Owl-M
A Material Design Study App

Submitted by:

Sam Raja Durai J

Ajay M

Gnanavinoth A

Sabari Ganesh M

Karthikeyan R V

Project Report Description

1. INTRODUCTION

1.1 Description about your project Overview A brief

1.2 Purpose The use of this project. What can be achieved using this.

2. Problem Definition & Design Thinking

2.1 Empathy Map Paste the empathy map screenshot

2.2 Ideation & Brainstorming Map Paste the Ideation & brainstorming map screenshot

3. RESULT

Final findings (Output) of the project along with screenshots.

4. ADVANTAGES & DISADVANTAGES

List of advantages and disadvantages of the proposed solution

5. APPLICATIONS

The areas where this solution can be applied

6. CONCLUSION

Conclusion summarizing the entire work and findings.

7. FUTURE SCOPE

Enhancements that can be made in the future.

INTRODUCTION

1.1 Description:

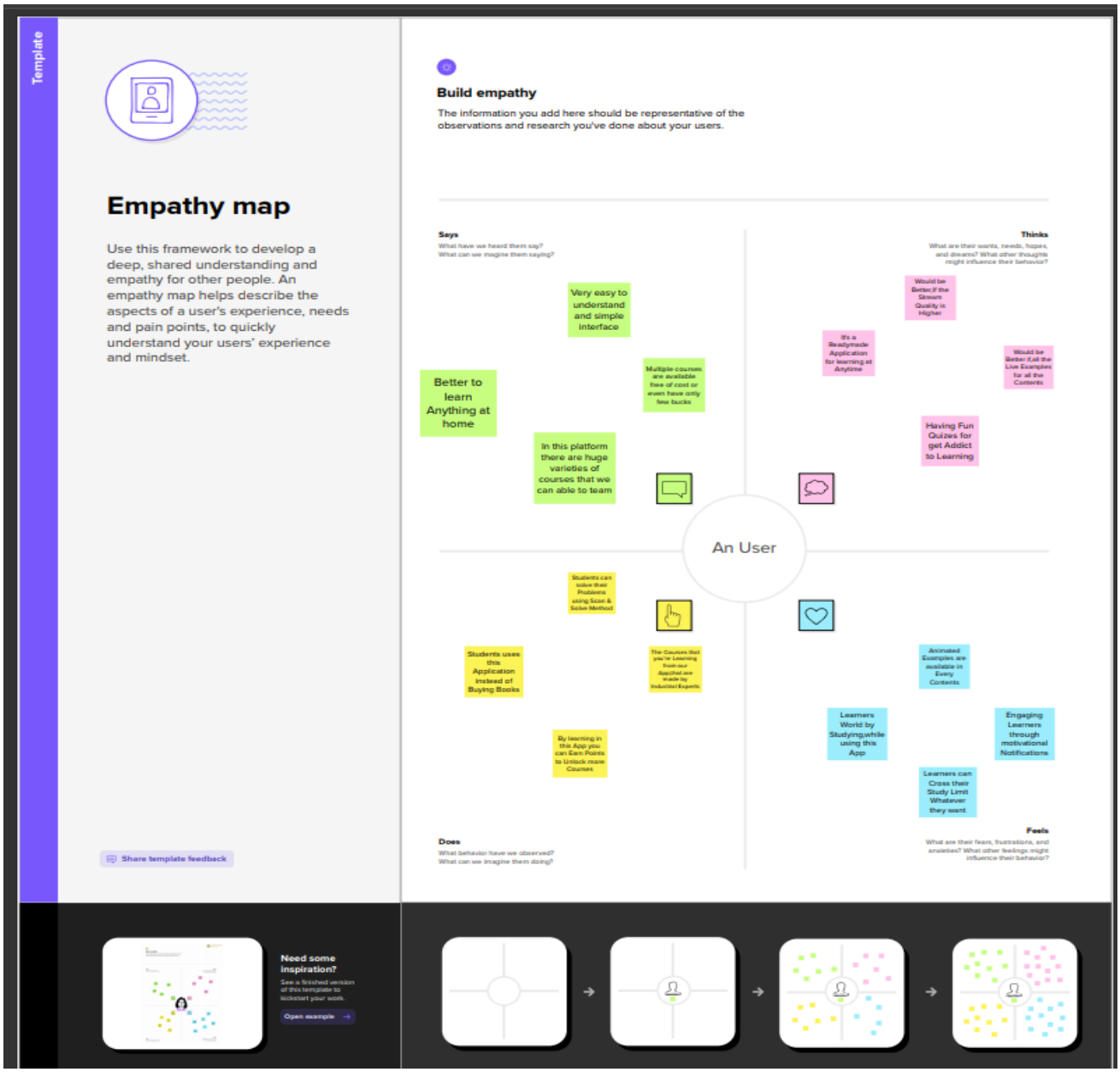
A study material app is a mobile application designed to provide users with access to educational resources and study materials. The app can be used by students of all ages and levels of education, including K-12, undergraduate, and graduate students. Many study material apps also offer features to help users track their progress and stay organized. This may include the ability to set reminders for study sessions, create to-do lists, and set goals for grades or test scores. Overall, a study material app can be a helpful tool for students looking to supplement their learning and improve their academic performance.

