

LISTING PROGRAM

login_page.dart

```
import
'package:flutter/material.dart';
import
'package:flutter_bloc/flutter_bloc.dart';
import
'package:flutter_posresto_app/data/datasources/auth_local_data_source.dart';

import
'package:flutter_svg/flutter_svg.dart';

import
'../core/assets/assets.gen.dart';
import
'../core/components/buttons.dart';
import
'../core/components/custom_text_field.dart';

import
'../core/components/spaces.dart';
import
'../core/constants/colors.dart';
import
'../home/pages/dashboard_page.dart';
import
'bloc/login/login_bloc.dart';

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

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

class _LoginPageState
extends
State<LoginPage> {
  final emailController
  =
  TextEditingController();
  final
  passwordController =
  TextEditingController();

  @override
  void dispose() {
    emailController.dispose();
    passwordController.dispose();
    super.dispose();
  }

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

body: ListView(	fontSize:	controller:
padding: const	16,	emailController,
EdgeInsets.symmetric	fontWeight:	label: 'Email',
(horizontal: 124.0,	FontWeight.w700,	),
vertical: 20.0),	color:	const
children: [	Colors.black,	SpaceHeight(12.0),
const	),	
SpaceHeight(80.0),	),	CustomTextField(
Padding(	),	controller:
padding:	const	passwordController,
const	SpaceHeight(8.0),	label:
EdgeInsets.symmetric	const Center(	'Password',
(horizontal: 130.0),	child: Text(	obscureText:
child:	'Akses Login	true,
SvgPicture.asset(	Kasir',	),
	style:	const
Assets.icons.homeRes	TextStyle(	SpaceHeight(24.0),
to.path,	fontSize:	
width: 100,	12,	BlocListener<LoginB
height: 100,	fontWeight:	loc, LoginState>(
color:	FontWeight.w400,	listener:
AppColors.primary,	color:	(context, state) {
)),	Colors.grey,	
const	),	state.maybeWhen(
SpaceHeight(24.0),	),	orElse: ()
const Center(	),	{},
child: Text(	const	success:
'RESTO	SpaceHeight(40.0),	(authResponseModel)
WITH BAHRI',		{
style:	CustomTextField(	
TextStyle(		AuthLocalDataSource

().saveAuthData(auth	),	passwordController.te
ResponseModel);	);	xt,
	},	),
Navigator.pushReplac	);	);
ement(	},	},
context,	child:	label:
	BlocBuilder<LoginBl	'Masuk',
MaterialPageRoute(	oc, LoginState>(	);
builder:	builder:	},
(context) => const	(context, state) {	loading: ()
DashboardPage(),	return	{
),	state.maybeWhen(	return
);	orElse: () {	const Center(
},	return	child:
error:	Button.filled(	CircularProgressIndic
(message) {		ator(),
	onPressed: () {	);
ScaffoldMessenger.of		},
(context).showSnack	context.read<LoginBl	);
Bar(	oc>().add(	},
		),
SnackBar(	LoginEvent.login(	),
content:		],
Text(message),	email:	),
	emailController.text,	);
backgroundColor:		}
AppColors.red,	password:	}

dashboard_page.dart

```
import
'package:flutter/material.dart';
import
'package:flutter_bloc/flutter_bloc.dart';
import
'package:flutter_posresto_app/core/constants/colors.dart';
import
'package:flutter_posresto_app/core/extensions/build_context_ext.dart';
import
'package:flutter_posresto_app/data/datasources/auth_local_datasource.dart';
import
'package:flutter_posresto_app/presentation/auth/login_page.dart';
import
'package:flutter_posresto_app/presentation/report/pages/report_page.dart';
```

```
import
'package:flutter_posresto_app/presentation/settings/pages/settings_page.dart';
import
'package:flutter_posresto_app/presentation/settings/pages/sync_data_page.dart';
import
'../../core/assets/assets.gen.dart';
import
'../../auth/bloc/logout/logout_bloc.dart';
import
'../widgets/nav_item.dart';
import
'home_page.dart';

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

```
@override
State<DashboardPage>
> createState() =>
_DashboardPageState
();
}

class
_DashboardPageState
extends
State<DashboardPage>
> {
  int _selectedIndex =
0;

  final List<Widget>
_pages = [
    const HomePage(),
    // const
Center(child:
Text("This is page 1")),
    // const
Center(child:
Text("This is page 2")),
    const ReportPage(),
    const Center(child:
Text("This is page 3")),
```

```

//          const          children: [          Assets.icons.homeRes
Center(child:          to.path,
Text("This is page 4")),          SingleChildScrollVie
          const          w(          isActive:
SettingsPage(),          child:          _selectedIndex == 0,
//          const          ClipRRect(
ManagePrinterPage(),          onTap:      ()      =>
//          const          borderRadius:          _onItemTapped(0),
SettingsPage(),          const          ),
];          BorderRadius.horizon
          tal(right:          NavItem(
          void          Radius.circular(16.0))
_onItemTapped(int          ,          iconPath:
index) {          child:          Assets.icons.discount.
    _selectedIndex      =          SizedBox(          path,
index;          height:
    setState(() {});          context.deviceHeight
}          - 20.0,          _selectedIndex == 1,
          child:
@override          ColoredBox(          onTap:      ()      =>
void initState() {          color:          _onItemTapped(1),
    super.initState();          AppColors.primary,          ),
}          child:
          Column(          NavItem(
@override          children: [          iconPath:
Widget          NavItem(          Assets.icons.dashboar
build(BuildContext          d.path,
context) {          isActive:
    return SafeArea(          _selectedIndex == 2,
        child: Scaffold(
            body: Row(

