**Objectives**

This session is all about getting started with **React** and understanding how modern web applications work using the **SPA (Single Page Application)** approach.

By the end of this hands-on lab, you’ll be able to:

* Understand what SPA is and why it’s useful.
* Learn what React is and how it works.
* See the difference between SPA and MPA.
* Explore the pros and cons of using SPA.
* Get a basic understanding of key concepts like virtual DOM and React features.
* Set up your own first React application and see it running.

**Conceptual Overview**

**What is SPA?**

SPA stands for **Single Page Application**. It’s a type of web app where all content is loaded dynamically without refreshing the entire page. Only parts of the page change, which makes the app feel fast and responsive.

**Why use SPA?**

* Faster user interactions
* Less data sent between client and server
* Seamless transitions between pages

**What is React?**

React is a popular JavaScript library developed by Facebook for building user interfaces, especially for SPAs. It helps developers create reusable UI components and efficiently update the UI when data changes.

React’s strength lies in its use of the **virtual DOM**, which is a lightweight copy of the real DOM. Instead of updating the entire page, React compares the virtual DOM with the real DOM and updates only what’s necessary.

**SPA vs. MPA**

| **Feature** | **SPA (Single Page App)** | **MPA (Multi Page App)** |
| --- | --- | --- |
| Page Loading | Loads once, updates dynamically | Reloads full page each time |
| Speed | Fast after initial load | Slower navigation |
| Server Interaction | Less frequent | Frequent full-page requests |
| SEO | More difficult to manage | Easier and traditional |
| Development | More frontend-heavy | Server-side rendering driven |

**Pros and Cons of SPA**

**Pros:**

* Smooth and responsive experience
* Faster navigation between views
* Works well as a mobile app base (using React Native)

**Cons:**

* Initial load can be large
* SEO is more complex
* Client-side errors may be harder to handle

**Virtual DOM in React**

The **Virtual DOM** is a concept where React creates a virtual copy of the real DOM. When a change occurs, React compares the virtual DOM with the real DOM (a process called *diffing*) and updates only the parts that changed. This improves performance and user experience.

**Key Features of React**

* **JSX:** JavaScript syntax extension that lets you write HTML inside JS.
* **Components:** Build UI using independent, reusable pieces.
* **Virtual DOM:** Efficient updates to the user interface.
* **One-Way Data Flow:** Data flows in a single direction, making debugging easier.
* **Hooks:** Functions that let you use state and lifecycle features in functional components.

**Hands-on Lab: Setting Up React**

Let’s build your very first React app and run it locally.

**Time Required**

Around 30 minutes

**Requirements**

Before you start, make sure the following are installed on your system:

* **Node.js** and **npm** – Download from <https://nodejs.org/en/download/>
* **Visual Studio Code** – Code editor for working with the project

**Step-by-step Instructions**

1. **Install Create React App**  
   Open your command prompt or terminal and run:

npm install -g create-react-app

1. **Create a New React Project**  
   Type the following command to create a new React app called myfirstreact:

npx create-react-app myfirstreact

1. **Move into the App Folder**

cd myfirstreact

1. **Open the Project in Visual Studio Code**

code .

1. **Update the App Component**  
   Go to the src folder and open App.js. Delete the existing content and replace it with:

import React from 'react';

function App() {

return (

<div>

<h1>Welcome to the first session of React</h1>

</div>

);

}

export default App;

1. **Run the React App**

npm start

1. **View It in the Browser**

After the app starts, open your browser and go to:

http://localhost:3000

You should see the message:  
**Welcome to the first session of React**