Lappeenrannan teknillinen yliopisto
School of Business and Management

Sofware Development Skills

Yiran Liu, 000972772

1 LEARNING DIARY, SOFTWARE DEVELOPMENT SKILLS: MOBILE 2025-26

LEARNING DIARY

Course Selection Motivation

I chose this course because I have a strong interest in programming. During my mandatory courses, I achieved full marks in both MATLAB and Python. Later, I self-studied C and web programming, which I found very enjoyable. When I looked through the list of elective courses at LUT, I discovered this module and decided to take it as an opportunity to further develop my skills and apply them to a real project.

Project Topic Selection

For my project, I decided to create a piano-themed Android app. I have a strong personal interest in piano and own a digital piano at home, which made it a natural choice. The course allowed us to develop a mobile application, so I combined my programming skills with my passion for music. The project turned out to be very successful, and I was able to implement multiple functionalities including practice tracking, recent activity, statistics, and a virtual piano.

Course Feedback

Overall, the course was very engaging, but some of the materials were somewhat outdated. For example, the instructional videos on YouTube were from 8 years ago, and many of the tutorials used Java 8 syntax. Now, with Java 25 and Kotlin syntax, there are significant differences, which made parts of the learning process more challenging. Despite this, I found the course highly enjoyable and believe I will continue to apply the knowledge gained here in my future studies. Thank you for this course also thank you for reading my texts.

The diary is starting in the next page.

10.09.2025

I installed Android Studio but faced memory issues. I adjusted the IDE settings to allocate more memory and managed to launch the first empty project successfully. I started exploring the interface and checked how Gradle builds the project.

I learned how to assign names, IDs, and hints in XML layouts. I noticed that some layouts did not display correctly, so I manually adjusted layout files and experimented with switching between layout.xml templates.

11.09.2025

I started experimenting with Kotlin and Java together. I explored the differences between them and realized that Kotlin is more concise, but some older Java examples in tutorials do not work directly. I also tried composing a simple “Calculator” app to understand event handling and component activities.

I encountered the “R unresolved reference” issue. After researching online, I learned that restarting Android Studio and running Gradle commands (./gradlew clean, ./gradlew assembleDebug) can fix many build errors.

13.09.2025

I realized that inconsistent package names (e.g., package com.mentor.myapplicationsecond) could cause the app to crash. I corrected the package declaration and tested the app successfully.

14.09.2025

I accidentally triggered unwanted behavior by double-clicking Shift. I learned to disable this in Advanced Settings. I also studied how to configure the manifest properly for intents, including opening URLs from the app using:

```
<queries>
    <intent>
        <action android:name="android.intent.action.VIEW" />
        <data android:scheme="http" />
```

```
    </intent>
    <intent>
        <action android:name="android.intent.action.VIEW" />
        <data android:scheme="https" />
    </intent>
</queries>
```

15.09.2025

I practiced creating ListViews with images and text, paying attention to correct item descriptions and layout structure. This helped me understand data binding and adapters in Android.

I reviewed Java vs Kotlin differences, focusing on syntax, null safety, and concise code structures. I became more confident in resolving errors and understanding the Android lifecycle.

15.09.2025

I continued experimenting with intents and external apps. I learned to open web pages, YouTube videos, and other apps from my own app. I also fixed remaining layout issues and ensured all buttons and views responded correctly.

16.09.2025 – 26.09.2025

Project Development Diary

Project Planning & Initial Testing

Today I focused on planning the structure of my Piano Trainer Plus app and started implementing core features. I created the initial submenus and buttons in the main activity and began testing the navigation flow.

Issues Encountered

Submenu Not Opening:

When I clicked on buttons to open submenus, the app either crashed immediately or got stuck in a long wait. After careful investigation, I realized that I had forgotten to declare

some activities in the AndroidManifest.xml. Adding the missing activities solved the crash issue.

Intent Queries for External Apps:
Some features, such as opening YouTube or other web pages, did not work because I had not properly defined <queries> in the manifest for URL schemes. I added the following block to allow the app to open external URLs:

```
<queries>
    <intent>
        <action android:name="android.intent.action.VIEW" />
        <data android:scheme="http" />
    </intent>
    <intent>
        <action android:name="android.intent.action.VIEW" />
        <data android:scheme="https" />
    </intent>
</queries>
```

View ID Confusion in a Large Project:
While working on multiple activities and layouts, I found it easy to mix up view IDs when using findViewById. I spent some time double-checking IDs in activity_*.xml files and making sure they matched in the Kotlin code.

Title Design Problem:
I initially wanted to use artistic text for the app's title, but implementing dynamic custom fonts and painting text on a canvas involved complex code. As a simpler alternative, I decided to use a static image for the title. This solution was straightforward and visually appealing.

Audio Files and Piano Notes:
Finding suitable piano audio files took a lot of effort. I initially considered recording my own piano at school, but this was impractical. Eventually, I purchased high-quality audio samples online.

Audio Loading Issue:
At first, I stored the audio files in the drawable folder, but they could not be loaded with MediaPlayer. After researching, I learned that audio files must be placed in the res/raw folder in MP3 format to be accessed correctly. After moving the files, I successfully loaded and played them in the app.

Reflections & Lessons Learned

Careful planning of the manifest and activity declarations is crucial for avoiding crashes when navigating between submenus.

Understanding Android file structure, especially where to place media resources (drawable vs raw), saves a lot of debugging time.

For UI design, sometimes a simpler solution like using images is better than implementing complicated custom drawing, especially for a single-person project.

Large projects require strict naming conventions and double-checking of IDs to avoid errors and confusion.

28.09.2025

I consolidated all the lessons learned and successfully built a small project (Piano Trainer Plus) using multiple views, ListViews, MediaPlayer, and Intents. I documented all issues, fixes, and improvements made during development.

Thank you for this course content. Thank you.