```

```

        (context).showSnackBar
onTap: () => Bar(
),
_onItemTapped(2),
);
),
SnackBar(
Navigator.pushReplac
NavItem(
content:
ement(context,
Text(e.message),
MaterialPageRoute(b
iconPath:
backgroundColor:
uilder: (context) {
Assets.icons.setting.p
AppColors.red,
return
const
ath,
),
LoginPage();
isActive:
),
_selectedIndex == 3,
);
},
));
onTap: () => },
_onItemTapped(3),
success: (value) {
});
),
AuthLocalDataSource
child:
BlocListener<Logout
().removeAuthData();
NavItem(
Bloc, LogoutState>(
ScaffoldMessenger.of
iconPath:
listener: (context,
(context).showSnack
Assets.icons.logout.pa
state) {
Bar(
th,
state.maybeMap(
const SnackBar(
isActive: false,
orElse: () {},
content: Text('Logout
success'),
onTap: () {
error: (e) {
context
ScaffoldMessenger.of
AppColors.primary,
.read<LogoutBloc>()

```

```

),
],
.add(const
),
),
LogoutEvent.logout()
),
),
);
),
);
},
Expanded(
}
),
child:
}
),
_pages[_selectedIndex],
),
),

```

logout_page.dart

```

import
Widget
'package:flutter/material.dart';
class LogoutPage
build(BuildContext
context) {
import
StatefulWidget {
return Scaffold(
'package:flutter_bloc/f
const
appBar: AppBar(
lutter_bloc.dart';
LogoutPage({super.ke
title: const
y});
Text('Dashboard'),
import
),
'../../core/constants/c
@override
body: Center(
olors.dart';
State<LogoutPage>
child: Column(
import
createState() =>
children: [
'../../data/datasource
_LoginoutPageState();
const
s/auth_local_datasour
}
Text('Welcome to
ce.dart';
Dashboard'),
import
class
_logout_logout.dart';
_LoginoutPageState
const
extends
SizedBox(
import
State<LogoutPage> {
height: 100,
'../../auth/login_page.d
@override
),
art';

```

	success:	return
BlocListener<Logout	(value) {	const LoginPage();
Bloc, LogoutState>({		});
listener:	AuthLocalDataSource	},
(context, state) {	()removeAuthData();	);
		},
state.maybeMap(	ScaffoldMessenger.of	child:
orElse: ()	(context).showSnack	ElevatedButton(
{},	Bar(	onPressed:
error: (e) {	const	() {
	SnackBar(	
ScaffoldMessenger.of		context.read<Logout
(context).showSnack	content: Text('Logout	Bloc>().add(const
Bar(	success'),	LogoutEvent.logout()
		);
SnackBar(	backgroundColor:	},
	AppColors.primary,	child: const
content:	),	Text('Logout'),
Text(e.message),	);	),
		),
backgroundColor:	Navigator.pushReplac	],
AppColors.red,	ement(context,	),
),		),
);	MaterialPageRoute(b	);
},	uilder: (context) {	}
		}

home_page.dart

```
import
'package:flutter/material.dart';
import
'package:flutter_bloc/flutter_bloc.dart';
import
'package:flutter_posresto_app/core/extensions/build_context_ext.dart';
import
'package:flutter_posresto_app/core/extensions/int_ext.dart';
import
'package:flutter_posresto_app/core/extensions/string_ext.dart';
import
'package:flutter_posresto_app/presentation/home/bloc/local_product/local_product_bloc.dart';
import
'package:flutter_posresto_app/presentation/home/dialog/discount_dialog.dart';
```

```
import
'package:flutter_posresto_app/presentation/home/dialog/tax_dialog.dart';
import
'package:flutter_posresto_app/presentation/home/pages/confirm_payment_page.dart';
import
'../../core/assets/assets.gen.dart';
import
'../../core/components/buttons.dart';
import
'../../core/components/spaces.dart';
import
'../../core/constants/colors.dart';
import
'../bloc/checkout/checkout_bloc.dart';
import
'../dialog/service_dialog.dart';
```

```
import
'../models/product_category.dart';
import
'../models/product_model.dart';
import
'../widgets/column_button.dart';
import
'../widgets/custom_tab_bar.dart';
import
'../widgets/home_title.dart';
import
'../widgets/order_menu.dart';
import
'../widgets/product_card.dart';

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

  @override
```

State<HomePage>		child:
createState() =>	searchController.clear	Column(
_HomePageState());	();	
}	setState(() {});	mainAxisAlignment:
	}	MainAxisAlignment.s
		tart,
class _HomePageState		
extends	@override	
State<HomePage> {	Widget	children: [
final	build(BuildContext	
searchController =	context) {	HomeTitle(
TextEditingController	return Hero(	
());	tag:	controller:
	'confirmation_screen',	searchController,
@override	child: Scaffold(	
void initState() {	body: Row(	onChanged: (value) {
// searchResults =	children: [	},
products;	Expanded(	),
context	flex: 3,	const
	child: Align(	SizedBox(height: 24),
.read<LocalProductBl	alignment:	
oc>()	AlignmentDirectional	CustomTabBar(
.add(const	.topStart,	
LocalProductEvent.ge	child:	tabTitles: const [
tLocalProduct());	Padding(	
super.initState();	padding:	'Semua',
}	const	
	EdgeInsets.all(16.0),	'Makanan',
	child:	
void		'Minuman',
onCategoryTap(int	SingleChildScrollView	
index) {	w(	
		'Snack'

