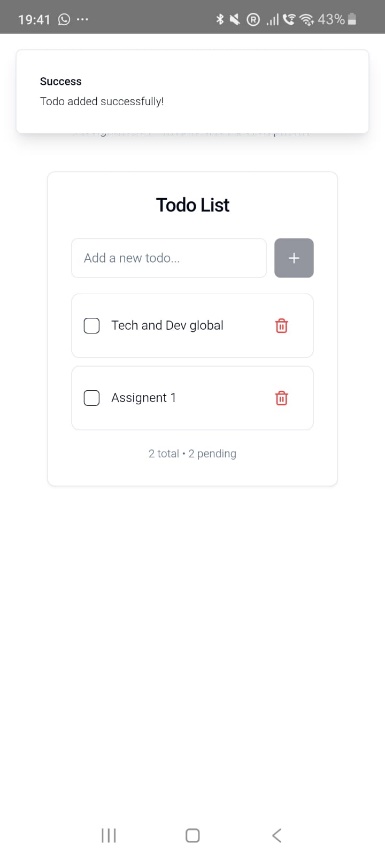
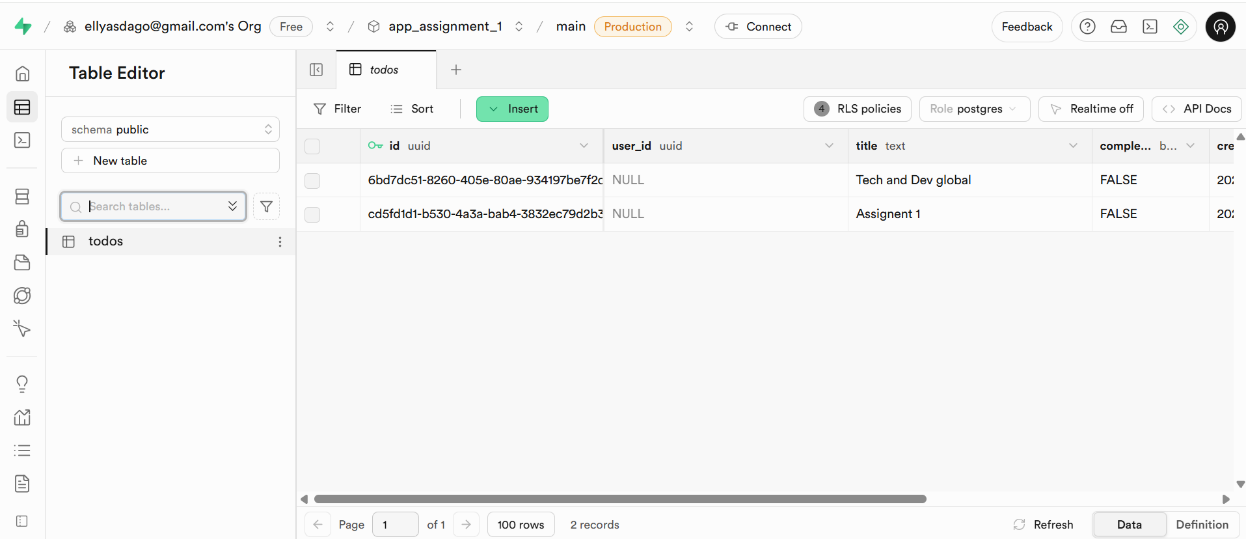
Whaya Ellyas DAGO

## Description of the Application and Learning Objectives

My application is a **to-do list.**  
The user can perform three main actions:

* Add new tasks to the list.
* Mark tasks as completed.
* Delete tasks that are no longer relevant.

All tasks, along with their status, are stored in a **Supabase database.**

**Figure 1&2 : To-do list app deployed on an Android smartphone connected to Supabase**

Since this was my **first experience developing a mobile application**, I chose to build something straightforward, with only a few core features, but still practical. A to-do list seemed like the ideal project: it is achievable to design for a newbie, yet it covers the essential steps of mobile development and respect this assignment requirements.

I decided to use **Supabase** because it provides a simple backend-as-a-service solution, which allows me to learn how to integrate a database without spending too much time on server configuration. Also, this database can be **easily integrated with Lovable**, the platform I used to create the application. This gave me the opportunity to focus on understanding the structure of the application and the deployment process.

My main objective was not only to create a useful app but also to **gain familiarity with the tools, environments, and steps required for deploying a mobile application**. Through this project, I hoped to build a foundation that would make it easier for me to develop more complex and ambitious applications in the future.

