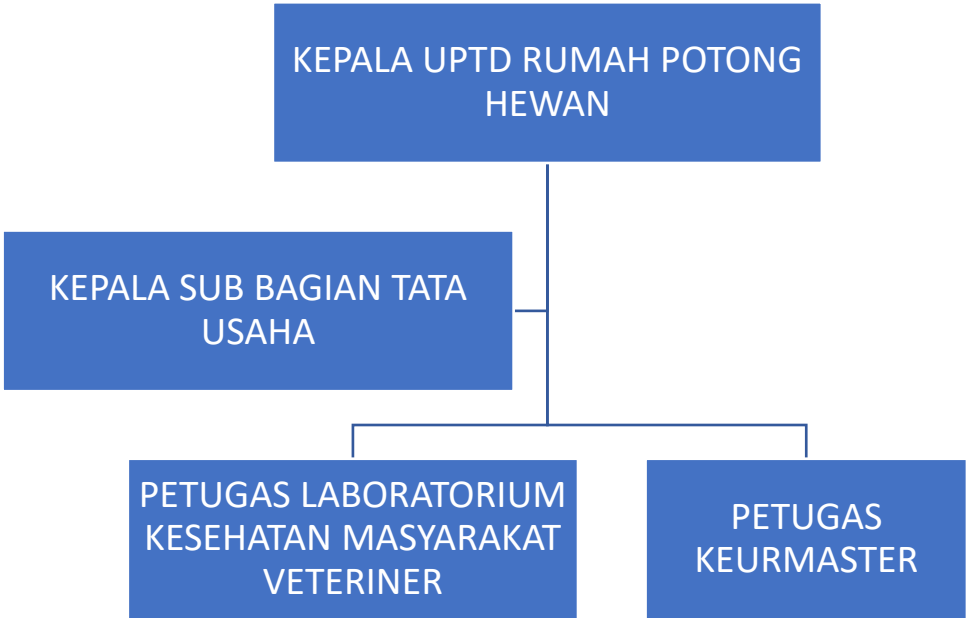


LAMPIRAN

UPTD RUMAH PTONG HEWAN PAREPARE
LAPORAN PEMOTONGAN BULAN JULI 2024

Tanggal	Jenis Sapi	Berat (kg)	Kondisi	Asal	Nama Pemotong
01/07/2024	Sapi Angus	350	Sehat	Barru	Lukman Hasan
04/07/2024	Sapi Bali	220	Sehat	Sidrap	Lukman Muklis
08/07/2024	Sapi Madura	280	Sehat	Enrekang	Hatta
11/07/2024	Sapi Angus	420	Sehat	Parepare	Bambang
15/07/2024	Sapi Bali	390	Sehat	Barru	Lukman Hasan
18/07/2024	Sapi Madura	300	Sehat	Sidrap	Lukman Muklis
22/07/2024	Sapi Angus	310	Sehat	Enrekang	Hatta
25/07/2024	Sapi Bali	350	Sehat	Parepare	Bambang
29/07/2024	Sapi Madura	420	Sehat	Barru	Lukman Hasan
01/08/2024	Sapi Angus	480	Sehat	Sidrap	Lukman Muklis

STRUKTUR KEPEGAWAIAN UPTD RPH PAREPARE



```

Login.dart
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import
    'package:flutter_firebase/features/user_auth/presentation/pages/sign_up_page.dart';
import
    'package:flutter_firebase/features/user_auth/presentation/widgets/form_container_widget.dart';
import 'package:flutter_firebase/global/common/toast.dart';

// import 'package:google_sign_in/google_sign_in.dart';

import '../firebase_auth_implementation/firebase_auth_services.dart';

class LoginPage extends StatefulWidget {
  const LoginPage({super.key});

  @override
  State<LoginPage> createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  bool _isSigning = false;
  final FirebaseAuthService _auth = FirebaseAuthService();
  final FirebaseAuth _firebaseAuth = FirebaseAuth.instance;
  TextEditingController _emailController = TextEditingController();
  TextEditingController _passwordController = TextEditingController();

  @override
  void dispose() {
    _emailController.dispose();
    _passwordController.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {

```

```
return Scaffold(  
  appBar: AppBar(  
    automaticallyImplyLeading: false,  
    title: Text(""),  
  ),  
  body: Center(  
    child: Padding(  
      padding: const EdgeInsets.symmetric(horizontal: 15),  
      child: Column(  
        mainAxisAlignment: MainAxisAlignment.center,  
        children: [  
  
          Spacer(flex: 1,),  
          Image.asset(  
            'assets/images/rphlogo.png',  
            width: 200,  
            height: 200,  
  
          ),  
          Text(  
            "Login To MYRPH",  
            style: TextStyle(fontSize: 27, fontWeight: FontWeight.bold),  
          ),  
  
          SizedBox(  
            height: 30,  
          ),  
          FormContainerWidget(  
            controller: _emailController,  
            hintText: "Email",  
            isPasswordField: false,  
          ),  
          SizedBox(  
            height: 10,  
          ),  
          FormContainerWidget(  
            controller: _passwordController,  
            hintText: "Password",  
            isPasswordField: true,  
          ),  
        ],  
      ),  
    ),  
  ),  
);
```

```

    SizedBox(
      height: 30,
    ),
    GestureDetector(
      onTap: () {
        _signIn();
      },
      child: Container(
        width: double.infinity,
        height: 45,
        decoration: BoxDecoration(
          color: Colors.blue,
          borderRadius: BorderRadius.circular(10),
        ),
        child: Center(
          child: _isSigning
            ? CircularProgressIndicator(
                color: Colors.white,
              )
            : Text(
                "Login",
                style: TextStyle(
                  color: Colors.white,
                  fontWeight: FontWeight.bold,
                ),
              ),
        ),
      ),
    ),
    SizedBox(
      height: 10,
    ),
    SizedBox(
      height: 20,
    ),

    // Row(
    //   mainAxisAlignment: MainAxisAlignment.center,
    //   children: [

```

```

        // Text("Don't have an account?"),
        // SizedBox(
        //   width: 5,
        // ),
        // GestureDetector(
        //   onTap: () {
        //     Navigator.pushAndRemoveUntil(
        //       context,
        //       MaterialPageRoute(builder: (context) => SignUpPage()),
        //       (route) => false,
        //     );
        //   },
        //   child: Text(
        //     "Sign Up",
        //     style: TextStyle(
        //       color: Colors.blue,
        //       fontWeight: FontWeight.bold,
        //     ),
        //   ),
        // ),
        // ],
        // ),
        Spacer(flex: 2,),
      ],
    ),
  ),
);
}

```

```

void _signIn() async {
  setState() {
    _isSigning = true;
  });
}

```

```

String email = _emailController.text;
String password = _passwordController.text;

```

```

User? user = await _auth.signInWithEmailAndPassword(email, password);

```

```

    setState() {
      _isSigning = false;
    });

    if (user != null) {
      showToast(message: "Pengguna Berhasil Login");
      Navigator.pushNamed(context, "/home");
    }
  }

// _signInWithGoogle()async{

//   final GoogleSignIn _googleSignIn = GoogleSignIn();

//   try {

//       final GoogleSignInAccount? googleSignInAccount = await
//       _googleSignIn.signIn();

//       if(googleSignInAccount != null ){
//         final GoogleSignInAuthentication googleSignInAuthentication = await
//         googleSignInAccount.authentication;

//         final AuthCredential credential = GoogleAuthProvider.credential(
//           idToken: googleSignInAuthentication.idToken,
//           accessToken: googleSignInAuthentication.accessToken,
//         );

//         await _firebaseAuth.signInWithCredential(credential);
//         Navigator.pushNamed(context, "/home");
//       }

//     }catch(e) {
//       showToast(message: "some error occured $e");
//     }

//   }
}

```

```

home_page.dart
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

import 'package:carousel_slider/carousel_slider.dart' ;
import '../global/common/toast.dart';

class HomePage extends StatefulWidget {
  const HomePage({super.key});

  @override
  State<HomePage> createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
  List imageList = [
    {"id": 1, "image": 'assets/images/neww1.png'},
    {"id": 2, "image": 'assets/images/neww2.jpg'},
    {"id": 3, "image": 'assets/images/neww3.png'},
  ];

  final List<Map<String, dynamic>> buttons = [
    {
      "text": "Tambah Data",
      "icon": Icons.add, // Example icon
      "route": "/hewan",
    },
    {
      "text": "Lihat Data",
      "icon": Icons.remove_red_eye, // Another example icon
      "route": "/lihat",
    },
    // Add more buttons as needed
  ];

  // IconData getIconData(dynamic iconName) {
  //   if (iconName is String && iconMap.containsKey(iconName)) {
  //     return iconMap[iconName]!;
  //   }
  //   return Icons.error; // Fallback icon

