Learning summary report

Huy Vu Tran - 104177995

Overview

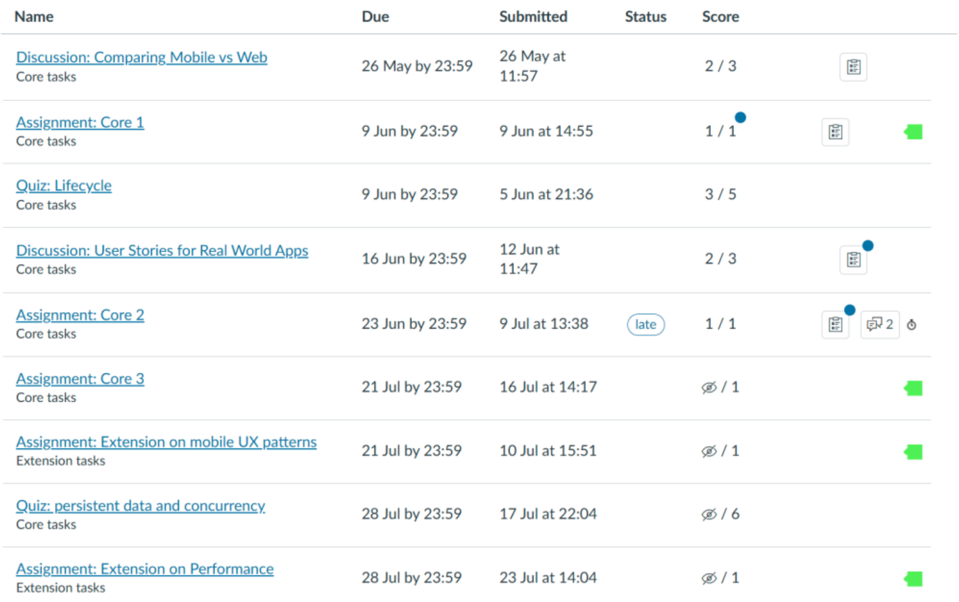
I have gained knowledge of the fundamentals of both software programming and user interface and user experience design for mobile apps in this unit. The distinct hardware limitations of mobile devices—such as constrained memory capacity, processor speed, and screen area—require careful thought in order to develop effective apps. The importance of testing and debugging as critical stages of the mobile application lifecycle was also underlined. To guarantee functionality and dependability, standard libraries and resources must be included.

For my custom application project, I developed a fitness app that allows users to manage their workout routines and track their fitness progress. The app features functionalities such as adding, editing, and viewing exercises, updating profile information with image uploads, and selecting workout plans based on fitness levels.

Evidence (in Portfolio Pieces)

I have completed the following assignments.

- All Core outcomes
- All Extension tasks
- A project at Level 1
- An app at Level 2 (including level 1)



Name	Due	Submitted	Status	Score		
Discussion: Comparing Mobile vs Web Core tasks	26 May by 23:59	26 May at 11:57		2 / 3		
Assignment: Core 1 Core tasks	9 Jun by 23:59	9 Jun at 14:55		1 / 1		🟩
Quiz: Lifecycle Core tasks	9 Jun by 23:59	5 Jun at 21:36		3 / 5		
Discussion: User Stories for Real World Apps Core tasks	16 Jun by 23:59	12 Jun at 11:47		2 / 3		
Assignment: Core 2 Core tasks	23 Jun by 23:59	9 Jul at 13:38	late	1 / 1	2	
Assignment: Core 3 Core tasks	21 Jul by 23:59	16 Jul at 14:17		⊘ / 1		🟩
Assignment: Extension on mobile UX patterns Extension tasks	21 Jul by 23:59	10 Jul at 15:51		⊘ / 1		🟩
Quiz: persistent data and concurrency Core tasks	28 Jul by 23:59	17 Jul at 22:04		⊘ / 6		
Assignment: Extension on Performance Extension tasks	28 Jul by 23:59	23 Jul at 14:04		⊘ / 1		🟩

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.

- ***Core 1 : Taking Chances***
    - Incorporate localisation into the application, ensuring the app is usable in multiple languages.
    - Implement listeners to handle button presses and update UI elements dynamically, similar to event handling in web applications.
    - Save the program state upon rotation to maintain user data and app state.
- ***Core 2: Sharing is Caring***
    - Use intents to share data between activities, highlighting the differences in data handling between mobile and typical PC environments.
    - Implement Parcelable protocol for efficient data transfer between activities.
    - Design a multi-activity app with images and advanced UI widgets to manage transitions and data sharing effectively.
- ***Core 3: The Clubhouse***
    - Read data from a file and display it in a RecyclerView, showcasing efficient data handling.
    - Implement filtering and sorting of the list to manage data presentation and user interaction.
    - Apply differential colouring or image selection for list items to enhance user experience.

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.

- ***Core Task 1: Taking Chances***
    - Implement a dynamic UI that updates elements like buttons and scores efficiently, taking into account screen size and memory constraints.
    - Ensure the app performs well in different orientations and languages, addressing hardware restrictions.
- ***Core Task 2: Sharing is Caring***
    - Optimize image sizes and use Parcelable for data sharing to manage memory and processor load.
    - Design UI elements and transitions to be smooth and responsive, considering the limitations of mobile hardware.
- ***Extension Task: Performance***
    - Profile app performance with different icon loading strategies to understand and optimize memory usage and processor capability.
    - Experiment with constant icons, generated icons on bind, and generated icons on initialization to find the most efficient loading strategy.

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.

- ***Core Task 1: Taking Chances***
  - Use Android's standard libraries and tools to build and debug a single-activity game app.
  - Implement listeners and log messages for testing and debugging purposes.
- ***Core Task 2: Sharing is Caring***
  - Utilize standard Android libraries for UI design and testing, ensuring robust handling of activity states and data sharing.
  - Include UI tests to validate the app’s functionality and performance.
- ***Core Task 3: The Clubhouse***
  - Leverage standard libraries for file handling, UI components, and data manipulation to build a complex mobile application.
  - Ensure efficient performance and user experience through proper use of RecyclerView and data filtering.
- ***Extension Task: Performance***
  - Use Android Profiler to identify and address performance bottlenecks.
  - Test and debug the app under different icon loading scenarios to ensure optimal performance.

Challenges in Mobile Development

Here are the aspects I found particularly challenging or different from my expectations:

1. **User Interface (UI) Design**: It turned out to be more difficult than expected to create an aesthetically pleasing and intuitive interface that functions well in a variety of screen sizes and orientations.
2. **Performance Optimization**: When it comes to processing power, memory, and battery life, mobile devices are inferior to desktop PCs. One major difficulty was to optimize application performance to guarantee seamless operation without using excessive resources.
3. **Testing and Debugging**: In order to produce an application of the highest caliber, thorough testing and debugging are essential. This method was made especially difficult by the large range of Android devices and OS versions.
4. **Integration with Cloud Databases**: There were particular difficulties in linking the program with online databases to provide data synchronization and storage. It was more complicated than anticipated to handle offline access, ensure safe and effective data transmission, and keep data consistent across devices.

Assumptions and Expectations

Developing mobile applications presents unique challenges compared to web development, particularly due to the fragmented ecosystem of devices with varying screen sizes, resolutions, hardware capabilities, and operating system versions. This necessitates extensive testing and optimization to ensure a consistent user experience. UI design in mobile development is more constrained, requiring intuitive and visually appealing interfaces that function well on different screen sizes and orientations, adhering to platform-specific guidelines like Android's Material

Design. Performance optimization is also more critical in mobile development, involving efficient memory management and minimizing background processes to ensure smooth operation without draining resources.

Testing and debugging in mobile development are more complex due to the wide variety of devices and operating systems, requiring a systematic approach to simulate real-world scenarios. Integration with cloud databases introduces additional challenges, such as handling intermittent network connections and maintaining data consistency across devices. My experience reinforced the importance of understanding platform constraints and optimizing accordingly, while challenging the assumption that mobile UI design would be straightforward. Overall, mobile development demands specialized skills and approaches to address its unique requirements, enhancing my ability to adapt development practices for each platform.

Explorations

For my custom application project, I developed a fitness app that integrates with Firebase Firestore, a NoSQL cloud database, to manage user data and workout routines. The app allows users to add, edit, and view exercises, update their profile information, and select workout plans based on their fitness levels. I explored advanced Firestore features such as transactions for atomic operations, efficient data modeling, and security rules to protect user data. Additionally, I provided feedback on peer projects, which broadened my understanding of different development approaches and user experience design.

Looking ahead, I plan to delve deeper into advanced Firestore features, Firebase Cloud Functions for server-side logic, and integrating machine learning models with Firebase ML Kit for personalized workout recommendations. My learning activities will include online courses, reading documentation, and engaging with developer communities to stay updated with new features and best practices.

Final Words

The most useful takeaway from this unit is the deep understanding of how to design and optimize applications for mobile devices, considering their unique constraints and diverse ecosystem. This includes mastering the intricacies of UI design for different screen sizes, optimizing performance for limited hardware resources, and implementing effective testing and debugging strategies. Additionally, gaining practical experience in integrating mobile applications with cloud databases has equipped me with valuable skills to tackle real-world development challenges. This comprehensive knowledge will greatly enhance my ability to create efficient, user-friendly mobile applications in the future.