CSCI3310 Mobile Computing & Application Development

Assignment 2 (Navigation and Dynamic Views)

CU Sweet Spot

Waffle at Café Tolo
captured by CPRO CUHK

Due : Mar 7, 2022 11:59pm

CUHK is not just having the largest number of canteens among universities in Hong Kong, many canteens on campus also provide many gourmet desserts. This assignment needs you to write an **Android** App which can keep and display a list of most liked desserts on our campus. The platform for implementation should be **Android Studio Arc Fox** please try to implement the program on **Android10.0 (API 29)**. The default testing simulator is **Pixel 4**.

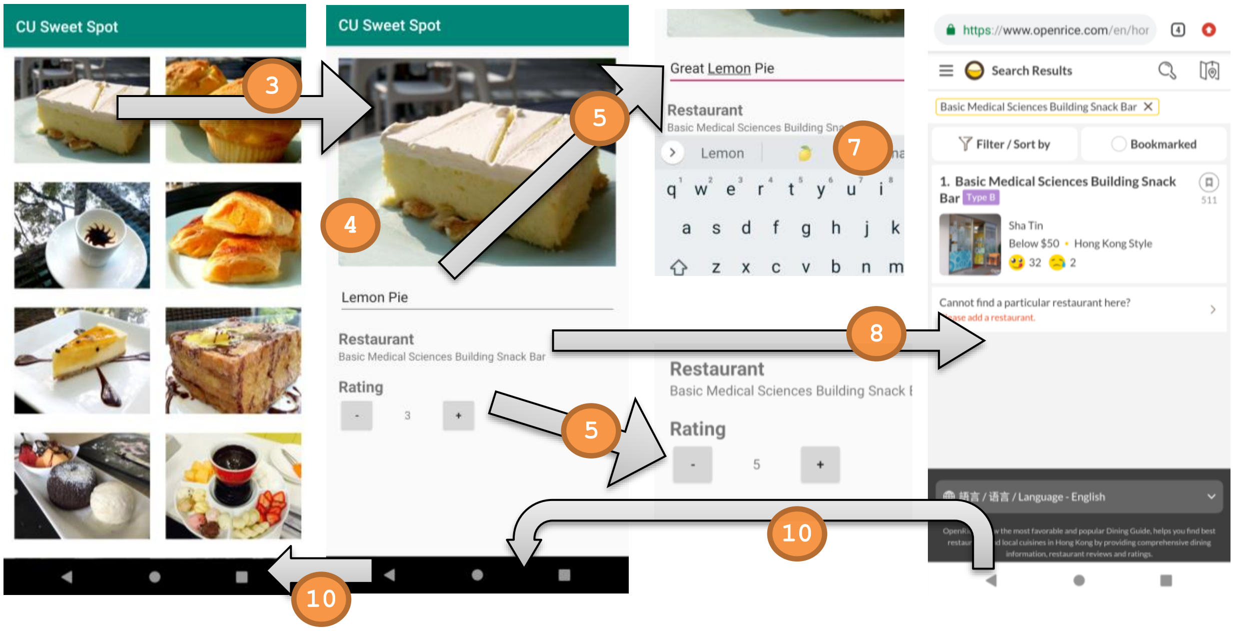
In this assignment, you need to set up a list of **10 sweet items** when the program starts up. As a practice of various programming skills in this assignment, you need to use a tabular format via *RecyclerView* to display a number of dessert images as a grid format (master view) on screen.

When the user taps on any of the image cell entries, a detail view will be shown. Each dessert item should consist of a thumbnail **photo**, the dessert **name**, the dessert **location** and a "**rating**" indicator. The user should be able to edit and AUTO-SAVE the **name** and **rating** entry. The program should also have a data persistence feature such that after the first run, the data will be saved in the app's local storage (shared preference or in a file). During the next run, the details which the user had modified will still be there.

Requirements

The functionalities and major steps on developing the app are as follows:

1. Download a Starter Project "*CU SweetSpot-Starter.zip*" from Background to start your work. Resources are included, in /raw/, text-based details of each dessert are given in a comma-separated value (CSV) file *cu_sweeties.csv*. Your program should be able read information from this file.
2. At program first start, your program should display the given thumbnail sweet images[†] included in the starter-pack. Your program should support scrolling via `RecyclerView` in a grid layout.
3. Your program should be able to transit to another details page (via an Intent) for clicking any image in the grid.
4. In the "Details" screen, the image should be on top while all text should be aligned to the left of the layout; the **sweet name**, its **restaurant** and **rating** should be shown below the sweet image.

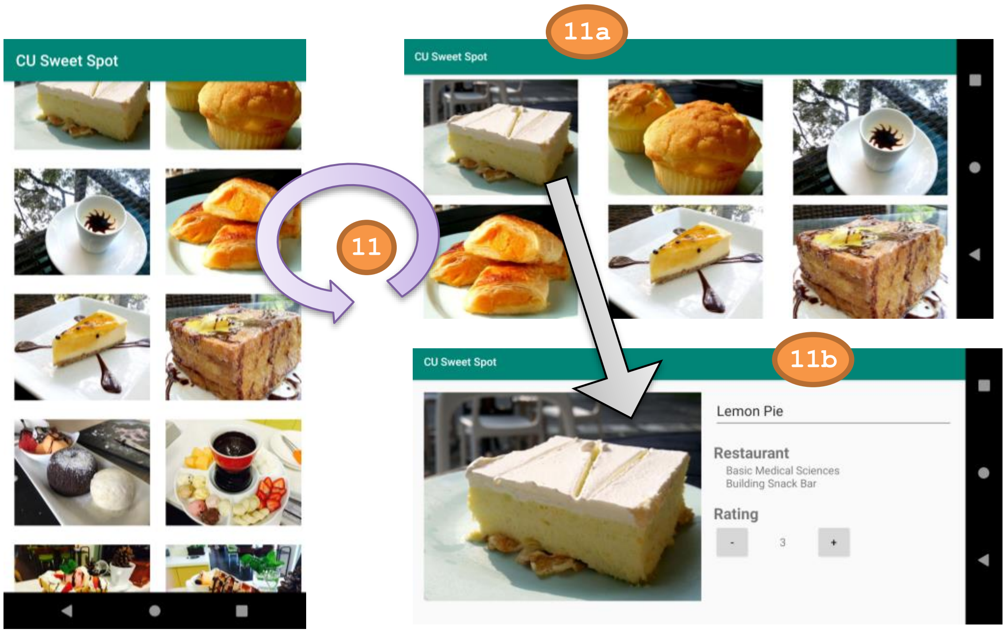


CU Sweet Spot
CU Sweet Spot
3
4
5
7
8
5
10
10
Lemon Pie
Restaurant
Basic Medical Sciences Building Snack Bar
Rating
-
3
+
Great Lemon Pie
Restaurant
Basic Medical Sciences Building Sna
Lemon
Restaurant
Basic Medical Sciences Building Snack I
Rating
-
5
+
https://www.openrice.com/en/hor
Search Results
Basic Medical Sciences Building Snack Bar
Filter / Sort by
Bookmarked
1. Basic Medical Sciences Building Snack Bar
Type B
511
Sha Tin
Below $50 • Hong Kong Style
32
2
Cannot find a particular restaurant here?
Please add a restaurant.
語言 / 语言 / Language - English

5. Besides showing details, your program should be able to let the user edit the **name**, and let the user update a **"Rating"** ranging from <u>1 to 5</u> (of a step size of a <u>one</u>), where initial ratings should be 3.
6. The result can be **AUTO-SAVED** after an edit is done and the updated details, including names or ratings, should be persisted through the program execution or re-launch.
7. Your program should be able to save sweet details including name, and rating. In the detailed view, `EditText` should be used to launch Android's default keyboard for updating information.
8. When the restaurant name is clicked, your app should be able to transit to a webpage at openrice.com (via an implicit Intent) given the "restaurant name" as the search term.

[†] ***NOTE:*** *All photos shot at CU are provided under the permission of* <u>*Communications and Public Relations Office (CPRO), CUHK*</u>*, please use only within this assignment*

9. Your program should be able to save the edited rating(s) into the app's local storage such as shared preference or local document directory*. In other words, if you have updated the rating of sweet(s) to the app and quit, the updated rating(s) will be there the next time you open the app.
10. User can go back (via Android's default **BACK** ◁ button) to choose another sweet.



11. On rotating the phone, the layout should be adjusted accordingly:
    a. In the grid view, number of columns is 2 in portrait mode while 3 columns should be used in landscape mode;
    b. In the details view, textual information should be displayed on the right of the image in landscape mode.

*Note: files stored under /res/ cannot be over-written. A temporary file could be created for updating via the Java standard I/O and placed e.g. under device's `data/data/`***app_name***`/files`

Submission

You should pack all your app folders and related files into an archive named "**3310_asg2.zip**".

Do packaging via Android Studio's main menu: **File** > **Export** > **Export to Zip File…** (this would include necessary all /java, /res and /Gradle files), and

Submit it into our assignment box in the Blackboard system before the deadline, **Mar 7, 2022, at 11:59pm**.

Late submissions will risk a score deduction of range between *10%* to *50%* if they are being done within 48 hours after the deadline. Submission later than **Mar 9, 2022,** won't be considered.

Grading Remarks:

1. Follow the project/package naming stated in the specification.
2. Put down personal information (Name and SID) in the .java.
3. The indentation of the code in the .java file should make the code easy to read. The .java file should contain comments to indicate computational logic.
4. The submitted code should be free of any typing mistakes, compilation errors/warnings under the specified API level.
5. The submitted app should be runnable at least on the virtual device stated in spec.
6. The program has to pass the corresponding test steps stated in the specification. In general, if you've followed the instructions above, you'll earn the vast majority of the points below.

   - **(10%) Basic** – does the project/package named correctly with personal particulars and properly styled in code?
   - **(20%) Running** – Does the app compile and run properly?
   - **(35%) View** – Is RecyclerView used properly? Are all updates can be done?
   - **(20%) Navigation** – Is the navigation correct?
   - **(15%) Data persistence** – Are update saved properly? Any state lost due to device config changes?

References

- Table | Android Developers
  https://developer.android.com/guide/topics/ui/layout/grid
- RecyclerViews | Android Developers
  https://developer.android.com/guide/topics/ui/layout/recyclerview
- Input events overview | Android Developers
  https://developer.android.com/guide/topics/ui/ui-events
- Data and file storage overview | Android Developers
  https://developer.android.com/training/data-storage
- View on-device files with Device File Explorer | Android Developers
  https://developer.android.com/studio/debug/device-file-explorer

Extensions

There exist quite several extensions you may work with in this assignment if you find it interesting. Here are some of the suggestions.

1. Gallery is now of fixed number of images and could be extended to provide function for adding or deleting image. Adding or deleting an image could be done by floating action bar (FAB) .
2. Extra sorting or filtering features, e.g. based on name, rating etc. can be added. Option Menu is one of the ways to control them.