HOW SOFTWARE WORKS · START HERE · 14 MIN READ

How apps work: front end, back end, database

The parts of every web app and the words for them, so you can describe what you want and understand what the AI built.

Pass the check (4 of 5) online to earn the “Stack Mapper” badge:
https://digitalbuilders757.com/university/study/how-apps-work

A restaurant, not a box

Think of a web app as a restaurant. The front end is the dining room: what customers see and touch. The back end is the kitchen: where orders are checked and prepared. The database is the pantry and the order book: where everything is stored. The API is the waiter carrying orders between them.

- Front end (client): the pages, buttons and forms in the browser or phone app.
- Back end (server): code that runs on a computer you control, checks permissions and does the work users mustn't be able to tamper with.
- Database: organised storage for users, posts, orders and anything else that must be remembered.
- API: the agreed way for one piece of software to ask another for something.

The three languages of the web page

- HTML: the structure. Headings, paragraphs, buttons, images.
- CSS: the look. Colours, fonts, spacing, layout. Tailwind is a popular way to write CSS with short class names.
- JavaScript: the behaviour. What happens when you click, type or load. TypeScript is JavaScript with labels (types) that catch mistakes early.

Frameworks and packages

Almost nobody builds from scratch. A framework is a ready-made structure for building apps: React is the most popular for interfaces, and Next.js builds on React to handle pages, server code and fast loading. (This very website is built with Next.js.)

Packages are reusable pieces of code other people publish. npm is the package registry for JavaScript; a project's package.json lists the packages it uses. When the AI says 'install

a package', it means adding one of these.

Where code runs: your computer, the terminal and the cloud

While building, the app runs on your own computer (local development, usually at an address like localhost:3000). The terminal (or command line) is the text window where you run commands such as npm install or npm run dev.

When you deploy, the app's code is copied to computers in the cloud so anyone can use it at a real web address. Environment variables are settings and secrets (like database keys) that the app reads at run time, kept outside the code.

Sign-in: authentication and authorisation

Authentication answers 'who are you?' (signing in). Authorisation answers 'what are you allowed to do?' (permissions). Apps need both. A common vibe-coding bug is checking permissions only in the front end, which anyone can bypass. Real permission checks live in the back end or the database.

Key terms

Front end
The part of an app people see and touch: pages, buttons, forms.

Back end
Code on a server that checks permissions and does work users mustn't tamper with.

Database
Organised storage for everything an app must remember.

API
An agreed way for one piece of software to ask another for something.

HTML
The language for a web page's structure: headings, paragraphs, buttons, images.

CSS
The language for how a page looks: colours, fonts, spacing, layout.

JavaScript
The programming language that makes web pages interactive.

TypeScript
JavaScript with types that catch mistakes before the code runs.

Framework
A ready-made structure and toolkit for building apps.

React
The most popular JavaScript library for building interfaces from components.

Next.js
A React framework for pages, server code and fast loading, made by Vercel.

npm / package
npm is the registry of reusable JavaScript code; each reusable piece is a package.

Terminal (command line)
A text window where you run commands like npm install.

Localhost
Your own computer acting as the server while you build, e.g. localhost:3000.

Environment variable
A setting or secret the app reads at run time, kept outside the code.

Authentication
Checking who someone is (signing in).

Authorisation
Checking what a signed-in person is allowed to do.

Try it: Take an app apart

1. Pick an app you use daily (a food delivery app works well).
2. List what lives in its front end, its back end and its database.
3. Name two APIs it probably calls (payments, maps, messages).
4. In any web page, right-click and choose Inspect to see its HTML and CSS.

KNOWLEDGE CHECK

How apps work: front end, back end, database

5 questions. Circle one answer for each; 4 or more correct passes. Answers are at the back.

1. In the restaurant picture, the database is:

A. The dining room where customers sit and order
B. The pantry and order book
C. The waiter who carries orders to the kitchen
D. The menu font

2. Which language controls how a web page looks (colours, spacing, fonts)?

A. HTML
B. SQL
C. CSS
D. Python

3. What is an API?

A. A kind of computer virus
B. A file format for photos
C. A programming language for web pages
D. An agreed way for one piece of software to ask another for something

4. Why must permission checks live in the back end or database, not only the front end?

A. Anyone can change or bypass code running in their own browser
B. The front end is too slow to check permissions before the page appears
C. Front ends can't show buttons
D. It looks nicer

5. What is Next.js?

A. A database service that stores your app's users and their data
B. A framework built on React for building web apps
C. A web host that puts your site online
D. A password manager

Official sites

MDN: Learn web development: https://developer.mozilla.org/en-US/docs/Learn_web_development
React: https://react.dev
Next.js: https://nextjs.org

Last reviewed 2026-11-25. AI tools change fast; check each tool's own site for current features, plans and prices.

ANSWER KEY

How apps work: front end, back end, database

1. B: The pantry and order book
The database is where the app keeps what it must remember.

2. C: CSS
HTML is structure, CSS is the look, JavaScript is behaviour.

3. D: An agreed way for one piece of software to ask another for something
APIs are how software talks to other software, like an app asking Stripe to take a payment.

4. A: Anyone can change or bypass code running in their own browser
Code in the browser is in the user's hands. Trust only checks that run on your side.

5. B: A framework built on React for building web apps
Next.js is a React framework for pages, server code and performance.