1.2 Overview:

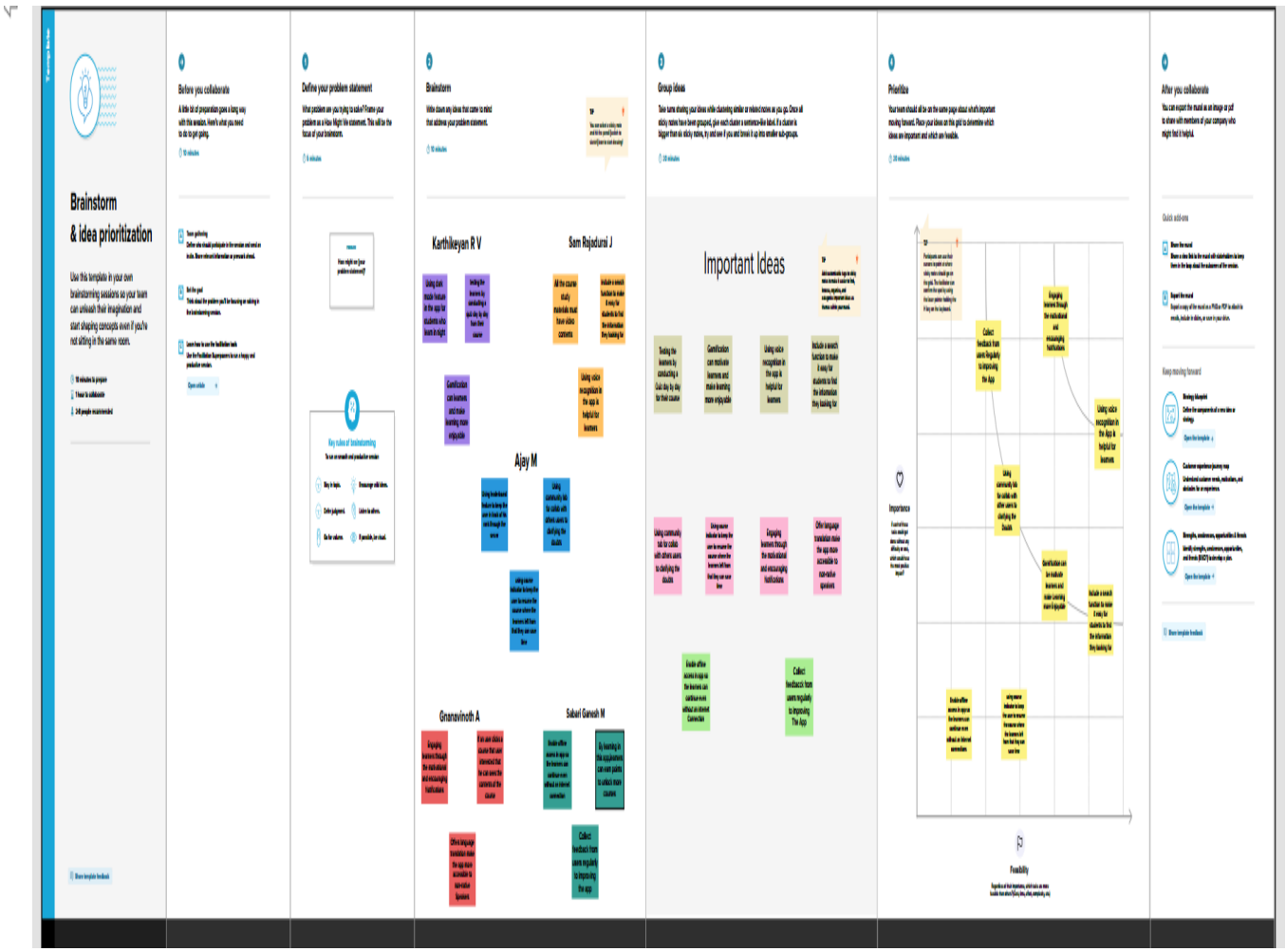
A study material app is a software application designed to provide students with easy access to educational resources and study materials on their mobile devices. Study material apps are available on both iOS and Android devices and are designed to help students of all ages and levels of education. Study material apps also offer features to help students track their progress and stay organized. This may include the ability to set reminders for study sessions, create to-do lists, and set goals for grades or test scores. Some apps may also offer social features, such as the ability to connect with classmates or share study materials with others.

