Binaya Rimal

02/09/2025

Milestone 4

Like all of my other milestones, the artifact that I am using is still the Event Tracker App. It was created in my Mobile Development class. It is a CRUD app that allows users to track events. For this section, I have decided to convert my database from SQLite to Firebase. This transition enhances the app’s scalability, real-time data synchronization, and cloud storage capabilities, making it more suitable for multi-user environments.

I selected this artifact for my portfolio because it highlights my ability to develop mobile applications with a strong backend integration. The Event Tracker App showcases my proficiency in Java, XML, and database management, as well as my ability to implement CRUD functionality in a real-world application. By transitioning from SQLite to Firebase, I demonstrated my ability to work with cloud-based databases, implement authentication, and manage real-time data. This improvement enhances the app’s functionality by allowing multiple users to access and update events simultaneously, providing a more seamless and dynamic user experience.

The firebase is properly set up; however, the database change seems to have had a big impact on the overall app so I am still in the process of debugging to make the app run. My primary goal was to improve my ability to integrate cloud-based services into mobile applications, and by implementing Firebase, I would successfully achieve that. Additionally, this enhancement helped me strengthen my knowledge of database migration and authentication implementation.

Throughout the enhancement process, I learned how to transition from a local database to a cloud-based one, which involved restructuring data storage and optimizing queries for Firebase’s NoSQL architecture. I also gained experience in implementing Firebase Authentication to manage user access and security.

One of the biggest challenges I faced was adapting my existing data models to Firebase’s document-based structure, as it differs significantly from relational databases like SQLite. Additionally, handling real-time updates while maintaining efficient data retrieval required careful optimization. However, overcoming these challenges strengthened my problem-solving skills and expanded my understanding of modern mobile app development techniques.

Overall, this enhancement improved the functionality and scalability of my Event Tracker App, making it a more robust and practical mobile application.

Based on the feedback, I would be giving a video demo upon completion of the project. I have included the direction on how to run it below if that is of interest.

How To Start the app:

Option - 1

When the app is fully complete, Android Package Kit will be sent to download the app. With it users will be able to download the app and use it from their android devices.

* Open **Settings** > **Apps & notifications** (or **Security**).
* Select **Special app access** > **Install unknown apps**.
* Choose the app (e.g., Chrome, File Manager) from which you will install the APK.
* Toggle on **Allow from this source**.

Option – 2:

Besides the APK, I will also be sending my entire codebase. Users can download it in the computer and download Android Studio and run the app from android studio. Users must open the app from Android Studio.

**2. Open the Project in Android Studio**

1. Open **Android Studio** on your laptop.
2. Click **"Open"** or **"Open an Existing Project"**.
3. Navigate to the project folder you extracted/downloaded.
4. Click **OK** and wait for Android Studio to load the project.
5. **Wait for Gradle to sync** – If prompted, click **"Sync Now"**.
6. If you see a **missing SDK error**, go to **File > Project Structure > SDK Location** and set the correct Android SDK path.
7. If necessary, update dependencies in build.gradle and click **"Sync Now"** again.
8. Connect a **physical Android device** via USB **or** use an **Android Emulator** (AVD).
9. Click **Run**
10. Select the target device/emulator and wait for the app to launch.