



React Context

Faggruppemøte 26. april 2022


Av Robert Kittilsen og Anna Li



- Anna Li
- Oslo/Egersund
- Fysikk fra UiO
- Medisinsk
fysiker og
utvikler

- Robert Kittilsen
- Siljan
- Høyskolen
Kristiania
- Programmering
m/ fordypning i
mobilutvikling



- 
- Hva er React Context?
 - Når bruke det?
 - Hvilke problemer løser det?
 - Hvordan bruke det?
 - Eksempler



Hva er React Context?

- Med React context kan man sende data til komponenter langt nedi komponent-treet uten å måtte sende det via alle komponenter mellom
- Brukes typisk til å håndtere global data som tema, brukerinnstillinger o.l



Når bruke det?

- Brukes typisk til å sende data som kan brukes i alle komponenter i appen.
- Eksempler
 - Tema
 - Brukerdata
 - Lokasjonsspesifikk data (språk og klokkeslett)
- Burde være nokså statisk data



Hvilke problemer løser det?

- Unngår “prop drilling”
- Global state

Hvordan bruke context?



Lage context

1. Lage et kontekst-objekt ved bruk av createContext-metoden fra Context APllet
2. Lag en provider og wrap komponentreet med provideren
3. Send verdi i context providere ved bruk av value-proppen.

```
export const ThemeContext = React.createContext(null);

export const App = () => {
  return (
    <ThemeContext.Provider value="darkTheme">
      <User />
    </ThemeContext.Provider>
  );
};
```

