

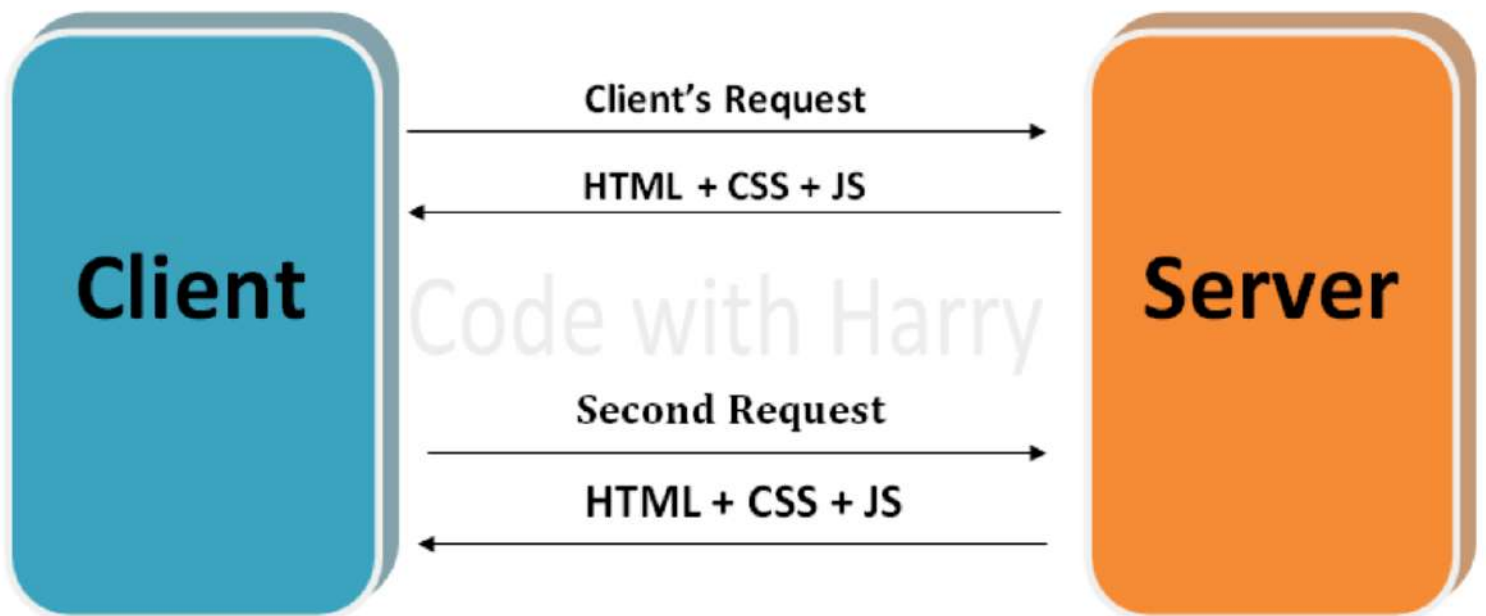
Project 1: Setup + Adding Bootstrap to React | Complete React Course in Hindi #5

Before creating our project, let's learn how a basic website works.

Multi-Page Application:

A request is being sent by the client to the server and in response to this request the HTML, CSS, JS of that specific page is rendered by the server in the client's device. In Multi-Page Applications, corresponding to every page their HTML, CSS, JS is rendered by the server.

For example: While visiting *codewithharry.com/contact* its HTML will be served and while moving to *codewithharry.com/about* then again its HTML will be served.



Single Page Application:

In this case, the request will be sent once by the client to the server, and all the HTML, CSS, JS will be rendered with the first request. Then, Javascript will take control of the whole application and with the help of APIs, the same page will be populated. This means that on the new request of the user it will serve the page without reloading it.