],	return const Center(	physics:
initialTabIndex: 0,		
tabViews: [	child:	const
	CircularProgressIndic	NeverScrollableScroll
	ator(),	Physics(),
SizedBox(		
	);	gridDelegate:
child:		
BlocBuilder<LocalPr	}, loaded: (products) {	const
oductBloc,		SliverGridDelegateWi
	if (products.isEmpty)	thFixedCrossAxisCou
LocalProductState>(	{	nt(
builder: (context,	return const Center(	childAspectRatio:
state) {		0.85,
	child: Text('No	
return	Items'),	crossAxisCount: 3,
state.maybeWhen(orE		
lse: () {	);	crossAxisSpacing:
		30.0,
return const Center(	}	
		mainAxisSpacing:
child:	return	30.0,
CircularProgressIndic	GridView.builder(	
ator(),		),
	shrinkWrap: true,	
);		itemBuilder: (context,
	itemCount:	index) =>
}, loading: () {	products.length,	
		ProductCard(

data: products[index],	child:	return
onCartButton: () {},	CircularProgressIndic	GridView.builder(
	ator(),	shrinkWrap: true,
),	);	itemCount: products
);	}, loading: () {	.where((element) =>
});	return const Center(	element.category!.id!
},		== 2)
),	child:	.toList()
	CircularProgressIndic	
	ator(),	.length,
SizedBox(	);	
child:	}, loaded: (products) {	physics:
BlocBuilder<LocalPr		const
oductBloc,	if (products.isEmpty)	NeverScrollableScroll
	{	Physics(),
LocalProductState>(		
	return const Center(	gridDelegate:
builder: (context,		const
state) {	child: Text('No	SliverGridDelegateWi
	Items'),	thFixedCrossAxisCou
return	);	nt(
state.maybeWhen(orE		
lse: () {	}	
return const Center(		

```

childAspectRatio:      );      }, loading: () {
0.85,
      });      return const Center(
crossAxisCount: 3,      },
      ),      child:
crossAxisSpacing:      ),      CircularProgressIndic
30.0,      ator(),
      SizedBox(
mainAxisSpacing:      );
30.0,      child:
      BlocBuilder<LocalPr      }, loaded: (products) {
      oductBloc,
      if (products.isEmpty)
itemBuilder: (context, LocalProductState>( {
index) =>
      builder:      (context,      return const Center(
      ProductCard(      state) {
      data: products      return
      .where((element) =>      state.maybeWhen(orE
      element.category!.id!      lse: () {
      == 2)      );
      .toList()[index],      return
      onCartButton: () {},      GridView.builder(
      ),      shrinkWrap: true,
      itemCount: products

```

.where((element) =>	mainAxisSpacing:	SizeBox(
	30.0,	
element.category!.id!		child:
== 1)	),	BlocBuilder<LocalPr
		oductBloc,
.toList()	itemBuilder: (context,	
	index) =>	LocalProductState>(
.length,		
	ProductCard(	builder: (context,
physics:		state) {
	data: products	
const		return
NeverScrollableScroll	.where((element) =>	state.maybeWhen(orE
Physics(),		lse: () {
gridDelegate:	element.category!.id!	
	== 1)	return const Center(
const	.toList()[index],	child:
SliverGridDelegateWi		CircularProgressIndic
thFixedCrossAxisCou	onCartButton: () {},	ator(),
nt(		
	),	);
childAspectRatio:		
0.85,	);	}, loading: () {
crossAxisCount: 3,	});	return const Center(
	},	
crossAxisSpacing:	),	child:
30.0,	),	CircularProgressIndic
		ator(),

);	.length,	ProductCard(
}, loaded: (products) {	physics:	data: products
if (products.isEmpty)	const	.where((element) =>
{	NeverScrollableScroll	
	Physics(),	element.category!.id!
return const Center(		== 3)
	gridDelegate:	
child: Text('No		.toList()[index],
Items'),	const	
	SliverGridDelegateWi	onCartButton: () {},
);	thFixedCrossAxisCou	
	nt(	),
}		
	childAspectRatio:	);
return	0.85,	
GridView.builder(		});
	crossAxisCount: 3,	},
shrinkWrap: true,		),
	crossAxisSpacing:	),
itemCount: products	30.0,	],
		),
.where((element) =>	mainAxisSpacing:	],
	30.0,	),
element.category!.id!		),
== 3)	),	),
		),
.toList()	itemBuilder: (context,	),
	index) =>	Expanded(

