Adellia Alma Vimaylakirti

A11.2022.14196

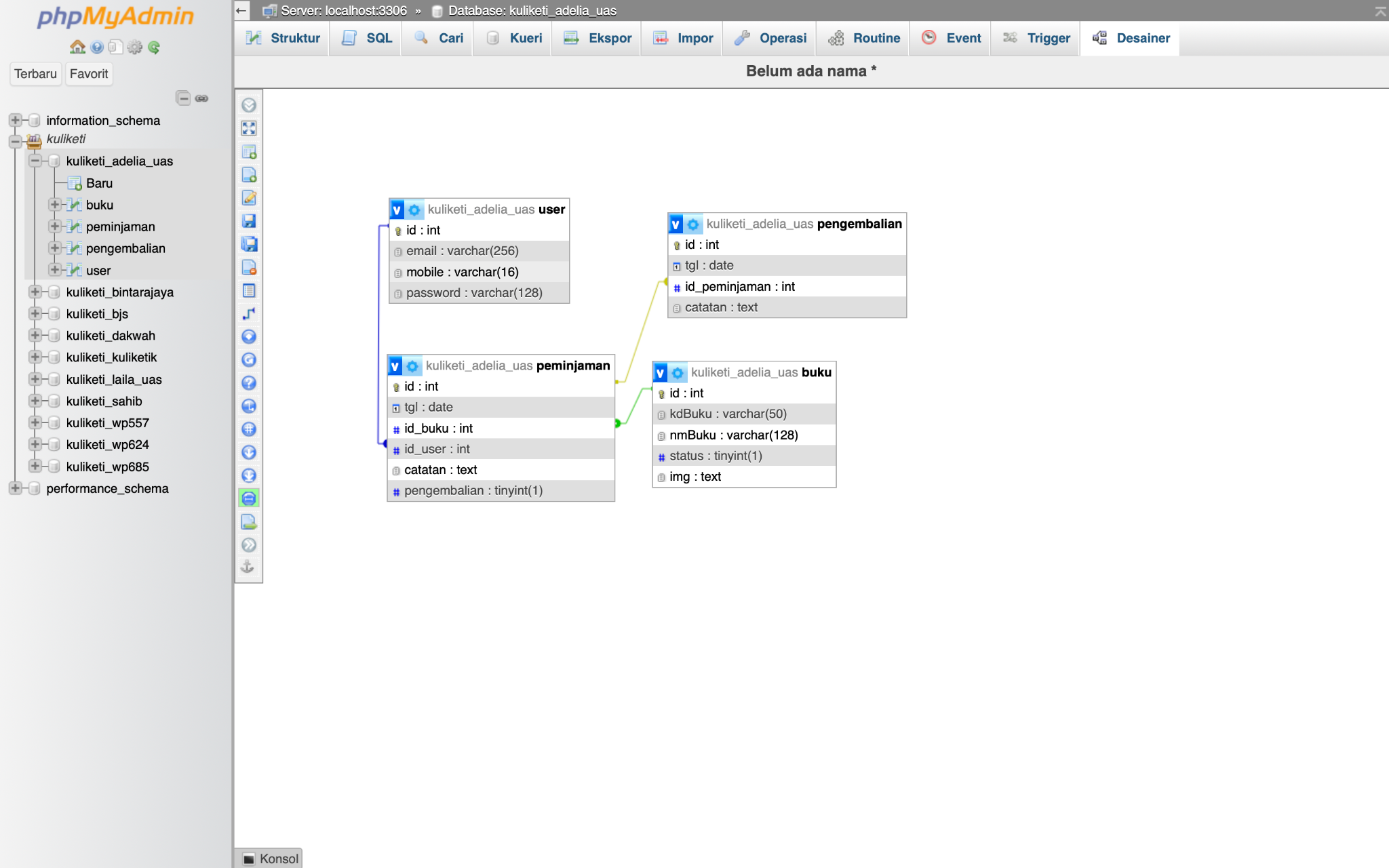
A11.4704

File2 : <https://drive.google.com/drive/folders/1-z6jH1kd568jiiLE2GM0Z6mWjWgTGseA?usp=share_link>

User Interface Result

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Structure Database



Source Code Restful API Web Based

buku/all.php

<?php

require '../koneksi.php';

$data = mysqli\_query($conn, "select \* from buku");

$data =mysqli\_fetch\_all($data, MYSQLI\_ASSOC);

echo json\_encode($data);

?>

buku/changeStatus.php

<?php

require '../koneksi.php';

