

UNIVERSIDAD DON BOSCO



DOCENTE: ALEXANDER ALBERTO SINGUENZA CAMPOS

ALUMNO:

**JOSÉ ROBERTO DORADEA HERRERA
MARCELO YASER ANDRADE GARZA**

**DH211056
AG210653**

**MATERIA: DISEÑO Y PROGRAMACIÓN DE SOFTWARE
MULTIPLATAFORMA.**

TALLER 2: SEGUNDO DESAFÍO PRACTICO

1. App.js: Este archivo nos permite mostrar nuestra aplicación, también, nos ayuda a gestionar las diferentes pantallas.

```
import React from 'react';
import { StyleSheet, Platform } from 'react-native';
import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';
import { NavigationContainer } from '@react-navigation/native';
import Ionicons from 'react-native-vector-icons/Ionicons';
import Platillo from '../components/Screens/Platillo';
import Bebida from '../components/Screens/Bebida';
import CarritoContext from '../components/Screens/CarritoContext';
import {CarritoProvider} from '../components/Screens/CarritoContext';
import Carrito from '../components/Screens/Carrito'

export default function App() {
  const Tab = createBottomTabNavigator();

  const TabNavigator = () => (
    <Tab.Navigator
      screenOptions={({ route }) => ({
        tabBarIcon: ({ focused, color, size }) => {
          let iconName;
          if (route.name === 'Menú') {
            iconName = Platform.OS === 'ios' ? 'restaurant-outline' : 'restaurant';
          } else if (route.name === 'Bebidas') {
            iconName = Platform.OS === 'ios' ? 'beer-outline' : 'beer';
          }
          else if (route.name === 'Carrito') {
            iconName = Platform.OS === 'ios' ? 'cart-outline' : 'cart';
          }

          return <Ionicons name={iconName} size={size} color={color} />;
        },
        tabBarStyle: {
          backgroundColor: 'black',
        },
        tabBarLabelStyle: {
          color: 'white',
        },
        tabBarActiveTintColor: 'tomato',
        tabBarInactiveTintColor: 'gray',
      })}
    >
  )
}
```

2. ListaPlatillo.js: Nos permite leer la API y poder mostrar la lista de productos que se ofrecen:

```

import React, { useState, useEffect } from 'react';
import {
  View,
  Image,
  Text,
  TouchableOpacity,
  FlatList,
  StyleSheet,
  Dimensions,
} from 'react-native';

const ListaPlatillos = ({navigation}) => {
  const [food, setfood] = useState([]);
  useEffect( () => {
    fetchFood();
  }, []);
  const fetchFood = async () => {
    try{
      const response = await fetch('https://66e3588b494df9a478e4fd61.mockapi.io/api/platillos/food');
      const data = await response.json();
      setfood(data);
    } catch(error){
      console.error('Error fetching countries', error);
    }
  };
  const renderItem = ({item}) => (
    <TouchableOpacity
      onPress={() => navigation.navigate('descripcion', { info: item })}
    >
    <View style={styles.Lista}>
      <Image source={{ uri: item.imagen }} style={styles.ImagenF} />
      <View style={styles.ListaText}>
        <Text style={styles.TextNombre}>{item.nombre}</Text>
        <Text style={styles.TextPrecio}>{item.precio} $</Text>
      </View>
    </View>
  </TouchableOpacity>
  );
  return (
    <View style={styles.container}>

```

3. DetallePlatillo: Este archivo nos permite ver los detalles de cada producto y nos da la opción de poder agregar al carrito un producto o solo regresar a ver el menú.

```

import React, {useContext} from 'react';
import {View, StyleSheet, Text, Button, Image} from 'react-native';
import {CarritoContext} from './CarritoContext';

const DetallePlatillo = ({ route, navigation}) => {
  const {info} = route.params;
  const {agregarCarrito} = useContext(CarritoContext);
  return (
    <View style={styles.container}>
      <Text style={styles.heading}>{info.nombre}</Text>
      <Image source={{uri: info.imagen}} style={styles.imagen} />
      <Text style={styles.infoTitle}>{info.precio} $</Text>
      <View style={styles.infocontainer}>
        <Text style={styles.info}>{info.descripcion}</Text>
      </View>
      <View style={styles.buttons}>
        <Button title="Volver" onPress={() => navigation.goBack()} />
        <Button title="Agregar" onPress={() => agregarCarrito(info)} />
      </View>
    </View>
  )
}