flex: 2,		
child: Align(	fontWeight:	onPressed: () {},
alignment:	FontWeight.w600,	
Alignment.topCenter,	),	label: 'To Go',
child: Stack(	),	),
children: [	const	],
	SpaceHeight(8.0),	),
SingleChildScrollView	Row(	const
w(		SpaceHeight(16.0),
padding:	children: [	const
const		Row(
EdgeInsets.all(24.0),	Button.filled(	
child:		mainAxisAlignment:
Column(	width: 120.0,	MainAxisAlignment.s
		paceBetween,
crossAxisAlignment:	height: 40,	
CrossAxisAlignment.		children: [
start,	onPressed: () {},	
		Text(
children: [	label: 'Dine In',	
const	),	'Item',
Text(		
	const	style: TextStyle(
'Orders #1',	SpaceWidth(8.0),	
		color:
style: TextStyle(	Button.outlined(	AppColors.primary,
color:	width: 100.0,	fontSize: 16,
AppColors.primary,		
	height: 40,	fontWeight:
fontSize: 20,		FontWeight.w600,

```

        ),
        child: Text(
          builder: (context, state) {
            return
            state.maybeWhen(
              color:
              AppColors.primary,
              Center(
                width: 50.0,
                fontSize: 16,
                child: Text('No
                Items'),
                child: Text(
                  fontWeight:
                  FontWeight.w600,
                  'Qty',
                  loaded:
                    ),
                    style: TextStyle(
                      ),
                      (products, discount,
                      tax, serviceCharge) {
                        color:
                        ],
                        if (products.isEmpty)
                        {
                          fontSize: 16,
                          SpaceHeight(8),
                          const
                          return const Center(
                            fontWeight:
                            Divider(),
                            const
                            child: Text('No
                            Items'),
                            ),
                            ),
                            BlocBuilder<Checkou
                            );
                            tBloc,
                            }
                            CheckoutState>(
                          SizedBox(

```

```

    ),
    return const ColumnButton(
    ListView.separated(      SpaceHeight(8.0),
                             Row(      label: 'Pajak',
    shrinkWrap: true,
                             mainAxisAlignment:      svgGenImage:
    physics:      MainAxisAlignment.spaceEvenly,      Assets.icons.pajak,
    const      onPressed: () =>
    NeverScrollableScroll      children: [      showDialog(
    Physics(),
                             ColumnButton(      context: context,
    itemBuilder: (context,
    index) =>      label: 'Diskon',      builder: (context) =>
    OrderMenu(data:      svgGenImage:      ),
    products[index]),      Assets.icons.diskon,      ),
    separatorBuilder:      onPressed: () =>      ColumnButton(
    (context, index) =>      showDialog(
    const      context: context,
    SpaceHeight(1.0),
                             barrierDismissible:      svgGenImage:
    itemCount:      false,      Assets.icons.layanan,
    products.length,
    onPressed: () =>
    );      builder: (context) =>      showDialog(
    const      DiscountDialog(),      context: context,
    ),      ),
    ),      ),
    },      ),
    },      ),

```

File 1: main.dart	File 2: checkout.dart	File 3: service.dart
builder: (context) =>	CheckoutState>(	style: const TextStyle(
const ServiceDialog(),		
),	builder: (context,	color:
),	state) {	AppColors.primary,
],		
),	final tax =	fontWeight:
const	state.maybeWhen(	FontWeight.w600,
SpaceHeight(8.0),		
const	orElse: () => 0,	),
Divider(),		
const	loaded: (products,	);
SpaceHeight(8.0),	discount, tax,	},
Row(		),
	serviceCharge) {	],
mainAxisAlignment:		),
MainAxisAlignment.s	if (products.isEmpty)	const
paceBetween,	{	SpaceHeight(8.0),
		Row(
children: [	return 0;	
		mainAxisAlignment:
const Text(	}	MainAxisAlignment.s
		paceBetween,
'Pajak',	return tax;	
		children: [
style: TextStyle(color:	});	
AppColors.grey),		const Text(
),	return Text(	
		'Diskon',
BlocBuilder<Checkou	'\$tax %',	

style: TextStyle(color: AppColors.grey),	.toIntegerFromText;	children: [
),	});	const Text(
BlocBuilder<CheckoutBloc,	return Text(	'Sub total',
CheckoutState>(builder: (context, state) {	'\$discount %',	style: TextStyle(color: AppColors.grey),
	style: const TextStyle(	),
	color:	BlocBuilder<CheckoutBloc,
final discount = state.maybeWhen(	AppColors.primary,	tBloc,
	fontWeight:	CheckoutState>(builder: (context, state) {
orElse: () => 0,	FontWeight.w600,	
loaded: (products,	),	
discount, tax,		final price = state.maybeWhen(
	);	
serviceCharge) {	},	orElse: () => 0,
	),	
if (discount == null) {	],	loaded: (products,
	),	discount, tax,
return 0;	const	
	SpaceHeight(8.0),	serviceCharge) {
}	Row(	
		if (products.isEmpty)
return discount.value!	mainAxisAlignment:	{
	MainAxisAlignment.s	
.replaceAll('.00', '')	paceBetween,	

```

return 0;                                color:                                padding:                                const
                                        AppColors.primary,                    EdgeInsets.symmetric
}
                                        fontWeight:
return products                          FontWeight.w600,                    horizontal:                24.0,
                                        vertical: 16.0),
.map((e) =>                                ),                                child:
                                        Button.filled(
e.product.price!                            );
                                        },                                onPressed: () {
.toIntegerFromText *                        ),                                context.push(const
                                        ],                                ConfirmPaymentPage
e.quantity)                                ),                                ()
                                        const                                });
.reduce((value,                            SpaceHeight(100.0),                    },
element) =>                                ],
                                        ),                                label:                'Lanjutkan
value + element);                        ),                                Pembayaran',
                                        Align(                                ),
});
                                        alignment:                                ),
                                        Alignment.bottomCen                    ),
return Text(                            ter,                                ],
                                        child:                                ),
price.currencyFormat                    ColoredBox(                                ),
Rp,                                    color:                                ),
                                        AppColors.white,                    ],
style: const TextStyle(                    child:                                ),
                                        Padding(                                ),
                                        );