$status = $\_POST['status'];

$id = $\_POST['id'];

$result = mysqli\_query($conn, "update buku set status=$status

where id=$id"); if($result){

echo json\_encode([

'status' => true,

'message' => 'Buku berhasil diupdate'

]); }else{

echo json\_encode([

'status' => false,

'message' => 'Buku gagal diupdate']); }

?>

buku/create.php

<?php

require '../koneksi.php';

$kdBuku = $\_POST['kdBuku'];

$nmBuku = $\_POST['nmBuku'];

$status = $\_POST['status'];

$img = $\_POST['img'];

$result = mysqli\_query($conn, "insert into buku set

kdBuku='$kdBuku', nmBuku='$nmBuku', status=$status, img='$img'");

if($result){

echo json\_encode([

'res' => true,

'msg' => 'Buku berhasil diinput'

]); }else{

echo json\_encode([

'res' => false,

'msg' => 'Buku gagal diinput'

]); }

?>

buku/delete.php

<?php

require '../koneksi.php';

$id = $\_POST['id'];

$result = mysqli\_query($conn, "delete from buku where

id=".$id);

if($result){

echo json\_encode([

'status' => true,

'message' => 'Buku terhapus'

]); }else{

echo json\_encode([

'status' => false,

'message' => 'Buku gagal terhapus'

]); }

?>

buku/update.php

<?php

require '../koneksi.php';

$kdBuku = $\_POST['kdBuku'];

$nmBuku = $\_POST['nmBuku'];

$status = $\_POST['status'];

$img = $\_POST['img'];

$id = $\_POST['id'];

$result = mysqli\_query($conn, "update buku set

kdBuku='$kdBuku', nmBuku='$nmBuku', status=$status, img='$img'

where id='$id'"); if($result){

echo json\_encode([

'status' => true,

'message' => 'Buku berhasil diupdate'

]); }else{

echo json\_encode([

'status' => false,

'message' => 'Buku gagal diupdate']); }

?>

peminjaman/all.php

<?php

require '../koneksi.php';

$data = mysqli\_query($conn, "select buku.\*, peminjaman.\*, buku.id as id\_buku from peminjaman, buku where peminjaman.id\_buku = buku.id AND peminjaman.id\_user=".$\_POST['id\_user']);

$data =mysqli\_fetch\_all($data, MYSQLI\_ASSOC);

echo json\_encode($data);

?>

peminjaman/cek\_pengembalian.php

<?php

require '../koneksi.php';

$data = mysqli\_query($conn, "select \* from pengembalian where id\_peminjaman=".$\_POST['id']);

$data =mysqli\_fetch\_array($data, MYSQLI\_ASSOC);

echo json\_encode($data);

?>

peminjaman/create.php

<?php

require '../koneksi.php';

$id\_buku = $\_POST['id\_buku'];

$id\_user = $\_POST['id\_user'];

$tgl = $\_POST['tgl'];

$catatan = $\_POST['catatan'];

$result = mysqli\_query($conn, "insert into peminjaman set

id\_buku=$id\_buku, id\_user=$id\_user, tgl='$tgl', catatan='$catatan'");

if($result){

mysqli\_query($conn, "update buku set status=1

where id=$id\_buku");

echo json\_encode([

'status' => true,

'message' => 'Peminjaman berhasil'

]); }else{

echo json\_encode([

'status' => false,

'message' => 'Peminjaman gagal'

]); }

?>

peminjaman/delete.php

<?php

require '../koneksi.php';

$id = $\_POST['id'];

$id\_buku = $\_POST['id\_buku'];

$result = mysqli\_query($conn, "delete from peminjaman where

id=".$id);

if($result){

mysqli\_query($conn, "update buku set status=0

where id=$id\_buku");

mysqli\_query($conn, "delete from pengembalian where

id\_peminjaman=".$id);

echo json\_encode([

'status' => true,

'message' => 'Peminjaman terhapus'

]); }else{

echo json\_encode([

'status' => false,

'message' => 'Peminjaman gagal terhapus'

]); }

?>

peminjaman/update.php

<?php

require '../koneksi.php';

$id\_buku = $\_POST['id\_buku'];

$id\_user = $\_POST['id\_user'];

$tgl = $\_POST['tgl'];

$id = $\_POST['id'];

$result = mysqli\_query($conn, "update peminjaman set

id\_buku=$id\_buku, id\_user=$id\_user, tgl='$tgl', catatan='$catatan'

where id='$id'"); if($result){

echo json\_encode([

'status' => true,

'message' => 'Peminjaman berhasil diupdate'

