

Web Tabanlı Programlama Final Ödevi Projesi (E–Yemek Siparişı)

Burak MERCAN

1191602020

Proje Özeti

Kullanıcıya;

- 1) Restoran menüsünden dilediği yemek çeşidini ve yemekleri tercih etme
- 2) Tercih ettiği yemekleri sepetine ekleme, sepetinden çıkarma, miktarını arttırıp azaltma
- 3) Sepetinin toplam tutarını görme
- 4) Bilgilerini girdikten sonra sepetini satın alma ve satın alım bilgisini görme

imkanları sağlar.

Header Kısım



Burak Mercan

```
import React from 'react';
import {} from "@mui/material";
import {ShoppingBagRounded } from '@mui/icons-material';
import { useStateValue } from "../StateProvider";

function Header() {
  const [{ bag }] = useStateValue();

  return (
    <header>
      <img src='https://i.hizliresim.com/33qbxdt.png' alt='' className='logo' />

      <div className="shoppingBag">
        <ShoppingBagRounded className="bag" />
        <div className={` ${!bag ? "noBagItem" : "bag_content"} `}>
          <p>{bag ? bag.length : "0"} </p>
        </div>
      </div>

      <div className='profile'>
        <h3 className='name'> Burak Mercan </h3>
      </div>
    </header>
  );
}

export default Header;
```

Sepete ürün eklendiğinde veya
sepetten ürün çıkarıldığında değeri artıp
azalan sepet logosu

```
<div className={` ${!bag ? "noBagItem" : "bag_content"} `}>
  <p>{bag ? bag.length : "0"} </p>
</div>
```

Header.js Bileşeni

MainContainer (Row Container) Kısmı



İstenilen yemek çeşidi seçilirken arka plan rengi değişir ve seçilince ilgili çeşitteki yemekler dishItemContainer'a gelir.

MainContainer (Row Container) Kısmı

```
<div className="rowContainer">
  {MenuItems &&
    MenuItem.map((data) => (
      <div key={data.id} onClick={() => setData(data.itemId)}>
        <FoodCard
          imgSrc={data.imgSrc}
          name={data.name}
          isActive={data.id === 1 ? true : false}
        />{ " " }
      </div>
    ))}
</div>
```

App.js bileşenindeki rowcontainer bölgesi.
Bu bölgede map() fonksiyonu ile Data.js bileşeninden yemek çeşitleri listemize ulaştık.

FoodCard.js bileşenimiz yemek çeşitlerini gösteren kartlardır. Aktif olan yemek çeşidinin Data.js'teki bilgilerini göstermeye yarar.

```
import { ChevronRightRounded, } from '@mui/icons-material'
import React from 'react'

function FoodCard({imgSrc,name,isActive}) {
  return (
    <div className={`rowFoodCard ${isActive ? `active` : ''}`}>
      <div className='imgBox'>
        <img src={imgSrc} alt="" />
      </div>
      <h3 className='h3'>{name}</h3>
      <i className='menuLoad'>
        <ChevronRightRounded/>
      </i>
    </div>
  )
}

export default FoodCard;
```

MainContainer (Row Container) Kısmı

```
const [isMainData, setMainData] = useState(  
  Items.filter((element) => element.itemId === "salata1")  
);
```

Uygulama çalıştırıldığı ilk anda salata
FoodCard'ını ve altındaki veriyi
gösterecektir.

Yemek çeşidi kartlarımızı aktif eden
fonksiyon tanımı ve query selector
ile FoodCard'larımızı seçme kısmı.

```
useEffect(() => {  
  const menuCards = document  
    .querySelector(".rowContainer")  
    .querySelectorAll("rowFoodCard");  
  
  function setFoodCardActive() {  
    menuCards.forEach((n) => n.classList.remove("active"));  
    this.classList.add("active");  
  }  
  
  menuCards.forEach((n) => n.addEventListener("click", setFoodCardActive));  
}, [isMainData, bag, total, totalPrice]);
```


MainContainer (dishItemContainer) Kısmı



Seçtiğimiz çeşidin altındaki yemek listesini görebileceğimiz kısımdır. +'ya tıklanarak istenilen yemek sepete eklenebilir.

MainContainer (dishItemContainer) Kısmı

```
<div className="dishItemContainer">
  {isMainData &&
    isMainData.map((data) => (
      <FoodItemCard
        addToBag={() => addToBag(data.id)}
        key={data.id}
        id={data.id}
        imgSrc={data.imgSrc}
        name={data.name}
        price={data.price}
      />
    ))}
</div>
```

rowContainer'da olduğu gibi dishItemContainer'da da map fonksiyonunu kullandık.

