

Learning summary report

Trac Duc Anh Luong - 103488117

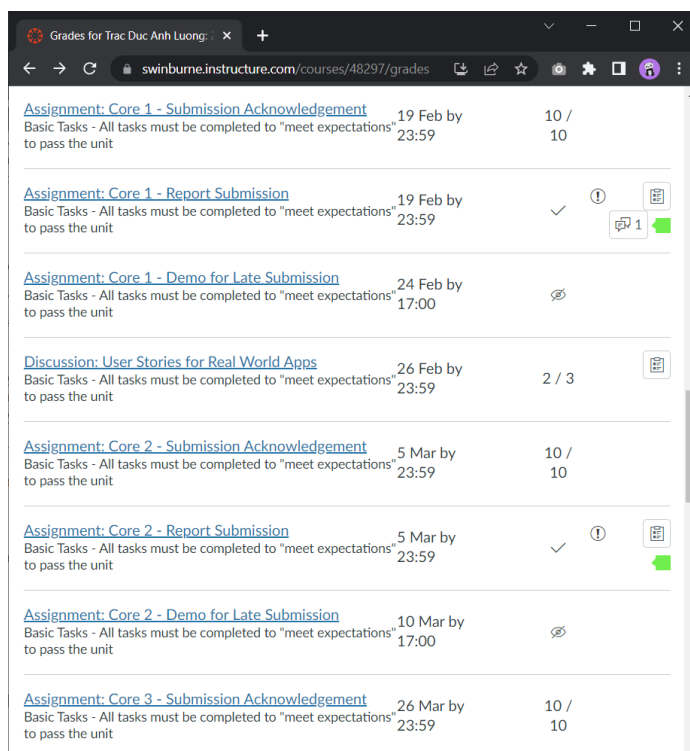
Overview

In this unit, I have learnt the core elements of software development and UI and UX design for mobile applications. Throughout the course, some key concepts and skills were introduced. First, there are differences between mobile devices' operating systems in comparison to our personal computers and online environments, and awareness of this fact can be applied to the mobile software development process. Second, mobile devices have constraints of their own in terms of screen real estate, memory size, and processing power, therefore thorough considerations are needed to create efficient applications. Third, testing and debugging are essential phrases in the mobile development lifecycle, therefore we need to integrate standard libraries and toolkits for these tests.

Evidence

I have completed the following assignments:

- All Core tasks
- All Extension tasks



The screenshot shows a web browser window with the address bar displaying 'swinburne.instructure.com/courses/48297/grades'. The page lists several assignments with their titles, due dates, and completion status. The assignments are:

Assignment Title	Due Date	Status
Assignment: Core 1 - Submission Acknowledgement	19 Feb by 23:59	10 / 10
Assignment: Core 1 - Report Submission	19 Feb by 23:59	✓
Assignment: Core 1 - Demo for Late Submission	24 Feb by 17:00	🔒
Discussion: User Stories for Real World Apps	26 Feb by 23:59	2 / 3
Assignment: Core 2 - Submission Acknowledgement	5 Mar by 23:59	10 / 10
Assignment: Core 2 - Report Submission	5 Mar by 23:59	✓
Assignment: Core 2 - Demo for Late Submission	10 Mar by 17:00	🔒
Assignment: Core 3 - Submission Acknowledgement	26 Mar by 23:59	10 / 10

Figure 1: Completed Core tasks

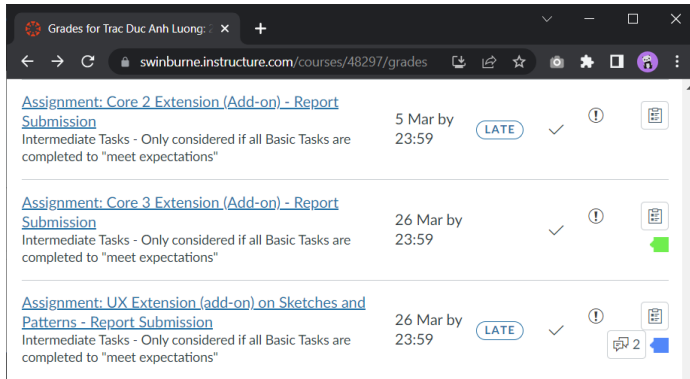


Figure 2: Completed Extension tasks

Learning Summary

ULO 1: Apply the understanding of designing a mobile device software by outlining the fundamental distinctions between developing systems to operate on mobile devices and developing systems to run on standard personal computers or internet-based environments.

