# Source code: Kotlin

*/\*\*  
 \* this is the implementation of the Intro screen  
 \* the UI was implemented in activity\_intro.xml file  
 \*/*class IntroActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_intro*)  
 *window*.*decorView*.*systemUiVisibility* = View.*SYSTEM\_UI\_FLAG\_FULLSCREEN* var atg = AnimationUtils.loadAnimation(this, R.anim.*atg*)  
 iv\_picstatictic.startAnimation(atg)  
 tv\_welcome.startAnimation(AnimationUtils.loadAnimation(this, R.anim.*bgone*))  
 tv\_description.startAnimation(AnimationUtils.loadAnimation(this, R.anim.*bgone*))  
 l\_buttons.startAnimation(AnimationUtils.loadAnimation(this, R.anim.*bgtwo*))  
  
 var currentUserID = FireStoreClass().getCurrentUserID()  
  
 //STAY LOGGED IN  
 //intent to TractivityMain activity as soon as the user enter the app if the user is logged in already  
 if(currentUserID.*isNotEmpty*()){  
 startActivity(Intent(this, TractivityMain::class.*java*))  
 finish()  
 }  
  
 //direct user to the Login Screen when login button is clicked  
 bt\_login.setOnClickListener**{** val intent = Intent(this, LoginActivity::class.*java*)  
 startActivity(intent)  
 finish()  
 **}** //direct user to the Sign-Up Screen when sign up button is clicked  
 bt\_signUp.setOnClickListener**{** val intent = Intent(this, SignUpActivity::class.*java*)  
 startActivity(intent)  
 finish()  
 **}** }  
}

\*Login Screen of the application

\* GUI designed in activity\_login.xml

\*/

class LoginActivity : BaseActivity() {

private lateinit var auth: FirebaseAuth

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_login)

window.decorView.systemUiVisibility = View.SYSTEM\_UI\_FLAG\_FULLSCREEN

auth = Firebase.auth

bt\_login.setOnClickListener{

loginUser()

}

tv\_signUP.setOnClickListener{

val intent = Intent(this, SignUpActivity::class.java)

startActivity(intent)

finish()

}

tv\_forgot\_pass.setOnClickListener{

startActivity(Intent(this, PasswordResetActivity::class.java))

}

}

//logs in the existing user

private fun loginUser (){

val email:String = et\_email\_login.text.toString().trim{ it <= ' '}

val password:String = et\_password\_login.text.toString()

if(validateLogin(email,password)){

auth.signInWithEmailAndPassword(email, password)

.addOnCompleteListener(this) { task ->

if (task.isSuccessful) {

Log.d("Login", "LoginInWithEmail:success")

val user = auth.currentUser

Toast.makeText(this,

"Welcome back to Tractivity",Toast.LENGTH\_SHORT).show()

val intent = Intent(this, TractivityMain::class.java)

startActivity(intent)

finish()

} else {

// If sign in fails, display a error message to the user.

Log.w("Login", "LogInWithEmail:failure", task.exception)

Toast.makeText(baseContext, "Authentication failed, incorrect email or password was typed",

Toast.LENGTH\_SHORT).show()

}

}

}

}

//validates the login form

private fun validateLogin ( email: String ,

password: String) :Boolean{

when {

TextUtils.isEmpty(email) ->{

showError("Please enter your email address")

return false

}

TextUtils.isEmpty(password) ->{

showError("Please enter your password")

return false

}

else ->{

return true

}

}

}

}

/\*\*

\* The sign Screen of the application was implemented here

\* with Gui designed in activity\_sign\_up.xml file

\*/

class SignUpActivity : BaseActivity() {

private lateinit var auth: FirebaseAuth

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_sign\_up)

window.decorView.systemUiVisibility = View.SYSTEM\_UI\_FLAG\_FULLSCREEN

auth = Firebase.auth

bt\_SignUp.setOnClickListener {

createUser()

}

//Intend to login page

tv\_login.setOnClickListener {

val intent = Intent(this, LoginActivity::class.java)

startActivity(intent)

finish()

}

}

// user is created and stored on authentication server and firestore

private fun createUser(){

val name: String = et\_name.text.toString().trim{ it <= ' '}

val email: String = et\_email.text.toString().trim{ it <= ' '}

val password : String = et\_password.text.toString()

val reTypedPassword : String = et\_reTypePassword.text.toString()

if(validateSignUp(name,email,password,reTypedPassword)){

auth.createUserWithEmailAndPassword(email,password)

.addOnCompleteListener(this){

task ->

if (task.isSuccessful) {

val user = UserClass(Firebase.auth.currentUser!!.uid,name,email)

FireStoreClass().registerUserOnDB(user)

Toast.makeText(this,

"Welcome $name to Tractivity",Toast.LENGTH\_SHORT).show()

val intent = Intent(this, TractivityMain::class.java)

startActivity(intent)

finish()

}else {

Toast.makeText(this,task.exception!!.message.toString(),Toast.LENGTH\_LONG).show()

}

}

}

}

//function to validate the sighUp form

private fun validateSignUp (name:String , email: String ,

password: String, reTypedPassword : String) :Boolean{

when {

TextUtils.isEmpty(name) ->{

showError("Please enter your name")

return false

}

TextUtils.isEmpty(email) ->{

showError("Please enter your email address")

return false

}

TextUtils.isEmpty(password) ->{

showError("Please enter a password")

return false

}

TextUtils.isEmpty(reTypedPassword) ->{

showError("Please re-Type your password")

return false

}

!isPasswordMatched(password,reTypedPassword) ->{

return false

}

else ->{

return true

}

}

}

//checks if both the passwords matches

private fun isPasswordMatched (password:String, reTypedPassword:String) :Boolean{

return if(password == reTypedPassword){

true

}else{

showError("Passwords do not match, please retype")

false

}

}

}

/\*\*

\* this class contains the of the functionality the home screen, Tractivity

\* The stop watch, app navigation and storing the activity data in the database was implemented here.

\* the ui was designed in the activity\_trmain.xml, activity\_stopwatch.xml, dialog\_save.xml

\*/

@Suppress("DEPRECATION")

class TractivityMain : BaseActivity(), NavigationView.OnNavigationItemSelectedListener {

private var pauseTime : Long = 0

private var running : Boolean = false

private var activitySubmited : Boolean = false

val firestore = Firebase.firestore

var progress : String ?= null

var activityName : String =""

var isActivityStarted : Boolean = false

var isActivitySaved : Boolean = false

var isActivitySelected: Boolean = false

val CHANNEL\_ID = "channel\_id"

val CHANNEL\_NAME = "channel\_name"

val NOTIFICATION\_ID = 1

@SuppressLint("SetTextI18n")

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_trmain)

//HIDE STATUS BAR

hideStatusBar()

//CUSTOM ACTION BAR

setupActionBar()

//NAVIGATION LISTENER

navigation.setNavigationItemSelectedListener(this)

val notificationIntent = Intent(this, TractivityMain::class.java)

notificationIntent.setAction(Intent.ACTION\_MAIN)

val pendingIntent : PendingIntent = PendingIntent.getActivity(this,0,notificationIntent,PendingIntent.FLAG\_UPDATE\_CURRENT)

createNotificationChannel()

// to build the notification panel

val buildNotification = NotificationCompat.Builder(this, CHANNEL\_ID)

.setSmallIcon(R.drawable.ic\_stat\_name)

.setContentTitle("Tractivity")

.setContentText("is running")

.setNotificationSilent()

.setContentIntent(pendingIntent)

.setPriority(NotificationCompat.PRIORITY\_DEFAULT).build()

// start and pause the activity

btn\_start.setOnClickListener {

isActivityStarted = true

NotificationManagerCompat.from(this).notify(NOTIFICATION\_ID,buildNotification)

startAndPauseActivity()

}

//STOP STOPWATCH and call Save Activity Dialog

btn\_stop.setOnClickListener{

if( isActivityStarted){

NotificationManagerCompat.from(this).cancel(NOTIFICATION\_ID)

progress = c\_chronometer.text.toString()

pauseStopWatch()

saveDialogFunction()

}else{

Toast.makeText(this, "Activity not started", Toast.LENGTH\_SHORT).show()

}

}

// Reset stopwatch and confirmation dialog

btn\_reset.setOnClickListener {

if(isActivityStarted){

pauseStopWatch()

confirmationDialog()

}

}

populateNavUsername()

}

// start and pause the activity function

private fun startAndPauseActivity(){

if (!running) {

startStopWatch()

} else {

pauseStopWatch()

}

}

// START THE STOPWATCH

private fun startStopWatch(){

iArrow.startAnimation(AnimationUtils.loadAnimation(this, R.anim.rotating\_arrow))

c\_chronometer.base = SystemClock.elapsedRealtime() + pauseTime

c\_chronometer.start()

running = true

btn\_start.text = "Pause"

}

// PAUSE THE STOPWATCH

private fun pauseStopWatch(){

iArrow.clearAnimation()

pauseTime = c\_chronometer.base - SystemClock.elapsedRealtime()

c\_chronometer.stop()

running = false

btn\_start.text= "Resume"

}

//reset the stopwatch

private fun resetStopwatch(){

c\_chronometer.base= SystemClock.elapsedRealtime()

iArrow.clearAnimation()

pauseTime = 0

running=false

btn\_start.text="Start"

}

//display a confirmation dialog

private fun confirmationDialog(){

val confBuilder = AlertDialog.Builder(this)

confBuilder.setTitle("Reset stopwatch?")

confBuilder.setMessage("resetting the stopwatch will cancel the progress")

confBuilder.setPositiveButton("RESET") { dialog: DialogInterface, i: Int ->

resetStopwatch()

isActivityStarted=false

dialog.cancel()

}

confBuilder.setNegativeButton("Cancel") { dialog: DialogInterface, i: Int ->

dialog.cancel()

}

confBuilder.show()

}

//populate username on navigation

fun populateNavUsername(){

FirebaseFirestore.getInstance().collection(Constants.USERS)

.document(FireStoreClass().getCurrentUserID()).get()

.addOnSuccessListener {documentSnapshot ->

val user = documentSnapshot.toObject<UserClass>()!!

val navHeader = navigation.getHeaderView(0)

val navUsername = navHeader.findViewById<TextView>(R.id.tv\_username)

navUsername.text = user.name

}

}

//prevent accidental exit

override fun onBackPressed() {

doubleBackExit()

}

//CUSTOM ACTION BAR

private fun setupActionBar() {

setSupportActionBar(toolbar\_trmain)

toolbar\_trmain.setNavigationIcon(R.drawable.ic\_menu)

toolbar\_trmain.setNavigationOnClickListener {

toggleDrawer()

}

}

//TOGGLE NAVIGATION DRAWER

private fun toggleDrawer() {

if (drawer\_layout.isDrawerOpen(GravityCompat.START)) {

drawer\_layout.closeDrawer(GravityCompat.START)

} else {

drawer\_layout.openDrawer(GravityCompat.START)

}

}

// Intend to the selected activity screen from the navigation menu. Menus were defined in the menu/drawer\_menu.xml file

override fun onNavigationItemSelected(item: MenuItem): Boolean {

when (item.itemId){

R.id.main\_pg -> {

drawer\_layout.closeDrawer(GravityCompat.START)

return true

}

R.id.activities ->{

val intent = Intent(this, ActivitiesActivity::class.java)

startActivity(intent)

}

R.id.charts ->{

val intent = Intent(this, ChartsActivity::class.java)

startActivity(intent)

}

R.id.profile ->{

val intent = Intent(this, UserProfileActivity::class.java)

startActivity(intent)

}

R.id.logout ->{

FirebaseAuth.getInstance().signOut()

val intent = Intent(this, IntroActivity::class.java)

intent.addFlags(Intent.FLAG\_ACTIVITY\_CLEAR\_TOP or Intent.FLAG\_ACTIVITY\_NEW\_TASK)

startActivity(intent)

finish()

}

}

drawer\_layout.closeDrawer(GravityCompat.START)

return true

}

//creating a notification channel

private fun createNotificationChannel() {

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.O) {

val importance = NotificationManager.IMPORTANCE\_DEFAULT

val channel1 = NotificationChannel(CHANNEL\_ID, CHANNEL\_NAME, importance).apply {

//behaviour of the notification

setSound(null,null)

}

// Register the channel with the system

val notificationManager: NotificationManager =

getSystemService(Context.NOTIFICATION\_SERVICE) as NotificationManager

notificationManager.createNotificationChannel(channel1)

}

}

//SAVE ACTIVITY CUSTOM DIALOG inflated from dialog\_save.xml file

@SuppressLint("SetTextI18n")

private fun saveDialogFunction() {

val saveDialogView = LayoutInflater.from(this).inflate(R.layout.dialog\_save,null)

saveDialogView.tv\_description.text= "Following $progress is spent on:"

val saveDialogbuilder = AlertDialog.Builder(this)

.setView(saveDialogView)

val saveActivityDialog = saveDialogbuilder.show()

saveActivityDialog.setCancelable(false) // prevent user to close the dialog by clicking outside the dialog

saveDialogView.bt\_selectActivity.setOnClickListener {

// ACTIVITY NAMES DIALOG populated with the activity lists retrieved from the DB

firestore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES)

.get()

.addOnSuccessListener { activities ->

Log.d("DB", "All documents received")

val activityNameList = ArrayList<String>()

for (activity in activities) {

val activityClass: ActivityClass? = activity.toObject(ActivityClass::class.java)

if (activityClass != null){

activityNameList.add(activityClass.name)

}

}

val items:Array<String> = activityNameList.toTypedArray()

val selectActivityBuilder = AlertDialog.Builder(this)

selectActivityBuilder.setTitle("Choose activity")

selectActivityBuilder.setSingleChoiceItems(items,-1){

dialogInterface: DialogInterface, i :Int ->

saveDialogView.et\_activityName.setText(items[i])

isActivitySelected = true

dialogInterface.dismiss()

}

selectActivityBuilder.setNeutralButton("cancel"){ dialog:DialogInterface, which->

dialog.cancel()

}

val activitySelectDialog = selectActivityBuilder.create()

activitySelectDialog.show()

}

.addOnFailureListener { exception ->

Log.d("DB", "Error getting documents: ", exception)

}

}

// storing the data on the firestore once the submit button in clicked

saveDialogView.bt\_submit.setOnClickListener{

activityName = saveDialogView.et\_activityName.text.toString()

if (TextUtils.isEmpty(activityName)){

Toast.makeText(applicationContext, "Enter or select Activity Name", Toast.LENGTH\_SHORT).show()

} else {

val activity = ActivityClass(activityName)

val record = ActivityRecordClass(parseProgress(progress!!))

if(isActivitySelected){

FireStoreClass().saveRecordOnDB(record,activity.name)

} else{

FireStoreClass().saveActivityOnDB(activity)

FireStoreClass().saveRecordOnDB(record,activity.name)

}

isActivitySelected = false

isActivitySelected = false

resetStopwatch()

saveActivityDialog.dismiss()

}

}

saveDialogView.bt\_cancel.setOnClickListener{

Toast.makeText(applicationContext, "Activity not saved", Toast.LENGTH\_SHORT).show()

saveActivityDialog.dismiss()

}

}

//PARSE STRING PROGRESS TO LONG (mm:ss or hh:mm:ss)

fun parseProgress (progress:String) : Long{

var progressInSeconds :Long = 0

val strArray : Array<String> = progress.split(":").toTypedArray()

if(strArray.size<3){

progressInSeconds += strArray[0].toLong()\*60

progressInSeconds += strArray[1].toLong()

return progressInSeconds

}

progressInSeconds += strArray[0].toLong()\*3600

progressInSeconds += strArray[1].toLong()\*60

progressInSeconds += strArray[2].toLong()

return progressInSeconds

}

}

/\*\*

\* The activity list screen was implemented in this file

\* the GUI was designed in the activity\_activities.xml file

\* resources used: https://www.youtube.com/watch?v=6Gm3eMG8KqI

https://www.youtube.com/watch?v=afl\_i6uvvU0

https://guides.codepath.com/android/using-the-recyclerview

\*/

@Suppress("DEPRECATION")

class ActivitiesActivity : AppCompatActivity() {

private val firestore = Firebase.firestore

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_activities)

window.setFlags(

WindowManager.LayoutParams.FLAG\_FULLSCREEN,

WindowManager.LayoutParams.FLAG\_FULLSCREEN

)

displayActivityListFromDB()

iv\_back.setOnClickListener{

onBackPressed()

}

iv\_add.setOnClickListener{

val intent = Intent(this, AddActivity::class.java)

startActivity(intent)

finish()

}

}

// POPULATE RECYCLEVIEW

private fun populateActivityListRv (activityList: ArrayList<ActivityClass>){

if (activityList.size> 0){

rv\_activitiesList.layoutManager = LinearLayoutManager(this)

rv\_activitiesList.setHasFixedSize(true)

val adapter = ActivityAdapter(activityList)

rv\_activitiesList.adapter = adapter

adapter.setOnclickListener(object: ActivityAdapter.OnItemClickListener {

override fun onItemClick(position: Int, item: ActivityClass) {

val intent = Intent(this@ActivitiesActivity, DetailsActivity::class.java)

intent.putExtra(Constants.ACTIVITYNAME,item.name)

startActivity(intent)

}

})

}else{

Toast.makeText(this, "no activity stored", Toast.LENGTH\_SHORT).show()

}

}

// retrieve the activity details from the firestore and display them

private fun displayActivityListFromDB(){

firestore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES)

.get()

.addOnSuccessListener { results ->

Log.d("DB", "All documents received")

val activityList = ArrayList<ActivityClass>()

for (result in results) {

val activity: ActivityClass? = result.toObject(ActivityClass::class.java)

if (activity != null) {

activityList.add(activity)

}

}

populateActivityListRv(activityList)

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting documents: ", exception)

}

}

}

/\*\*

\* Add/Created activity manually was implemnted in this class

\* the gui design: activity\_add\_activity.xml

\*/

class AddActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_add\_activity)

window.setFlags(

WindowManager.LayoutParams.FLAG\_FULLSCREEN,

WindowManager.LayoutParams.FLAG\_FULLSCREEN

)

setupActionBar()

//store the activity on the firestore

bt\_add.setOnClickListener{

val activityName = et\_activityName.text.toString()

val description = et\_description.text.toString()

val dueDate = et\_dueDate.text.toString()

if (TextUtils.isEmpty(activityName)) {

Toast.makeText(

applicationContext,

"Please Enter Activity Name",

Toast.LENGTH\_SHORT

).show()

} else {

val activity = ActivityClass(activityName,

SimpleDateFormat("dd/MM/yyyy", Locale.getDefault()).format(Date())

,description, dueDate)

FireStoreClass().saveActivityOnDB(activity)

val intent = Intent(this, ActivitiesActivity::class.java)

startActivity(intent)

finish()

}

}

}

//Custom app bar

private fun setupActionBar() {

setSupportActionBar(toolbar\_addActivity)

val actionBar = supportActionBar

actionBar!!.setDisplayHomeAsUpEnabled(true)

actionBar.setHomeAsUpIndicator(R.drawable.ic\_back)

actionBar.title = "Add Activity"

toolbar\_addActivity.setNavigationOnClickListener{

val intent = Intent(this, ActivitiesActivity::class.java)

startActivity(intent)

finish()

}

}

}

/\*\*

\* some of the basic functionalities that were used in more than one class were implemented in this class and inherited from.

\*/

@Suppress("DEPRECATION")

open class BaseActivity : AppCompatActivity() {

private var backPressedOnce = false

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_base)

}

fun hideStatusBar(){

window.setFlags(

WindowManager.LayoutParams.FLAG\_FULLSCREEN,

WindowManager.LayoutParams.FLAG\_FULLSCREEN

)

}

fun getCurrentUserID():String{

return FirebaseAuth.getInstance().currentUser!!.uid

}

//prevent user from accidentally exiting the app

fun doubleBackExit() {

if(backPressedOnce){

super.onBackPressed()

return

}else{

this.backPressedOnce = true

Toast.makeText(this, "press back again to exit",Toast.LENGTH\_SHORT).show()

Handler().postDelayed({backPressedOnce = false},2000 )

}

}

//display error massage as snack bar

fun showError(errorMessage:String){

val snackBar= Snackbar.make(findViewById(android.R.id.content),errorMessage,Snackbar.LENGTH\_LONG)

snackBar.show()

}

}

/\*\*

\* the activity details screen was implented in this class with the GUI designed in activity\_details.xml file

\*/

class DetailsActivity : AppCompatActivity() {

private val fireStore = Firebase.firestore

private lateinit var activityName : String

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_details)

window.setFlags(

WindowManager.LayoutParams.FLAG\_FULLSCREEN,

WindowManager.LayoutParams.FLAG\_FULLSCREEN

)

if(intent.hasExtra(Constants.ACTIVITYNAME)){

activityName = intent.getStringExtra(Constants.ACTIVITYNAME)!!

}

displayActivityDetailsFromDB()

displayRecordListFromDB()

displayProgress()

iv\_back.setOnClickListener{

onBackPressed()

}

iv\_delete.setOnClickListener{

confirmationDialog()

}

iv\_edit.setOnClickListener{

displayActivityEditDialog()

}

}

//query the progress records from the database and display in the recycleView

fun displayRecordListFromDB(){

fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activityName).collection(Constants.RECORDS)

.get()

.addOnSuccessListener { records ->

Log.d("DB", "All documents received")

val recordList = ArrayList<ActivityRecordClass>()

for (record in records) {

val record: ActivityRecordClass? = record.toObject(ActivityRecordClass::class.java)

if (record != null) {

recordList.add(record)

}

}

populateRecordsRv(recordList)

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting documents: ", exception)

}

}

//populate teh record RecycleView with retrieved data from the firestore

private fun populateRecordsRv (recordList: ArrayList<ActivityRecordClass>) {

if (recordList.size> 0){

rv\_recordList.layoutManager = LinearLayoutManager(this)

rv\_recordList.setHasFixedSize(true)

val adapter = RecordAdapter(recordList)

rv\_recordList.adapter = adapter

}

}

//query the details of the activity clicked from the database

private fun displayActivityDetailsFromDB (){

fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activityName).get()

.addOnSuccessListener{ document ->

Log.d("DB", "document received")

val activity : ActivityClass? = document.toObject(ActivityClass::class.java)

populateActivityDetails(activity!!)

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting the document: ", exception)

}

}

// Populate the details sections

@SuppressLint("SetTextI18n")

fun populateActivityDetails(activity:ActivityClass){

tv\_activityName.text=activity.name

tv\_description.text= activity.description

tv\_startedDate.text= "Started Date: ${activity.date}"

tv\_dueDate.text= "Due date: ${activity.dueDate}"

}

// Delete an activity from the database

private fun deleteActivity(){

fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activityName).delete()

.addOnSuccessListener {

Log.d("DATA", "Activity has been deleted")

Toast.makeText(this,

"Activity has been deleted", Toast.LENGTH\_SHORT).show()

}

.addOnFailureListener { e -> Log.w("DATA", "Error deleting document", e) }

}

//display and confirmation dialog before deleting the activity

private fun confirmationDialog(){

val confBuilder = AlertDialog.Builder(this)

confBuilder.setTitle("Delete Activity?")

confBuilder.setPositiveButton("YES") { dialog: DialogInterface, i: Int ->

deleteActivity()

val intent = Intent(this, ActivitiesActivity::class.java)

intent.addFlags(Intent.FLAG\_ACTIVITY\_CLEAR\_TOP or Intent.FLAG\_ACTIVITY\_NEW\_TASK)

startActivity(intent)

finish()

dialog.cancel()

}

confBuilder.setNegativeButton("NO") { dialog: DialogInterface, i: Int ->

dialog.cancel()

}

confBuilder.show()

}

//Display a custom dialog to let user edit the activity details

// the dialog was designed in the dialog\_edit\_activity.xml

private fun displayActivityEditDialog(){

val editDialogView = LayoutInflater.from(this).inflate(R.layout.dialog\_edit\_activity,null)

fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activityName).get()

.addOnSuccessListener{ document ->

Log.d("DB", "document received")

val activity : ActivityClass? = document.toObject(ActivityClass::class.java)

if (activity != null) {

editDialogView.et\_description.setText(activity.description)

editDialogView.et\_dueDate.setText(activity.dueDate)

val editDialogBuilder =AlertDialog.Builder(this).setView(editDialogView)

val editDialog = editDialogBuilder.show()

editDialogView.bt\_submit.setOnClickListener {

val description = editDialogView.et\_description.text.toString()

val dueDate = editDialogView.et\_dueDate.text.toString()

fireStore.collection(Constants.USERS)

.document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activityName).update(

mapOf(

"description" to description,

"dueDate" to dueDate

)

)

val intent = Intent(this, ActivitiesActivity::class.java)

intent.addFlags(Intent.FLAG\_ACTIVITY\_CLEAR\_TOP or Intent.FLAG\_ACTIVITY\_NEW\_TASK)

startActivity(intent)

finish()

Toast.makeText(this, "Activity details changed", Toast.LENGTH\_SHORT).show()

editDialog.dismiss()

}

editDialogView.bt\_cancel.setOnClickListener {

editDialog.cancel()

}

}

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting the document: ", exception)

}

}

//display the total time spent in a ctivity

@SuppressLint("SetTextI18n")

fun displayProgress(){

fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activityName).collection(Constants.RECORDS).get()

.addOnSuccessListener{ documents ->

Log.d("DB", "collection received")

var totalTime = 0L

for (document in documents){

val record : ActivityRecordClass = document.toObject(ActivityRecordClass::class.java)

totalTime += record.progress

}

tv\_progress.text= "Total time spent: ${formatProgress(totalTime)}"

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting the document: ", exception)

}

}

//format the total progress from seconds to string in hh mm ss format

private fun formatProgress(progressInSec : Long) : String{

val hours = progressInSec / 3600

val minutes = (progressInSec % 3600) / 60

val seconds = progressInSec % 60

return when {

hours >0 -> {

String.format("%02dh %02dm %02ds", hours, minutes, seconds)

}

minutes>0 -> {

String.format("%02dm %02ds",minutes, seconds)

}

else -> {

String.format("%02ds", seconds)

}

}

}

}

/\*\*

\* Displaying PieCharts and bar charts were implemented here

\* with the GUI implemented in the activity.charts.xml

\*/

class ChartsActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_charts)

window.setFlags(

WindowManager.LayoutParams.FLAG\_FULLSCREEN,

WindowManager.LayoutParams.FLAG\_FULLSCREEN

)

setupActionBar()

showPieChart()

showBarChart()

}

//query and display activity data in bar chart

private fun showBarChart(){

FireStoreClass().fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES)

.get()

.addOnSuccessListener { documents ->

Log.d("DB", "All documents received")

val activityBarEntries = ArrayList<BarEntry>()

val activityBarLabels = ArrayList<String>()

var index = 0f

for (document in documents) {

var totalProgress: Long = 0

val activity = document.toObject(ActivityClass::class.java)

FireStoreClass().fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activity.name).collection(

Constants.RECORDS)

.get().addOnSuccessListener { results ->

for (result in results){

val record : ActivityRecordClass = result.toObject(ActivityRecordClass::class.java)

totalProgress += record.progress

}

if (totalProgress > 0){

activityBarEntries.add(BarEntry(index++,totalProgress.toFloat()))

activityBarLabels.add(activity.name)

}

createBarChart(activityBarEntries,activityBarLabels)

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting records: ", exception)

}

}

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting activities: ", exception)

}

}

//query and display activity data in Pie chart

private fun showPieChart(){

FireStoreClass().fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES)

.get()

.addOnSuccessListener { documents ->

Log.d("DB", "All documents received")

val activityPieData = ArrayList<PieEntry>()

for (document in documents) {

var totalProgress: Long = 0

val activity = document.toObject(ActivityClass::class.java)

FireStoreClass().fireStore.collection(Constants.USERS).document(FireStoreClass().getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activity.name).collection(

Constants.RECORDS)

.get().addOnSuccessListener { results ->

for (result in results){

val record : ActivityRecordClass = result.toObject(ActivityRecordClass::class.java)

totalProgress += record.progress

}

if (totalProgress > 0){

activityPieData.add(PieEntry(totalProgress.toFloat(), activity.name))

}

createPieChart(activityPieData)

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting records: ", exception)

}

}

}.addOnFailureListener { exception ->

Log.d("DB", "Error getting activities: ", exception)

}

}

// Populate the Barchart view with defined features

private fun createBarChart(activityBarEntries: ArrayList<BarEntry>, activityBarLabels : ArrayList<String>){

var colors :ArrayList<Int> = ArrayList<Int>()

for (color in ColorTemplate.JOYFUL\_COLORS){

colors.add(color)

}

for (color in ColorTemplate.COLORFUL\_COLORS){

colors.add(color)

}

val barDataSet = BarDataSet(activityBarEntries,"Bar data")

barDataSet.valueTextSize=10f

barDataSet.valueTextColor = Color.BLACK

barDataSet.colors=colors

barDataSet.valueFormatter= ProgressValueFormatter()

val barData = BarData(barDataSet)

crt\_barChart.setDrawGridBackground(false)

crt\_barChart.setFitBars(true)

crt\_barChart.xAxis.valueFormatter= IndexAxisValueFormatter(activityBarLabels)

crt\_barChart.xAxis.position = XAxis.XAxisPosition.BOTTOM

crt\_barChart.description.isEnabled= false

crt\_barChart.data = barData

crt\_barChart.axisRight.isEnabled = false

crt\_barChart.invalidate()

}

// Populate the Pie Chart view with defined features

private fun createPieChart(activityData: ArrayList<PieEntry>){

var colors :ArrayList<Int> = ArrayList<Int>()

for (color in ColorTemplate.JOYFUL\_COLORS){

colors.add(color)

}

for (color in ColorTemplate.COLORFUL\_COLORS){

colors.add(color)

}

val pieDataSet = PieDataSet(activityData,"Progress")

pieDataSet.valueFormatter= ProgressValueFormatter()

pieDataSet.colors = colors

pieDataSet.valueTextSize = 10f

pieDataSet.valueTextColor = Color.BLACK

pieDataSet.sliceSpace= 2f

pieDataSet.xValuePosition = PieDataSet.ValuePosition.OUTSIDE\_SLICE

val pieData = PieData(pieDataSet)

crt\_pieChart.data = pieData

crt\_pieChart.setEntryLabelColor(Color.BLACK)

crt\_pieChart.legend.isWordWrapEnabled = true

crt\_pieChart.animateXY(50, 50)

crt\_pieChart.description.isEnabled = false

crt\_pieChart.centerText="Time Spent"

crt\_pieChart.invalidate()

}

//CUSTOM ACTION BAR

private fun setupActionBar() {

setSupportActionBar(toolbar\_charts)

val actionBar = supportActionBar

actionBar!!.setDisplayHomeAsUpEnabled(true)

actionBar.setHomeAsUpIndicator(R.drawable.ic\_back)

actionBar.title = "Charts"

toolbar\_charts.setNavigationOnClickListener{

onBackPressed() }

}

//A class to override the chart value format

private class ProgressValueFormatter : ValueFormatter(){

override fun getFormattedValue(value:Float):String {

return formatProgress(value.toLong())

}

private fun formatProgress(progressInSec : Long) : String{

val hours = progressInSec / 3600

val minutes = (progressInSec % 3600) / 60

val seconds = progressInSec % 60

return when {

hours >0 -> {

String.format("%02dh%02dm%02ds", hours, minutes, seconds)

}

minutes>0 -> {

String.format("%02dm%02ds",minutes, seconds)

}

else -> {

String.format("%02ds", seconds)

}

}

}

}

}

import android.app.AlertDialog

import android.content.ContentValues

import android.os.Bundle

import android.util.Log

import android.view.LayoutInflater

import android.widget.Toast

import com.google.firebase.auth.FirebaseAuth

import com.google.firebase.firestore.ktx.firestore

import com.google.firebase.ktx.Firebase

import com.tanvir.tractivity.R

import com.tanvir.tractivity.constants.Constants

import com.tanvir.tractivity.model.UserClass

import kotlinx.android.synthetic.main.activity\_user\_profile.\*

import kotlinx.android.synthetic.main.dialog\_change\_pass.\*

import kotlinx.android.synthetic.main.dialog\_change\_pass.view.\*

/\*\*

\* this class contains teh implementation on the profile screen

\* the Gui was designed in teh activity\_user\_profile.xml file

\*/

class UserProfileActivity : BaseActivity(){

lateinit var gbloggedUser : UserClass

val firestore = Firebase.firestore

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_user\_profile)

hideStatusBar()

setupActionBar()

populateProfileActivity()

bt\_change\_pass.setOnClickListener{

changePass()

}

}

//CUSTOM ACTION BAR

private fun setupActionBar() {

setSupportActionBar(toolbar\_profile)

val actionBar = supportActionBar

actionBar!!.setDisplayHomeAsUpEnabled(true)

actionBar.setHomeAsUpIndicator(R.drawable.ic\_back)

actionBar.title = "Profile"

toolbar\_profile.setNavigationOnClickListener{

onBackPressed() }

}

// POPULATE THE PROFILE PAGE WITH USER DATA FROM DB

fun populateProfileActivity () {

firestore.collection(Constants.USERS)

.document(getCurrentUserID()).get()

.addOnSuccessListener {documentSnapshot ->

Log.d(ContentValues.TAG, "Document received")

val loggedUser: UserClass? = documentSnapshot.toObject(UserClass::class.java)

if (loggedUser != null) {

gbloggedUser = loggedUser

tv\_username\_profile.text = loggedUser.name

tv\_profile\_name.text=loggedUser.name

tv\_profile\_email.text = loggedUser.email

tv\_score.text = ""

}

}

.addOnFailureListener{

exception ->

Log.d(ContentValues.TAG, "get failed with ", exception)

}

}

//Display a custom dialog designed in dialog\_change\_pass that allows users to change their password

// the new entered password will be update don the authentication server

private fun changePass(){

val changePassDialogView = LayoutInflater.from(this).inflate(R.layout.dialog\_change\_pass,null)

val changePassDialogBuilder = AlertDialog.Builder(this).setView(changePassDialogView)

val changePassDialog = changePassDialogBuilder.show()

changePassDialogView.bt\_submit.setOnClickListener{

val password = changePassDialogView.et\_pass.text.toString()

val reTypedPassword = changePassDialogView.et\_rePass.text.toString()

if(isPasswordMatched(password,reTypedPassword)){

FirebaseAuth.getInstance().currentUser!!.updatePassword(password).addOnCompleteListener{

task ->

if(task.isSuccessful){

Toast.makeText(this, "Password changed Successfully",

Toast.LENGTH\_SHORT).show()

changePassDialog.dismiss()

}else{

Toast.makeText(this,task.exception!!.message.toString(),Toast.LENGTH\_LONG).show()

}

}

}else{

Toast.makeText(this, "Passwords do not match",

Toast.LENGTH\_SHORT).show()

}

}

changePassDialog.bt\_cancel.setOnClickListener{

changePassDialog.cancel()

}

}

//checks if both the passwords matches

private fun isPasswordMatched (password:String, reTypedPassword:String) :Boolean{

return password == reTypedPassword

}

}

/\*\*

\* the Adapter class of Activity list RecycleView

\* source followed: https://developer.android.com/guide/topics/ui/layout/recyclerview

\* https://www.youtube.com/watch?v=6Gm3eMG8KqI

\* https://www.youtube.com/watch?v=afl\_i6uvvU0

\* https://guides.codepath.com/android/using-the-recyclerview

\*/

class ActivityAdapter (private val list : ArrayList<ActivityClass> ) :RecyclerView.Adapter<RecyclerView.ViewHolder>(){

private var onItemClickListener : OnItemClickListener? = null

override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): RecyclerView.ViewHolder {

return ActivitiesViewHolder(LayoutInflater.from(parent.context)

.inflate(R.layout.item\_activity,parent,false))

}

override fun onBindViewHolder(holder: RecyclerView.ViewHolder, position: Int) {

val item = list[position]

if (holder is ActivitiesViewHolder){

holder.itemView.tv\_activityName.text=item.name

holder.itemView.tv\_date.text = item.date

holder.itemView.setOnClickListener{

if(onItemClickListener!=null){

onItemClickListener!!.onItemClick(position,item)

}

}

}

}

override fun getItemCount(): Int {

return list.size

}

class ActivitiesViewHolder(itemView:View) : RecyclerView.ViewHolder(itemView){

val textView1 : TextView = itemView.tv\_activityName

val textView2 : TextView = itemView.tv\_date

}

fun setOnclickListener(onClickListener: OnItemClickListener){

this.onItemClickListener = onClickListener

}

interface OnItemClickListener {

fun onItemClick(position: Int, item: ActivityClass)

}

}

/\*\*

\* the Adapter class of Record list RecycleView

\* source followed: https://developer.android.com/guide/topics/ui/layout/recyclerview

\* https://www.youtube.com/watch?v=6Gm3eMG8KqI

\* https://www.youtube.com/watch?v=afl\_i6uvvU0

\* https://guides.codepath.com/android/using-the-recyclerview

\*/

class RecordAdapter (private val list : ArrayList<ActivityRecordClass> ) :

RecyclerView.Adapter<RecyclerView.ViewHolder>() {

override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): RecyclerView.ViewHolder {

return RecordViewHolder(

LayoutInflater.from(parent.context)

.inflate(R.layout.item\_details, parent, false)

)

}

override fun onBindViewHolder(holder: RecyclerView.ViewHolder, position: Int) {

val item = list[position]

if (holder is RecordViewHolder){

holder.itemView.tv\_date.text = item.date

holder.itemView.tv\_timeSpent.text= formatProgress(item.progress)

}

}

override fun getItemCount(): Int {

return list.size

}

class RecordViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView){

}

private fun formatProgress(progressInSec : Long) : String{

val hours = progressInSec / 3600

val minutes = (progressInSec % 3600) / 60;

val seconds = progressInSec % 60;

return String.format("%02d:%02d:%02d", hours, minutes, seconds);

}

}

/\*\*

\* Constant variable used though out the development

\*/

object Constants {

const val USERS: String ="Users"

const val ACTIVITIES : String = "activities"

const val RECORDS: String = "records"

const val ACTIVITYNAME : String = "activityName"

}

/\*\*

\* A data class to hold the Activity details

\*/

data class ActivityClass(

val name: String = "",

val date : String = SimpleDateFormat("dd/MM/yyyy",Locale.getDefault()).format(Date()),

val description: String = "",

val dueDate : String = ""

) : Parcelable {

constructor(parcel: Parcel) : this(

parcel.readString()!!,

parcel.readString()!!,

parcel.readString()!!,

parcel.readString()!!) {

}

override fun writeToParcel(parcel: Parcel, flags: Int) {

parcel.writeString(name)

parcel.writeString(date)

parcel.writeString(description)

parcel.writeString(dueDate)

}

override fun describeContents(): Int {

return 0

}

companion object CREATOR : Parcelable.Creator<ActivityClass> {

override fun createFromParcel(parcel: Parcel): ActivityClass {

return ActivityClass(parcel)

}

override fun newArray(size: Int): Array<ActivityClass?> {

return arrayOfNulls(size)

}

}

}

/\*\*

\* A data class to hold the record details

\*/

data class ActivityRecordClass (

val progress : Long = 0,

val date : String = SimpleDateFormat("dd/MM/yyyy", Locale.getDefault()).format(Date())

) : Parcelable {

constructor(parcel: Parcel) : this(

parcel.readLong(),

parcel.readString()!!) {

}

override fun writeToParcel(parcel: Parcel, flags: Int) {

parcel.writeLong(progress)

parcel.writeString(date)

}

override fun describeContents(): Int {

return 0

}

companion object CREATOR : Parcelable.Creator<ActivityRecordClass> {

override fun createFromParcel(parcel: Parcel): ActivityRecordClass {

return ActivityRecordClass(parcel)

}

override fun newArray(size: Int): Array<ActivityRecordClass?> {

return arrayOfNulls(size)

}

}

}

/\*\*

\* All the methods of storing the data on the firestore were implemented in this class

\* note: retrieving data methods were not implemented in this file

\* resources used: https://firebase.google.com/docs/auth/android/password-auth

\* https://firebase.google.com/docs/firestore/quickstart#android

\* https://firebase.google.com/docs/auth/android/manage-users

\* https://www.youtube.com/watch?v=dRYnm\_k3w1w&t=269s

\* https://firebase.google.com/docs/firestore/manage-data/add-data

\*/

class FireStoreClass {

val fireStore = Firebase.firestore

private val auth = FirebaseAuth.getInstance()

//store the user info on the firestore

fun registerUserOnDB (user: UserClass){

fireStore.collection(Constants.USERS).document(getCurrentUserID())

.set(user, SetOptions.merge()).addOnSuccessListener {

Log.d("DDDBB", "Document saved")

}

.addOnFailureListener{ e->

Log.w("DDDBB", "Error adding document")

}

}

// get the ID of currently logged in user from Authentication server

fun getCurrentUserID() : String {

var currentUser = auth.currentUser

var currentUserID= ""

if(currentUser != null){

currentUserID =currentUser.uid

}

return currentUserID

}

// this function takes a Activity object class and store it in the Firetore as a document in the Activity collection

fun saveActivityOnDB(activity: ActivityClass) {

fireStore.collection(Constants.USERS)

.document(getCurrentUserID())

.collection(Constants.ACTIVITIES).document(activity.name).set(activity, SetOptions.merge())

.addOnSuccessListener {

Log.d("DDDBB", "Activity saved")

}

.addOnFailureListener{ e ->

Log.e("DDDBB", "activity not saved")

}

}

// store the activity record on the firestore

fun saveRecordOnDB(record: ActivityRecordClass, activityName :String){

fireStore.collection(Constants.USERS)

.document(getCurrentUserID())

.collection(Constants.ACTIVITIES)

.document(activityName).collection(Constants.RECORDS)

.add(record).addOnSuccessListener {

Log.d("DDDBB", "record saved")

}

.addOnFailureListener{ e ->

Log.e("DDDBB", "record not saved")

}

}

}

/\*\*

\* A data class to hold the User details

\*/

data class UserClass (

val id: String = "",

val name : String="",

val email :String ="",

val score : Long = 0

) : Parcelable {

constructor(parcel: Parcel) : this(

parcel.readString()!!,

parcel.readString()!!,

parcel.readString()!!,

parcel.readLong()

) {

}

override fun writeToParcel(parcel: Parcel, flags: Int) {

parcel.writeString(id)

parcel.writeString(name)

parcel.writeString(email)

parcel.writeLong(score)

}

override fun describeContents(): Int {

return 0

}

companion object CREATOR : Parcelable.Creator<UserClass> {

override fun createFromParcel(parcel: Parcel): UserClass {

return UserClass(parcel)

}

override fun newArray(size: Int): Array<UserClass?> {

return arrayOfNulls(size)

}

}

}

# Source code: XML

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"

android:orientation="vertical"

android:gravity="center"

tools:context=".activities.ActivitiesActivity">

<include layout="@layout/app\_bar2"/>

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/rv\_activitiesList"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:padding="5dp"

tools:listitem="@layout/item\_activity"

/>

</LinearLayout>

<LinearLayout

xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

tools:context=".activities.AddActivity">

<com.google.android.material.appbar.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="30dp"

>

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbar\_addActivity"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="#6C4FF1"

/>

</com.google.android.material.appbar.AppBarLayout>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="8dp"

android:orientation="vertical">

<EditText

android:id="@+id/et\_activityName"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Activity name"

android:inputType="text"

android:textSize="20sp"

/>

<EditText

android:id="@+id/et\_description"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="description"

android:inputType="text"

android:textSize="20sp"

/>

<EditText

android:id="@+id/et\_dueDate"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="due date"

android:inputType="date"

android:textSize="20sp"

/>

</LinearLayout>

<android.widget.Button

android:id="@+id/bt\_add"

android:layout\_gravity="center"

android:layout\_width="200dp"

android:layout\_height="wrap\_content"

android:layout\_margin="5dp"

android:background="@drawable/my\_button"

android:padding="15dp"

android:gravity="center"

android:text="add"

android:textColor="@color/white"

/>

</LinearLayout>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".activities.ChartsActivity"

android:orientation="vertical">

<com.google.android.material.appbar.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="15dp"

>

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbar\_charts"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="#6C4FF1"

/>

</com.google.android.material.appbar.AppBarLayout>

<androidx.cardview.widget.CardView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="2dp"

android:layout\_marginRight="8dp"

android:layout\_marginLeft="8dp">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="8dp"

android:text="Pie Chart"

android:textColor="@color/black"

/>

<com.github.mikephil.charting.charts.PieChart

android:id="@+id/crt\_pieChart"

android:layout\_width="match\_parent"

android:layout\_height="250dp"/>

</LinearLayout>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:layout\_marginRight="8dp"

android:layout\_marginLeft="8dp">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="8dp"

android:text="Bar Chart"

android:textColor="@color/black"

/>

<com.github.mikephil.charting.charts.BarChart

android:id="@+id/crt\_barChart"

android:layout\_width="match\_parent"

android:layout\_height="250dp"/>

</LinearLayout>

</androidx.cardview.widget.CardView>

</LinearLayout>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".activities.DetailsActivity"

android:orientation="vertical">

<include layout="@layout/app\_bar3"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

android:gravity="center">

<TextView

android:id="@+id/tv\_activityName"

android:layout\_marginTop="10dp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

tools:text="name"

android:textColor="@color/black"

android:textSize="20sp"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

>

<TextView

android:id="@+id/tv\_description"

android:layout\_marginTop="10dp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

tools:text="Description"

android:textColor="@color/black"

android:textSize="15sp"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_margin="4dp"

>

<TextView

android:id="@+id/tv\_startedDate"

android:layout\_marginTop="10dp"

android:layout\_width="wrap\_content"

android:layout\_weight="1"

android:layout\_height="wrap\_content"

tools:text="started date:"

android:textSize="15sp"/>

<TextView

android:id="@+id/tv\_dueDate"

android:layout\_marginTop="10dp"

android:layout\_width="wrap\_content"

android:layout\_weight="1"

android:layout\_height="wrap\_content"

tools:text="Due date:"

android:textSize="15sp"/>

</LinearLayout>

<TextView

android:id="@+id/tv\_progress"

android:layout\_marginTop="10dp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

tools:text="Time spent:"

android:textSize="15sp"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:padding="15dp"

>

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/rv\_recordList"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" />

</LinearLayout>

</LinearLayout>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".activities.IntroActivity"

android:gravity="center\_horizontal"

android:orientation="vertical"

android:background="@drawable/ic\_bg"

>

<ImageView

android:id="@+id/iv\_picstatictic"

android:layout\_width="531dp"

android:layout\_height="444dp"

android:src="@drawable/picstatistic" />

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:text="Tractivity"

android:layout\_marginTop="0dp"

android:textStyle="bold"

android:textSize="40sp"

android:textColor="@android:color/white"

/>

<TextView

android:id="@+id/tv\_welcome"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5dp"

android:gravity="center"

android:text="Welcome"

android:textColor="#363A43"

android:textSize="30sp"

android:textStyle="bold" />

<TextView

android:id="@+id/tv\_description"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:layout\_marginBottom="40dp"

android:gravity="center"

android:text="An app to boost productivity"

android:textColor="@color/white"

android:textSize="20sp"

/>

<LinearLayout

android:id="@+id/l\_buttons"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<android.widget.Button

android:id="@+id/bt\_login"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:background="@drawable/my\_button"

android:text="Login"

android:layout\_weight="1"

android:padding="10dp"

android:textColor="@color/white"

android:layout\_margin="5dp"

/>

<android.widget.Button

android:id="@+id/bt\_signUp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:background="@drawable/my\_button"

android:layout\_weight="1"

android:text="Sign Up"

android:padding="10dp"

android:textColor="@color/white"

android:layout\_margin="5dp"/>

</LinearLayout>

</LinearLayout>

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".activities.LoginActivity"

android:background="@drawable/ic\_bg"

android:fitsSystemWindows="true"

>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Tractivity"

android:layout\_marginTop="40dp"

android:layout\_marginBottom="50dp"

android:layout\_marginLeft="10dp"

android:textStyle="bold"

android:textSize="40sp"

android:textColor="@android:color/white">

</TextView>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="10dp"

android:layout\_marginTop="10dp"

android:layout\_marginBottom="100dp"

android:text="Welcome Back"

android:textColor="#363A43"

android:textSize="30sp"

android:textStyle="bold"

android:layout\_marginStart="10dp" />

<EditText

android:id="@+id/et\_email\_login"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="10dp"

android:background="@drawable/edit\_text\_background"

android:drawablePadding="10dp"

android:ems="10"

android:hint="Email"

android:inputType="textEmailAddress"

android:padding="10dp"

android:textColor="@color/white"

android:textColorHint="@color/white"

android:textSize="15sp"

android:drawableRight="@drawable/ic\_email" />

<EditText

android:id="@+id/et\_password\_login"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="10dp"

android:background="@drawable/edit\_text\_background"

android:drawablePadding="10dp"

android:ems="10"

android:hint="Password"

android:inputType="textPassword"

android:padding="10dp"

android:textColor="@color/white"

android:textColorHint="@color/white"

android:textSize="15sp"

android:drawableRight="@drawable/ic\_password" />

<TextView

android:id="@+id/tv\_forgot\_pass"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="50dp"

android:textSize="15sp"

android:gravity="right"

android:text="@string/forgot\_pass"

android:textColor="@color/white" />

<android.widget.Button

android:id="@+id/bt\_login"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="5dp"

android:background="@drawable/my\_button"

android:padding="15dp"

android:gravity="center"

android:text="Login"

android:textColor="@color/white"

/>

<TextView

android:id="@+id/tv\_signUP"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="@string/sign\_up"

android:textColor="@color/white"

android:gravity="center"

android:layout\_marginTop="20dp">

</TextView>

</LinearLayout>

</ScrollView>

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".activities.PasswordResetActivity"

android:background="@drawable/ic\_bg"

android:fitsSystemWindows="true"

>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Tractivity"

android:layout\_marginTop="40dp"

android:layout\_marginBottom="50dp"

android:layout\_marginLeft="10dp"

android:textStyle="bold"

android:textSize="40sp"

android:textColor="@android:color/white">

</TextView>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="10dp"

android:layout\_marginTop="10dp"

android:layout\_marginBottom="100dp"

android:text="Reset your Password"

android:textColor="#363A43"

android:textSize="30sp"

android:textStyle="bold"

android:layout\_marginStart="10dp" />

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="20dp"

android:textSize="15sp"

android:gravity="center"

android:text="Please enter your email address, a email will be set to reset your password"

android:textColor="@color/white" />

<EditText

android:id="@+id/et\_email\_passReset"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="30dp"

android:background="@drawable/edit\_text\_background"

android:drawablePadding="10dp"

android:ems="10"

android:hint="Email"

android:inputType="textEmailAddress"

android:padding="10dp"

android:textColor="@color/white"

android:textColorHint="@color/white"

android:textSize="15sp"

android:drawableRight="@drawable/ic\_email" />

<android.widget.Button

android:id="@+id/btn\_resetPass"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="5dp"

android:background="@drawable/my\_button"

android:padding="15dp"

android:gravity="center"

android:text="Submit"

android:textColor="@color/white"

/>

</LinearLayout>

</ScrollView>

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".activities.SignUpActivity"

android:background="@drawable/ic\_bg"

android:fitsSystemWindows="true"

>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Tractivity"

android:layout\_marginTop="40dp"

android:layout\_marginBottom="50dp"

android:layout\_marginLeft="10dp"

android:textStyle="bold"

android:textSize="40sp"

android:textColor="@android:color/white">

</TextView>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:layout\_marginBottom="50dp"

android:text="Sign up"

android:textColor="#363A43"

android:textSize="30sp"

android:textStyle="bold"

android:paddingLeft="10dp"

android:paddingStart="10dp" />

<EditText

android:id="@+id/et\_name"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="10dp"

android:background="@drawable/edit\_text\_background"

android:drawablePadding="10dp"

android:ems="10"

android:hint="Username"

android:inputType="textPersonName"

android:padding="10dp"

android:textColor="@color/white"

android:textColorHint="@color/white"

android:textSize="15sp"

android:drawableRight="@drawable/ic\_user" />

<EditText

android:id="@+id/et\_email"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="10dp"

android:background="@drawable/edit\_text\_background"

android:drawablePadding="10dp"

android:ems="10"

android:hint="Email"

android:inputType="textEmailAddress"

android:padding="10dp"

android:textColor="@color/white"

android:textColorHint="@color/white"

android:textSize="15sp"

android:drawableRight="@drawable/ic\_email" />

<EditText

android:id="@+id/et\_password"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="10dp"

android:background="@drawable/edit\_text\_background"

android:drawablePadding="10dp"

android:ems="10"

android:hint="Password"

android:inputType="textPassword"

android:padding="10dp"

android:textColor="@color/white"

android:textColorHint="@color/white"

android:textSize="15sp"

android:drawableRight="@drawable/ic\_password" />

<EditText

android:id="@+id/et\_reTypePassword"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="50dp"

android:background="@drawable/edit\_text\_background"

android:drawablePadding="10dp"

android:ems="10"

android:hint="Re-enter password"

android:inputType="textPassword"

android:padding="10dp"

android:textColor="@color/white"

android:textColorHint="@color/white"

android:textSize="15sp"

android:drawableRight="@drawable/ic\_password" />

<android.widget.Button

android:id="@+id/bt\_SignUp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="5dp"

android:background="@drawable/my\_button"

android:padding="15dp"

android:gravity="center"

android:text="Sign Up"

android:textColor="@color/white"

/>

<TextView

android:id="@+id/tv\_login"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="@string/login"

android:textColor="@color/white"

android:gravity="center"

android:layout\_marginTop="20dp"

>

</TextView>

</LinearLayout>

</ScrollView>

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/ic\_bg"

tools:context=".activities.TractivityMain">

<ImageView

android:id="@+id/bg\_Clock"

android:layout\_width="317dp"

android:layout\_height="294dp"

android:layout\_marginStart="8dp"

android:layout\_marginLeft="8dp"

android:layout\_marginTop="80dp"

android:layout\_marginEnd="8dp"

android:layout\_marginRight="8dp"

android:src="@drawable/clock"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.469"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<ImageView

android:id="@+id/iArrow"

android:layout\_width="285dp"

android:layout\_height="229dp"

android:layout\_marginStart="8dp"

android:layout\_marginLeft="8dp"

android:layout\_marginTop="116dp"

android:layout\_marginEnd="8dp"

android:layout\_marginRight="8dp"

android:foregroundGravity="center"

android:src="@drawable/arrow\_red"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.465"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<android.widget.Button

android:id="@+id/btn\_start"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="68dp"

android:layout\_marginLeft="68dp"

android:layout\_marginTop="49dp"

android:layout\_marginBottom="240dp"

android:background="@drawable/button\_start"

android:text="Start"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/bg\_Clock"

app:layout\_constraintVertical\_bias="1.0"

tools:ignore="MissingConstraints" />

<android.widget.Button

android:id="@+id/btn\_stop"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="49dp"

android:layout\_marginEnd="40dp"

android:layout\_marginRight="40dp"

android:layout\_marginBottom="240dp"

android:background="@drawable/button\_stopwatch"

android:text="Stop"

android:textColor="@color/white"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.756"

app:layout\_constraintStart\_toEndOf="@+id/btn\_start"

app:layout\_constraintTop\_toBottomOf="@+id/bg\_Clock"

app:layout\_constraintVertical\_bias="1.0"

tools:ignore="MissingConstraints" />

<Chronometer

android:id="@+id/c\_chronometer"

android:layout\_width="117dp"

android:layout\_height="58dp"

android:layout\_marginStart="40dp"

android:layout\_marginLeft="40dp"

android:layout\_marginEnd="40dp"

android:layout\_marginRight="40dp"

android:layout\_marginBottom="308dp"

android:gravity="center"

android:text="TextView"

android:textColor="@color/black"

android:textSize="30sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.488"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/bg\_Clock"

app:layout\_constraintVertical\_bias="0.0"

tools:ignore="MissingConstraints" />

<Button

android:id="@+id/btn\_reset"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:background="@drawable/button\_stopwatch"

android:text="Reset"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/c\_chronometer"

app:layout\_constraintVertical\_bias="0.326" />

</androidx.constraintlayout.widget.ConstraintLayout>

<androidx.drawerlayout.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:id="@+id/drawer\_layout"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:openDrawer="start">

<include

layout="@layout/app\_bar"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" />

<com.google.android.material.navigation.NavigationView

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

android:id="@+id/navigation"

android:layout\_gravity="start"

android:fitsSystemWindows="true"

app:headerLayout="@layout/nav\_header"

app:menu="@menu/drawer\_menu" />

</androidx.drawerlayout.widget.DrawerLayout>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".activities.UserProfileActivity"

android:gravity="top"

android:orientation="vertical">

<com.google.android.material.appbar.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="30dp"

>

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbar\_profile"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="#6C4FF1"

/>

</com.google.android.material.appbar.AppBarLayout>

<de.hdodenhof.circleimageview.CircleImageView

android:id="@+id/profile\_image"

android:layout\_width="match\_parent"

android:layout\_height="100dp"

android:src="@drawable/profile\_pic"

app:civ\_border\_color="#FF000000"

app:civ\_border\_width="2dp"

/>

<TextView

android:id="@+id/tv\_username\_profile"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:layout\_gravity="center"

android:textColor="@android:color/black"

android:textSize="20sp"

tools:text="Username"

android:gravity="center"

android:layout\_marginBottom="20dp"

/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_gravity="center"

android:paddingLeft="15dp"

>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

android:gravity="center"

android:text="Name:"

android:layout\_marginRight="10dp"

/>

<TextView

android:id="@+id/tv\_profile\_name"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

tools:text="name"

/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_gravity="center"

android:paddingLeft="15dp"

>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

android:gravity="center"

android:text="Email:"

android:layout\_marginRight="10dp"

/>

<TextView

android:id="@+id/tv\_profile\_email"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

tools:text="example@email.com"

android:text=""

/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_gravity="center"

android:paddingLeft="15dp"

>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

android:gravity="center"

android:text="Productivity Score:"

android:layout\_marginRight="10dp"

/>

<TextView

android:id="@+id/tv\_score"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

tools:text="0"

/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_gravity="center"

android:paddingLeft="15dp"

>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

android:gravity="center"

android:text="Mobile:"

android:layout\_marginRight="10dp"

/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

tools:text="07123145678"

/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_gravity="center"

android:paddingLeft="15dp"

>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

android:gravity="center"

android:text="Address:"

android:layout\_marginRight="10dp"

/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textColor="@android:color/black"

android:textSize="20sp"

tools:text="address"

/>

</LinearLayout>

<androidx.appcompat.widget.AppCompatButton

android:id="@+id/bt\_change\_pass"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:layout\_marginTop="50dp"

android:text="Change password"

/>

</LinearLayout>

<androidx.coordinatorlayout.widget.CoordinatorLayout

xmlns:android="http://schemas.android.com/apk/res/android" android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<com.google.android.material.appbar.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

>

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbar\_trmain"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="#6C4FF1"

/>

</com.google.android.material.appbar.AppBarLayout>

<include layout="@layout/activity\_stopwatch" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>

<RelativeLayout

xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="#6C4FF1">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:padding="16dp">

<ImageView

android:id="@+id/iv\_back"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:src="@drawable/ic\_back"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toEndOf="@+id/iv\_back"

android:layout\_marginStart="16dp"

android:text="Activities"

android:textSize="20sp"

android:layout\_toRightOf="@+id/iv\_back"

android:layout\_marginLeft="15dp" />

<ImageView

android:id="@+id/iv\_add"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:src="@drawable/ic\_add"

android:layout\_alignParentEnd="true"

android:layout\_alignParentRight="true" />

</RelativeLayout>

</RelativeLayout>

<RelativeLayout

xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="#6C4FF1">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:padding="16dp">

<ImageView

android:id="@+id/iv\_back"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:src="@drawable/ic\_back"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toEndOf="@+id/iv\_back"

android:layout\_marginStart="16dp"

android:text="Activity Details"

android:textSize="20sp"

android:layout\_toRightOf="@+id/iv\_back"

android:layout\_marginLeft="15dp" />

<ImageView

android:id="@+id/iv\_edit"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:src="@drawable/ic\_edit"

android:layout\_toLeftOf="@+id/iv\_delete"

android:layout\_marginRight="20dp"

/>

<ImageView

android:id="@+id/iv\_delete"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:src="@drawable/ic\_delete"

android:layout\_alignParentEnd="true"

android:layout\_alignParentRight="true" />

</RelativeLayout>

</RelativeLayout>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Change Password"

android:textColor="@android:color/black"

android:textSize="26sp"

android:layout\_gravity="center"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="8dp"

android:orientation="vertical">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_margin="10dp">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="3"

android:text="Enter new password"

android:textSize="15sp"

android:paddingBottom="10dp"/>

<EditText

android:id="@+id/et\_pass"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:inputType="textPassword"

android:textSize="15sp"

android:paddingTop="10dp"

/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_margin="10dp">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="3"

android:text="Re-enter password"

android:textSize="15sp"

android:paddingBottom="10dp"/>

<EditText

android:id="@+id/et\_rePass"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:inputType="textPassword"

android:textSize="15sp"

/>

</LinearLayout>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:orientation="horizontal">

<androidx.appcompat.widget.AppCompatButton

android:id="@+id/bt\_cancel"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:foreground="?attr/selectableItemBackground"

android:gravity="center"

android:padding="10dp"

android:text="CANCEL"

android:textColor="@android:color/black"/>

<androidx.appcompat.widget.AppCompatButton

android:id="@+id/bt\_submit"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:foreground="?attr/selectableItemBackground"

android:gravity="center"

android:padding="10dp"

android:text="SUBMIT"

android:textColor="@android:color/holo\_red\_dark"/>

</LinearLayout>

</LinearLayout>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

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

android:orientation="vertical">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Edit Activity"

android:textColor="@android:color/black"

android:textSize="26sp"

android:layout\_gravity="center"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="8dp"

android:orientation="vertical">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="3"

android:text="Description:"

android:textSize="15sp"/>

<EditText

android:id="@+id/et\_description"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:inputType="text"

android:textSize="15sp"

/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="3"

android:text="Due Date:"

android:textSize="15sp"/>

<EditText

android:id="@+id/et\_dueDate"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:inputType="date"

android:textSize="15sp"

/>

</LinearLayout>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:orientation="horizontal">

<androidx.appcompat.widget.AppCompatButton

android:id="@+id/bt\_cancel"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:foreground="?attr/selectableItemBackground"

android:gravity="center"

android:padding="10dp"

android:text="CANCEL"

android:textColor="@android:color/black"/>

<androidx.appcompat.widget.AppCompatButton

android:id="@+id/bt\_submit"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:foreground="?attr/selectableItemBackground"

android:gravity="center"

android:padding="10dp"

android:text="SUBMIT"

android:textColor="@android:color/holo\_red\_dark"/>

</LinearLayout>

</LinearLayout>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

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

android:orientation="vertical">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Save Activity"

android:textColor="@android:color/black"

android:textSize="26sp"

android:layout\_gravity="center"/>

<TextView

android:id="@+id/tv\_description"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

tools:text="time spent:"

android:textColor="@android:color/darker\_gray"

android:textSize="22sp" />

<EditText

android:id="@+id/et\_activityName"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Activity name"

android:inputType="text"

android:textSize="20sp"

/>

<androidx.appcompat.widget.AppCompatButton

android:id="@+id/bt\_selectActivity"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:text="Or Select Activity"

/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:orientation="horizontal">

<androidx.appcompat.widget.AppCompatButton

android:id="@+id/bt\_cancel"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:foreground="?attr/selectableItemBackground"

android:gravity="center"

android:padding="10dp"

android:text="CANCEL"

android:textColor="@android:color/black"/>

<androidx.appcompat.widget.AppCompatButton

android:id="@+id/bt\_submit"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:foreground="?attr/selectableItemBackground"

android:gravity="center"

android:padding="10dp"

android:text="SUBMIT"

android:textColor="@android:color/holo\_red\_dark"/>

</LinearLayout>

</LinearLayout>

<androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"

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

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="5dp">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:padding="5dp">

<TextView

android:id="@+id/tv\_activityName"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

tools:text="activity name"

android:textColor="@color/black"

android:textSize="20sp"

android:textStyle="bold" />

<TextView

android:id="@+id/tv\_startedOn"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/tv\_activityName"

android:text="started on:" />

<TextView

android:id="@+id/tv\_date"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/tv\_activityName"

android:layout\_toEndOf="@id/tv\_startedOn"

android:layout\_toRightOf="@id/tv\_startedOn"

android:paddingStart="3dp"

android:paddingLeft="3dp"

tools:text="dd/mm/yyyy" />

</RelativeLayout>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"

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

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="2dp">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:padding="5dp">

<TextView

android:id="@+id/tv\_date"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

tools:text="date"

android:textColor="@color/black"

android:textStyle="bold" />

<TextView

android:id="@+id/tv\_startedOn"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/tv\_date"

android:text="Time spent:" />

<TextView

android:id="@+id/tv\_timeSpent"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/tv\_date"

android:layout\_toEndOf="@id/tv\_startedOn"

android:layout\_toRightOf="@id/tv\_startedOn"

android:paddingStart="3dp"

android:paddingLeft="3dp"

tools:text="1hr" />

</RelativeLayout>

</androidx.cardview.widget.CardView>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:layout\_width="match\_parent"

android:layout\_height="200dp"

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

android:background="@drawable/nav\_header\_background"

android:gravity="bottom"

android:orientation="vertical"

>

<de.hdodenhof.circleimageview.CircleImageView

android:id="@+id/profile\_image"

android:layout\_width="96dp"

android:layout\_height="96dp"

android:layout\_marginStart="15dp"

android:layout\_marginEnd="15dp"

android:layout\_gravity="bottom"

android:layout\_marginLeft="15dp"

android:src="@drawable/profile\_pic"

app:civ\_border\_color="#FF000000"

app:civ\_border\_width="2dp" />

<TextView

android:id="@+id/tv\_username"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="15dp"

android:layout\_marginTop="5dp"

android:layout\_marginBottom="10dp"

android:textColor="@android:color/white"

android:textSize="15sp"

tools:text="Username"

android:layout\_marginLeft="15dp" />

</LinearLayout>

<menu xmlns:android="http://schemas.android.com/apk/res/android">

<item

android:id="@+id/main\_pg"

android:title="Tractivity"

android:icon="@drawable/ic\_stopwatch"/>

<item

android:id="@+id/activities"

android:title="Activities"

android:icon="@drawable/ic\_activities"

/>

<item

android:id="@+id/charts"

android:title="Charts"

android:icon="@drawable/ic\_charts"

/>

<item

android:id="@+id/profile"

android:title="Profile"

android:icon="@drawable/ic\_profile"

/>

<item

android:id="@+id/logout"

android:title="Logout"

android:icon="@drawable/ic\_logout"

/>

</menu>