LAMPIRAN

Lampiran 1 - MainActivity

```
package  com.example.appsampanins
import
io.flutter.embedding.android.Flu
tterActivity
classMainActivity:
FlutterActivity()
```

Lampiran 2 – Main.dart

```
import
'package:flutter/material.dar
t';
import
'package:provider/provider.da
rt';
import
'package:appsampanins/core/th
eme/app_theme.dart';
import
'package:appsampanins/di/inje
ction.dart';
import
'package:appsampanins/provide
rs/book_provider.dart';
import
'package:appsampanins/provide
rs/cart_provider.dart';
import
'package:appsampanins/provide
rs/review_provider.dart';
import
'package:appsampanins/provide
rs/wishlist_provider.dart';
import
'package:appsampanins/provide
rs/auth_provider.dart';
import
'package:appsampanins/provide
rs/blog_provider.dart';
import
'package:appsampanins/provide
rs/product_provider.dart';
import
'package:appsampanins/provide
rs/user_provider.dart';
import
'package:appsampanins/service
s/user_service.dart';
import
'package:appsampanins/present
ation/screens/cart/cart_scree
n.dart';
import
'package:appsampanins/present
ation/screens/checkout/checko
ut_screen.dart';
import
'package:appsampanins/present
ation/screens/order/order_con
firmation_screen.dart';
import
'package:appsampanins/present
ation/screens/book_detail/boo
k_detail_screen.dart';
import
'package:appsampanins/present
ation/screens/blog/blog_detai
l_screen.dart';
import
'package:appsampanins/present
ation/screens/home/home_scree
n.dart';
import
'package:appsampanins/present
ation/screens/wishlist/wishli
st_screen.dart';
import
'package:appsampanins/present
ation/screens/account/account
_screen.dart';
import
'package:appsampanins/present
ation/screens/book_list/new_b
ooks_screen.dart';
import
'package:appsampanins/present
ation/screens/auth/login_scre
en.dart';
```

```
import
'package:appsampanins/present
ation/screens/blog/blog_list_
screen.dart';
import
'package:appsampanins/present
ation/screens/catalog/catalog
_screen.dart';
import
'package:appsampanins/present
ation/widgets/main_navigation
.dart';
import
'package:appsampanins/present
ation/screens/auth/register_s
creen.dart';
import
'package:appsampanins/present
ation/screens/account/book_su
bmission_screens.dart';
import
'package:appsampanins/provide
rs/shipping_provider.dart';
import
'package:appsampanins/present
ation/screens/splash/splash_s
creen.dart';
import
'package:appsampanins/present
ation/screens/checkout/track_
shipment_screen.dart';
import
'package:logging/logging.dart
';

Future<void> main() async {

WidgetsFlutterBinding.ensureI
nitialized();

  // Konfigurasi Logger
  Logger.root.level =
Level.ALL; // Mengaktifkan
semua level log

Logger.root.onRecord.listen((
record) {

debugPrint('${record.level.na
me}: ${record.time}:
${record.message}');
  });

  setupDependencies();
  await getIt.allReady(); //
Wait for all async
dependencies
  runApp(const MyApp());
}

// Observer untuk menangani
lifecycle aplikasi
class AppLifecycleObserver
extends
WidgetsBindingObserver {
  final CartProvider
cartProvider;


AppLifecycleObserver(this.car
tProvider);

  @override
  void
didChangeAppLifecycleState(Ap
pLifecycleState state) {
    debugPrint('App lifecycle
state changed to: $state');
    if (state ==
AppLifecycleState.paused ||
state ==
AppLifecycleState.detached) {
      // Aplikasi di-pause
atau ditutup, simpan
keranjang
      debugPrint('App is
paused or closed, syncing
cart to server');

cartProvider.syncCartToServer
();
    }
  }
}

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

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

```
}

class _MyAppState extends
State<MyApp> {
  late AppLifecycleObserver
_lifecycleObserver;
  final CartProvider
_cartProvider =
getIt<CartProvider>();

  @override
  void initState() {
    super.initState();
    _lifecycleObserver =
AppLifecycleObserver(_cartPro
vider);

WidgetsBinding.instance.addOb
server(_lifecycleObserver);
  }

  @override
  void dispose() {

WidgetsBinding.instance.remov
eObserver(_lifecycleObserver)
;
    super.dispose();
  }