```

```
// }
```

```
final CarouselController carouselController = CarouselController();
int _current = 0;
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      automaticallyImplyLeading: false,
      title: Text(""),
    ),
    body: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.start,
        children: [
          Padding(
            padding: const EdgeInsets.only(left: 15),
            child: Align(
              alignment: Alignment.topLeft,
              child: Text(
                "Portal Berita Peternakan",
                style:
                  TextStyle(fontSize: 20, fontWeight: FontWeight.bold),
              )),
        ),
          SizedBox(
            height: 20,
          ),
          Stack(
            children: [
              InkWell(
                onTap: () {
                  print(_current);
                },
                child: CarouselSlider(
                  items: imageUrl
                    .map((item) => Container(
                      width: 500, // Lebar tetap untuk gambar
                      height: 200, // Tinggi tetap untuk gambar
                      child: Image.asset(
```

```

        item['image'],
        fit: BoxFit
            .cover, // Mengisi kontainer sepenuhnya sambil
mempertahankan rasio aspek gambar
    ),
    ))
    .toList(),
    carouselController: carouselController,
    options: CarouselOptions(
        scrollPhysics: const BouncingScrollPhysics(),
        autoPlay: true,
        aspectRatio: 2.0,
        viewportFraction: 1,
        onPageChanged: (index, reason) {
            setState(() {
                _current = index;
            });
        },
    )),
),
Positioned(
    child: Row(
        mainAxisAlignment: MainAxisAlignment.center,
        children: imageList.asMap().entries.map((entry) {
            print(entry);
            print(entry.key);
            return GestureDetector(
                onTap: () =>
                    carouselController.animateToPage(entry.key),
                child: Container(
                    width: _current == entry.key ? 17 : 7,
                    height: 7.0,
                    margin: const EdgeInsets.symmetric(horizontal: 3.0),
                    decoration: BoxDecoration(
                        borderRadius: BorderRadius.circular(10),
                        color: _current == entry.key
                            ? Colors.blue
                            : Colors.grey.withOpacity(0.5),
                    ),
                ),
            ),
        ),
    ),
),

```

```

        );
      }).toList(),
    ),
  )
],
),
 SizedBox(
   height: 50,
 ),
 Expanded(
   child: GridView.builder(
     padding: const EdgeInsets.all(10),
     itemCount: buttons.length,
     gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
       crossAxisCount: 2, // Adjust number of columns
       crossAxisSpacing: 10, // Horizontal space between items
       mainAxisSpacing: 10, // Vertical space between items
     ),
     itemBuilder: (context, index) {
       return ElevatedButton(
         onPressed: () {
           Navigator.pushNamed(context, buttons[index]["route"]!);
         },
         style: ElevatedButton.styleFrom(
           backgroundColor: Colors.blue,
           foregroundColor: Colors.white,
           shape: RoundedRectangleBorder(
             borderRadius: BorderRadius.circular(10),
           ),
           padding:
             EdgeInsets.symmetric(horizontal: 20, vertical: 10),
         ),
         child: Column(
           mainAxisAlignment: MainAxisAlignment.min,
           children: [
             Icon(
               buttons[index]["icon"] != null
                 ? buttons[index]["icon"] as IconData
                 : Icons.error,
             ),
           ],
         ),
       );
     },
   ),
 ),

```



```

        ],
      ),
    )
      ),
      ],
    ),
  ));
}
}
Lihat_data.dart
import 'dart:ffi';
import 'package:intl/intl.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/material.dart';
import
  'package:flutter_firebase/features/user_auth/presentation/model/hewan.dart
  ';
import
  'package:flutter_firebase/features/user_auth/presentation/pages/addHewan.
  dart';
import 'package:flutter_firebase/features/user_auth/presentation/model/time.dart';
import
  'package:flutter_firebase/features/user_auth/presentation/services/database.
  dart';
import 'package:fluttertoast/fluttertoast.dart';

class LookData extends StatefulWidget {
  const LookData({super.key});

  @override
  State<LookData> createState() => _LookDataState();
}

String? dropdownValue;

class _LookDataState extends State<LookData> {
  Stream? HewanStream;
  final CollectionReference collection =
    FirebaseFirestore.instance.collection('time');

  TextEditingController jenisHewanController = TextEditingController();

```

```

TextEditingController beratHewanController = TextEditingController();
TextEditingController asalHewanController = TextEditingController();
        TextEditingController        namaPemotongHewanController        =
        TextEditingController();
final selectedDate = DateTime.now();
// final retrievedAnimalData = Time.fromMap(doc.data() as Map<String,
        dynamic>);

// final retrievedTimestamp = retrievedAnimalData.timestamp;
// final retrievedDateTime = retrievedTimestamp.toDate();

// Future<void> getAnimal() async {
// final doc = await collection.doc('time').get();
// final retrievedAnimalData = hewanList(doc.data());
// // Use retrievedAnimalData here
// }

Widget allHewanList() {
return StreamBuilder(
    stream: FirebaseFirestore.instance.collection('hewan').snapshots(),
    builder: (context, AsyncSnapshot snapshot) {
        if (!snapshot.hasData) return const Text('Loading...');
        // final filteredDocs = snapshot.data.docs.where((doc) {
        // return search.isEmpty ||
        //     doc['jenisHewan'].toLowerCase().contains(search.toLowerCase());
        // }).toList();
        // final filteredDocs = snapshot.data.docs.where((doc) {
        // return search.isEmpty ||
        //     doc['beratHewan']
        //         .toString()
        //         .toLowerCase()
        //         .contains(search.toLowerCase());
        // }).toList();
        // final filteredDocs = snapshot.data.docs.where((doc) {
        // final date =
        //     DateFormat('yyyy-MM-dd').format(doc['timestamp'].toDate());
        // return search.isEmpty || date.contains(search);
        // }).toList();
        final filteredDocs = snapshot.data.docs.where((doc) {
            final date =
                DateFormat('yyyy-MM-dd').format(doc['timestamp'].toDate());

