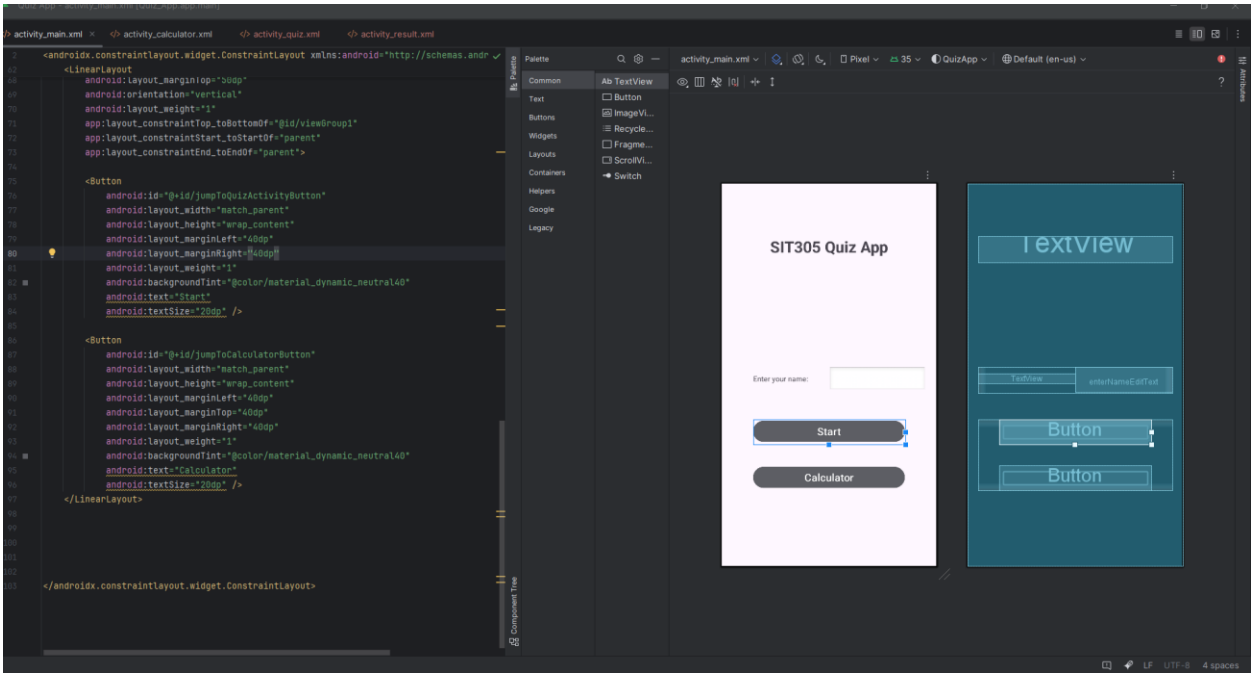
Video Link: https://deakin.au.panopto.com/Panopto/Pages/Viewer.aspx?id=2cb1a602-c9ec-48cb-96bb-b2bd007d690d

