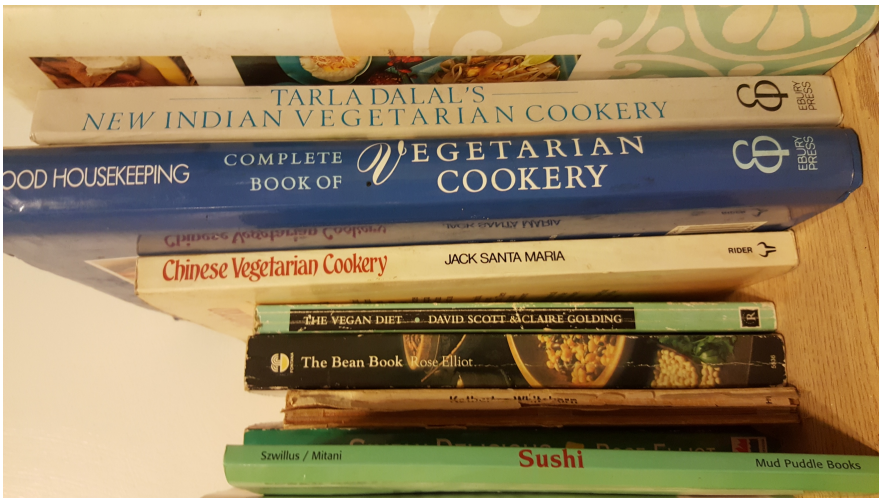
G53MDP Coursework 1-3 Recipe Book



Summary

In this exercise you are required to build an Android Recipe Book application. This is an assessed exercise and will account for **10% of your final module mark**. This is an individual coursework, and your submission must be entirely your own work – please pay particular attention to the section of this document regarding plagiarism. This document sets out general requirements and broad instructions for developing the application.

Your application should be submitted no later than:

- **3pm on Friday the 14th of December 2018**

Submissions should be made electronically via Moodle. Standard penalties of 5% per working day will be applied to late submissions.

Your application should be submitted as a .zip or .tar.gz file containing all relevant source code, configuration and related files, and a compiled .apk file – i.e. the contents of the directory containing your Android Studio project. Do not submit RAR files.

Specification

You should create an application with the functionality of a simple recipe book management system for Android, which allows the user to enter and store simple cooking recipes, and to browse and view recipes that have been previously stored.

Your application should consist of:

- An *Activity* presenting an interface for the user that allows the user to enter a new recipe, consisting of:
  - A large text field to enter recipe instructions
  - A text field to enter the title of the recipe

- An *Activity* that allows the user to browse all currently stored recipes by title, and to sort recipes by their rating or by title
- An *Activity* for a recipe that:
    - Displays the title, instructions of a recipe
    - Allows it to be given a rating score between 1 to 5
    - Allows it to be deleted
- A *ContentProvider* that provides storage for the recipes
    - With an appropriate *Contract class*

You must implement a **ContentProvider** component to handle the storage requirements of the application. You should make sure to use the ContentProvider when retrieving and adding data, rather than directly accessing the underlying storage from other components. There is no requirement that your ContentProvider is used remotely.

You should consider the following when implementing your application:

- Decomposition of the task into logical, discrete Activity components.
- Appropriate use of Activities, Intents and appreciation of the Activity life-cycle.
- Appropriate use of Widgets and ViewGroups for layouts that support devices of differing screen sizes and resolutions.
- Appropriate implementation of a ContentProvider component and Contract class, and appreciation of the ContentProvider life-cycle.
- Your application should have appropriate comments and variable / class names, so that a reader can easily understand how it works at the code level

Your application must be written in Java and make use of the Android SDK. There are no requirements to target a specific Android API version, however you can assume that your application will be tested on an emulated device (1080 x 1920 420dpi) running Android API version 28 (Android 9.0 Pie).

Plagiarism

N.B. Use of third party assets (tutorials, images, example code, libraries etc.) MUST be credited and referenced, and you MUST be able to demonstrate that they are available under a license that allows their reuse.

Making significant use of tutorial code while referencing it is poor academic practice, and will result in a lower mark that reflects the significance of your own original contribution.

Copying code from other students, from previous students, from any other source, or soliciting code from online sources and submitting it as your own is plagiarism and will be penalized as such. FAILING TO ATTRIBUTE a source will result in a mark of zero – and can potentially result in failure of coursework, module or degree.

All submissions are checked using both plagiarism detection software and manually for signs of cheating. If you have any doubts, then please ask.

Assessment Criteria

As this is a constrained exercise marks are awarded for achieving specific functionality as follows. For all elements either 0 or full marks are awarded as appropriate. There are no additional marks available for additional functionality in this exercise:

	Marks
The application has an Activity that allows the entry of a new recipe	2
The application has an Activity that lists all recipes	4
The list Activity can be sorted by title	1
The list Activity can be sorted by rating	1
The application has an Activity that displays an existing recipe	2
The display Activity allows a recipe to be rated	2
The display Activity allows a recipe to be deleted	2
The application has a ContentProvider that stores recipes	4
The ContentProvider has a Contract class that abstracts the database structure	2
Total	20

Hints

A common mistake is to not check that the database creation code still works after you've made revisions, as it is only executed on the first execution after installation, or when the database version is upgraded.

The database can be removed from the device by *uninstalling* your app to force the deletion of its associated internal storage. Do this via Settings->Apps->Your App Name->Uninstall on the phone, or alternatively long-pressing the app icon and dragging it upwards will reveal the trash can icon which, when the application is dragged onto it, will prompt confirming of the uninstallation of the app.

References

https://developer.android.com/guide/topics/data/data-storage

https://developer.android.com/guide/topics/providers/content-providers