

CODINGAN PROGRAM DETEKSI BATIK

Lampiran 1

Features Dashboard

```
// ignore_for_file: library_private_types_in_public_api

import 'package:batik_detection/features/detail/list_batik_sample.dart';

import 'package:batik_detection/features/list_batik/list_batik.dart';

import 'package:batik_detection/features/scan_image/take_photo_scan.dart';

import 'package:batik_detection/widget/custom_card.dart';

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

import 'package:batik_detection/data/model/global.dart';

import 'package:batik_detection/services/locator.dart';

import 'package:batik_detection/util/appTheme.dart';

import 'package:batik_detection/util/base_state.dart';

import 'package:batik_detection/util/preferences.dart';

import 'package:batik_detection/util/utility.dart';

import 'package:batik_detection/widget/ui_spacing.dart';

import 'package:rounded_loading_button/rounded_loading_button.dart';

class DashboardPage extends StatefulWidget {

  const DashboardPage({Key? key}) : super(key: key);

  static const String route = '/dashboard';

  @override

  DashboardPageState createState() => DashboardPageState();

}
```

```
class DashboardPageState extends BaseState<DashboardPage> {  
  
    final RoundedLoadingButtonController buttonController=  
        RoundedLoadingButtonController();  
  
    var visible = true;  
    var global = locator<Global>();  
  
    String flag = 'ic_flagid';  
  
    @override  
    void initState() {  
        super.initState();  
        setLanguage();  
        Preferences.getId(context);  
    }  
  
    setLanguage() async {  
        locator<Global>().languageId = await Prefs.getLanguage;  
        setState() {  
            if (global.languageId == 'japan') {  
                flag = "ic_flagjapan";  
            } else if (global.languageId == 'en') {  
                flag = "ic_flagen";  
            } else {  
                flag = "ic_flagid";  
            }  
        });  
    }  
}
```

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: themeData().backgroundColor,
    body: bodyView(),
  );
}
```

```
@override
Widget bodyView() {
  return Scaffold(
    backgroundColor: AppTheme.white,
    body: SizedBox(
      height: heightLayer(),
      width: widthLayer(),
      child: Stack(
        children: [
          Positioned(
            top: 0,
            right: 0,
            left: 0,
            bottom: 0,
            child: Image.asset(
              imagePathJpg('background.jpg'),
              fit: BoxFit.fill,
            ),
          ),
        ],
      ),
    ),
  );
}
```

```
Positioned(
  top: 0,
  right: 24,
  left: 24,
  bottom: 0,
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      // Text(
      // 'MENU UTAMA',
      // style: Theme.of(context)
      //   .textTheme
      //   .titleLarge!,
      //   .copyWith(fontSize: 32),
      // ),
      // const VerticalSpacing(
      //   height: 200,
      // ),
      Row(
        mainAxisAlignment: MainAxisAlignment.spaceBetween,
        children: [
          InkWell(
            onTap: () {
              navigationTo(TakePhotoScanPage.route, {});
            },
            child: const CustomCard(
              width: 150,
              height: 100,
```

```
        iconSize: 50,
        borderColor: AppTheme.chartRed,
        valueColor: AppTheme.black,
        value: CupertinoIcons.search,
        title: 'Scan Batik',
    ),
),
InkWell(
    onTap: () {
        navigationTo(ListBatikSamplePage.route, {});
    },
    child: const CustomCard(
        width: 150,
        height: 100,
        iconSize: 50,
        borderColor: AppTheme.chartRed,
        valueColor: AppTheme.black,
        value: CupertinoIcons.list_bullet_below_rectangle,
        title: 'List Batik',
    ),
),
],
),
const VerticalSpacing(
    height: 50,
),
InkWell(
    onTap: () {
```

```
        navigationTo(ListBatikPage.route, {});
    },
    child: const CustomCard(
      width: 150,
      height: 100,
      iconSize: 50,
      borderColor: AppTheme.chartRed,
      valueColor: AppTheme.black,
      value: CupertinoIcons.camera,
      title: 'Take Photo Batik',
    ),
  ),
],
),
),
],
),
),
);
}
```

