Learning summary report template

Marco Giacoppo - 104071453

Overview

Throughout the semester, I set out to learn about the intricacies of mobile application development. The main takeaways, difficulties, and learning objectives from this unit are summarised in this report. I will give an overview of the things that have influenced my learning journey in this section and lay the groundwork for the following sections of this report.

To better understand the unique environment of mobile application development, I engaged with a variety of assignments and projects throughout my journey. Also, I had the chance to work on a unique mobile app, which gave me the chance to put my knowledge to use in the real world and come up with insightful new ideas. I've created a movie rating and reviewing app with research on how to optimize API fetching using Coroutines.

Evidence (in Portfolio Pieces)

I have completed the following assignments.

- All Core outcomes
- All Extension tasks.

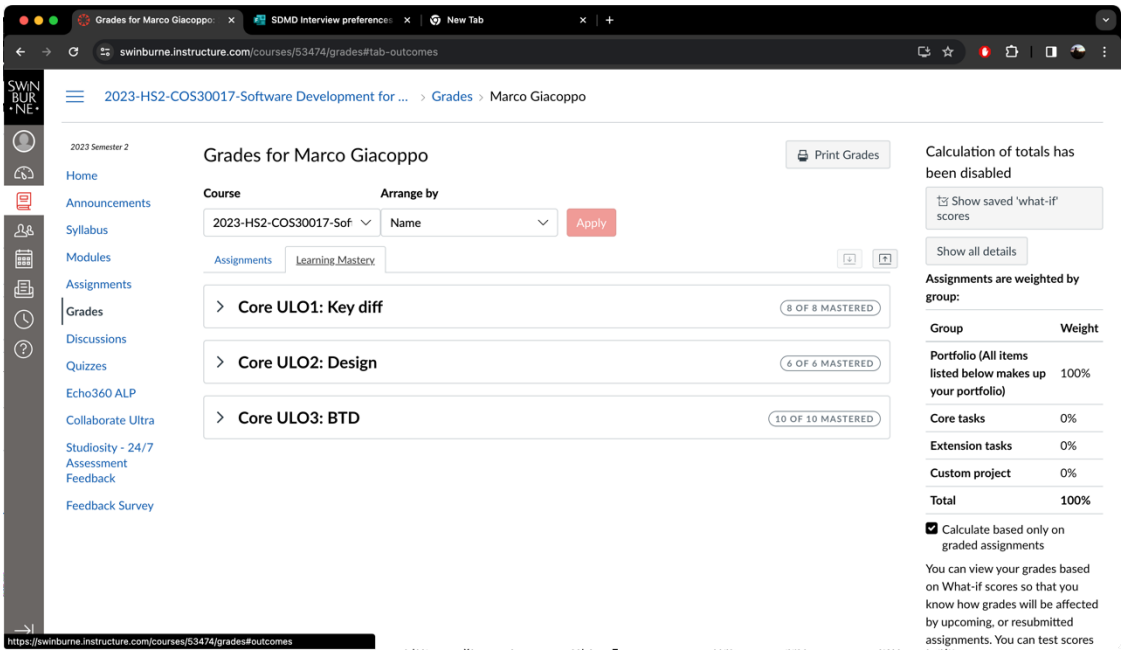

Figure 1: All Core and Extension Completed.

- A project at Level 1
- An app at Level 2 (including level 1)
- An extended app or project at Level 3 (including levels 1 and 2)

Learning Summary

ULO 1: Explain the key differences between development of systems to run on mobile devices and on typical personal computing or internet-based environments and apply this knowledge in the design of mobile device software.

- **Assignment: Core 1:**
    - Handling the saving of the app's state on rotation; this is the process of preserving and restoring the app's current data configuration from portrait to landscape or vice versa.
    - Incorporating localization; this involves adapting the application to support multiple languages and regions.
    - Using linear and constraint orientation. I used linear for portrait just because it made more sense to use linear in a simple straight-line design.
    - Writing/using appropriate listeners; as a core functionality of the task, listeners were used in all the buttons (Add, Subtract, Reset, and Roll).

- **Assignment: Core 2:**
    - Understand intents; in this task, the intent was used to navigate between screens and performing actions.
    - Toasts and Snackbars; I decided to use Toast because I think it's easier to understand. I used toasts whenever I need to tell users about what the app requires or doing now.
    - Implement validation for user's input.

- **Assignment: Core 3:**
    - Understand file system structure.

ULO 2: Design effective applications for a mobile device by taking into consideration the underlying hardware-imposed restrictions such as screen size, memory size and processor capability.

