



การพัฒนาโปรแกรมประยุกต์สำหรับอุปกรณ์เคลื่อนที่  
เรื่อง Navigation และ Routing ในแอปโมบายด้วย React Native

จัดทำโดย  
นายปฏิวัฒน์ กฤษฏีสกุลรัตน์ 031

เสนอ  
ผศ. เสถียร จันท์ปลา  
รหัสวิชา CSD3201  
ภาคเรียนที่ 2

คณะวิทยาศาสตร์และเทคโนโลยี  
มหาวิทยาลัยราชภัฏสวนสุนันทา

## แบบฝึกหัดที่ 5

### Navigation และ Routing ในแอปโมบายด้วย React Native

คำสั่ง

1. ส่งงานให้ตรงเวลา
2. จัดเอกสารตามรูปแบบการทำรายงาน
3. ห้ามลอกกัน

#### โปรแกรมในเอกสาร

1. Stack Navigator
  - โปรแกรม
1. ไฟล์: /src/screens/HomeScreen.js

```
import React from "react";
import { View, Text, Button, StyleSheet } from "react-native";

export default function HomeScreen({ navigation }) {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Home Screen</Text>
      <Button
        title="Go to Details"
        onPress={() =>
          navigation.navigate("Details", {
            itemId: 31,
            itemName: "Pharthiwath",
          })
        }
      />
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});
```

2. ไฟล์: /src/screens/DetailsScreen.js

```
import React from "react";
import { View, Text, Button, StyleSheet } from "react-native";

export default function DetailsScreen({ route, navigation }) {
  const { itemId, itemName } = route.params;
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Details Screen</Text>
      <Text style={styles.text}>Item ID: {itemId}</Text>
      <Text style={styles.text}>Item Name: {itemName}</Text>
      <Button title="Go Back" onPress={() => navigation.goBack()} />
      <Button
        title="Go to Home"
        onPress={() =>
          navigation.reset({
            index: 0,
            routes: [{ name: "Home" }],
          })
        />
      />
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
  text: {
    fontSize: 18,
  },
});
```

3. ไฟล์: /src/navigators/StackNavigator.js

```
import React from "react";
import { createStackNavigator } from "@react-navigation/stack";
import HomeScreen from "../screens/HomeScreen";
import DetailsScreen from "../screens/DetailsScreen";
const Stack = createStackNavigator();
export default function StackNavigator() {
  return (
    <Stack.Navigator
      screenOptions={{
        headerStyle: {
          backgroundColor: "#6200ee",
        },
        headerTintColor: "#fff",
        headerTitleStyle: {
          fontWeight: "bold",
        },
      }}
    >
      <Stack.Screen
        name="Home"
        component={HomeScreen}
        options={{ title: "Hello, Welcome" }}
      />
      <Stack.Screen
        name="Details"
        component={DetailsScreen}
        options={({ route }) => ({
          title: `Details of ${route.params.itemName}`,
        })}
      />
    </Stack.Navigator>
  );
}
```

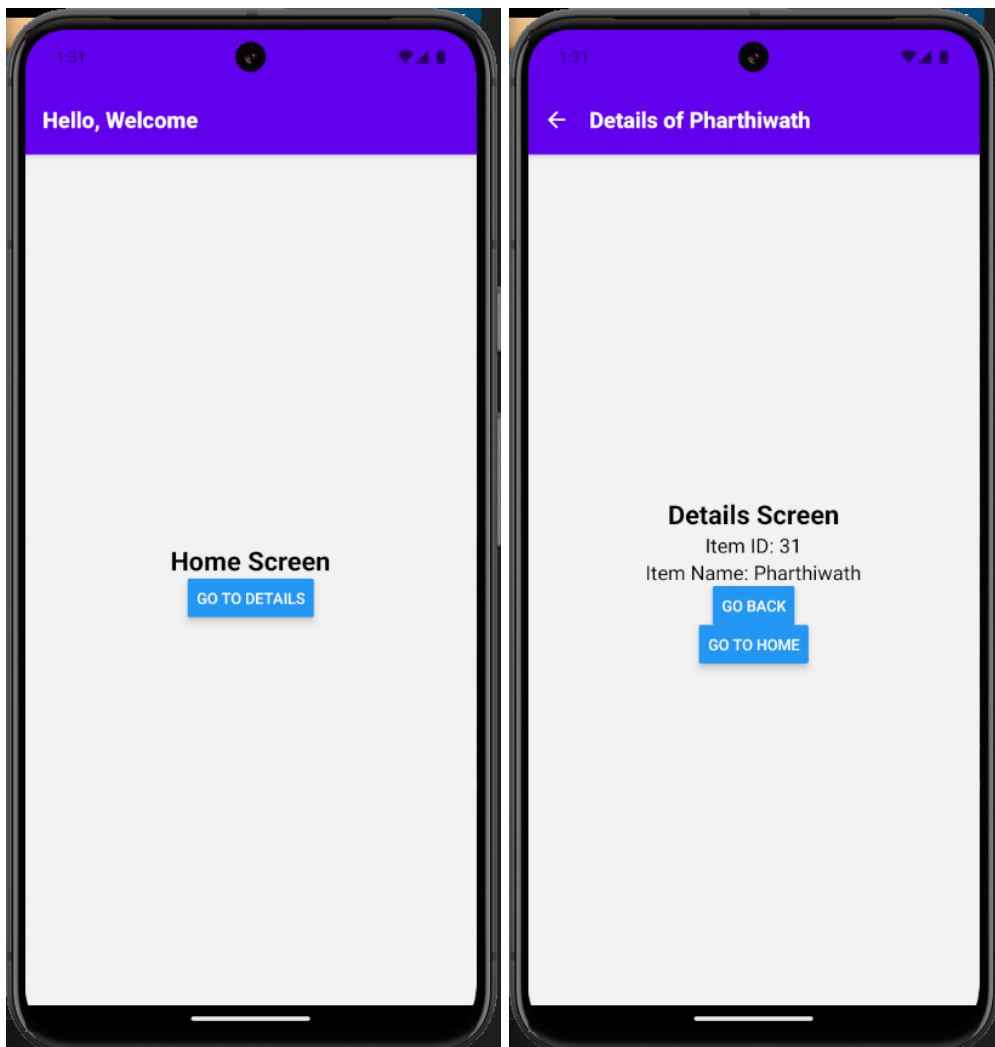
4. ไฟล์: /src/navigators/RootNavigator.js

```
import React from "react";
import { NavigationContainer } from "@react-navigation/native";
import StackNavigator from "./StackNavigator";
export default function RootNavigator() {
  return (
    <NavigationContainer>
      <StackNavigator />
    </NavigationContainer>
  );
}
```

