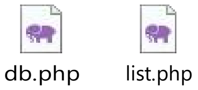
NIM : 223303030226
Nama : HABIBI

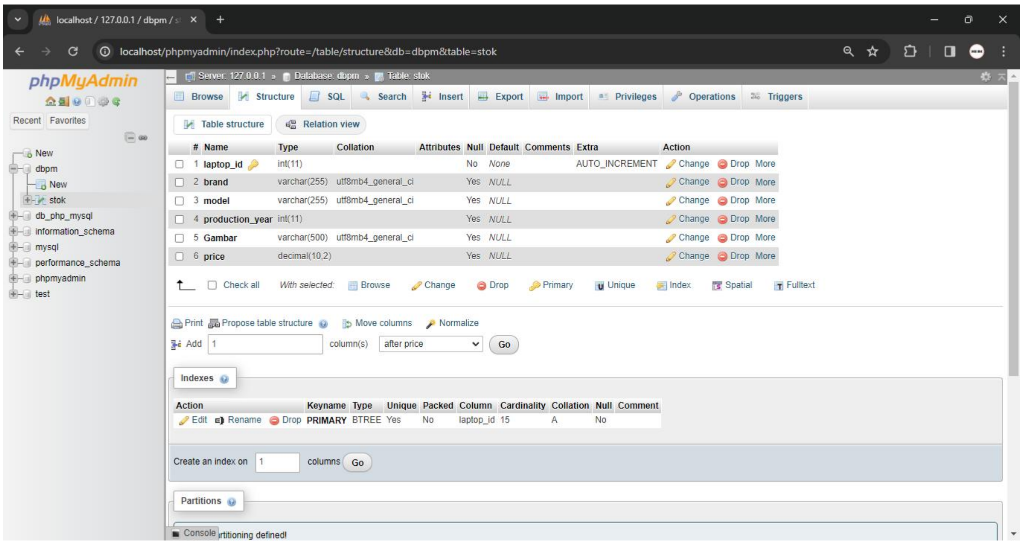
TUGAS PEMROGRAMAN MOBILE

1. Webservice, Database dan Table

a. Webservice yang digunakan (Fake atau bangun sendiri) : Bangun sendiri



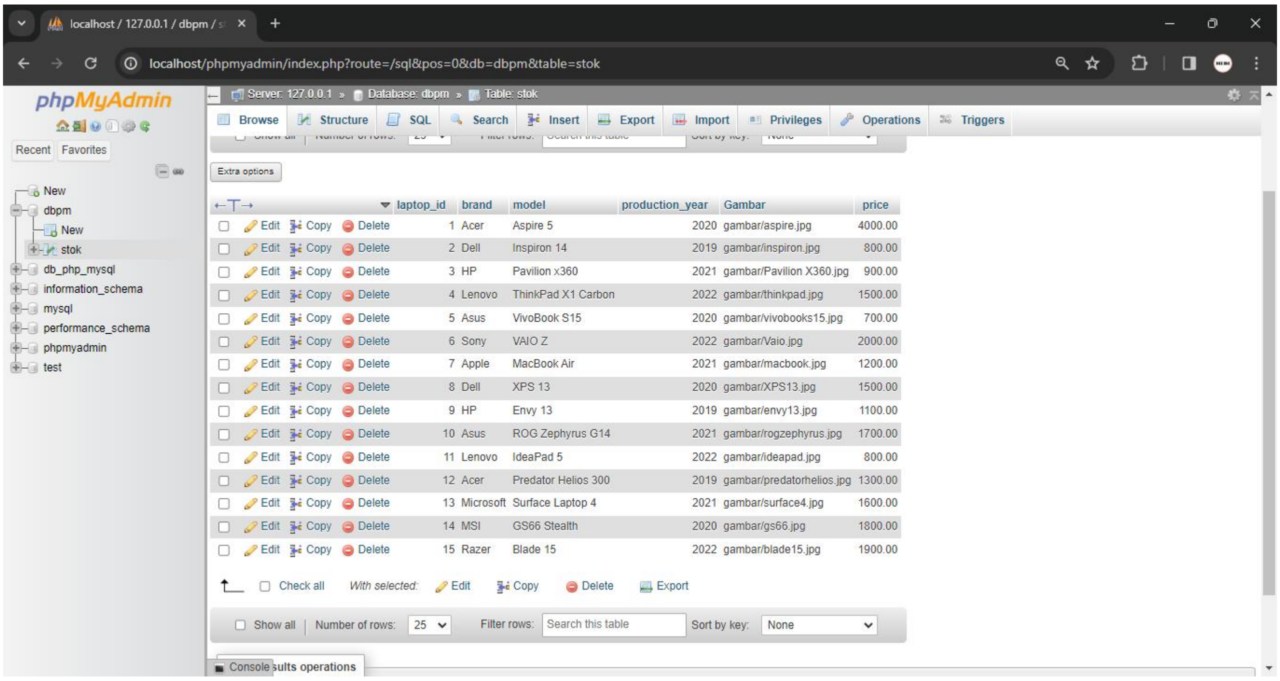
b. Database menggunakan (MySQL/SQLite) : (MySQL)