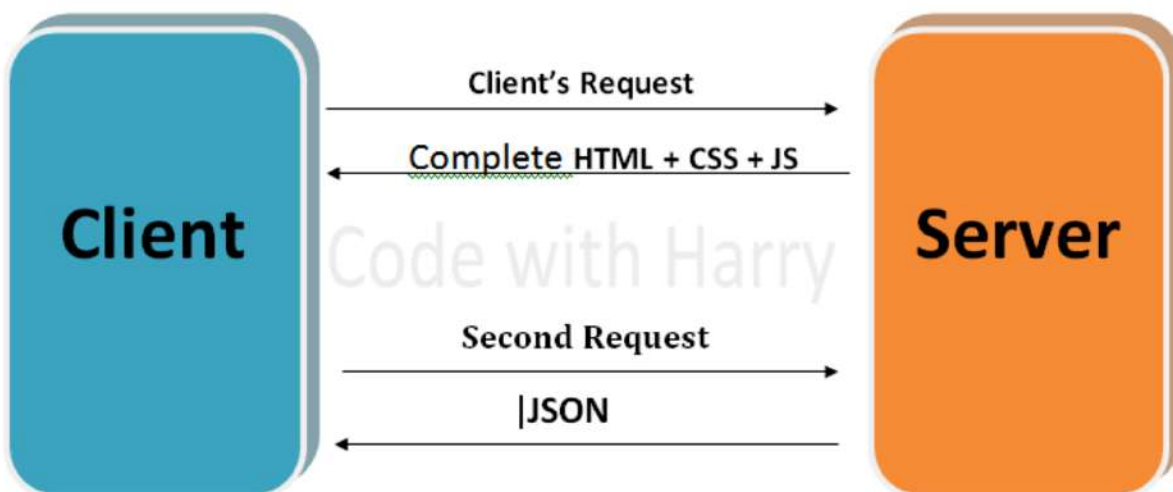


Figure-1.2: Single Page Application Working

Advantages of Single-Page Applications:

By using a single-page Application we can save a lot of bandwidth in the case of those applications which are getting a huge amount of traffic. It provides an improved user experience and also results in quick loading time.

Node module:

It contains all the packages which we needed to develop our React application.

Tip: Whenever we share our application, we make sure that we don't share the `Node_modules` folder. This folder can easily be recreated with the help of `Package.json` and `Package-lock.json` by using the `npm install` command. Due to this reason we put our `node_module` folder in `Gitignore` so that we accidentally don't share it.

Project-1: Creating a TextUtils app:

Description of our app: It is a text document editor. It will contain a Text box in which we will be adding our text.

Some of its features will be:

- Word Counting.
- Removing Extra spaces.
- Capitalizing the text of the document.
- Adding Lowercase and uppercase to the text.

Adding Bootstrap to our React App (Textutils):

Bootstrap: It is the framework of CSS. With the help of Bootstrap, you can easily put UI components to your Application. To access Bootstrap click [here](#).

In the Documentation of Bootstrap, we will find the starter template which has two main things for adding bootstrap in our application.

1. Bootstrap bundle with popper: It contains the JQuery files, Bootstrap.js, and popper.js files. Copy the code and paste it inside the body tag of index.html of your react application.

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" in
```

2. Bootstrap CSS: It contains the CSS files to enhance your application. Copy the code of Bootstrap CSS and add it inside the head tag of index.html (In the Public folder) of your application.

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="sty
```

Now we can use the components of Bootstrap in our Application. Open App.js and Remove the Previous content from return(). Let's begin creating our application.

```
function app(){  
  return(  
  
  );  
};
```

Navbar of our app:

Get the Navbar component from the Bootstrap Component section and paste it inside the return(). Make sure of the following things while doing so:

- Close those tags which don't have a closing tag.
- Replace the 'class' keyword with 'ClassName'.
- Replace href= "#" with href= "/"
- The code must be in one tag or use a JSX fragment.

Editing the Bootstrap component:

You can override the Bootstrap CSS code and replace it with your desired one. You can even delete those components which aren't required.

Point to remember: Whenever you are building the app for production, you should use the *npm run build* command instead of *npm start* to host your application.

In the upcoming tutorials, we will be using React routers to surf different pages of our Application without reloading them.