**Project Report**

**Introduction:**

Material design is a comprehensive guide for visual, motion, and interaction design across platforms and devices. To use material design in your Android apps, follow the guidelines defined in the material design specification and use the new components and styles available in the material design support library.

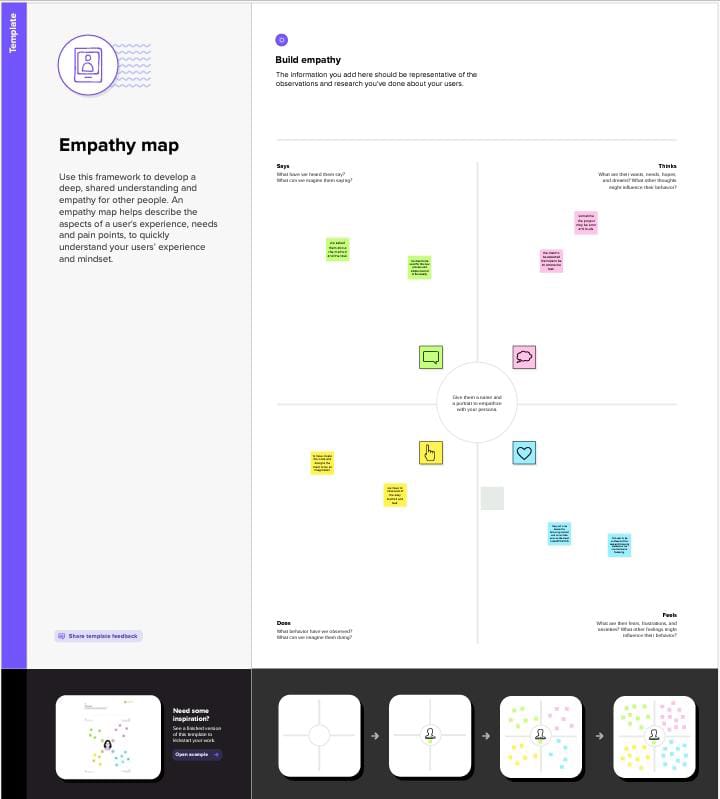
**Purpose:**

The overall goal of Material Design was to enable designers to quickly build apps that were responsive, usable, and scalable.

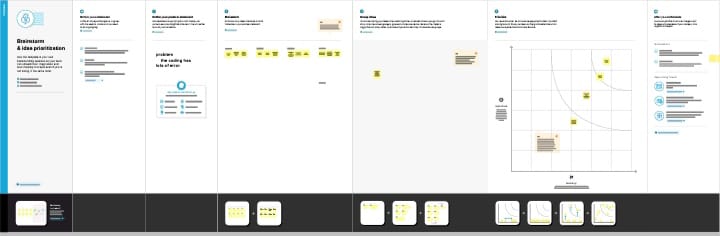
The effective use of design gives customers a reason for buying from you and not from your competitors. It’s a valuable source of differentiation – a well-designed product or service will stand out from the competition. Design also adds value to products and services.

