

Express.js

→ Express is a Node.js framework that makes building web server easy. It helps handle requests and responses, so you don't have to write lots of code.

Eg:

you can create routes like `GET /api/quotes` to send data back.

Prisma ORM

→ Prisma is a tool that lets you work with databases using JS instead of SQL. It acts like a bridge between our code and the database.

Its benefit is that it saves time and reduces error compared to writing raw SQL.

Middleware

→ Middleware are functions that run between the request and response. They can do things like parse JSON, check authentication or log in before the final response is sent.

Fetch API

→ Fetch is a browser API that lets us request data from a server asynchronously. It returns a promise, which resolves when the data is received.

eg: To get quotes from our API we use:

```
fetch('/api/quotes')
```

```
.then(response => response.json())
```

```
.then(data => {
```

```
});
```

It helps build dynamic web pages that update data without needing to reload the whole page.

Routing

→ Routing means deciding how your app responds to different URLs or endpoints. For eg.

For eg. When someone visits /api/quotes, the server runs the code that sends back all quotes. Routing helps organize different features by URL paths.

MongoDB

→ MongoDB is a NoSQL database that stores data in JSON-like documents. It's flexible and great for projects where data structures can change often. It works well with Prism and Node.js.