Server

In this lesson, we will explore the server component of the client-server architecture.

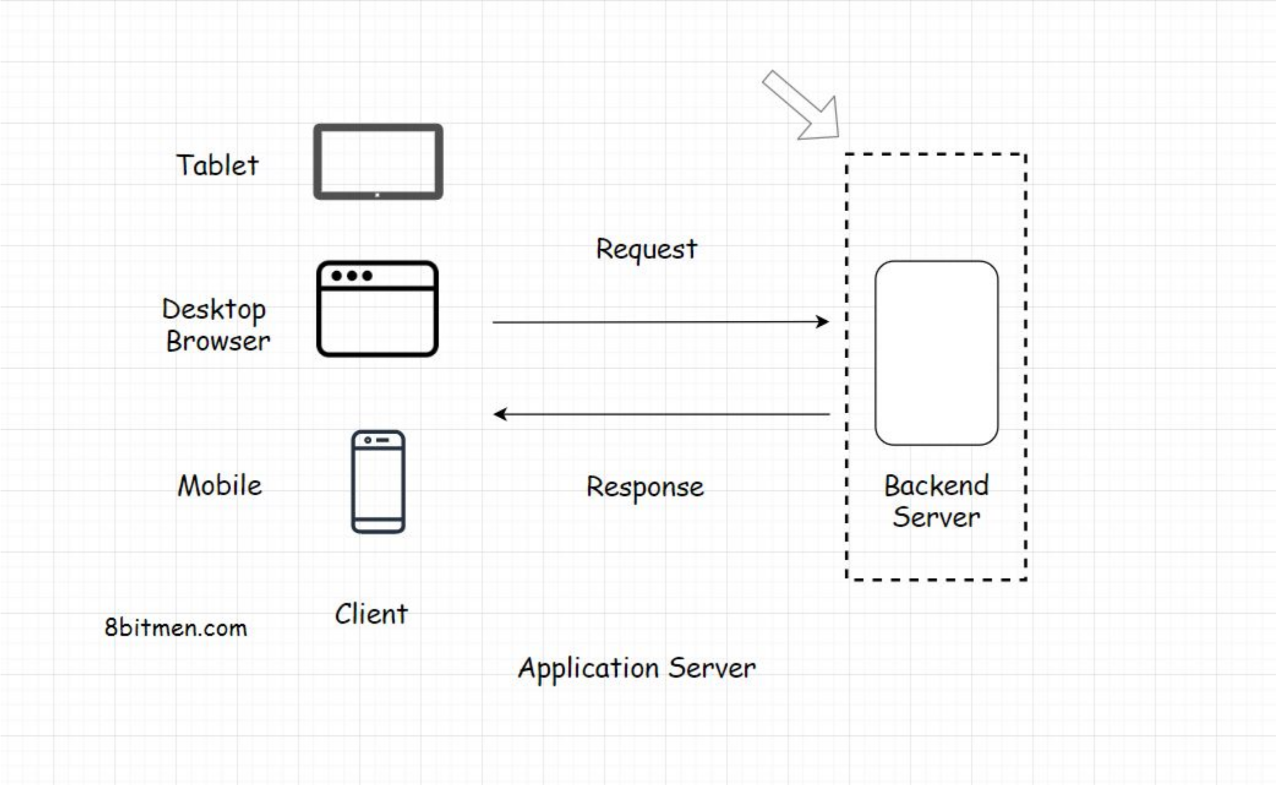
We'll cover the following

- What is a web server?
- Server-side rendering

What is a web server?#

> The primary task of a web server is to receive the requests from the client and provide the response after executing the business logic based on the request parameters received from the client.

Every service, running online, needs a server to run. Servers running web applications are commonly known as the *application servers*.



Besides the *application servers*, there are also other kinds of servers with specific tasks assigned to them. These include the:

- *Proxy server*
- *Mail server*
- *File server*
- *Virtual server*

The server configuration and the type can differ depending on the use case.

- For instance, if we run a backend application code written in *Java*, we would pick *Apache Tomcat* or *Jetty*.

- For simple use cases such as hosting websites, we would pick the *Apache HTTP Server*.

In this lesson, we will stick to the *application server*.

All the components of a web application need a server to run, be it a database, a message queue, a cache, or any other component. In modern application development, even the user interface is hosted separately on a dedicated server.

Server-side rendering#

Often the developers use a server to render the user interface on the backend and then send the rendered data to the client. This technique is known as *server-side rendering*. I will discuss the pros and cons of *client-side* vs *server-side* rendering further down the course.

Now we have a fundamental understanding of both the client and the server. Let’s delve into some of the concepts involved in the communication between them.

← Back

Types of Clients

Next →

Communication Between the Client an...

☑ Completed

Report an Issue