INTRODUCTION

1.1PROJECT OVERVIEW

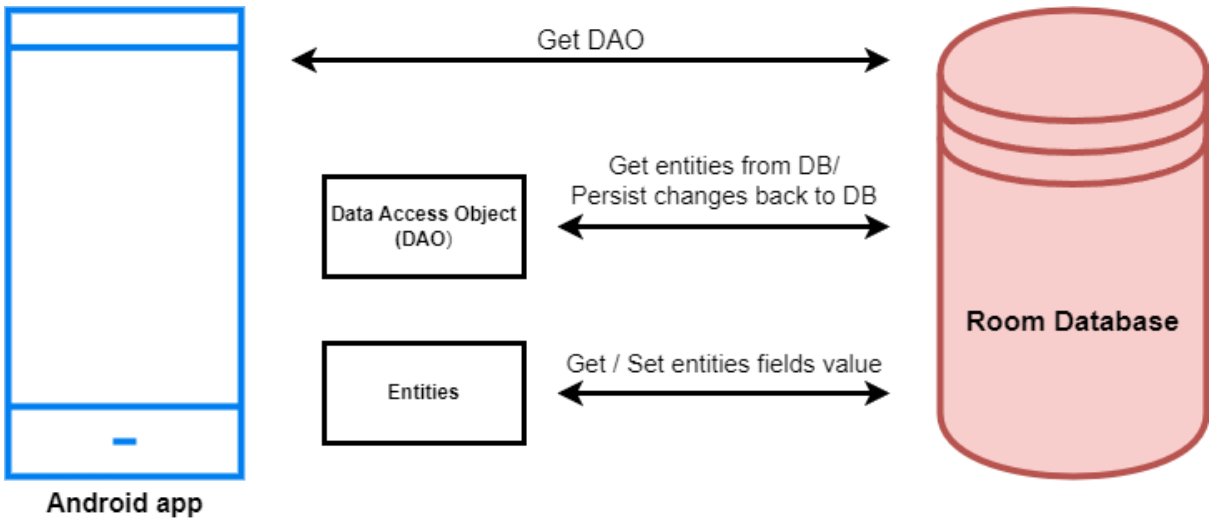
Owl-M: A Material Design Study App

Project Description:

A project that demonstrates the use of Android Jetpack Compose to build a UI for a Owl-M: a material design study app. Owl-M app is a sample project built using the Android Compose UI toolkit. A Compose implementation of the Owl Material

study.

ARCHITECHURE



Learning Outcomes :

By end of this project:

- You'll be able to work on Android studio and build an app.
- You'll be able to integrate the database accordingly.

Project Workflow:

- Users register into the application.
- After registration , user logins into the application.
- User enters into the main page
- User can view the subject themes on selecting theme he can read about it.
- **Note:**

- To complete the project you need to finish up the tasks listed below

Tasks:

