



- **React**
- **Javascripts**



Diseño y Programación de Software Multiplataformas

Ing. Karen Medrano

**Desafio Practico 2**



## **Universidad Don Bosco**


❖ **Integrantes:**

- Miguel Angel Herrera Arreaga  
HA160257
- Anderson Alessandro Pablo Beltran  
PB230898

❖ **Grupo Laboratorio: 03L**

- **Funcionalidad del ejercicio 1**

- Damos clic en registrar usuario para poder iniciar sesión después con las credenciales registradas



- Luego iniciamos Sesión con las credenciales que hemos registrado anteriormente una vez la credencial sea correcta pasa a siguiente sección que es menú principal de las actividades



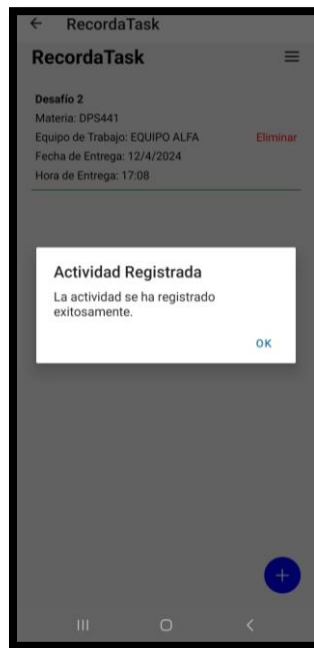
- Ver iniciado sesión entra a la sección principal donde se en listara la actividad registrada



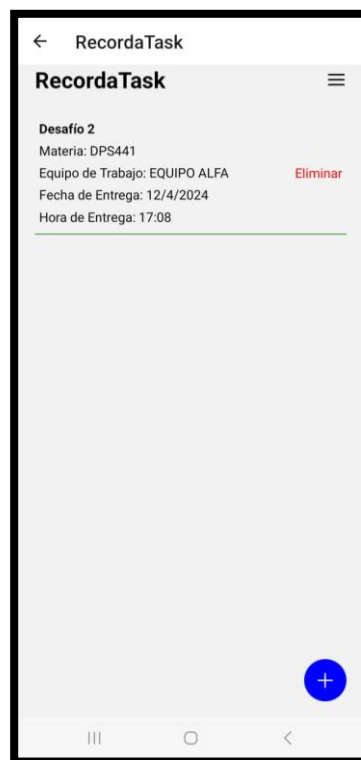
- Le damos Clic a botón azul en donde se abrirá el formulario para crear la nueva actividad

A screenshot of a mobile application titled "Registro de Actividades". The screen shows a header with a back arrow and the title "Registro de Actividades". Below the header, the text "Registro de Actividad" is displayed. There are five input fields: "Nombre", "Materia", "Equipo de Trabajo", "12/4/2024", and "17:08:08". Below these fields is a blue button labeled "Registrar Actividad". The bottom of the screen shows standard Android navigation icons.

- Una vez hayamos terminado de llenar el formulario de la actividad que guardaremos, nos aparecerá un aviso que la actividad se registro exitosamente



- Después que damos en botón ok, aparecerá el menú principal en donde se guardaran toda la actividad por hacer que hayamos registrado.
- En donde podremos ir eliminando las actividades que ya no necesitamos y si queremos editar alguna actividad solo damos clic en la actividad y se nos abrirá una sección donde esta el formulario para hacer el cambio