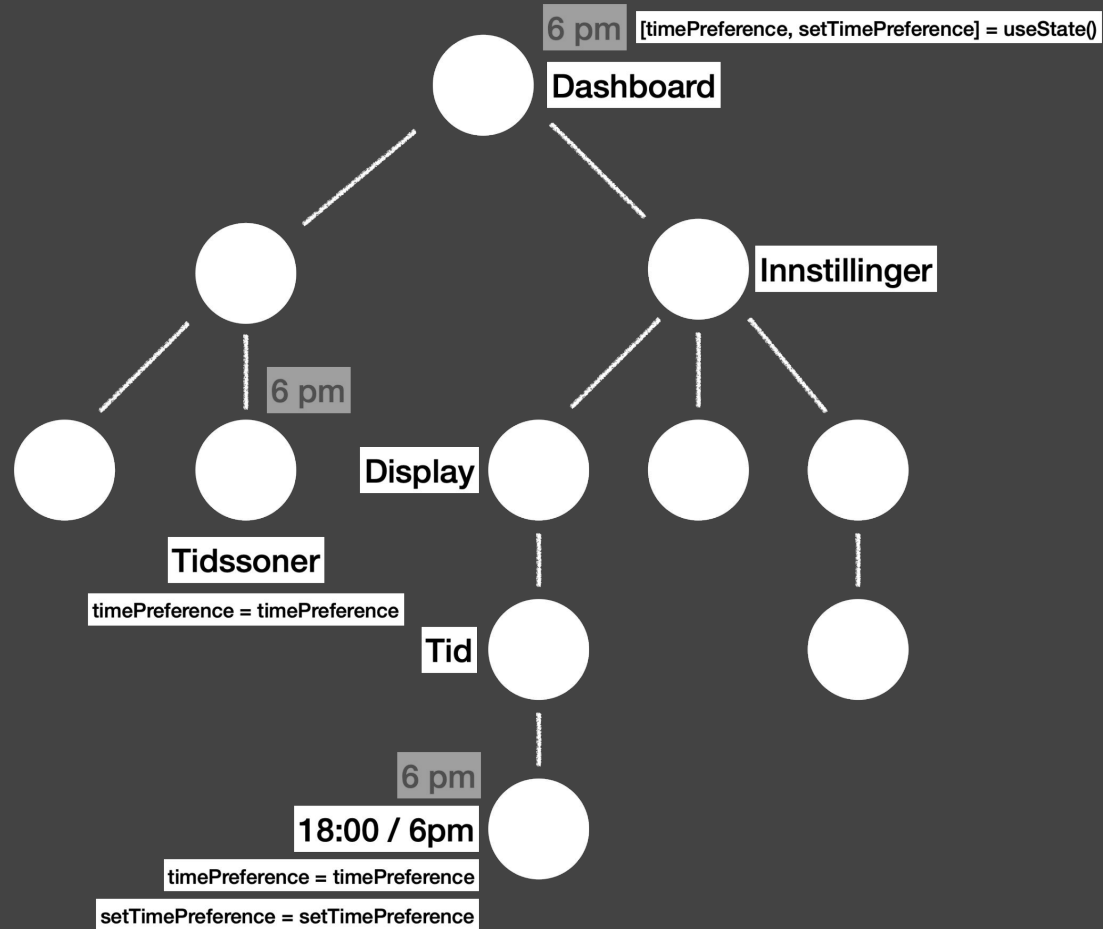


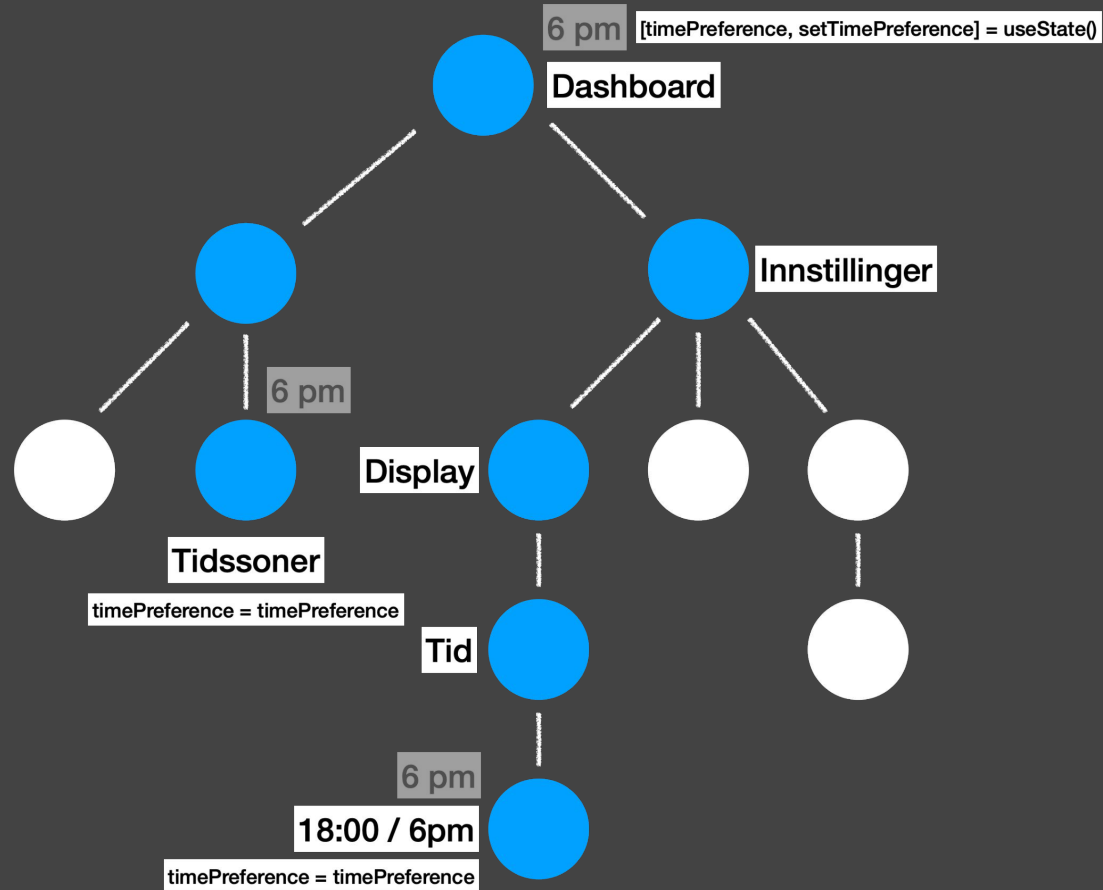

Bruke context

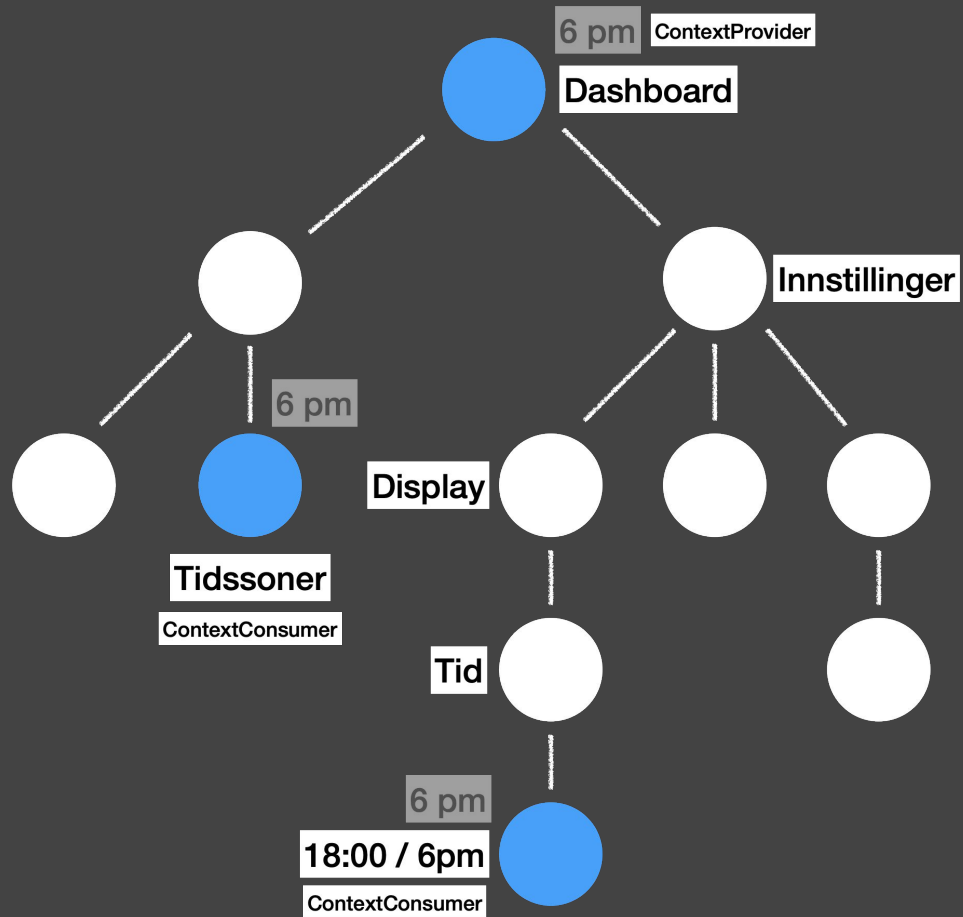
1. Importer useContext-hooken fra React for å hente ut verdi
2. Hent ut verdiene med å sende inn context-objektet inn i hooken.
3. Bruk verdiene

```
import { useContext } from "react";
import { ThemeContext } from "../App";