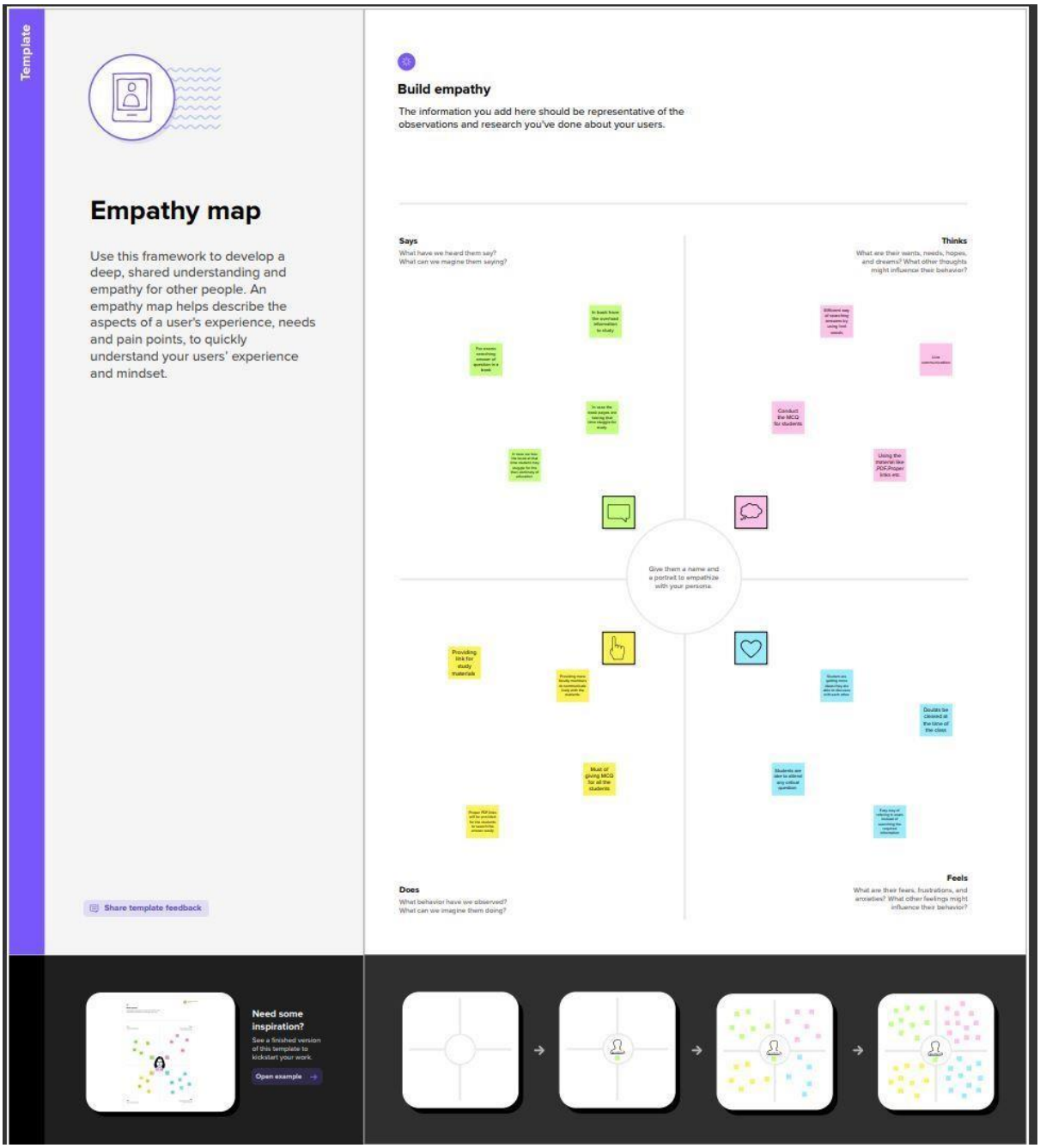
1.Required initial steps
2.Creating a new project.
3.Adding required dependencies.
4.Creating the database classes.
5.Building application UI and connecting to database.
6.Using AndroidManifest.xml
7.Running the application.

1.2 PURPOSE OF PROJECT

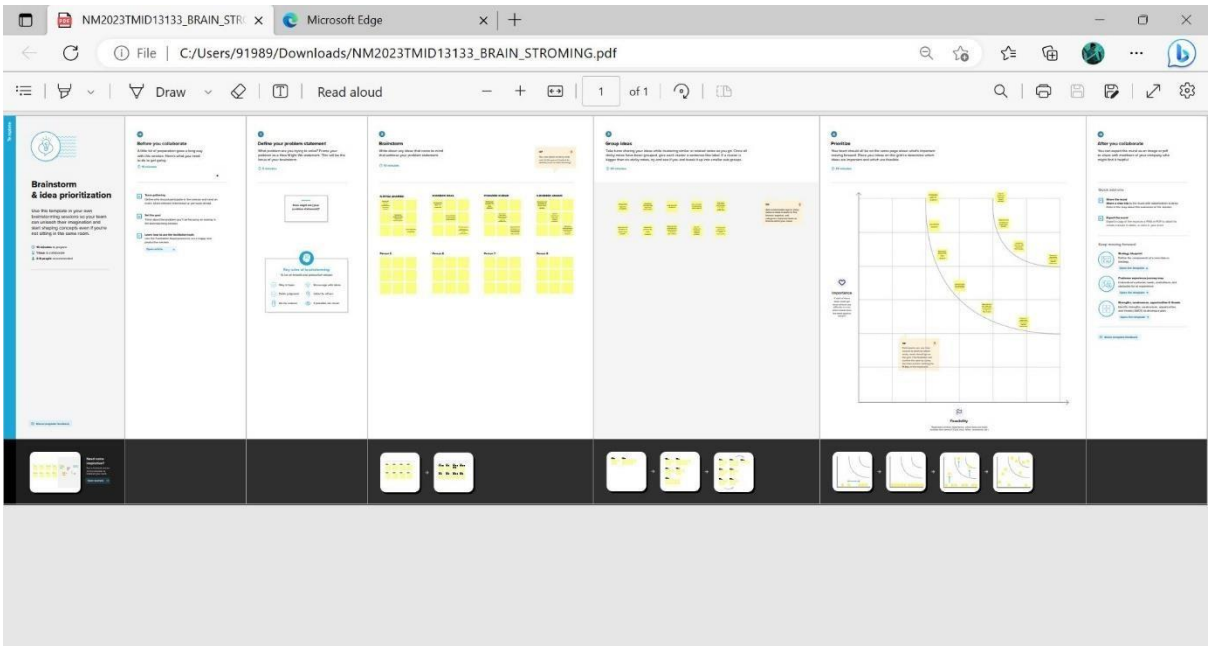
Goal of Material Design was **to enable designers to quickly build apps that were responsive, usable, and scalable**.

2 PROBLEM DEFINITION & DESIGN THINKING

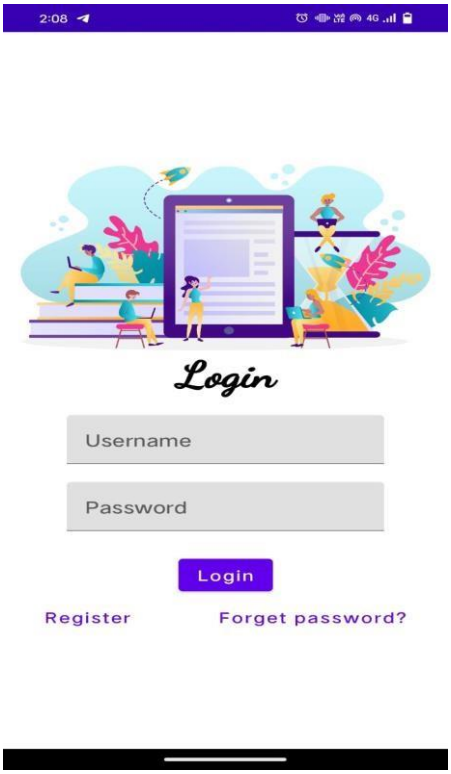
2.1 EMPATHY MAP

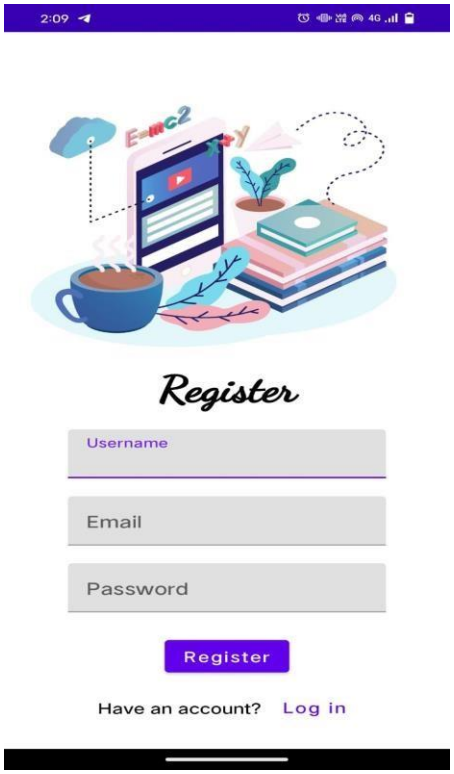


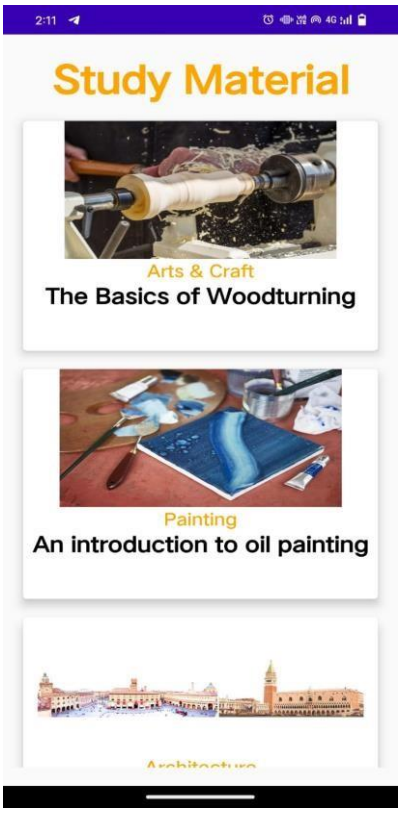
2.2 IDEATION & BRAINSTROMING

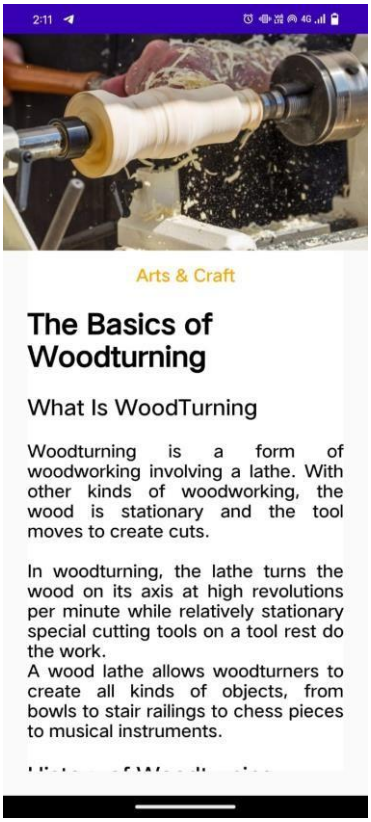


3.RESULT









4.ADVANTAGES

User-friendly interface: Material design provides a clear and intuitive interface that is easy for users to navigate and understand.

Consistency: Material design principles ensure consistency across different devices and platforms, making it easy for users to access the app and use it effectively.

Flexibility: Material design is adaptable to different screen sizes, resolutions, and orientations, ensuring that the app looks great on all devices.

DIS-ADVANTAGES

Technical Expertise: Implementing a Material Design Study App requires technical expertise and experience in using Material Design tools, which may not be readily available to all developers.

Limited Customization: Material Design principles prioritize consistency, which may limit the ability to customize the app's appearance or add unique features.

Compatibility: Material Design may not be compatible with older devices or operating systems, which can limit the audience who can use the app.

5.APPLICATION

A Material Design Study App can be used in various ways to enhance learning and education. Here are a few possible applications:

Educational Institutions: Educational institutions such as schools, colleges, and universities can use the app to create interactive study material for their students. The app can include various features such as interactive quizzes, video tutorials, and study guides.

Corporate Training: Companies can use the app to train their employees on various topics such as sales, marketing, and

customer service. The app can include interactive modules and simulations to make learning more engaging and effective.