- Sección donde podremos editar cualquier actividad registrada

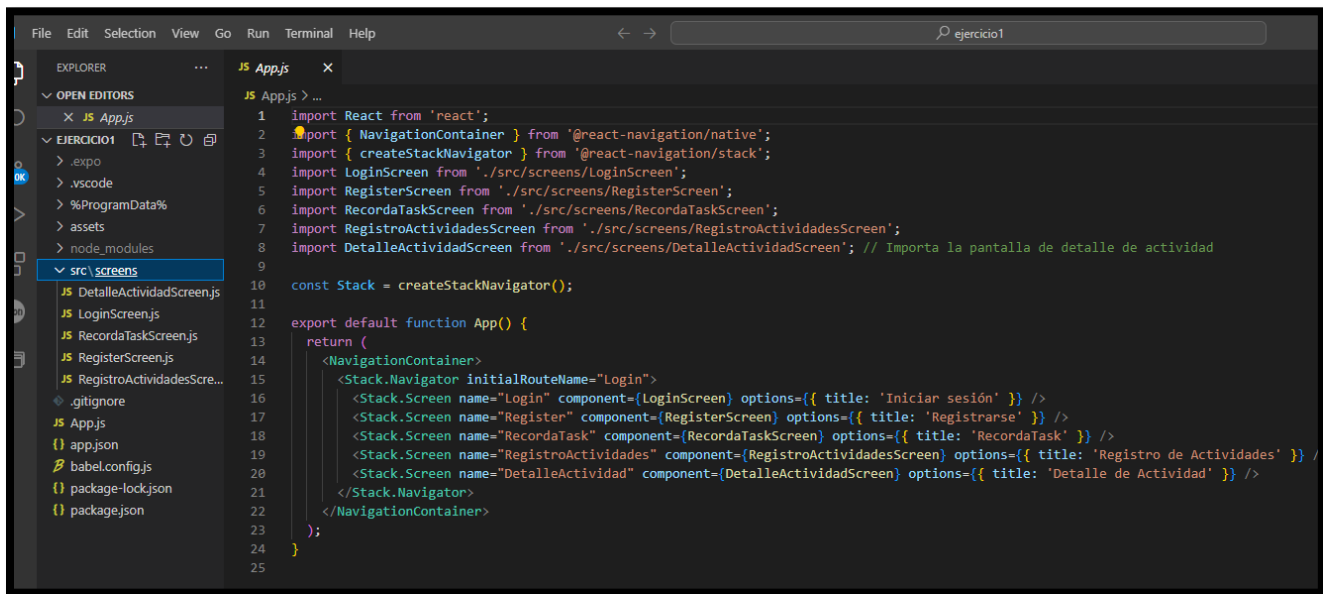


The screenshot shows a mobile application interface for editing an activity. At the top, there is a back arrow and the title 'Detalle de Actividad'. Below this, the section 'Editar Actividad' is centered. The form contains six input fields: 'Desafio 2', 'DPS441', 'Equipo Alfa', and two date-time fields both showing '2024-04-12T23:09:42.256Z'. A blue button labeled 'Guardar Cambios' is positioned below the form. The bottom of the screen features a standard Android navigation bar with three icons.

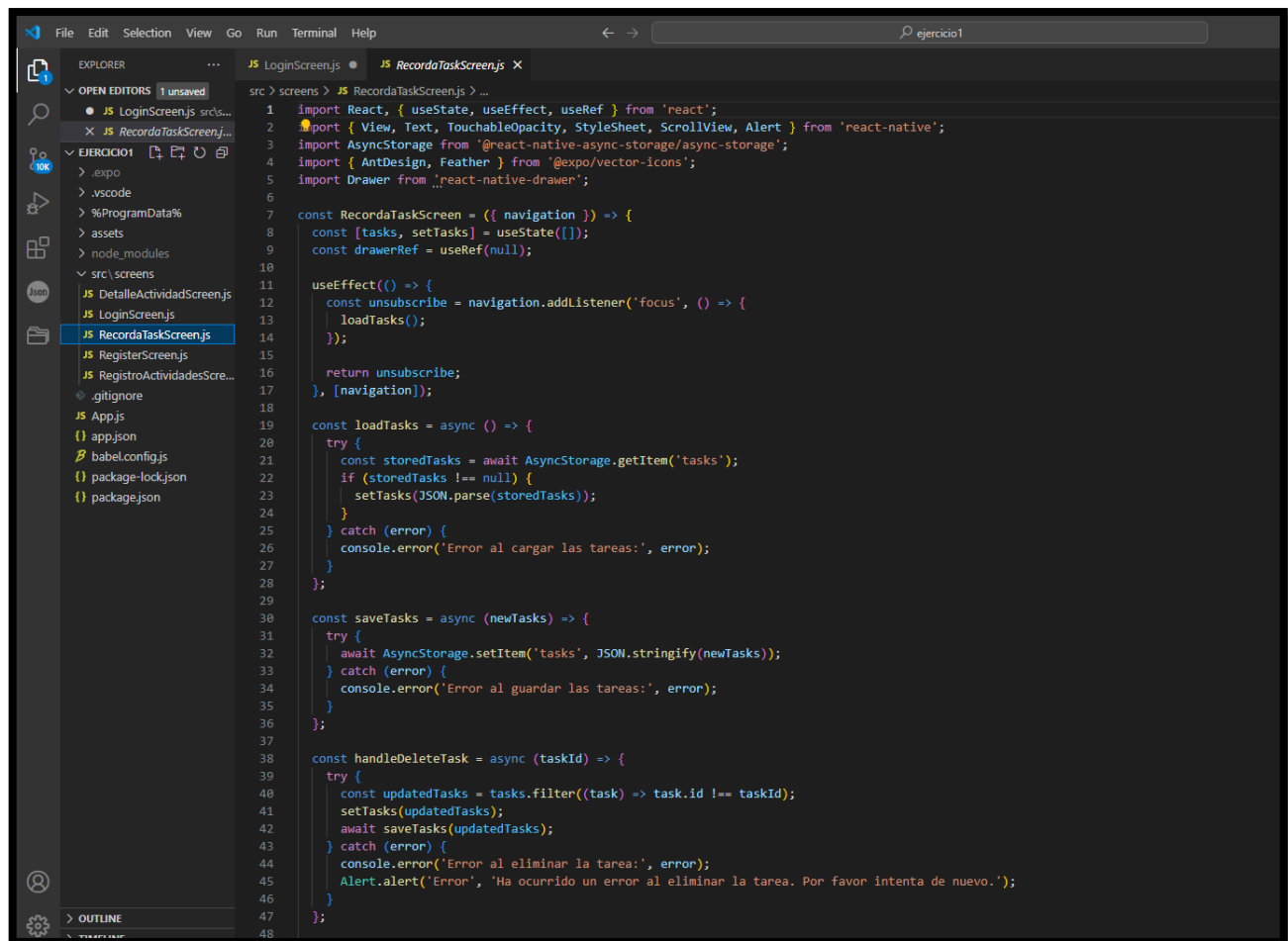
- Menu de navegación dentro de la aplicación



