

LAMPIRAN

Anatomi.kt

package com.example.anatomimanusia

```
import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.view.MotionEvent
import android.widget.Button
import android.widget.ImageView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import coil.decode.GifDecoder
import coil.load
import coil.request.CachePolicy
import android.media.MediaPlayer
import android.widget.ImageButton
import android.os.Handler
import android.text.Spanned
import androidx.core.text.HtmlCompat
```

class Anatomi : AppCompatActivity() {

```
    private lateinit var mediaPlayer: MediaPlayer
    private var isPaused: Boolean = false
    private val handler = Handler()
```

```
    private val audioFiles = listOf(
        R.raw.s_animasi
    )
    private var currentIndex = 0
```

```
    private var isPlaying = false
```

```
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_anatomi)
```

```
        val imageBody =
            findViewById<ImageView>(R.id.imageBody)
        val playButton: ImageButton =
            findViewById(R.id.playButton)
        val pauseButton: ImageButton =
            findViewById(R.id.pauseButton)
        val stopButton: ImageButton =
            findViewById(R.id.stopButton)
        val backButton: ImageButton =
            findViewById(R.id.backButton1)
```

```
        // Load GIF using Coil
        imageBody.load(R.raw.animasi) {
            decoderFactory { result, options, _ ->
                GifDecoder(result.source, options) }
            diskCachePolicy(CachePolicy.DISABLED)
        }
```

```
        imageBody.setOnTouchListener { v, event ->
            if (event.action ==
                MotionEvent.ACTION_DOWN) {
                val x = event.x
                val y = event.y
                val imageWidth = v.width
                val imageHeight = v.height
```

```
                val bodyPart = when {
```

Materi.kt

package com.example.anatomimanusia

```
import android.content.Intent
import android.os.Bundle
import android.util.Log
//import android.view.MotionEvent
//import android.view.ScaleGestureDetector
import android.widget.Button
//import android.widget.ImageView
//import android.widget.TextView
//import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import androidx.cardview.widget.CardView
//import kotlin.math.max
//import kotlin.math.min
```

class Materi : AppCompatActivity() {

```
    override fun onCreate(savedInstanceState:
        Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_materi)
```

```
        // Mendapatkan referensi ke setiap CardView
        val cardView1: CardView =
            findViewById(R.id.cardView1)
        val cardView2: CardView =
            findViewById(R.id.cardView2)
        val cardView3: CardView =
            findViewById(R.id.cardView3)
        val cardView4: CardView =
            findViewById(R.id.cardView4)
        val cardView5: CardView =
            findViewById(R.id.cardView5)
        val cardView6: CardView =
            findViewById(R.id.cardView6)
        val cardView7: CardView =
            findViewById(R.id.cardView7)
        val cardView8: CardView =
            findViewById(R.id.cardView8)
        val cardView9: CardView =
            findViewById(R.id.cardView9)
        val cardView10: CardView =
            findViewById(R.id.cardView10)
        val cardView11: CardView =
            findViewById(R.id.cardView11)
        val cardView12: CardView =
            findViewById(R.id.cardView12)
```

```
        // Menambahkan OnClickListener untuk setiap
        CardView
```

```
        cardView1.setOnClickListener {
            val audioFiles = listOf(
                R.raw.muskuloskeletal,
                R.raw.kerangka,
                R.raw.tulang,
                R.raw.tulang_rawan,
                R.raw.ligamen,
                R.raw.sendi,
                R.raw.otot,
                R.raw.jenis_otot,
                R.raw.bentuk_otot,
                R.raw.cara_kerja_otot
```

```

        isClickWithinHeadArea(x, y, imageWidth,
imageHeight) -> "Kepala"
        isClickWithinBodyArea(x, y, imageWidth,
imageHeight) -> "Badan"
        isClickWithinHandArea(x, y, imageWidth,
imageHeight) -> "Tangan"
        isClickWithinLegArea(x, y, imageWidth,
imageHeight) -> "Kaki"
        else -> "Bagian lain"
    }

    when (bodyPart) {
        "Kepala" -> {
            val audioFiles = arrayListOf(
                R.raw.tulang_parietal,
                R.raw.frontal,
                R.raw.temporal,
                R.raw.oksipital,
                R.raw.zygoma,
                R.raw.rahang_bawah,
                R.raw.tulang_hidung,
                R.raw.hyoid,
                R.raw.gigi
            )

            val textIsi = HtmlCompat.fromHtml(
                getString(R.string.Tulang_parietal) +
                getString(R.string.Tulang_frontal)
            )

            +

            getString(R.string.Tulang_temporal) +

            getString(R.string.Tulang_oksipital) +

            getString(R.string.Tulang_zigomatik) +

            getString(R.string.Tulang_rahang_bawah) +
                getString(R.string.Tulang_hidung)
            +

                getString(R.string.Tulang_hyoid) +
                getString(R.string.Gigi),

            HtmlCompat.FROM_HTML_MODE_LEGACY
        )

        val glbFile = "zi_kepala"

        navigateToActivity(View3DActivity::class.java,
audioFiles, textIsi, glbFile)
    }
    "Tangan" -> {

        val audioFiles = arrayListOf(
            R.raw.selangka,
            R.raw.belikat,
            R.raw.lengan_atas,
            R.raw.sendi_siku,
            R.raw.pengumpil,
            R.raw.hasta,
            R.raw.telapak_tangan,
            R.raw.jari_tangan
        )

        val textIsi = HtmlCompat.fromHtml(

```

```

    )

    tampilkanPopup(getString(R.string.muskuloskeletal_t
ext), R.drawable.muskuloskeletal, audioFiles)
    }

    cardView2.setOnClickListener {
        val audioFiles = arrayListOf(
            R.raw.kardiovaskular,
            R.raw.jantung,
            R.raw.awal_kehidupan,
            R.raw.sirkulasi,
            R.raw.panggilan_denyut,
            R.raw.denyut_jantung,
            R.raw.dalam_jantung,
            R.raw.darah,
            R.raw.asal_usul_darah,
            R.raw.pembentuk_darah,
            R.raw.pembuluh_darah,
            R.raw.tekanan_darah,
            R.raw.aliran_darah,
            R.raw.gol_darah
        )

        tampilkanPopup(getString(R.string.kardiovaskular_te
xt), R.drawable.kardiovaskular, audioFiles)
    }

    cardView3.setOnClickListener {
        val audioFiles = arrayListOf(
            R.raw.nafas_1,
            R.raw.nafas_2,
            R.raw.nafas_3,
            R.raw.nafas_4,
            R.raw.nafas_5
        )

        tampilkanPopup(getString(R.string.pernafasan_text),
R.drawable.pernafasan, audioFiles)
    }

    cardView4.setOnClickListener {
        val audioFiles = arrayListOf(
            R.raw.cerna_1,
            R.raw.cerna_2,
            R.raw.cerna_3,
            R.raw.cerna_4,
            R.raw.cerna_5,
            R.raw.cerna_6,
            R.raw.cerna_7
        )

        tampilkanPopup(getString(R.string.pencernaan_text),
R.drawable.pencernaan, audioFiles)
    }

    cardView5.setOnClickListener {
        val audioFiles = arrayListOf(
            R.raw.urin1,
            R.raw.urin2,
            R.raw.urin3,
            R.raw.urin4
        )
    }

```

```

        getString(R.string.Tulang_selangka) +
        getString(R.string.Tulang_belikat)
+
        getString(R.string.Tulang_lengan_atas) +
        getString(R.string.Sendi_siku) +

        getString(R.string.Tulang_pengumpil) +
        getString(R.string.Tulang_hasta) +

        getString(R.string.Tulang_telapak_tangan) +

        getString(R.string.Tulang_jari_tangan),
        HtmlCompat.FROM_HTML_MODE_LEGACY
    )

    val glbFile = "zi_tangan"

    navigateToActivity(View3DActivity::class.java,
        audioFiles, textIsi, glbFile)
    }
    "Badan" -> {

        val audioFiles = arrayListOf(
            R.raw.rusuk_pertama,
            R.raw.rusuk_sejati,
            R.raw.rusuk_palsu,
            R.raw.tulang_kepala_dada,
            R.raw.badan_tulang_dada,
            R.raw.taju_pedang,
            R.raw.rusuk_melayang
        )

        val textIsi = HtmlCompat.fromHtml(

            getString(R.string.Tulang_rusuk_pertama) +

            getString(R.string.Tulang_rusuk_sejati) +

            getString(R.string.Tulang_rusuk_palsu) +
            getString(R.string.Tulang_dada) +
            getString(R.string.Tulang_badan) +
            getString(R.string.Tulang_taju) +
            getString(R.string.Tulang_rusuk),

            HtmlCompat.FROM_HTML_MODE_LEGACY
        )

        val glbFile = "zi_badan"

        navigateToActivity(View3DActivity::class.java,
            audioFiles, textIsi, glbFile)
        }
        "Kaki" -> {

            val audioFiles = arrayListOf(
                R.raw.pinggul,
                R.raw.femur,
                R.raw.patella,
                R.raw.fibula,
                R.raw.tibia,

