

## LAMPIRAN

### Lampiran 1. Pedoman Wawancara

Narasumber: Pemilik UMKM Pabrik Kerupuk Anugrah

Tujuan: Mengidentifikasi proses pencatatan piutang, kendala yang terjadi, dan kebutuhan sistem aplikasi mobile.

| No. | Pertanyaan                                                                                     |
|-----|------------------------------------------------------------------------------------------------|
| 1.  | Bagaimana proses pencatatan pengiriman barang kepada agen/distributor yang dilakukan saat ini? |
| 2.  | Bagaimana proses pencatatan pembayaran dari agen/distributor?                                  |
| 3.  | Bagaimana cara menghitung sisa saldo piutang setiap agen?                                      |
| 4.  | Kendala apa yang paling sering terjadi dalam pencatatan menggunakan buku?                      |
| 5.  | Apakah pernah terjadi kesalahan perhitungan, kehilangan, atau kerusakan catatan transaksi?     |
| 6.  | Data apa saja yang menurut Bapak/Ibu perlu dicatat dalam sistem?                               |
| 7.  | Apakah diperlukan aplikasi mobile untuk membantu pemantauan piutang?                           |
| 8.  | Apakah diperlukan laporan atau rekap transaksi dalam bentuk PDF?                               |
| 9.  | Apakah diperlukan pengiriman bukti transaksi kepada agen melalui WhatsApp?                     |
| 10. | Bagaimana pendapat Bapak/Ibu mengenai keamanan aplikasi, termasuk penggunaan autentikasi?      |
| 11. | Setelah mencoba prototype, fitur apa yang menurut Bapak/Ibu perlu ditambahkan atau diperbaiki? |

## Lampiran 2. Lembar & Hasil Observasi

Lokasi: UMKM Pabrik Kerupuk Anugrah, Desa Bangbayang

Objek observasi: Proses distribusi barang, pencatatan transaksi, penerimaan pembayaran, perhitungan piutang, dan pengelolaan arsip.

| No. | Aspek               | Hasil Observasi                                                                                 | Status          |
|-----|---------------------|-------------------------------------------------------------------------------------------------|-----------------|
| 1.  | Media pencatatan    | Menggunakan buku/catatan fisik.                                                                 | Teramati        |
| 2.  | Urutan transaksi    | Pengiriman barang dan pembayaran dicatat sebagai transaksi.                                     | Teramati        |
| 3.  | Data yang dicatat   | Data agen, tanggal, jenis transaksi, nominal, dan keterangan menjadi informasi yang dibutuhkan. | Teridentifikasi |
| 4.  | Perhitungan saldo   | Saldo piutang perlu dihitung dari transaksi pengiriman dan pembayaran.                          | Teramati        |
| 5.  | Potensi kesalahan   | Pencatatan dan perhitungan manual berpotensi menimbulkan human error.                           | Teridentifikasi |
| 6.  | Kondisi arsip       | Arsip fisik memiliki risiko kerusakan atau kehilangan.                                          | Teridentifikasi |
| 7.  | Kebutuhan teknologi | Aplikasi mobile diperlukan untuk membantu pengelolaan dan pemantauan piutang.                   | Teridentifikasi |

### Lampiran 3. System Usability Scale (SUS)

| Form Responses 1 | Timestamp           | Nama Lengkap | Jabatan/Peran dalam Usaha | Saya pikir saya akan sering menggunakan ap | Saya merasa aplikasi ini terlalu rumit (komplek | Saya merasa aplikasi ini terla |
|------------------|---------------------|--------------|---------------------------|--------------------------------------------|-------------------------------------------------|--------------------------------|
| 2                | 23/07/2026 18:02:17 | Iyan Diana   | Bos/Pemilik/Pengrajin     | Sangat Setuju                              | Tidak Setuju                                    | Tidak Setuju                   |
| 3                |                     |              |                           |                                            |                                                 |                                |
| 4                |                     |              |                           |                                            |                                                 |                                |
| 5                |                     |              |                           |                                            |                                                 |                                |
| 6                |                     |              |                           |                                            |                                                 |                                |
| 7                |                     |              |                           |                                            |                                                 |                                |
| 8                |                     |              |                           |                                            |                                                 |                                |
| 9                |                     |              |                           |                                            |                                                 |                                |
| 10               |                     |              |                           |                                            |                                                 |                                |
| 11               |                     |              |                           |                                            |                                                 |                                |
| 12               |                     |              |                           |                                            |                                                 |                                |
| 13               |                     |              |                           |                                            |                                                 |                                |
| 14               |                     |              |                           |                                            |                                                 |                                |
| 15               |                     |              |                           |                                            |                                                 |                                |

