OWL-M: A MATERIAL DESIGN STUDY APP

Gayathri G
710422104012

DESCRIPTION

- **Owl-M**: Owl-M is an educational app designed to help users understand and explore the principles of Google's Material Design. It serves as a comprehensive study tool for developers and designers aiming to implement Material Design in their projects.
- **Material Design Principles**: The app demonstrates key principles of Material Design, including responsive layouts, dynamic transitions, bold typography, and the effective use of color, shadows, and hierarchy.
- **Interactive Learning**: Users can engage with interactive lessons and examples that showcase how Material Design components like **Floating Action Buttons (FAB)**, **Cards**, and **Bottom Navigation Bars** are implemented.
- **Customizable Themes**: Owl-M features options for light and dark themes, helping users see the impact of Material Design in different visual settings and environments.
- **Design Guidelines**: The app offers detailed explanations of Material Design guidelines, including tips for consistency, accessibility, and creating intuitive user interfaces.
- **Developer Tools**: Owl-M integrates developer-focused tools and resources, such as code snippets, design blueprints, and downloadable assets, to simplify the implementation process.
- **Quizzes and Challenges**: To enhance learning, the app includes quizzes and challenges that test the user's knowledge of Material Design concepts and practices.
- **Modern User Interface**: With a clean, minimalistic UI built on Material Design itself, Owl-M provides a hands-on learning experience while serving as a living example of its principles.

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.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 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.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)
        }
    }
}
```

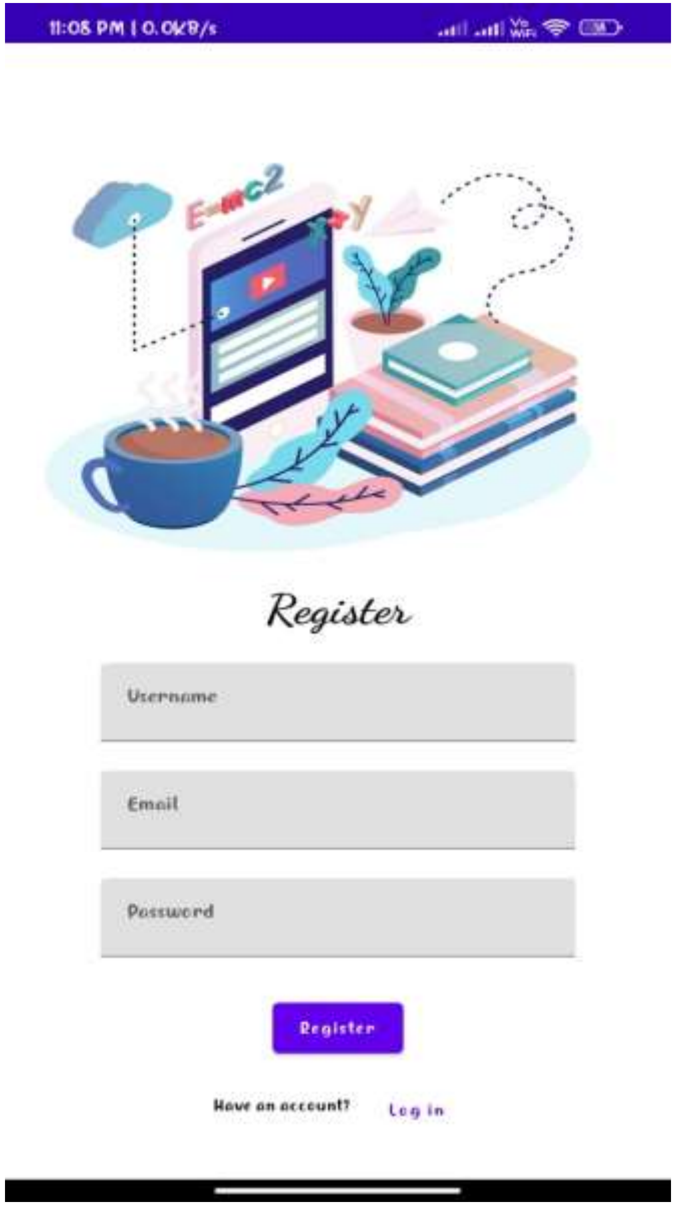
```
@Composable
fun RegistrationScreen(context: Context, databaseHelper: UserDatabaseHelper) {

    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"
        )
```

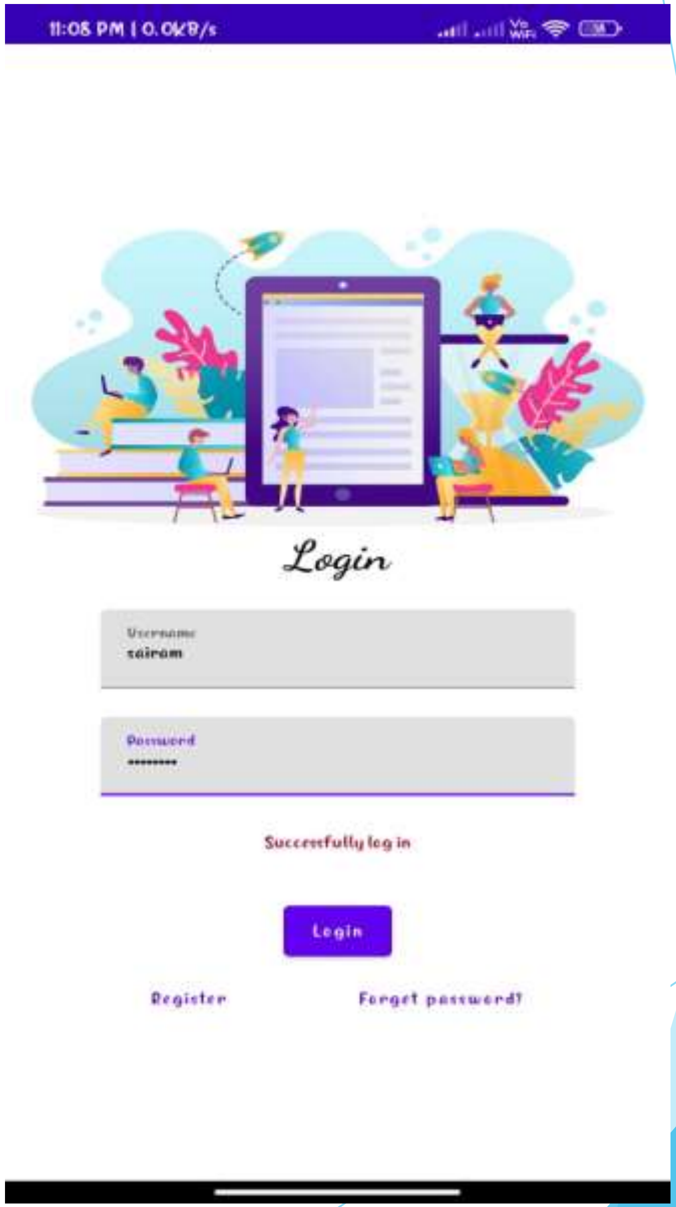
```
Spacer(modifier = Modifier.height(10.dp))
    TextField(
      value = username,
      onValueChange = { username = it },
      label = { Text("Username") },
      modifier = Modifier
        .padding(10.dp)
        .width(280.dp)

    )

    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)
}
```

OUTPUT:

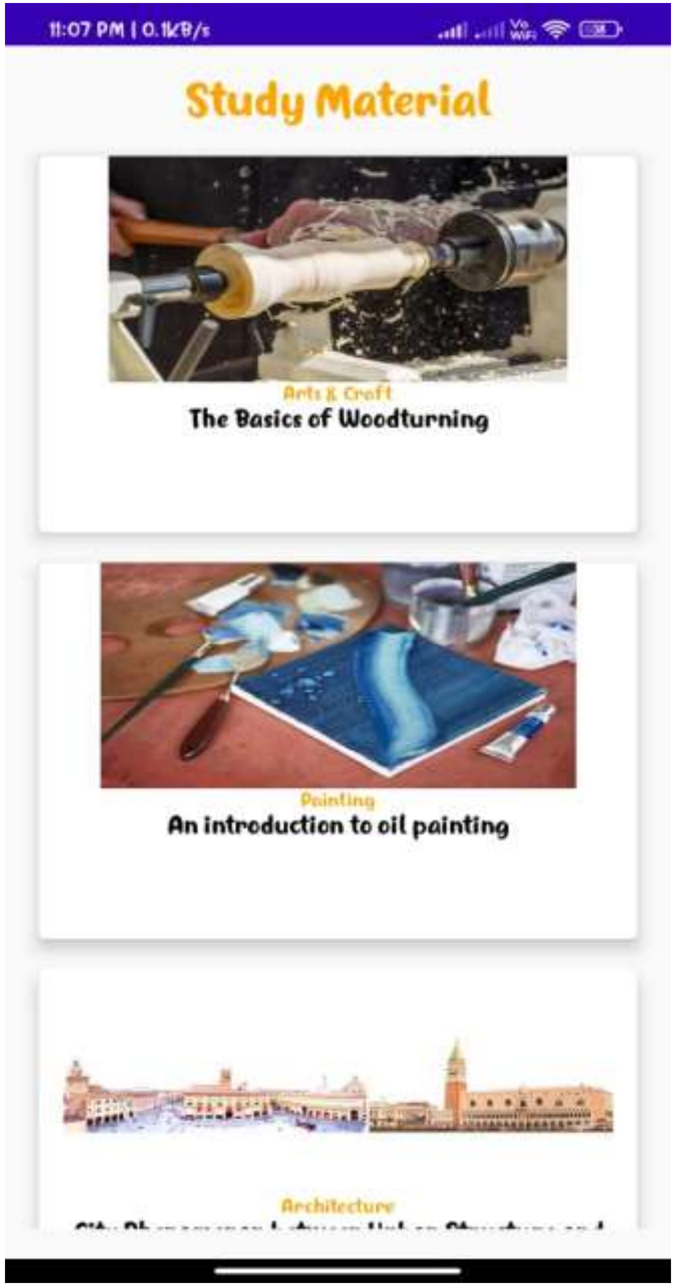




Username
sairam

Password
••••••••

Successfully log in



Study Material

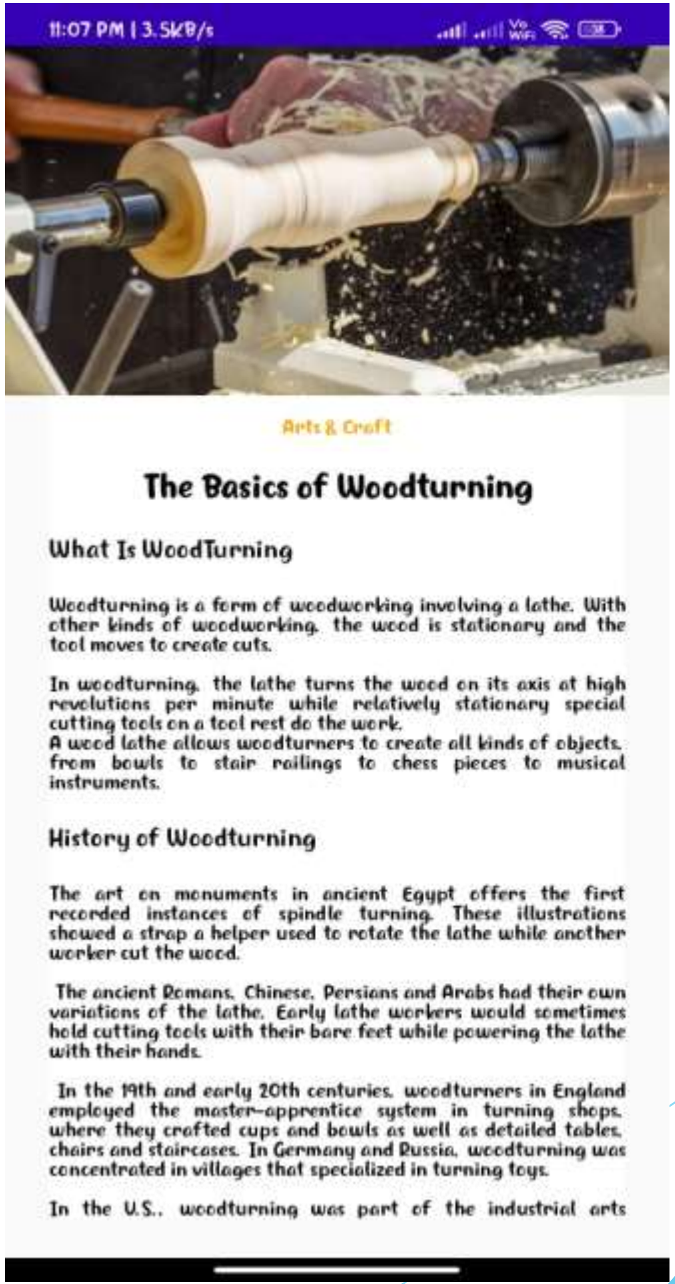
Arts & Craft

The Basics of Woodturning

Painting

An introduction to oil painting

Architecture



Arts & Craft

The Basics of Woodturning

What Is WoodTurning

Woodturning is a form of woodworking involving a lathe. With other kinds of woodworking, the wood is stationary and the tool moves to create cuts.

In woodturning, the lathe turns the wood on its axis at high revolutions per minute while relatively stationary special cutting tools on a tool rest do the work.
A wood lathe allows woodturners to create all kinds of objects, from bowls to stair railings to chess pieces to musical instruments.

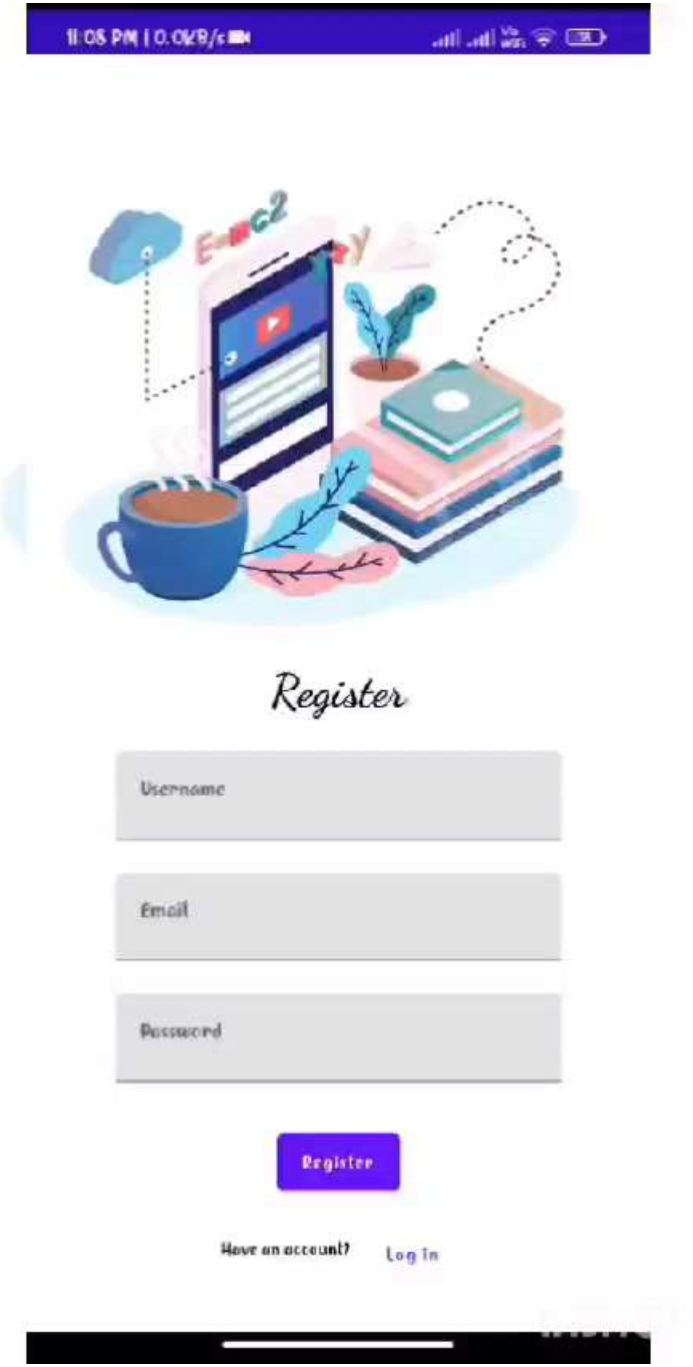
History of Woodturning

The art on monuments in ancient Egypt offers the first recorded instances of spindle turning. These illustrations showed a strap a helper used to rotate the lathe while another worker cut the wood.

The ancient Romans, Chinese, Persians and Arabs had their own variations of the lathe. Early lathe workers would sometimes hold cutting tools with their bare feet while powering the lathe with their hands.

In the 19th and early 20th centuries, woodturners in England employed the master-apprentice system in turning shops, where they crafted cups and bowls as well as detailed tables, chairs and staircases. In Germany and Russia, woodturning was concentrated in villages that specialized in turning toys.

In the U.S., woodturning was part of the industrial arts



Register

Have an account? Log in