5. ไฟล์: /App.js

```
import React from "react";
import RootNavigator from "./navigators/RootNavigator";
export default function App() {
  return <RootNavigator />;
}
```

- ผลลัพธ์ของโปรแกรม



## 2. Tab Navigation

- โปรแกรม

### 1. ไฟล์: /src/features/home/HomeScreen.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";
export default function HomeScreen() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Home Screen</Text>
    </View>
  );
}
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});
```

### 2. ไฟล์: /src/features/profile/ProfileScreen.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";

export default function ProfileScreen() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Profile Screen</Text>
    </View>
  );
}
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});
```

3. ไฟล์: /src/features/settings/SettingsScreen.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";

export default function SettingsScreen() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Settings Screen</Text>
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});
```

4. ไฟล์: /src/navigators/TabNavigator.js

```
import React from "react";
import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
import HomeScreen from "../features/home/HomeScreen";
import ProfileScreen from "../features/profile/ProfileScreen";
import SettingsScreen from "../features/settings/SettingsScreen";
import { Ionicons } from "@expo/vector-icons";

const Tab = createBottomTabNavigator();
export default function TabNavigator() {
  return (
    <Tab.Navigator
      screenOptions={({ route }) => ({
        tabBarIcon: ({ focused, color, size }) => {
          let iconName;

          if (route.name === "Home") {
            iconName = focused ? "home" : "home-outline";
          } else if (route.name === "Profile") {
            iconName = focused ? "person" : "person-outline";
          } else if (route.name === "Settings") {
            iconName = focused ? "settings" : "settings-outline";
          }

          return <Ionicons name={iconName} size={size} color={color} />;
        },

        tabBarActiveTintColor: "tomato",
        tabBarInactiveTintColor: "gray",
      })}
    >
    <Tab.Screen name="Home" component={HomeScreen} />
    <Tab.Screen name="Profile" component={ProfileScreen} />
    <Tab.Screen name="Settings" component={SettingsScreen} />
    </Tab.Navigator>
  );
}
```

5. ไฟล์: /src/navigators/RootNavigator.js

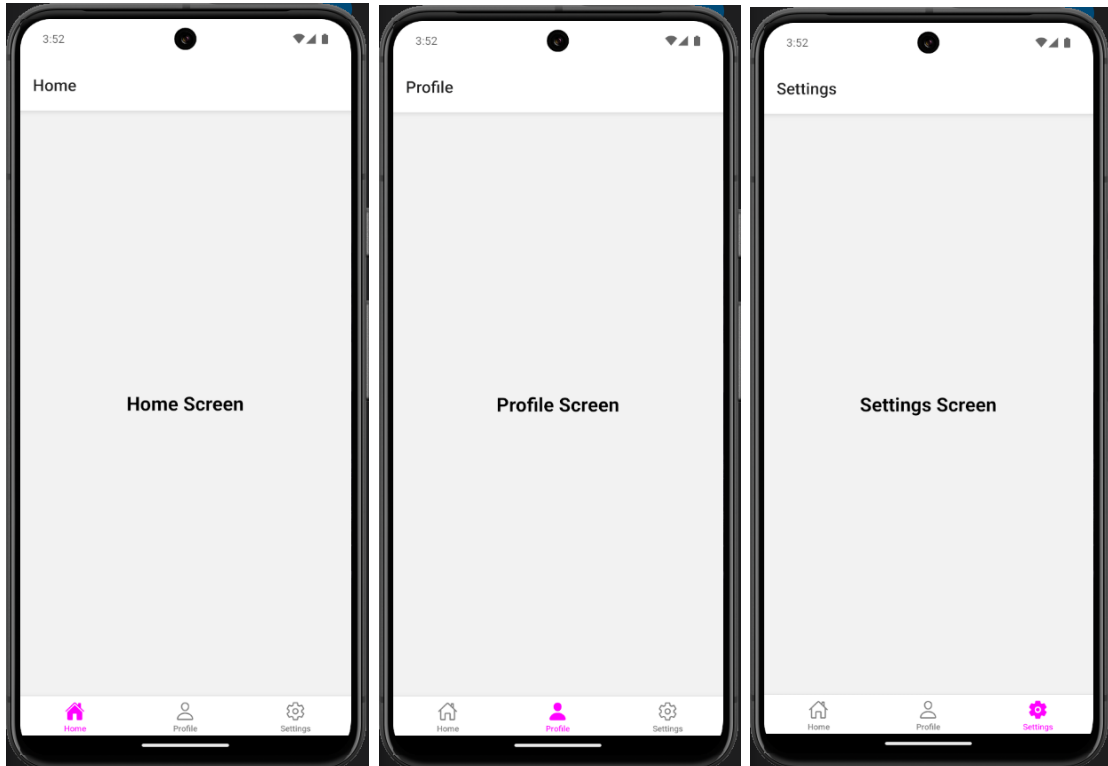
```
import React from "react";
import { NavigationContainer } from "@react-navigation/native";
import TabNavigator from "../TabNavigator";
export default function RootNavigator() {
  return (
    <NavigationContainer>
      <TabNavigator />
    </NavigationContainer>
  );
}
```



## 6. App.js

```
import React from "react";
import RootNavigator from "../navigators/RootNavigator";
export default function App() {
  return <RootNavigator />;
}
```

- ผลลัพธ์ของโปรแกรม



### 3. Drawer Navigation

- โปรแกรม

#### 1. ไฟล์ /src/components/CustomDrawerContent.js