2 Problem Definition & Design Thinking

2.1 Empathy Map:

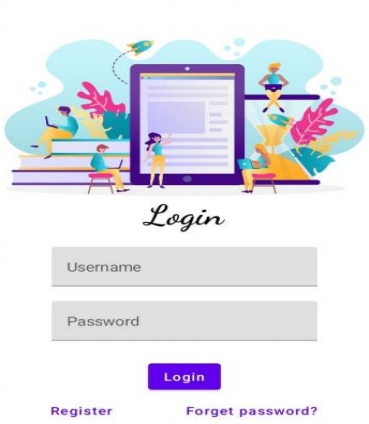


2.2 Brainstorming Map :

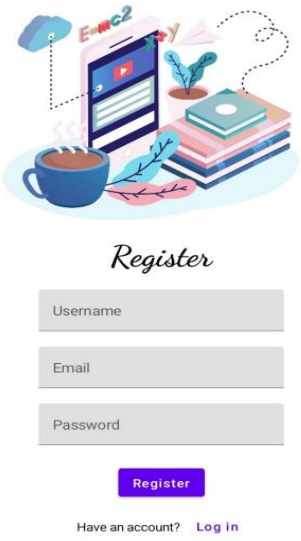


3 Result Final Findings (Output)

Login Page:

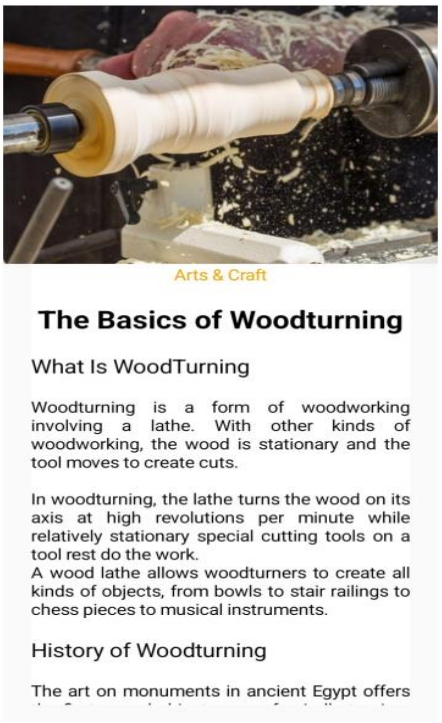


Register Page:



Main Page :

Defining the Topic :



4 Advantages & Disadvantages Of Proposed Solution

Advantages :

Convenience: With a study material app, students can access educational resources and study materials on their mobile devices at any time and from anywhere. This makes it easy for students to study on-the-go or during a busy schedule.

Accessibility: Study material apps can provide students with access to a wide range of educational resources and study materials that may not be available in traditional classrooms or libraries. This can include interactive lesarning tools, videos, and practice quizzes.

Organization: Study material apps often organize educational resources and study materials by subject, course, or topic, making it easy for students to find the information they need. Some apps also offer features to help students stay organized, such as setting reminders for study sessions or creating to-do lists.

Motivation: Study material apps can help motivate students to study by setting goals, tracking progress, and offering rewards for meeting milestones or achieving high scores on practice quizzes.

Collaboration: Some study material apps offer social features that allow students to connect with classmates, share study materials, and collaborate on assignments.

Overall, study material apps can be a helpful tool for students looking to supplement their learning and improve their academic performance. By providing easy access to a wide range of educational resources, study material apps can help students stay organized, motivated, and achieve their academic goals.

