



# fi'gico

CAMILLA ERIKSSON, MARKUS KARLSSON, LISA BENGTSSON





# PROCESS

- Välja designsystem
- Välja produkt
- Grundläggande kod-struktur
- Tilldela issues
- "Ren" design

---

# GROMMET



SÅG TILLTALANDE UT



VERKADE  
ENKELT/SIMPELT



ANVÄNDES AV MÅNGA  
STORA FÖRETAG



DOKUMENTATIONEN  
KÄNDES ENKEL OCH  
"REN"

# NACKDELAR

- Dålig dokumentation
- Svårt att hitta hjälp/info
- Vissa komponenter svårare än andra
- Saknat vissa styling-attribut
- Problem med tex video samt bild i tabell
- Responsivitet

# FÖRDELAR

- Enkelt att få ihop en första layout
- Ibland skönt att dokumentationen är enkel
- Klara och tydliga komponenter

---

# SVÅRIGHETER MED PROJEKTET

- Svårt projekt
- Tid
- Jobbigt att arbeta helt på distans
- Lösa problem på nya sätt
- Knyta ihop all funktionalitet

# GROMMET UPPLÄGG

```
import React from "react";
import { Grommet } from "grommet";
import { grommet } from "grommet/themes";
```

```
function App() {
  return (
    <BrowserRouter basename="/">
      <CartProvider>
        <div className="App">
          <Grommet theme={grommet}>
            <ScrollToTop />
            <Header1 />
            <Switch>
              <Route
                path="/productpage/:productIndex"
                component={ProductPage}
              />
              <Route path="/checkoutpage/" component={CheckoutPage}></Route>
            </Switch>
          </div>
        </CartProvider>
      </BrowserRouter>
    )
}
```

- `npm install grommet grommet-icons styled-components -save`
- Importera Grommet
- Importera komponenterna
- Wrapa allt i `app.tsx` i Grommet

Importera grommet-komponenter där de ska användas

```
import {
  Box,
  Text,
  Table,
  TableHeader,
  TableRow,
  TableCell,
  TableBody,
  ResponsiveContext
} from "grommet";
import { Cart, FormAdd, FormSubtract, FormTrash } from "grommet-icons";
```

Så här används komponenterna

```
render() {
  return (
    <CartConsumer>
      ({ addProductToCart }) => (
        <div>
          <Box pad="medium" gap="medium" width="medium">
            <Text>{this.props.product.title}</Text>
            <Link to="/">
              <Close />
            </Link>
          </Box>
          <Text size={"20pt"}>{this.props.product.price}</Text>
          <Text>Size: 70x100 </Text>
          <Button
            primary
            color="dark-1"
            label="Add to cart"
            onClick={() => {
              this.handleClick(addProductToCart);
            }}
          ></Button>
          <Text size={"12pt"}>PRODUCT INFO</Text>
          <Text size={"10pt"}>{this.props.product.description}</Text>
        </Box>
      </div>
    </CartConsumer>
  );
}
```

CartContext

```
import React, { createContext } from "react";
import { Product } from "../Components/AllProducts";

const defaultState: State = {
  items: [],
  addProductToCart: () => {},
  deleteProductFromCart: () => {},
  getTotalPrice: () => 0,
  getTotalQuantity: () => 0,
  removeOneProduct: () => {},
  deleteAllProductsFromCart: () => {}
};

const CartContext = createContext<State>(defaultState);

Camilla Eriksson, 8 days ago | 2 authors (Camilla Eriksson and others)
interface CartItem {
  product: Product;
  quantity: number;
}
```



DEMO &  
GENOMGÅNG  
AV SIDAN