Sepete ekleme

```
const addToBag = (id) => {
  let tmpBeg = [...bag];

  const newItem = Items.find((n) => n.id === id);
  let itemWithQty;
  const existing = bag.find((x) => x.item.id === id);
  if (existing === undefined) {
    itemWithQty = { qty: 1, item: newItem };
    tmpBeg.push(itemWithQty);
  } else {
    itemWithQty = { ...existing, qty: existing.qty + 1 };
    tmpBeg = tmpBeg.map((x) => {
      if (x.item.id === id) {
        return itemWithQty;
      }
      return x;
    });
  }
  dispatch({
    type: actionTypes.SET_BAG,
    bag: tmpBeg,
  });
};
```


MainContainer (dishItemContainer) Kısmı

```
import { AddRounded } from "@mui/icons-material";
import React from "react";
import { useStateValue } from "../StateProvider";

function FoodItemCard({ addToBag, imgSrc, name, price }) {
  const [{}] = useStateValue();

  return (
    <div className="ItemCard">
      <div className="imgBox">
        <img src={imgSrc} className="ItemImg" />
      </div>
      <div className="itemContent">
        <h3 className="itemName">{name}</h3>
        <div className="bottom">
          <h3 className="price">
            <span>₺</span>
            {price}
          </h3>
          <i
            className="addToBag"
            onClick={() => {
              addToBag();
            }}
          >
            <AddRounded />
          </i>
        </div>
      </div>
    </div>
  );
}

export default FoodItemCard;
```

FoodItemCard.js bileşenimiz.

BuyingMenu(BagCheckOut)Kismi

Boş Sepet



Dolu Sepet

Siparişiniz:



Falafel Salatası

x3

−

+

₺105



Farruj Meşvi

x2

−

+

₺100



Tahinli Dondurma

x1

−

+

₺30



Reyhan Limonata

x1

−

+

₺10

BuyingMenu (BagCheckOut) Kısmı

Boş Sepet (AddSomeItem)

```
{bag.length == 0 ? (  
  <div className="addSomeItem">  
    <h2 className="emptyBagh2">Sepetiniz Boş</h2>  
      
  </div>  
)
```

Dolu Sepet (BagCheckOut)

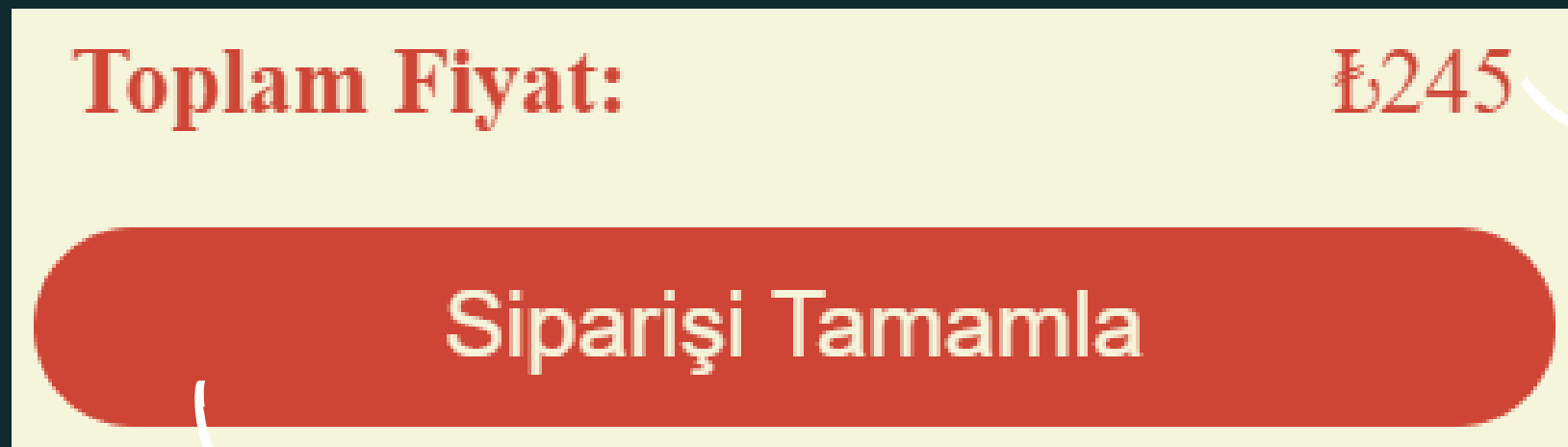
```
: (  
  <div className="bagCheckOut">  
    <div className="bagContainer">  
      <SubMenuContainer name={"Siparişiniz:"} />  
      <div className="itemsBag">  
        {bag &&  
          bag.map((data) => (  
            <BagItem  
              initialQty={data.qty}  
              name={data.item.name}  
              imgSrc={data.item.imgSrc}  
              qty={data.item.qty}  
              price={data.item.price}  
              id={data.item.id}  
              key={Math.random()}  
            />  
          ))}  
      </div>  
    </div>  
  )
```