- **Assignment: Core 1:**

    - Multiple layouts for landscape and portrait. This is done so that users can use the app on either vertical or horizontal.
    - Using linear and constraint layouts.

- **Assignment: Core 2:**
    - Sketch and implement simple screens. Here I made a sketch on how I want the app to be. The screen size of a phone compared to desktop is a big difference, therefore an initial design on how the app looks like would be logical.
- **Extension: UX design:**
    - Investigating on the common patterns that are being used for android development.
- **Extension: Performance:**
    - Testing on the best methods of a fixed resource.

ULO 3: Build, test, and debug graphical applications for mobile devices by using the standard libraries that are bundled as part of the developers' toolkit for the mobile device.

- **Assignment: Core 1:**
    - Using logs to check if the state is correct before moving forward.
- **Assignment: Core 2:**
    - Unit and UI tests; using test to check if everything is doing what it's supposed to do. Such as, when users borrow for 7 days, the total should be (initial price x 7)
- **Assignment: Core 3:**
    - Use Logs. Checking the size of the data being passed.

Challenges in Mobile Development

We must concentrate on a few key areas when developing mobile applications. Here are a few instances of them:

1. User Interface (UI) Design:
    - Developing a beautiful, intuitive user interface for the app that offers a unique experience.
2. Functionality:
    - Ensuring that the app's essential features and functions operate faultlessly.
3. Security:
    - Safeguarding user information, preventing unauthorised access, and encrypting data transfers to and from the app's server.
4. Screen Size Compatibility:

- Android devices have screens of all shapes, sizes, and resolutions, from tiny smartphones to sizable tablets and even foldable gadgets. The app should modify its design and content to fit various screen sizes.

5. Performance:
   - Improving the app's resource usage efficiency and responsiveness for users.

Some aspects I found challenging are:

- **Fragmentation**
  Android devices comes in various screen sizes, resolutions, and hardware capabilities. This makes it challenging to create a consistent user experience across all devices, and time consuming.
- **Version Compatibility**
  Android versions vary, and not all users have the latest versions. As I experienced, higher android API versions works way faster than lower versions. The problem is that not everyone has the latest versions. So, this is also one of the challenges I face.
- **Performance Optimization**
  Achieving optimal performance on Android is very challenging due to differences in device capabilities. Performance optimizing is tricky as I know. I've tried to optimize my custom app in every way possible, from using coroutines, integrating an external database, to network request optimization.

Assumptions and Expectations

We must consider platform-specific guidelines, APIs, and user interface conventions when creating mobile apps, particularly for Android. Unlike more general software development, where we might focus on a variety of platforms or web applications, this is different. Mobile devices also have limited resources (e.g., CPU, memory) compared to desktop environment. Managing these constraints and optimizing performance is a unique challenge in mobile development. The diversity of Android devices and operating system versions are also a significant challenge. We need to ensure compatibility across various screen sizes, hardware capabilities, and Android OS versions.

Android development made me learn a new platform-specific languages like Kotlin, as well as understanding Android's unique development paradigms. I've also gained a deep understanding of UI/UX design and performance optimization. My prior knowledge before learning Android development was mostly in creating websites. Learning this new knowledge was a very fun

experience for me. I've dived into designing interfaces, learning about data structures, design patterns, etc.

Explorations

I've gained a lot of new knowledge while creating my custom app. I've tried using coroutines and asynchronous programming. I used this method to make my app more responsive and efficient, and ensuring that time-consuming tasks don't block the main thread, leading to a smoother user experience. I've also integrated Firebase for user authentication and database operations. This provides scalable and efficient solutions for managing user data and movie ratings. I've used Glide to efficiently load images and caching. This optimization improves the app's performance by reducing the need for repeated network requests for the same images.

I've also participated in the custom app discussion where I posted my own video and commented and gave feedback to others. There are quite a lot of different ideas posted there, some created a to-do list, financial management app, even a music player. I've given me feedbacks on some of this including the management app, where I gave idea where he could improve his UX usage.

My plan after finishing the unit is to probably continue with my custom app. I'm planning to upgrade my custom app by including even more layouts and extend the functionality of the app. I'll probably try figuring out even better ways to optimize my app and maybe someday in the future I can publish my app into Play Store. Another thing I want to learn is to extend my knowledge in mobile development, which is doing development in IOS. I heard that it's more and less the same but using a different language.

Final Words

This unit is one of my favourite one so far. I've learned a lot about mobile development and the paradigm. I'm very happy that I decided to take on the custom project, which makes me gain more interest in this unit. The IDE's design makes me keen to learn more, the way we write the codes, design the layouts, etc. There's still a lot of areas that I haven't explored, and I most certainly will dive deeper into this. Thank you for the experience!