]); }else{

echo json\_encode([

'status' => false,

'message' => 'Peminjaman gagal diupdate']); }

?>

pengembalian/create.php

<?php

require '../koneksi.php';

$id\_peminjaman = $\_POST['id\_peminjaman'];

$tgl = $\_POST['tgl'];

$catatan = $\_POST['catatan'];

$result = mysqli\_query($conn, "insert into pengembalian set

id\_peminjaman=$id\_peminjaman, tgl='$tgl', catatan='$catatan'");

if($result){

mysqli\_query($conn, "update peminjaman set

pengembalian = 1 where id=$id\_peminjaman");

echo json\_encode([

'status' => true,

'message' => 'Pengembalian berhasil'

]); }else{

echo json\_encode([

'status' => false,

'message' => 'Pengembalian gagal'

]); }

?>

user/login.php

<?php

require '../koneksi.php';

$email = $\_POST['email'];

$password = $\_POST['password'];

$res = mysqli\_query($conn, "select \* from user");

// $res = mysqli\_fetch\_array($res, MYSQLI\_ASSOC);

$a =0;

$d;

foreach($res as $r){

if($r['email'] == $email && $r['password'] == $password){

$a++;

$d=$r;

}

}

if($a>0){

echo json\_encode([

'status' => TRUE,

'message' => 'Login berhasil',

'data' => $d

]);

}else{

echo json\_encode([

'status' => FALSE,

'message' => 'Login gagal'

]);

}

?>

user/register.php

<?php

require '../koneksi.php';

$email = $\_POST['email'];

$mobile = $\_POST['mobile'];

$password = $\_POST['password'];

$result = mysqli\_query($conn, "insert into user set

email='$email', mobile='$mobile',password='$password'");

if($result){

echo json\_encode([

'status' => true,

'message' => 'Registrasi berhasil'

]); }else{

echo json\_encode([

'status' => false,

'message' => 'Registrasi gagal'

]); }

?>

koneksi.php (lokal)

<?php

$conn = new mysqli("localhost", "root", "root", "adelia\_uas");

?>

Source Code Flutter User Interface

detail\_peminjaman.dart

import 'package:adellia\_uas/login.dart';

import 'package:adellia\_uas/service.dart';

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

import 'package:flutter/widgets.dart';

import 'package:intl/intl.dart';

// ignore: must\_be\_immutable

class DetailPeminjaman extends StatefulWidget {

Map peminjaman;

DetailPeminjaman(this.peminjaman);

@override

State < DetailPeminjaman > createState() => \_DetailPeminjamanState();

}

class \_DetailPeminjamanState extends State < DetailPeminjaman > {

@override

Widget build(BuildContext context) {

return Scaffold(appBar: AppBar(title: Text('Detail Transaksi'), ), body: SafeArea(child: SingleChildScrollView(child: Column(

crossAxisAlignment: CrossAxisAlignment.center,

children: [

SizedBox(width: 300, height: 300, child: Image.network(widget.peminjaman['img']),),

SizedBox(height: 10,),

Text('Detail Buku', style: TextStyle(fontWeight: FontWeight.bold, fontSize: 18),),

SizedBox(height: 30,),

Text('Kode Buku: ${widget.peminjaman['kdBuku']}'),

SizedBox(height: 10,),

Text('Nama Buku: ${widget.peminjaman['nmBuku']}'),

SizedBox(height: 10,),

Text('Tanggal Pinjam: ${widget.peminjaman['tgl']}'),

SizedBox(height: 10,),

Text('Catatan: ${widget.peminjaman['catatan']}'),

SizedBox(height: 10,),

Container(height: 100, child: FutureBuilder(future: Api().cekPengembalian(int.parse(widget.peminjaman['id'])), builder: (BuildContext context, AsyncSnapshot det) {

if (det.hasData) {

return det.data != null?