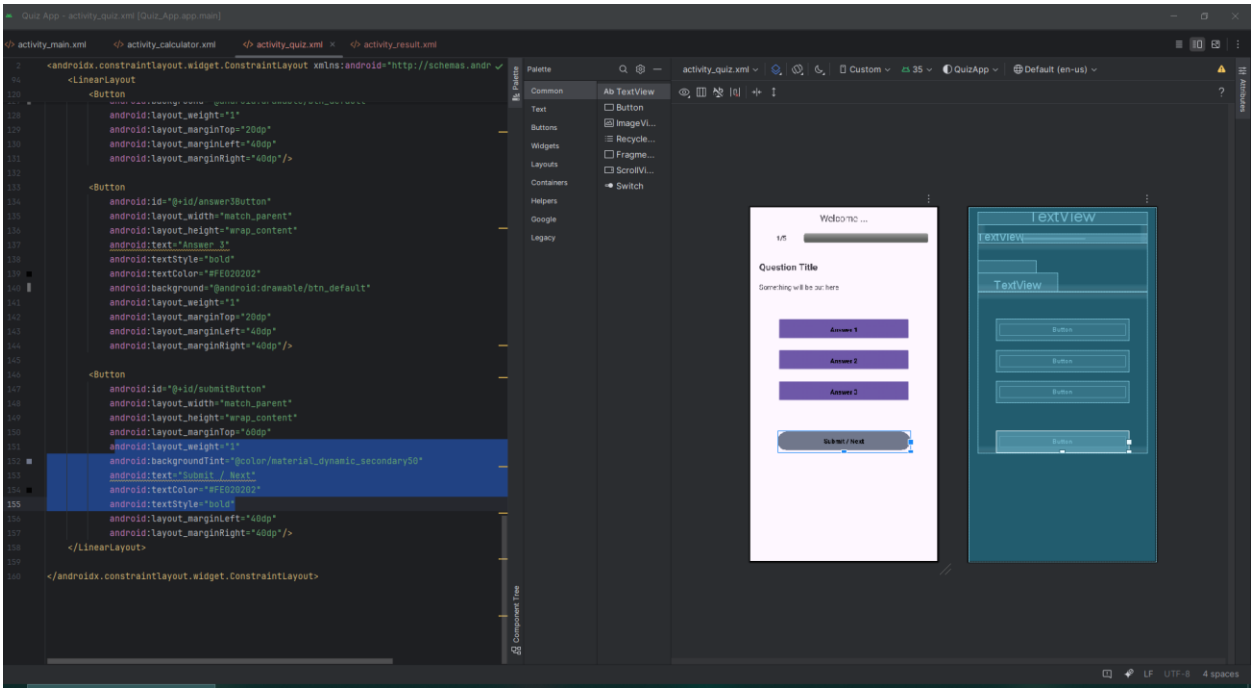
GitHub Link: https://github.com/HaydenDuong/SIT305-Mobile_Application_Development/tree/main/Task3.1C

Sceenshots:

