Task Core 3– Spike: [The Clubhouse]

https://github.com/SoftDevMobDev-2023-Classrooms/core3-MarcoGiacoppo

Goals:

The goal of this task is to create an android app in Android Studio. The aim of the app is to read data from a file and store it in a list. The list will be held inside a recyclerView where users will be able to sort the list on a click. Users will be shown a list of data from the provided csv file, each data will be color-coded to make it easier for users to differentiate.

The app consists of these main functions:

- The groups.csv file is located inside res/raw folder because it makes it simpler to open and read the file using *recources.openRawResourse(R.raw.groups)*
- There's an ImageView called *sort* which allows users to sort the data based on the logic that's implemented to this ImageView. In this case, I made it so when users clicks on it, it sorts the data based on the data group "Xsports".
- I ordered the list based on the date time using java's *LocalDateTime.* When I only used the default date time sorter, it only sorts the data based on the date (excluding the month). That's why I needed to make a *DateTimeFormatter* and make a custom pattern.

Tools and Resources Used

This section lists related software, tools, libraries, API's, and other resources used for this knowledge gap.

- W3School
- YouTube Videos
- Modules on previous weeks
- GitHub
- Android Studio IDE

Knowledge Gaps and Solutions

This section presents the listed knowledge gaps and their solutions with supporting images, screenshots and captions where appropriate/required.

Gap 1: Understand the filesystem

I've placed the 'groups.csv' file in the '/res/raw' directory, which is a common location for storing read-only resource files in Android apps. I've also used the 'resources.openRawResource(R.raw.groups' method to access and read the content of the CSV file. This approach allows me to work with read-only files securely, as these files cannot be modified at runtime.

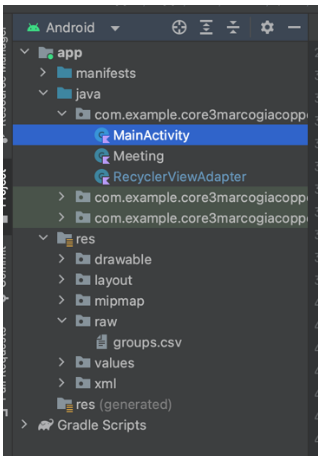
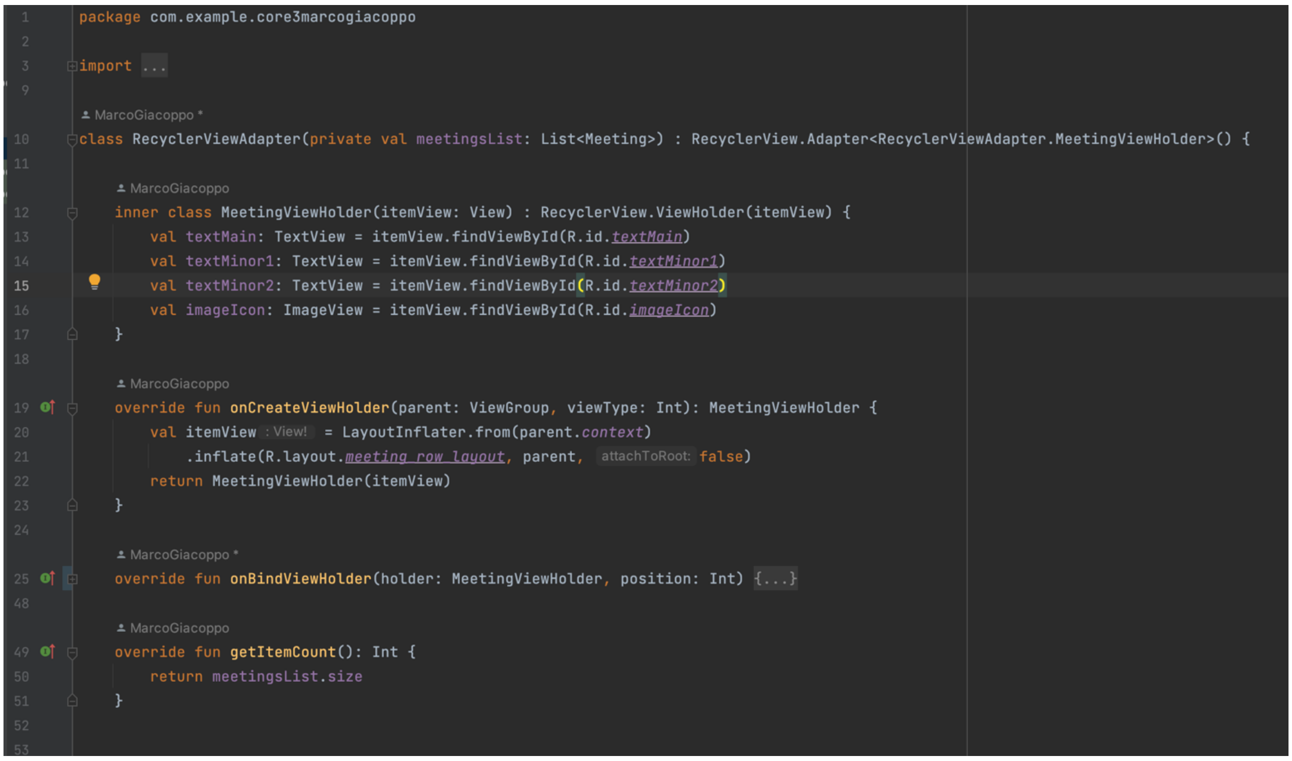


