**1. Define SPA and its Benefits**

A **Single-Page Application (SPA)** is a web application or website that interacts with the user by dynamically rewriting the current page rather than loading entire new pages from the server. This approach avoids interruption of the user experience between successive pages.

**Benefits of SPA:**

* **Faster navigation:** Only the required content is updated, making the app quicker.
* **Improved user experience:** Seamless interaction similar to a desktop application.
* **Reduced server load:** Less data transferred due to fewer full page reloads.
* **Offline support:** SPAs can cache data and work offline or with poor internet.
* **Efficient development:** Frontend and backend can be developed separately.

**2. Define React and Identify Its Working**

**React** is an open-source JavaScript library developed by Facebook for building user interfaces, especially for SPAs.

**How React Works:**

* Uses a **virtual DOM** to efficiently update the real DOM.
* Components represent UI elements.
* When data (state/props) changes, React re-renders the virtual DOM.
* React compares the new virtual DOM with the old one (diffing algorithm).
* Only the changed elements are updated in the real DOM (reconciliation).

**3. Identify the Differences Between SPA and MPA**

|  |  |  |
| --- | --- | --- |
| Aspect | SPA (Single-Page Application) | MPA (Multi-Page Application) |
| Page Reload | No | Yes |
| Speed | Fast (after initial load) | Slower due to page reloads |
| User Experience | Seamless | Traditional |
| SEO Optimization | Difficult without SSR | Better SEO |
| Backend Dependency | Less | More |
| Examples | Gmail, Twitter | Amazon, Wikipedia |

**4. Explain Pros & Cons of Single-Page Applications**

**Pros:**

* Fast and responsive
* App-like experience
* Less server load
* Can work offline

**Cons:**

* SEO limitations
* Initial loading time can be longer
* Requires client-side routing
* Heavily dependent on JavaScript

**5. Explain About React**

React is a JavaScript library used for building fast, scalable, and simple user interfaces. It allows developers to create reusable UI components and manage the application's state efficiently.

**Key Concepts:**

* **JSX**: JavaScript + XML syntax
* **Components**: Building blocks of UI
* **State & Props**: Data management
* **Hooks**: Functional components with lifecycle features
* **React Router**: Navigation in SPAs

**6. Define Virtual DOM**

The **Virtual DOM** is a lightweight in-memory representation of the actual DOM. React uses it to optimize DOM manipulation:

* React creates a virtual DOM copy of the real DOM.
* When the state changes, it updates the virtual DOM.
* React then compares the new and old virtual DOMs.
* Only the differences are updated in the real DOM.

**7. Features of React**

* **JSX Syntax:** HTML-like syntax for writing UI
* **Component-Based:** Reusable, modular UI blocks
* **Virtual DOM:** Efficient UI updates
* **Unidirectional Data Flow:** Makes application logic predictable
* **Hooks:** Add state and lifecycle to functional components
* **React Router:** Routing in SPAs
* **Strong Community:** Vast ecosystem of tools and libraries