## Código de Ejercicio 1



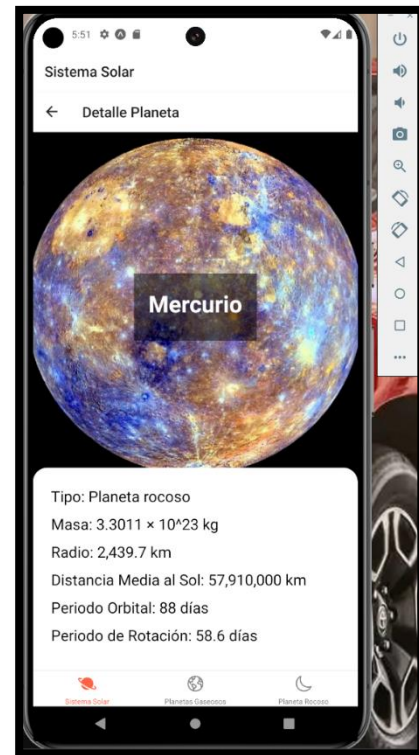
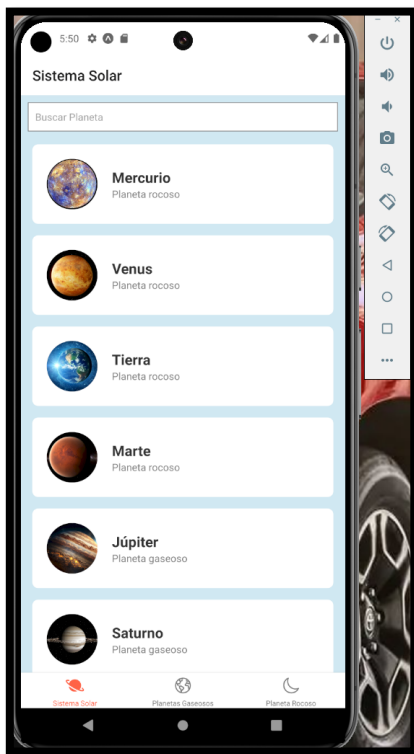
```
1 import React from 'react';
2 import { NavigationContainer } from '@react-navigation/native';
3 import { createStackNavigator } from '@react-navigation/stack';
4 import LoginScreen from '../src/screens/LoginScreen';
5 import RegisterScreen from '../src/screens/RegisterScreen';
6 import RecordTaskScreen from '../src/screens/RecordTaskScreen';
7 import RegistroActividadesScreen from '../src/screens/RegistroActividadesScreen';
8 import DetalleActividadScreen from '../src/screens/DetalleActividadScreen'; // Importa la pantalla de detalle de actividad
9
10 const Stack = createStackNavigator();
11
12 export default function App() {
13   return (
14     <NavigationContainer>
15     <Stack.Navigator initialRouteName="Login">
16       <Stack.Screen name="Login" component={LoginScreen} options={{ title: 'Iniciar sesión' }} />
17       <Stack.Screen name="Register" component={RegisterScreen} options={{ title: 'Registrarse' }} />
18       <Stack.Screen name="RecordTask" component={RecordTaskScreen} options={{ title: 'Recordar Tarea' }} />
19       <Stack.Screen name="RegistroActividades" component={RegistroActividadesScreen} options={{ title: 'Registro de Actividades' }} />
20       <Stack.Screen name="DetalleActividad" component={DetalleActividadScreen} options={{ title: 'Detalle de Actividad' }} />
21     </Stack.Navigator>
22   ) />
23 }
24
25
```



