Task

This is an individual task with a fixed deadline -- no submissions will be accepted after the closing date.

If Turnitin indicates 25% or higher similarity (or a teacher detects a similar level of similarity) with other published work and/or work submitted by other students, the submission will be reviewed and a mark potentially withheld or delayed. Similar sections will **not** be assessed, i.e., if the introduction is copied from elsewhere, the introduction criteria will be awarded 0.

- PDF: A completed report for this task -- the report should set out your development plan/time logs, the key design decisions made, issues encountered and the solutions considered/used, and a reflection on what worked well/what could be improved. Please ensure that explanation/decisions relating to the rubric elements are covered. Any use of genAI must be acknowledged, with prompts and outputs included in an appendix. If genAI was not used, the statement "No Generative AI tools were used for this task" must be included in the acknowledgement. Please refer to the Unit Outline for more details around resubmission/redo of work where the teaching staff feel that genAI has been inappropriately used.
- The report **must** contain a link to code in a repository visible to teaching staff on GitHub Classroom (see GitHub Classroom links for the invite link). Repos are not to be public, nor should tutors be added manually -- **the task will not be assessed until the repo is in a suitable location**.
- Before or after submission, a demo in class is required, otherwise a penalty will be applied. The demo will be assessed according to the Demo guidelines. Note that these will need to be spread over 2 weeks of classes; it is expected that students aiming for high marks will demo in the first week, as this will allow you to take on feedback ahead of submission. **Please note a demo is required for this assessment, otherwise a 0 will be awarded.**
- For this task, 1 video is required from those students who have completed an investigation. This is a 3 minute video explaining your experiment -- we will stop watching at 3 minutes precisely. This needs to be submitted to Assignment 3 -- investigation video.
- **An additional interview (either synchronous or asynchronous) may be required in weeks 14-15.**

Instructions

1. Start with planning and researching. This will include working through the tutorial exercises and quizzes. Outline and describe the major things you will need to do to create your app.
2. Develop an app that meets criteria as shown below.
3. The report needs to cover key knowledge gaps and process that was undertaken in this task, however note that this should be pitched at a fellow student (e.g., "Go to the Android website and download Android Studio" is not required). A reflection on Assignment 2 and any changes made for Assignment 3 would also be appropriate.
    - The focus for this report is on **why** particular choices have been made in the context of current practice, not just a description of **what** you have done.

    - Architectural designs (including diagrams as an appendix) are also expected.
4. Submissions must be uploaded as a single PDF.

The app idea

Your app must allow the user to keep track of something, that ideally is of interest to you. This could be:

- a list of games to play or books to read
- a list of locations to visit
- a list of tasks to undertake
- a list of exercises that have been done

Please run ideas by your tutor or the convenor.

The design

Before developing your app, a design needs to be worked out. This involves a vision, user stories, use cases, low and/or high fidelity prototypes.

The app requirements

- The app needs to contain at least one RecyclerView with some complexity (i.e., not just a single TextView in a row -- make use of icons, colours etc.).
- The data may be stored using files or databases (Room or Firebase), however full CRUD functionality will attract more marks.
- While there is no minimum expectation on the number of activities/fragments as in the past, a rough guide for a good/excellent level would be 3-4 novel activities or fragments of reasonable complexity (e.g., not just a single TextView, and not just the same activity copied 3 times with a different name; think about one activity/fragment that takes data in, another that doesn't, a RecyclerView by itself, a RecyclerView alongside other views, different row layouts etc. in terms of distinctly different activities/fragments). At lower levels, at least 2 activities/fragments with communication should be included.
- It is expected that apps include use of advanced components, such as fragments, view models, concurrency and Live Data.
- Note that while some concepts from A1 and A2 are not explicitly required for A3, they may still be useful for this task and could be implemented with justification.

Disallowed concepts for this task: Jetpack Compose is **can** be used for this task. Data storage must be either Room, Firebase or file/device-based; Amplify and other services are not permitted.