College of Engineering, Construction and Living Sciences
Bachelor of Information Technology
IN721: Design and Development of Applications for Mobile Devices
Level 7, Credits 15

# Practical 07: Search View & Shared Preferences

## Assessment Table

| Assessment Activity | Weighting | Learning Outcomes | Assessment Grading Scheme | Completion Requirements |
|---|---|---|---|---|
| Practicals | 25% | 1, 3, 4 | CRA | Cumulative |
| Language Translator | 20% | 1, 3, 4 | CRA | Cumulative |
| Wishlist | 25% | 1, 3, 4 | CRA | Cumulative |
| Exams 1-5 | 30% | 2, 3, 4 | CRA | Cumulative |

## Conditions of Assessment

This assessment will need to be completed by Friday, 12 June 2020.

## Pass Criteria

This assessment is criterion-referenced with a cumulative pass mark of 50%.

## Submission Details

You must submit your program files via **GitHub Classroom**. Here is the link to the repository you will be using for your submission – https://classroom.github.com/a/ifyWTPlw. For ease of marking, please submit the marking sheet with your name & student id number via **Microsoft Teams** under the **Assignments** tab.

## Authenticity

All parts of your submitted assessment must be completely your work and any references must be cited appropriately.

# Policy on Submissions, Extensions, Resubmissions & Resits

The school's process concerning **Submissions, Extensions, Resubmissions and Resits** complies with Otago Polytechnic policies. Students can view policies on the Otago Polytechnic website located at https://www.op.ac.nz/about-us/governance-and-management/policies.

## Extensions

Please familiarise yourself with the assessment due dates. If you need an extension, please contact your lecturer before the due date. If you require more than a week's extension, a medical certificate or support letter from your manager may be needed.

## Resubmissions

Students may be requested to resubmit an assessment following a rework of part/s of the original assessment. Resubmissions are completed within a short time frame (usually no more than 5 working days) and usually must be completed within the timing of the course to which the assessment relates. Resubmissions will be available to students who have made a genuine attempt at the first assessment opportunity. The maximum grade awarded for resubmission will be C-.

## Learning Outcomes

At the successful completion of this course, students will be able to:

1. Implement complete, non-trivial, industry-standard mobile applications following sound architectural and code-quality standards.
2. Explain relevant principles of human perception and cognition and their importance to software design.
3. Identify relevant use cases for a mobile computing scenario and incorporate them into an effective user experience design.
4. Follow industry standard software engineering practice in the design of mobile applications.

## Assessment Overview

In this practical, you will complete a series of tasks covering today's lecture. This practical is worth 1% of the final mark for the Design and Development of Applications for Mobile Devices.

We are nearing the end of our small-medium sized application. Hopefully, you have been able to display an artist's albums in a recycler view & an artist's individual album data in a card view. Today, we will add search functionality to our application.

## Task 1

- Create two menu XML files in the **menu** resource directory. Call your menu files **menu_main.xml** & **menu_search.xml**.
- Make sure you add the **actionViewClass** attribute in the **menu_search.xml**. This will allow you to implement the action of a search view on that specific menu item.
- **Note:** showAsAction="ifRoom" means that if there is room on the toolbar, show that item. There may be times where you need four or five menu items. The toolbar may not be be appropriate. This is where you should consider a navigation drawer.
- Inflate the **menu_main.xml** in **MainActivity**. The implementation is same as the previous practicals.
- Create a **searchable.xml** file. You will need to create a **xml** resource directory. It is important to have both the **hint** & **label** attributes. It tells the user the search view's expected value.
- Add the **intent-filter** & **meta-data** tags to the **SearchActivity** tag in the **AndroidManifest.xml** file. This references the **searchable.xml** we created earlier.

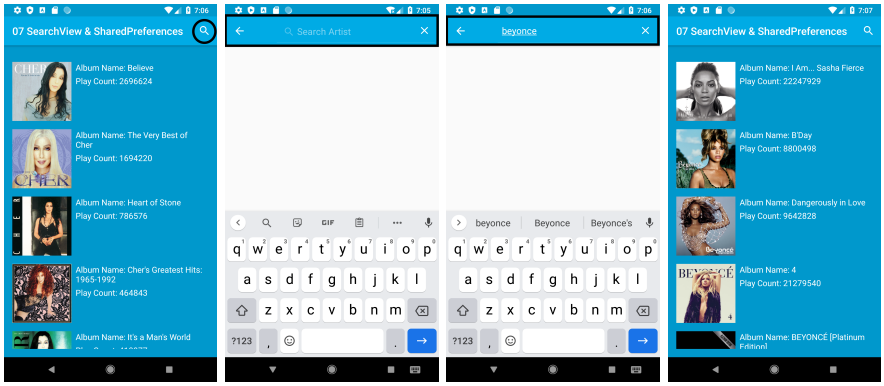
## Task 2

- Create a new basic activity called **SearchActivity**. If you choose to create an empty activity, you will need to add a toolbar to your activity XML file. If you don't, later on when you implement **BaseActivity**, you will encounter a null exception because it is looking for a reference to a toolbar.
- Make sure your activity is added to the **AndroidManifest.xml** file. Add the appropriate **meta-data** tag for using the back arrow button on the toolbar. **android.support.PARENT_ACTIVITY** sound familiar?
- Implement the **SearchActivity** as describe in the lecture slides. **Please comment as you go!!! You won't learn from copying & pasting**.

## Task 3

- In the **SearchActivity**, we want to be able to save the query from the search view. The reason why we are using shared preferences is because the data is persistently saved when the application is terminated. If we don't store the search view's value, it will default back to the original value.
- The **album_query** key/value will be saved in shared preferences. **Note:** applicationContext refers to **this** activity.
- Add the **onResume** override function to the **MainActivity**. This function will get the **album_query** key/value from the shared preferences. The value from **album_query** will be used instead of a hard coded artist. This is then used by the async task to download the data from the Last.fm API.

## Expected Output




## Submission

- Create a new branch named 07-checkpoint within your practicals GitHub repository
- Create a new pull request and assign Grayson-Orr to review your submission
- Deadline: Friday, 12 June at 5pm

**Note:** Please don't merge your own pull request.