```

}	child: Column(	'Belum Ada
}		Produk',
	mainAxisAlignment:	textAlign:
class _IsEmpty	MainAxisAlignment.c	TextAlign.center,
extends	enter,	style:
StatelessWidget {	children: [	TextStyle(fontSize:
const _IsEmpty();		24),
	Assets.icons.noProdu	),
@override	ct.svg(),	],
Widget	const	),
build(BuildContext	SizedBox(height:	);
context) {	80.0),	}
return Center(	const Text(	}

confirm_payment_page.dart

import	import	home/bloc/order/order
'package:flutter/materi	'package:flutter_posre	_bloc.dart';
al.dart';	sto_app/core/extensio	import
import	ns/int_ext.dart';	'package:flutter_posre
'package:flutter_bloc/f	import	sto_app/presentation/
lutter_bloc.dart';	'package:flutter_posre	home/models/product
import	sto_app/core/extensio	_quantity.dart';
'package:flutter_posre	ns/string_ext.dart';	
sto_app/core/extensio	import	import
ns/build_context_ext.	'package:flutter_posre	'../../core/assets/asse
dart';	sto_app/presentation/	ts.gen.dart';

import	extends	category:
'../../core/component	StatefulWidget {	ProductCategory.drin
s/buttons.dart';	const	k,
import	ConfirmPaymentPage	price: 200000,
'../../core/component	({super.key});	stock: 10),
s/spaces.dart';		ProductModel(
import	@override	image:
'../../core/constants/c		Assets.images.product
olors.dart';	State<ConfirmPayme	2.path,
import	ntPage> createState()	name: 'V60',
'../bloc/checkout/chec	=>	category:
kout_bloc.dart';	_ConfirmPaymentPag	ProductCategory.drin
import	eState();	k,
'../models/product_cat	}	price: 1200000,
egory.dart';		stock: 10),
import	class	];
'../models/product_mo	_ConfirmPaymentPag	@override
del.dart';	eState extends	Widget
import	State<ConfirmPayme	build(BuildContext
'../widgets/column_bu	ntPage> {	context) {
ttion.dart';	final	return SafeArea(
import	totalPriceController =	child: Hero(
'../widgets/order_men	TextEditingController	tag:
u.dart';	());	'confirmation_screen',
import	final products = [	child: Scaffold(
'../widgets/success_pa	ProductModel(	body: Row(
yment_dialog.dart';	image:	children: [
	Assets.images.product	// LEFT
	1.path,	CONTENT
class		
ConfirmPaymentPage	name: 'Vanila	Expanded(
	Late Vanila itu',	flex: 2,

```

        child:
Align(                                children: [                                ),
        alignment:                                ),
Alignement.topCenter,                Text(                                ],
        child:                                ),
SingleChildScrollView                'Konfirmasi',
w(                                    GestureDetector(
        padding:                                style: TextStyle(
const                                onTap: () {},
EdgeInsets.all(24.0),                color:
        child:                                AppColors.primary,                child: Container(
Column(                                fontSize: 20,                                padding:                                const
crossAxisAlignment:                                EdgeInsets.all(16.0),
CrossAxisAlignment.                    fontWeight:
start,                                FontWeight.w600,                                height: 60.0,

children: [                                ),                                width: 60.0,

        Row(                                ),                                decoration:                                const
mainAxisAlignment:                    Text(                                BoxDecoration(
MainAxisAlignment.s                    'Orders #1',                                color:
paceBetween,                            AppColors.primary,

children: [                                style: TextStyle(
const Column(                            fontSize: 16,
crossAxisAlignment:                    fontWeight:
CrossAxisAlignment.                    FontWeight.w500,                                ),
start,

```

[illegible]

```

const
Divider(), child: Text('No
const Items'),
SpaceHeight(8),
); const
BlocBuilder<Checkou } SpaceHeight(16.0),
tBloc, // Row(
CheckoutState>( return //
ListView.separated( mainAxisAlignment:
builder: (context, MainAxisAlignment.s
state) { shrinkWrap: true, paceEvenly,
//
return physics: const children: [
state.maybeWhen( NeverScrollableScroll //
Physics(), ColumnButton(
//
orElse: () => const label: 'Diskon',
Center( itemBuilder: (context, //
index) =>
child: Text('No svgGenImage:
Items'), OrderMenu(data: Assets.icons.diskon,
), products[index]), //
loaded: (products, separatorBuilder: onPressed: () =>
discount, tax, service) (context, index) => showDialog(
{ //
context: context,
//
if const
(products.isEmpty) { SpaceHeight(16.0), builder: (context) =>
const
return const Center( itemCount: DiscountDialog(),
products.length, // ),
); // ),

```

<pre> // ColumnButton(// label: 'Pajak', // svgGenImage: Assets.icons.pajak, // onPressed: () => showDialog(// context: context, // builder: (context) => const TaxDialog(), //), //), // ColumnButton(// label: 'Layanan', // svgGenImage: Assets.icons.layanan, // onPressed: () => showDialog(// context: context, </pre>	<pre> // builder: (context) => const ServiceDialog(), //), //), //], //), const SpaceHeight(8.0), const Divider(), const SpaceHeight(8.0), Row(mainAxisAlignment: MainAxisAlignment.s paceBetween, children: [const Text('Pajak', style: TextStyle(color: AppColors.grey),), </pre>	<pre> tBloc, CheckoutState>(builder: (context, state) { final tax = state.maybeWhen(orElse: () => 0, loaded: (products, discount, tax, service) => tax,); final price = state.maybeWhen(orElse: () => 0, loaded: (products, discount, tax, service) => products.fold(0, </pre>
---	--	--