**Problem definition and Design thinking**:

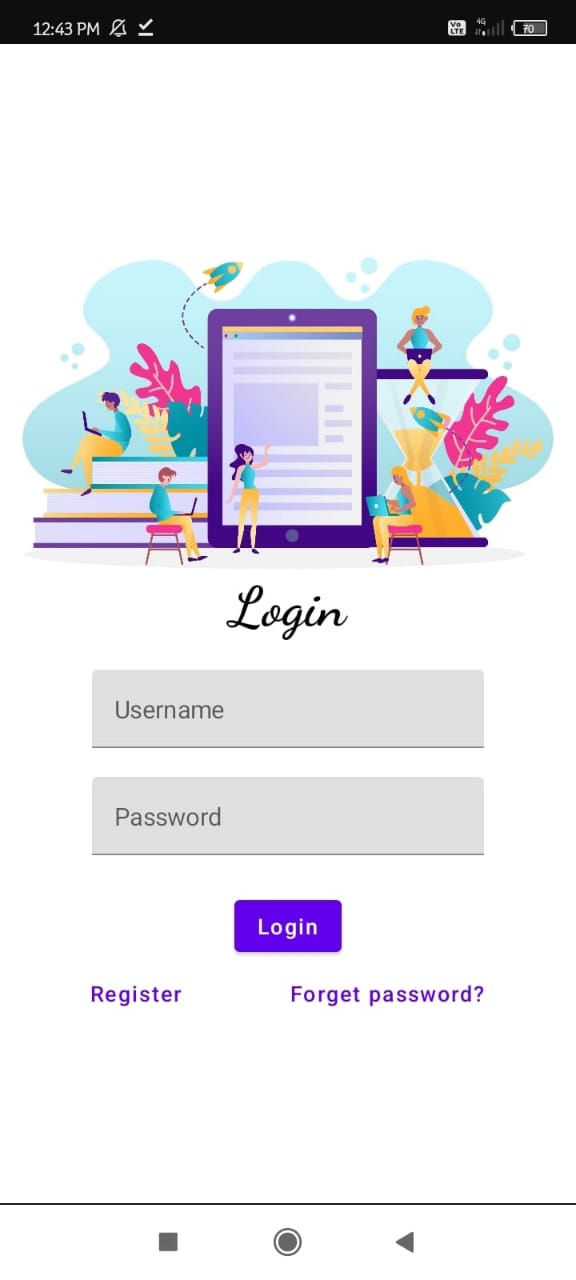
**Empathy Map:**

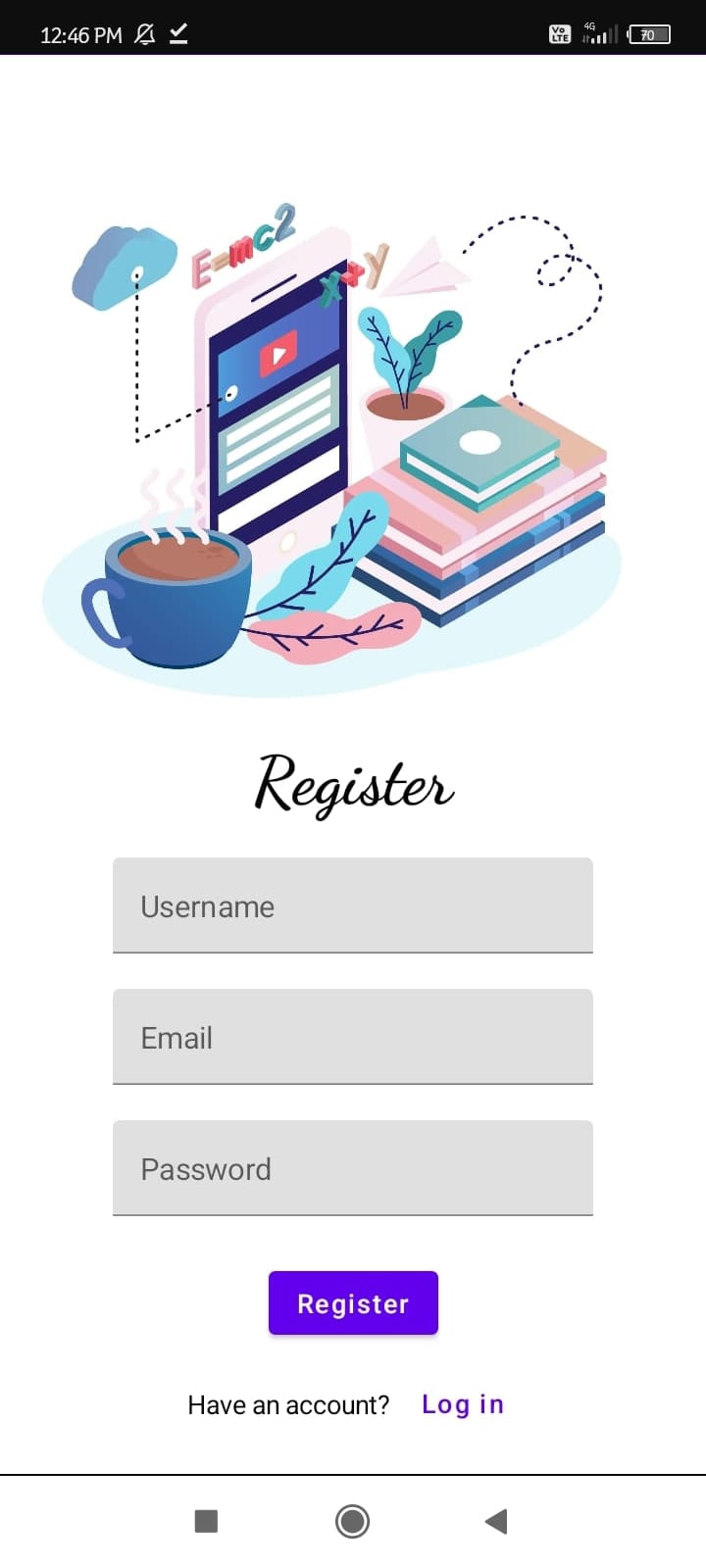
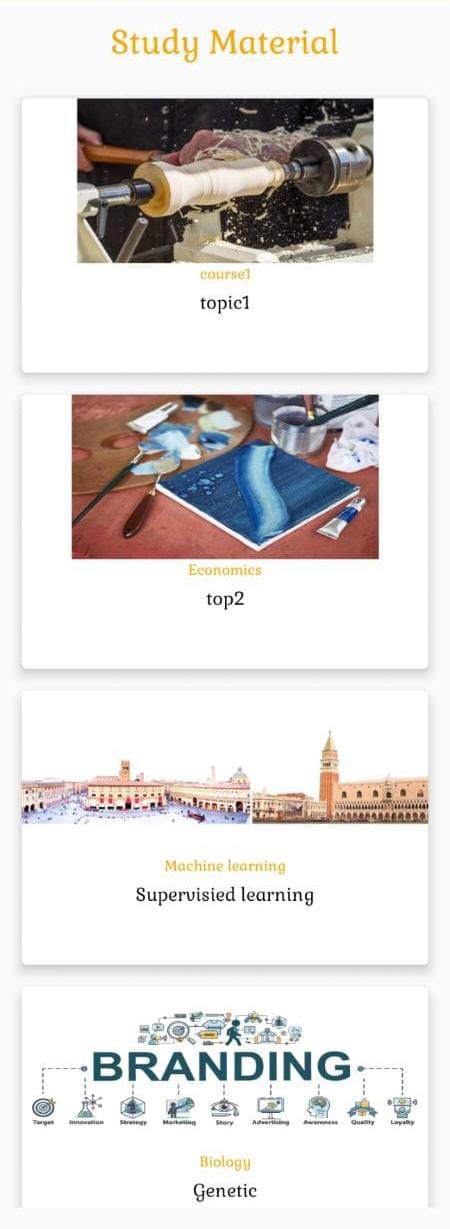
****

**Ideation&Brainstormng map:**

****

**Result:**

****

**  
**

**Advantage:**

**Material design is good in some ways because it allows users to interact with content easily by performing certain actions .**

1. **More intuitive in nature**
2. **Flexibility**
3. **Rapid prototyping.**

**Disadvantage:**

**1.Too many colors. Because the interface consists of many objects and different components, it becomes difficult for users to look at screens that contain too much color; especially if they're older or color blind. ...**

**2.The animations. ...**

**3.Too many distractions. ...**

**4.It has a learning curve.**

**5.Overuse of images and colors can be distracting.**

**Material Design still promotes use of vibrant colors and images in its specification. While this can make an interface lively, this style is prone to overuse and can be very distracting to users trying to get something done**

**Future scope :**

**1.Augmented Reality with Integration**

**2. Personalized Learning Paths**

**3. Collaborative Learning features**

**4.Lots of Animation.**

**5.We will upgrade more materials of several categories.**

**Application:**

**A material design study app can be used for various purposes, including:**

**Learning and Education: A material design study app can provide a platform for students to access educational content such as lessons, quizzes, and interactive tutorials in a visually appealing and user-friendly interface. The app can incorporate material design principles, such as bold colors, meaningful animations, and intuitive navigation, to create an engaging learning experience for users.**

**Test Preparation: A material design study app can also be used for test preparation, such as for standardized exams, where users can access practice tests, review materials, and track their progress. The app can use material design elements to present information in a clear and organized manner, making it easier for users to focus on their studies and track their performance.**

**Language Learning: A material design study app can be designed specifically for language learning, where users can access lessons, vocabulary drills, speaking exercises, and other language-related content. Material design can be used to create a visually appealing and interactive interface that promotes effective learning through engaging visuals, interactive elements, and smooth transitions.**

**Skill Development: A material design study app can also be used for skill development, such as learning coding, graphic design, or other professional skills. The app can provide tutorials, practice exercises, and challenges in a material design interface that is visually appealing and easy to navigate, helping users develop their skills in a user-friendly environment.**

**Personal Development: A material design study app can also be used for personal development purposes, such as self-paced courses, goal-setting, and habit tracking. Material design can be used to create an aesthetically pleasing and motivational interface that encourages users to engage in personal growth activities and track their progress towards their goals.**

**Overall, a material design study app can be applied in various educational and skill-building contexts to provide a visually appealing, user-friendly, and engaging learning experience for users.**

**Conclusion:**

**The future scope of material design is vast and holds immense potential for incorporating advanced technologies,personalized learning approaches to provide a comprehensive and engaging learning experience for users.**

**Appendix:**

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.rememberScrollState  
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,  
 )  
 **}  
 }**

}

**Build Gradle:**

lugins **{** id 'com.android.application'  
 id 'org.jetbrains.kotlin.android'  
**}**android **{** namespace 'com.example.owlapplication'  
 compileSdk 33  
  
 defaultConfig **{** applicationId "com.example.owlapplication"  
 minSdk 24  
 targetSdk 33  
 versionCode 1  
 versionName "1.0"  
  
 testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"  
 vectorDrawables **{** useSupportLibrary true  
 **}  
 }** buildTypes **{** release **{** minifyEnabled false  
 proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'  
 **}  
 }** compileOptions **{** sourceCompatibility JavaVersion.*VERSION\_1\_8* targetCompatibility JavaVersion.*VERSION\_1\_8* **}** kotlinOptions **{** jvmTarget = '1.8'  
 **}** buildFeatures **{** compose true  
 **}** composeOptions **{** kotlinCompilerExtensionVersion '1.2.0'  
 **}** packagingOptions **{** resources **{** excludes += '/META-INF/{AL2.0,LGPL2.1}'  
 **}  
 }  
}**dependencies **{** implementation 'androidx.core:core-ktx:1.7.0'  
 implementation 'androidx.lifecycle:lifecycle-runtime-ktx:2.3.1'  
 implementation 'androidx.activity:activity-compose:1.3.1'  
 implementation "androidx.compose.ui:ui:$compose\_ui\_version"  
 implementation "androidx.compose.ui:ui-tooling-preview:$compose\_ui\_version"  
 implementation 'androidx.compose.material:material:1.2.0'  
 testImplementation 'junit:junit:4.13.2'  
 androidTestImplementation 'androidx.test.ext:junit:1.1.3'  
 androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'  
 androidTestImplementation "androidx.compose.ui:ui-test-junit4:$compose\_ui\_version"  
 debugImplementation "androidx.compose.ui:ui-tooling:$compose\_ui\_version"  
 debugImplementation "androidx.compose.ui:ui-test-manifest:$compose\_ui\_version"  
 // Adding Room dependencies  
  
 implementation 'androidx.room:room-common:2.5.0'  
  
 implementation 'androidx.room:room-ktx:2.5.0'

**Build gradle(Owl application)**

uildscript **{** ext **{** compose\_ui\_version = '1.2.0'  
 **}  
}**// Top-level build file where you can add configuration options common to all sub-projects/modules.  
plugins **{** id 'com.android.application' version '7.4.2' apply false  
 id 'com.android.library' version '7.4.2' apply false  
 id 'org.jetbrains.kotlin.android' version '1.7.0' apply false  
**}**

**Userkt:**

package com.example.owlapplication  
  
import androidx.room.ColumnInfo  
import androidx.room.Entity  
import androidx.room.PrimaryKey  
  
@Entity(tableName = "user\_table")  
data class User(  
 @PrimaryKey(autoGenerate = true) val id: Int?,  
 @ColumnInfo(name = "first\_name") val firstName: String?,  
 @ColumnInfo(name = "last\_name") val lastName: String?,  
 @ColumnInfo(name = "email") val email: String?,  
 @ColumnInfo(name = "password") val password: String?,  
  
 )

**User dao.kt:**

package com.example.owlapplication  
import androidx.room.\*  
  
@Dao  
interface UserDao {  
  
 @Query("SELECT \* FROM user\_table WHERE email = :email")  
 suspend fun getUserByEmail(email: String): User?  
  
 @Insert(onConflict = OnConflictStrategy.REPLACE)  
 suspend fun insertUser(user: User)  
  
 @Update  
 suspend fun updateUser(user: User)  
  
 @Delete  
 suspend fun deleteUser(user: User)  
}

**Userdatabase.kt:**

package com.example.owlapplication  
import android.content.Context  
import androidx.room.Database  
import androidx.room.Room  
import androidx.room.RoomDatabase  
  
@Database(entities = [User::class], version = 1)  
abstract class UserDatabase : RoomDatabase() {  
  
 abstract fun userDao(): UserDao  
  
 companion object {  
  
 @Volatile  
 private var instance: UserDatabase? = null  
  
 fun getDatabase(context: Context): UserDatabase {  
 return instance ?: *synchronized*(this) **{** val newInstance = Room.databaseBuilder(  
 context.*applicationContext*,  
 UserDatabase::class.*java*,  
 "user\_database"  
 ).build()  
 instance = newInstance  
 newInstance  
 **}** }  
 }

}

**Userdatabasehelper:**

package com.example.owlapplication  
import android.annotation.SuppressLint  
import android.content.ContentValues  
import android.content.Context  
import android.database.Cursor  
import android.database.sqlite.SQLiteDatabase  
import android.database.sqlite.SQLiteOpenHelper  
  
class UserDatabaseHelper(context: Context) :  
 SQLiteOpenHelper(context, DATABASE\_NAME, null, DATABASE\_VERSION) {  
  
 companion object {  
 private const val DATABASE\_VERSION = 1  
 private const val DATABASE\_NAME = "UserDatabase.db"  
  
 private const val TABLE\_NAME = "user\_table"  
 private const val COLUMN\_ID = "id"  
 private const val COLUMN\_FIRST\_NAME = "first\_name"  
 private const val COLUMN\_LAST\_NAME = "last\_name"  
 private const val COLUMN\_EMAIL = "email"  
 private const val COLUMN\_PASSWORD = "password"  
 }  
  
 override fun onCreate(db: SQLiteDatabase?) {  
 val createTable = "CREATE TABLE $TABLE\_NAME (" +  
 "$COLUMN\_ID INTEGER PRIMARY KEY AUTOINCREMENT, " +  
 "$COLUMN\_FIRST\_NAME TEXT, " +  
 "$COLUMN\_LAST\_NAME TEXT, " +  
 "$COLUMN\_EMAIL TEXT, " +  
 "$COLUMN\_PASSWORD TEXT" +  
 ")"  
  
 db?.execSQL(createTable)  
 }  
  
 override fun onUpgrade(db: SQLiteDatabase?, oldVersion: Int, newVersion: Int) {  
 db?.execSQL("DROP TABLE IF EXISTS $TABLE\_NAME")  
 onCreate(db)  
 }  
  
 fun insertUser(user: User) {  
 val db = writableDatabase  
 val values = ContentValues()  
 values.put(COLUMN\_FIRST\_NAME, user.firstName)  
 values.put(COLUMN\_LAST\_NAME, user.lastName)  
 values.put(COLUMN\_EMAIL, user.email)  
 values.put(COLUMN\_PASSWORD, user.password)  
 db.insert(TABLE\_NAME, null, values)  
 db.close()  
 }  
  
 @SuppressLint("Range")  
 fun getUserByUsername(username: String): User? {  
 val db = readableDatabase  
 val cursor: Cursor = db.rawQuery("SELECT \* FROM $TABLE\_NAME WHERE $COLUMN\_FIRST\_NAME = ?", arrayOf(username))  
 var user: User? = null  
 if (cursor.moveToFirst()) {  
 user = User(  
 id = cursor.getInt(cursor.getColumnIndex(COLUMN\_ID)),  
 firstName = cursor.getString(cursor.getColumnIndex(COLUMN\_FIRST\_NAME)),  
 lastName = cursor.getString(cursor.getColumnIndex(COLUMN\_LAST\_NAME)),  
 email = cursor.getString(cursor.getColumnIndex(COLUMN\_EMAIL)),  
 password = cursor.getString(cursor.getColumnIndex(COLUMN\_PASSWORD)),  
 )  
 }  
 cursor.close()  
 db.close()  
 return user  
 }  
 @SuppressLint("Range")  
 fun getUserById(id: Int): User? {  
 val db = readableDatabase  
 val cursor: Cursor = db.rawQuery("SELECT \* FROM $TABLE\_NAME WHERE $COLUMN\_ID = ?", arrayOf(id.toString()))  
 var user: User? = null  
 if (cursor.moveToFirst()) {  
 user = User(  
 id = cursor.getInt(cursor.getColumnIndex(COLUMN\_ID)),  
 firstName = cursor.getString(cursor.getColumnIndex(COLUMN\_FIRST\_NAME)),  
 lastName = cursor.getString(cursor.getColumnIndex(COLUMN\_LAST\_NAME)),  
 email = cursor.getString(cursor.getColumnIndex(COLUMN\_EMAIL)),  
 password = cursor.getString(cursor.getColumnIndex(COLUMN\_PASSWORD)),  
 )  
 }  
 cursor.close()  
 db.close()  
 return user  
 }  
  
 @SuppressLint("Range")  
 fun getAllUsers(): List<User> {  
 val users = mutableListOf<User>()  
 val db = readableDatabase  
 val cursor: Cursor = db.rawQuery("SELECT \* FROM $TABLE\_NAME", null)  
 if (cursor.moveToFirst()) {  
 do {  
 val user = User(  
 id = cursor.getInt(cursor.getColumnIndex(COLUMN\_ID)),  
 firstName = cursor.getString(cursor.getColumnIndex(COLUMN\_FIRST\_NAME)),  
 lastName = cursor.getString(cursor.getColumnIndex(COLUMN\_LAST\_NAME)),  
 email = cursor.getString(cursor.getColumnIndex(COLUMN\_EMAIL)),  
 password = cursor.getString(cursor.getColumnIndex(COLUMN\_PASSWORD)),  
 )  
 users.add(user)  
 } while (cursor.moveToNext())  
 }  
 cursor.close()  
 db.close()  
 return users  
 }  
  
}

**Login activity.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.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:**

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.PasswordVisualTransformation  
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)

**Main Activity.kt**

ackage 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.rememberScrollState  
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,  
 )  
 **}  
 }  
  
  
  
 }**

**MainActivity2kt:**

package com.example.owlapplication  
  
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.foundation.rememberScrollState  
import androidx.compose.foundation.verticalScroll  
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  
import com.example.owlapplication.ui.theme.OwlApplicationTheme  
  
class MainActivity2 : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *setContent* **{** *Greeting*()  
 **}** }  
}  
@Composable  
fun Greeting() {  
 *Column*(  
 modifier = Modifier.*padding*(start = 26.*dp*, end = 26.*dp*, bottom = 26.*dp*)  
 .*verticalScroll*(*rememberScrollState*())  
 .*background*(Color.White),  
 verticalArrangement = Arrangement.Top  
 ) **{** *Image*(  
 *painterResource*(id = R.drawable.*img\_1*),  
 contentDescription = "",  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
 .*scale*(scaleX = 1.5F, scaleY = 1.5F)  
 )  
  
 *Spacer*(modifier = Modifier.*height*(60.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*course1*),  
 color = *Color*(0xFFFFA500),  
 fontSize = 16.*sp*,  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
 )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*topic1*),  
 fontWeight = FontWeight.Bold,  
 fontSize = 26.*sp*,  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
  
 )  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
 *Text*(  
 text = *stringResource*(id = R.string.*subheading1\_1*),  
 modifier = Modifier.*align*(Alignment.Start),  
 fontSize = 20.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*text1\_1*),  
 modifier = Modifier.*align*(Alignment.Start),  
 textAlign = TextAlign.Justify,  
 fontSize = 16.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
 *Text*(  
 text = *stringResource*(id = R.string.*subheading1\_2*),  
 modifier = Modifier.*align*(Alignment.Start),  
 fontSize = 20.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*text1\_2*),  
 modifier = Modifier.*align*(Alignment.Start),  
 textAlign = TextAlign.Justify,  
 fontSize = 16.*sp* )  
  
  
  
  
 **}**}

**MainActivity3kt:**

package com.example.owlapplication  
  
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.foundation.rememberScrollState  
import androidx.compose.foundation.verticalScroll  
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 MainActivity3 : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *setContent* **{** *Greeting1*()  
 **}** }  
}  
@Composable  
fun Greeting1() {  
 *Column*(  
 modifier = Modifier.*padding*(start = 26.*dp*, end = 26.*dp*, bottom = 26.*dp*)  
 .*verticalScroll*(*rememberScrollState*())  
 .*background*(Color.White),  
 verticalArrangement = Arrangement.Top  
 ) **{** *Image*(  
 *painterResource*(id = R.drawable.*img\_2*),  
 contentDescription = "",  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
 .*scale*(scaleX = 1.2F, scaleY = 1F)  
 )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*course2*),  
 color = *Color*(0xFFFFA500),  
 fontSize = 16.*sp*,  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
 )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*topic2*),  
 fontWeight = FontWeight.Bold,  
 fontSize = 26.*sp*,  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
  
 )  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
 *Text*(  
 text = *stringResource*(id = R.string.*subheading2\_1*),  
 modifier = Modifier.*align*(Alignment.Start),  
 fontSize = 20.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*text2\_1*),  
 modifier = Modifier.*align*(Alignment.Start),  
 textAlign = TextAlign.Justify,  
 fontSize = 16.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
 *Text*(  
 text = *stringResource*(id = R.string.*subheading2\_2*),  
 modifier = Modifier.*align*(Alignment.Start),  
 fontSize = 20.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*text2\_2*),  
 modifier = Modifier.*align*(Alignment.Start),  
 textAlign = TextAlign.Justify,  
 fontSize = 16.*sp* )  
  
  
  
  
 **}**

**}**

**MainActivity4kt:**

package com.example.owlapplication  
  
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.foundation.rememberScrollState  
import androidx.compose.foundation.verticalScroll  
import androidx.compose.material.MaterialTheme  
import androidx.compose.material.Surface  
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.tooling.preview.Preview  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import com.example.owlapplication.ui.theme.OwlApplicationTheme  
  
class MainActivity4 : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContent **{** Greeting2()  
 **}** }  
}  
@Composable  
fun Greeting2() {  
 Column(  
 modifier = Modifier.padding(start = 26.dp, end = 26.dp, bottom = 26.dp)  
 .verticalScroll(rememberScrollState())  
 .background(Color.White),  
 verticalArrangement = Arrangement.Top  
 ) **{** Image(  
 painterResource(id = R.drawable.img\_3),  
 contentDescription = "",  
 modifier = Modifier.align(Alignment.CenterHorizontally)  
 .scale(scaleX = 1.5F, scaleY = 2F)  
 )  
  
 Spacer(modifier = Modifier.height(60.dp))  
  
 Text(  
 text = stringResource(id = R.string.course3),  
 color = Color(0xFFFFA500),  
 fontSize = 16.sp,  
 modifier = Modifier.align(Alignment.CenterHorizontally)  
 )  
  
 Spacer(modifier = Modifier.height(20.dp))  
  
 Text(  
 text = stringResource(id = R.string.topic3),  
 fontWeight = FontWeight.Bold,  
 fontSize = 26.sp,  
 modifier = Modifier.align(Alignment.CenterHorizontally)  
  
 )  
 Spacer(modifier = Modifier.height(20.dp))  
 Text(  
 text = stringResource(id = R.string.subheading3\_1),  
 modifier = Modifier.align(Alignment.Start),  
 fontSize = 20.sp  
 )  
  
 Spacer(modifier = Modifier.height(20.dp))  
  
 Text(  
 text = stringResource(id = R.string.text3\_1),  
 modifier = Modifier.align(Alignment.Start),  
 textAlign = TextAlign.Justify,  
 fontSize = 16.sp  
 )  
  
 Spacer(modifier = Modifier.height(20.dp))  
 Text(  
 text = stringResource(id = R.string.subheading3\_2),  
 modifier = Modifier.align(Alignment.Start),  
 fontSize = 20.sp  
 )  
  
 Spacer(modifier = Modifier.height(20.dp))  
  
 Text(  
 text = stringResource(id = R.string.text3\_2),  
 modifier = Modifier.align(Alignment.Start),  
 textAlign = TextAlign.Justify,  
 fontSize = 16.sp  
 )  
  
  
  
  
 **}**

**}**

**MainActivity5kt:**

package com.example.owlapplication  
  
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.foundation.rememberScrollState  
import androidx.compose.foundation.verticalScroll  
import androidx.compose.material.MaterialTheme  
import androidx.compose.material.Surface  
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.tooling.preview.Preview  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import com.example.owlapplication.ui.theme.OwlApplicationTheme  
  
class MainActivity5 : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *setContent* **{** *Greeting3*()  
 **}** }  
}  
@Composable  
fun Greeting3() {  
 *Column*(  
 modifier = Modifier.*padding*(start = 26.*dp*, end = 26.*dp*, bottom = 26.*dp*)  
 .*verticalScroll*(*rememberScrollState*())  
 .*background*(Color.White),  
 verticalArrangement = Arrangement.Top  
 ) **{** *Image*(  
 *painterResource*(id = R.drawable.*img\_4*),  
 contentDescription = "",  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
 .*scale*(scaleX = 1.5F, scaleY = 1.5F)  
 )  
  
 *Spacer*(modifier = Modifier.*height*(60.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*course4*),  
 color = *Color*(0xFFFFA500),  
 fontSize = 16.*sp*,  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
 )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*topic4*),  
 fontWeight = FontWeight.Bold,  
 fontSize = 26.*sp*,  
 modifier = Modifier.*align*(Alignment.CenterHorizontally)  
  
 )  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
 *Text*(  
 text = *stringResource*(id = R.string.*subheading4\_1*),  
 modifier = Modifier.*align*(Alignment.Start),  
 fontSize = 20.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*text4\_1*),  
 modifier = Modifier.*align*(Alignment.Start),  
 textAlign = TextAlign.Justify,  
 fontSize = 16.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
 *Text*(  
 text = *stringResource*(id = R.string.*subheading4\_2*),  
 modifier = Modifier.*align*(Alignment.Start),  
 fontSize = 20.*sp* )  
  
 *Spacer*(modifier = Modifier.*height*(20.*dp*))  
  
 *Text*(  
 text = *stringResource*(id = R.string.*text4\_2*),  
 modifier = Modifier.*align*(Alignment.Start),  
 textAlign = TextAlign.Justify,  
 fontSize = 16.*sp* )  
  
  
  
  
 **}**

**Android Manifest.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="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.OwlApplication"  
 tools:targetApi="31">  
 <activity  
 android:name=".LoginActivity"  
 android:exported="false"  
 android:label="@string/title\_activity\_login"  
 android:theme="@style/Theme.OwlApplication" />  
  
 <activity  
 android:name=".RegisterActivity"  
 android:exported="false"  
 android:label="@string/title\_activity\_register"  
 android:theme="@style/Theme.OwlApplication" />  
  
 <activity  
 android:name=".MainActivity5"  
 android:exported="false"  
 android:label="@string/title\_activity\_main5"  
 android:theme="@style/Theme.OwlApplication" />  
 <activity  
 android:name=".MainActivity4"  
 android:exported="false"  
 android:label="@string/title\_activity\_main4"  
 android:theme="@style/Theme.OwlApplication" />  
 <activity  
 android:name=".MainActivity3"  
 android:exported="false"  
 android:label="@string/title\_activity\_main3"  
 android:theme="@style/Theme.OwlApplication" />  
 <activity  
 android:name=".MainActivity2"  
 android:exported="false"  
 android:label="@string/title\_activity\_main2"  
 android:theme="@style/Theme.OwlApplication" />  
  
  
 <activity  
 android:name=".MainActivity"  
 android:exported="true"  
 android:label="@string/app\_name"  
 android:theme="@style/Theme.OwlApplication">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>

</manifest>

**Example unit test.kt:**

package com.example.owlapplication  
  
import org.junit.Test  
  
import org.junit.Assert.\*  
  
*/\*\*  
 \* Example local unit test, which will execute on the development machine (host).  
 \*  
 \* See [testing documentation](http://d.android.com/tools/testing).  
 \*/*class ExampleUnitTest {  
 @Test  
 fun addition\_isCorrect() {  
 assertEquals(4, 2 + 2)  
 }  
}