```
@override
Widget footerView() {
  return Container();
}
```

```
@override
Widget bloc() {
```

```
    throw UnimplementedError();
```

```
  }
```

```
}
```

Lampiran 2

Feature Detail

```
import 'package:batik_detection/widget/custom_button.dart';
import 'package:batik_detection/widget/ui_spacing.dart';
import 'package:flutter/material.dart';
import 'package:batik_detection/util/appTheme.dart';
import 'package:batik_detection/util/base_state.dart';
import 'dart:convert';
import 'package:flutter/services.dart';
import 'package:url_launcher/url_launcher.dart';

class DetailPage extends StatefulWidget {
  const DetailPage({Key? key, required this.name}): super(key: key);
  static const String route = '/detail';
  final String name;
  @override
  DetailPageState createState() => DetailPageState();
}

class DetailPageState extends BaseState<DetailPage> {
  List items = [];
  // ignore: prefer_typing_uninitialized_variables
  var dataItem;

  @override
  void initState() {
    super.initState();
    readJson();
  }
}
```



```
}
```

```
Future<void> readJson() async {  
  
  final String response =  
    await rootBundle.loadString('assets/data_batik.json');  
  final data = await json.decode(response);  
  setState() {  
    items = data["data"];  
    dataItem = items  
      .where((element) => element['name']  
        .toString()  
        .toLowerCase()  
        .contains(widget.name.toLowerCase()))  
      .first;  
  });  
}
```

```
@override
```

```
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      title: const Text('Detail Batik'),  
    ),  
    backgroundColor: themeData().backgroundColor,  
    body: bodyView(),  
  );  
}
```

```
@override
Widget bodyView() {
  return Scaffold(
    backgroundColor: AppTheme.white,
    body: Container(
      padding: const EdgeInsets.symmetric(horizontal: 24),
      height: heightLayer(),
      width: widthLayer(),
      child: SingleChildScrollView(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.start,
          children: [
            const VerticalSpacing(
              height: 20,
            ),
            Container(
              width: 300,
              height: 200,
              color: AppTheme.black,
              margin: const EdgeInsets.symmetric(vertical: 10),
              child: Image.asset(
                imagePathJpg(dataItem['image']),
                fit: BoxFit.fill,
              ),
            ),
            const VerticalSpacing(),
            Text(
              dataItem['name'],
```

```
        style: const TextStyle(fontSize: 30),
      ),
      const VerticalSpacing(
        height: 50,
      ),
      Text(
        dataItem['description'],
        style: const TextStyle(fontSize: 18),
        textAlign: TextAlign.justify,
      ),
      const VerticalSpacing(
        height: 50,
      ),
      Text(
        dataItem['price'],
        style: const TextStyle(
          fontSize: 18,
        ),
      ),
      const VerticalSpacing(
        height: 50,
      ),
      CustomButtonRounded(
        text: 'Cari Produk',
        onPressed: () async {
          final Uri url = Uri.parse(
            // ignore: prefer_interpolation_to_compose_strings
            'https://shopee.co.id/search?keyword=' +
```

```
        dataItem['name'];
        if (!await launchUrl(url)) {
            throw Exception('Could not launch $url');
        }
    },
),
],
),
),
),
);
}
```

```
@override
Widget footerView() {
    return Container();
}
```

```
@override
Widget bloc() {
    throw UnimplementedError();
}
}
```

Lampiran 3

Feature List Batik