// ListView.builder(itemCount: det.data.length, itemBuilder: (context, index2) {

// return ListTile(leading: Text((1+index2).toString()), trailing: Icon(Icons.delete),

// title: Text(det.data != null?det.data[index2]['tgl'].toString():''),

// // subtitle: Text('Rp '+det.data[index2]['hrgBrg'] + ' x ' + det.data[index2]['qty'] )

// );

// })

Text('Tanggal Kembali: ${det.data['tgl'].toString()}')

: Center(child: Column(children: [

Text('Belum ada pengembalian'),

TextButton(onPressed: ()async{

TextEditingController catatan = TextEditingController();

showDialog(context: context, builder: (context) {

return AlertDialog(title: Text('Pinjam Buku'), content: Container(

height: 300,

padding: EdgeInsets.symmetric(horizontal: 10, vertical: 10),

child: SingleChildScrollView(child: Center(child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Container(width: 100, height: 100, child: Image.network(widget.peminjaman['img'], fit: BoxFit.fill, ), ),

SizedBox(height: 15 ),

Text('${widget.peminjaman['kdBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.bold,fontSize: 18, color: Colors.black54),maxLines: 1, ),

SizedBox(height: 10 ),

Text('${widget.peminjaman['nmBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.normal,fontSize: 16 ,color: Colors.black54), maxLines: 2 ),

SizedBox(height: 10 ),

Text('${widget.peminjaman['status'].toString()}' == '0'?'tersedia':'Dipinjam', style: TextStyle(fontWeight: FontWeight.bold, color: '${widget.peminjaman['status'].toString()}' == '0'?Colors.green:Colors.red),maxLines: 1, ),

SizedBox(height: 10 ),

TextField(controller: catatan, decoration: InputDecoration(labelText: "Catatan", border: OutlineInputBorder(borderRadius: BorderRadius.circular(5.0)))),

SizedBox(height: 20 ),

TextButton(onPressed: () async{

var add = await Api().addPengembalian(int.parse(widget.peminjaman['id'].toString()), DateFormat('yyyy-MM-dd').format(DateTime.now()).toString(), catatan.text);

if(add['status'] == true){

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(add['message'], style: TextStyle(color: Colors.white),), backgroundColor: Colors.green,));

}else{

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(add['message'], style: TextStyle(color: Colors.white),), backgroundColor: Colors.red,));

}

Navigator.pop(context);

setState(() {

});

}, child: Text('Kirim Pengembalian'))

],),),)));

});

// TextEditingController catatan = TextEditingController();

// var data = await Api().addPengembalian(widget.peminjaman['id'], DateFormat('yyyy-MM-dd').format(DateTime.now()).toString(), catatan.text);

}, child: Text('Kembalikan Buku', style: TextStyle(fontWeight: FontWeight.bold, fontSize: 18)))

],),);

} else {

return Center(child: Column(children: [

Text('Belum ada pengembalian'),

TextButton(onPressed: ()async{

TextEditingController catatan = TextEditingController();

showDialog(context: context, builder: (context) {

return AlertDialog(title: Text('Pinjam Buku'), content: Container(

height: 300,

padding: EdgeInsets.symmetric(horizontal: 10, vertical: 10),

child: SingleChildScrollView(child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Container(width: 150, height: 150, child: Image.network(widget.peminjaman['img'], fit: BoxFit.fill, ), ),

SizedBox(height: 15 ),

Text('${widget.peminjaman['kdBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.bold,fontSize: 18, color: Colors.black54),maxLines: 1, ),

SizedBox(height: 10 ),

Text('${widget.peminjaman['nmBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.normal,fontSize: 16 ,color: Colors.black54), maxLines: 2 ),

SizedBox(height: 10 ),

Text('${widget.peminjaman['status'].toString()}' == '0'?'tersedia':'Dipinjam', style: TextStyle(fontWeight: FontWeight.bold, color: '${widget.peminjaman['status'].toString()}' == '0'?Colors.green:Colors.red),maxLines: 1, ),

SizedBox(height: 10 ),

TextField(controller: catatan, decoration: InputDecoration(labelText: "Catatan", border: OutlineInputBorder(borderRadius: BorderRadius.circular(5.0)))),

SizedBox(height: 20 ),

TextButton(onPressed: () async{

var add = await Api().addPengembalian(int.parse(widget.peminjaman['id'].toString()), DateFormat('yyyy-MM-dd').format(DateTime.now()).toString(), catatan.text);

if(add['status'] == true){

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(add['message'], style: TextStyle(color: Colors.white),), backgroundColor: Colors.green,));

}else{

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(add['message'], style: TextStyle(color: Colors.white),), backgroundColor: Colors.red,));

}

Navigator.pop(context);

setState(() {

});

}, child: Text('Kirim Pengembalian'))

],),)));

});

// TextEditingController catatan = TextEditingController();

// var data = await Api().addPengembalian(widget.peminjaman['id'], DateFormat('yyyy-MM-dd').format(DateTime.now()).toString(), catatan.text);

}, child: Text('Kembalikan Buku', style: TextStyle(fontWeight: FontWeight.bold, fontSize: 18)))

],),);