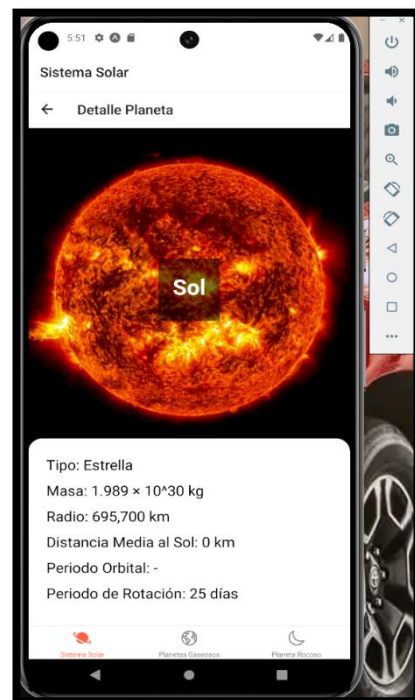
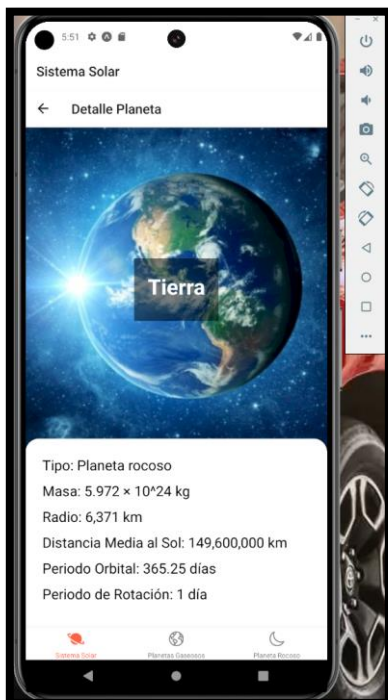
```
1 import React, { useState, useEffect, useRef } from 'react';
2 import { View, Text, TouchableOpacity, StyleSheet, ScrollView, Alert } from 'react-native';
3 import AsyncStorage from '@react-native-async-storage/async-storage';
4 import { AntDesign, Feather } from '@expo/vector-icons';
5 import Drawer from 'react-native-drawer';
6
7 const RecordTaskScreen = ({ navigation }) => {
8   const [tasks, setTasks] = useState([]);
9   const drawerRef = useRef(null);
10
11   useEffect(() => {
12     const unsubscribe = navigation.addListener('focus', () => {
13       loadTasks();
14     });
15     return unsubscribe;
16   }, [navigation]);
17
18   const loadTasks = async () => {
19     try {
20       const storedTasks = await AsyncStorage.getItem('tasks');
21       if (storedTasks !== null) {
22         setTasks(JSON.parse(storedTasks));
23       }
24     } catch (error) {
25       console.error('Error al cargar las tareas:', error);
26     }
27   };
28
29   const saveTasks = async (newTasks) => {
30     try {
31       await AsyncStorage.setItem('tasks', JSON.stringify(newTasks));
32     } catch (error) {
33       console.error('Error al guardar las tareas:', error);
34     }
35   };
36
37   const handleDeleteTask = async (taskId) => {
38     try {
39       const updatedTasks = tasks.filter(task => task.id !== taskId);
40       setTasks(updatedTasks);
41       await saveTasks(updatedTasks);
42     } catch (error) {
43       console.error('Error al eliminar la tarea:', error);
44       Alert.alert('Error', 'Ha ocurrido un error al eliminar la tarea. Por favor intenta de nuevo.');
```