```
import 'package:batik_detection/features/take_photo/take_photo.dart';
```

```
import 'package:batik_detection/util/appTheme.dart';
```

```
import 'package:batik_detection/util/base_state.dart';
```

```
import 'package:flutter/material.dart';
```

```
class ListBatikPage extends StatefulWidget {
```

```
  const ListBatikPage({Key? key}) : super(key: key);
```

```
  static const String route = '/list_batik';
```

```
  @override
```

```
  ListBatikPageState createState() => ListBatikPageState();
```

```
}
```

```
class ListBatikPageState extends BaseState<ListBatikPage> {
```

```
  @override
```

```
  void initState() {
```

```
    super.initState();
```

```
  }
```

```
  @override
```

```
  Widget build(BuildContext context) {
```

```
    return Scaffold(
```

```
      backgroundColor: themeData.backgroundColor,
```

```
      body: bodyView(),
```

```
    );
```

```
  }
```

```
@override  
Widget bloc() {  
  return Container();  
}
```

```
@override  
Widget bodyView() {  
  return Scaffold(  
    appBar: AppBar(  
      title: const Text('Jenis Model Baju Batik'),  
    ),  
    body: SizedBox(  
      height: heightLayer(),  
      width: widthLayer(),  
      child: SingleChildScrollView(  
        child: Column(  
          children: List.generate(10, (index) {  
            return InkWell(  
              onTap: () {  
                navigationTo(TakePhotoPage.route, [index + 1]);  
              },  
              child: Container(  
                color: const Color.fromARGB(255, 1, 143, 244),  
                margin:  
                  const EdgeInsets.symmetric(vertical: 10, horizontal: 24),  
                child: Image.asset(imagePath('batik_${index + 1}')),  
              ),  
            ),  
          ),  
        ),  
      ),  
    ),  
  ),  
);
```

```
);  
  }  
),  
),  
),  
);  
}
```

```
@override  
Widget footerView() {  
  return Container();  
}  
}
```

Lampiran 4

Feature Scean Batik

```
import 'dart:io';
import 'dart:ui';

import 'package:batik_detection/features/detail/detail.dart';
import 'package:batik_detection/util/app_extensions.dart';
import 'package:batik_detection/widget/custom_button.dart';
import 'package:flutter/material.dart';
import 'package:batik_detection/util/appTheme.dart';
import 'package:batik_detection/util/base_state.dart';
import 'package:rounded_loading_button/rounded_loading_button.dart';
import 'package:camera/camera.dart';
import 'package:flutter/services.dart';
import 'package:google_mlkit_image_labeling/google_mlkit_image_labeling.dart';
import 'package:path/path.dart' as paths;
import 'package:path_provider/path_provider.dart';

class TakePhotoScanPage extends StatefulWidget {
  const TakePhotoScanPage({
    Key? key,
  }) : super(key: key);
  static const String route = '/take_photo_scan';
  @override
  TakePhotoScanPageState createState() => TakePhotoScanPageState();
}

class TakePhotoScanPageState extends BaseState<TakePhotoScanPage> {
```



```
final RoundedLoadingButtonController buttonController=  
    RoundedLoadingButtonController();
```

```
late CameraController _controller;
```

```
late ImageLabeler _imageLabeler;
```

```
List<CameraDescription> cameras = [];
```

```
bool busy = false;
```

```
String text = "";
```

```
XFile? image;
```

```
@override
```

```
void initState() {
```

```
    super.initState();
```

```
    _initializeLabeler();
```

```
}
```

```
void _initializeLabeler() async {
```

```
    const path = 'assets/model_unquant.tflite';
```

```
    final modelPath = await _getModel(path);
```

```
    final options = LocalLabelerOptions(  
        modelPath: modelPath,
```

```
        confidenceThreshold: 0.7,
```

```
    );
```

```
    _imageLabeler = ImageLabeler(options: options);
```

```
}
```

```
Future<String> _getModel(String assetPath) async {
```

```
    if (Platform.isAndroid) {
```

```

    return 'flutter_assets/$assetPath';
}

final path = '${(await getApplicationSupportDirectory()).path}/$assetPath';

await Directory(paths.dirname(path)).create(recursive: true);

final file = File(path);

if (!await file.exists()) {

    final byteData = await rootBundle.load(assetPath);

    await file.writeAsBytes(byteData.buffer
        .asUint8List(byteData.offsetInBytes, byteData.lengthInBytes));
}

return file.path;
}

```

```

Future<void> processImage(InputImage inputImage) async {

    try {

        text = 'Batik tidak ditemukan';

        final labels = await _imageLabeler.processImage(inputImage);

        text = "";

        for (final label in labels) {

            text = label.label;

        }

        // ignore: use_build_context_synchronously

        await Navigator.of(context).push(

            MaterialPageRoute(

                builder: (context) => DisplayPictureScan(

                    imagePath: image!.path,

                    text: text,

                ),
            ),

```

```
    ),
    );
} catch (e) {
    appLog(e);
}
}
```

```
_processCameraImage(CameraImage image) {
    if (!busy) {
        busy = true;
        final inputImage = _inputImageFromCameraImage(image);
        if (inputImage != null) {
            processImage(inputImage);
        }
    }
}
```