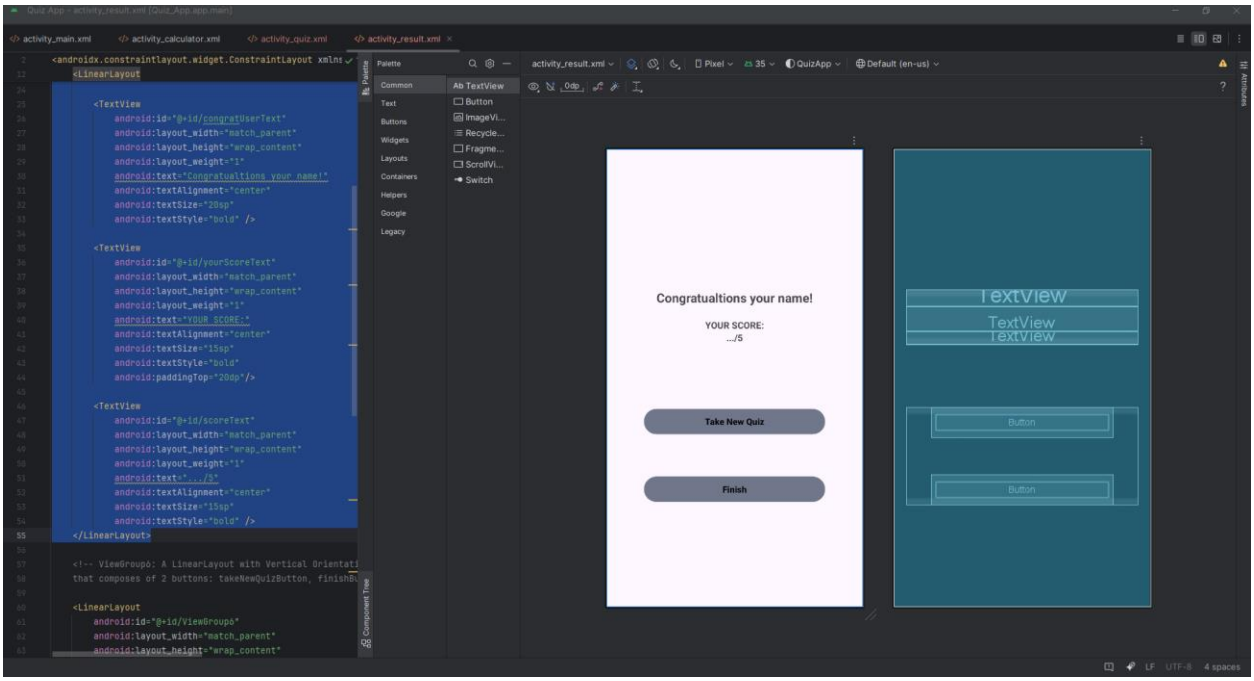
Main_Activity



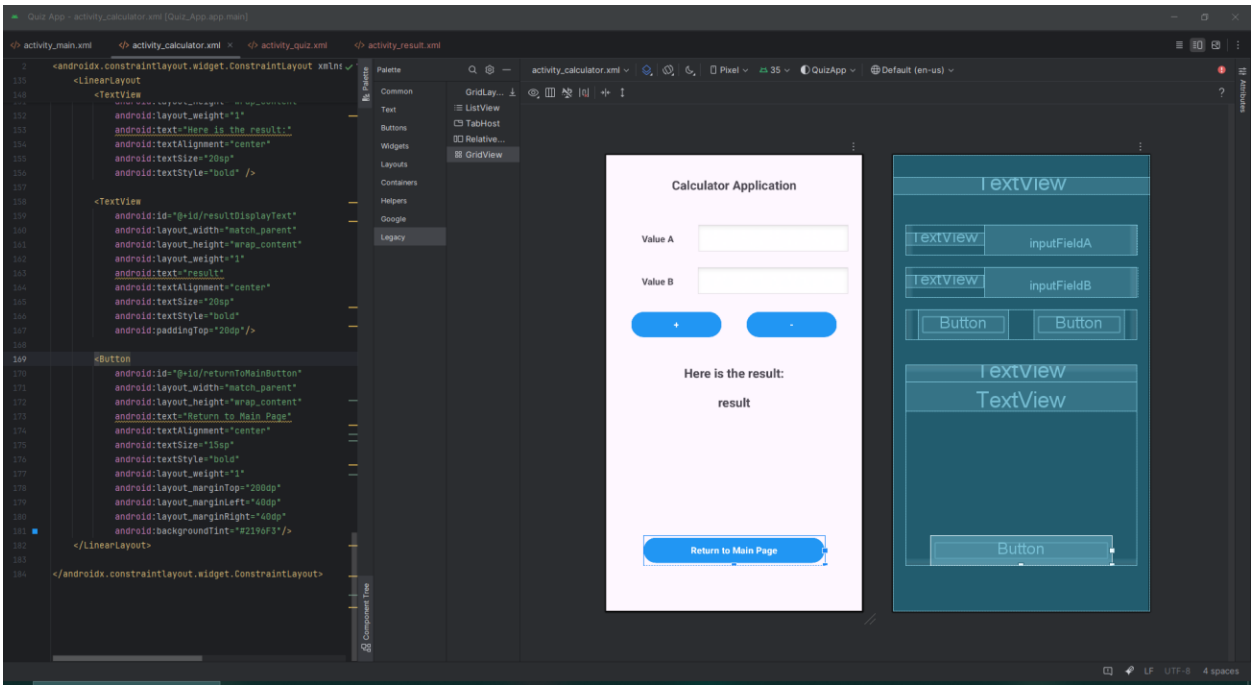
Quiz_Activity



Result_Activity



Calculator_Activity



4.

Using LLaMA 2 to Make My Quiz App Even Better

- While working on my Quiz App for SIT305 Task 3.1C, I started thinking about how I could make it smarter and more fun by using **LLaMA 2**, an awesome large language model from Meta AI. LLaMA 2 is really good at understanding and generating natural language, and I believe it can help improve my app in so many ways—like adding new features, making the experience more personalized, and even helping users learn better. Here are some ideas I came up with to bring LLaMA 2 into my app, while also using the Android concepts I've learned so far.

Idea 1: Automatically Generate New Questions

- Right now, my Quiz App uses a fixed list of questions in the QuestionBank class. But with LLaMA 2, I think I can make it generate new questions on the fly! For example, in the MainActivity, I could add a feature where users pick a topic they like, such as "Android Basics" or "General Knowledge." The app could then send this topic to LLaMA 2 (maybe through an API if I can access one), and LLaMA 2 would give me back a bunch of new questions with multiple-choice answers. I'd save these

questions into my QuestionBank and show them in the QuizActivity. This way, users won't get bored with the same old questions, and the app will always have fresh content to keep things exciting.

Idea 2: Give Personalized Feedback and Tips

- I'd love for my Quiz App to feel more like a learning buddy, and LLaMA 2 can help with that by giving personalized feedback. In the ResultActivity, where I show the final score, I could use LLaMA 2 to look at how the user did (like their score and which questions they got wrong) and create custom feedback. For instance, if someone struggles with questions about Android Activities, LLaMA 2 might say, "Looks like Android Activities were tricky for you! An Activity is a screen in your app—maybe check out its lifecycle to get better at this." I'd display this tip in a TextView right under the score. LLaMA 2 could even suggest extra resources, like a quick explanation or a link to learn more, making the app more helpful for studying.

Idea 3: Smart Helper in the Calculator

- For those doing the SIT708 task, I added a CalculatorActivity to add and subtract numbers. I think LLaMA 2 can make this part more interesting! Right now, if someone forgets to enter a number, the app just shows a basic Toast message like "Please enter values." With LLaMA 2, I could make this message more friendly and detailed, like, "Oops, you didn't enter any numbers! Try putting in two numbers, like 100 and 300, and I'll help you calculate." LLaMA 2 could also share little math tips, like explaining what addition means, and I'd show that in a TextView below the result. This would turn the calculator into a mini learning tool, not just a place to do math.

Idea 4: Smart Difficulty Adjustment

- Another cool idea is using LLaMA 2 to adjust the quiz difficulty based on how the user is doing. Right now, my app picks questions randomly with Collections.shuffle(). But if I save the user's scores (maybe using SharedPreferences), LLaMA 2 could look at their past performance and suggest the right difficulty level for the next quiz. For example, if someone keeps getting high scores, LLaMA 2 might say, "Let's try some harder questions!" I could add a difficulty field to my Question class (like 1 for easy, 2 for medium, 3 for hard) and filter questions in QuizActivity based on LLaMA 2's suggestion. This would make the quiz feel just challenging enough for each user.

How to Add This to My App?

- To use LLaMA 2, I'd probably need to connect my app to a server that runs LLaMA 2 (since it can't run directly on an Android device). I could use Android libraries like Volley or Retrofit to send requests—like the user's chosen topic or their quiz results—and get back responses from LLaMA 2, like new questions or feedback. These responses might come as JSON, which I'd parse and use to update my app's UI, like adding questions to QuestionBank or showing tips in ResultActivity. To keep the app smooth, I'd use AsyncTask or Coroutines to handle these network requests without slowing things down.

Why This Is Awesome

- Adding LLaMA 2 to my Quiz App makes it more than just a quiz—it becomes a smart learning tool! Users will love having new questions, getting personalized tips, and facing challenges that match their skill level. Plus, I get to practice Android concepts like activities, intents, and networking, which is great for my learning. In the future, I could even use LLaMA 2 to add features like voice quizzes or translating questions into other languages, making the app even more fun and accessible.