```
import React from "react";
import { View, Text, StyleSheet, TouchableOpacity, Image } from "react-native";
import {
  DrawerContentScrollView,
  DrawerItemList,
} from "@react-navigation/drawer";
export default function CustomDrawerContent(props) {
  return (
    <DrawerContentScrollView {...props}>
      <View style={styles.header}>
        <Image
          source={{
            uri: "https://cdn.pixabay.com/photo/2020/07/01/12/58/icon-5359553_1280.png",
          }}
          style={styles.profileImage}
        />
        <Text style={styles.userName}>Pharthiwath</Text>
        <Text style={styles.userEmail}>s65122250031@ssru.ac.th</Text>
      </View>
      <View style={styles.menuSection}>
        <DrawerItemList {...props} />
      </View>
      <TouchableOpacity
        style={styles.logoutButton}
        onPress={() => handleLogout(props)}
      >
        <Text style={styles.logoutText}>Logout</Text>
      </TouchableOpacity>
    </DrawerContentScrollView>
  );
}

function handleLogout(props) {
  alert("Logged out successfully!");
  props.navigation.reset({
    index: 0,
    routes: [{ name: "Login" }],
  });
}

const styles = StyleSheet.create({
  header: {
    backgroundColor: "#6200ee",
    padding: 16,
    alignItems: "center",
  },
  profileImage: {
    width: 80,
    height: 80,
    borderRadius: 40,
  },
});
```

```
        marginBottom: 10,
      },
      userName: {
        color: "#fff",
        fontSize: 18,
        fontWeight: "bold",
      },
      userEmail: {
        color: "#fff",
        fontSize: 14,
      },
      menuSection: {
        flex: 1,
        paddingTop: 10,
      },
      logoutButton: {
        marginTop: 20,
        padding: 16,
        backgroundColor: "#f44336",
        alignItems: "center",
        borderRadius: 5,
        marginHorizontal: 10,
      },
      logoutText: {
        color: "#fff",
        fontSize: 16,
        fontWeight: "bold",
      },
    },
  });
```

2. ไฟล์ : /src/navigators/DrawerNavigator.js

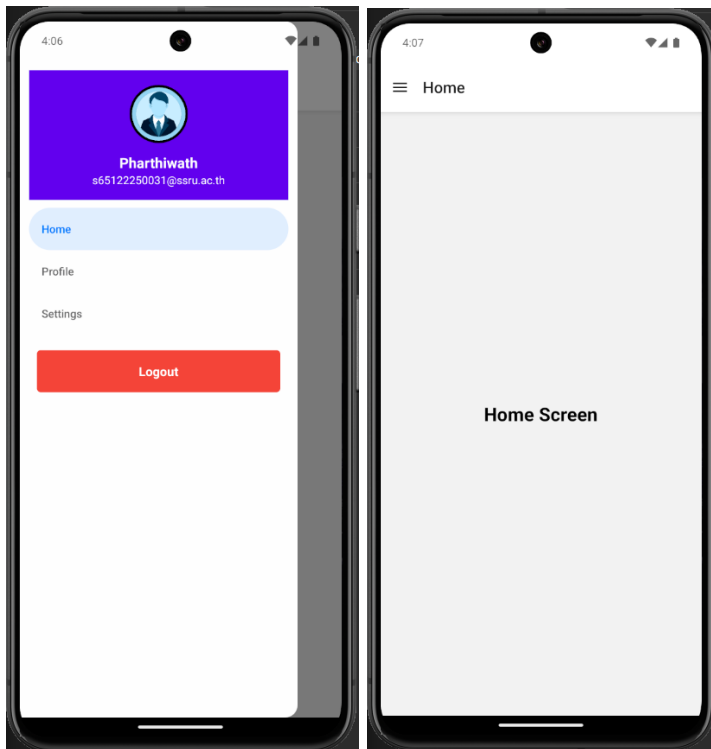
3. ไฟล์ : /src/navigators/RootNavigator.js

```
import React from "react";
import { NavigationContainer } from "@react-navigation/native";
import DrawerNavigator from "../DrawerNavigator";
export default function RootNavigator() {
  return (
    <NavigationContainer>
      <DrawerNavigator />
    </NavigationContainer>
  );
}
```

4. App.js

```
import React from "react";
import RootNavigator from "../navigators/RootNavigator";
export default function App() {
  return <RootNavigator />;
}
```

- ผลลัพธ์ของโปรแกรม



#### 4. Hybrid Navigation

- โปรแกรม

1. ไฟล์: /src /features/home/HomeScreen.js

```
import React from "react";
import { View, Text, Button, StyleSheet } from "react-native";
export default function HomeScreen({ navigation }) {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Home Screen</Text>
      <Button
        title="Go to Details"
        onPress={() => navigation.navigate("Details")}
      />
    </View>
  );
}
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});
```

2. ไฟล์: /src/features/home/DetailsScreen.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";
export default function DetailsScreen() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Details Screen</Text>
    </View>
  );
}
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});
```

### 3. StackNavigator.js

```
import React from "react";
import { createStackNavigator } from "@react-navigation/stack";
import HomeScreen from "../features/home/HomeScreen";
import DetailsScreen from "../features/detail/DetailsScreen";
const Stack = createStackNavigator();
export default function StackNavigator() {
  return (
    <Stack.Navigator>
      <Stack.Screen name="Home" component={HomeScreen} />
      <Stack.Screen name="Details" component={DetailsScreen} />
    </Stack.Navigator>
  );
}
```

### 4. TabNavigator.js

```
import React from "react";
import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
import StackNavigator from "../StackNavigator";
import ProfileScreen from "../features/profile/ProfileScreen";
const Tab = createBottomTabNavigator();
export default function TabNavigator() {
  return (
    <Tab.Navigator>
      <Tab.Screen name="HomeStack" component={StackNavigator} />
      <Tab.Screen name="Profile" component={ProfileScreen} />
    </Tab.Navigator>
  );
}
```

