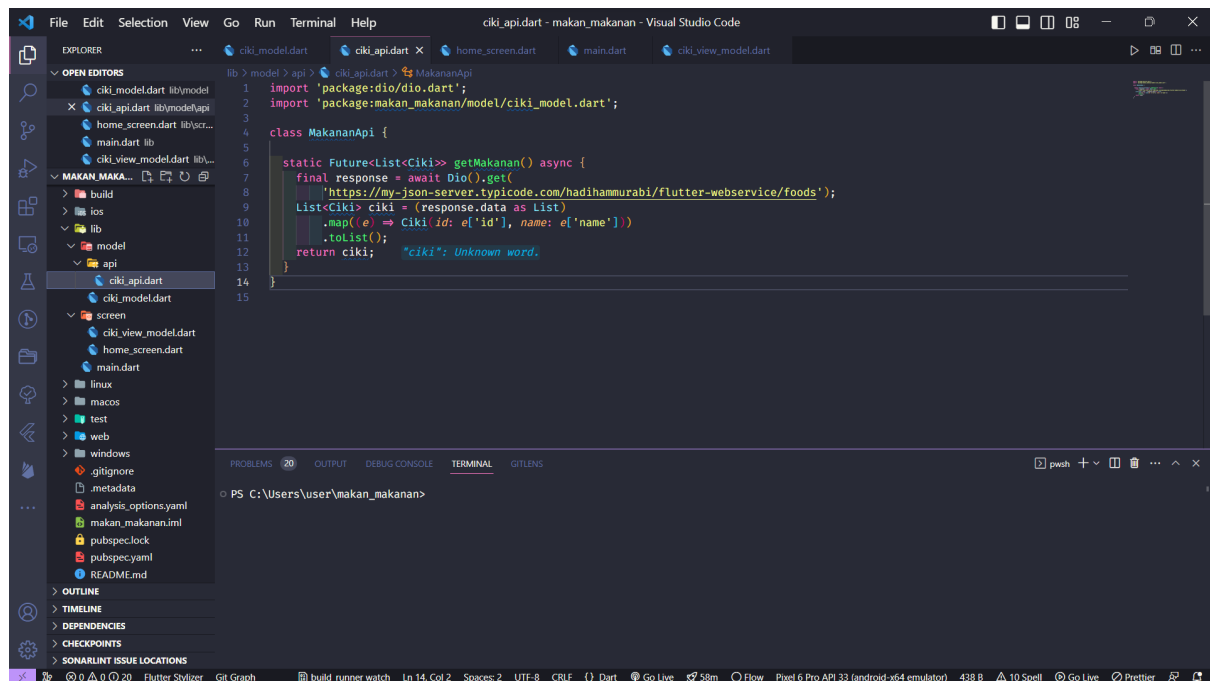


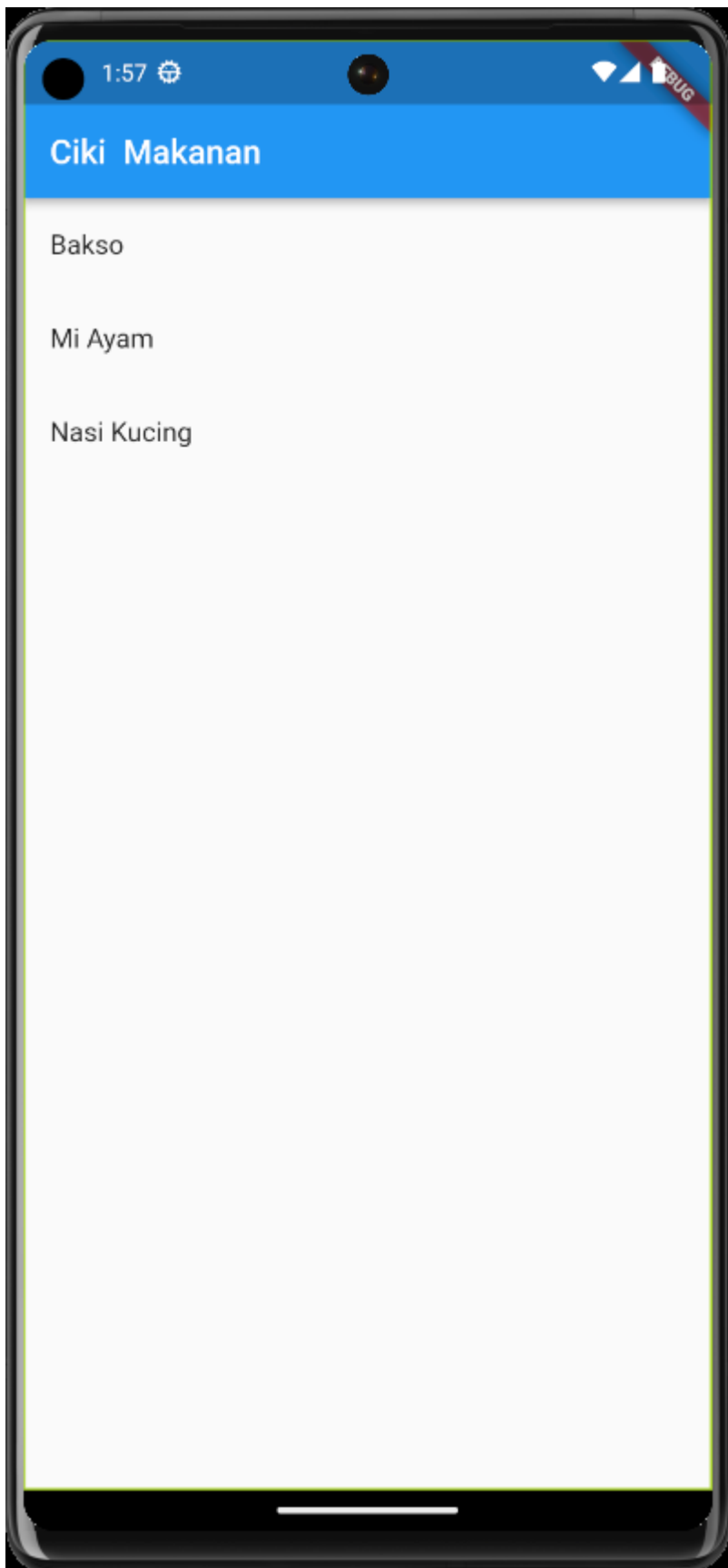
Nama : Mohd Ryan Obillah
Kelas : SIB 4 Flutter Kelas B