Figure 1: File system

Gap 2: List performance and adapters

I've chose to use a RecyclerView to display the list of meetings. RecyclerView is a high-performance UI component specifically designed for displaying large lists efficiently. I've created a 'RecyclerViewAdapter' that efficiently binds data into the RecyclerView's views. I've also implemented the 'onCreateViewHolder' , 'onBindViewHolder' and 'getItemCount' methods.



```kotlin
package com.example.core3marcogiacoppo

import ...

MarcoGiacoppo *
class RecyclerViewAdapter(private val meetingsList: List<Meeting>) : RecyclerView.Adapter<RecyclerViewAdapter.MeetingViewHolder>() {

    MarcoGiacoppo
    inner class MeetingViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        val textMain: TextView = itemView.findViewById(R.id.textMain)
        val textMinor1: TextView = itemView.findViewById(R.id.textMinor1)
        val textMinor2: TextView = itemView.findViewById(R.id.textMinor2)
        val imageIcon: ImageView = itemView.findViewById(R.id.imageIcon)
    }

    MarcoGiacoppo
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): MeetingViewHolder {
        val itemView :View! = LayoutInflater.from(parent.context)
            .inflate(R.layout.meeting_row_layout, parent,  attachToRoot: false)
        return MeetingViewHolder(itemView)
    }

    MarcoGiacoppo *
    override fun onBindViewHolder(holder: MeetingViewHolder, position: Int) {...}

    MarcoGiacoppo
    override fun getItemCount(): Int {
        return meetingsList.size
    }

```

Figure 2: Adapter for recyclerView

Gap 3: Options menu

I created a 'button' on top of the recyclerView to filter the data based on the property of group.

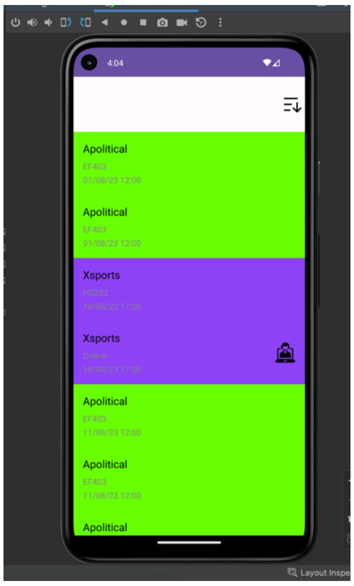
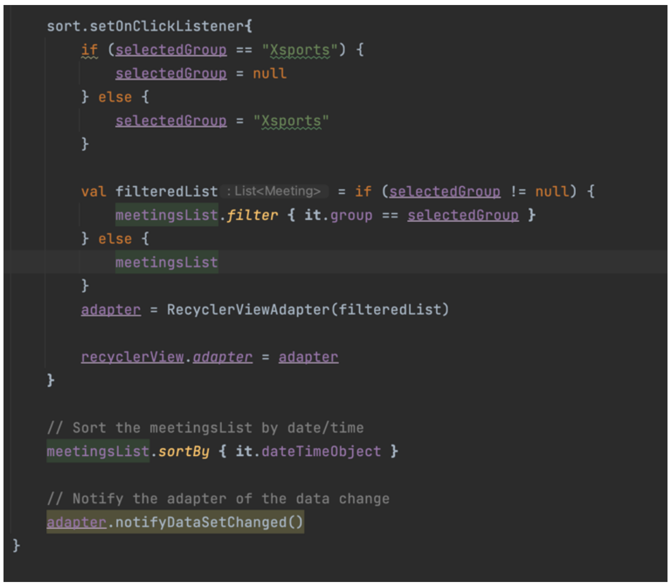