### 5. DrawerNavigator.js

```
import React from "react";
import { createDrawerNavigator } from "@react-navigation/drawer";
import TabNavigator from "../TabNavigator";
import SettingsScreen from "../features/settings/SettingsScreen";
const Drawer = createDrawerNavigator();
export default function DrawerNavigator() {
  return (
    <Drawer.Navigator>
      <Drawer.Screen name="Tabs" component={TabNavigator} />
      <Drawer.Screen name="Settings" component={SettingsScreen} />
    </Drawer.Navigator>
  );
}
```

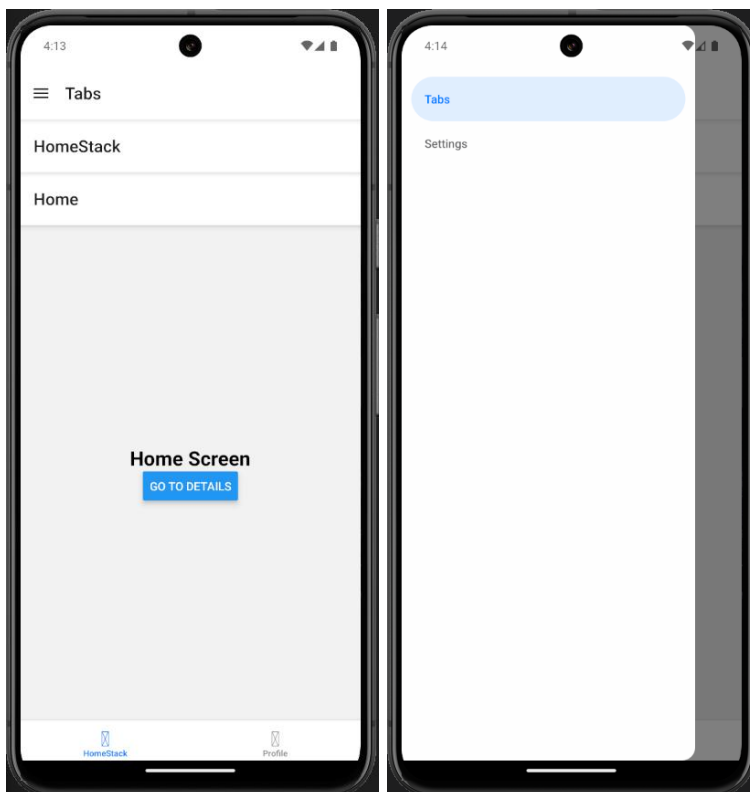
## 6. RootNavigator.js

```
import React from "react";
import { NavigationContainer } from "@react-navigation/native";
import DrawerNavigator from "../DrawerNavigator";
export default function RootNavigator() {
  return (
    <NavigationContainer>
      <DrawerNavigator />
    </NavigationContainer>
  );
}
```

## 7. App.js

```
import React from "react";
import RootNavigator from "../navigators/RootNavigator";
export default function App() {
  return <RootNavigator />;
}
```

- ผลลัพธ์ของโปรแกรม





## แบบฝึกหัด

โจทย์ที่ 1: Stack Navigator

ชื่อแอปพลิเคชัน: Book Store App

การทำงานของแอปพลิเคชัน:

แอปพลิเคชันนี้มีหน้าจอ "Home" สำหรับแสดงรายการหนังสือ และหน้าจอ "Details" ที่แสดงรายละเอียดของหนังสือที่ผู้ใช้เลือก

- โปรแกรม
  - screens/HomeScreen.js

```
import React from "react";
import { View, Text, Button, StyleSheet, ScrollView } from "react-native";

const books = [
  { id: 1, name: "Book 1", author: "Author 1" },
  { id: 2, name: "Book 2", author: "Author 2" },
  { id: 3, name: "Book 3", author: "Author 3" },
  { id: 4, name: "Book 4", author: "Author 4" },
  { id: 5, name: "Book 5", author: "Author 5" },
  { id: 6, name: "Book 6", author: "Author 6" },
  { id: 7, name: "Book 7", author: "Author 7" },
  { id: 8, name: "Book 8", author: "Author 8" },
  { id: 9, name: "Book 9", author: "Author 9" },
  { id: 10, name: "Book 10", author: "Author 10" },
];

export default function HomeScreen({ navigation }) {
  return (
    <ScrollView>
      <View style={styles.container}>
        {books.map((book) => (
          <View key={book.id} style={styles.card}>
            <Text style={styles.title}>{book.name}</Text>
            <Button
              title="Book's Details"
              color={"#186b50"}
              onPress={() =>
                navigation.navigate("Details", {
                  itemId: book.id,
                  itemName: book.name,
                  itemAuthor: book.author,
                })
              }
            </Button>
          </View>
        ))}
      </View>
    </ScrollView>
  )}
}
```

```

    </View>
  </ScrollView>
);
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    padding: 16,
  },
  card: {
    backgroundColor: "#fff",
    padding: 16,
    marginVertical: 8,
    borderRadius: 8,
    shadowColor: "#000",
    shadowOpacity: 0.1,
    shadowRadius: 10,
    shadowOffset: { width: 0, height: 5 },
  },
  title: {
    fontSize: 18,
    fontWeight: "bold",
    marginBottom: 8,
  },
});

```

- screens/DetailScreen.js

```

import React from "react";
import { View, Text, Button, StyleSheet } from "react-native";

export default function DetailsScreen({ route, navigation }) {
  const { itemId, itemName, itemAuthor } = route.params;

  return (
    <View style={styles.container}>
      <View style={styles.card}>
        <Text style={styles.title}>{itemName}</Text>
        <Text style={styles.text}>Author: {itemAuthor}</Text>
        <Text style={styles.text}>ID: {itemId}</Text>
        <Button
          title="Go to Home"
          color={"#186b50"}
          onPress={() =>
            navigation.reset({
              index: 0,
              routes: [{ name: "Home" }],
            })
          />
        </View>
      </View>
    </View>
  );
}

```

```

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
    padding: 16,
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
    marginBottom: 8,
  },
  text: {
    fontSize: 18,
    marginBottom: 8,
  },
  card: {
    backgroundColor: "#fff",
    padding: 32,
    marginVertical: 16,
    borderRadius: 8,
    shadowColor: "#000",
    shadowOpacity: 0.1,
    shadowRadius: 10,
    shadowOffset: { width: 0, height: 5 },
    elevation: 5,
    alignItems: "center",
  },
});