```

```

            tampilkanPopup(getString(R.string.urinaria_text),
                R.drawable.urinaria, audioFiles)
            }

            cardView6.setOnClickListener {
                val audioFiles = arrayListOf(
                    R.raw.guren1,
                    R.raw.guren2,
                    R.raw.guren3,
                    R.raw.guren4
                )

                tampilkanPopup(getString(R.string.integumen_text),
                    R.drawable.integumen, audioFiles)
            }

            cardView7.setOnClickListener {
                val audioFiles = arrayListOf(
                    R.raw.indera1,
                    R.raw.indera2,
                    R.raw.indera3,
                    R.raw.indera4,
                    R.raw.indera5,
                    R.raw.indera6,
                    R.raw.indera7,
                    R.raw.indera8
                )

                tampilkanPopup(getString(R.string.indra_text),
                    R.drawable.indra, audioFiles)
            }

            cardView8.setOnClickListener {
                val audioFiles = arrayListOf(
                    R.raw.repro1,
                    R.raw.repro2,
                    R.raw.repro3,
                    R.raw.repro4
                )

                tampilkanPopup(getString(R.string.sr_text),
                    R.drawable.sr, audioFiles)
            }

            cardView9.setOnClickListener {
                val audioFiles = arrayListOf(
                    R.raw.endo1,
                    R.raw.endo2,
                    R.raw.endo3
                )

                tampilkanPopup(getString(R.string.endokrin_text),
                    R.drawable.endokrin, audioFiles)
            }

            cardView10.setOnClickListener {
                val audioFiles = arrayListOf(
                    R.raw.imun1,
                    R.raw.imun2,
                    R.raw.imun3,
                    R.raw.imun4,
                    R.raw.imun5
                )

                tampilkanPopup(getString(R.string.si_text),
                    R.drawable.si, audioFiles)
            }

```

```

        R.raw.metatarsal,
        R.raw.calcaneus,
        R.raw.phalanges
    )

    val textIsi = HtmlCompat.fromHtml(
        getString(R.string.tulang_panggul) +
        getString(R.string.tulang_paha) +
        getString(R.string.patella) +
        getString(R.string.fibula) +
        getString(R.string.tibia) +

        getString(R.string.tulang_metatarsal) +
        getString(R.string.calcaneus) +
        getString(R.string.phalanges),

        HtmlCompat.FROM_HTML_MODE_LEGACY
    )

    val glbFile = "zi_kaki"

    navigateToActivity(View3DActivity::class.java,
        audioFiles, textIsi, glbFile)
    }
    else -> Toast.makeText(this, "Kurang tepat,
    ulangi kembali", Toast.LENGTH_SHORT).show()
    }

    v.performClick() // Menambahkan
    performClick() untuk tujuan aksesibilitas
    }
    true
    }

    // Menangani klik tombol kembali
    backButton.setOnClickListener {
    //      try {
    //          // Pastikan media player berhenti dan
    //          dilepaskan sebelum kembali ke MainActivity
    //          if (this::mediaPlayer.isInitialized &&
    //              (mediaPlayer.isPlaying || isPaused)) {
    //              mediaPlayer.stop()
    //              mediaPlayer.release()
    //              isPaused = false
    //          }
    //          val intent = Intent(this,
    //              MainActivity::class.java).apply {
    //              flags =
    //              Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
    //          }
    //          startActivity(intent)
    //      } catch (e: Exception) {
    //          Log.e("Anatomi-Main", "Error :
    //              ${e.message}")
    //      }
    //      try {
    //          // Ensure the media player stops and is released
    //          before returning to MainActivity
    //          if (this::mediaPlayer.isInitialized) {
    //              try {
    //                  if (mediaPlayer.isPlaying || isPaused) {
    //                      mediaPlayer.stop()
    //                  }
    //              } catch (e: IllegalStateException) {

```

```

//          e.printStackTrace() // Log the exception
if mediaPlayer is in an illegal state
//          } finally {
//          mediaPlayer.release() // Always release
the media player
//          isPaused = false
//          }
//          }

// Pastikan media player berhenti dan dilepaskan
sebelum kembali ke MainActivity
if (this::mediaPlayer.isInitialized) {
    if (mediaPlayer.isPlaying || isPaused) {
        mediaPlayer.stop()
    }
    mediaPlayer.release()
    isPaused = false
}
val intent = Intent(this,
MainActivity::class.java).apply {
    flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
}
startActivity(intent)

//
//          val intent = Intent(this,
MainActivity::class.java).apply {
//          flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
//          }
//          startActivity(intent)
} catch (e: Exception) {
    Log.e("Anatomi-Main", "Error: ${e.message}")
}
}

imageBody.setOnClickListener {
    // You can leave this empty or add additional
behavior here if needed
}

// Play Button Listener
playButton.setOnClickListener {
//          if (isPaused) {
//          mediaPlayer.start()
//          isPaused = false
//          } else {
//          playNextAudio()
//          }
try {
    if (this::mediaPlayer.isInitialized &&
!mediaPlayer.isPlaying) {
        mediaPlayer.start()
        isPaused = false
    }
} catch (e: IllegalStateException) {
    Log.e("Anatomi-Play", "Error: ${e.message}")
    e.printStackTrace()
}
}

// Pause Button Listener
pauseButton.setOnClickListener {
//          if (this::mediaPlayer.isInitialized &&
mediaPlayer.isPlaying) {

```

```

import android.graphics.text.LineBreaker
import android.media.MediaPlayer
import android.os.Bundle
import android.os.Handler
import android.text.SpannableString
import android.text.Spanned
import android.util.Log
//import android.text.Html
import android.view.Choreographer
import android.view.ScaleGestureDetector
import android.view.SurfaceView
import android.view.ViewTreeObserver
//import android.widget.Button
import android.widget.ImageButton
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.core.text.HtmlCompat
import com.google.android.filament.Engine
import com.google.android.filament.EntityManager
import com.google.android.filament.LightManager
import com.google.android.filament.Skybox
import
com.google.android.filament.utils.ModelViewer
import com.google.android.filament.utils.Utils
import java.nio.ByteBuffer

```

```

class View3DActivity : AppCompatActivity() {

    private lateinit var mediaPlayer: MediaPlayer
    private var isPaused: Boolean = false
    private val handler = Handler()

    companion object {
        init { Utils.init() }
    }

    private var currentIndex = 0

    private lateinit var surfaceView: SurfaceView
    private lateinit var choreographer: Choreographer
    private lateinit var modelViewer: ModelViewer
    private lateinit var scaleGestureDetector:
ScaleGestureDetector
    private var light: Int = 0
    private var engine: Engine? = null

    private lateinit var playButton: ImageButton
    private lateinit var pauseButton: ImageButton
    private lateinit var stopButton: ImageButton
    private lateinit var backButton: ImageButton

    private lateinit var audioFiles: ArrayList<Int>
    private lateinit var textIsi: Spanned
    private lateinit var glbFile: String

    override fun onCreate(savedInstanceState:
Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_objectglb)

        audioFiles = arrayListOf()
        textIsi = SpannableString("")
        glbFile = ""
    }
}

```

```

// mediaPlayer.pause()
// isPaused = true
// }
if (this::mediaPlayer.isInitialized &&
mediaPlayer.isPlaying) {
    try {
        mediaPlayer.pause()
        isPaused = true
    } catch (e: IllegalStateException) {
        e.printStackTrace()
    }
}

// Stop Button Listener
stopButton.setOnClickListener {
// if (this::mediaPlayer.isInitialized &&
(mediaPlayer.isPlaying || isPaused)) {
// mediaPlayer.stop()
// mediaPlayer.release()
// isPaused = false
// }
if (this::mediaPlayer.isInitialized) {
    try {
        if (mediaPlayer.isPlaying || isPaused) {
            mediaPlayer.stop()
            mediaPlayer.reset() // Reset the player to
avoid using a released player
            isPaused = false
        }
    } catch (e: IllegalStateException) {
        // Handle the exception if mediaPlayer is in
an illegal state
        e.printStackTrace()
    } finally {
        mediaPlayer.release() // Release after
stopping
    }
}

// Start playing when activity opens
playNextAudio()
}

private fun navigateToActivity(
    activityClass: Class<*>,
    audioFiles: ArrayList<Int>,
    textIsi: Spanned,
    glbFile: String
) {
    try {
        val intent = Intent(this, activityClass).apply {
putIntegerArrayListExtra("EXTRA_AUDIO_FILES",
audioFiles)
putExtra("EXTRA_TEXT", textIsi)
putExtra("EXTRA_GLB_FILE", glbFile)
flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
        }
        startActivity(intent)
    } catch (e: Exception) {
        Log.e("Anatomi", "Error navigating to activity:
${e.message}")
    }
}

```

```

// Mengambil data dari Intent
audioFiles =
intent.getIntegerArrayListExtra("EXTRA_AUDIO_F
ILES") ?: arrayListOf()

// Ambil textIsi yang sudah dalam format
Spanned dari Intent
val receivedText: Spanned? =
intent.getCharSequenceExtra("EXTRA_TEXT") as?
Spanned

// Inisialisasi textIsi hanya jika receivedText
tidak null
if (receivedText != null) {
    textIsi = receivedText
} else {
    // Jika null, berikan nilai default atau
tampilkan pesan error
    textIsi = HtmlCompat.fromHtml("File text
tidak ke Upload",
HtmlCompat.FROM_HTML_MODE_LEGACY)
}

glbFile =
intent.getStringExtra("EXTRA_GLB_FILE") ?: ""

playButton = findViewById(R.id.playButton)
pauseButton = findViewById(R.id.pauseButton)
stopButton = findViewById(R.id.stopButton)
backButton = findViewById(R.id.backButton1)

surfaceView = findViewById(R.id.surfaceView)
choreographer = Choreographer.getInstance()
modelViewer = ModelViewer(surfaceView)
surfaceView.setOnTouchListener(modelViewer)

setupLighting()
engine = modelViewer.engine
//loadGlb object
loadGlb(glbFile)
setupSkybox()

// Atur tinggi SurfaceView secara dinamis

surfaceView.viewTreeObserver.addOnGlobalLayout
Listener(object :
ViewTreeObserver.OnGlobalLayoutListener {
    override fun onGlobalLayout() {

surfaceView.viewTreeObserver.removeOnGlobalLay
outListener(this)
        val aspectRatio = 1.0f // Sesuaikan ini
dengan rasio aspek objek GLB Anda
        val newHeight = (surfaceView.width /
aspectRatio).toInt() + 16 // Tambahkan padding
(16dp di atas dan 16dp di bawah)
        surfaceView.layoutParams.height =
newHeight
        surfaceView.requestLayout()
    }
})

scaleGestureDetector =
ScaleGestureDetector(this, object :

```

```

    }
}

private fun playNextAudio() {
    if (this::mediaPlayer.isInitialized &&
(mediaPlayer.isPlaying || isPaused)) {
        mediaPlayer.stop()
        mediaPlayer.release()
    }

    if (currentIndex < audioFiles.size) {
        mediaPlayer = MediaPlayer.create(this,
audioFiles[currentIndex])
        mediaPlayer.setOnCompleteListener {
            currentIndex++
            handler.postDelayed({
                playNextAudio()
            }, 2000)
        }
        mediaPlayer.start()
        isPlaying = true
    } else {
        currentIndex = 0
    }
}

override fun onPause() {
    super.onPause()
    if (this::mediaPlayer.isInitialized &&
mediaPlayer.isPlayingOrPaused()) {
        mediaPlayer.pause()
        isPaused = true
    }
}

// Extension function to safely check if MediaPlayer is
playing or paused
private fun MediaPlayer.isPlayingOrPaused(): Boolean
{
    return try {
        this.isPlaying || this.currentPosition > 0
    } catch (e: IllegalStateException) {
        false
    }
}

override fun onResume() {
    super.onResume()
    if (this::mediaPlayer.isInitialized && isPaused) {
        mediaPlayer.start()
        isPaused = false
    }
}

override fun onDestroy() {
    super.onDestroy()
    if (this::mediaPlayer.isInitialized &&
mediaPlayer.isPlaying) {
        mediaPlayer.stop()
        mediaPlayer.release()
    }
}

private fun isClickWithinHeadArea(x: Float, y: Float,
imageWidth: Int, imageHeight: Int): Boolean {

```

```

ScaleGestureDetector.SimpleOnScaleGestureListener
() {
    override fun onScale(detector:
ScaleGestureDetector): Boolean {
        val scale = detector.scaleFactor
        modelViewer.transformToUnitCube()
        modelViewer.asset?.apply {
            val transformManager =
modelViewer.engine.transformManager
            val instance =
transformManager.getInstance(root)
            val currentTransform = FloatArray(16)

transformManager.getTransform(instance,
currentTransform)
            val newTransform = FloatArray(16)

android.opengl.Matrix.scaleM(newTransform, 0,
currentTransform, 0, scale, scale, scale)

transformManager.setTransform(instance,
newTransform)
        }
        return true
    }
})

surfaceView.setOnTouchListener { v, event ->
    scaleGestureDetector.onTouchEvent(event)
    modelViewer.onTouchEvent(event)
    true
}

//val backButton =
findViewById<Button>(R.id.backButton)
backButton.setOnClickListener {
    try {
        val intent = Intent(this,
Anatomy::class.java).apply {
            flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
        }
        startActivity(intent)
    } catch (e: Exception) {
        Log.e("KepalaAct", "Error :
${e.message}")
    }
}

// Log untuk memastikan bahwa textIsi tidak
null atau kosong
Log.d("View3DActivity", "Text from Intent:
$textIsi")

val textView =
findViewById<TextView>(R.id.textView)
textView.text = textIsi
textView.justificationMode =
LineBreaker.JUSTIFICATION_MODE_INTER_WO
RD

// Play Button Listener
playButton.setOnClickListener {
    if (isPaused) {
        mediaPlayer.start()
    }
}

```



```

        val headMinX = 0.4f * imageWidth
        val headMaxX = 0.6f * imageWidth
        val headMinY = 0.0f * imageHeight
        val headMaxY = 0.2f * imageHeight
        return x in headMinX..headMaxX && y in
headMinY..headMaxY
    }

    private fun isClickWithinBodyArea(x: Float, y: Float,
imageWidth: Int, imageHeight: Int): Boolean {
        val bodyMinX = 0.3f * imageWidth
        val bodyMaxX = 0.7f * imageWidth
        val bodyMinY = 0.2f * imageHeight
        val bodyMaxY = 0.6f * imageHeight
        return x in bodyMinX..bodyMaxX && y in
bodyMinY..bodyMaxY
    }

    private fun isClickWithinHandArea(x: Float, y: Float,
imageWidth: Int, imageHeight: Int): Boolean {
        val leftHandMinX = 0.0f * imageWidth
        val leftHandMaxX = 0.3f * imageWidth
        val rightHandMinX = 0.7f * imageWidth
        val rightHandMaxX = 1.0f * imageWidth
        val handMinY = 0.2f * imageHeight
        val handMaxY = 0.6f * imageHeight
        return (x in leftHandMinX..leftHandMaxX || x in
rightHandMinX..rightHandMaxX) && y in
handMinY..handMaxY
    }

    private fun isClickWithinLegArea(x: Float, y: Float,
imageWidth: Int, imageHeight: Int): Boolean {
        val legMinX = 0.3f * imageWidth
        val legMaxX = 0.7f * imageWidth
        val legMinY = 0.6f * imageHeight
        val legMaxY = 1.0f * imageHeight
        return x in legMinX..legMaxX && y in
legMinY..legMaxY
    }
}

```

Kepala.kt

package com.example.anatomimanusia

```

import android.content.Intent
import android.os.Bundle
import android.util.Log
//import android.text.Html
import android.view.Choreographer
import android.view.ScaleGestureDetector
import android.view.SurfaceView
import android.view.ViewTreeObserver
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.core.text.HtmlCompat

```

```

import com.google.android.filament.Engine
import com.google.android.filament.EntityManager
import com.google.android.filament.LightManager
import com.google.android.filament.Skybox
import com.google.android.filament.utils.ModelViewer
import com.google.android.filament.utils.Utils
import java.nio.ByteBuffer

```

```

        isPaused = false
    } else {
        playNextAudio()
    }
}

// Pause Button Listener
pauseButton.setOnClickListener {
    if (mediaPlayer.isPlaying) {
        mediaPlayer.pause()
        isPaused = true
    }
}

// Stop Button Listener
stopButton.setOnClickListener {
    if (mediaPlayer.isPlaying || isPaused) {
        mediaPlayer.stop()
        mediaPlayer.release()
        isPaused = false
    }
}

// Start playing when activity opens
playNextAudio()
}

private val frameCallback = object :
Choreographer.FrameCallback {
    override fun doFrame(currentTime: Long) {
        choreographer.postFrameCallback(this)
        modelViewer.render(currentTime)
    }
}

override fun onDestroy() {
    super.onDestroy()
    engine?.let {
        it.flushAndWait() // Pastikan memanggil
flushAndWait sebelum shutdown
        modelViewer.destroyModel()
        it.destroy()
    }
}

// Lepaskan media player saat activity
dihancurkan
if (this::mediaPlayer.isInitialized) {
    mediaPlayer.release()
}
}

private fun loadGlb(name: String) {
    val buffer = readAsset("models/${name}.glb")
    modelViewer.loadModelGlb(buffer)
    modelViewer.transformToUnitCube()
    modelViewer.scene.skybox =
Skybox.Builder().build(modelViewer.engine)
}

private fun readAsset(assetName: String):
ByteBuffer {
    val input = assets.open(assetName)
    val bytes = ByteArray(input.available())
    input.read(bytes)
    input.close()
}

```

```

class KepalaActivity : AppCompatActivity() {

    companion object {
        init { Utils.init() }
    }

    private lateinit var surfaceView: SurfaceView
    private lateinit var choreographer: Choreographer
    private lateinit var modelViewer: ModelViewer
    private lateinit var scaleGestureDetector:
ScaleGestureDetector
    private var light: Int = 0
    private var engine: Engine? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_objectglb)

        surfaceView = findViewById(R.id.surfaceView)
        //surfaceView = SurfaceView(this).apply {
setContentView(this) }
        choreographer = Choreographer.getInstance()
        modelViewer = ModelViewer(surfaceView)
        surfaceView.setOnTouchListener(modelViewer)

        setupLighting()
        engine = modelViewer.engine
        //loadGlb("kepala")
        loadGlb("kepala2")
        //modelViewer.scene.skybox =
Skybox.Builder().build(modelViewer.engine)
        setupSkybox()

        // Atur tinggi SurfaceView secara dinamis

        surfaceView.viewTreeObserver.addOnGlobalLayoutListe
ner(object : ViewTreeObserver.OnGlobalLayoutListener
{
    override fun onGlobalLayout() {

        surfaceView.viewTreeObserver.removeOnGlobalLayoutL
istener(this)
        val aspectRatio = 1.0f // Sesuaikan ini dengan
rasio aspek objek GLB Anda
        val newHeight = (surfaceView.width /
aspectRatio).toInt() + 16 // Tambahkan padding (16dp di
atas dan 16dp di bawah)
        surfaceView.layoutParams.height = newHeight
        surfaceView.requestLayout()
    }
})

        scaleGestureDetector = ScaleGestureDetector(this,
object :
ScaleGestureDetector.SimpleOnScaleGestureListener() {
    override fun onScale(detector:
ScaleGestureDetector): Boolean {
        val scale = detector.scaleFactor
        modelViewer.transformToUnitCube()
        modelViewer.asset?.apply {
            val transformManager =
modelViewer.engine.transformManager

                return ByteBuffer.wrap(bytes)
            }

        private fun setupLighting() {
            light = EntityManager.get().create()

LightManager.Builder(LightManager.Type.DIRECTI
ONAL)
                .color(1.0f, 1.0f, 1.0f)
                .intensity(100_000.0f)
                .direction(0.0f, -1.0f, -1.0f)
                .castShadows(true)
                .build(modelViewer.engine, light)
            modelViewer.scene.addEntity(light)
        }

        // For Skybox with white background
        private fun setupSkybox() {
            modelViewer.scene.skybox =
Skybox.Builder().color(1.0f, 1.0f, 1.0f,
1.0f).build(modelViewer.engine)
        }

        override fun onResume() {
            super.onResume()

            choreographer.postFrameCallback(frameCallback)

            // Initialize media player on resume
            // if (!this::mediaPlayer.isInitialized) {
            //     playNextAudio()
            // } else if (mediaPlayer.isSafeToUse() &&
!mediaPlayer.isPlaying) {
            //     mediaPlayer.start()
            // }
            if (!this::mediaPlayer.isInitialized) {
                playNextAudio()
            } else {
                try {
                    if (!mediaPlayer.isPlaying) {
                        mediaPlayer.start()
                    }
                } catch (e: IllegalStateException) {
                    e.printStackTrace()
                }
            }
        }

        override fun onPause() {
            super.onPause()

            choreographer.removeFrameCallback(frameCallback
)

            // // Pause and release media player on pause
            // if (this::mediaPlayer.isInitialized &&
mediaPlayer.isSafeToUse()) {
            //     if (mediaPlayer.isPlaying) {
            //         mediaPlayer.pause()
            //     }
            //     mediaPlayer.release()
            // }
            if (this::mediaPlayer.isInitialized) {

```

```

        val instance =
transformManager.getInstance(root)
        val currentTransform = FloatArray(16)
        transformManager.getTransform(instance,
currentTransform)
        val newTransform = FloatArray(16)

android.opengl.Matrix.scaleM(newTransform, 0,
currentTransform, 0, scale, scale, scale)
        transformManager.setTransform(instance,
newTransform)
    }
    return true
}
})

surfaceView.setOnTouchListener { v, event ->
    scaleGestureDetector.onTouchEvent(event)
    modelViewer.onTouchEvent(event)
    true
}

val backButton =
findViewById<Button>(R.id.backButton)
backButton.setOnClickListener {
    // val intent = Intent(this, Anatomi::class.java)
    ///
intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP
or Intent.FLAG_ACTIVITY_NEW_TASK)
    // intent.flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
    // startActivity(intent)
    // finish() // Kembali ke activity Anatomi
    try {
        val intent = Intent(this,
Anatomi::class.java).apply {
            flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
        }
        startActivity(intent)
    } catch (e: Exception) {
        Log.e("KepalaAct", "Error : ${e.message}")
    }
}

val textView =
findViewById<TextView>(R.id.textView)
textView.text = HtmlCompat.fromHtml(
    getString(R.string.kepala_desk) +
        getString(R.string.kepala_tengkorak) +
        getString(R.string.kepala_otak) +
        getString(R.string.kepala_mata) +
        getString(R.string.kepala_telinga) +
        getString(R.string.kepala_hidung) +
        getString(R.string.kepala_mulut),
    HtmlCompat.FROM_HTML_MODE_LEGACY
)
}

private val frameCallback = object :
Choreographer.FrameCallback {
    override fun doFrame(currentTime: Long) {
        choreographer.postFrameCallback(this)
        modelViewer.render(currentTime)
    }
}

```

```

    try {
        if (mediaPlayer.isPlaying) {
            mediaPlayer.pause()
        }
        mediaPlayer.reset() // Reset to avoid
accessing an invalid player
    } catch (e: IllegalStateException) {
        e.printStackTrace()
    } finally {
        mediaPlayer.release()
    }
}
}

// Extension function to safely check if
MediaPlayer is in a state where it can be used
private fun MediaPlayer.isSafeToUse(): Boolean {
    return try {
        this.isPlaying || this.currentPosition > 0
    } catch (e: IllegalStateException) {
        false
    }
}

private fun playNextAudio() {
    if (currentIndex < audioFiles.size) {
        mediaPlayer = MediaPlayer.create(this,
audioFiles[currentIndex])
        mediaPlayer.setOnCompletionListener {
            handler.postDelayed({
                currentIndex++
                playNextAudio()
            }, 2000) // 2-second delay
        }
        mediaPlayer.start()
    } else {
        Toast.makeText(this, "All audio files have
been played", Toast.LENGTH_SHORT).show()
        currentIndex = 0
    }
}
}

Tubuh.kt
package com.example.anatomimanusia

import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.view.Choreographer
import android.view.ScaleGestureDetector
import android.view.SurfaceView
import android.view.ViewTreeObserver
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.core.text.HtmlCompat

import com.google.android.filament.Engine
import com.google.android.filament.EntityManager
import com.google.android.filament.LightManager
import com.google.android.filament.Skybox
import
com.google.android.filament.utils.ModelViewer

```

```

    }

    override fun onResume() {
        super.onResume()
        choreographer.postFrameCallback(frameCallback)
    }

    override fun onPause() {
        super.onPause()
    }

    choreographer.removeFrameCallback(frameCallback)
    }

    override fun onDestroy() {
        super.onDestroy()
        engine?.let {
            it.flushAndWait() // Pastikan mematikan
flushAndWait sebelum shutdown
            modelViewer.destroyModel()
            it.destroy()
        }
    }

    private fun loadGlb(name: String) {
        val buffer = readAsset("models/${name}.glb")
        modelViewer.loadModelGlb(buffer)
        modelViewer.transformToUnitCube()
        modelViewer.scene.skybox =
Skybox.Builder().build(modelViewer.engine)
    }

    private fun readAsset(assetName: String): ByteBuffer {
        val input = assets.open(assetName)
        val bytes = ByteArray(input.available())
        input.read(bytes)
        input.close()
        return ByteBuffer.wrap(bytes)
    }

    private fun setupLighting() {
        light = EntityManager.get().create()

        LightManager.Builder(LightManager.Type.DIRECTION
AL)
            .color(1.0f, 1.0f, 1.0f)
            .intensity(100_000.0f)
            .direction(0.0f, -1.0f, -1.0f)
            .castShadows(true)
            .build(modelViewer.engine, light)
        modelViewer.scene.addEntity(light)
    }

    // For Skybox with white background
    private fun setupSkybox() {
        modelViewer.scene.skybox =
Skybox.Builder().color(1.0f, 1.0f, 1.0f,
1.0f).build(modelViewer.engine)
    }
}

Tangan.kt
package com.example.anatomimanusia

import android.content.Intent

import com.google.android.filament.utils.Utils
import java.nio.ByteBuffer

class TubuhActivity : AppCompatActivity() {

    companion object {
        init { Utils.init() }
    }

    private lateinit var surfaceView: SurfaceView
    private lateinit var choreographer: Choreographer
    private lateinit var modelViewer: ModelViewer
    private lateinit var scaleGestureDetector:
ScaleGestureDetector
    private var light: Int = 0
    private var engine: Engine? = null

    override fun onCreate(savedInstanceState:
Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_objectglb)

        surfaceView = findViewById(R.id.surfaceView)
        //surfaceView = SurfaceView(this).apply {
setContentView(this) }
        choreographer = Choreographer.getInstance()
        modelViewer = ModelViewer(surfaceView)
        surfaceView.setOnTouchListener(modelViewer)

        setupLighting()
        engine = modelViewer.engine
        loadGlb("badan")
        //modelViewer.scene.skybox =
Skybox.Builder().build(modelViewer.engine)
        setupSkybox()

        // Atur tinggi SurfaceView secara dinamis

        surfaceView.viewTreeObserver.addOnGlobalLayout
Listener(object :
ViewTreeObserver.OnGlobalLayoutListener {
            override fun onGlobalLayout() {

                surfaceView.viewTreeObserver.removeOnGlobalLay
outListener(this)
                val aspectRatio = 1.0f // Sesuaikan ini
dengan rasio aspek objek GLB Anda
                val newHeight = (surfaceView.width /
aspectRatio).toInt() + 16 // Tambahkan padding
(16dp di atas dan 16dp di bawah)
                surfaceView.layoutParams.height =
newHeight
                surfaceView.requestLayout()
            }
        })

        scaleGestureDetector =
ScaleGestureDetector(this, object :
ScaleGestureDetector.SimpleOnScaleGestureListener
() {
            override fun onScale(detector:
ScaleGestureDetector): Boolean {
                val scale = detector.scaleFactor
                modelViewer.transformToUnitCube()
                modelViewer.asset?.apply {