### Lampiran 4. firebase\_option.dart

Kode Program Inisialisasi Aplikasi dan Firebase

```

import 'package:firebase_core/firebase_core.dart' show FirebaseOptions;
import 'package:flutter/foundation.dart'
    show defaultTargetPlatform, kIsWeb, TargetPlatform;

class DefaultFirebaseOptions {
  static FirebaseOptions get currentPlatform {
    if (kIsWeb) {
      return web;
    }
    switch (defaultTargetPlatform) {
      case TargetPlatform.android:
        return android;
      case TargetPlatform.iOS:
        return ios;
      case TargetPlatform.macos:
        return macos;
      case TargetPlatform.windows:
        return windows;
      case TargetPlatform.linux:
        throw UnsupportedError(
          'DefaultFirebaseOptions have not been configured for linux - '
          'You can reconfigure this by running the FlutterFire CLI again.'
        );
      default:
        throw UnsupportedError(
          'DefaultFirebaseOptions are not supported for this platform.'
        );
    }
  }

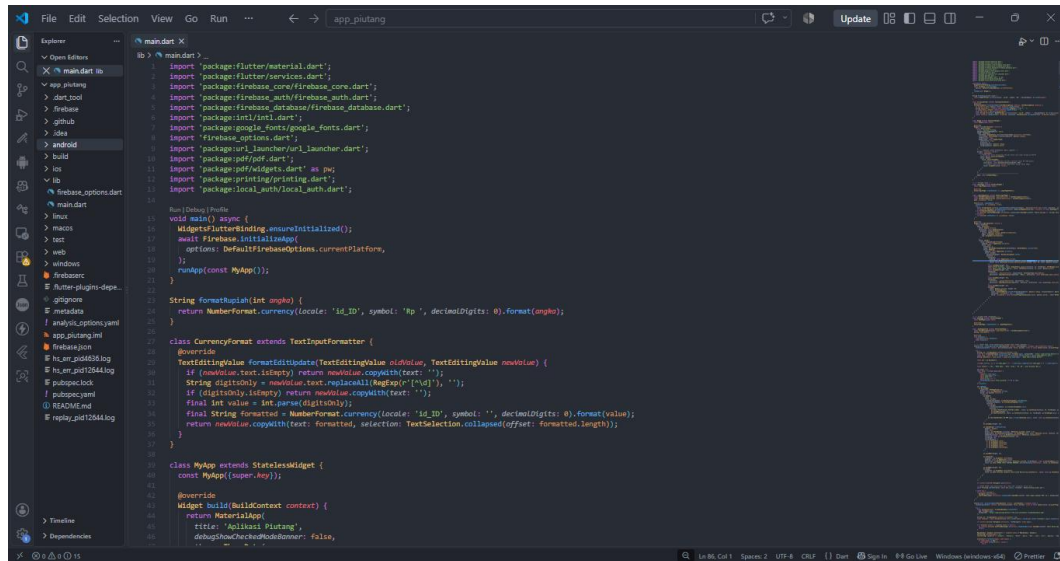
  static const FirebaseOptions web = FirebaseOptions(
    apiKey: 'AIzaSyMwrazVlPp0x8y9NtL7D0dehVpufU',
    appId: '1:542199378487:web:3d682a707a4b37c0f092',
    messagingSenderId: '542199378487',
    projectId: 'app-piutang',
    storageBucket: 'app-piutang.firebaseioapp.com',
    measurementId: 'G-6H038N7V4',
  );

  static const FirebaseOptions android = FirebaseOptions(
    apiKey: 'AIzaSyMwrazVlPp0x8y9NtL7D0dehVpufU',
    appId: '1:542199378487:android:78837929e4b5d6eb7092',
    messagingSenderId: '542199378487',
    projectId: 'app-piutang',
    storageBucket: 'app-piutang.firebaseioapp.com',
  );
}

```

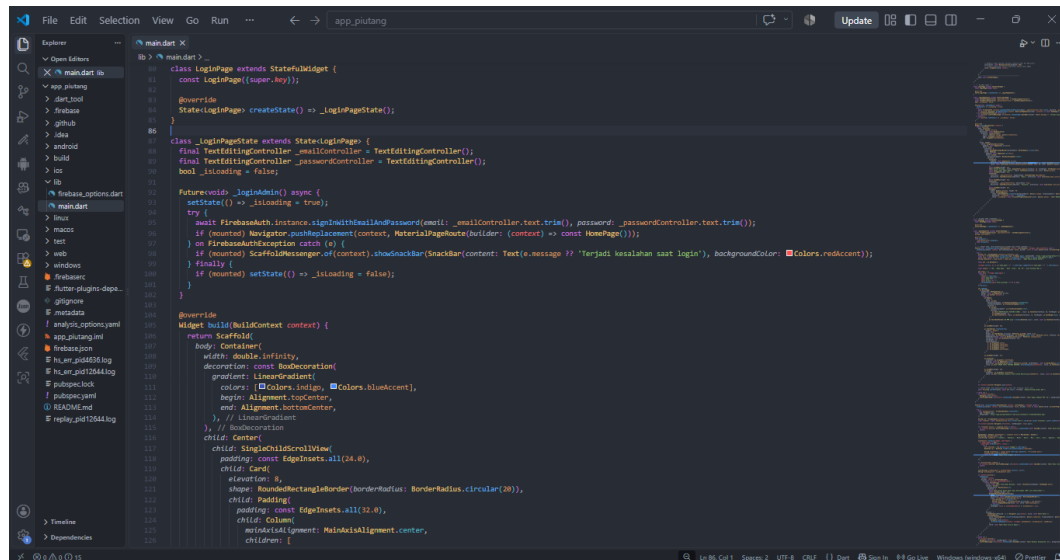
## Lampiran 5. *main.dart*

### Lampiran 5.1 Implementasi Inisialisasi Firebase



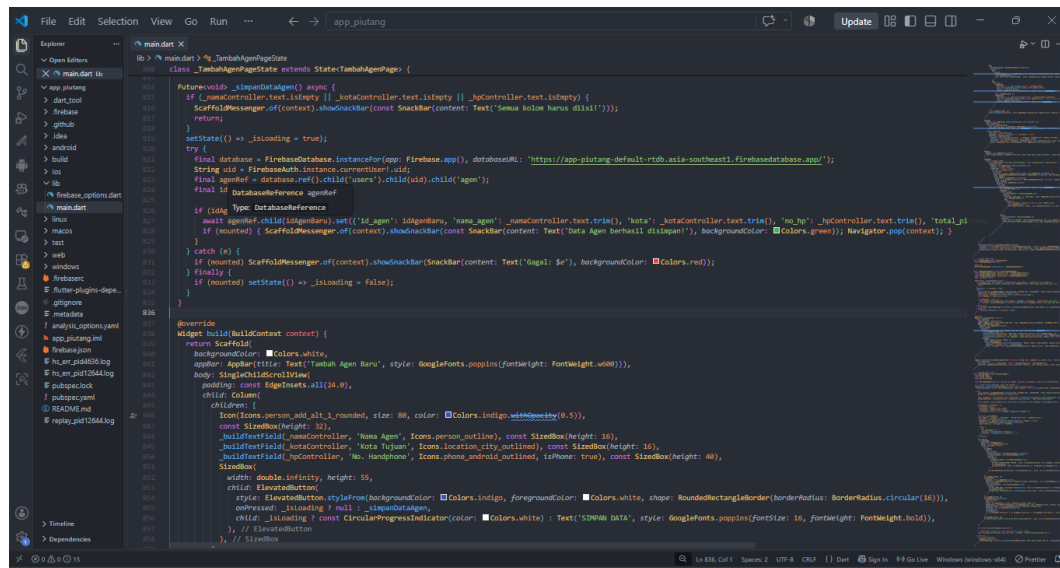
```
1 import 'package:flutter/material.dart';
2 import 'package:flutter/services.dart';
3 import 'package:firebase_core/firebase_core.dart';
4 import 'package:firebase_auth/firebase_auth.dart';
5 import 'package:firebase_database/firebase_database.dart';
6 import 'package:intl/intl.dart';
7 import 'package:google_fonts/google_fonts.dart';
8 import 'package:url_launcher/url_launcher.dart';
9 import 'package:pdf/pdf.dart';
10 import 'package:pdf/widgets.dart as pwj';
11 import 'package:printing/printing.dart';
12 import 'package:local_auth/local_auth.dart';
13
14 Run [Debug] (Profile)
15 void main() async {
16   WidgetsFlutterBinding.ensureInitialized();
17   await Firebase.initializeApp(
18     options: DefaultFirebaseOptions.currentPlatform,
19   );
20   runApp(const MyApp());
21 }
22
23 String formatRp(int rupiah) {
24   return NumberFormat.currency(locale: 'id_ID', symbol: 'Rp ', decimalDigits: 0).format(rupiah);
25 }
26
27 class CurrencyFormat extends TextInputFormatter {
28   @override
29   TextEditingValue formatEditingValue(TextEditingValue oldValue, TextEditingValue newValue) {
30     if (newValue.text.isEmpty) return newValue.copyWith(text: '');
31     String digitOnly = newValue.text.replaceAll(RegExp('[^0-9]'), '');
32     if (digitOnly.isEmpty) return newValue.copyWith(text: '');
33     final int value = int.parse(digitOnly);
34     final String formatted = NumberFormat.currency(locale: 'id_ID', symbol: ' ', decimalDigits: 0).format(value);
35     return newValue.copyWith(text: formatted, selection: TextSelection.collapsed(offset: formatted.length));
36   }
37 }
38
39 class MyApp extends StatelessWidget {
40   const MyApp({super.key});
41
42   @override
43   Widget build(BuildContext context) {
44     return MaterialApp(
45       title: 'Molikiel Putang',
46       debugShowCheckedStackBanner: false,
47     );
48   }
49 }
```