```

4. ContextCarrito: Este archivo nos ayuda para poder ordenar los productos y poder contar que cantidad ha agregado el cliente de cada cosa. Nos permite agregar, actualizar cantidad y eliminar los datos del carrito.

```
import React, { createContext, useState } from 'react';

export const CarritoContext = createContext();

export const CarritoProvider = ({ children }) => {
  const [carrito, setCarrito] = useState([]);

  const agregarCarrito = (producto) => {
    const productoExistente = carrito.find((item) => item.nombre === producto.nombre);
    if (productoExistente) {
      setCarrito(
        carrito.map((item) =>
          item.nombre === producto.nombre
            ? { ...item, cantidad: item.cantidad + 1 }
            : item
        )
      );
    } else {
      setCarrito([...carrito, { ...producto, cantidad: 1 }]);
    }
  };

  const eliminarProducto = (nombreProducto) => {
    setCarrito(carrito.filter((producto) => producto.nombre !== nombreProducto));
  };

  const actualizarCantidad = (nombreProducto, nuevaCantidad) => {
    setCarrito(
      carrito.map((producto) =>
        producto.nombre === nombreProducto ? { ...producto, cantidad: nuevaCantidad } : producto
      )
    );
  };

  return (
    <CarritoContext.Provider value={{ carrito, agregarCarrito, eliminarProducto, actualizarCantidad }}>
      {children}
    </CarritoContext.Provider>
  );
};
```

5. Carrito: Esta screen nos permite ver la lista de productos seleccionados, además nos presenta los botones que harán que podamos modificar las cantidades de los productos y eliminarlos si necesitamos.

```

import React, { useContext } from 'react';
import { View, Text, FlatList, StyleSheet, Image, TouchableOpacity } from 'react-native';
import { CarritoContext } from './CarritoContext';

const Carrito = () => {
  const { carrito, eliminarProducto, actualizarCantidad } = useContext(CarritoContext);

  const handleQuantityChange = (item, change) => {
    const newQuantity = item.cantidad + change;
    if (newQuantity > 0) {
      actualizarCantidad(item.nombre, newQuantity);
    }
  };

  const handleRemoveItem = (item) => {
    eliminarProducto(item.nombre);
  };

  // Calcular el total sumando precios y cantidades
  const total = carrito.reduce((sum, item) => {
    return sum + item.precio * item.cantidad;
  }, 0);

  return (
    <View style={styles.container}>
      {carrito.length === 0 ? (
        <Text style={styles.emptyMessage}>El carrito está vacío</Text>
      ) : (
        <>
          <FlatList
            data={carrito}
            keyExtractor={({ item }) => item.nombre}
            renderItem={({ item }) => (
              <View style={styles.infocontainer}>
                <Image source={{ uri: item.imagen }} style={styles.imagen} />
                <View style={styles.textContainer}>
                  <Text style={styles.infoTitle}>{item.nombre}</Text>
                  <Text style={styles.info}>{item.precio} $</Text>
                  <View style={styles.quantityContainer}>
                    <TouchableOpacity
                      style={styles.button}
                      onPress={() => handleQuantityChange(item, -1)}
                    />
                  </View>
                </View>
              </View>
            )}
          />
        </>
      )}
    </View>
  );
};

```

6. Funcionamiento:

11:08

Menú



Tacos al
Pastor
3 \$



Torta de Res
5 \$



Nachos de
Res
7.5 \$

Chilaquiles
4 \$



Sopa de
Tortilla
5.5 \$



Menú



Bebidas



Carrito

11:08

Bebidas

Agua de Horchata



4.75 \$

Bebida refrescante hecha a base de arroz, canela y azúcar.

[Volver](#)

[Agregar](#)



Menú



Bebidas



Carrito