}

}))

],), ), ), );

}

}

home.dart

import 'package:adellia\_uas/peminjaman.dart';

import 'package:adellia\_uas/service.dart';

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

import 'package:flutter/widgets.dart';

import 'package:intl/intl.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'login.dart';

// import 'database/db\_helper.dart';

// import 'model/product.dart';

class HomeScreen extends StatefulWidget {

const HomeScreen({

super.key

});

@override

State < HomeScreen > createState() => \_HomeScreenState();

}

void clearPref() async {

SharedPreferences prefs = await SharedPreferences.getInstance();

await prefs.clear();

}

class \_HomeScreenState extends State < HomeScreen > {

Widget gridContent(Map data) {

return Stack(

children: [

Container(

width: 0.5\*MediaQuery.of(context).size.width,

margin: EdgeInsets.all(5),

padding: EdgeInsets.symmetric(horizontal: 8),

decoration: BoxDecoration(border: Border.all(color: Colors.grey),

borderRadius: BorderRadius.circular(30)),

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Container(width: 100, height: 100, child: Image.network(data['img'], fit: BoxFit.fill, ), ),

SizedBox(height: 5 ),

Text('${data['nmBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.normal,fontSize: 12 ,color: Colors.black54), maxLines: 1, overflow: TextOverflow.ellipsis, ),

Text('${data['status'].toString()}' == '0'?'tersedia':'Dipinjam', style: TextStyle(fontWeight: FontWeight.bold, color: '${data['status'].toString()}' == '0'?Colors.green:Colors.red),maxLines: 1, ),

], ),

),

Container(

width: 0.5\*MediaQuery.of(context).size.width,

padding: EdgeInsets.only(left: 120, bottom: 20, top: 20),

child: Column(

mainAxisAlignment: MainAxisAlignment.spaceBetween,

children: [

GestureDetector(

child: CircleAvatar(radius: 12, backgroundColor: Colors.teal, child: Icon(Icons.info, color: Colors.white, size: 15,),),

onTap: (){

showDialog(context: context, builder: (context) {

return AlertDialog(title: Text('Detail Product'), content: Container(

height: 300,

padding: EdgeInsets.symmetric(horizontal: 10, vertical: 10),

child: SingleChildScrollView(child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Container(width: 150, height: 150, child: Image.network(data['img'], fit: BoxFit.fill, ), ),

SizedBox(height: 15 ),

Text('${data['kdBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.bold,fontSize: 18, color: Colors.black54),maxLines: 1, ),

SizedBox(height: 10 ),

Text('${data['nmBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.normal,fontSize: 16 ,color: Colors.black54), maxLines: 2 ),

SizedBox(height: 10 ),

Text('${data['status'].toString()}' == '0'?'tersedia':'Dipinjam', style: TextStyle(fontWeight: FontWeight.bold, color: '${data['status'].toString()}' == '0'?Colors.green:Colors.red),maxLines: 1, ),

],),)));

});

},

),

'${data['status'].toString()}' == '0'?GestureDetector(

child: CircleAvatar(radius: 12, backgroundColor: Colors.purple, child: Icon(Icons.arrow\_drop\_down\_circle, color: Colors.white, size: 15,),),

onTap: (){

TextEditingController catatan = TextEditingController();

showDialog(context: context, builder: (context) {

return AlertDialog(title: Text('Pinjam Buku'), content: Container(

height: 300,

padding: EdgeInsets.symmetric(horizontal: 10, vertical: 10),

child: SingleChildScrollView(child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Container(width: 150, height: 150, child: Image.network(data['img'], fit: BoxFit.fill, ), ),

SizedBox(height: 15 ),

Text('${data['kdBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.bold,fontSize: 18, color: Colors.black54),maxLines: 1, ),

SizedBox(height: 10 ),

Text('${data['nmBuku'].toString()}', style: TextStyle(fontWeight: FontWeight.normal,fontSize: 16 ,color: Colors.black54), maxLines: 2 ),

SizedBox(height: 10 ),

Text('${data['status'].toString()}' == '0'?'tersedia':'Dipinjam', style: TextStyle(fontWeight: FontWeight.bold, color: '${data['status'].toString()}' == '0'?Colors.green:Colors.red),maxLines: 1, ),

SizedBox(height: 10 ),

TextField(controller: catatan, decoration: InputDecoration(labelText: "Catatan", border: OutlineInputBorder(borderRadius: BorderRadius.circular(5.0)))),

SizedBox(height: 20 ),

TextButton(onPressed: () async{

var add = await Api().addPeminjaman(int.parse(dataUser['id'].toString()),int.parse(data['id'].toString()), DateFormat('yyyy-MM-dd').format(DateTime.now()).toString(), catatan.text);

if(add['status'] == true){

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(add['message'], style: TextStyle(color: Colors.white),), backgroundColor: Colors.green,));

}else{

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(add['message'], style: TextStyle(color: Colors.white),), backgroundColor: Colors.red,));

}

Navigator.pop(context);

setState(() {

});

}, child: Text('Kirim Peminjaman'))

],),)));

});

},

):SizedBox()

],),

)

],

);

}

@override

Widget build(BuildContext context) {

return Scaffold(appBar: AppBar(backgroundColor: Colors.purple, automaticallyImplyLeading: false, actions: [

IconButton(onPressed: () {

dataUser.clear();

clearPref();

Navigator.pushAndRemoveUntil(context, MaterialPageRoute(builder: (BuildContext context){ return LoginScreen();}), (route) => false);

}, icon: Icon(Icons.login\_outlined, color: Colors.white, )),

], title: Text('Daftar Buku', style: TextStyle(fontWeight: FontWeight.bold, color: Colors.white, fontSize: 18), ), ), body: SafeArea(child: Container(height: 0.95 \* MediaQuery.of(context).size.height, child: SingleChildScrollView(child: Column(children: [

Container(height: 650, child: FutureBuilder(future: Api().getBuku(), builder: (BuildContext context, AsyncSnapshot snap) {

if (snap.hasData) {

return GridView(gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(crossAxisCount: 2, crossAxisSpacing: 20), children: List.generate(snap.data.length,

(index) => gridContent(snap.data[index])));

} else {

return SizedBox();

}

}), ),

SizedBox(height: 5 ),

Padding(padding: EdgeInsets.only(left: 5), child: Row(mainAxisAlignment: MainAxisAlignment.spaceEvenly, children: [

IconButton(onPressed: () {

Navigator.push(context, MaterialPageRoute(builder: (context) {

return PeminjamanScreen();

}));

}, icon: Icon(Icons.book, size: 35, color: Colors.purple[900], )),

IconButton(onPressed: () {

setState(() {

});

}, icon: Icon(Icons.sync, size: 35, color: Colors.green[900], )),

], ), ),

], ), ), )));

}

}

login.dart

import 'package:adellia\_uas/register.dart';

import 'package:adellia\_uas/service.dart';

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'home.dart';

Map dataUser = Map();

class LoginScreen extends StatelessWidget {

LoginScreen({ super.key});

TextEditingController emailCont = TextEditingController();

TextEditingController passCont = TextEditingController();

void setSharedPreferences(String email, String mobile, String id) async {

SharedPreferences prefs = await SharedPreferences.getInstance();

await prefs.setString("email", email);

await prefs.setString("mobile",mobile);

await prefs.setString("id", id);

}

void getSharedPreferences() async {

SharedPreferences prefs = await SharedPreferences.getInstance();

dataUser = {

"email": prefs.getString("email"),

"mobile": prefs.getString("mobile"),

"id": prefs.getString("id"),

};

}

@override

Widget build(BuildContext context) {

return Scaffold(body: SafeArea(child: Center(child: SingleChildScrollView(child: Column(crossAxisAlignment: CrossAxisAlignment.center, children: [

SizedBox(height: 30, ),

Container(width: 150, height: 150, child: Image.network('https://pengadaan.penerbitdeepublish.com/wp-content/uploads/2022/09/Logo-Perpus-Merdeka.png'), ),

Text('Halaman Login', style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold, color: Colors.black54), ),

Padding(padding: EdgeInsets.symmetric(horizontal: 30), child: Column(crossAxisAlignment: CrossAxisAlignment.center, children: [

SizedBox(height: 30, ),

TextField(decoration: InputDecoration(label: Text('Email', style: TextStyle(fontSize: 14), )), controller: emailCont, ),

SizedBox(height: 10, ),

TextField(obscureText: true, decoration: InputDecoration(label: Text('Password', style: TextStyle(fontSize: 14), )), controller: passCont, ),

], ), ),

SizedBox(height: 80, ),

ElevatedButton(style: ButtonStyle(backgroundColor: MaterialStatePropertyAll(Colors.purple)), onPressed: () async {

var data = await Api().login(emailCont.text, passCont.text);

dataUser = data;

if (data['status'] == true) {

setSharedPreferences(data['data']['email'].toString(), data['data']['mobile'].toString(), data['data']['id'].toString());

getSharedPreferences();

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(data['message'], style: TextStyle(color: Colors.white), ), backgroundColor: Colors.green, ));

Navigator.pushReplacement(context, MaterialPageRoute(builder: (context) {

return HomeScreen();

}));

} else {

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(data['message'], style: TextStyle(color: Colors.white), ), backgroundColor: Colors.red, ));

}

}, child: Text('Login', style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold, color: Colors.white), )),

SizedBox(height: 30, ),

TextButton(onPressed: () {

Navigator.push(context, MaterialPageRoute(builder: (context) {

return RegisterScreen();

}));

}, child: Text('Register ?'))