```

```

return search.isEmpty ||
    doc['jenisHewan'].toLowerCase().contains(search.toLowerCase()) ||
    doc['namaPemotongHewan']
        .toLowerCase()
        .contains(search.toLowerCase()) ||
    doc['beratHewan']
        .toString()
        .toLowerCase()
        .contains(search.toLowerCase()) ||
    date.contains(search);
}).toList();
return ListView.builder(
    itemCount: filteredDocs.length,
    itemBuilder: (context, index) {
        DocumentSnapshot ds = filteredDocs[index];
        return Padding(
            padding: const EdgeInsets.all(8.0),
            child: Material(
                elevation: 5.0,
                borderRadius: BorderRadius.circular(10),
                child: Container(
                    width: MediaQuery.of(context).size.width,
                    decoration: BoxDecoration(
                        color: Colors.white,
                        borderRadius: BorderRadius.circular(10),
                    ),
                    padding: EdgeInsets.all(10),
                    child: Column(
                        children: [
                            Container(
                                width: 150.0, // Tentukan lebar tetap
                                height: 150.0, // Tentukan tinggi tetap
                                decoration: BoxDecoration(
                                    color: Colors.white,
                                    borderRadius: BorderRadius.circular(
                                        50), // Membuat sudut kontainer menjadi circular
                                ),
                                child: ClipRRect(
                                    borderRadius: BorderRadius.circular(
                                        20), // Membuat gambar menjadi circular

```

```

child: Image.network(
  ds["imageUrl"],
  fit: BoxFit
    .cover, // Mengisi kontainer sepenuhnya tanpa mengubah
rasio aspek
  errorBuilder: (BuildContext context,
    Object exception, StackTrace? stackTrace) {
    return Text(
      'Could not load image'); // Tampilkan teks ini jika gambar
tidak dapat dimuat
    },
  ),
),
),
Align(
  alignment: Alignment.centerLeft,
  child: Row(
    mainAxisAlignment: MainAxisAlignment.spaceBetween,
    children: [
      RichText(
        text: TextSpan(
          children: [
            TextSpan(
              text: "Jenis Hewan : ",
              style: TextStyle(
                fontWeight: FontWeight.bold,
                fontSize: 17,
                color: Colors.blue,
              ),
            ),
            TextSpan(
              text: ds["jenisHewan"],
              style: TextStyle(
                fontWeight: FontWeight.bold,
                fontSize: 17,
                color: Colors
                  .orange, // Change this to your desired color
            ),
          ],
        ),
      ),
    ],
  ),
),

```

```

    ),
    Spacer(),
    GestureDetector(
      onTap: () {
        jenisHewanController.text =
          ds["jenisHewan"];
        beratHewanController.text =
          ds["beratHewan"];
        asalHewanController.text = ds["asalHewan"];
        dropdownValue = ds["kondisiKesehatan"];
        namaPemotongHewanController.text =
          ds["namaPemotongHewan"];
        EditHewan(ds["id"]);
      },
      child: Icon(
        Icons.edit,
        color: Colors.blue,
        size: 24,
      ),
    ),
    SizedBox(
      width: 10,
    ),
    GestureDetector(
      onTap: () {
        showDialog(
          context: context,
          builder: (context) {
            return AlertDialog(
              title: const Text('Hapus data'),
              content:
                const Text('Apa anda yakin'),
              actions: [
                TextButton(
                  onPressed: () {
                    Navigator.pop(context);
                  },
                  child:
                    const Text('Batalkan')),
                TextButton(

```

```
onPressed: () {  
    DatabaseMethods()  
        .deleteHewanDetails(  
            ds["id"]);  
    Navigator.pop(context);  
    Fluttertoast.showToast(  
        msg:  
            "Data berhasil dihapus",  
        toastLength:  
            Toast.LENGTH_SHORT,  
        gravity:  
            ToastGravity.BOTTOM,  
        timeInSecForIosWeb: 1,  
        backgroundColor:  
            Colors.red,  
        textColor: Colors.white,  
        fontSize: 16.0);  
},  
child: const Text("Hapus"))  
],  
);  
});  
  
},  
child: Icon(  
    Icons.delete,  
    color: Colors.red,  
    size: 24,  
),  
),  
],  
),  
),  
SizedBox(  
    height: 5,  
),  
Align(  
    alignment: Alignment.centerLeft,  
    child: RichText(  
        text: TextSpan(  
            children: [
```

```

        TextSpan(
          text: "Berat Hewan : ",
          style: TextStyle(
            fontWeight: FontWeight.bold,
            fontSize: 17,
            color: Colors.blue,
          ),
        ),
        TextSpan(
          text: ds["beratHewan"],
          style: TextStyle(
            fontWeight: FontWeight.bold,
            fontSize: 17,
            color: Colors
              .orange, // Change this to your desired color
          ),
        ),
      ],
    ),
  )),
  SizedBox(
    height: 5,
  ),
  Align(
    alignment: Alignment.centerLeft,
    child: RichText(
      text: TextSpan(
        children: [
          TextSpan(
            text: "Asal Hewan : ",
            style: TextStyle(
              fontWeight: FontWeight.bold,
              fontSize: 17,
              color: Colors.blue,
            ),
          ),
          TextSpan(
            text: ds["asalHewan"],
            style: TextStyle(
              fontWeight: FontWeight.bold,

```

```

        fontSize: 17,
        color: Colors
            .orange, // Change this to your desired color
    ),
    ),
    ],
    ),
    )),
    SizedBox(
        height: 5,
    ),
    Align(
        alignment: Alignment.centerLeft,
        child: RichText(
            text: TextSpan(children: [
                TextSpan(
                    text: "Kondisi Kesehatan : ",
                    style: TextStyle(
                        fontWeight: FontWeight.bold,
                        fontSize: 17,
                        color: Colors.blue,
                    ),
                ),
                TextSpan(
                    text: ds["kondisiKesehatan"],
                    style: TextStyle(
                        fontWeight: FontWeight.bold,
                        fontSize: 17,
                        color: Colors
                            .orange, // Change this to your desired color
                    ),
                )
            ])),
    ),
    SizedBox(
        height: 5,
    ),
    Align(
        alignment: Alignment.centerLeft,
        child: RichText(
            text: TextSpan(

```

```

        children: [
          TextSpan(
            text: "Atas Nama Pemotong : ",
            style: TextStyle(
              fontWeight: FontWeight.bold,
              fontSize: 16,
              color: Colors.blue,
            ),
          ),
          TextSpan(
            text: ds["namaPemotongHewan"],
            style: TextStyle(
              fontWeight: FontWeight.bold,
              fontSize: 16,
              color: Colors
                .orange, // Change this to your desired color
            ),
          ),
        ],
      )),
    ),
    Text(
      DateFormat('yyyy-MM-dd')
        .format(ds["timestamp"].toDate()),
      style: TextStyle(fontWeight: FontWeight.bold),
    ),
  ],
),
)),
);
});
},
);
}

```

```
String search = "";
```

```

@override
Widget build(BuildContext context) {
  return Scaffold(

```

```

appBar: AppBar(
  automaticallyImplyLeading: false,
  title: Text("Lihat Data"),
  leading: IconButton(
    icon: Icon(Icons.arrow_back),
    onPressed: () {
      Navigator.pop(context);
    },
  ),
),
body: Container(
  margin: EdgeInsets.symmetric(horizontal: 20, vertical: 20),
  child: Column(
    children: [
      // ElevatedButton(
      //   onPressed: () {
      //     // Navigate to addHewan.dart and pass the addDataToList function
      //     Navigator.push(
      //       context,
      //       MaterialPageRoute(
      //         builder: (context) => AddHewan(addDataCallback:
addDataToList),
      //       ),
      //     );
      //   },
      //   child: Text('Add Data'),
      // ),
      const SizedBox(
        height: 20,
      ),
      TextField(
        onChanged: (value) {
          setState(() {
            search = value;
          });
        },
        decoration: InputDecoration(
          hintText: "Cari Hewan",
          prefixIcon: Icon(Icons.search),
          border: OutlineInputBorder(

```

```

        borderRadius: BorderRadius.circular(10),
      ),
    ),
  ),
  Expanded(child: allHewanList()),
],
)),
);
}