### Lampiran 5.2 Implementasi Login Admin

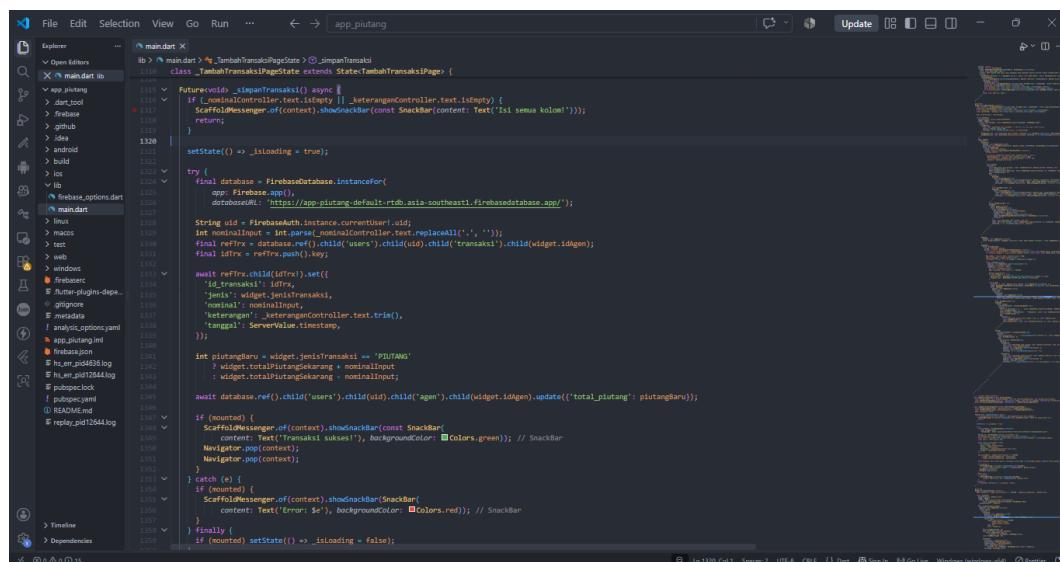


```
1 class LoginPage extends StatefulWidget {
2   const LoginPage({super.key});
3
4   @override
5   State<LoginPage> createState() => _LoginPageState();
6 }
7
8 class _LoginPageState extends State<LoginPage> {
9   final TextEditingController _emailController = TextEditingController();
10   final TextEditingController _passwordController = TextEditingController();
11   bool _isLoading = false;
12
13   Future<void> _loginAdmin() async {
14     setState(() => _isLoading = true);
15     try {
16       await FirebaseAuth.instance.signInWithEmailAndPassword(
17         email: _emailController.text.trim(),
18         password: _passwordController.text.trim(),
19       );
20       on FirebaseAuthException catch (e) {
21         if (mounted) ScaffoldMessenger.of(context).showSnackBar(
22           SnackBar(content: Text(e.message ?? 'Terjadi kesalahan saat login'), backgroundColor: Colors.redAccent));
23       }
24       if (mounted) setState(() => _isLoading = false);
25     }
26   }
27
28   @override
29   Widget build(BuildContext context) {
30     return Scaffold(
31       body: Container(
32         width: double.infinity,
33         decoration: const BoxDecoration(
34           gradient: LinearGradient(
35             colors: [Colors.indigo, Colors.blueAccent],
36             begin: Alignment.topCenter,
37             end: Alignment.bottomCenter,
38           ), // LinearGradient
39         ), // BoxDecoration
40         child: Center(
41           child: SingleChildScrollView(
42             padding: const EdgeInsets.all(24.0),
43             child: Card(
44               elevation: 8,
45               shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(20)),
46               child: Padding(
47                 padding: const EdgeInsets.all(32.0),
48                 child: Column(
49                   mainAxisAlignment: MainAxisAlignment.center,
50                   children: [
51
52 
```

