

React Teorihandbok DEL I

1. Ramverket React - en översikt (en översikt, vad som är unikt för ramverket, fördelar, ev nackdelar, osv)

React är en Js bibliotek utvecklat av Facebook i syfte att underlätta utvecklandet av webbapplikationer genom ett renare informations arkitektur som dessutom presterar bättre då ramverket är mer optimerad i jämförelse med många andra alternativ på marknaden. Denna informations arkitektur är en av aspekterna som gör att man snabbare kan bygga olika komponenter och återanvända dem i senare skede - arbetsprocessen effektiviseras. En klar fördel med React är att produkter byggd i med hjälp av denna bibliotek är att produkten är lättare att skala men också möjliggör rendering av både enklare och mer avancerade tekniker såsom VR och 360-video. En nackdel som är värt att nämna är att React uppdateras snabbt och det är svårt att hålla sig uppdaterad kring dessa förändringar. Unikt för React är hur virtual DOM används, men mer om detta i nästa fråga...

2. Vad innebär Rendering och Virtual DOM?

Rendering innebär att göra om React-komponenter till DOM-element som sedan skriver ut en output på skärmen. När React-komponent renderas genereras en träliknande struktur av virtuella DOM-element som motsvarar de faktiska HTML-element som ska visas. Den virtuella DOM är en lättviktig kopia av den verkliga DOM och fungerar som en mellanhand mellan komponenter och webbläsarens renderingsmotor. Alltså är det just Virtual DOM som möjliggör att endast den delen som behöver att uppdateras, utan att påverka någon annan del. Detta koncept är anledningen till att React är så snabb i jämförelse med många andra ramverk.

Ett annat sätt att se på det är att DOM är en abstraktion av HTML, medan Virtual DOM är en vidare abstraktion av DOM som renderar endast delar av trädet där förändringen har skett istället för att rendera hela trädet

3. Vad är JSX? Vad används det till?

JSX är en Js syntax förlängning vilket gör att man kan lättare kan skapa ett element som i utseendet liknar HTML-kod. Man kan alltså använda sig av både Js och HTML i samma fil för att förenkla läsningen av användargränssnittet. JSX gör att koden är SEO-vänlig.

4. Vad är ett undantag inom programmering? Eng. Exception

Ett undantag används för att hänvisa till undatag eller fel som uppstår under körningen av en komponent. Det kan handla om ohanterad fel eller oväntat beteende som uppstår under renderingen. När ett undantag inträffar i en komponent, fångar React det och kan rendera en fallback-komponent eller visa ett felmeddelande, vilket hjälper att förhindra applikationer krascha helt. För att hantera undantag kan React-komponenter använda koden “componentDidCatch” eller liknande. Man kan också utveckla sätt som och metoder byggd på egna logik för att felhantera.

5. Vad innebär autentisering inom webbapplikationer? Vad används det till?

Autentisering inom webbapplikationer syftar på verifiering och godkännandet av användarens identitet och behörighet för att få åtkomst till webbapplikationen. Frågar den besvarar är “Hur bekräftar man att användaren är den de utger sig att vara?” Autentisering används för att upprätthålla säkerheten och skydda användarens känsliga information, i vissa fall även information på plattformen. Genom att implementera autentiseringssystem kan utvecklare se till att användaren kan få åtkomst till resurser och samtidigt att deras identitet förblir skyddad. Detta är en viktig fråga eftersom id-kapning är en större problematik som är svår att lösa.

Det finns olika sätt att inför autentisering inom webb. Något vi främst känner till i vardagen är att använda oss av andra konton såsom Google, Facebook, Twitter m.fl. Ett annat, något som anses vara mer säker är två stegs verifiering där man genom en annan enhet, vanligtvis telefonen får en kod som man knäpper in. Båda har sina för och nackdelar och frågan är mer aktuell än någonsin p.g.a. internet säkerheten blir svårare och svårare att sköta.

Del 2 - sammanfattning av projekt

Fråga 1 - Ramverket React - en översikt

Denna uppgift är ett perfekt tillfälle för att lära sig och börja bygga upp ett React bibliotek för att kunna använda komponenter i framtida projekt. React i sig är mer anpassad för större, mindre statiska projekt och är nog där man får se hela kraften av biblioteket, eftersom det är där man behöver snabbare och oftare implementera förändringar. Det blir intressant att se hur man kommer att använda React i framtiden när man arbetar i team. En av nackdelarna för ramverket är brant inlärningskurva, vilket är en av anledningarna till varför denna uppgift fyller en viktig funktion. I mitt fall är finns, har jag byggt bibliotek som heter “components” som är indelad i “hooks” och “pages” för att underlätta processen och läsningen för mig själv.

Fråga 2 - React DOM vs Virtual Dom

Exempel i kod:

```
function DropdownMenu({ options , nameButton }) {
  const [isOpen, setIsOpen] = useState(false);

  return (
    <div className="dropdown">
      <p className='text-primary-color text-[40px] mb-1 mt-5'>
        {nameButton} <button className='text-white text-[25px] mb-1 mt-5'
          onClick={()=>isOpen ? setIsOpen(false) : setIsOpen(true)} > <i
            className="fa-solid fa-chevron-down"></i></button></p>
      {isOpen && (
        <ul className="dropdown-menu transition opacity delay-300
          ease-in-out">
          {options.map((option) => (
            <li className='mb-2 text-[20px]' key={option.value}>
              {option.label}
            </li>
          ))}
        </ul>
      )}
    </div>
  );
}

export default DropdownMenu;
```

Processen i ovanstående kod ser ut som följande (när det kommer till DOM vs Virtual DOM):

Vid renderingen skapas ett Virtual DOM-träd baserad på JSX-koden inuti “return” och “DropdownMenu”. Denna DOM-trädet innehåller representation av de element som ska renderas på sidan - alltså det som är inom t.ex <div> , <button> och . React gemför det virtuella DOM-trädet med det befintliga DOM för att se om det finns skillnader. Om det finns skillnader, uppdaterar React den riktiga DOM.

Om jag skulle vilja ändra koden och t.ex lägga till ett element såsom eller , kommer React lägga till denna element i det riktiga DOM-strukturen direkt.

Fråga 3. JSX

Som nämnt ovan ser JSX ut till utseendet som HTML kod. Nedan är exempel från min kod.

```
<h1 className="text-[55px] text-primary-color font-semibold  
leading-[1.2]">Hi there, Welcome to my portfolio!</h1>  
  
    <p className="text-[18px] mt-5 mb-5 text-second-color  
font-semibold ">I am building digital experiences that drive convertible  
traffic.</p>
```

Som man kan se har jag använt mig av h1 och p-taggar. Att göra det på detta sätt gör det mycket enklare att läsa om man har mer text och fler typografiska nivåer. Som detta:

```
<h2 className="text-[35px] text-primary-color  
font-semibold">Project Overview</h2>
```