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	laptop_id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
□	2	brand	varchar(255)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
□	3	model	varchar(255)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
□	4	production_year	int(11)			Yes	*NULL*			Change Drop More
□	5	Gambar	varchar(500)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
□	6	price	decimal(10,2)			Yes	*NULL*			Change Drop More

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	laptop_id	15	A	No	

c. Nama table pada database : stok



←T→	laptop_id	brand	model	production_year	Gambar	price
□ Edit Copy Delete	1	Acer	Aspire 5	2020	gambar/aspire.jpg	4000.00
□ Edit Copy Delete	2	Dell	Inspiron 14	2019	gambar/inspiron.jpg	800.00
□ Edit Copy Delete	3	HP	Pavilion x360	2021	gambar/Pavilion X360.jpg	900.00
□ Edit Copy Delete	4	Lenovo	ThinkPad X1 Carbon	2022	gambar/thinkpad.jpg	1500.00
□ Edit Copy Delete	5	Asus	VivoBook S15	2020	gambar/vivobooks15.jpg	700.00
□ Edit Copy Delete	6	Sony	VAIO Z	2022	gambar/Vaio.jpg	2000.00
□ Edit Copy Delete	7	Apple	MacBook Air	2021	gambar/macbook.jpg	1200.00
□ Edit Copy Delete	8	Dell	XPS 13	2020	gambar/XPS13.jpg	1500.00
□ Edit Copy Delete	9	HP	Envy 13	2019	gambar/envy13.jpg	1100.00
□ Edit Copy Delete	10	Asus	ROG Zephyrus G14	2021	gambar/rogzephyrus.jpg	1700.00
□ Edit Copy Delete	11	Lenovo	IdeaPad 5	2022	gambar/ideapad.jpg	800.00
□ Edit Copy Delete	12	Acer	Predator Helios 300	2019	gambar/predatorhelios.jpg	1300.00
□ Edit Copy Delete	13	Microsoft	Surface Laptop 4	2021	gambar/surface4.jpg	1600.00
□ Edit Copy Delete	14	MSI	GS66 Stealth	2020	gambar/gs66.jpg	1800.00
□ Edit Copy Delete	15	Razer	Blade 15	2022	gambar/blade15.jpg	1900.00