```

```

Future EditHewan(String id) => showDialog(
  context: context,
  builder: (context) => AlertDialog(
    content: SingleChildScrollView(
      child: Container(
        child: Column(
          children: [
            Row(
              children: [
                GestureDetector(
                  onTap: () => {Navigator.pop(context)},
                  child: Icon(Icons.cancel)),
                SizedBox(width: 30),
                Text(
                  "Edit Data Hewan",
                  style: TextStyle(
                    fontSize: 20, fontWeight: FontWeight.bold),
                )
              ],
            ),
            Align(
              alignment: Alignment.centerLeft,
              child: Text(
                'Jenis Hewan',
                style: TextStyle(
                  fontSize: 16, fontWeight: FontWeight.bold),
              )),
            SizedBox(height: 8),
            TextField(
              controller: jenisHewanController,

```

```

        decoration: InputDecoration(
          border: OutlineInputBorder(),
        ),
      ),
      SizedBox(height: 16),
      Align(
        alignment: Alignment.centerLeft,
        child: Text(
          'Berat Hewan (Kg)',
          style: TextStyle(
            fontSize: 16, fontWeight: FontWeight.bold),
        ),
      ),
      SizedBox(height: 8),
      TextField(
        controller: beratHewanController,
        decoration: InputDecoration(
          border: OutlineInputBorder(),
        ),
      ),
      SizedBox(height: 16),
      Align(
        alignment: Alignment.centerLeft,
        child: Text(
          'Asal Hewan',
          style: TextStyle(
            fontSize: 16, fontWeight: FontWeight.bold),
        ),
      ),
      SizedBox(height: 8),
      TextField(
        controller: asalHewanController,
        decoration: InputDecoration(
          border: OutlineInputBorder(),
        ),
      ),
      SizedBox(height: 16),
      Align(
        alignment: Alignment.centerLeft,
        child: Text(

```

```

        'Kondisi Kesehatan',
        style: TextStyle(
          fontSize: 16, fontWeight: FontWeight.bold),
      ),
    ),
    SizedBox(height: 8),
    Align(
      alignment: Alignment.centerLeft,
      child: Container(
        width: 200,
        padding: EdgeInsets.only(left: 8),
        decoration: BoxDecoration(
          border: Border.all(color: Colors.grey),
          borderRadius: BorderRadius.circular(5),
        ),
      ),
      child: DropdownButtonHideUnderline(
        child: DropdownButton<String>(
          isExpanded: true,
          value: dropdownValue,
          icon: const Icon(Icons.add),
          style: const TextStyle(color: Colors.deepPurple),
          onChanged: (String? newValue) {
            setState(() {
              dropdownValue = newValue;
            });
          },
        ),
        items: <String>['Sehat', 'Sakit', 'Mati']
          .map<DropdownMenuItem<String>>((String value) {
            return DropdownMenuItem<String>(
              value: value,
              child: Container(
                margin: EdgeInsets.only(left: 8),
                child: Text(value,
                  style: TextStyle(
                    fontSize: 16,
                    fontWeight: FontWeight.bold,
                    color: Colors.black))),
            );
          }).toList(),
      ),

```

```

    ),
    ),
),
SizedBox(height: 8),
Align(
  alignment: Alignment.centerLeft,
  child: Text(
    'Atas Nama Pemotong',
    style: TextStyle(
      fontSize: 16, fontWeight: FontWeight.bold),
  ),
),
SizedBox(height: 8),
TextField(
  controller: namaPemotongHewanController,
  decoration: InputDecoration(
    border: OutlineInputBorder(),
  ),
),
),
SizedBox(height: 16),
ElevatedButton(
  onPressed: () {
    Map<String, dynamic> hewanInfoMap = {
      "jenisHewan": jenisHewanController.text,
      "beratHewan": beratHewanController.text,
      "asalHewan": asalHewanController.text,
      "kondisiKesehatan": dropdownValue,
      "namaPemotongHewan":
        namaPemotongHewanController.text,
    };
    DatabaseMethods()
      .editHewanDetails(id, hewanInfoMap)
      .then((value) {
        Navigator.pop(context);
        Fluttermtoast.showToast(
          msg: "Data berhasil diubah",
          toastLength: Toast.LENGTH_SHORT,
          gravity: ToastGravity.BOTTOM,
          timeInSecForIosWeb: 1,
          backgroundColor: Colors.green,

```

```

        textColor: Colors.white,
        fontSize: 16.0
      );
    }).catchError((error) {
      Fluttern.toast.showToast(
        msg: "Data berhasil diubah",
        toastLength: Toast.LENGTH_SHORT,
        gravity: ToastGravity.BOTTOM,
        timeInSecForIosWeb: 1,
        backgroundColor: Colors.green,
        textColor: Colors.white,
        fontSize: 16.0
      );
    });
    Navigator.pop(context);
  },
  child: Text("Edit Data"))
],
),
),
),
));
}

