Project Report

Waste Management App

Course Code : CSE226

Course Title: Developing Android Apps

Submitted By:

Name:- Nagakalyan Ravuri

Reg No:- 12220944

Name:- Mandadapu Chandravadhan

Reg No:- 12222491

Name:- A. Akash Goud

Reg No:- 12221995

Submitted To:

DR. Vikas Sharma

LOVELY PROFESSIONAL UNIVERSITY

DECLARATION

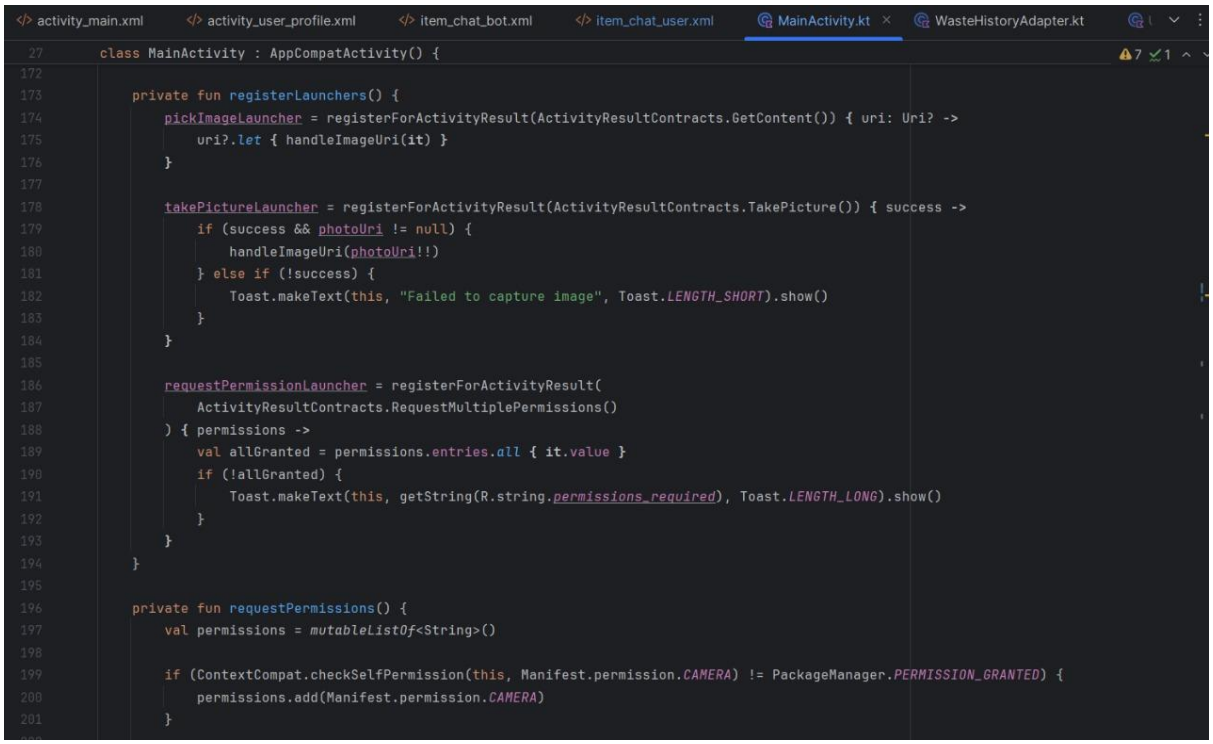
I A.Akash Goud a student of Bachelor of Technology under CSE discipline at Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on my own work and is genuine

Your name: A.Akash Goud

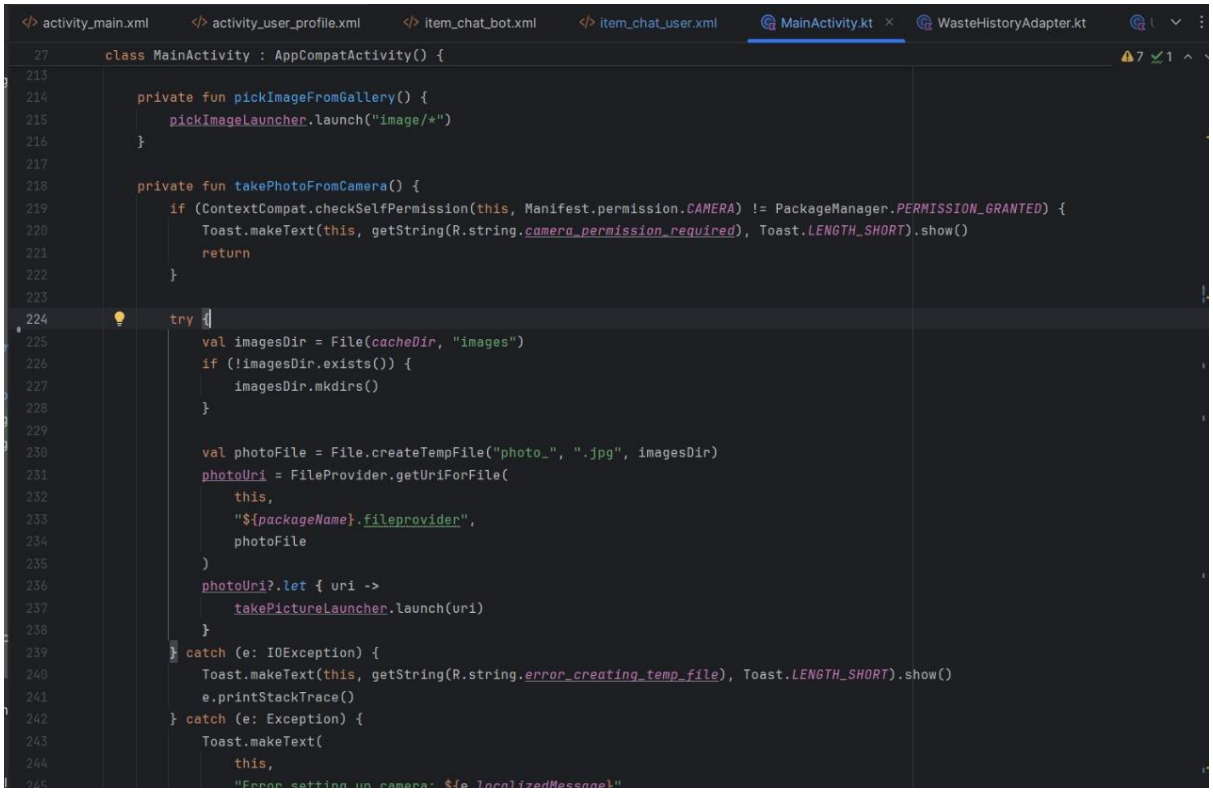
Registrartion Number: 12221995

Table of Contents

1.Main Activity Code:-

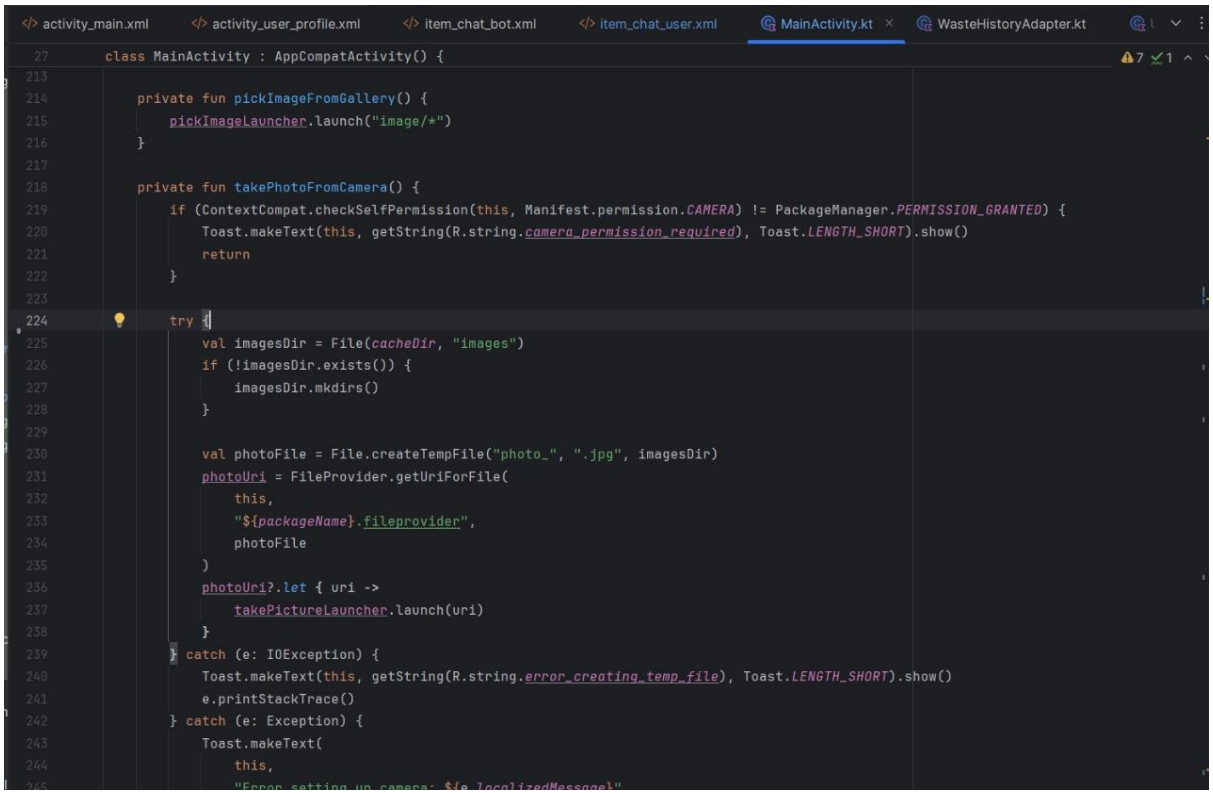


```
activity_main.xml  activity_user_profile.xml  item_chat_bot.xml  item_chat_user.xml  MainActivity.kt  WasteHistoryAdapter.kt

class MainActivity : AppCompatActivity() {

    private fun registerLaunchers() {
        pickImageLauncher = registerForActivityResult(ActivityResultContracts.GetContent()) { uri: Uri? ->
            uri?.let { handleImageUri(it) }
        }

        takePictureLauncher = registerForActivityResult(ActivityResultContracts.TakePicture()) { success ->
            if (success && photoUri != null) {
                handleImageUri(photoUri!!)
            } else if (!success) {
                Toast.makeText(this, "Failed to capture image", Toast.LENGTH_SHORT).show()
            }
        }

        requestPermissionLauncher = registerForActivityResult(
            ActivityResultContracts.RequestMultiplePermissions()
        ) { permissions ->
            val allGranted = permissions.entries.all { it.value }
            if (!allGranted) {
                Toast.makeText(this, getString(R.string.permissions_required), Toast.LENGTH_LONG).show()
            }
        }
    }

    private fun requestPermissions() {
        val permissions = mutableListOf<String>()

        if (ContextCompat.checkSelfPermission(this, Manifest.permission.CAMERA) != PackageManager.PERMISSION_GRANTED) {
            permissions.add(Manifest.permission.CAMERA)
        }
```

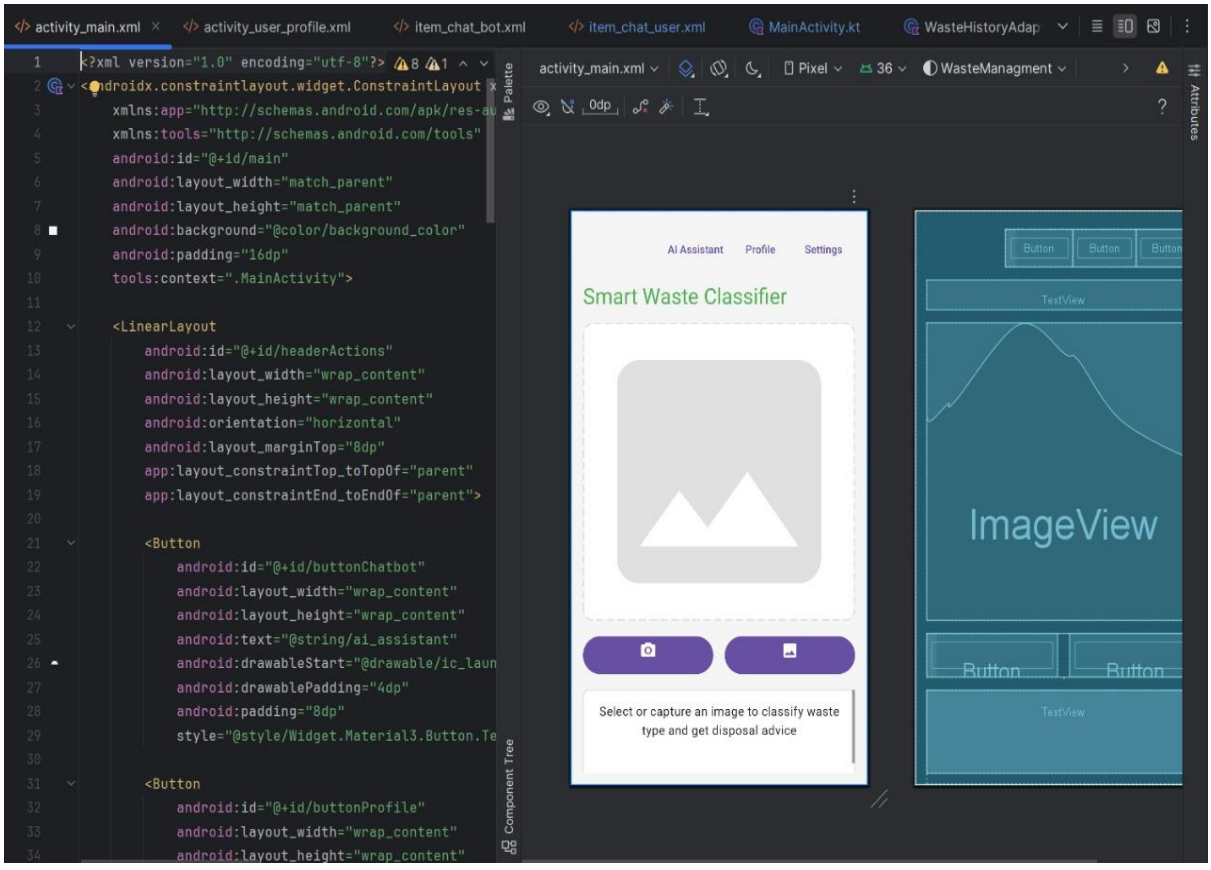


```
activity_main.xml  activity_user_profile.xml  item_chat_bot.xml  item_chat_user.xml  MainActivity.kt  WasteHistoryAdapter.kt

class MainActivity : AppCompatActivity() {

    private fun pickImageFromGallery() {
        pickImageLauncher.launch("image/*")
    }

    private fun takePhotoFromCamera() {
        if (ContextCompat.checkSelfPermission(this, Manifest.permission.CAMERA) != PackageManager.PERMISSION_GRANTED) {
            Toast.makeText(this, getString(R.string.camera_permission_required), Toast.LENGTH_SHORT).show()
            return
        }

        try {
            val imagesDir = File(cacheDir, "images")
            if (!imagesDir.exists()) {
                imagesDir.mkdirs()
            }

            val photoFile = File.createTempFile("photo_", ".jpg", imagesDir)
            photoUri = FileProvider.getUriForFile(
                this,
                "${packageName}.fileprovider",
                photoFile
            )
            photoUri?.let { uri ->
                takePictureLauncher.launch(uri)
            }
        } catch (e: IOException) {
            Toast.makeText(this, getString(R.string.error_creating_temp_file), Toast.LENGTH_SHORT).show()
            e.printStackTrace()
        } catch (e: Exception) {
            Toast.makeText(
                this,
                "Error setting up camera: ${e.localizedMessage}"
```

2.Tensor Flow Code:-



3.Activity Main XML Code:-



AI Assistant Profile Settings

Smart Waste Classifier

Select or capture an image to classify waste type and get disposal advice


15:03
0.18 KB/S
66%
Waste management
Classifier

Conclusion

The project already delivers a cohesive experience: capture/choose waste images, classify them reliably on-device, surface contextual advice, keep a running history, and chat with an assistant for broader sustainability questions. The codebase is cleanly structured (MVVM), dependency-ready for cloud AI, and localized, making it easy to maintain or extend. Even with the mock fallback, the classifier handles common waste categories, ensuring the app stays functional without network or model file issues.

Future Scope

- **Real LLM integration**: swap the rule-based chatbot with OpenAI/Gemini/Claude calls, including streaming replies and better context handling.
- **Realtime camera classification**: continuous preview inference for faster feedback, plus batching multiple photos.
- **Model improvements**: auto-download updated TFLite models, apply quantization, add edge detection or preprocessing filters.
- **Analytics & gamification**: show personal impact stats, leaderboards, streaks, or community challenges.
- **Voice + accessibility**: voice commands, TTS responses, and better screen-reader support.
- **Cloud sync & sharing**: sync history across devices, export PDFs, or share classification details with municipalities.

4. GitHub URL Link
   https://github.com/akashgoudaithagoni/Waste-Management-Android-App1