NIM : 223303030226
Nama : HABIBI

2. **Isi dari file pada folder lib**

   d. **main.dart**

```
        import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
import 'model.dart';
import 'service.dart';

void main() => runApp(MyApp());

class MyApp extends StatefulWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  State<MyApp> createState() => _MyAppState();
}

class _MyAppState extends State<MyApp> {
  late List<stok> laptops;
  late List<stok> filteredLaptops;

  @override
  void initState() {
    super.initState();
    laptops = [];
    filteredLaptops = [];
    fetchLaptop(http.Client()).then((data) {
      setState(() {
        laptops = data;
      });
    });
  }

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        primarySwatch: Colors.blue,
        fontFamily: 'Roboto',
      ),
      home: Scaffold(
appBar: AppBar(
  title: Text(
    'My Laptop App',
    style: TextStyle(fontWeight: FontWeight.bold),
  ),
  centerTitle: true,
  backgroundColor: Colors.white,
  leading: Icon(Icons.laptop),
),
body: Container(
  decoration: BoxDecoration(
    gradient: LinearGradient(
      begin: Alignment.topLeft,
      end: Alignment.bottomRight,
      colors: [Colors.blue, Colors.white],
    ),
  ),
  child: Column(
    children: [
      Padding(
        padding: const EdgeInsets.all(8.0),
        child: TextField(
          onChanged: (value) {
            setState(() {
              filteredLaptops = filterLaptops(value);
            });
          },
          decoration: InputDecoration(
            labelText: 'Search',
            hintText: 'Enter laptop model...',
            prefixIcon: Icon(Icons.search),
            border: OutlineInputBorder(),
            focusedBorder: OutlineInputBorder(
              borderSide: BorderSide(color: Colors.white, width: 2.0),
              borderRadius: BorderRadius.circular(8.0),
            ),
            hintStyle: TextStyle(color: Colors.white),
          ),
        ),
      ),
      Expanded(
        child: LaptopList(
          Laptoplist: filteredLaptops.isNotEmpty
```

```
                      ? filteredLaptops
                      : laptops,
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }

  List<stok> filterLaptops(String query) {
    return laptops.where((laptop) {
      final modelLower = laptop.model.toLowerCase();
      final queryLower = query.toLowerCase();
      return modelLower.contains(queryLower);
    }).toList();
  }
}

class LaptopList extends StatelessWidget {
  final List<stok> Laptoplist;

  LaptopList({Key? key, required this.Laptoplist}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return ListView.builder(
      itemCount: Laptoplist.length,
      itemBuilder: (_, index) {
        var data = Laptoplist[index];
        return Card(
          elevation: 8.0,
          margin: EdgeInsets.all(8.0),
          shape: RoundedRectangleBorder(
            borderRadius: BorderRadius.circular(15.0),
          ),
          child: Container(
            decoration: BoxDecoration(
              gradient: LinearGradient(
                begin: Alignment.topLeft,
                end: Alignment.bottomRight,
                colors: [Colors.blueAccent, Colors.blue],
          ),
          borderRadius: BorderRadius.circular(15.0),
        ),
        child: ListTile(
          title: Text(
            data.model,
            style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold, color: Colors.white),
          ),
          subtitle: Text(
            '${data.brand} - ${data.production_year}',
            style: TextStyle(fontSize: 14, color: Colors.white),
          ),
          trailing: Icon(Icons.keyboard_arrow_right, color: Colors.white),
          onTap: () {
            Navigator.push(
              context,
              MaterialPageRoute(
                builder: (context) => DetailScreen(data: data),
              ),
            );
          },
        ),
      ),
    );
   },
  );
 }
}

class DetailScreen extends StatelessWidget {
  final stok data;

  DetailScreen({required this.data});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(
          'Product Details',
          style: TextStyle(fontWeight: FontWeight.bold),
        ),
        backgroundColor: Colors.blue,
      ),
      body: Container(
        decoration: BoxDecoration(
          gradient: LinearGradient(
            begin: Alignment.topLeft,
            end: Alignment.bottomRight,
            colors: [Colors.blue, Colors.white],
          ),
        ),
        padding: EdgeInsets.all(16.0),
        child: ListView(
          children: [
            Card(
              elevation: 5,
              margin: EdgeInsets.all(10),
              shape: RoundedRectangleBorder(
                borderRadius: BorderRadius.circular(15.0),
              ),
              child: Padding(
                padding: EdgeInsets.all(16.0),
                child: Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  children: [
                    Text(
                      'Model: ${data.model}',
                      style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold, color:
Colors.black),
                    ),
                    SizedBox(height: 16.0),
                    Text('Brand: ${data.brand}', style: TextStyle(fontSize: 18, color: Colors.black)),
                    Text('Production Year: ${data.production_year.toString()}',
                        style: TextStyle(fontSize: 18, color: Colors.black)),
                    Text('Price: \$${data.price.toString()}',
                        style: TextStyle(fontSize: 18, color: Colors.black)),
                  ],
                ),
              ),
            ),
            Card(
              elevation: 5,
              margin: EdgeInsets.all(10),
              shape: RoundedRectangleBorder(
                borderRadius: BorderRadius.circular(15.0),
              ),
              child: ClipRRect(
                borderRadius: BorderRadius.circular(15.0),
                child: Container(
                  height: 200,
                  width: double.infinity,
                  decoration: BoxDecoration(
                    image: DecorationImage(
                      image: NetworkImage(
                        'http://10.0.2.2/dbpm_api/${data.Gambar}',
                      ),
                      fit: BoxFit.cover,
                    ),
                  ),
                ),
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

main.dart

e. model.dart

```
            class stok {
final String laptop_id;
final String brand;
final String model;
final String production_year;
```

```
  final String Gambar;
  final String price;

  stok(
     {required this.laptop_id,
      required this.brand,
      required this.model,
      required this.production_year,
      required this.Gambar,
      required this.price
     });