  @override
  Widget build(BuildContext
context) {
    return MultiProvider(
      providers: [

ChangeNotifierProvider(
          create: (_) =>
getIt<BookProvider>(),
        ),

ChangeNotifierProvider.value(
          value:
_cartProvider,
        ),

ChangeNotifierProvider(
          create: (_) =>
getIt<ReviewProvider>(),
        ),

ChangeNotifierProvider(
          create: (_) =>
WishlistProvider(),
        ),

ChangeNotifierProvider(
          create: (_) =>
getIt<AuthProvider>(),
        ),

ChangeNotifierProvider(
          create: (_) =>
BlogProvider(),
        ),

ChangeNotifierProvider(
          create: (_) =>
getIt<ProductProvider>(),
        ),

ChangeNotifierProxyProvider<A
uthProvider, UserProvider>(
          create: (_) =>
UserProvider(
            userService:
UserService(
              baseUrl:
'https://sampaninstitute.my.i
d/',
              token: '',
              userId: 0,
            ),
          ),
          update: (_, auth,
previousUserProvider) {
            // Debug info
untuk melihat status auth
provider

debugPrint('AuthProvider
update - isLoggedIn:
${auth.isLoggedIn}, token:
${auth.token != null},
userId: ${auth.id}');

            if
(auth.isLoggedIn &&
auth.token != null && auth.id
!= null && auth.id! > 0) {

debugPrint('Creating new
```

```
UserProvider with userId:
${auth.id}');
              // Jika user
sudah login dan memiliki
token dan id yang valid
              return
UserProvider(
                userService:
UserService(
                  baseUrl:
'https://sampaninstitute.my.i
d/',
                  token:
auth.token!,
                  userId:
auth.id!,
                ),
              );
            }

            // Jika user
belum login atau tidak
memiliki token/id yang valid
            // Gunakan
provider sebelumnya atau buat
yang baru dengan nilai
default
            final provider =
previousUserProvider ??
UserProvider(
              userService:
UserService(
                baseUrl:
'https://sampaninstitute.my.i
d/',
                token: '',
                userId: 0,
              ),
            );

            debugPrint('Using
default or previous
UserProvider with userId:
${provider.userService.userId
}');
            return provider;
          },
        ),

ChangeNotifierProvider(
          create: (_) =>
getIt<ShippingProvider>(),
        ),
      ],
      child: MaterialApp(
        title: 'Sampanins
Book Store',
        theme:
AppTheme.lightTheme,
        initialRoute:
'/splash', // SplashScreen
sebagai halaman pertama
        routes: {
          '/splash':
(context) => const
SplashScreen(),
          '/': (context) =>
MainNavigation(
                pages: [