```
InputImage? _inputImageFromCameraImage(CameraImage image) {
    // get camera rotation
    final camera = cameras[0];
    final rotation =
        InputImageRotationValue.fromRawValue(camera.sensorOrientation);
    if (rotation == null) return null;

    // get image format
    final format = InputImageFormatValue.fromRawValue(image.format.raw);
    // validate format depending on platform
    // only supported formats:
```

```

// * nv21 for Android

// * bgra8888 for iOS

if (format == null ||
    (Platform.isAndroid && format != InputImageFormat.nv21) ||
    (Platform.isIOS && format != InputImageFormat.bgra8888)) return null;

// since format is constraint to nv21 or bgra8888, both only have one plane
if (image.planes.length != 1) return null;
final plane = image.planes.first;

// compose InputImage using bytes
return InputImage.fromBytes(
    bytes: plane.bytes,
    metadata: InputImageMetadata(
        size: Size(image.width.toDouble(), image.height.toDouble()),
        rotation: rotation, // used only in Android
        format: format, // used only in iOS
        bytesPerRow: plane.bytesPerRow, // used only in iOS
    ),
);
}

```

```

Future<void> initializeCamera() async {
    cameras = await availableCameras();
    _controller = CameraController(cameras[0], ResolutionPreset.max,
        imageFormatGroup: ImageFormatGroup.nv21);
    _controller.setFlashMode(FlashMode.off);
    await _controller.initialize();
}

```

```
}
```

```
@override
```

```
void dispose() {  
  _controller.dispose();  
  super.dispose();  
}
```

```
@override
```

```
Widget build(BuildContext context) {  
  return Scaffold(  
    backgroundColor: themeData().backgroundColor,  
    body: bodyView(),  
  );  
}
```

```
@override
```

```
Widget bodyView() {  
  return SafeArea(  
    child: Scaffold(  
      appBar: AppBar(title: const Text('Photo')),  
      backgroundColor: AppTheme.black,  
      body: SizedBox(  
        width: widthLayer(),  
        height: heightLayer(),  
        child: FutureBuilder<void>(  
          future: initializeCamera(),  
          builder: (context, snapshot) {
```

```

if (snapshot.connectionState == ConnectionState.done) {
  return Stack(
    children: [
      Positioned(
        top: 0,
        bottom: 0,
        right: 0,
        left: 0,
        child: CameraPreview(_controller),
      ),
      Positioned(
        top: 150,
        bottom: -100,
        right: -70,
        left: -70,
        child: Container(),
      ),
    ],
  );
} else {
  return const Center(child: CircularProgressIndicator());
}
),
),
floatingActionButton: FloatingActionButton(
  onPressed: () async {
    try {
      busy = false;
    }
  }
)

```

```
        image = await _controller.takePicture();
        _controller.startImageStream(_processCameraImage);
    } catch (e) {
        appLog(e);
    }
},
child: const Icon(Icons.camera_alt),
),
),
);
}
```

```
@override
Widget footerView() {
    return Container();
}
```

```
@override
Widget bloc() {
    throw UnimplementedError();
}
}
```

```
class DisplayPictureScan extends StatelessWidget {
    final String imagePath;
    final String text;
    const DisplayPictureScan(
        {super.key, required this.imagePath, required this.text});
```

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: const Text('Photo')),
    body: Stack(
      children: [
        Positioned(
          top: 0,
          left: 0,
          right: 0,
          bottom: 0,
          child: SizedBox(
            height: MediaQuery.of(context).size.height,
            width: MediaQuery.of(context).size.width,
            child: Image.file(
              fit: BoxFit.fill,
              File(imagePath),
            ),
          ),
        ),
        Positioned(
          bottom: 120,
          left: 0,
          right: 0,
          child: Center(
            child: Text(
              text,
```



```
        style: const TextStyle(fontSize: 32, color: AppTheme.white),
      ),
    ),
  ),
  Positioned(
    bottom: 50,
    left: 0,
    right: 0,
    child: CustomButtonRounded(
      text: 'Lihat Detail',
      onPressed: () async {
        Navigator.pushNamed(context, DetailPage.route,
          arguments: [text]);
      },
    ),
  )
],
),
);
}
}
```