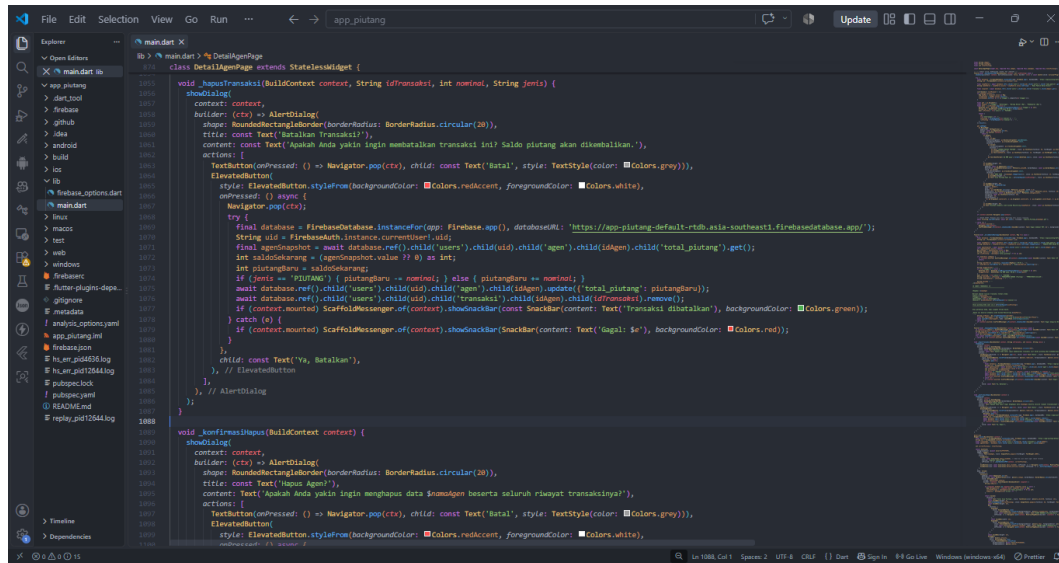
### Lampiran 5.3 Implementasi Penyimpanan Data Agen



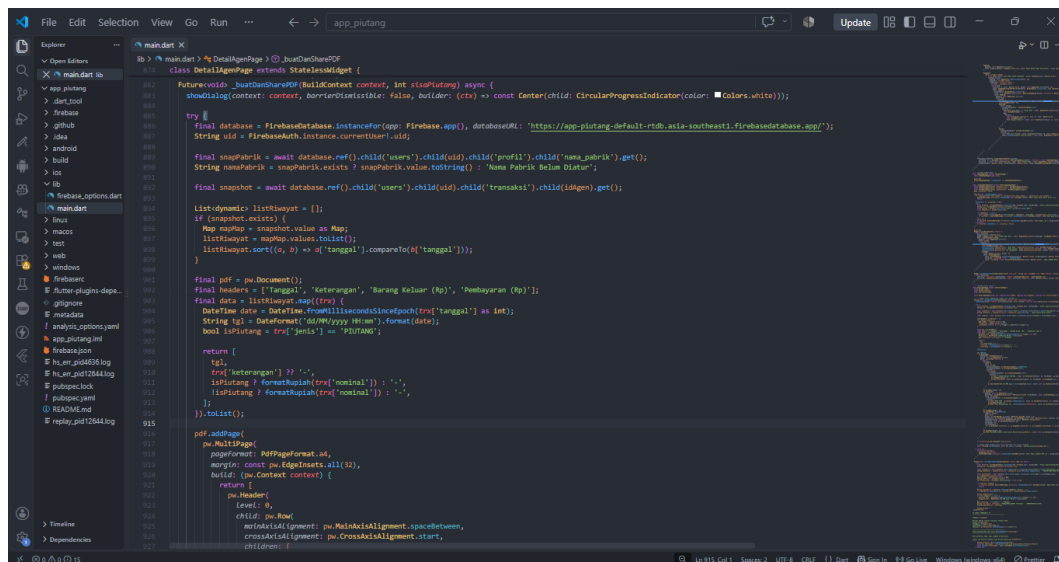
### Lampiran 5.4 Implementasi Penyimpanan Transaksi dan Perhitungan Piutang