```

- navigators/StackNavigator.js

```

import React from "react";
import { createStackNavigator } from "@react-navigation/stack";
import HomeScreen from "../screens/HomeScreen";
import DetailsScreen from "../screens/DetailsScreen";
const Stack = createStackNavigator();
export default function StackNavigator() {
  return (
    <Stack.Navigator
      screenOptions={{
        headerStyle: {
          backgroundColor: "#186b50",
        },
        headerTintColor: "#fff",
        headerTitleStyle: {
          fontWeight: "bold",
        },
      }}
    >
      <Stack.Screen
        name="Home"
        component={HomeScreen}
        options={{ title: "Book Store App" }}
      />
    </Stack.Navigator>
  );
}

```

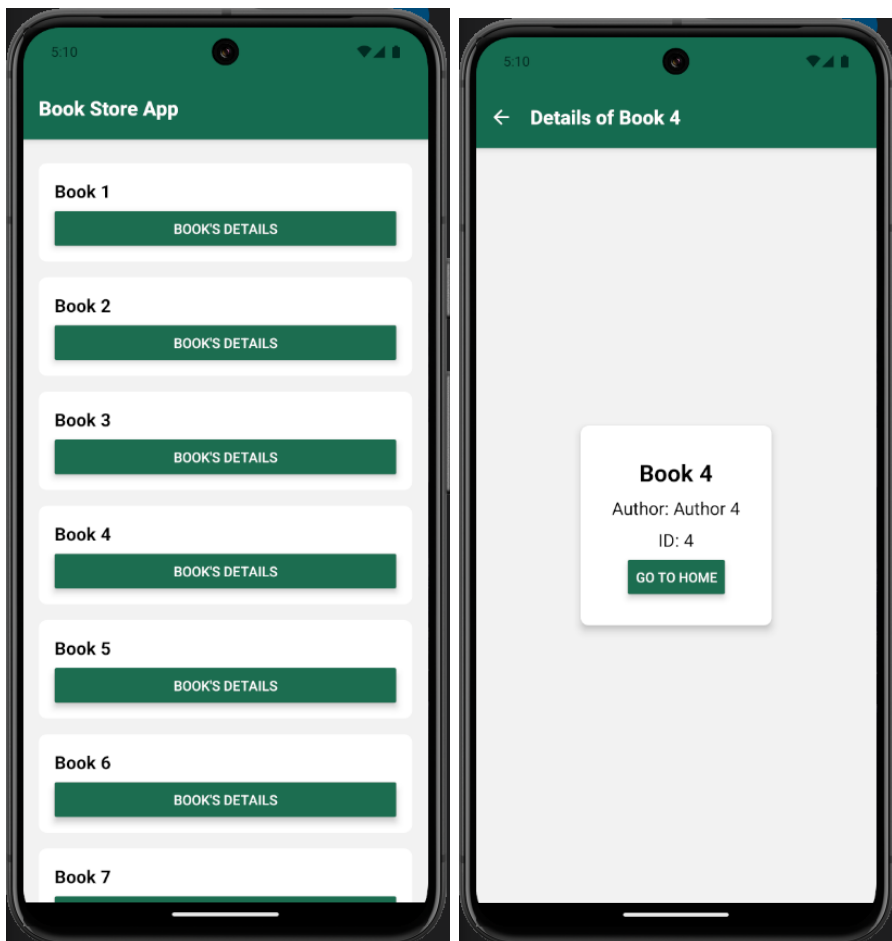
```

    />
    <Stack.Screen
      name="Details"
      component={DetailsScreen}
      options={({ route }) => ({
        title: `Details of ${route.params.itemName}`,
      })}
    />
  </Stack.Navigator>
);
}

```

- คำอธิบายโปรแกรม
- 1. สร้าง Stack Navigator:
  - ใช้ createStackNavigator จาก @react-navigation/stack
- 2. กำหนดหน้าต่างใน Stack Navigator:
  - HomeScreen: แสดงรายการหนังสือ
  - DetailsScreen: แสดงรายละเอียดหนังสือ
- 3. ตั้งค่า Stack Navigator:
  - กำหนดสไตล์ header: สีพื้นหลัง, สีข้อความ, สไตล์ข้อความ
- 4. การนำทาง:
  - จาก HomeScreen ไป DetailsScreen พร้อมพารามิเตอร์หนังสือ
- 5. การแสดงผลใน DetailsScreen:
  - รับพารามิเตอร์จาก route.params และแสดงรายละเอียดหนังสือ

- ผลลัพธ์ของโปรแกรม



## โจทย์ที่ 2: Tab Navigation

ชื่อแอปพลิเคชัน: Fitness Tracker App

การทำงานของแอปพลิเคชัน:

แอปนี้มี 3 Tab ได้แก่ Dashboard แสดงข้อมูลภาพรวม, Workouts แสดงรายการการออกกำลังกาย, และ Profile แสดงข้อมูลผู้ใช้

- โปรแกรม
  - Dashboard.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";
import { Icons } from "@expo/vector-icons";

export default function Dashboard() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Dashboard</Text>
      <Icons name={"desktop"} size={256} color="#7ab2d3" />
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    marginTop: 50,
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});
```

- Workouts.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";
import { Icons } from "@expo/vector-icons";

export default function Workouts() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Workouts</Text>
      <Icons name={"barbell"} size={256} color="#7ab2d3" />
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
```

```

    flex: 1,
    marginTop: 50,
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});

```

- Profile.js

```

import React from "react";
import { View, Text, StyleSheet } from "react-native";
import { Icons } from "@expo/vector-icons";

export default function Profile() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Profile</Text>
      <Icons name="person" size={256} color="#7ab2d3" />
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    marginTop: 50,
    alignItems: "center",
  },
  title: {
    fontSize: 24,
    fontWeight: "bold",
  },
});