## References, Tools, and Development Process

This was my **first experience in mobile application development**, and I therefore relied on **generative AI tools,** in particular ChatGPT and Lovable. These tools were especially helpful for tasks such as writing code, connecting the application to the database, and understanding the overall structure of such a project, as well as the different environments involved.

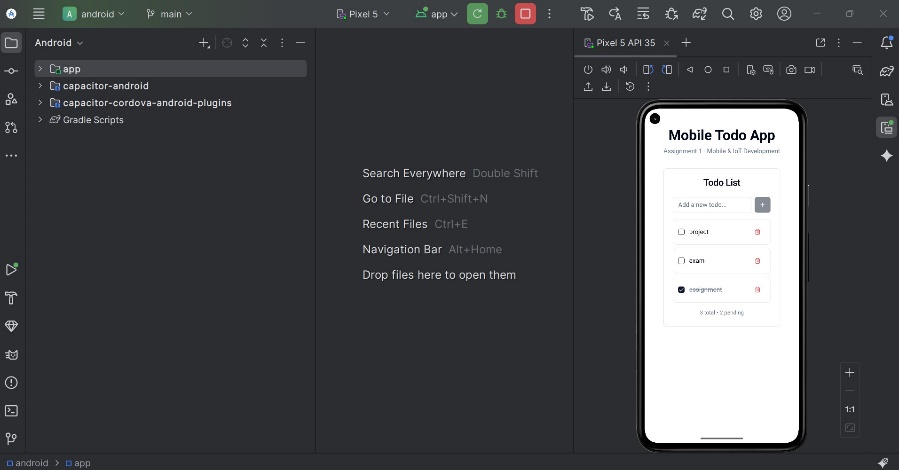
I also used **Android Studio**, which allowed me to **compile the project, simulate the app on different devices, and test each version before deploying it to an Android smartphone**. In addition to the AI tools mentioned earlier, several YouTube videos helped me get familiar with Android Studio and Lovable—both of which I had never used before:

* [Code Android YouTube Channel](https://www.youtube.com/@codeandroid5303?utm_source=chatgpt.com) (in French, focused on Android Studio).
* [AI Tuto YouTube Channel](https://www.youtube.com/@ai_tuto?utm_source=chatgpt.com) (in French, focused on Lovable).

These tutorials were particularly useful for learning how to **connect Supabase, manage GitHub integration, and handle the technical steps of deployment**.

The main challenges I faced were, first, **understanding the generated code and trying to personalize it**, and second, the **setup of different environments and the handling of unexpected issues**. For example:

* I had to install **Node.js** in order to run the dependencies and build processes required for the application.
* I had to **reorganize the project structure** so that the code could be placed in an **Android-specific folder** and properly recognized by Android Studio via Capacitor. This step was necessary to make the application run as a native Android app rather than just a web project.
* I also had to configure the parameters of an Android smartphone (borrowed from a teammate) to be able to test the app outside of the emulator.

I encountered numerous issues during these steps. In particular, during the initial tests, the application opened only a blank page and fail to run. As a workaround, I first used the **Android Studio emulator**, but even there, I faced difficulties: some emulator settings caused the app to freeze or crash. I had to carefully choose the most relevant parameters to make the app run smoothly. Once it was working in the emulator, I was finally able to deploy and test it successfully on the physical Android device.

**Figure 3: Application tests on the emulator before deployment**

This debugging phase was one of the most challenging parts of the project but also one of the most rewarding, as it gave me a much clearer view of the practical difficulties involved in mobile app development and deployment.

## URLs & Instructions to Run the Project

**URL to GitHub repository (frontend/app):** <https://github.com/ellyas-dago/firebase-pair-tasks>

**URL to Backend API (Supabase):** <https://lkjaqamhpyjettfxovql.supabase.co>

1. **Clone the repository**

git clone https://github.com/ellyas-dago/firebase-pair-tasks.git

cd firebase-pair-tasks

1. **Install dependencies**

npm install

1. **Build the web project**

npm run build

1. **Prepare the Android project with Capacitor**

npx cap copy android

npx cap sync android

npx cap open android

1. **Run on a physical device**

* Enable **Developer Options** and **USB Debugging** on the Android phone used for tests.
* Connect the phone to the computer via USB and authorize the connection.
* In **Android Studio**, open the android/ project.
* Select your physical device from the menu at the top of the Android studio page.
* Press **Run** to build and install the app directly on your phone.

Teamwork