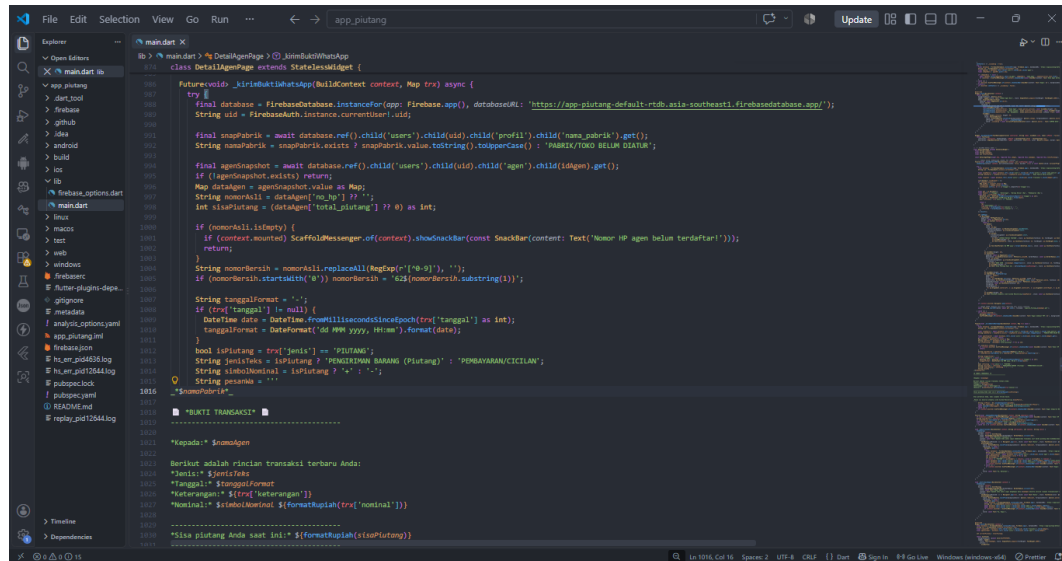
## Lampiran 5.5 Implementasi Penghapusan Transaksi dan Penyesuaian Piutang