HomeScreen(),

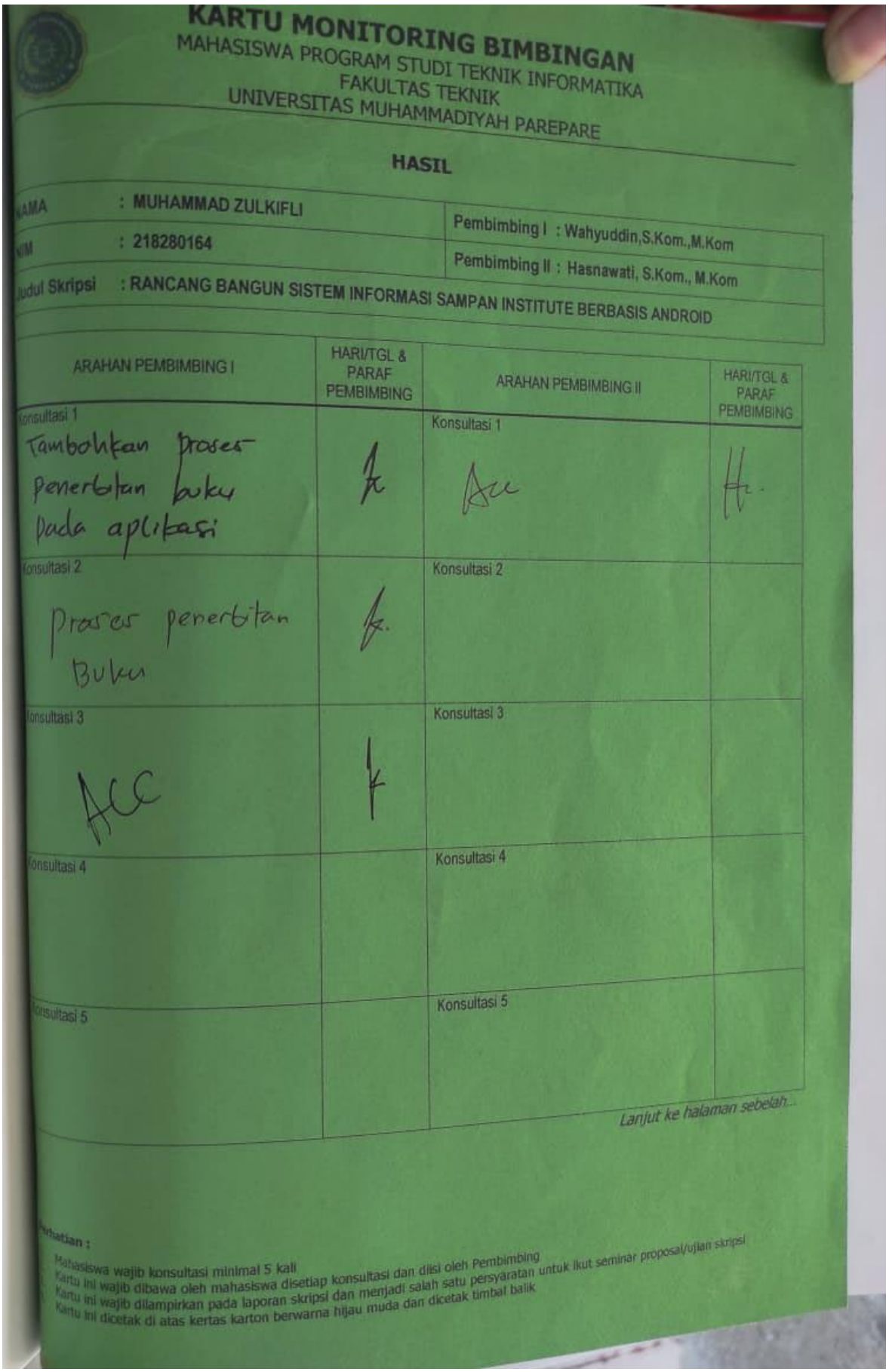
WishlistScreen(),

CartScreen(),

CatalogScreen(),

AccountScreen(),
                ],
              ),
          '/register':
(context) => const
RegisterScreen(),
          '/cart': (context)
=> const CartScreen(),
          '/checkout':
(context) => const
CheckoutScreen(),
          '/order-
confirmation': (context) =>
const
OrderConfirmationScreen(),
          '/book-detail':
(context) => const
BookDetailScreen(),
          '/new-books':
(context) => const
NewBooksScreen(),
          '/login': (context)
=> const LoginScreen(),
          '/blog-detail':
```

```
(context) => const
BlogDetailScreen(),
          '/blog-list':
(context) => const
BlogListScreen(),
          '/catalog':
(context) => const
CatalogScreen(),
          '/publish-book':
(context) => const
BookSubmissionHomeScreen(),
          '/track-shipment':
(context) => const
TrackShipmentScreen(),
        },
      ),
    );
  }
}
```



KARTU MONITORING BIMBINGAN

MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

HASIL

NAMA	: MUHAMMAD ZULKIFLI	Pembimbing I : Wahyuddin,S.Kom.,M.Kom
NIM	: 218280164	Pembimbing II : Hasnawati, S.Kom., M.Kom
Judul Skripsi	: RANCANG BANGUN SISTEM INFORMASI SAMPAN INSTITUTE BERBASIS ANDROID	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 Tambahkan proses penerbitan buku pada aplikasi		Konsultasi 1 Acc	
Konsultasi 2 Proses penerbitan Buku		Konsultasi 2	
Konsultasi 3 ACC		Konsultasi 3	
Konsultasi 4		Konsultasi 4	
Konsultasi 5		Konsultasi 5	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik