOT NULL,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
);
```

Prisma code to create table

```
// schema.prisma
model User {
  id      Int      @id @default(autoincrement())
  name    String
  email   String   @unique
  createdAt DateTime @default(now())
  posts   Post[]
}

model Post {
  id      Int      @id @default(autoincrement())
  userId  Int
  title   String
  content String
  createdAt DateTime @default(now())
  user    User     @relation(fields: [userId], references: [id])
}
```

RAW SQL VS PRISMA CODE - TO FETCH POST DATA OF USERS

SQL query

```
SELECT
  u.id AS user_id,
  u.name AS user_name,
  u.email AS user_email,
  p.id AS post_id,
  p.title AS post_title,
  p.content AS post_content,
  p.created_at AS post_created_at
FROM
  users u
JOIN
  posts p ON u.id = p.user_id;
```

Prisma code

```
async function main() {
  const postsWithUsers = await prisma.post.findMany({
    include: {
      user: true, // Includes the user data in the post query
    },
  });

  console.log('Posts with users:', postsWithUsers);
}
```

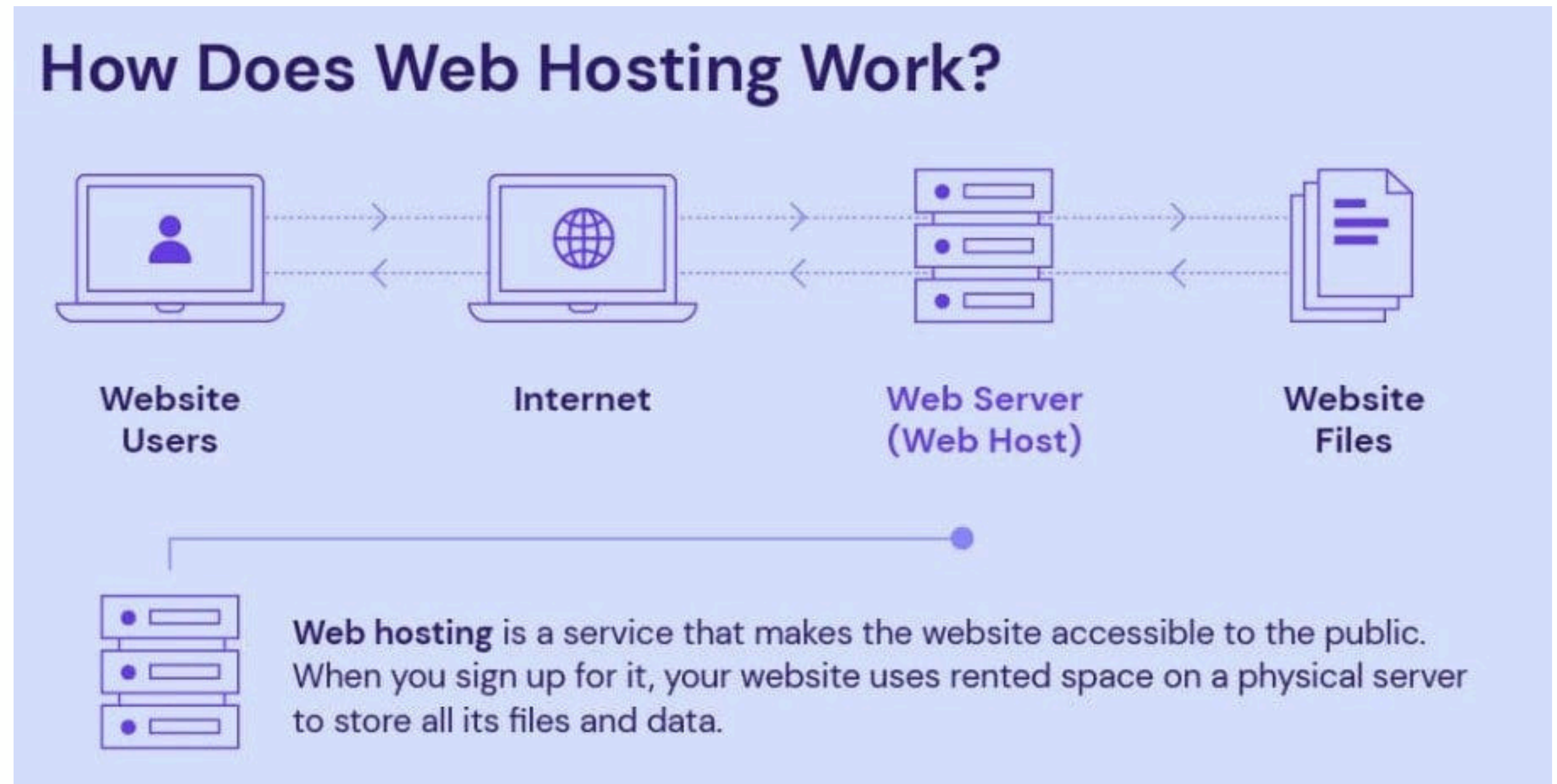
WEB HOSTING

- Essential for making your website **live** and **publicly** available.
- Think of web hosting like **renting an apartment** for your website.
- Service provider provides **hosting facility** for our website i.e. **web host**.
- Popular web host:
 - Amazon Web Services (AWS), Google Cloud Platform (GCP), Google Cloud Platform (GCP)



KEY COMPONENT WEB HOSTING

- Domain Name
- Bandwidth
- Security
- Server
- Storage
- Email Accounts
- Support



ESSENTIAL SKILLS FOR BACK-END DEVELOPERS

- Basic understanding of the frontend ecosystem
- Knowledge of server side languages
- Knowledge of database technologies
- Security and authentication best practice
- Problem-solving skills

Thank you!

KEEP LEARNING, KEEP GROWING WITH