```

- TabNavigator.js

```

import React from "react";
import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
import Dashboard from "../fitness/Dashboard";
import Workouts from "../fitness/Workouts";
import Profile from "../fitness/Profile";
import { Icons } from "@expo/vector-icons";

const Tab = createBottomTabNavigator();

export default function TabNavigator() {
  return (
    <Tab.Navigator
      screenOptions={({ route }) => ({
        tabBarIcon: ({ focused, color, size }) => {
          let iconName;

          if (route.name === "Dashboard") {
            iconName = focused ? "desktop" : "desktop-outline";
          } else if (route.name === "Workouts") {

```

```

        iconName = focused ? "barbell" : "barbell-outline";
    } else if (route.name === "Profile") {
        iconName = focused ? "person" : "person-outline";
    }

    return <Ionicons name={iconName} size={size} color={color} />;
},

tabBarActiveTintColor: "#3c6ca5",
tabBarInactiveTintColor: "gray",
}}}
>
<Tab.Screen name="Dashboard" component={Dashboard} />
<Tab.Screen name="Workouts" component={Workouts} />
<Tab.Screen name="Profile" component={Profile} />
</Tab.Navigator>
);
}

```

- คำอธิบายโปรแกรม

1. สร้าง Bottom Tab Navigator:

- ใช้ createBottomTabNavigator จาก @react-navigation/bottom-tabs เพื่อสร้าง bottom tab navigator สำหรับการนำทางระหว่างหน้าต่างๆ

2. กำหนดหน้าต่างใน Tab Navigator:

- มีสามหน้าต่างหลักคือ Dashboard, Workouts, และ Profile
- แต่ละหน้าต่างมีคอมโพเนนต์ของตัวเองที่แสดงเนื้อหาที่เกี่ยวข้อง

3. กำหนดการตั้งค่าของ Tab Navigator:

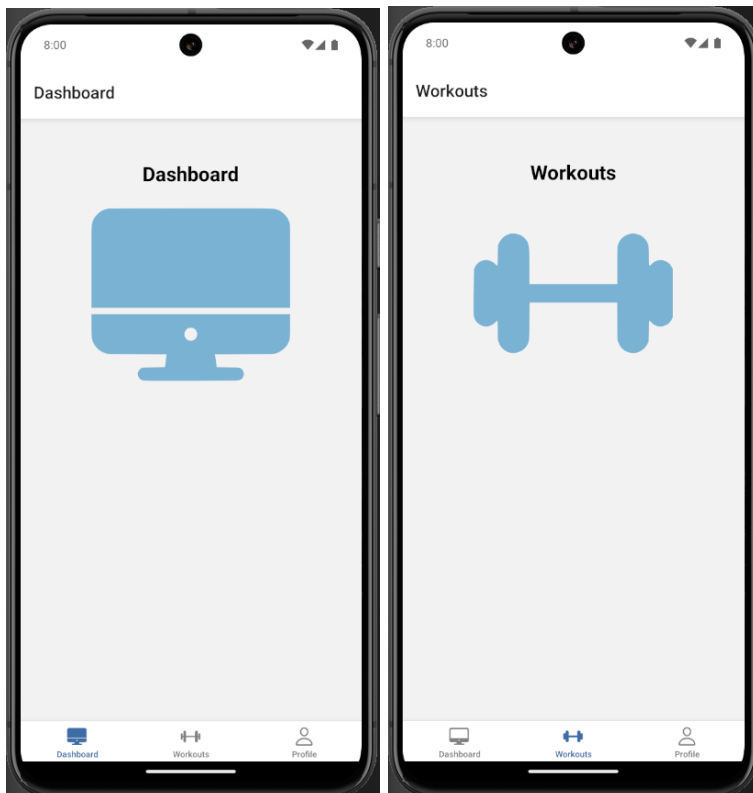
- screenOptions กำหนดไอคอนของแต่ละแท็บโดยใช้ Ionicons จาก @expo/vector-icons
- ไอคอนจะเปลี่ยนไปตามสถานะ focused (ถูกเลือกหรือไม่)
- กำหนดสีของแท็บที่ถูกเลือก (tabBarActiveTintColor) และแท็บที่ไม่ถูกเลือก (tabBarInactiveTintColor)

4. การแสดงผลในแต่ละหน้าต่าง:

- Dashboard, Workouts, และ Profile แต่ละหน้าต่างมีการแสดงผลที่แตกต่างกัน โดยใช้ View, Text, และ Ionicons เพื่อแสดงเนื้อหาและไอคอนที่เกี่ยวข้อง



- ผลลัพธ์ของโปรแกรม



### โจทย์ที่ 3: Drawer Navigation

ชื่อแอปพลิเคชัน: Travel App

การทำงานของแอปพลิเคชัน:

แอปนี้มีเมนู Drawer เพื่อเข้าถึงหน้าหลัก (Home), การจองโรงแรม (Hotels), และการจองเที่ยวบิน (Flights)

- โปรแกรม

- Home.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";
import { Icons } from "@expo/vector-icons";

export default function Home() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Home</Text>
      <Icons name="home" size={256} color="#7ab2d3" />
    </View>
  );
}
```

- Hotels.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";

export default function Hotels() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Hotels</Text>
      <View style={styles.card}>
        <Text style={styles.hotelName}>Hotel 1</Text>
        <Text style={styles.hotelName}>Hotel 2</Text>
        <Text style={styles.hotelName}>Hotel 3</Text>
        <Text style={styles.hotelName}>Hotel 4</Text>
        <Text style={styles.hotelName}>Hotel 5</Text>
      </View>
    </View>
  );
}
```

- Flights.js

```
import React from "react";
import { View, Text, StyleSheet } from "react-native";

export default function Flights() {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Flights</Text>
      <View style={styles.card}>
        <Text style={styles.flightName}>Flight 1</Text>
        <Text style={styles.flightName}>Flight 2</Text>
        <Text style={styles.flightName}>Flight 3</Text>
        <Text style={styles.flightName}>Flight 4</Text>
        <Text style={styles.flightName}>Flight 5</Text>
      </View>
    </View>
  );
}
```

