Part 2: Automated Testing

Scenario: Implement Automated Tests for the Flutter App

Instructions:

- Integration tests were written using the Flutter Integration Testing framework.
- The following scenarios were covered:
    1. Adding a new item
    2. Deleting an existing item
- The tests were executed successfully on both Android and iOS emulators.

Deliverables

1. Automated Test Code

The following test scripts can be executed within the app’s environment:

```
import 'package:flutter/material.dart';
import 'package:flutter_test/flutter_test.dart';
import 'package:integration_test/integration_test.dart';
import 'package:technoprise/main.dart';

void main() {
 IntegrationTestWidgetsFlutterBinding.ensureInitialized();

 group('Technoprise App Integration Tests', () {
   testWidgets('Add new item', (WidgetTester tester) async {
     // Launch the app
     await tester.pumpWidget(const Technoprise());

     // Navigate to Add Item screen
     await tester.tap(find.byIcon(Icons.add));
     await tester.pumpAndSettle();

     // Add item title and description
     await tester.enterText(
         find.byKey(const ValueKey('titleField')), 'New Item');
     await tester.enterText(find.byKey(const ValueKey('descriptionField')),
         'This is a new item description.');

     // Tap Add Item button
     await tester.tap(find.byKey(const ValueKey('addItemButton')));
     await tester.pumpAndSettle();

     // Verify that the new item appears in the list
     expect(find.text('New Item'), findsOneWidget);
     expect(find.text('This is a new item description.'), findsOneWidget);
   });

   testWidgets('Delete existing item', (WidgetTester tester) async {
     // Launch the app
     await tester.pumpWidget(const Technoprise());

     // Ensure that at least one item exists to delete
     expect(find.text('Rainforest Ecosystems'), findsOneWidget);

     // Tap delete icon on the first item
     await tester.tap(find.byIcon(Icons.delete_outlined).first);
     await tester.pumpAndSettle();

     // Verify that the item is removed from the list
     expect(find.text('Rainforest Ecosystems'), findsNothing);
   });
 });
}
```

2. Test Report

Test Cases Covered:

- **Adding a New Item**
    - **Steps**:
        1. Launch the app.
        2. Navigate to the Add Item screen.
        3. Enter the title and description.
        4. Tap the "Add Item" button.
        5. Verify that the new item appears in the list.
- **Deleting an Existing Item**
    - **Steps**:
        1. Launch the app.

2. Ensure that at least one item exists.
3. Tap the delete icon for the item.
4. Verify that the item is removed from the list.

Results of Test Execution:

- Both test cases executed successfully without any errors.
- The items were added and deleted as expected.

Challenges Faced:

- **Flutter Integration Testing Setup**: Initially, there were issues with dependencies related to the integration testing framework. The issue was resolved by ensuring the correct version of the Flutter SDK was being used and that the `integration_test` package was properly included in `pubspec.yaml`.
- **Ambiguous Widget Finders**: In the initial implementation, multiple instances of the same widget were detected, leading to test failures. This was resolved by utilizing unique keys to precisely identify and interact with the intended widgets.