], )), )));

}

}

main.dart

import 'package:adellia\_uas/login.dart';

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

SharedPreferences ? sp;

void main() async {

WidgetsFlutterBinding.ensureInitialized();

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({

super.key

});

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

return MaterialApp(title: 'Adellia UAS', home: LoginScreen(), );

}

}

peminjaman.dart

import 'package:adellia\_uas/detail\_peminjaman.dart';

import 'package:adellia\_uas/login.dart';

import 'package:adellia\_uas/service.dart';

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

class PeminjamanScreen extends StatefulWidget {

const PeminjamanScreen({

super.key

});

@override

State < PeminjamanScreen > createState() => \_PeminjamanScreenState();

}

class \_PeminjamanScreenState extends State < PeminjamanScreen > {

@override

Widget build(BuildContext context) {

return Scaffold(appBar: AppBar(title: Text('Daftar Peminjaman'), ), body: SafeArea(child: SingleChildScrollView(child: Container(height: 700, child: FutureBuilder(future: Api().listPeminjaman(int.parse(dataUser['id'].toString())), builder: (BuildContext context, AsyncSnapshot snap) {

if (snap.hasData) {

return ListView.builder(itemCount: snap.data.length, itemBuilder: (context, index) {

return ListTile(leading: SizedBox(width: 80,height: 80,child: Image.network(snap.data[index]['img'].toString()),), title: Text(snap.data[index]['nmBuku'].toString(), style: TextStyle(fontSize: 12),maxLines: 2,),

trailing: SizedBox(width: 100, child: Row(

mainAxisAlignment: MainAxisAlignment.spaceAround,

children: [

GestureDetector(

onTap: ()async{

var data = await Api().delPeminjaman(int.parse(snap.data[index]['id'].toString()), int.parse(snap.data[index]['id\_buku'].toString()));

if(data['status'] == true){

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text('Transaksi Berhasil Dihapus')));

setState(() {});

}else{

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text('Transaksi Gagal Dihapus')));

}

},

child: Icon(Icons.delete, color: Colors.red,),

),

GestureDetector(

onTap: ()async{

Navigator.push(context, MaterialPageRoute(builder: (context) {

return DetailPeminjaman(snap.data[index]);

}));

},

child: Icon(Icons.arrow\_forward\_ios\_rounded, color: Colors.green,),

)

],),),

subtitle: Text(snap.data[index]['kdBuku'].toString()),

);

});

} else {

return SizedBox();

}

}), ), )), );

}

}

register.dart

import 'package:adellia\_uas/service.dart';

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'login.dart';

class RegisterScreen extends StatefulWidget {

const RegisterScreen({

Key ? key

}): super(key: key);

@override

State < RegisterScreen > createState() => \_RegisterScreenState();

}

class \_RegisterScreenState extends State < RegisterScreen > {

TextEditingController emailController = TextEditingController();

TextEditingController mobileController = TextEditingController();

TextEditingController passwordController = TextEditingController();

@override

Widget build(BuildContext context) {

return Scaffold(appBar: AppBar(title: const Text('Register')), body: ListView(padding: const EdgeInsets.all(8.0),

children: [

SizedBox(height: 30, ),

Container(width: 150, height: 150, child: Image.network('https://pengadaan.penerbitdeepublish.com/wp-content/uploads/2022/09/Logo-Perpus-Merdeka.png'), ),

TextField(controller: emailController, decoration: InputDecoration(labelText: "Email")),

const Padding(padding: EdgeInsets.only(top: 4)),

TextField(controller: mobileController, decoration: InputDecoration(labelText: "Mobile")),

const Padding(padding: EdgeInsets.only(top: 4)),

TextField(controller: passwordController, decoration: InputDecoration(labelText: "Password")),

// button untuk menyimpan email dan password ke dalam Shared Preferences melalui method