BlocBuilder<Checkou

```

(previousValue,
element) =>
    return discount.value!
previousValue +
    .replaceAll('.00', '')
(element.product.price!
    .toIntegerFromText;
    .toIntegerFromText *
    });
element.quantity),
    final subTotal =
    ),
    );
    price - (discount / 100
    * price);
    final discount =
    state.maybeWhen(
    final finalTax =
    subTotal * 0.11;
    orElse: () => 0,
    return Text(
    Text(
    loaded: (products,
    discount, tax,
    '$tax %
    ({finalTax.toInt().currencyFormatRp})),
    'Diskon',
    serviceCharge) {
    style: TextStyle(color:
    AppColors.grey),
    style: const TextStyle(
    ),
    if (discount == null) {
    color:
    BlocBuilder<CheckoutBloc,
    return 0;
    AppColors.primary,
    tBloc,
    CheckoutState>(

```

builder: (context, state) {	orElse: () => 0,	return Text(
	loaded:	finalDiscount.toInt().c
final discount =	(products, discount,	urrencyFormatRp,
state.maybeWhen(	tax, service) =>	style: TextStyle(
orElse: () => 0,	products.fold(	color:
loaded: (products,	0,	AppColors.primary,
discount, tax,		fontWeight:
serviceCharge) {	(previousValue,	FontWeight.w600,
	element) =>	
if (discount == null) {		),
	previousValue +	);
return 0;		},
	(element.product.pric	),
}	e!	],
		),
return discount.value!	.toIntegerFromText *	const
		SpaceHeight(16.0),
.replaceAll('.00', '')	element.quantity),	Row(
.toIntegerFromText;	));	mainAxisAlignment:
		MainAxisAlignment.s
});		paceBetween,
	final finalDiscount =	
	discount / 100 *	children: [
final subTotal =	subTotal;	
state.maybeWhen(		const Text(

```

const
'Sub total', previousValue + SpaceHeight(16.0),
Row(
style: TextStyle(color: (element.product.pric
AppColors.grey), e!
mainAxisAlignment:
), MainAxisAlignment.s
paceBetween,
BlocBuilder<Checkou
tBloc, element.quantity), children: [
CheckoutState>(
)); const Text(
builder: (context,
state) { return Text(
'Total',
final price = price.currencyFormat
state.maybeWhen( Rp,
style: TextStyle(
color:
orElse: () => 0, style: const TextStyle( AppColors.grey,
loaded: color: fontWeight:
AppColors.primary, FontWeight.bold,
(products, discount,
tax, service) => fontWeight:
FontWeight.w600,
fontSize: 16),
products.fold(
),
BlocBuilder<Checkou
0, ); tBloc,
}, CheckoutState>(
(previousValue, ),
builder: (context,
element) => ], state) {
),

```

final price =	final discount =	final tax = subTotal *
state.maybeWhen(	state.maybeWhen(	0.11;
orElse: () => 0,	orElse: () => 0,	final total = subTotal +
		tax;
loaded: (products,	loaded: (products,	
discount, tax, service)	discount, tax,	
=>	serviceCharge) {	totalPriceController.te
		xt =
products.fold(	if (discount == null) {	total.ceil().toString();
0,		
(previousValue,	return 0;	return Text(
element) =>	}	total.ceil().currencyFo
		rmatRp,
previousValue +	return discount.value!	
		style: const TextStyle(
(element.product.pric	.replaceAll('.00', '')	color:
e!		AppColors.primary,
	.toIntegerFromText;	
.toIntegerFromText *	});	fontWeight:
		FontWeight.w600,
element.quantity),		
	final subTotal =	fontSize: 16,
),		
	price - (discount / 100	),
	* price);	);
		},

```

        ),
    ],
    ),
    // const
    SpaceHeight(20.0),
    //
    Button.filled(
        //
        onPressed: () {},
        //
        label: 'Lanjutkan
        Pembayaran',
        // ),
    ],
    ),
    ),
    ),
    ),
    'Pembayaran',

    // RIGHT
    CONTENT
    Expanded(
        flex: 3,
        child:
    Align(
        alignment:
    Alignment.topCenter,
        child:
    Stack(
        children:
    [
        ),
        SingleChildScrollView
        w(
        '1
        opsi pembayaran
        tersedia',
        style: TextStyle(
        fontSize: 16,
        fontWeight:
        FontWeight.w500,
        ),
        children: [
        ),
        const Text(
        const
        SpaceHeight(8.0),
        const Divider(),
        style: TextStyle(
        const
        SpaceHeight(8.0),
        const Text(
        'Metode Bayar',
        style: TextStyle(
        color:
        AppColors.primary,