Disadvantages :

Distractions: With easy access to social media and other apps on their mobile devices, students may become distracted and spend more time browsing social media instead of studying.

Limited interaction with teachers: Study material apps may not provide the same level of interaction with teachers or professors as traditional classrooms or online courses. This can be especially challenging for students who need additional help or clarification on difficult topics.

Technology issues: Students may encounter technical issues when using study material apps, such as slow internet speeds or compatibility issues with their mobile devices.

Cost: Some study material apps may require students to pay a subscription fee or purchase access to premium content, which can be a financial burden for some students.

Dependence on technology: Students may become overly dependent on technology to access educational resources, which can lead to a lack of critical thinking and problem-solving skills.

Overall, it is important for students to weigh the advantages and disadvantages of study material apps and use them as a supplement to traditional learning methods, rather than a replacement.

5 Applications

K-12 education: Study material apps can provide K-12 students with access to educational resources and study materials, such as textbooks, videos, and interactive learning tools. This can help students stay organized and motivated, and improve their academic performance.

Higher education: Study material apps can be used by college and university students to supplement their learning and improve their grades. These apps can provide access to course materials, practice quizzes, and interactive learning tools that can help students better understand complex topics.

Professional education and training: Study material apps can be used in professional education and training programs to provide learners with access to educational resources and study materials. This can help learners prepare for certification exams or advance their careers.

Self-learning: Study material apps can be used by individuals who are interested in self-learning and personal development. These apps can provide access to a wide range of educational resources and study materials on a variety of topics.

Test preparation: Study material apps can be used to prepare for standardized tests, such as the SAT, ACT, GRE, or GMAT. These apps can provide practice quizzes, flashcards, and other study materials to help students achieve higher scores on these exams.

6 Conclusion

In conclusion, a study material app is a software application designed to provide students with easy access to educational resources and study materials on their mobile devices. These apps offer a wide range of educational resources, including textbooks, notes, videos, quizzes, flashcards, and interactive learning tools. They can be used by learners of all ages and levels of education, including K-12 students, college and university students, professionals, and individuals interested in self-learning. Study material apps offer many advantages, including convenience, organization, accessibility, motivation, and collaboration. However, there are also some potential disadvantages, such as distractions, limited interaction with teachers, technology issues, cost, and dependence on technology. Overall, study material apps can be a helpful tool for learners looking to supplement their learning and improve their academic or professional performance.

7 FUTURE SCOPE:

There are several ways that a study material app can be enhanced to provide even greater value to users. Some potential areas of future scope enhancement for study material apps include:

Personalization: Study material apps can be enhanced to provide personalized learning experiences for each user. This can include features such as personalized study plans, adaptive quizzes, and tailored recommendations based on a user's learning preferences and past performance.

Augmented and Virtual Reality: Study material apps can be enhanced with augmented or virtual reality features to provide immersive learning experiences. This can include features such as interactive simulations, virtual labs, and 3D models that allow users to explore complex concepts in a more engaging and memorable way.

Artificial Intelligence: Study material apps can be enhanced with artificial intelligence features to provide intelligent tutoring and personalized feedback. This can include features such as chatbots, natural language processing, and machine learning algorithms that can help learners better understand difficult concepts and track their progress over time.

Gamification: Study material apps can be enhanced with gamification features to make learning more

fun and engaging. This can include features such as leaderboards, badges, and rewards that motivate learners to study and achieve their goals.

Social Learning: Study material apps can be enhanced with social learning features that allow users to connect with other learners, collaborate on assignments, and share study materials. This can create a more supportive and collaborative learning environment that can enhance learning outcomes.

Overall, there are many potential areas for enhancement and future scope for study material apps that can improve their functionality, engagement, and impact on learning outcomes. As technology continues to evolve, study material apps will likely continue to play an increasingly important role in education and lifelong learning.