Figure 3: App on run

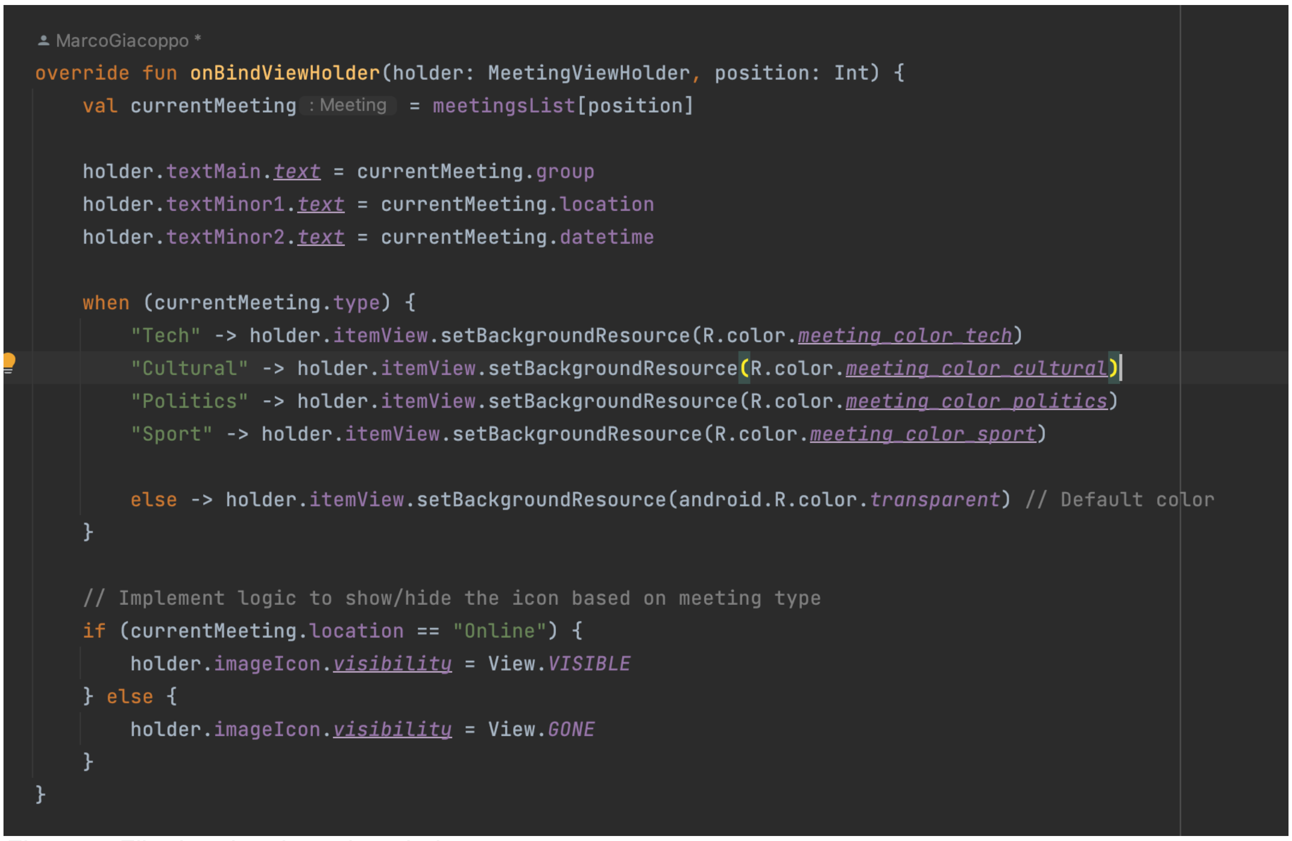


```kotlin
    sort.setOnClickListener{
        if (selectedGroup == "Xsports") {
            selectedGroup = null
        } else {
            selectedGroup = "Xsports"
        }

        val filteredList :List<Meeting> = if (selectedGroup != null) {
            meetingsList.filter { it.group == selectedGroup }
        } else {
            meetingsList
        }
        adapter = RecyclerViewAdapter(filteredList)

        recyclerView.adapter = adapter
    }

    // Sort the meetingsList by date/time
    meetingsList.sortBy { it.dateTimeObject }

    // Notify the adapter of the data change
    adapter.notifyDataSetChanged()
}
```

Figure 4: Listener for the sort button

Gap 4: Work with list data

I've implemented the sorting logic for the 'meetingsList' based on the 'datetime' property, ensuring that meetings are displayed in chronological order. I've also added an imageView to represent a button which allows users to filter meetings by the 'Xsports' group. When the button is clicked, it toggles the filter between showing all meetings and showing only 'Xsports' meetings. I've added a logic to implement the background colors for each meeting type based on the 'type' property, also an icon to show if the meeting location is online.

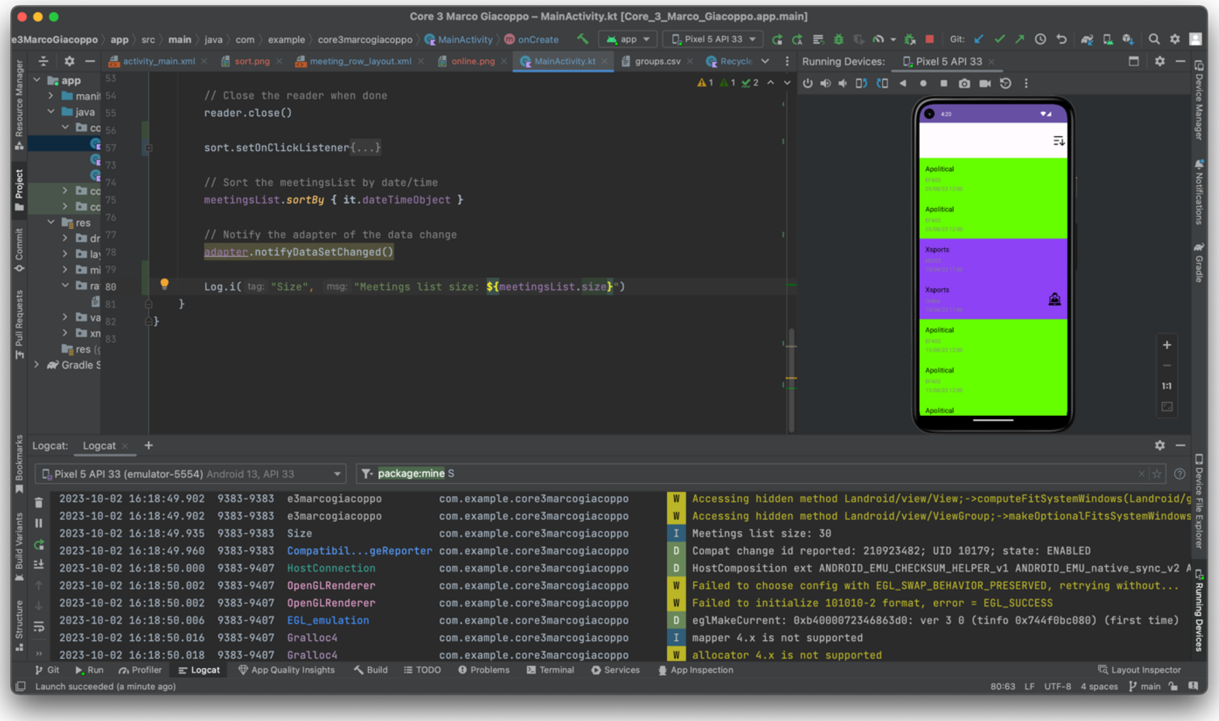


```
MarcoGiacoppo *
override fun onBindViewHolder(holder: MeetingViewHolder, position: Int) {
    val currentMeeting : Meeting  = meetingsList[position]

    holder.textMain.text = currentMeeting.group
    holder.textMinor1.text = currentMeeting.location
    holder.textMinor2.text = currentMeeting.datetime

    when (currentMeeting.type) {
        "Tech" -> holder.itemView.setBackgroundResource(R.color.meeting_color_tech)
        "Cultural" -> holder.itemView.setBackgroundResource(R.color.meeting_color_cultural)
        "Politics" -> holder.itemView.setBackgroundResource(R.color.meeting_color_politics)
        "Sport" -> holder.itemView.setBackgroundResource(R.color.meeting_color_sport)

        else -> holder.itemView.setBackgroundResource(android.R.color.transparent) // Default color
    }

    // Implement logic to show/hide the icon based on meeting type
    if (currentMeeting.location == "Online") {
        holder.imageIcon.visibility = View.VISIBLE
    } else {
        holder.imageIcon.visibility = View.GONE
    }
}
```

Figure 5: Filtering data based on their names

Gap 5: Command of IDE

For this task, there was no problem doing the requirements. I've been using recyclerView for my custom app, so I think that's why I didn't have any problem doing this assignment. Nonetheless, I used a log to check on the meeting size to make sure it has the exact amount of data inside the csv file.



Core 3 Marco Giacoppo – MainActivity.kt [Core_3_Marco_Giacoppo.app.main]

```kotlin
        // Close the reader when done
        reader.close()

        sort.setOnClickListener{...}

        // Sort the meetingsList by date/time
        meetingsList.sortBy { it.dateTimeObject }

        // Notify the adapter of the data change
        adapter.notifyDataSetChanged()

        Log.i( tag: "Size",  msg: "Meetings list size: ${meetingsList.size}")
    }
}
```

Logcat: Pixel 5 API 33 (emulator-5554) Android 13, API 33 — package:mine S

```
2023-10-02 16:18:49.902  9383-9383  e3marcogiacoppo          com.example.core3marcogiacoppo  W  Accessing hidden method Landroid/view/View;->computeFitSystemWindows(Landroid/g
2023-10-02 16:18:49.902  9383-9383  e3marcogiacoppo          com.example.core3marcogiacoppo  W  Accessing hidden method Landroid/view/ViewGroup;->makeOptionalFitsSystemWindows
2023-10-02 16:18:49.935  9383-9383  Size                     com.example.core3marcogiacoppo  I  Meetings list size: 30
2023-10-02 16:18:49.960  9383-9383  Compatibil...geReporter  com.example.core3marcogiacoppo  D  Compat change id reported: 210923482; UID 10179; state: ENABLED
2023-10-02 16:18:50.000  9383-9407  HostConnection           com.example.core3marcogiacoppo  D  HostComposition ext ANDROID_EMU_CHECKSUM_HELPER_v1 ANDROID_EMU_native_sync_v2 A
2023-10-02 16:18:50.002  9383-9407  OpenGLRenderer           com.example.core3marcogiacoppo  W  Failed to choose config with EGL_SWAP_BEHAVIOR_PRESERVED, retrying without...
2023-10-02 16:18:50.002  9383-9407  OpenGLRenderer           com.example.core3marcogiacoppo  W  Failed to initialize 101010-2 format, error = EGL_SUCCESS
2023-10-02 16:18:50.006  9383-9407  EGL_emulation            com.example.core3marcogiacoppo  D  eglMakeCurrent: 0xb4000072346863d0: ver 3 0 (tinfo 0x744f0bc080) (first time)
2023-10-02 16:18:50.016  9383-9407  Gralloc4                 com.example.core3marcogiacoppo  I  mapper 4.x is not supported
2023-10-02 16:18:50.018  9383-9407  Gralloc4                 com.example.core3marcogiacoppo  W  allocator 4.x is not supported
```

Launch succeeded (a minute ago)

Figure 6: Checking the size of the data

Open Issues and Recommendations

No issues.