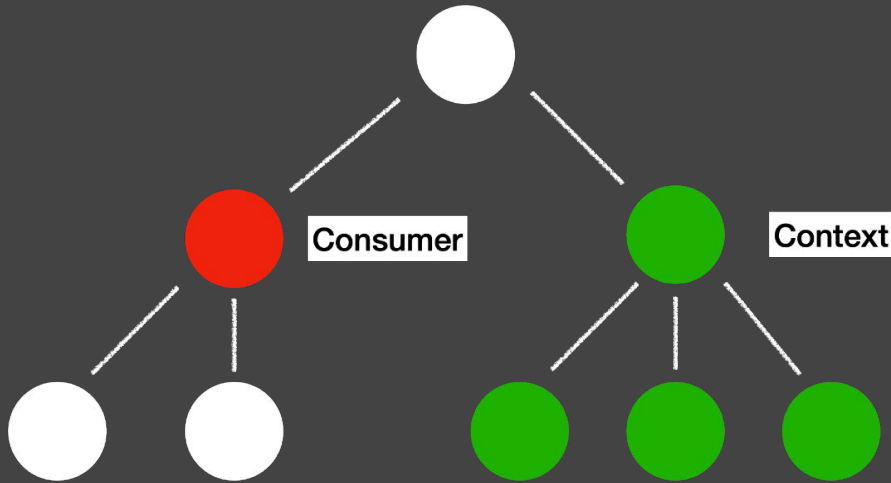
export const User = () => {
  const value = useContext(ThemeContext);
  return <h1>{value}</h1>;
};
```







Hva skjer hvis man bruker konteksten utenfor konteksten?



```
export const ThemeContext = React.createContext("Hello");

function UtenforKontekst() {
  const value = useContext(ThemeContext);
  console.log(value); // undefined
}
```

