**1.React\_Hands\_On**

**1.Define SPA and its benefits**

**Definition: SPA in React**

SPA stands for Single Page Application.

In React, an SPA is a web application that loads a single HTML page and dynamically updates the UI without reloading the entire page when the user navigates between different views or routes.

Instead of fetching a new HTML page from the server on each click, React uses JavaScript (React Router) to change the view on the client side, making the app feel fast and fluid.

**Benefits of SPA in React**

**Fast User Experience**

Only necessary data is fetched and updated — no full page reloads.

Results in faster transitions and smoother navigation.

**Efficient Rendering with React**

React uses a Virtual DOM that updates only changed parts of the UI.

Boosts performance and responsiveness.

**Client-Side Routing**

React Router allows seamless navigation between components without hitting the server.

**Reduced Server Load**

Since only data (not full HTML pages) is requested, server traffic is lighter.

**Better Code Re-usability**

Components can be reused across different views, reducing duplication.

**Improved Developer Experience**

Hot reloading, reusable components, and modularity make it easier to build and maintain apps.

**2.Define React and identify its working**

**Definition: React**

React is a JavaScript library developed by Facebook for building user interfaces (UI), especially for Single Page Applications (SPAs).

It allows developers to build reusable UI components that efficiently update and render when data changes, without reloading the whole page.

**How React Works**

React’s core working is based on three main concepts:

**1. Component-Based Architecture**

React apps are made of small, self-contained components.

Each component manages its own state and renders UI based on that state.

**Eg:**

function Welcome(props) {

return <h1>Hello, {props.name}</h1>;

}

**2. Virtual DOM**

React maintains a lightweight Virtual DOM, a copy of the actual DOM in memory.

When state changes, React:

Creates a new Virtual DOM,Diffs it with the previous one (using a “reconciliation” process).Updates only the changed parts in the real DOM.This makes React fast and efficient.

**3. Unidirectional Data Flow**

Data flows from parent to child using props.State is held in top-level components and passed down, ensuring predictable behavior.

**Eg:**

<ChildComponent name="React" />

### ****SPA vs MPA: Comparison Table****

| **Feature** | **SPA (Single Page Application)** | **MPA (Multi Page Application)** |
| --- | --- | --- |
| **Definition** | Loads a single HTML page and dynamically updates content | Loads a new HTML page from the server for each user interaction |
| **Page Reloads** | No page reload (client-side routing) | Full page reload on every navigation |
| **Speed & Performance** | Faster after initial load | Slower due to repeated server requests |
| **Development Frameworks** | React, Angular, Vue | Traditional web tech (JSP, PHP, ASP.NET, etc.) |
| **Routing** | Handled on the client side (e.g., React Router) | Handled on the server side |
| **Initial Load Time** | Slightly higher (needs to load JS bundle) | Faster (just loads the current page) |
| **User Experience** | Smooth and app-like | Can feel slower and less interactive |
| **SEO Friendliness** | Harder to implement (requires SSR or pre-rendering) | Better by default, each page is SEO-friendly |
| **Server Load** | Lower (once loaded, fewer server hits) | Higher (every action hits the server) |
| **Use Case** | Dynamic platforms (e.g., Gmail, Facebook, Trello) | Content-heavy websites (e.g., blogs, news sites) |

1. **Pros of Single Page Application**

**Fast and Responsive UI**  
SPAs provide a faster user experience because once the page is loaded, only data is exchanged between the browser and the server, not the entire page. This makes the interface quick and responsive.

**Seamless User Experience**  
SPAs feel like native mobile apps. When users click on links or buttons, the content changes without reloading the whole page, which makes the experience smooth and fluid.

**Efficient Front-End Development**  
Since SPAs use reusable components in frameworks like React, the code is modular, easier to manage, and faster to develop and maintain.

**Reduced Server Load**  
Only data is requested from the server via APIs. There is no need to reload the entire HTML from the server every time, which reduces the load on the backend server.

**Easy to Convert into PWA**  
SPAs can be easily converted into Progressive Web Apps, which work offline and can be installed like mobile apps.

**Cons of Single Page Application**

**Poor SEO by Default**  
Because SPAs load content using JavaScript after the initial page load, search engines may not be able to read or index the content properly unless additional tools like server-side rendering are used.

**Long Initial Load Time**  
SPAs often have to load a large JavaScript bundle at the beginning. This can make the first-time loading slower, especially on slow internet connections.

**Browser History Issues**  
Managing the browser's back and forward buttons can be tricky in SPAs and needs to be handled using tools like React Router.

**More Client-Side Code**  
In SPAs, most logic happens in the browser. This can make the app heavier, and managing memory and performance becomes more complex.

**JavaScript Dependency**  
SPAs rely completely on JavaScript. If JavaScript is disabled in the browser, the app won't work at all.

**Explain about React**

React is a JavaScript library developed by Facebook for building user interfaces, especially for single-page applications. It allows developers to build reusable UI components that update efficiently when the data changes. React is focused only on the view layer of the application, which means it handles how the app looks and responds to user actions. It is widely used because of its speed, simplicity, and flexibility.

**Define Virtual DOM**

The Virtual DOM is a lightweight copy of the real DOM that React keeps in memory. When changes happen in the UI, React first updates the Virtual DOM instead of updating the real DOM directly. Then it compares the new Virtual DOM with the previous one, finds the differences, and only applies those changes to the real DOM. This process makes the UI updates faster and more efficient, improving overall performance.

**Explain Features of React**

React has several powerful features. One important feature is its component-based architecture, where the UI is broken into small, reusable parts called components. Another feature is JSX, a syntax that allows writing HTML elements directly inside JavaScript code, making it easier to design interfaces. React also uses one-way data binding, which means data flows in a single direction, making the code more predictable and easier to debug.