```

fontSize: 20,	width: 120.0,	fontSize: 16,
fontWeight:	height: 50.0,	fontWeight:
FontWeight.w600,		FontWeight.w500,
),	onPressed: () {},	),
),	label: 'QRIS',	),
const		const
SpaceHeight(12.0),	disabled: true,	SpaceHeight(12.0),
	),	
Row(	],	TextFormField(
	),	
children: [		controller:
	const	totalPriceController,
Button.filled(	SpaceHeight(8.0),	
		keyboardType:
width: 120.0,	const Divider(),	TextInputType.numbe
		r,
height: 50.0,	const	
	SpaceHeight(8.0),	decoration:
onPressed: () {},		InputDecoration(
	const Text(	
label: 'Cash',		border:
),	'Total Bayar',	OutlineInputBorder(
const	style: TextStyle(	borderRadius:
SpaceWidth(8.0),		BorderRadius.circular
	color:	(8.0),
Button.outlined(	AppColors.primary,	),

```

        ),
        child:
        hintText: 'Total harga',
        Padding(
        ),
        const
        ),
        SpaceWidth(20.0),
        padding: const
        EdgeInsets.symmetric
        const
        Button.filled(
        (
        SpaceHeight(45.0),
        width: 150.0,
        horizontal: 24.0,
        Row(
        vertical: 16.0),
        onPressed: () {},
        child: Row(
        children: [
        label: 'Rp 300.000',
        Button.filled(
        ),
        children: [
        ],
        width: 150.0,
        ),
        Flexible(
        onPressed: () {},
        const
        child:
        SpaceHeight(100.0),
        Button.outlined(
        label: 'UANG PAS',
        ],
        ),
        onPressed: () =>
        context.pop(),
        const
        Align(
        child:
        SpaceWidth(20.0),
        label: 'Batalan',
        alignment:
        Alignment.bottomCen
        ),
        Button.filled(
        Alignment.bottomCen
        ter,
        ),
        width: 150.0,
        child:
        ColoredBox(
        const
        onPressed: () {},
        color:
        SpaceWidth(8.0),
        AppColors.white,
        BlocBuilder<Checkou
        label: 'Rp 250.000',

```

```

tBloc,
CheckoutState>(      final      price      =      // final discount =
                    state.maybeWhen(      state.maybeWhen(

builder:      (context,
state) {      orElse: () => 0,      // orElse: () => 0,

final discount =      loaded:      // loaded: (products,
state.maybeWhen(      discount, tax,
                    (products, discount,
orElse: () => 0,      tax, service) =>      // serviceCharge)
                    {

loaded:      (products,      products.fold(
discount, tax,      // if (discount ==
                    0,      null) {

serviceCharge) {      (previousValue,      // return 0;

if (discount == null) {      element) =>      // }

return 0;      previousValue +      // return

}      (element.product.pric      discount.value!
e!

return discount.value!      //

.toIntegerFromText *      .replaceAll('.00', '')

.replaceAll('.00', '')      //

.toIntegerFromText;      .toIntegerFromText;

),

});      // });

);

```

```

final subTotal = price - (discount / 100 * price);
final totalDiscount = discount / 100 * price;
final finalTax = subTotal * 0.11;

List<ProductQuantity> items = state.maybeWhen(
  orElse: () => [],
  loaded: (products, discount, tax, service) =>
    products,
);

final totalQty = items.fold(
  0,
  (previousValue, element) =>
    previousValue + element.quantity,
);

final totalPrice = subTotal + finalTax;

await showDialog(
  context: context,
  barrierDismissible: false,
  builder: (context) =>
    SuccessPaymentDialog(
      data: items,
      totalQty: totalQty,

```

```

),
totalPrice: ),
totalPrice.toInt(), ),
);
totalTax: ],
finalTax.toInt(), },
),
totalDiscount: label: 'Bayar', ),
],
totalDiscount.toInt(), ),
),
subTotal: );
subTotal.toInt(), );
}, }
normalPrice: price, ), }
],

```

report_page.dart

import	import	import
'package:flutter/material.dart';	'package:flutter_posresto_app/presentation/report/bloc/transaction_report_bloc.dart';	'../../core/components/dashed_line.dart';
import		import
'package:flutter_bloc/flutter_bloc.dart';		'../../core/components/spaces.dart';
import	import	import
'package:flutter_posresto_app/core/extensions/date_time_ext.dart';	'../../core/components/custom_date_picker.dart';	'../widgets/report_menu.dart';
;		import
		'../widgets/report_title.dart';

	Widget	CrossAxisAlignment.
class ReportPage	build(BuildContext	start,
extends	context) {	children: [
StatefulWidget {	String	const
const	searchDateFormatted	ReportTitle(),
ReportPage({super.ke	=	Padding(
y});	'\${fromDate.toFormat	padding:
	tedDate2()}' to	const
@override	`\${toDate.toFormatted	EdgeInsets.all(20.0),
State<ReportPage>	Date2()');	child:
createState() =>	return Scaffold(	Row(
_ReportPageState());	body: Row(	mainAxisAlignment:
}	children: [	MainAxisAlignment.s
	// LEFT	paceBetween,
class	CONTENT	
_ReportPageState	Expanded(	children: [
extends	flex: 3,	
State<ReportPage> {	child: Align(	Flexible(
int selectedMenu = 0;	alignment:	
String title =	Alignment.topLeft,	child:
'Summary Sales	child:	CustomDatePicker(
Report';	SingleChildScrollView(	
DateTime fromDate	w(	prefix: const
=	padding:	Text('From: '),
DateTime.now().subtr	const	
act(const	EdgeInsets.all(24.0),	initialDate: fromDate,
Duration(days: 30));	child:	
DateTime toDate =	Column(	onDateSelected:
DateTime.now());		(selectedDate) {
	crossAxisAlignment:	
@override		

```

),
fromDate = ), TransactionReportEve
selectedDate; Padding( nt.getReport(
padding:
setState(() {}); const startDate:
}, EdgeInsets.all(15.0), DateTime.now(),
), child:
), Wrap( endDate:
const DateTime.now()
SpaceWidth(100.0), children: [
.subtract(const
Flexible( ReportMenu( Duration(days: 30))));
},
child: label: 'Transaction
CustomDatePicker( Report', isActive:
selectedMenu == 1,
prefix: const Text('To: onPressed: () {
}),
selectedMenu = 1; ReportMenu(
initialDate: toDate,
title = 'Transaction label: 'Item Sales
onDateSelected: Report'; Report',
(selectedDate) { setState(() {}); onPressed: () {
toDate = selectedDate; //enddate is 1 month selectedMenu = 4;
setState(() {}); before the current date
title = 'Item Sales
}, context.read<Transact Report';
), ionReportBloc>().add
), ( setState(() {});
], (