```

```

import android.os.Bundle
import android.util.Log
//import android.text.Html
import android.view.Choreographer
import android.view.ScaleGestureDetector
import android.view.SurfaceView
import android.view.ViewTreeObserver
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.core.text.HtmlCompat

```

```

import com.google.android.filament.Engine
import com.google.android.filament.EntityManager
import com.google.android.filament.LightManager
import com.google.android.filament.Skybox
import com.google.android.filament.utils.ModelViewer
import com.google.android.filament.utils.Utils
import java.nio.ByteBuffer

```

```

class TanganActivity : AppCompatActivity() {

```

```

    companion object {
        init { Utils.init() }
    }

```

```

    private lateinit var surfaceView: SurfaceView
    private lateinit var choreographer: Choreographer
    private lateinit var modelViewer: ModelViewer
    private lateinit var scaleGestureDetector:
ScaleGestureDetector
    private var light: Int = 0
    private var engine: Engine? = null

```

```

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_objectglb)

```

```

        surfaceView = findViewById(R.id.surfaceView)
        //surfaceView = SurfaceView(this).apply {
setContentView(this) }
        choreographer = Choreographer.getInstance()
        modelViewer = ModelViewer(surfaceView)
        surfaceView.setOnTouchListener(modelViewer)

```

```

        setupLighting()
        engine = modelViewer.engine
        loadGlb("tangan")
        //modelViewer.scene.skybox =
Skybox.Builder().build(modelViewer.engine)
        setupSkybox()

```

```

        // Atur tinggi SurfaceView secara dinamis

```

```

        surfaceView.viewTreeObserver.addOnGlobalLayoutListe
ner(object : ViewTreeObserver.OnGlobalLayoutListener
{

```

```

            override fun onGlobalLayout() {

```

```

                surfaceView.viewTreeObserver.removeOnGlobalLayoutL
istener(this)

```

```

                val aspectRatio = 1.0f // Sesuaikan ini dengan
rasio aspek objek GLB Anda

```

```

                val transformManager =
modelViewer.engine.transformManager
                val instance =
transformManager.getInstance(root)
                val currentTransform = FloatArray(16)

```

```

                transformManager.getTransform(instance,
currentTransform)
                val newTransform = FloatArray(16)

```

```

                android.opengl.Matrix.scaleM(newTransform, 0,
currentTransform, 0, scale, scale, scale)

```

```

                transformManager.setTransform(instance,
newTransform)
            }
            return true
        }
    }
}

```

```

        surfaceView.setOnTouchListener { v, event ->
            scaleGestureDetector.onTouchEvent(event)
            modelViewer.onTouchEvent(event)
            true
        }

```

```

        val backButton =
findViewById<Button>(R.id.backButton)
        backButton.setOnClickListener {
            // val intent = Intent(this, Anatomi::class.java)
            //
            intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_
TOP or Intent.FLAG_ACTIVITY_NEW_TASK)
            // startActivity(intent)
            // finish() // Kembali ke activity Anatomi
            try {
                val intent = Intent(this,
Anatomi::class.java).apply {
                    flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
                }
                startActivity(intent)
            } catch (e: Exception) {
                Log.e("Tubuh", "Error : ${e.message}")
            }
        }

```

```

        val textView =
findViewById<TextView>(R.id.textView)
        textView.text = HtmlCompat.fromHtml(
            getString(R.string.badan_desk) +
            getString(R.string.badan_tulang) +
            getString(R.string.badan_rongga_dada)
+
            getString(R.string.badan_rongga_perut)
+
            getString(R.string.badan_rongga_panggul),
            HtmlCompat.FROM_HTML_MODE_LEGACY
        )
    }

```

```

        private val frameCallback = object :
Choreographer.FrameCallback {

```

```

        val newHeight = (surfaceView.width /
aspectRatio).toInt() + 16 // Tambahkan padding (16dp di
atas dan 16dp di bawah)
        surfaceView.layoutParams.height = newHeight
        surfaceView.requestLayout()
    }
})

scaleGestureDetector = ScaleGestureDetector(this,
object :
ScaleGestureDetector.SimpleOnScaleGestureListener() {
    override fun onScale(detector:
ScaleGestureDetector): Boolean {
        val scale = detector.scaleFactor
        modelViewer.transformToUnitCube()
        modelViewer.asset?.apply {
            val transformManager =
modelViewer.engine.transformManager
            val instance =
transformManager.getInstance(root)
            val currentTransform = FloatArray(16)
            transformManager.getTransform(instance,
currentTransform)
            val newTransform = FloatArray(16)

            android.opengl.Matrix.scaleM(newTransform, 0,
currentTransform, 0, scale, scale, scale)
            transformManager.setTransform(instance,
newTransform)
        }
        return true
    }
})

surfaceView.setOnTouchListener { v, event ->
    scaleGestureDetector.onTouchEvent(event)
    modelViewer.onTouchEvent(event)
    true
}

val backButton =
findViewById<Button>(R.id.backButton)
backButton.setOnClickListener {
    //    val intent = Intent(this, Anatomi::class.java)
    ////
    intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP
or Intent.FLAG_ACTIVITY_NEW_TASK)
    //    intent.flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
    //    startActivity(intent)
    //    finish() // Kembali ke activity Anatomi
    try {
        val intent = Intent(this,
Anatomi::class.java).apply {
            flags =
Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
        }
        startActivity(intent)
    } catch (e: Exception) {
        Log.e("Tangan", "Error : ${e.message}")
    }
}

val textView =
findViewById<TextView>(R.id.textView)

```

```

        override fun doFrame(currentTime: Long) {
            choreographer.postFrameCallback(this)
            modelViewer.render(currentTime)
        }
    }

    override fun onResume() {
        super.onResume()

        choreographer.postFrameCallback(frameCallback)
    }

    override fun onPause() {
        super.onPause()

        choreographer.removeFrameCallback(frameCallback
)
    }

    override fun onDestroy() {
        super.onDestroy()
        engine?.let {
            it.flushAndWait() // Pastikan memanggil
flushAndWait sebelum shutdown
            modelViewer.destroyModel()
            it.destroy()
        }
    }

    private fun loadGlb(name: String) {
        val buffer = readAsset("models/${name}.glb")
        modelViewer.loadModelGlb(buffer)
        modelViewer.transformToUnitCube()
        modelViewer.scene.skybox =
Skybox.Builder().build(modelViewer.engine)
    }

    private fun readAsset(assetName: String):
ByteBuffer {
        val input = assets.open(assetName)
        val bytes = ByteArray(input.available())
        input.read(bytes)
        input.close()
        return ByteBuffer.wrap(bytes)
    }

    private fun setupLighting() {
        light = EntityManager.get().create()

        LightManager.Builder(LightManager.Type.DIRECTI
ONAL)
            .color(1.0f, 1.0f, 1.0f)
            .intensity(100_000.0f)
            .direction(0.0f, -1.0f, -1.0f)
            .castShadows(true)
            .build(modelViewer.engine, light)
        modelViewer.scene.addEntity(light)
    }

    // For Skybox with white background
    private fun setupSkybox() {
        modelViewer.scene.skybox =
Skybox.Builder().color(1.0f, 1.0f, 1.0f,
1.0f).build(modelViewer.engine)
    }

```

```

        textView.text = HtmlCompat.fromHtml(
            getString(R.string.tangan_desk) +
            getString(R.string.tangan_lengan_atas) +
            getString(R.string.tangan_lengan_bawah) +
            getString(R.string.tangan_pergelangan),
            HtmlCompat.FROM_HTML_MODE_LEGACY
        )
    }

    private val frameCallback = object :
        Choreographer.FrameCallback {
            override fun doFrame(currentTime: Long) {
                choreographer.postFrameCallback(this)
                modelViewer.render(currentTime)
            }
        }

    override fun onResume() {
        super.onResume()
        choreographer.postFrameCallback(frameCallback)
    }

    override fun onPause() {
        super.onPause()

        choreographer.removeFrameCallback(frameCallback)
    }

    override fun onDestroy() {
        super.onDestroy()
        engine?.let {
            it.flushAndWait() // Pastikan memanggil
            flushAndWait sebelum shutdown
            modelViewer.destroyModel()
            it.destroy()
        }
    }

    private fun loadGlb(name: String) {
        val buffer = readAsset("models/${name}.glb")
        modelViewer.loadModelGlb(buffer)
        modelViewer.transformToUnitCube()
        modelViewer.scene.skybox =
            Skybox.Builder().build(modelViewer.engine)
    }

    private fun readAsset(assetName: String): ByteBuffer {
        val input = assets.open(assetName)
        val bytes = ByteArray(input.available())
        input.read(bytes)
        input.close()
        return ByteBuffer.wrap(bytes)
    }

    private fun setupLighting() {
        light = EntityManager.get().create()
        LightManager.Builder(LightManager.Type.DIRECTIONAL)
            .color(1.0f, 1.0f, 1.0f)
            .intensity(100_000.0f)
            .direction(0.0f, -1.0f, -1.0f)
            .castShadows(true)
            .build(modelViewer.engine, light)
        modelViewer.scene.addEntity(light)
    }
}

Kaki.kt
package com.example.anatomimanusia

import android.content.Intent
import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.text.Html
import android.view.Choreographer
import android.view.ScaleGestureDetector
import android.view.SurfaceView
import android.view.ViewTreeObserver
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.core.text.HtmlCompat

import com.google.android.filament.Engine
import com.google.android.filament.EntityManager
import com.google.android.filament.LightManager
import com.google.android.filament.Skybox
import com.google.android.filament.utils.ModelViewer
import com.google.android.filament.utils.Utils
import java.nio.ByteBuffer

class KakiActivity : AppCompatActivity() {

    companion object {
        init { Utils.init() }
    }

    private lateinit var surfaceView: SurfaceView
    private lateinit var choreographer: Choreographer
    private lateinit var modelViewer: ModelViewer
    private lateinit var scaleGestureDetector:
        ScaleGestureDetector
    private var light: Int = 0
    private var engine: Engine? = null

    override fun onCreate(savedInstanceState:
        Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_objectglb)

        surfaceView = findViewById(R.id.surfaceView)
        //surfaceView = SurfaceView(this).apply {
        setContentView(this) }
        choreographer = Choreographer.getInstance()
        modelViewer = ModelViewer(surfaceView)
        surfaceView.setOnTouchListener(modelViewer)

        setupLighting()
        engine = modelViewer.engine
        loadGlb("feet")
        //modelViewer.scene.skybox =
        Skybox.Builder().build(modelViewer.engine)
        setupSkybox()

        // Atur tinggi SurfaceView secara dinamis
        surfaceView.viewTreeObserver.addOnGlobalLayout

```

```

    }

    // For Skybox with white background
    private fun setupSkybox() {
        modelViewer.scene.skybox =
        Skybox.Builder().color(1.0f, 1.0f, 1.0f,
        1.0f).build(modelViewer.engine)
    }
}

Tentang.kt
package com.example.anatomimanusia

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log
import android.widget.Button

class Tentang : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_tentang)

        val backButton =
        findViewById<Button>(R.id.backTentang)
        backButton.setOnClickListener {
            try {
                val intent = Intent(this,
                MainActivity::class.java).apply {
                    flags =
                    Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
                }
                startActivity(intent)
            } catch (e: Exception) {
                Log.e("Tentang-Main", "Error : ${e.message}")
            }
        }
    }
}

MainActifity.kt
package com.example.anatomimanusia

import android.app.ActivityManager
import android.content.Context
import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.widget.Button
import android.widget.ImageView
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
import android.os.Process
import kotlin.system.exitProcess
class MainActivity : AppCompatActivity() {

    private lateinit var materi: ImageView
    private lateinit var anatomi: ImageView
    private lateinit var tentang: ImageView

    override fun onCreate(savedInstanceState: Bundle?) {

```

```

        Listener(object :
        ViewTreeObserver.OnGlobalLayoutListener {
            override fun onGlobalLayout() {

                surfaceView.viewTreeObserver.removeOnGlobalLay
                outListener(this)
                val aspectRatio = 1.0f // Sesuaikan ini
                dengan rasio aspek objek GLB Anda
                val newHeight = (surfaceView.width /
                aspectRatio).toInt() + 16 // Tambahkan padding
                (16dp di atas dan 16dp di bawah)
                surfaceView.layoutParams.height =
                newHeight
                surfaceView.requestLayout()
            }
        })

        scaleGestureDetector =
        ScaleGestureDetector(this, object :
        ScaleGestureDetector.SimpleOnScaleGestureListener
        () {
            override fun onScale(detector:
            ScaleGestureDetector): Boolean {
                val scale = detector.scaleFactor
                modelViewer.transformToUnitCube()
                modelViewer.asset?.apply {
                    val transformManager =
                    modelViewer.engine.transformManager
                    val instance =
                    transformManager.getInstance(root)
                    val currentTransform = FloatArray(16)

                    transformManager.getTransform(instance,
                    currentTransform)
                    val newTransform = FloatArray(16)

                    android.opengl.Matrix.scaleM(newTransform, 0,
                    currentTransform, 0, scale, scale, scale)

                    transformManager.setTransform(instance,
                    newTransform)
                }
                return true
            }
        })

        surfaceView.setOnTouchListener { v, event ->
            scaleGestureDetector.onTouchEvent(event)
            modelViewer.onTouchEvent(event)
            true
        }

        val backButton =
        findViewById<Button>(R.id.backButton)
        backButton.setOnClickListener {
            val intent = Intent(this, Anatomi::class.java)
            //
            intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_
            TOP or Intent.FLAG_ACTIVITY_NEW_TASK)
            //
            startActivity(intent)
            //
            finish() // Kembali ke activity Anatomi
            try {
                val intent = Intent(this,
                Anatomi::class.java).apply {

```



```

        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)

        ViewCompat.setOnApplyWindowInsetsListener(findView<
            wById(R.id.main)) { v, insets ->
                val systemBars =
            insets.getInsets(WindowInsetsCompat.Type.systemBars()
            )
                v.setPadding(systemBars.left, systemBars.top,
            systemBars.right, systemBars.bottom)
            insets
        }

        materi = findViewById(R.id.materi_anatomi)
        anatomi =
        findViewById<ImageView>(R.id.anatomi_tubuh)
        tentang =
        findViewById<ImageView>(R.id.tentang_anatomi)

        materi.setOnClickListener {
            //val i = Intent(applicationContext,
        Materi::class.java)
            //startActivity(i)
            try {
                val intent = Intent(this, Materi::class.java).apply
            {
                flags =
            Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
                }
                startActivity(intent)
            } catch (e: Exception) {
                Log.e("Main-Materi", "Error : ${e.message}")
            }
        }

        anatomi.setOnClickListener {
            //val i = Intent(applicationContext,
        Anatomi::class.java)
            //startActivity(i)
            try {
                val intent = Intent(this,
            Anatomi::class.java).apply {
                    flags =
            Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
                }
                startActivity(intent)
            } catch (e: Exception) {
                Log.e("Main-Anatomi", "Error :
            ${e.message}")
            }
        }

        tentang.setOnClickListener {
            //val i = Intent(applicationContext,
        Tentang::class.java)
            //startActivity(i)
            try {
                val intent = Intent(this,
            Tentang::class.java).apply {
                    flags =
            Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
                }
                startActivity(intent)
            } catch (e: Exception) {

```

```

                flags =
            Intent.FLAG_ACTIVITY_REORDER_TO_FRONT
                }
                startActivity(intent)
            } catch (e: Exception) {
                Log.e("Kaki", "Error : ${e.message}")
            }
        }

        val textView =
        findViewById<TextView>(R.id.textView)
        textView.text = HtmlCompat.fromHtml(
            getString(R.string.kaki_desk) +
            getString(R.string.kaki_paha) +
            getString(R.string.kaki_betis) +
            getString(R.string.kaki_pergelangan),
            HtmlCompat.FROM_HTML_MODE_LEGACY
        )
    }

    private val frameCallback = object :
        Choreographer.FrameCallback {
        override fun doFrame(currentTime: Long) {
            choreographer.postFrameCallback(this)
            modelViewer.render(currentTime)
        }
    }

    override fun onResume() {
        super.onResume()

        choreographer.postFrameCallback(frameCallback)
    }

    override fun onPause() {
        super.onPause()

        choreographer.removeFrameCallback(frameCallback
        )
    }

    // override fun onDestroy() {
    //     super.onDestroy()
    //     modelViewer.destroyModel()
    //     modelViewer.engine.destroy()
    // }

    override fun onDestroy() {
        super.onDestroy()
        engine?.let {
            it.flushAndWait() // Pastikan memanggil
            flushAndWait sebelum shutdown
            modelViewer.destroyModel()
            it.destroy()
        }
    }

    private fun loadGlb(name: String) {
        val buffer = readAsset("models/${name}.glb")
        modelViewer.loadModelGlb(buffer)
        modelViewer.transformToUnitCube()
        modelViewer.scene.skybox =
        Skybox.Builder().build(modelViewer.engine)
    }

```

```

        Log.e("Main-tentang", "Error : ${e.message}")
    }
}

//val activity: MainActivity = MainActivity()
val buttonClose: Button =
findViewById(R.id.buttonClose)
buttonClose.setOnClickListener {
    val activityManager =
getSystemService(Context.ACTIVITY_SERVICE) as
ActivityManager
    val appTask = activityManager.appTasks
    for (task in appTask) {
        task.finishAndRemoveTask()
    }
    finishAndRemoveTask()
    System.exit(0)
}
}
}

```

AndroidManifest.xml

```

<manifest>
<uses-permission
android:name="android.permission.INTERNET"/>
<application android:allowBackup="true"
android:dataExtractionRules="@xml/data_extraction_rule
s" android:fullBackupContent="@xml/backup_rules"
android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
android:roundIcon="@mipmap/ic_launcher_round"
android:supportRtl="true"
android:theme="@style/Theme.AnatomiManusia"
tools:targetApi="31">
<activity android:name=".MainActivity"
android:exported="true">
<intent-filter>
<action android:name="android.intent.action.MAIN"/>
<category
android:name="android.intent.category.LAUNCHER"/>
</intent-filter>
</activity>
<activity android:name=".Materi"/>
<activity android:name=".Anatomi"/>
<activity android:name=".Tentang"/>
<!--
<activity android:name=".KepalaActivity" />
-->
<!-- <activity android:name=".KakiActivity" />-->
<!--
<activity android:name=".TanganActivity" />
-->
<!-- <activity android:name=".TubuhActivity" />-->
<activity android:name=".View3DActivity"/>
<activity android:name=".DetailActivity"/>
</application>
</manifest>
Anatomi.xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/andr
oid"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

```

```

private fun readAsset(assetName: String):
ByteBuffer {
    val input = assets.open(assetName)
    val bytes = ByteArray(input.available())
    input.read(bytes)
    input.close()
    return ByteBuffer.wrap(bytes)
}

private fun setupLighting() {
    light = EntityManager.get().create()

LightManager.Builder(LightManager.Type.DIRECTI
ONAL)
    .color(1.0f, 1.0f, 1.0f)
    .intensity(100_000.0f)
    .direction(0.0f, -1.0f, -1.0f)
    .castShadows(true)
    .build(modelViewer.engine, light)
    modelViewer.scene.addEntity(light)
}

// For Skybox with white background
private fun setupSkybox() {
    modelViewer.scene.skybox =
Skybox.Builder().color(1.0f, 1.0f, 1.0f,
1.0f).build(modelViewer.engine)
}
}

```

Materi.xml

```

<?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="@color/white"
    tools:context=".Materi">

<ScrollView
    android:id="@+id/scrollView"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_margin="16dp"
    app:layout_constraintTop_toTopOf="parent"
app:layout_constraintBottom_toTopOf="@+id/back
Materi"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical" >

<!-- CardViews with numbers 1 to 12 -->
<androidx.cardview.widget.CardView
    android:id="@+id/cardView1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"

```

```

<ImageView
    android:id="@+id/imageBody"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:scaleType="fitXY"
    android:clickable="true"
    android:focusable="true"/>

<Button
    android:id="@+id/backButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Kembali"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="16dp" />

</RelativeLayout>

<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:padding="16dp"
    android:background="@color/white"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="8dp"
            android:elevation="4dp">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical"
                android:padding="16dp">

                <ImageView
                    android:id="@+id/id_imageView"
                    android:layout_width="match_parent"
                    android:layout_height="250dp"/>

                <TextView
                    android:id="@+id/textView"
                    android:layout_marginTop="5dp"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:textColor="@color/black"
                    android:text=""
                    android:textSize="15sp" />

            </LinearLayout>
        </androidx.cardview.widget.CardView>

    </LinearLayout>

    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:elevation="4dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="16dp"
            android:text="Sistem Muskuloskeletal"
            android:gravity="center"

            android:textColor="@android:color/black"/>
        </androidx.cardview.widget.CardView>

        <!-- Add similar blocks for numbers 2 to 12 -->
    >

    <!-- Example for number 2 -->
    <androidx.cardview.widget.CardView
        android:id="@+id/cardView2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="4dp">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="16dp"
            android:text="Sistem Kardiovaskular"
            android:gravity="center"

            android:textColor="@android:color/black"/>
        </androidx.cardview.widget.CardView>

        <!-- Add similar blocks for numbers 3 to 12 -->
    >

    <!-- Example for number 3 -->
    <androidx.cardview.widget.CardView
        android:id="@+id/cardView3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="4dp">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="16dp"
            android:text="Sistem Pernafasan"
            android:gravity="center"

            android:textColor="@android:color/black"/>
        </androidx.cardview.widget.CardView>

    <!-- Repeat for numbers 4 to 12 -->
    <androidx.cardview.widget.CardView
        android:id="@+id/cardView4"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="4dp">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="16dp"

```

```

</ScrollView>

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/and
oid"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    android:background="@android:color/white">

    <Button
        android:id="@+id/backButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Back"
        android:layout_gravity="start"
        android:layout_marginBottom="16dp" />

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="300dp"
        android:layout_gravity="center">

        <SurfaceView
            android:id="@+id/surfaceView"
            android:layout_width="match_parent"
            android:layout_height="match_parent" />
    </FrameLayout>

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="16dp"
        android:padding="16dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:id="@+id/popupTextView"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:textSize="16sp"
                android:padding="8dp" />
        </LinearLayout>
    </ScrollView>

</LinearLayout>

    android:text="Sistem Pencernaan"
    android:gravity="center"

    android:textColor="@android:color/black"/>
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView5"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="4dp">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp"
        android:text="Sistem Urinaria"
        android:gravity="center"

    android:textColor="@android:color/black"/>
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView6"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="4dp">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp"
        android:text="Sistem Integumen"
        android:gravity="center"

    android:textColor="@android:color/black"/>
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView7"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="4dp">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp"
        android:text="Sistem Panca Indra"
        android:gravity="center"

    android:textColor="@android:color/black"/>
</androidx.cardview.widget.CardView>

<!--<?xml version="1.0" encoding="utf-8"?-->
<!--<LinearLayout-->
<!--
xmlns:android="http://schemas.android.com/apk/res/andr
oid"-->
<!-- android:layout_width="wrap_content"-->
<!-- android:layout_height="wrap_content"-->
<!-- android:orientation="vertical"-->
<!-- android:padding="16dp"-->

```

```

<!-- <ImageView-->
<!--     android:id="@+id/popupImageView"-->
<!--     android:layout_width="100dp"-->
<!--     android:layout_height="100dp"-->
<!--     android:layout_gravity="center"/>-->
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="16dp"
    android:text="Sistem Reproduksi"
    android:gravity="center"

<!-- <TextView-->
<!--     android:id="@+id/popupTextView"-->
<!--     android:layout_width="wrap_content"-->
<!--     android:layout_height="wrap_content"-->
<!--     android:layout_gravity="center"-->
<!--     android:layout_marginTop="16dp"/>-->
    android:textColor="@android:color/black"/>
</androidx.cardview.widget.CardView>

<!-- <Button-->
<!--     android:id="@+id/nextButton"-->
<!--     android:layout_width="wrap_content"-->
<!--     android:layout_height="wrap_content"-->
<!--     android:text="Halaman Detail"-->
<!--     android:layout_below="@id/popupTextView"-->
<!--     android:layout_centerHorizontal="true"-->
<!--     android:layout_marginTop="16dp"/>-->
<!--</LinearLayout>-->
<androidx.cardview.widget.CardView
    android:id="@+id/cardView9"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="4dp">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp"
        android:text="Sistem Endokrin"
        android:gravity="center"

<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/and
roid"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <SurfaceView
        android:id="@+id/surfaceView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

    <ScrollView
        android:id="@+id/scrollView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/surfaceView"
        android:layout_marginBottom="80dp">

        <TextView
            android:id="@+id/textView"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Some description here"
            android:padding="16dp" />

    </ScrollView>

    <Button
        android:id="@+id/backButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:text="Back"
        android:layout_marginBottom="16dp" />

</RelativeLayout>

<FrameLayout
xmlns:android="http://schemas.android.com/apk/res/andr
oid"
    android:layout_width="match_parent"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    <androidx.cardview.widget.CardView
        android:id="@+id/cardView10"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="4dp">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="16dp"
            android:text="Sistem Imun"
            android:gravity="center"

        android:textColor="@android:color/black"/>
        </androidx.cardview.widget.CardView>

        <androidx.cardview.widget.CardView
            android:id="@+id/cardView11"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="8dp"
            app:cardCornerRadius="8dp"
            app:cardElevation="4dp">
            <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:padding="16dp"
                android:text="Sistem Limfatik"
                android:gravity="center"

            android:textColor="@android:color/black"/>
            </androidx.cardview.widget.CardView>

            <androidx.cardview.widget.CardView
                android:id="@+id/cardView12"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginBottom="8dp"

```



```

<!--&lt;!&ndash;
android:id="@+id/textView"&ndash;&gt;-->
<!--&lt;!&ndash;
android:layout_width="match_parent"&ndash;&gt;-->
<!--&lt;!&ndash;
android:layout_height="wrap_content"&ndash;&gt;-->
<!--&lt;!&ndash;      android:padding="16dp"
/>&ndash;&gt;
<!--&lt;!&ndash;    </ScrollView>&ndash;&gt;-->

```

```

<!--</RelativeLayout>-->

```

```

<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/and
roid"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/main"
    tools:context=".MainActivity">

```

```

<!-- Container for images and texts -->

```

```

<LinearLayout
    android:id="@+id/container"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="150dp"
    android:orientation="vertical">

```

```

<!-- First row with two images and texts -->

```

```

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

```

```

<!-- Image 1 and Text 1 -->

```

```

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="8dp">

```

```

<ImageView
    android:id="@+id/materi_anatomi"
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:src="@drawable/materi" />

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:paddingTop="5dp"
    android:text="Materi"
    android:layout_gravity="center"/>

```

```

</LinearLayout>

```

```

<!-- Image 2 and Text 2 -->

```

```

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"

```

dalam ensiklopedia memuat sebuah informasi berdasarkan satu bahasan pokok"/>

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="10dp"
    android:textColor="@color/black"
    android:text="Anatomi adalah ilmu yang
menngkaji mengenai bagian bagian tubuh manusia
dan hubungan antara mereka"/>

```

```

<TableLayout
    android:layout_width="match_parent"
    android:layout_height="194dp"
    android:layout_marginTop="10dp"
    android:background="@color/white"
    android:orientation="vertical">

```

```

<TableRow
    android:background="@color/white">

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginTop="10dp"
    android:text="Nama"
    android:textColor="@color/black" />

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginTop="10dp"
    android:text=" : "
    android:textColor="@color/black" />

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginTop="10dp"
    android:text="Agnes Mulyana"
    android:textColor="@color/black" />

```

```

</TableRow>

```

```

<TableRow
    android:background="@color/white">

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginTop="10dp"
    android:text="NIM"
    android:textColor="@color/black" />

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginTop="10dp"
    android:text=" : "

```

```

        android:padding="8dp">

        <ImageView
            android:id="@+id/anatomi_tubuh"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:src="@drawable/anatomy" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:paddingTop="5dp"
            android:layout_gravity="center"
            android:text="Anatomi" />
    </LinearLayout>
</LinearLayout>

<!-- Second row with one image and text -->
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:padding="8dp"
    android:layout_gravity="center_horizontal"
    android:orientation="vertical">

    <ImageView
        android:id="@+id/tentang_anatomi"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:layout_gravity="center"
        android:src="@drawable/team" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:layout_gravity="center"
        android:text="Tentang" />
    </LinearLayout>
</LinearLayout>

<!-- Button Tutup -->
<Button
    android:id="@+id/buttonClose"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="16dp"
    android:text="Tutup"/>

</RelativeLayout>

<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission
        android:name="android.permission.INTERNET" />

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

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left"
            android:layout_marginTop="10dp"
            android:text="220280047"
            android:textColor="@color/black" />
    </TableRow>

    <TableRow
        android:background="@color/white">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left"
            android:layout_marginTop="10dp"
            android:text="Program Studi"
            android:textColor="@color/black" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left"
            android:layout_marginTop="10dp"
            android:text=":" />
            android:textColor="@color/black" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left"
            android:layout_marginTop="10dp"
            android:text="Teknik Informatika"
            android:textColor="@color/black" />
    </TableRow>

</TableLayout>

<Button
    android:id="@+id/backTentang"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="16dp"
    android:text="Kembali" />

</LinearLayout>

```



```

        android:allowBackup="true"

        android:dataExtractionRules="@xml/data_extraction_rules"
    }

    android:fullBackupContent="@xml/backup_rules"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.AnatomiManusia"
    tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action
                    android:name="android.intent.action.MAIN" />

                    <category
                        android:name="android.intent.category.LAUNCHER" />
                </intent-filter>
            </activity>
            <activity android:name=".Materi" />
            <activity android:name=".Anatomi" />
            <activity android:name=".Tentang" />
            <activity android:name=".KepalaActivity" />
            <activity android:name=".KakiActivity" />
            <activity android:name=".TanganActivity" />
            <activity android:name=".TubuhActivity" />
            <activity android:name=".WebViewActivity" />
        </application>

</manifest>

```



SERTIFIKAT

Nomor : 330/PRK/LB-TI/II.3.AU/D/2023

Diberikan Kepada :

Agnes Mulyana

NIM. 220280047

Telah Menyelesaikan

PRAKTIKUM III

Laboratorium Teknik Informatika

Program Studi Teknik Informatika Fakultas Teknik

Universitas Muhammadiyah Parepare

Pada tanggal 16 Januari 2023 - 29 Maret 2023

Parepare, 15 Juni 2023



**KETUA PRODI
TEKNIK INFORMATIKA**

Marlina. S.kom., M.kom

NBM. 1162 680



**KEPALA
LABORATORIUM**

Ir. Untung Suwardoyo, S.Kom., MT., IPP.

NBM. 1288 973





**Nilai Praktikum Laboratorium Teknik Informatika
Fakultas Teknik Universitas Muhammadiyah Parepare Tahun 2023**



MATERI	NILAI ANGKA
PEMROGRAMAN WEB 2	82,5
PEMROGRAMAN JAVA	96
PEMROGRAMAN DELPHI 10	97,5
JARINGAN KOMPUTER 2	85
RATA-RATA	90,25
NILAI HURUF	A





SERTIFIKAT

Nomor : 097/PRK/LB-TI/11.3.AU/D/2022

diberikan kepada :

AGNES MULYANA

NIM. 220280047

telah menyelesaikan

PRAKTIKUM IV

Laboratorium Teknik Informatika

Program Studi Teknik Informatika Fakultas Teknik

Universitas Muhammadiyah Parepare

Pada tanggal 14 September 2022 - 25 Desember 2022

Parepare, 02 Januari 2023



**KETUA PRODI
TEKNIK INFORMATIKA**

MARLINA. S.Kom., M.Kom

**KEPALA
LABORATORIUM**

Ir. UNTUNG SUWARDYO, S.Kom., MT., I.PP.



**Nilai Praktikum Laboratorium Teknik Informatika
Fakultas Teknik Universitas Muhammadiyah Parepare Tahun 2022**



MATERI	NILAI ANGKA
DESAIN 3D BLENDER	90
UI/UX ADOBE XD	79
PEMROGRAMAN PYTHON	92
MATLAB	99
RATA-RATA	90
NILAI HURUF	A





SERTIFIKAT

Nomor : 751/PRK/LB-TI/II.3.AU/D/2023

Diberikan Kepada :

Agnes Mulyana

NIM. 220280047

Telah Menyelesaikan

PRAKTIKUM V

Laboratorium Teknik Informatika

Program Studi Teknik Informatika Fakultas Teknik

Universitas Muhammadiyah Parepare

Pada tanggal 18 September - 16 Desember 2023

Parepare, 03 Januari 2024



**KETUA PRODI
TEKNIK INFORMATIKA**



Marlina S.kom., M.kom

NBM. 1162 680

**KEPALA
LABORATORIUM**



Ir. Untung Suwardoyo, S.Kom., MT., IPP.

NBM. 1288 973



Nilai Praktikum Laboratorium Teknik Informatika
Fakultas Teknik Universitas Muhammadiyah Parepare Tahun 2023



MATERI	NILAI ANGKA
PENGOLAHAN CITRA DIGITAL	98
AUGMENTED DAN VIRTUAL REALITY	86
MIKROKONTROLER	75
PEMROGRAMAN MOBILE	92
RATA-RATA	88
NILAI HURUF	A





KARTU MONITORING BIMBINGAN
MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

PROPOSAL

Mahasiswa : AGNES MULYANA	Pembimbing I : Wahyuddin, S.Kom., M.Kom.
NIM : 220280047	Pembimbing II : Andi Warfiah, S.Kom., M.Kom.
Judul Skripsi : PERANCANGAN SISTEM YANG INTERAKTIF PADA ENSIKLOPEDIA ANATOMI TUBUH MANUSIA	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 - Perbaiki desain sistem		Konsultasi 1 - BAB III di Tambahkan desain sistem - Pelajari Panduan Penulisan Skripsi	f
Konsultasi 2 - sistem interface - Buku anatomi tubuh		Konsultasi 2 - Format penulisan - Rimpitan halaman - ket. Gambar & tabel	f
Konsultasi 3 Tambahkan Animasi 3D		Konsultasi 3 - titik Margin disesuaikan dgn pedoman - Bahasa asing dicetak miring	f
Konsultasi 4 Perbaiki desain sistem		Konsultasi 4 Bab. III - kerangka pikir. - Desain sistem dilengkapi	f
Konsultasi 5 Acc		Konsultasi 5 ACC	f

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

Lanjutan ...

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 6		Konsultasi 6	
Konsultasi 7		Konsultasi 7	
Konsultasi 8		Konsultasi 8	
Konsultasi 9		Konsultasi 9	
Konsultasi 10		Konsultasi 10	

Mengetahui
Ketua Program Studi


Marlina, S.Kom., M.Kom.
NBM. 1162 680

Parepare, 25 Oktober 2023

Mahasiswa

AGNES MULYANA
NIM. 220280047

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik



KARTU MONITORING BIMBINGAN
MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

SKRIPSI

Mahasiswa : AGNES MULYANA	Pembimbing I : Wahyuddin, S.Kom., M.Kom.
NIM : 220280047	Pembimbing II : Andi Warfiah, S.Kom., M.Kom.
Judul Skripsi : APLIKASI ENSIKLOPEDIA ANATOMI TUBUH MANUSIA INTERAKTIF BERBASIS ANDROID	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 perbaiki Activity Diagram		Konsultasi 1 Penjelasan aplikasi	4
Konsultasi 2 - Ikuti panduan skripsi		Konsultasi 2 Penulisan dokumen sesuai dgn pedoman	4
Konsultasi 3 - Acc ujian Hasil		Konsultasi 3 Acc	4
Konsultasi 4 Acc ujian tutup		Konsultasi 4 Acc Ujian Tutup	4
Konsultasi 5		Konsultasi 5	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

Lanjutan ...

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 6		Konsultasi 6	
Konsultasi 7		Konsultasi 7	
Konsultasi 8		Konsultasi 8	
Konsultasi 9		Konsultasi 9	
Konsultasi 10		Konsultasi 10	

Parepare, 7 Mei 2024

Mengetahui
Ketua Program Studi

Mahasiswa

Marlina, S.Kom.,M.Kom.
NBM. 1162 680

AGNES MULYANA
NIM. 220280047

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik