

Teorihandbok - Del 1

1. React är ett Javascript-baserat ramverk som används för att bygga användargränssnitt/frontend.

Fördelar

- Använder sig av en virtuell DOM istället för att direkt uppdatera DOM-element vid varje uppdatering, jämför React den virtuella DOM:en med den faktiska DOM:en och uppdaterar endast de nödvändiga delarna, vilket resulterar i bättre prestanda.
- I react så finns det möjlighet att dela upp gränssnittet i komponenter som tillåter att man enkelt kan återanvända, underhålla och dela komponenter.
- Det finns ett stort community med tredjepartsbibliotek tillgängligt.
- Roligt.

Nackdelar

- Kan upplevas svårt att lära sig JSX speciellt om man är inbiten Javascriptare. Detta gäller dock inte för oss som haft Kristian som lärare, då blir allt enkelt.
- Även om det finns många react-bibliotek så är det tyvärr inte alltid deras dokumentation hålls aktuell.
- På gott och ont så finns det mycket tillgängliga bibliotek vilket gör att man som React:are behöver hänga med i utvecklingen och se vad som är hett och inte. Detta kan uppfattas som ansträngande.

2. Virtuellt DOM och rendering

Rendering

Rendering sker när HTML, CSS och JavaScript bildar en visuell representation som användare kan interagera med.

Virtual DOM

Virtual DOM är en teknik som React använder för att optimera uppdateringar av användargränssnittet. Istället för att direkt ändra den faktiska DOM:en vid varje uppdatering, används en virtuell representation av DOM-trädet i minnet. Genom att jämföra skillnaderna mellan det nuvarande tillståndet och det tidigare tillståndet kan React applicera endast de nödvändiga ändringarna på den faktiska DOM:en, detta görs för att effektivisera. Alltså behöver endast en del av UI:t renderas precis som tidigare nämnt.

3. JSX

JSX (JavaScript XML) är en syntaxutökning för JavaScript som används i React. JSX gör det möjligt att skriva HTML-liknande kod direkt i JavaScript-filer.

4. Undantag

Inom programmering hänvisar ett undantag till en ovanlig händelse eller felaktigt beteende som uppstår under programmets körning och avviker från det förväntade flödet.

5. Autentisering

Autentisering innebär att man verifierar och bekräftar en användares identitet innan man tillåter användaren att ta del av eventuellt innehåll. Exempel på detta: Är Bankid och användarnamn och lösenord på mejl.

Teorihandbok - Del 2

Jag använder mig av React där jag skriver mina komponenter med hjälp av JSX. När detta görs så skapar React först en Virtual DOM-struktur i minnet. Efter att komponenten har renderats till Virtual DOM jämförs den med den faktiska DOM:en, och endast de nödvändiga ändringarna appliceras för att uppdatera det synliga användargränssnittet på webbsidan.

Ett exempel på detta är mina knappar. De alla har en visuell förändring när man hoverar på dem. React kommer att upptäcka denna händelse och reagera på den genom att uppdatera Virtual DOM för den aktuella knappen. React jämför sedan den uppdaterade Virtual DOM med den faktiska DOM:en som redan visas i webbläsaren. Genom att jämföra Virtual DOM med den icke-virtuella DOM:en kan React hitta de specifika ändringarna som behöver appliceras på knappen. Då renderar den om knappen istället för hela sidan.

När jag skriver en `return`-sats inuti en komponent i React, kallas det som finns där för JSX-element eller JSX-kod.

I mitt kodexempel har jag ett JSX-element som är skrivet inuti `return`-satsen:

```
<StyledSun as={motion.div}
  initial={{ opacity: 0, scale: 0.5 }}
  animate={{ opacity: 1, scale: 1 }}
  transition={{
    duration: 5.8,
    delay: 0.5,
    ease: [0, 0.71, 0.2, 1.01]
```

```
}}  
></StyledSun>
```

Här skapar jag ett JSX-element med namnet `StyledSun` och det renderas som en motion.div-komponent från framer-motion-paketet med olika attribut som jag har lagt till för att skapa min sol. Det som finns inuti `return`-satsen JSX-element. Detta JSX-element beskriver strukturen och egenskaperna för det element som ska renderas i det synliga användargränssnittet.