## Lampiran 5.6 Implementasi Pembuatan dan Berbagi Laporan PDF

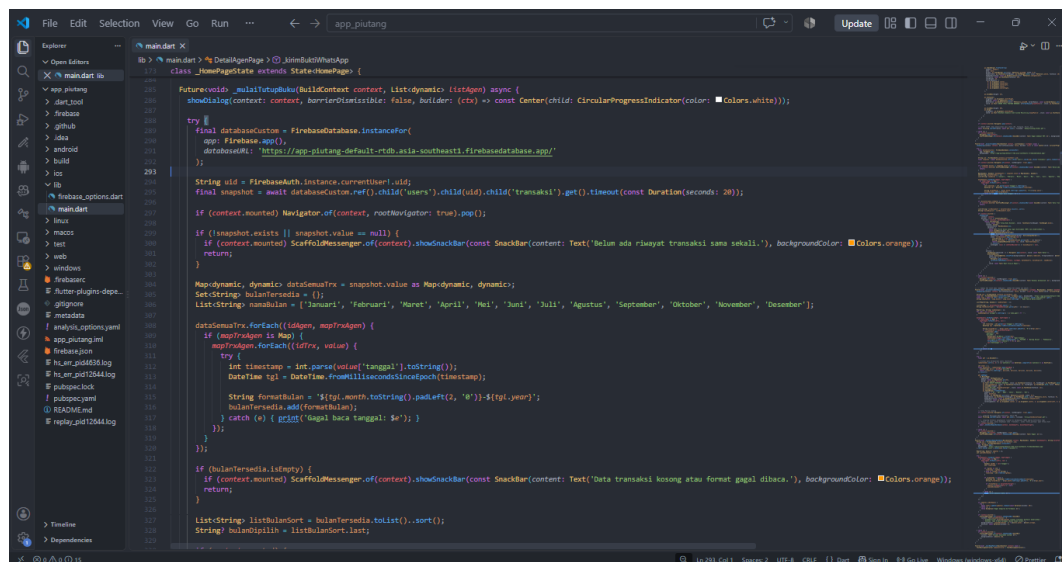


## Lampiran 5.7 Implementasi Pengiriman Bukti Transaksi melalui WhatsApp

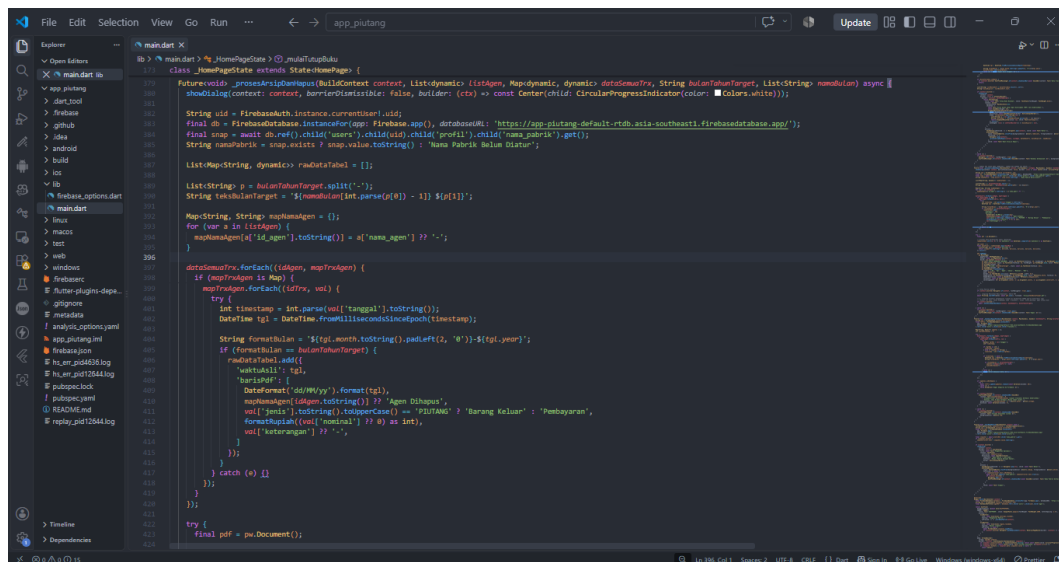


```
File Edit Selection View Go Run ... app.piutang
main.dart X
main.dart > % DetailAgePage > @ JomBudiWhatsApp
class DetailAgePage extends StatelessWidget {
  Future<void> kirimBuktiTransaksi(BuildContext context, Map<tr> async {
    try {
      final database = FirebaseDatabase.instanceFor(app: Firebase.app(), databaseURL: 'https://app-piutang-default-rtdb.asia-southeast1.firebaseio.com/app/');
      String uid = FirebaseAuth.instance.currentUser!.uid;
      final snapFabrik = await database.ref().child('users').child(uid).child('profil').child('nama_fabrik').get();
      String nomorFabrik = snapFabrik.exists ? snapFabrik.value.toString().toUpperCase() : 'PABRIK/TOKO BELUM BUKTI';
      final agenSnapshot = await database.ref().child('users').child(uid).child('agen').child('idagen').get();
      Map<dataagen> = agenSnapshot.value as Map;
      Map<dataagen> = dataagen['no_hp'] ?? '';
      String nomorKall = dataagen['no_hp'] ?? '';
      int sisaPiutang = (dataagen['total_piutang'] ?? 0) as int;
      if (nomorKall.isEmpty) {
        ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content: Text('Nomor HP agen belum terdaftar!')));
        return;
      }
      String nomorBerisih = nomorKall.replaceAll(RegExp(r'[^0-9]'), '');
      if (nomorBerisih.startsWith('0')) nomorBerisih = '62${nomorBerisih.substring(1)}';
      String tanggalFormat = '-';
      if (true[tanggal]) != null {
        DateTime date = DateTime.fromMillisecondsSinceEpoch(true[tanggal] as int);
        tanggalFormat = DateFormat('dd MMM yyyy, HH:mm').format(date);
      }
      bool isPiutang = true['jenis'] == 'PIUTANG';
      String jenisTeks = isPiutang ? 'PENGIKHTAN BARANG (Piutang)' : 'PEMBAYARAN/CICILAN';
      String simbolNominal = isPiutang ? 'Rp' : '-';
      String nominal = '1016';
      "Bukti Transaksi"
      "Kapan:" $nomoragen
      Berikut adalah rincian transaksi terbaru Anda:
      *Jenis* $jenisTeks
      *Tanggal* $tanggalFormat
      *Keterangan* $true[teterangan]
      *Nominal* $simbolNominal ${formatRupiah(true[nominal])}
      "Sisa piutang Anda saat ini:" ${formatRupiah(sisaPiutang)}
```