The tasks that are related to this ULO were:

- Core Task 1:
 - Incorporate localisation into the application, and create an app that can support multiple languages and provide the same functionalities. This can be done in web applications through plugins or libraries.
 - Listeners: The use of listeners in mobile development is somewhat similar to that of web apps that consist of the use of JavaScript.
 - Saving the program state upon rotation.
- Core Task 2:
 - Text input validation as errors can occur when the users enter information.
 - Implementation of snack bars and toasts.
 - Using UI Widgets.
 - Using Add-ons.
- Extension Task 3:
 - Understand file system structure and permissions.
 - Implement features like saving the last click.
 - Use Fragments to display the bottom sheet dialogue.

ULO 2: Consider the limitations imposed by the underlying hardware, such as screen size, memory size, and processing capability, while designing efficient apps for a mobile device.

- Core Task 1:
 - Multiple layouts for landscape and portrait so that the user can use the app in multiple orientations of the screen. This can be compared with screen liquidity in web development, however, web development focuses mainly on setting a

threshold width for the web to know when the screen should be rearranged for smaller screen estate - e.g. mobile devices.

- Core Task 2:
 - Create sketches for new layouts: The screen size is small, therefore standard layouts and principles around them are a must to create a logical app display.
 - Data communication: Parsing data through ActivityResultContracts and Intents.
- UX Extension: Figure out common patterns that exist among software applications to support the limited amount of screen space.
- Performance Extension: Figure out the best methods in terms of static approaches in comparison to concurrent and dynamic approaches when generating new data.

ULO 3: Use the standard libraries that come with the mobile device's developer tools to create, test, and debug graphical applications for mobile devices.

- Core Task 1:
 - Logging usage: Display different states of the program using messages in Logcat.
 - Using the IDE built-in feature like debug and build.
 - Apply the given code for application testing from the GitHub repository.
- Core Task 2:
 - Use high-level features in the Android Studio IDE to test and debug the program.
- Extension Task 2:
 - Write a reusable test script for the program.
- Extension Task 3:
 - Use commands of build, test and debug from the IDE.

Challenges in Mobile Development

There are key areas that we need to consider when it comes to mobile development, some of which are:

1. UI and UX design: An app that contains an appealing graphic and approachable interface for users from different backgrounds will achieve more success in comparison to those who do not. In the designing phase, consider the attributes of screen size, resolutions, etc.
2. Processor performance: Any application should be loaded quickly, given certain thresholds to the user's patience. The app should also be responsive when receiving inputs from users and preserve the device resources, including CPU and RAM.
3. Security: Data privacy has always been an issue for app developers. Personal data from users is always a must, with security approaches including encryption, user authentication, secure network connection, etc. This is considered one of the most challenging aspects when it comes to software development for mobile devices.
4. Compatibility: There is a wide range of devices that come in different screen resolutions, sizes, shapes, and processor specifications. Therefore, we need to ensure our application compatibility across these devices.

5. Testing and debugging: Any technical development requires the product to be thoroughly tested before moving from the development process to deployment and bringing the product, in this case, our mobile apps to the users.
6. Maintenance: In the development process, our app needs to be designed in a way so that its components are easily maintained and scaled after each patch and update.

Assumptions and Expectations

Mobile development shares some similarities but is also different from a field that I have learnt before, which is web development. Mobile applications are designed for a specific operating system, either IOS or Android. Web applications, on the other hand, can be accessed by any device that has a browser and internet connection, therefore the flexibility is better, however, web apps may not be as optimized for an environment as mobile apps. Mobile apps also have access to the machine's hardware, including the camera, microphone, etc. Web applications have limited access to these sensors. The accessibility is also different, as mobile applications have to be downloaded from the app store, while web applications can be accessed from any browser. In short, mobile development will focus on user experience and platform optimization while web apps focus on accessibility and the range of devices.

Software development and development share common features but also differentiate from each other. The aspects vary from platforms, UX, native vs web apps, hardware acceleration and access, installation, accessibility, etc. Mobile applications are designed for a specific platform with considerations for the user experience, with direct accessibility to the hardware, while software has less hardware access and is commonly built using a computer programming language or some of today's web technologies. The development of a mobile app is also more costly as they have a longer development cycle on various platforms in comparison to software in general. These assumptions have been reinforced throughout the learning process in this unit.

Explorations

After completing this unit, I plan to expand my knowledge in some of the following topics:

1. Cloud technology: integrate cloud-hosted databases, and storage into mobile applications
2. Cross-platform software: After learning Android, I hope that I can find out more about cross-platform frameworks like React Native or Flutter which can run on both IOS and Android.
3. Security measures: Implement security practices for mobile applications like multi-layer authentication and access control.

These topics can be easily found on the internet through online tutorials and courses.

Final Words

The most useful thing that I have learnt from this unit is the skills to develop and deploy software for mobile devices. The content has covered most of the development process, from UI and UX

design to advanced topics like optimization for the device's resources. Advanced features of an IDE like Android Studio were also covered. Overall, I have also learnt about problem-solving, reading and finding documentation. I believe this useful knowledge can be applied to different use cases, from my projects to professional contexts.