package com.project.womensafety.presentationLayer.commonView

import android.annotation.SuppressLint

import android.content.Intent

import android.os.Bundle

import android.view.LayoutInflater

import android.view.View

import android.view.ViewGroup

import androidx.appcompat.app.AppCompatActivity

import androidx.constraintlayout.widget.ConstraintLayout

import androidx.core.view.forEach

import androidx.core.view.get

import androidx.core.view.isVisible

import androidx.fragment.app.Fragment

import androidx.fragment.app.FragmentContainer

import androidx.fragment.app.FragmentContainerView

import com.google.android.material.internal.EdgeToEdgeUtils

import com.project.womensafety.dataLayer.interaction.InteractionView

import com.project.womensafety.databinding.LoginActivityBinding

import com.project.womensafety.presentationLayer.user.UserMainActivity

import com.project.womensafety.responsiveLayer.RetrofitC

import kotlinx.coroutines.CoroutineScope

import kotlinx.coroutines.Dispatchers.IO

import kotlinx.coroutines.Dispatchers.Main

import kotlinx.coroutines.async

import kotlinx.coroutines.withContext

class Login : AppCompatActivity(), InteractionView {

private val bind by lazy {

LoginActivityBinding.inflate(layoutInflater)

}

private val p by lazy {

CommonClass(this).p

}

@SuppressLint("RestrictedApi")

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(bind.root)

bind.create.setOnClickListener {

setViewPort(boolean = true)

}

bind.loginBtn2.setOnClickListener {

val mobile = bind.mobile.text.toString().trim()

val password = bind.password.text.toString().trim()

if (mobile.isEmpty()) {

showToast("Please enter your mobile number")

} else if (password.isEmpty()) {

showToast("Please enter your password")

} else {

p.show()

CoroutineScope(IO).async {

async {

try {

RetrofitC.api.login(

condition = "LoginPart",

mobile = mobile,

password = password

)

} catch (e: Exception) {

withContext(Main) {

p.dismiss()

showToast(e.message)

}

null

}

}.await().let {

withContext(Main) {

it?.body()?.data?.let {

if(it.isEmpty()){

showToast("Invalid User")

}else{

it[0].let { let2->

getSharedPreferences("user", MODE\_PRIVATE).edit().apply {

putString("id",let2.id)

putString("name",let2.name)

putString("mobile",let2.mobile)

putString("location",let2.location)

apply()

}

finishAffinity()

startActivity(Intent(this@Login,UserMainActivity::class.java))

}

}

}

p.dismiss()

}

}

}.start()

}

}

}

private fun setViewPort(boolean: Boolean) {

(bind.root[0] as ConstraintLayout).forEach {

if (it is FragmentContainerView) {

it.isVisible = boolean

} else {

it.isVisible = !boolean

}

}

}

override fun changeView(string: String) {

if (string == "Changer") {

setViewPort(boolean = false)

}

}

}