```

Tambah_data.dart

```

import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'dart:io';
import 'package:path/path.dart';
import 'package:flutter/widgets.dart';
import 'package:firebase_storage/firebase_storage.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:random_string/random_string.dart';
import 'package:fluttern.toast/fluttertoast.dart';
import 'package:image_picker/image_picker.dart';

```

```

class AddHewan extends StatefulWidget {
  const AddHewan({super.key});

```

```

  @override

```

```

State<AddHewan> createState() => _AddHewanState();
}

String? dropdownValue;
String? selectedPemotongHewan;

class _AddHewanState extends State<AddHewan> {
  TextEditingController jenisHewanController = TextEditingController();
  TextEditingController beratHewanController = TextEditingController();
  TextEditingController asalHewanController = TextEditingController();
  TextEditingController namaPemotongHewanController =
    TextEditingController();
  File? _image;

  Future<void> pickImage() async {
    final ImagePicker _picker = ImagePicker();
    // Pick an image
    final XFile? image = await _picker.pickImage(source: ImageSource.gallery);
    if (image != null) {
      setState() {
        _image = File(image.path);
      });
    }
  }
}

Future<String> uploadImageToFirebase(File imageFile) async {
  String downloadURL = "";
  try {
    String fileName = basename(imageFile.path);
    Reference firebaseStorageRef =
      FirebaseStorage.instance.ref().child('uploads/$fileName');
    UploadTask uploadTask = firebaseStorageRef.putFile(imageFile);
    TaskSnapshot taskSnapshot = await uploadTask;

    // Get the download URL
    downloadURL = await taskSnapshot.ref.getDownloadURL();
    print("Done: $downloadURL"); // Cetak URL untuk memeriksa konsol debug
  } catch (e) {
    print("Error uploading image: $e");
  }
}

```

```
    return downloadURL;
}
```

```
Future<void> saveHewanInfo() async {
  if (dropdownValue == 'Sakit' || dropdownValue == 'Mati') {
    // Tampilkan pesan kesalahan
    Fluttermtoast.showToast(
      msg:
        "Hewan yang sakit atau mati tidak dapat ditambahkan ke dalam sistem",
      backgroundColor: Colors.red,
      textColor: Colors.white,
      toastLength: Toast.LENGTH_SHORT,
      gravity: ToastGravity.BOTTOM,
      timeInSecForIosWeb: 1);
    return; // Keluar dari method jika dropdownValue 'Sakit' atau 'Mati'
  }
  if (_image == null) {
    Fluttermtoast.showToast(
      msg: "Silakan pilih gambar terlebih dahulu",
      backgroundColor: Colors.red,
      textColor: Colors.white,
      toastLength: Toast.LENGTH_SHORT,
      gravity: ToastGravity.BOTTOM,
      timeInSecForIosWeb: 1);
    return; // Keluar dari method jika _image null
  }
}
```

```
String imageUrl = await uploadImageToFirebase(_image!);
```

```
String Id = randomAlphaNumeric(10);
final timestamp = Timestamp.fromDate(selectedDate);
Map<String, dynamic> hewanInfoMap = {
  "jenisHewan": jenisHewanController.text,
  "beratHewan": beratHewanController.text,
  "asalHewan": asalHewanController.text,
  'id': Id,
  "kondisiKesehatan": dropdownValue,
  "namaPemotongHewan": selectedPemotongHewan,
  "timestamp": timestamp,
  "imageUrl": imageUrl,
```

```

};

await FirebaseFirestore.instance
  .collection('hewan')
  .doc(Id)
  .set(hewanInfoMap)
  .then((value) {
Fluttertoast.showToast(
  msg: "Data berhasil ditambahkan",
  backgroundColor: Colors.green,
  textColor: Colors.white,
  toastLength: Toast.LENGTH_SHORT,
  gravity: ToastGravity.BOTTOM,
  timeInSecForIosWeb: 1);
});
}

final selectedDate = DateTime.now();

final CollectionReference collection =
  FirebaseFirestore.instance.collection('time');

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      automaticallyImplyLeading: false,
      title: Text("Tambah Data Hewan"),
      leading: IconButton(
        icon: Icon(Icons.arrow_back),
        onPressed: () {
          Navigator.pop(context);
        },
      ),
    ),
    body: SingleChildScrollView(
      child: Container(
        padding: EdgeInsets.all(16),
        child: Column(
          children: [

```

```

Align(
  alignment: Alignment.centerLeft,
  child: Text(
    'Jenis Hewan',
    style:
      TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
  )),
  SizedBox(height: 8),
  TextField(
    controller: jenisHewanController,
    decoration: InputDecoration(
      border: OutlineInputBorder(),
    ),
  ),
  SizedBox(height: 16),
  Align(
    alignment: Alignment.centerLeft,
    child: Text(
      'Berat Hewan (Kg)',
      style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
    ),
  ),
  SizedBox(height: 8),
  TextField(
    controller: beratHewanController,
    decoration: InputDecoration(
      border: OutlineInputBorder(),
    ),
  ),
  SizedBox(height: 16),
  Align(
    alignment: Alignment.centerLeft,
    child: Text(
      'Asal Hewan',
      style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
    ),
  ),
  SizedBox(height: 8),
  TextField(
    controller: asalHewanController,

```

```

        decoration: InputDecoration(
          border: OutlineInputBorder(),
        ),
      ),
      SizedBox(height: 16),
      Align(
        alignment: Alignment.centerLeft,
        child: Text(
          'Kondisi Kesehatan',
          style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
        ),
      ),
      SizedBox(height: 8),
      Align(
        alignment: Alignment.centerLeft,
        child: Container(
          width: 200,
          padding: EdgeInsets.only(right: 8),
          decoration: BoxDecoration(
            border: Border.all(color: Colors.grey),
            borderRadius: BorderRadius.circular(5),
          ),
        child: DropdownButtonHideUnderline(
          child: DropdownButton<String>(
            isExpanded: true,
            value: dropdownValue,
            icon: const Icon(Icons.add),
            style: const TextStyle(color: Colors.deepPurple),
            onChanged: (String? newValue) {
              setState() {
                dropdownValue = newValue;
              };
            },
            items: <String>['Sehat', 'Sakit', 'Mati']
              .map<DropdownMenuItem<String>>((String value) {
                return DropdownMenuItem<String>(
                  value: value,
                  child: Container(
                    margin: EdgeInsets.only(left: 8),
                    child: Text(value,

```

[illegible]

```

    });
  },
  items: <String>[
    'Lukman Hasan',
    'Hatta',
    'Lukman Muklis',
    'Baharuddin',
  ] // Daftar pilihan baru
    .map<DropDownMenuItem<String>>((String value) {
  return DropDownMenuItem<String>(
    value: value,
    child: Container(
      margin: EdgeInsets.only(left: 8),
      child: Text(value,
        style: TextStyle(
          fontSize: 16,
          fontWeight: FontWeight.bold,
          color: Colors.black))),
    );
  }).toList(),
),
),
),
),
ElevatedButton(
  onPressed: () {
    pickImage();
  },
  child: Text('Pilih Gambar Hewan'),
),
_image != null ? Image.file(_image!) : Container(),
 SizedBox(height: 26),
 ElevatedButton(
  style: ElevatedButton.styleFrom(
    backgroundColor: Color.fromARGB(255, 255, 204, 0),
    minimumSize: Size(double.infinity, 50),
  ),
  onPressed: () async {
    if (_image == null) {
      Fluttertoast.showToast(

```

```

        msg: "Silakan pilih gambar terlebih dahulu",
        backgroundColor: Colors.red,
        textColor: Colors.white,
        toastLength: Toast.LENGTH_SHORT,
        gravity: ToastGravity.BOTTOM,
        timeInSecForIosWeb: 1);
    return;
}
await saveHewanInfo();
// await uploadImageToFirebase(context);
// if (dropdownValue == 'Sakit' || dropdownValue == 'Mati') {
//   // Show an error message
//   ScaffoldMessenger.of(context).showSnackBar(
//     SnackBar(
//       content: Text(
//         // 'Hewan yang sakit atau mati tidak dapat ditambahkan ke
dalam sistem')),
//     );
//   } else {
//     String Id = randomAlphaNumeric(10);
//     final timestamp = Timestamp.fromDate(selectedDate);
//     Map<String, dynamic> hewanInfoMap = {
//       "jenisHewan": jenisHewanController.text,
//       "beratHewan": beratHewanController.text,
//       "asalHewan": asalHewanController.text,
//       'id': Id,
//       "kondisiKesehatan": dropdownValue,
//       "namaPemotongHewan": namaPemotongHewanController.text,
//       "timestamp": timestamp,

//     };

//     final docRef = collection.doc('time');
//     await docRef.set(hewanInfoMap).then((value) {});

//     await DatabaseMethods()
//       .addHewanDetails(hewanInfoMap, Id)
//       .then((value) {
//         Fluttertoast.showToast(
//           msg: "Data berhasil ditambahkan",

```

```

        //      backgroundColor: Colors.green,
        //      textColor: Colors.white,
        //      toastLength: Toast.LENGTH_SHORT,
        //      gravity: ToastGravity.BOTTOM,
        //      timeInSecForIosWeb: 1);
        //  });
        //  }
    },
    child: Text("Tambah",
        style: TextStyle(
            fontSize: 16,
            color: Color.fromARGB(255, 33, 29, 29))),
    )
    ],
    ),
    ),
    ));
}
}

```