Lappeenrannan teknillinen yliopisto
School of Business and Management

Hung Nguyen 001736166

LEARNING DIARY

Software Development Skills: Mobile 2023-24

1 LEARNING DIARY PART 1: BUILDING MY FIRST ANDROID APP WITH ANDROID STUDIO

Today, I dove into the "Android Studio For Beginners Part 1" tutorial and learned how to set up and build my first Android app using Android Studio. The session was both informative and hands-on, guiding me through every step of the process.

1.1 Prerequisites

Before starting, I ensured I had basic knowledge of Java, Java installed on my computer, and Android Studio properly set up. These prerequisites are crucial for following along with the tutorial smoothly.

1.2 Setting Up My Project

I launched Android Studio and followed the setup wizard. The author explained how to name my application and choose a package name, recommending using something unique like my student email address in reverse. I selected the minimum Android version my app would support, which is important for compatibility with different devices. For my first app, I chose an empty activity, provided names for my Java code file and XML layout file, and waited for Gradle to build the project. This initial setup process was straightforward and well-explained.

1.3 Understanding Project Structure

The tutorial helped me understand the project structure within Android Studio. The Android folder contains all the files for the app, the Java folder houses the Java code files, and the res folder includes resources like images, layouts, and values. The manifest file specifies the main activity for the app, acting as a roadmap for how the app should function.

1.4 Exploring Project Files

Key project files were highlighted:

- **MainActivity.java**: Contains the Java code for the app's main activity.
- **activity_main.xml**: Defines the layout of the main activity. ● **strings.xml**: Stores string resources used in the app.

1.5 Developing the App

Using the visual designer, I added UI elements such as text boxes and buttons to the app. Setting constraints to position these elements on the screen was a new and interesting challenge. I adjusted properties like input type, text size, and content for each element. The preview feature, using an Android Virtual Device (AVD), allowed me to see how the app would look and function on a real device. The author also provided instructions for creating a new AVD if one wasn't already available.

1.6 Next Steps

The tutorial guided me to write Java code to handle user interactions and perform calculations. Testing the app on different devices and Android versions was emphasized as a crucial step to ensure compatibility and functionality.

1.7 Additional Notes

Enabling auto-import in Android Studio preferences was recommended to enhance code completion, making the development process smoother. The tutorial's clear demonstration of using the visual designer and setting constraints was particularly helpful. It encouraged experimentation, motivating me to continue building and refining my app.

Overall, today's experience was incredibly enriching. I gained a solid foundation in setting up and developing an Android app using Android Studio. The step-by-step guidance and practical tips provided by the author made the learning process enjoyable and accessible. I look forward to applying these skills to create more complex and functional apps in the future.

2 LEARNING DIARY PART 2: BUILDING THE "QUICK APP LAUNCHER" WITH ANDROID STUDIO

Today, I continued my journey in Android development with the second part of a beginner series focused on building an Android app named "Quick App Launcher." This tutorial was incredibly insightful, emphasizing key concepts such as activities, intents, button functionality, and passing information between screens.

2.1 Main Concepts Covered

1. **Activities**:
    - The instructor explained that activities are the building blocks of Android apps, representing individual screens.
    - We created multiple activities to navigate between different screens within the app.
2. **Intents**:
    - Intents are actions that request specific tasks to be performed by the system or other apps.
    - I learned how to use intents to navigate between activities and launch external applications like a web browser.
3. **Button Functionality**:
    - Implementing click listeners for buttons was a key focus.
    - We added functionality to buttons to trigger actions such as navigating to another activity or opening a web page.
4. **Passing Information Between Screens**:
    - The video demonstrated how to send data from one activity to another using intents.
    - This was particularly useful for maintaining context and passing relevant information between different parts of the app.

2.2 Key Steps Demonstrated

1. **Creating Buttons**:
    - We added two buttons to the main screen, one labeled "Second Activity" and another labeled "Google."
    - The visual designer in Android Studio made this process straightforward.
2. **Adding a Second Activity**:
    - A new activity was created with a text view displaying "Hello World."
    - This involved configuring the manifest file and setting up the layout for the new activity.
3. **Implementing Button Functionality**:
    - For the "Second Activity" button, we added a click listener.
    - An intent was created to launch the second activity, and we used this intent to start the activity.
4. **Adding Functionality to the "Google" Button**:

    - The tutorial checked if the device had a browser installed.
    - If available, it launched the browser with Google's website.

2.3 Benefits of Using the App

The "Quick App Launcher" app provides quick access to frequently used apps and functions. It simplifies navigation between app screens and streamlines interactions with the Android device, enhancing overall user experience.

2.4 Further Learning

The video suggested exploring additional core Android development concepts such as:

- **Intent Services**: For performing background tasks.
- **Broadcast Receivers**: For receiving broadcast messages from the system or other apps.

The tutorial encouraged me to practice and experiment with the provided examples to gain a deeper understanding.

Overall, today's tutorial was a significant step forward in my Android development journey. The hands-on approach and detailed explanations helped solidify my understanding of activities, intents, and UI interactions. I am excited to continue building and refining my skills, with the "Quick App Launcher" as a foundation for more complex projects.

3 LEARNING DIARY PART 3: ADVANCED ANDROID DEVELOPMENT WITH LISTVIEW, CUSTOM LAYOUTS, AND IMAGES

Today, I continued my Android development journey by watching the third part of the beginner series on getting started with Android Studio. This video covered three crucial topics: using the ListView component, creating custom layouts, and displaying images within the app.

3.1 Main Concepts Covered

1. **Using ListView Component**:
    - The tutorial began by demonstrating how to use the ListView component to display a list of items.
    - The instructor showed how to set up a ListView in the app and customize the layout of each item to display more information than just the item name.
2. **Creating Custom Layouts**:
    - Next, we explored creating custom layouts for each item in the list. This involved creating a new layout file.
    - Using TextViews, we displayed additional information like descriptions and prices, enhancing the app's functionality and appearance.
3. **Displaying Images**:
    - Finally, we learned how to display images using the ImageView control.
    - The instructor demonstrated linking images to corresponding items in the list and customizing their layout, significantly improving the app's visual appeal.

3.2 Building on Previous Lessons

The video emphasized the importance of understanding the concepts from previous lessons, such as creating and running a project and understanding activities. This continuity reinforced my learning and provided a solid foundation for implementing lists and image views within an app.

3.3 Testing Functionality

The tutorial concluded by showing how to test the implemented features. Testing is vital to ensure that all components work as expected and provide a smooth user experience. This step-by-step approach to testing reinforced the importance of thorough testing and debugging in the development process.

Overall, today’s session was incredibly productive. The hands-on approach and detailed explanations provided a deep understanding of using ListView components, creating custom layouts, and displaying images in Android apps. I am excited to apply these skills in my projects and look forward to continuing my journey in Android development.