- DrawerNavigator.js

```
import React from "react";
import { createDrawerNavigator } from "@react-navigation/drawer";
import Home from "../travel/Home";
import Hotels from "../travel/Hotels";
import Flights from "../travel/Flights";
import CustomDrawerContent from "../components/CustomDrawerContent";
const Drawer = createDrawerNavigator();
export default function DrawerNavigator() {
  return (
    <Drawer.Navigator
      drawerContent={({props}) => <CustomDrawerContent {...props} />}
    >
      <Drawer.Screen name="Home" component={Home} />
      <Drawer.Screen name="Hotels" component={Hotels} />
      <Drawer.Screen name="Flights" component={Flights} />
    </Drawer.Navigator>
  );
}
```

- คำอธิบายโปรแกรม

1. สร้าง Drawer Navigator:

- ใช้ `createDrawerNavigator` จาก `@react-navigation/drawer` เพื่อสร้าง drawer navigator สำหรับการนำทางระหว่างหน้าต่างๆ

2. กำหนดหน้าต่างใน Drawer Navigator:

- มีสามหน้าต่างหลักคือ Home, Hotels, และ Flights
- แต่ละหน้าต่างมีคอมโพเนนต์ของตัวเองที่แสดงเนื้อหาที่เกี่ยวข้อง

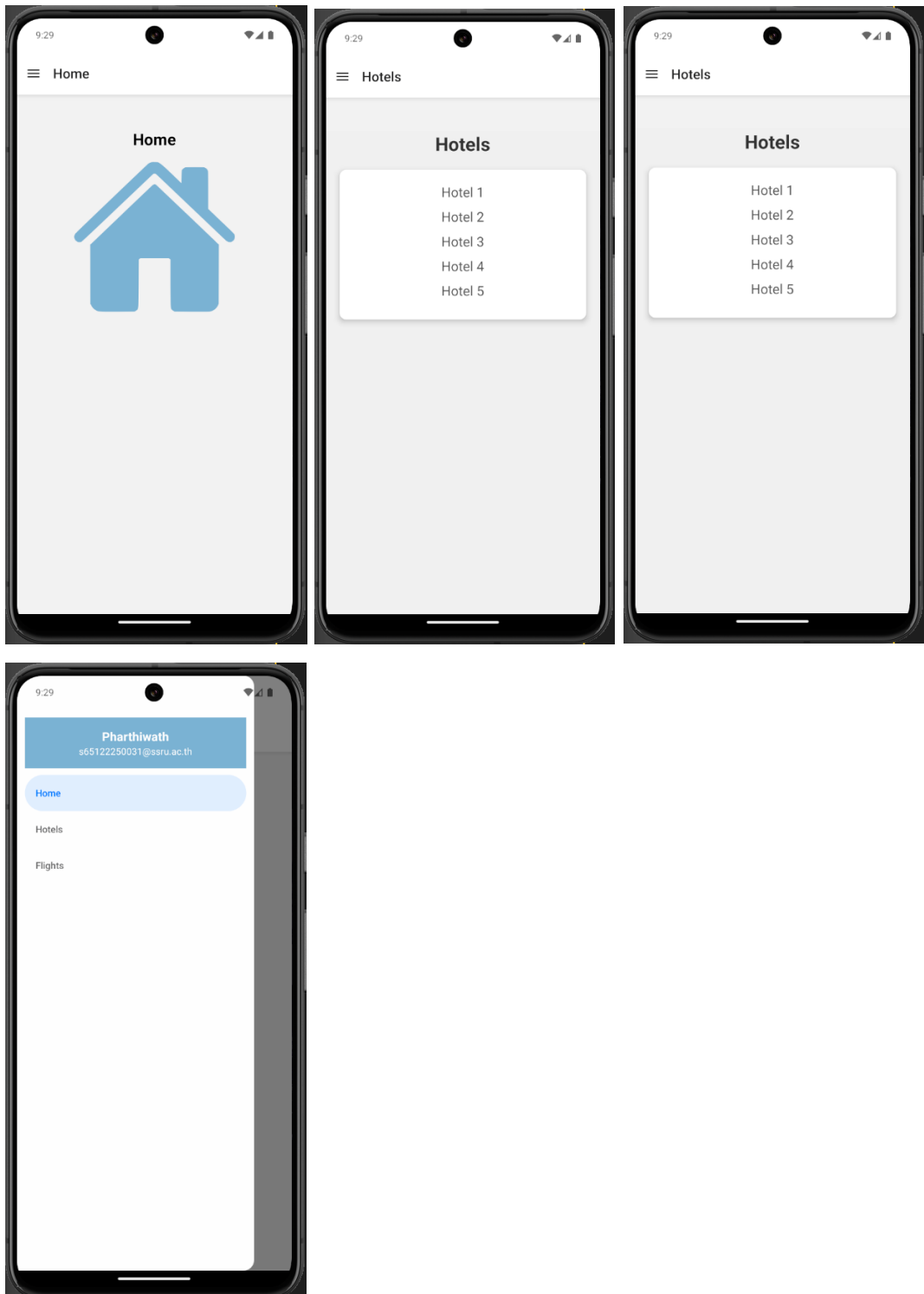
3. กำหนดเนื้อหาใน Drawer:

- ใช้ `CustomDrawerContent` เพื่อกำหนดเนื้อหาที่จะแสดงใน drawer เช่น ข้อมูลผู้ใช้และรายการเมนู

4. การแสดงผลในแต่ละหน้าต่าง:

- Home แสดงไอคอนบ้าน
- Hotels แสดงรายการโรงแรม
- Flights แสดงรายการเที่ยวบิน

- ผลลัพธ์ของโปรแกรม



#### โจทย์ที่ 4: Hybrid Navigation

ชื่อแอปพลิเคชัน: E-Commerce App

การทำงานของแอปพลิเคชัน:

ใช้ Drawer Navigation เป็นโครงสร้างหลัก โดยในเมนู Home จะใช้ Tab Navigation และแต่ละ Tab ใช้ Stack Navigation