Lampiran 5

Feature Take Foto Display Picture

```
import 'dart:io';

import 'dart:typed_data';

import 'package:batik_detection/widget/custom_button.dart';

import 'package:esys_flutter_share_plus/esys_flutter_share_plus.dart';

import 'package:flutter/material.dart';

import 'package:flutter/rendering.dart';

import 'package:gallery_saver/gallery_saver.dart';

import 'package:path_provider/path_provider.dart';

import 'dart:ui' as ui;

class DisplayPictureScreen extends StatelessWidget {

  final String imagePath;

  final int index;

  const DisplayPictureScreen(

    {super.key, required this.imagePath, required this.index });

  @override

  Widget build(BuildContext context) {

    var repaintKey = GlobalKey<ScaffoldState>();

    var globalKey = GlobalKey<ScaffoldState>();

    return Scaffold(

      key: globalKey,

      appBar: AppBar(title: const Text('Photo')),

      body: Stack(

        children: [

          Positioned(
```

```
top: 0,
left: 0,
right: 0,
bottom: 0,
child: RepaintBoundary(
  key: repaintKey,
  child: Stack(
    children: [
      Positioned(
        top: 0,
        left: 0,
        right: 0,
        bottom: 0,
        child: SizedBox(
          height: MediaQuery.of(context).size.height,
          width: MediaQuery.of(context).size.width,
          child: Image.file(
            fit: BoxFit.fill,
            File(imagePath),
          ),
        ),
      ),
    ],
  ),
  Positioned(
    top: 150,
    bottom: -100,
    right: -70,
    left: -70,
    child: Container(
```

```

margin: const EdgeInsets.symmetric(vertical: 10),

child: Image.asset(
  'assets/images/batik_${index}.PNG',
  fit: BoxFit.fill,
),
),
),
],
),
),
),
),
Positioned(
  bottom: 30,
  left: 10,
  right: 10,
  child: Row(
    mainAxisAlignment: MainAxisAlignment.spaceBetween,
    children: [
      CustomButtonRounded(
        width: 100,
        text: 'Simpan',
        onPressed: () async {
          RenderRepaintBoundary? renderRepaintBoundary =
            repaintKey.currentContext?.findRenderObject()
              as RenderRepaintBoundary?;

          double pixelRatioValue = 3.0;
          if (Platform.isIOS) {

```

```

    pixelRatioValue = 2.0;
}

ui.Image image = await renderRepaintBoundary!
    .toImage(pixelRatio: pixelRatioValue);
ByteData? byteData =
    await image.toByteData(format: ui.ImageByteFormat.png);
Uint8List uint8list = byteData!.buffer.asUint8List();

finalString path =
    (await getApplicationDocumentsDirectory()).path;
File imageFile =
    File('$path/${DateTime.now().toIso8601String()}.png');
imageFile.writeAsBytesSync(uint8list);
GallerySaver.saveImage(imageFile.path).then(
    (value) => showDialog<String>(
        context: context,
        builder: (BuildContext context) => Dialog(
            child: Padding(
                padding: const EdgeInsets.all(8.0),
                child: Column(
                    mainAxisAlignment: MainAxisAlignment.min,
                    mainAxisAlignment: MainAxisAlignment.center,
                    children: <Widget>[
                        const Text('Gambar Berhasil di Simpan'),
                        const SizedBox(height: 15),
                        TextButton(
                            onPressed: () {

```

```

        Navigator.pop(context);
    },
    child: const Text('Tutup'),
  ),
],
),
),
),
),
);
},
),
CustomButtonRounded(
  width: 100,
  text: 'Bagikan',
  onPressed: () async {
    RenderRepaintBoundary? renderRepaintBoundary =
      repaintKey.currentContext?.findRenderObject()
        as RenderRepaintBoundary?;

    double pixelRatioValue = 3.0;
    if (Platform.isIOS) {
      pixelRatioValue = 2.0;
    }

    ui.Image image = await renderRepaintBoundary!
      .toImage(pixelRatio: pixelRatioValue);
    ByteData? byteData =

```

```
        await image.toByteArray(format: ui.ImageByteFormat.png);
    Uint8List uint8list = byteData!.buffer.asUint8List();
    await Share.file(
        'Share Image',
        '${DateTime.now().toIso8601String()}.png',
        uint8list,
        '*/*');
    },
    ),
    ],
    ),
    )
    ],
    ),
    );
}
}
```