6.CONCLUSION

Material Design Study App is a powerful tool that can be used to enhance learning and education in a variety of contexts. By incorporating the principles of material design, the app can create a visually appealing and user-friendly interface that can help learners engage with the material in a more effective and efficient way.

The app can be used in educational institutions, corporate training, online learning, and self-study contexts. It can include features such as interactive quizzes, video tutorials, simulations, progress tracking, personalized learning, notetaking, and bookmarking.

By using a Material Design Study App, learners can benefit from a more engaging and interactive learning experience, which can help them better understand and retain the material. This can lead to improved academic and professional performance and can help learners achieve their goals.

Overall, a Material Design Study App is an excellent tool for anyone looking to enhance their learning and education.

7.SCOPE AND FUTURE:

SCOPE AND FUTURE OF A Material Design Study App

The scope of a Material Design Study App is vast and can potentially revolutionize the way we learn and educate ourselves. As technology continues to advance, the app can be

further developed and enhanced to provide an even more immersive and personalized learning experience.

In the future, the app can potentially incorporate advanced technologies such as artificial intelligence and virtual reality to create a more engaging and interactive learning environment. For example, AI can be used to personalize the learning experience by adapting to the learner's individual needs and preferences, while virtual reality can be used to create realistic simulations and scenarios for learners to practice and apply their knowledge.

Furthermore, the app can be expanded to cover a broader range of topics and subjects, including emerging fields such as blockchain, artificial intelligence, and cybersecurity. It can also potentially be used to provide education and training to people in developing countries, where access to quality education is limited.

Overall, the future of a Material Design Study App is bright and full of potential. By continuing to innovate and develop the app, we can potentially create a more accessible and effective education system that benefits everyone.

8.APPENDIX

A. SOURCE CODE

MainActivity.kt

```
package com.example.owlapplication

import android.content.Context import
android.content.Intent import
```

android.os.Bundle import
androidx.activity.ComponentActivity import
androidx.activity.compose.setContent import
androidx.compose.foundation.Image import
androidx.compose.foundation.clickable
import
androidx.compose.foundation.layout.*
import
androidx.compose.foundation.rememberScr
ollState import
androidx.compose.foundation.verticalScroll
import androidx.compose.material.Card
import androidx.compose.material.Text
import
androidx.compose.runtime.Composable
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier import
androidx.compose.ui.draw.scale import
androidx.compose.ui.graphics.Color import
androidx.compose.ui.res.painterResource
import
androidx.compose.ui.res.stringResource
import
androidx.compose.ui.text.font.FontWeight
import
androidx.compose.ui.text.style.TextAlign

```
import androidx.compose.ui.unit.dp import
androidx.compose.ui.unit.sp

class MainActivity : ComponentActivity() {     override
fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)        setContent {
            StudyApp(this)
        }
    }
}

@Composable
fun StudyApp(context: Context) {

    Column(
modifier = Modifier
            .padding(20.dp)
            .verticalScroll(rememberScrollState())

    ) {

        Text(text = "Study Material",
fontSize = 36.sp,            fontWeight =
FontWeight.Bold,             color =
Color(0xFFFFA500),
            modifier =
```

```
Modifier.align(Alignment.CenterHorizontally))

        Spacer(modifier = Modifier.height(20.dp))



//        01        Card(
modifier = Modifier
.fillMaxWidth()
                .height(250.dp)
                .clickable {
                    context.startActivity(
                        Intent(context, MainActivity2::class.java)


                    )
                },
            elevation = 8.dp
        )
        {
            Column(
                horizontalAlignment =
Alignment.CenterHorizontally
            ) {
                Image(                painterResource(id =
R.drawable.img_1), contentDescription = "",
                    modifier = Modifier
                        .height(150.dp)
                        .scale(scaleX = 1.2F, scaleY = 1F)
```

```
            )
            Text(text = stringResource(id =
R.string.course1),color = Color(0xFFFFA500),
                fontSize = 16.sp)

            Text(                text = stringResource(id =
R.string.topic1),                fontWeight =
FontWeight.Bold,
                fontSize = 20.sp,
textAlign = TextAlign.Center,
            )
        }
    }

    Spacer(modifier = Modifier.height(20.dp))

//      02      Card(
modifier = Modifier
.fillMaxWidth()
            .height(250.dp)
.clickable {
                context.startActivity(
                    Intent(context, MainActivity3::class.java)

                )
            },
```

```
            elevation = 8.dp
        )
        {
            Column(            horizontalAlignment =
Alignment.CenterHorizontally
        ) {
            Image(
                painterResource(id = R.drawable.img_2),
contentDescription = "",
                modifier = Modifier
                    .height(150.dp)
                    .scale(scaleX = 1.4F, scaleY = 1F)
            )
            Text(text = stringResource(id =
R.string.course2),color = Color(0xFFFFA500),
                fontSize = 16.sp)


            Text(                text = stringResource(id =
R.string.topic2),                fontWeight =
FontWeight.Bold,                fontSize = 20.sp,
textAlign = TextAlign.Center,
            )
        }
        }

        Spacer(modifier = Modifier.height(20.dp))
```

```
//      03      Card(
modifier = Modifier
.fillMaxWidth()
            .height(250.dp)
            .clickable {
                context.startActivity(
                    Intent(context, MainActivity4::class.java)


                )
            },
        elevation = 8.dp
    )
    {
        Column(
            horizontalAlignment =
Alignment.CenterHorizontally
        ) {
            Image(                  painterResource(id =
R.drawable.img_3), contentDescription = "",
                modifier = Modifier
                    .height(150.dp)
                    .scale(scaleX = 1.2F, scaleY = 1F)
            )
            Text(text = stringResource(id =
R.string.course3),color = Color(0xFFFFA500),
                fontSize = 16.sp)
```

```
            Text(                   text = stringResource(id =
R.string.topic3),                   fontWeight =
FontWeight.Bold,                    fontSize = 20.sp,
                textAlign = TextAlign.Center,
            )
        }
    }



    Spacer(modifier = Modifier.height(20.dp))

//      04      Card(
modifier = Modifier
.fillMaxWidth()
            .height(250.dp)
.clickable {
                context.startActivity(
                   Intent(context, MainActivity5::class.java)

                )
LoginActivity.kt package
com.example.owlapplication

import android.content.Context import
android.content.Intent import
```