Soal Prioritas 1 (80)

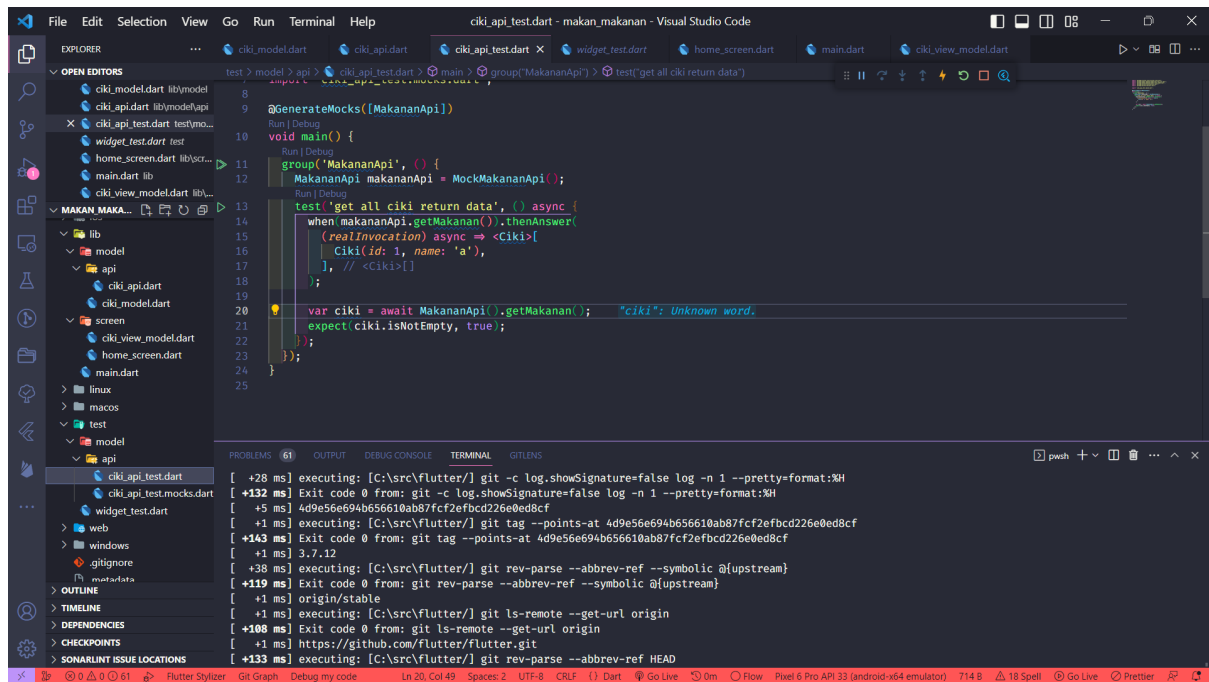
- Buatlah sebuah halaman baru lengkap dengan menggunakan MVVM untuk menampilkan data pada API dibawah ini
- <https://my-json-server.typicode.com/hadihammurabi/flutter-webservice/foods>
- Lakukanlah Testing pada model API diatas dengan menerapkan Mocking