- โปรแกรม
  - HybridDrawerNavigator.js

```
import React from "react";
import { createDrawerNavigator } from "@react-navigation/drawer";
import HybridTabNavigator from "../HybridTabNavigator";
import Settings from "../ecommerce/Settings";

const Drawer = createDrawerNavigator();
export default function HybridDrawerNavigator() {
  return (
    <Drawer.Navigator>
      <Drawer.Screen
        name="Home"
        component={HybridTabNavigator}
        options={{ headerTitle: "E-Commerce App" }}
      />
      <Drawer.Screen
        name="Settings"
        component={Settings}
        options={{ headerTitle: "E-Commerce App" }}
      />
    </Drawer.Navigator>
  );
}
```

- HybridTabNavigator.js

```
import React from "react";
import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
import Home from "../ecommerce/Home";
import Profile from "../ecommerce/Profile";
import HybridStackNavigator from "../HybridStackNavigator";
import { Ionicons } from "@expo/vector-icons";

const Tab = createBottomTabNavigator();
export default function HybridTabNavigator() {
  return (
    <Tab.Navigator
      screenOptions={({ route }) => ({
        tabBarIcon: ({ focused, color, size }) => {
          let iconName;

          if (route.name === "Home") {
            iconName = focused ? "home" : "home-outline";
          } else if (route.name === "Profile") {
            iconName = focused ? "person" : "person-outline";
          }

          return <Ionicons name={iconName} size={size} color={color} />;
        },

        tabBarActiveTintColor: "#3c6ca5",
        tabBarInactiveTintColor: "gray",
        headerShown: false,
      })}
    >
    <Tab.Screen name="Home" component={HybridStackNavigator} />
    <Tab.Screen name="Profile" component={Profile} />
    </Tab.Navigator>
  );
}
```

- HybridStackNavigator.js

```
import React from "react";
import { createStackNavigator } from "@react-navigation/stack";
import Home from "../ecommerce/Home";
import Product from "../ecommerce/Product";

const Stack = createStackNavigator();
export default function StackNavigator() {
  return (
    <Stack.Navigator
      screenOptions={{
        headerShown: false,
      }}
    >
      <Stack.Screen name="Home" component={Home} />
      <Stack.Screen name="Product" component={Product} />
    </Stack.Navigator>
  );
}
```

- Home.js

```
import React from "react";
import { View, Text, Button, StyleSheet, FlatList } from "react-native";

const products = [
  { id: 1, name: "Product 1", price: 100 },
  { id: 2, name: "Product 2", price: 200 },
  { id: 3, name: "Product 3", price: 300 },
  { id: 4, name: "Product 4", price: 400 },
  { id: 5, name: "Product 5", price: 500 },
  { id: 6, name: "Product 6", price: 600 },
  { id: 7, name: "Product 7", price: 700 },
  { id: 8, name: "Product 8", price: 800 },
  { id: 9, name: "Product 9", price: 900 },
  { id: 10, name: "Product 10", price: 1000 },
  { id: 11, name: "Product 11", price: 1100 },
  { id: 12, name: "Product 12", price: 1200 },
];

export default function Home({ navigation }) {
  return (
    <FlatList
      data={products}
      renderItem={({ item: product }) => (
        <View key={product.id} style={styles.card}>
          <Text style={styles.title}>{product.name}</Text>
          <Text style={styles.price}>₹{product.price}</Text>
          <Button
            title="Product's Details"
            color={"#3c6ca5"}
            onPress={() =>
              navigation.navigate("Product", {
                id: product.id,
              })
            }
          />
        </View>
      )}
    />
  );
}
```



```

        name: product.name,
        price: product.price,
      })
    }
  />
</View>
)}
keyExtractor={({product}) => product.id.toString()}
numColumns={2}
contentContainerStyle={styles.container}
  />
);
}

```

- Product.js

```

import React from "react";
import { View, Text, Button, StyleSheet } from "react-native";

export default function Product({ route, navigation }) {
  const { name, price } = route.params;

  return (
    <View style={styles.container}>
      <View style={styles.card}>
        <Text style={styles.title}>{name}</Text>
        <Text style={styles.text}>₹ {price}</Text>
        <Button
          title="Go to Home"
          color={"#3c6ca5"}
          onPress={() =>
            navigation.reset({
              index: 0,
              routes: [{ name: "Home" }],
            })
          />
        </View>
      </View>
    );
  }
}

```

- Profile.js

```

import React from "react";
import { View, Text, StyleSheet } from "react-native";
import { Icons } from "@expo/vector-icons";

export default function Profile() {
  return (
    <View style={styles.container}>
      <Icons name={"person"} size={256} color={"#7ab2d3"} />
      <Text style={styles.title}>Pharthiwath Gristsoopharruth</Text>
    </View>
  );
}

```

- Settings.js

```
import React from "react";
import { View, Text, StyleSheet, ScrollView } from "react-native";
import { Ionicons } from "@expo/vector-icons";

export default function Settings() {
  return (
    <ScrollView>
      <View style={styles.container}>
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
        <Ionicons name="settings" size={48} color="#7ab2d3" />
      </View>
    </ScrollView>
  );
}
```

- คำอธิบายโปรแกรม
  - สร้าง Drawer Navigator:
    - ใช้ createDrawerNavigator จาก @react-navigation/drawer
  - กำหนดหน้าต่างใน Drawer Navigator:
    - Home (ใช้ HybridTabNavigator)
    - Settings
  - สร้าง Tab Navigator:
    - ใช้ createBottomTabNavigator จาก @react-navigation/bottom-tabs
    - มีแท็บ Home และ Profile
  - สร้าง Stack Navigator:
    - ใช้ createStackNavigator จาก @react-navigation/stack
    - มีหน้าต่าง Home และ Product
  - หน้าหลัก (Home):
    - แสดงรายการสินค้าในรูปแบบ FlatList
    - หน้ารายละเอียดสินค้า (Product):
      - แสดงรายละเอียดของสินค้าที่เลือก

6. หน้าโปรไฟล์ (Profile):

- แสดงข้อมูลโปรไฟล์ผู้ใช้

7. หน้า Settings:

- แสดงไอคอนการตั้งค่า

- ผลลัพธ์ของโปรแกรม