android.os.Bundle import
androidx.activity.ComponentActivity import
androidx.activity.compose.setContent
import
androidx.compose.foundation.Image
import
androidx.compose.foundation.background
import
androidx.compose.foundation.layout.*
import androidx.compose.material.* import
androidx.compose.runtime.* import
androidx.compose.ui.Alignment import
androidx.compose.ui.Modifier import
androidx.compose.ui.graphics.Color import
androidx.compose.ui.layout.ContentScale
import
androidx.compose.ui.res.painterResource
import
androidx.compose.ui.text.font.FontFamily
import
androidx.compose.ui.text.font.FontWeight
import
androidx.compose.ui.text.input.PasswordVisualTransform
ation import
androidx.compose.ui.tooling.preview.Preview import
androidx.compose.ui.unit.dp import

```
androidx.compose.ui.unit.sp import
androidx.core.content.ContextCompat import
com.example.owlapplication.ui.theme.OwlApplicationThem
e

class LoginActivity : ComponentActivity() {     private
lateinit var databaseHelper: UserDatabaseHelper
override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)        databaseHelper
= UserDatabaseHelper(this)        setContent {
          LoginScreen(this, databaseHelper)
      }
   }
}
@Composable
fun LoginScreen(context: Context, databaseHelper:
UserDatabaseHelper) {

   var username by remember { mutableStateOf("") }
var password by remember { mutableStateOf("") }     var
error by remember { mutableStateOf("") }

   Column(
      modifier =
Modifier.fillMaxSize().background(Color.White),
horizontalAlignment = Alignment.CenterHorizontally,
verticalArrangement = Arrangement.Center
```

```
    ) {

        Image(painterResource(id = R.drawable.study_login),
contentDescription = "")

        Text(
fontSize = 36.sp,
fontWeight =
FontWeight.ExtraBold,
fontFamily =
FontFamily.Cursive,
text = "Login"
        )
        Spacer(modifier = Modifier.height(10.dp))

        TextField(            value = username,
onValueChange = { username = it },
label = { Text("Username") },
modifier = Modifier.padding(10.dp)
                .width(280.dp)
        )

        TextField(            value = password,
onValueChange = { password = it },
label = { Text("Password") },
```

```
        visualTransformation =
PasswordVisualTransformation(),
modifier = Modifier.padding(10.dp)
            .width(280.dp)
    )

    if (error.isNotEmpty()) {
Text(               text = error,
            color = MaterialTheme.colors.error,
modifier = Modifier.padding(vertical = 16.dp)
        )
    }

    Button(          onClick = {
if (username.isNotEmpty() &&
password.isNotEmpty()) {
                val user =
databaseHelper.getUserByUsername(username)
                if (user != null && user.password == password)
{                   error = "Successfully log
in"                     context.startActivity(
Intent(                         context,
                        MainActivity::class.java
                    )
                )
                //onLoginSuccess()
```

```
                }
else {
                    error =  "Invalid username or password"
                }

            } else {
                error = "Please fill all fields"
            }
        },
        modifier = Modifier.padding(top = 16.dp)
    ) {
        Text(text = "Login")
    }
    Row {
        TextButton(onClick = {context.startActivity(
            Intent(
context,
                RegisterActivity::class.java
            )
        )}
        )
        { Text(text = "Register") }
        TextButton(onClick = {
        })

        {
```

```
            Spacer(modifier = Modifier.width(60.dp))
            Text(text = "Forget password?")
        }
    }
  }
}
private fun startMainPage(context: Context) {    val
intent = Intent(context, MainActivity::class.java)
   ContextCompat.startActivity(context, intent, null)
```