8 Appendix

Source code:

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools">

    <application
        android:allowBackup="true"

android:dataExtractionRules="@xml/data_extraction_rules"

android:fullBackupContent="@xml/backup_rules"
        android:icon="@drawable/owl_m"
```

```
        android:label="@string/app_name"
        android:supportsRtl="true"

android:theme="@style/Theme.OwlApplicati
on"
        tools:targetApi="31">
        <activity
            android:name=".RegisterActivity"
            android:exported="false"

android:label="@string/title_activity_register
"

android:theme="@style/Theme.OwlApplicati
on" />
        <activity
            android:name=".MainActivity"
            android:exported="false"
            android:label="MainActivity"
```

```
android:theme="@style/Theme.OwlApplicati
on" />
        <activity
            android:name=".MainActivity5"
            android:exported="false"

android:label="@string/title_activity_main5"

android:theme="@style/Theme.OwlApplicati
on" />
        <activity
            android:name=".MainActivity4"
            android:exported="false"

android:label="@string/title_activity_main4"

android:theme="@style/Theme.OwlApplicati
on" />
```

```
        <activity
            android:name=".MainActivity3"
            android:exported="false"

android:label="@string/title_activity_main3"

android:theme="@style/Theme.OwlApplicati
on" />
        <activity
            android:name=".MainActivity2"
            android:exported="false"

android:label="@string/title_activity_main2"

android:theme="@style/Theme.OwlApplicati
on" />
        <activity
            android:name=".LoginActivity"
            android:exported="true"
```

```
        android:label="@string/app_name"

android:theme="@style/Theme.OwlApplicati
on">
        <intent-filter>
            <action
android:name="android.intent.action.MAIN"
/>

            <category
android:name="android.intent.category.LAUN
CHER" />
        </intent-filter>
      </activity>
   </application>

</manifest>
```

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.rememberScrol
lState
```

```
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)

                    )
                },
            elevation = 8.dp
        )
        {
            Column(
                horizontalAlignment =
Alignment.CenterHorizontally
            ) {
                Image(
```

```
                painterResource(id =
R.drawable.img_4), contentDescription = "",
                modifier = Modifier
                    .height(150.dp)
                    .scale(scaleX = 1.2F, scaleY = 1F)
            )
            Text(text = stringResource(id =
R.string.course4),color = Color(0xFFFFA500),
                fontSize = 16.sp)


            Text(
                text = stringResource(id =
R.string.topic4),
                fontWeight = FontWeight.Bold,
                fontSize = 20.sp,
                textAlign = TextAlign.Center,
            )
        }
```

```
        }
    }
}
```

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.PasswordVisualTransformatio
n
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.OwlApplicationTheme

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)
}
```

RegisterActivity.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.PasswordVisualTransformatio
n
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.OwlApplicationTheme
```

```
class RegisterActivity : ComponentActivity() {
    private lateinit var databaseHelper: UserDatabaseHelper
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = UserDatabaseHelper(this)
        setContent {
            RegistrationScreen(this, databaseHelper)
        }
    }
}

@Composable
fun RegistrationScreen(context: Context, databaseHelper:
UserDatabaseHelper) {

    var username by remember { mutableStateOf("") }
    var password by remember { mutableStateOf("") }
    var email 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_signup),
contentDescription = "")

        Text(
            fontSize = 36.sp,
            fontWeight = FontWeight.ExtraBold,
            fontFamily = FontFamily.Cursive,
            text = "Register"
        )

        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 = email,
        onValueChange = { email = it },
        label = { Text("Email") },
        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() && email.isNotEmpty()) {
                    val user = User(
                        id = null,
                        firstName = username,
                        lastName = null,
                        email = email,
                        password = password
```

```
            )
            databaseHelper.insertUser(user)
            error = "User registered successfully"
            // Start LoginActivity using the current context
            context.startActivity(
                Intent(
                    context,
                    LoginActivity::class.java
                )
            )

        } else {
            error = "Please fill all fields"
        }
    },
    modifier = Modifier.padding(top = 16.dp)
) {
    Text(text = "Register")
}
Spacer(modifier = Modifier.width(10.dp))
Spacer(modifier = Modifier.height(10.dp))
```

```
        Row() {
            Text(
                modifier = Modifier.padding(top = 14.dp), text =
"Have an account?"
            )
            TextButton(onClick = {
                context.startActivity(
                    Intent(
                        context,
                        LoginActivity::class.java
                    )
                )
            })


            {
                Spacer(modifier = Modifier.width(10.dp))
                Text(text = "Log in")
            }
        }
    }
}
private fun startLoginActivity(context: Context) {
```

```
    val intent = Intent(context, LoginActivity::class.java)
    ContextCompat.startActivity(context, intent, null)
}
```