OBJECTIVES -

1. Define SPA (Single Page Application) and Its Benefits

A Single Page Application (SPA) is a web application that loads a single HTML page and dynamically updates the content as the user interacts with the app—without reloading the entire page.

Benefits of SPAs :

Fast navigation

improved user experience

reduced server load

Easier to develop APIs

2. Define React and Identify Its Working

React is a JavaScript library for building user interfaces, particularly for single-page applications. It is maintained by Meta (Facebook).

Workings of React :

It uses components to build UI elements

It implements virtual DOM to efficiently update the real DOM

Uses state and props to manage dynamic data

It re-renders only parts of the page that change, improving performance.

3. Differences between SPA and MPA

1. SPA Loads once, then updates dynamically

MPA Each interaction loads a new HTML page

2. SPA are Faster (after initial load)

MPA Slower (due to full-page reloads)

3. SPA Seamless, fluid

MPA Less fluid, more like traditional sites

4. SPA Harder to implement SEO

MPA Easier for SEO

5. SPA More complex front-end logic

MPA Easier with server-side rendering

4. Explain Pros & Cons of Single-Page Application

PROS :

1. Faster user experience after initial load

2. Efficient network usage (only fetches data)

3. Reusable UI components

4. Easier to turn into mobile apps with frameworks like React Native

Cons:

1. SEO can be challenging

2. First load time can be longer

3. Requires JavaScript to be enabled

4. May be less secure if not handled properly

5. Explain About React

React is a declarative, component-based library used for building dynamic and interactive UIs.

Key Concepts:

1. JSX: HTML + JS syntax to write components

Components: Small, reusable code blocks (either functional or class-based)

2. State & Props: Used to manage and pass data

3. Hooks: Functions like useState, useEffect to add features to functional components

6. Define Virtual DOM

The Virtual DOM (VDOM) is a lightweight copy of the real DOM maintained by React.

How It Works:

1. When state changes, React updates the VDOM.

2. It compares the new VDOM with the old one (diffing).

3. Only the differences are updated in the real DOM (reconciliation).

4. This makes updates faster and more efficient than direct DOM manipulation.

7. Features of React

Component-Based Architecture: Reusable UI pieces.

Virtual DOM: Boosts rendering performance.

JSX Syntax: Clean and readable way to write components.

Unidirectional Data Flow: Predictable and controlled state updates.

React Hooks: Manage state and side-effects in functional components.

React Native: Build mobile apps with the same principles.

Rich Ecosystem: Supported by tools like Redux, React Router, etc.

1. Explain React components

A React component is a reusable piece of UI that can be used to build complex interfaces by combining multiple components.

Every React app is made up of components, and each component defines how a part of the UI should look and behave.

2. Identify the differences between components and JavaScript functions

React Components are used to define UI elements. JavaScript Functions are used to perform logic or return values.

React Components return JSX while JavaScript Functions return primitive values, objects, etc.

React Components always follow react rules while JavaScript Functions use general JS function syntax.

React Components are used in rendering UI while JavaScript Functions are called with functionName().

React Components accept props for dynamic content and JavaScript Functions accept parameters.

3. Identify the types of components

Functional Components (Recommended for modern React)

Class Components (Used in older React versions)

4. Explain class component

A class component is a traditional way to define components in React using JavaScript ES6 classes.

Uses render() method to return JSX.

Can have state and lifecycle methods.

Inherits from React.Component.

5. Explain function component

A function component is a simple JavaScript function that returns JSX.

Simpler and cleaner.

Can use hooks (like useState, useEffect) for state and lifecycle.

Preferred in modern React.

6. Define component constructor

The constructor in a class component is a special method that gets called when the component is created.

Purpose:

Used to initialize state.

Used to bind methods.

7. Define render() function

The render() function is a mandatory method in class components that returns JSX to describe what the UI should look like.

It must return a single JSX element (can wrap multiple in a <div> or fragment).

React automatically calls this method to render the component on the screen.

1.Explain the need and Benefits of component life cycle

React components go through a lifecycle — creation, updating, and destruction. The lifecycle methods/hooks help you:

Control what happens and when in a component’s life.

Run code at specific points, like fetching data after rendering, or cleaning up resources before removal.

Keep the app efficient and predictable.

Benefits of Component life cycle –

Control over rendering — update UI based on state/props changes.

Efficient resource management — start/stop things like timers or API calls.

Debugging ease — know when and why a component re-renders or unmounts.

Modular logic — separate code for initialization, updates, and cleanup.

2.Identify various life cycle hook methods

constructor()

Initialize state and bind methods

componentDidMount()

Run after component is rendered (e.g., API calls)

shouldComponentUpdate()

Decide whether to re-render

componentDidUpdate()

Run after updates

componentWillUnmount()

Cleanup (e.g., timers, subscriptions)

Hooks:

useEffect()

Acts like componentDidMount, componentDidUpdate, and componentWillUnmount

useState()

Manage component’s internal state

useRef()

Access DOM or persist values across renders

3.List the sequence of steps in rendering a component

Component Lifecycle = 3 Phases:

1.Mounting → Setup (e.g., data fetch).

2.Updating → React to state/prop changes.

3.Unmounting → Cleanup

Outputs –