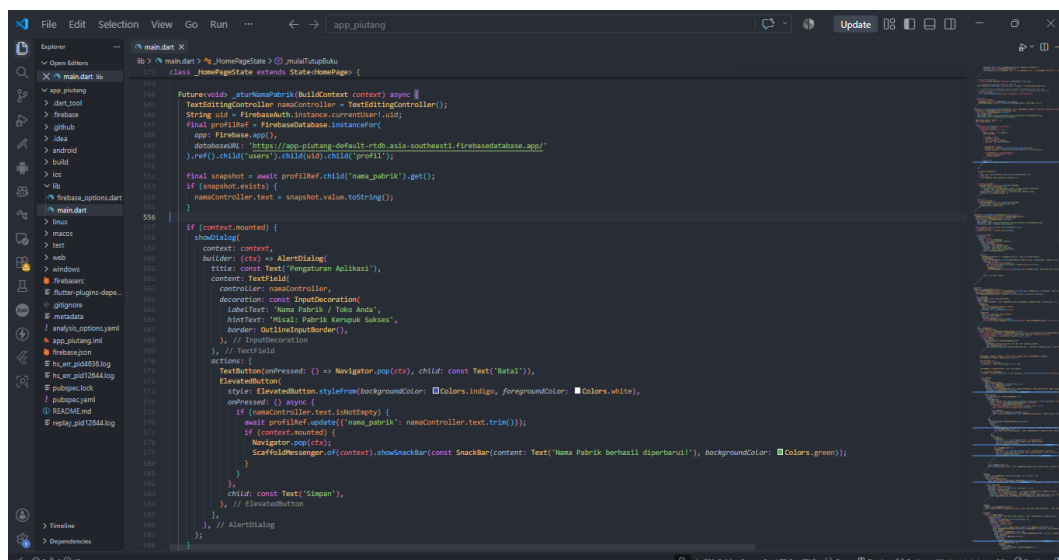
## Lampiran 5.8 Implementasi Fitur Tutup Buku dan Pengarsipan Data



```
File Edit Selection View Go Run ... app.piutang
main.dart X
main.dart > % DetailAgePage > @ JomBudiWhatsApp
class JomBudiState extends State<JomBudi> {
  Future<void> _mulaiTutupBuku(BuildContext context, List<dynamic> listAgen) async {
    showDialog(context: context, barrierDismissible: false, builder: (ctx) => const Center(child: CircularProgressIndicator(color: Colors.white)));
    try {
      final databaseCustom = FirebaseDatabase.instanceFor(
        app: Firebase.app(),
        databaseURL: 'https://app-piutang-default-rtdb.asia-southeast1.firebaseio.com/app/'
      );
      String uid = FirebaseAuth.instance.currentUser!.uid;
      final snapshot = await databaseCustom.ref().child('users').child(uid).child('transaksi').get().timeout(const Duration(seconds: 20));
      if (context.mounted) Navigator.of(context, rootNavigator: true).pop();
      if (snapshot.exists != null) {
        ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content: Text('Belum ada riwayat transaksi sama sekali.'), backgroundColor: Colors.orange));
        return;
      }
      Map<dynamic, dynamic> dataSnapshot = snapshot.value as Map<dynamic, dynamic>;
      Set<String> bulanTersedia = {};
      List<String> namaBulan = ['Januari', 'Februari', 'Maret', 'April', 'Mei', 'Juni', 'Juli', 'Agustus', 'September', 'Oktober', 'November', 'Desember'];
      dataSnapshot.forEach((idagen, mapTransaksi) {
        if (mapTransaksi is Map) {
          mapTransaksi.forEach((idTrx, value) {
            try {
              int timestamp = int.parse(value['tanggal'].toString());
              DateTime tgl = DateTime.fromMillisecondsSinceEpoch(timestamp);
              String formatBulan = '$tgl.month.toString().padLeft(2, '0')-$tgl.year';
              bulanTersedia.add(formatBulan);
            } catch (e) {
              print('Gagal baca tanggal: $e');
            }
          });
        }
      });
      if (bulanTersedia.isEmpty) {
        ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content: Text('Data transaksi kosong atau format gagal dibaca.'), backgroundColor: Colors.orange));
        return;
      }
      List<String> listBulanSort = bulanTersedia.toList().sort();
      String? bulanDipilih = listBulanSort.last;
```



## Lampiran 5.9 Implementasi Pengaturan Nama Pabrik



## Lampiran 5.10 Implementasi Autentikasi Keamanan dengan Biometrik/PIN