```

},		padding:
	selectedMenu = 0;	const
isActive:		EdgeInsets.all(24.0),
selectedMenu == 4,	title = 'Summary Sales	child:
),	Report';	
		BlocBuilder<Transact
ReportMenu(	setState(() {});	ionReportBloc,
	},	TransactionReportStat
label: 'Daily Sales		e>(builder:
Report',	isActive:	(context, state) {
	selectedMenu == 0,	final
onPressed: () {	),	totalRevenue =
	],	state.maybeMap(orElse:
selectedMenu = 5;	),	() => 0,
	],	loaded:
title = 'Daily Sales	),	(value) {
Report';	),	return
	),	value.transactionRepo
setState(() {});	),	rt.fold(0,
},		
isActive:	// RIGHT	
selectedMenu == 5,	CONTENT	(previousValue,
),	Expanded(flex: 2,	element) =>
	child: Align(alignment:	previousValue +
ReportMenu(	Alignment.topCenter,	element.total,
label: 'Summary Sales	child:	);
Report',	SingleChildScrollView	},
	w();	
onPressed: () {		

		return	},
	final	value.transactionRepo	);
subTotal	=	rt.fold(	return
state.maybeMap(		0,	state.maybeWhen(orE
orElse:			lse: () {
() => 0,		(previousValue,	return
loaded:		element) =>	const Center(
(value) {			child:
return		previousValue +	Text('No Data'),
value.transactionRepo		element.discount,	);
rt.fold(		);	},
0,		},	loading: () {
		);	return
(previousValue,			const Center(
element) =>		final tax =	child:
		state.maybeMap(	CircularProgressIndic
previousValue +		orElse:	ator(),
element.subTotal,		() => 0,	);
);		loaded:	}, loaded:
},		(value) {	(transactionReport) {
);		return	return
		value.transactionRepo	Column(
final		rt.fold(	
discount	=	0,	crossAxisAlignment:
state.maybeMap(			CrossAxisAlignment.
orElse:		(previousValue,	start,
() => 0,		element) =>	
loaded:			children: [
(value) {		previousValue +	
		element.tax,	Center(
		);	

child: Text(	Text('REVENUE : Row(	
	\$totalRevenue'),	
title,		mainAxisAlignment:
	const	MainAxisAlignment.s
style: const TextStyle(	SpaceHeight(8.0),	paceBetween,
fontWeight:	const DashedLine(),	children: [
FontWeight.w600,		
fontSize: 16.0),	const DashedLine(),	Text('Discount'),
	const	Text('\$discount'),
	SpaceHeight(8.0),	],
Center(		),
	Row(	
child: Text(		const
	mainAxisAlignment:	SpaceHeight(4.0),
searchDateFormatted,	MainAxisAlignment.s	
	paceBetween,	Row(
style: const		
TextStyle(fontSize:	children: [	mainAxisAlignment:
16.0),		MainAxisAlignment.s
	Text('Subtotal'),	paceBetween,
	const	children: [
SpaceHeight(16.0),	Text('\$subTotal'),	
	],	
	),	Text('Tax'),
//		
REVENUE INFO	const	Text('\$tax'),
	SpaceHeight(4.0),	],
...[		),

	//	//
const	const	const DashedLine(),
SpaceHeight(8.0),	Text('PAYMENT'),	//
	//	const DashedLine(),
const DashedLine(),	const	//
	SpaceHeight(8.0),	const
const DashedLine(),	//	SpaceHeight(8.0),
	const DashedLine(),	//
const	//	const Row(
SpaceHeight(8.0),	const DashedLine(),	//
	//	mainAxisAlignment:
Row(	const	MainAxisAlignment.s
	SpaceHeight(8.0),	paceBetween,
mainAxisAlignment:	//	//
MainAxisAlignment.s	const Row(	children: [
paceBetween,	//	//
	mainAxisAlignment:	Text('TOTAL'),
children: [	MainAxisAlignment.s	//
	paceBetween,	Text('0'),
Text('TOTAL'),	//	//],
	children: [	//),
Text('\$totalRevenue'),	//	//],
],	Text('Cash'),	],
),	//	);
],	Text('0'),	});
const	//],	},
SpaceHeight(32.0),	//),	),
	//	),
//	const	),
PAYMENT INFO	SpaceHeight(8.0),	),
// ...[		],

),
);

}
}