## • Funcionalidad del Ejercicio 2

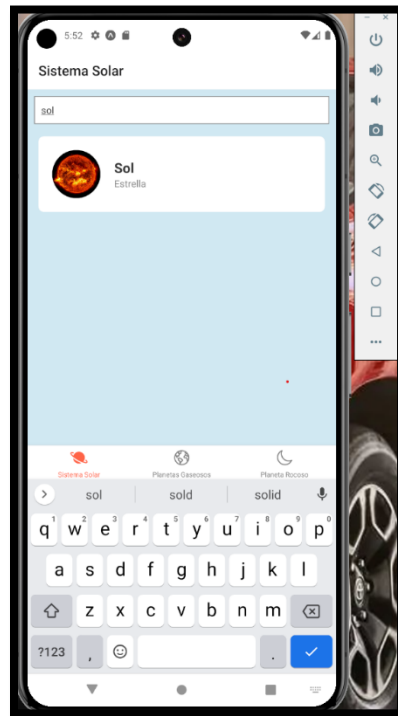
- Menú principal de la lista de los planetas



- Pantalla al darle clic a un planeta saldrá su información



- Buscador de Planeta según el nombre de cada Planeta



- opción de menú de Navegación llamada Planeta Gaseoso
- opción de menú de Navegación llamada Planeta Rocosos





- Código de App del Ejercicio 2

```

1 import React from 'react';
2 import { NavigationContainer } from '@react-navigation/native';
3 import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';
4 import { createStackNavigator } from '@react-navigation/stack';
5 import Ionicons from 'react-native-vector-icons/Ionicons';
6 import SistemaSolar from './screens/SistemaSolar';
7 import DetallePlaneta from './screens/DetallePlaneta';
8 import PlanetasGaseosos from './screens/PlanetaGaseoso';
9 import PlanetaRocoso from './screens/PlanetaRocoso';
10
11 const Tab = createBottomTabNavigator();
12 const Stack = createStackNavigator();
13 const SistemaSolarStack = () => {
14   return (
15     <Stack.Navigator>
16       <Stack.Screen name="Sistema Solar" component={SistemaSolar} options={{ headerShown: false }} />
17       <Stack.Screen name="Detalle Planeta" component={DetallePlaneta} />
18       <Stack.Screen name="Planeta Gaseoso" component={PlanetasGaseosos} />
19       <Stack.Screen name="Planeta" component={PlanetaRocoso} />
20     </Stack.Navigator>
21   );
22 };
23
24 const App = () => {
25   return (
26     <NavigationContainer>
27       <Tab.Navigator
28         screenOptions={({ route }) => ({
29           tabBarIcon: ({ focused, color, size }) => {
30             let iconName;
31
32             if (route.name === 'Sistema Solar') {
33               iconName = focused ? 'planet' : 'planet-outline';
34             } else if (route.name === 'Planetas Gaseosos') {
35               iconName = focused ? 'earth' : 'earth-outline';
36             } else if (route.name === 'Planeta Rocoso') {
37               iconName = focused ? 'moon' : 'moon-outline';
38             }
39
40             return <Ionicons name={iconName} size={size} color={color} />;
41           },
42         }) />
43       </Tab.Navigator>
44     </NavigationContainer>
45   );
46 };

```

```

1 import React, { useState, useEffect } from 'react';
2 import { View, Text, FlatList, TouchableOpacity, StyleSheet, TextInput, Image } from 'react-native';
3
4 const SistemaSolar = ({ navigation }) => {
5   const [planetas, setPlanetas] = useState([]);
6   const [searchQuery, setSearchQuery] = useState('');
7
8   useEffect(() => {
9     fetchPlanetas();
10   }, []);
11
12   const fetchPlanetas = async () => {
13     try {
14       const response = await fetch('https://661783e1ed6b8fa43482df1.mockapi.io/Planetas');
15       const data = await response.json();
16       setPlanetas(data);
17     } catch (error) {
18       console.error('Error fetching planetas:', error);
19     }
20   };
21
22   const renderItem = ({ item }) => (
23     <TouchableOpacity
24       style={styles.item}
25       onPress={() => navigation.navigate('Detalle Planeta', { planeta: item })}
26     >
27       <Image source={{ uri: item.imagen }} style={styles.planetaImage} />
28       <View style={styles.planetaInfo}>
29         <Text style={styles.planetaName}>{item.nombre}</Text>
30         <Text style={styles.planetaType}>{item.tipo}</Text>
31       </View>
32     </TouchableOpacity>
33   );
34
35   const handleSearch = () => {
36     const filteredPlanetas = planetas.filter(planeta =>
37       planeta.nombre.toLowerCase().includes(searchQuery.toLowerCase())
38     );
39     return filteredPlanetas;
40   };
41
42   return (
43     <View style={styles.container}>
44       <TextInput
45         style={styles.input}
46         placeholder="Buscar Planeta"
47         onChangeText={setSearchQuery}
48         value={searchQuery}

```