BuyingMenu (BagItem) Kısmı

```
let bagItems = [];  
  
function BagItem({ id, name, imgSrc, price, initialQty }) {  
  const [qty, setQty] = useState(initialQty);  
  const [itemPrice, setItemPrice] = useState(parseInt(qty) * parseFloat(price));  
  const [{ bag }, dispatch] = useStateValue();  
  
  useEffect(() => {  
    bagItems = bag;  
    setItemPrice(parseInt(qty) * parseFloat(price));  
  }, [qty]);  
  
  const updateQty = (action, id) => {  
    if (action === "add") {  
      addItem(1);  
    } else {  
      if (qty === 1) {  
        let tempBag = bagItems.filter((x) => x.item.id !== id);  
        dispatch({  
          type: actionTypes.SET_BAG,  
          bag: tempBag,  
        });  
      } else {  
        addItem(-1);  
      }  
    }  
  };  
};
```

```
const addItem = (val) => {  
  let tmpBeg = [...bag];  
  let itemWithQty;  
  const existing = bag.find((x) => x.item.id === id);  
  itemWithQty = { ...existing, qty: existing.qty + val };  
  tmpBeg = tmpBeg.map((x) => {  
    if (x.item.id === id) {  
      return itemWithQty;  
    }  
    return x;  
  });  
  setQty(itemWithQty.qty);  
  dispatch({  
    type: actionTypes.SET_BAG,  
    bag: tmpBeg,  
  });  
};
```

BuyingMenu (Tamamlama) Kısmı



```
<button className="tamamla" onClick={() => purchase()}>
  Siparişi Tamamla
</button>
```

```
<div className="toplamFiyat">
  <h3>Toplam Fiyat:</h3>
  <p>
    <span>₺</span>
    {bag.reduce(
      (partialSum, a) =>
        partialSum + a.qty * Number(a.item.price),
      0
    )}
  </p>
</div>
```

BuyingMenu (Purchase ve User) Kısmı

Kullanıcı Bilgileri

Kullanıcı Bilgileri Arayüzü

```
<h2 className="giris">Kullanıcı Bilgileri</h2>
<input
  type={"text"}
  className="adiniz"
  placeholder=" Ad"
  value={ad}
  onChange={(e) => setAd(e.target.value)}
></input>
<input
  type={"text"}
  className="soyadiniz"
  placeholder=" Soyad"
  value={soyad}
  onChange={(e) => setSoyad(e.target.value)}
></input>
<input
  type={"text"}
  className="email"
  placeholder=" E-Posta"
  value={eposta}
  onChange={(e) => setEposta(e.target.value)}
></input>
<input
  type={"text"}
  className="adres"
  placeholder=" Adres"
  value={adres}
  onChange={(e) => setAdres(e.target.value)}
></input>
```

Kullanıcı bilgileri
aldığımız input alanları

```
const purchase = () => {
  let orders = "";
  let user = "";
  user += "Sayın " + ad + " " + soyad + " siparişiniz alınmıştır.\n"
    + adres + " adresine gönderilecektir.\n" +
    "Sipariş bilginiz "+ eposta+" adresine gönderilmiştir.";
  bag.forEach((x) => {
    orders += x.item.name + ":  " + x.qty + " x " + x.item.price + "₺\n";
  });
  alert(user);
  alert(orders);
};
```

Alert ile satın alım
bilgilerini gösterme

Sipariş Tamamlama Bilgisi

localhost:3000 web sitesinin mesajı

Sayın Burak Mercan siparişiniz alınmıştır.
KYK Sultan Çelebi Mehmet Yurdu adresine gönderilecektir.
Sipariş bilginiz burakmercan03@gmail.com adresine gönderilmiştir.

Tamam

alert(user);

localhost:3000 web sitesinin mesajı

Falafel Salatası: 3 x 35₺
Farruj Meşvi: 2 x 50₺
Tahinli Dondurma: 1 x 30₺
Reyhan Limonata: 1 x 10₺

Tamam

alert(orders);



Teşekkürler