  Map<String, dynamic> toJson() => {
    'laptop_id': laptop_id,
    'brand': brand,
    'model': model,
    'production_year': production_year,
    'Gambar': Gambar,
    'price': price
  };

  factory stok.fromJson(Map<String, dynamic> data) => stok(
     laptop_id: data['laptop_id'].toString(),
     brand: data['brand'].toString(),
     model: data['model'].toString(),
     production_year: data['production_year'].toString(),
     Gambar: data['Gambar'].toString(),
     price: data['price'].toString());
}
```

model.dart

f. service.dart

```
import 'dart:convert';

import 'package:flutter/foundation.dart';

import 'model.dart';
import 'package:http/http.dart' as http;

const url = 'http://10.0.2.2/dbpm_api/';

Future<List<stok>> fetchLaptop(http.Client client) async {
  final response = await client.get(Uri.parse('${url}list.php'));
  //debugPrint(response.body);
  return compute(parseLaptop, response.body);
}

List<stok> parseLaptop(String responseBody){
  final parsed = jsonDecode(responseBody).cast<Map<String, dynamic>>();
  return parsed.map<stok>((json)=>stok.fromJson(json)).toList();
}
```

service.dart

g. pubspec.yaml

```
name: tugaspm
description: "A new Flutter project."
# The following line prevents the package from being accidentally published to
# pub.dev using `flutter pub publish`. This is preferred for private packages.
publish_to: 'none' # Remove this line if you wish to publish to pub.dev

# The following defines the version and build number for your application.
# A version number is three numbers separated by dots, like 1.2.43
# followed by an optional build number separated by a +.
# Both the version and the builder number may be overridden in flutter
# build by specifying --build-name and --build-number, respectively.
# In Android, build-name is used as versionName while build-number used as versionCode.
# Read more about Android versioning at
https://developer.android.com/studio/publish/versioning
# In iOS, build-name is used as CFBundleShortVersionString while build-number is used as
CFBundleVersion.
# Read more about iOS versioning at
#
https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyRe
ference/Articles/CoreFoundationKeys.html
# In Windows, build-name is used as the major, minor, and patch parts
# of the product and file versions while build-number is used as the build suffix.
```

NIM : 223303030226
Nama : HABIBI

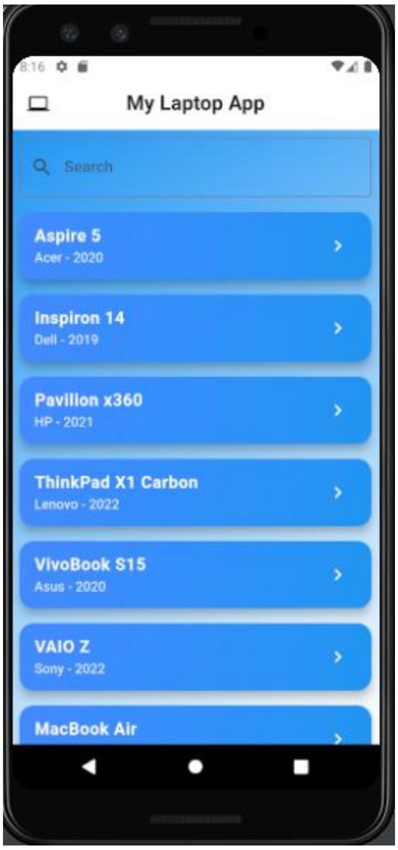
```
version: 1.0.0+1

environment:
  sdk: '>=3.2.4 <4.0.0'

# Dependencies specify other packages that your package needs in order to work.
# To automatically upgrade your package dependencies to the latest versions
# consider running `flutter pub upgrade --major-versions`. Alternatively,
# dependencies can be manually updated by changing the version numbers below to
# the latest version available on pub.dev. To see which dependencies have newer
# versions available, run `flutter pub outdated`.
dependencies:
  flutter:
    sdk: flutter
  http: ^1.1.2

  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^1.0.2

dev_dependencies:
  flutter_test:
    sdk: flutter

  # The "flutter_lints" package below contains a set of recommended lints to
  # encourage good coding practices. The lint set provided by the package is
  # activated in the `analysis_options.yaml` file located at the root of your
  # package. See that file for information about deactivating specific lint
  # rules and activating additional ones.
  flutter_lints: ^2.0.0

# For information on the generic Dart part of this file, see the
# following page: https://dart.dev/tools/pub/pubspec

# The following section is specific to Flutter packages.
flutter:

  # The following line ensures that the Material Icons font is
  # included with your application, so that you can use the icons in
  # the material Icons class.
  uses-material-design: true

  # To add assets to your application, add an assets section, like this:
```

```
# assets:
#   - images/a_dot_burr.jpeg
#   - images/a_dot_ham.jpeg

# An image asset can refer to one or more resolution-specific "variants", see
# https://flutter.dev/assets-and-images/#resolution-aware

# For details regarding adding assets from package dependencies, see
# https://flutter.dev/assets-and-images/#from-packages

# To add custom fonts to your application, add a fonts section here,
# in this "flutter" section. Each entry in this list should have a
# "family" key with the font family name, and a "fonts" key with a
# list giving the asset and other descriptors for the font. For
# example:
# fonts:
#   - family: Schyler
#     fonts:
#       - asset: fonts/Schyler-Regular.ttf
#       - asset: fonts/Schyler-Italic.ttf
#         style: italic
#   - family: Trajan Pro
#     fonts:
#       - asset: fonts/TrajanPro.ttf
#       - asset: fonts/TrajanPro_Bold.ttf
#         weight: 700
#
# For details regarding fonts from package dependencies,
# see https://flutter.dev/custom-fonts/#from-packages
```

NIM : 223303030226
Nama : HABIBI

3. Ouput dari aplikasi

a. Gambar 1



My Laptop App

Search

Aspire 5
Acer - 2020

Inspiron 14
Dell - 2019

Pavilion x360
HP - 2021

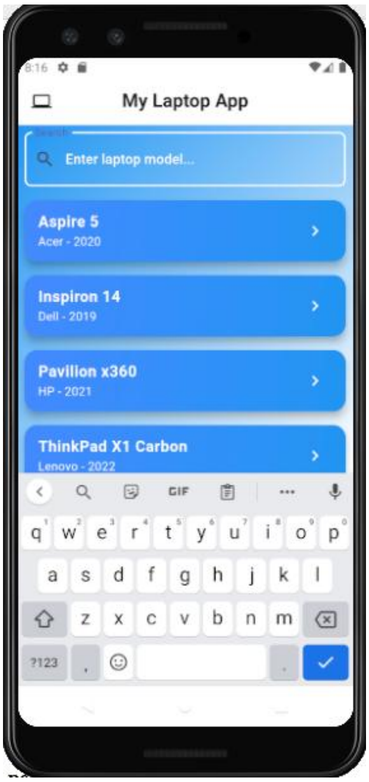
ThinkPad X1 Carbon
Lenovo - 2022

VivoBook S15
Asus - 2020

VAIO Z
Sony - 2022

MacBook Air

b. Gambar 2



My Laptop App

Search

Enter laptop model...

Aspire 5
Acer - 2020

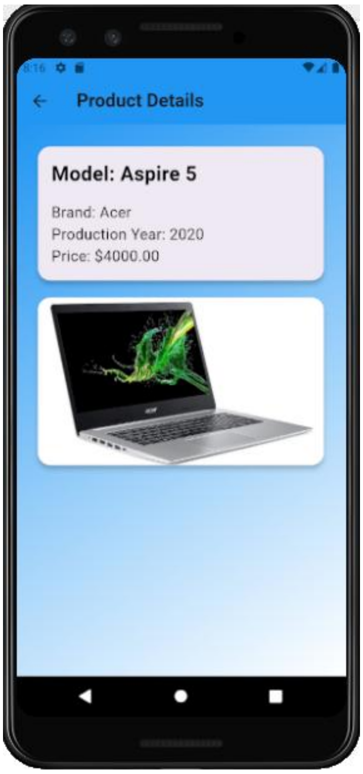
Inspiron 14
Dell - 2019

Pavilion x360
HP - 2021

ThinkPad X1 Carbon
Lenovo - 2022

NIM : 223303030226
Nama : HABIBI

a. Gambar 3



Model: Aspire 5

Brand: Acer
Production Year: 2020
Price: $4000.00

4. **Link channel Youtube berisi penjelasan dan penggunaan aplikasi**
   Link : https://youtu.be/tdwZkYoIKp0?si=F4TzTMyKRi1sYtM3

Catatan:

- *Mohon masing-masing mahasiswa/i mengikuti template ini.*
- *Jika anda tidak mengikuti format template ini, maka akan menyulitkan proses penilaian dan akan mengurangi penilaian anda.*