// setIntoSharedPreferences()

const Padding(padding: EdgeInsets.only(top: 10)),

ElevatedButton(child: const Text("Daftar"),

onPressed: () async {

var data = await Api().register(emailController.text, mobileController.text, passwordController.text);

if (data['status'] == true) {

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(data['message'], style: TextStyle(color: Colors.white), ), backgroundColor: Colors.green, ));

Navigator.pushReplacement(context, MaterialPageRoute(builder: (context) {

return LoginScreen();

}));

} else {

ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text(data['message'], style: TextStyle(color: Colors.white), ), backgroundColor: Colors.red, ));

}

// setIntoSharedPreferences();

}, ),

TextButton(child: const Text("Login Sekarang"),

onPressed: () {

Navigator.push(context, MaterialPageRoute(builder: (context) {

return LoginScreen();

}));

}, ),

], ), );

}

}

service.dart

import 'dart:convert';

import 'package:http/http.dart'

as http;

class Api {

// final url = 'http://192.168.1.38/adelia\_web\_uas';

// final url = 'http://10.0.2.2:80/adelia\_web\_uas';

final url = 'https://adelia.kuliketikteknik.com';

Future getBuku() async {

var url = Uri.parse('${this.url}/buku/all.php');

var response = await http.get(url, );

var data = json.decode(response.body);

print(data);

return data;

}

Future changeStatusBuku(int bukuId, int status) async {

var url = Uri.parse('${this.url}/buku/changeStatus.php');

var response = await http.post(url, body: {

'id\_buku': bukuId.toString(),

'status': status.toString()

});

var data = json.decode(response.body);

print(data);

return data;

}

Future login(String email, String password) async {

var url = Uri.parse('${this.url}/user/login.php');

var response = await http.post(url, body: {

'email': email,

'password': password

});

var data = json.decode(response.body);

print(data);

return data;

}

Future register(String email, String mobile, String password) async {

var url = Uri.parse('${this.url}/user/register.php');

var response = await http.post(url, body: {

'email': email,

'mobile': mobile,

'password': password

});

var data = json.decode(response.body);

print(data);

return data;

}

Future addPeminjaman(int userId, int bukuId, String date, String catatan) async {

var url = Uri.parse('${this.url}/peminjaman/create.php');

var response = await http.post(url, body: {

'id\_user': userId.toString(),

'id\_buku': bukuId.toString(),

'tgl': date.toString(),

'catatan': catatan.toString()

});

var data = json.decode(response.body);

print(data);

return data;

}

Future delPeminjaman(int id, int bukuId) async {

var url = Uri.parse('${this.url}/peminjaman/delete.php');

var response = await http.post(url, body: {

'id': id.toString(),

'id\_buku': bukuId.toString()

});

var data = json.decode(response.body);

print(data);

return data;

}

Future updatePeminjaman(int userId, int bukuId, String date, String catatan, int id) async {

var url = Uri.parse('${this.url}/peminjaman/create.php');

var response = await http.post(url, body: {

'id\_user': userId.toString(),

'id\_buku': bukuId.toString(),

'tgl': date.toString(),

'catatan': catatan.toString(),

'id':id.toString()

});

var data = json.decode(response.body);

print(data);

return data;

}

Future listPeminjaman(int userId) async {

var url = Uri.parse('${this.url}/peminjaman/all.php');

var response = await http.post(url, body: {

'id\_user': userId.toString(),

});

var data = json.decode(response.body);

print(data);

return data;

}

Future addPengembalian(int peminjamanId, String date, String catatan) async {

var url = Uri.parse('${this.url}/pengembalian/create.php');

var response = await http.post(url, body: {

'id\_peminjaman': peminjamanId.toString(),

'tgl': date.toString(),

'catatan': catatan.toString()

});

var data = json.decode(response.body);

print(data);

return data;

}

Future delPengembalian(int id) async {

var url = Uri.parse('${this.url}/pengembalian/delete.php');

var response = await http.post(url, body: {

'id': id.toString()

});

var data = json.decode(response.body);

print(data);

return data;

}

Future listPengembalian(int userId) async {

var url = Uri.parse('${this.url}/pengembalian/all.php');

var response = await http.post(url, body: {

'id\_user': userId.toString(),

});

var data = json.decode(response.body);

print(data);

return data;

}

Future cekPengembalian(int transId) async {

var url = Uri.parse('${this.url}/peminjaman/cek\_pengembalian.php');

var response = await http.post(url, body: {

'id': transId.toString(),

});

var data = json.decode(response.body);

print(data);

return data;

}

}