Jawab :





Testing :



The screenshot shows a Visual Studio Code editor with a Dart project. The file explorer on the left shows a project structure with folders like 'lib', 'model', 'api', 'screen', and 'main'. The main editor displays the file 'ciki_test.dart' with the following code:

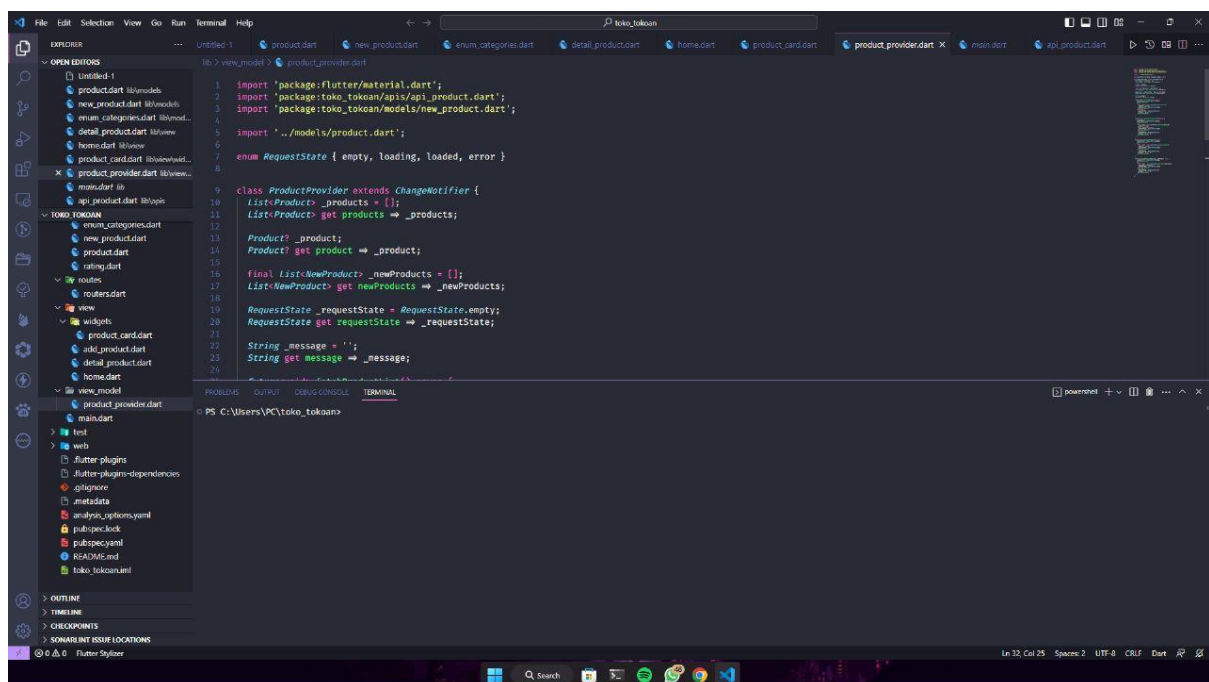
```
test('get all ciki return data', () async {
  @GenerateMocks([MakananApi])
  void main() {
    group('MakananApi', () {
      MakananApi makananApi = MockMakananApi();
      test('get all ciki return data', () async {
        when(makananApi.getMakanan()).thenReturn(
          (realInvocation) async => <Ciki>[
            Ciki(id: 1, name: 'a'),
          ], // <Ciki>[]
        );
        var ciki = await makananApi.getMakanan();
        expect(ciki.isNotEmpty, true);
      });
    });
  }
});
```

The terminal at the bottom shows the execution of the test, including the command 'flutter test' and the output of the test runner, which indicates that the test passed successfully.

Soal Prioritas 2 (20)

- Tambahkan pada project Contacts sebuah Enum state

Jawab :



The screenshot shows a Visual Studio Code editor with a Dart project. The file explorer on the left shows a project structure with folders like 'lib', 'model', 'api', 'screen', and 'main'. The main editor displays the file 'enum_categories.dart' with the following code:

```
enum RequestState { empty, loading, loaded, error }

class ProductProvider extends ChangeNotifier {
  List<Product> _products = [];
  List<Product> get products => _products;
  Product? _product;
  Product? get product => _product;
  final List<NewProduct> _newProducts = [];
  List<NewProduct> get newProducts => _newProducts;
  RequestState _requestState = RequestState.empty;
  RequestState get requestState => _requestState;
  String _message = '';
  String get message => _message;
}
```

The terminal at the bottom shows the execution of the test, including the command 'flutter test' and the output of the test runner, which indicates that the test passed successfully.

Soal Eksplorasi (20)

1. Lakukan unit testing pada aplikasi Contacts yang telah dikembangkan.

Jawab :