Eksempler

Eksempel 1

```
1 import { useState, createContext, useContext } from "react";
2 import "./styles.css";
3
4 const ThemeContext = createContext(null);
5
6 export function Eksempel1() {
7   const [theme, setTheme] = useState("light");
8
9   return (
10     <ThemeContext.Provider value={theme}>
11       <div className={`full-size theme-bg--${theme}`}>
12         <Text />
13         <label>
14           <input
15             type="checkbox"
16             onChange={(e) => {
17               setTheme(e.target.checked ? "dark" : "light");
18             }}
19           />
20           <span className={`theme-text--${theme}`}>Use dark mode</span>
21         </label>
22       </div>
23     </ThemeContext.Provider>
24   );
25 }
26
27 function Text() {
28   const theme = useContext(ThemeContext);
29
30   return <p className={`theme-text--${theme}`}>Noe tekst</p>;
31 }
```

Noe tekst

☐ Use dark mode

Eksempel 2a: Sende state med props

```
1 import { useState } from "react";
2 import "./styles.css";
3
4 export function Eksempel2a() {
5   const [theme, setTheme] = useState("light");
6
7   return (
8     <div className={`full-size theme-bg--${theme}`}>
9       <Section theme={theme} />
10      <label>
11        <input
12          type="checkbox"
13          onChange={(e) => {
14            setTheme(e.target.checked ? "dark" : "light");
15          }}
16        />
17        <span className={`theme-text--${theme}`}>Use dark mode</span>
18      </label>
19    </div>
20  );
21 }
```

```
1 function Section({ theme }) {
2   return (
3     <section>
4       <Card theme={theme} />
5     </section>
6   );
7 }
8
9 function Card({ theme }) {
10   return (
11     <div>
12       <Text theme={theme} />
13     </div>
14   );
15 }
16
17 function Text({ theme }) {
18   return <p className={`theme-text--${theme}`}>Noe tekst</p>;
19 }
```


Eksempel 2b: Sende state med context

```
1 import { useState, createContext, useContext } from "react";
2 import "./styles.css";
3
4 const ThemeContext = createContext(null);
5
6 export function Eksempel2b() {
7   const [theme, setTheme] = useState("light");
8
9   return (
10     <ThemeContext.Provider value={theme}>
11       <div className={`full-size theme-bg--${theme}`}>
12         <Section />
13         <label>
14           <input
15             type="checkbox"
16             onChange={(e) => {
17               setTheme(e.target.checked ? "dark" : "light");
18             }}
19           />
20           <span className={`theme-text--${theme}`}>Use dark mode</span>
21         </label>
22       </div>
23     </ThemeContext.Provider>
24   );
25 }
```

```
1 function Section() {
2   return (
3     <section>
4       <Card />
5     </section>
6   );
7 }
8
9 function Card() {
10   return (
11     <div>
12       <Text />
13     </div>
14   );
15 }
16
17 function Text() {
18   const theme = useContext(ThemeContext);
19
20   return <p className={`theme-text--${theme}`}>Noe tekst</p>;
21 }
22
```

Eksempel 3: Wrappe i hook



```
1 import { useState, createContext, useContext } from "react";
2
3 const LoginContext = createContext(null);
4
5 export const useLoginContext = () => {
6   const context = useContext(LoginContext);
7
8   if (!context) {
9     throw new Error("Du er utenfor Login-kontekst!");
10  }
11
12  return context;
13 };
14
```



```
1 export const SettingsProvider = ({ children }) => {
2   const [status, setStatus] = useState(false);
3
4   return (
5     <LoginContext.Provider
6       value={{
7         status: status,
8         setStatus: () => setStatus(!status),
9       }}
10     >
11       {children}
12     </LoginContext.Provider>
13   );
14 };
15
```

Eksempel 3: Bruke hookene

```
1 export const Eksempel3 = () => {
2   return (
3     <LoginProvider>
4       <LoginButton />
5     </LoginProvider>
6   );
7 };
8
9 export const LoginButton = () => {
10   const { status, setStatus } = useLoginContext();
11   return (
12     <div>
13       <p>Du er nå logget {status ? "inn" : "ut"} </p>
14       <button onClick={setStatus}>Logg inn/ut</button>
15     </div>
16   );
17 };
18
```