

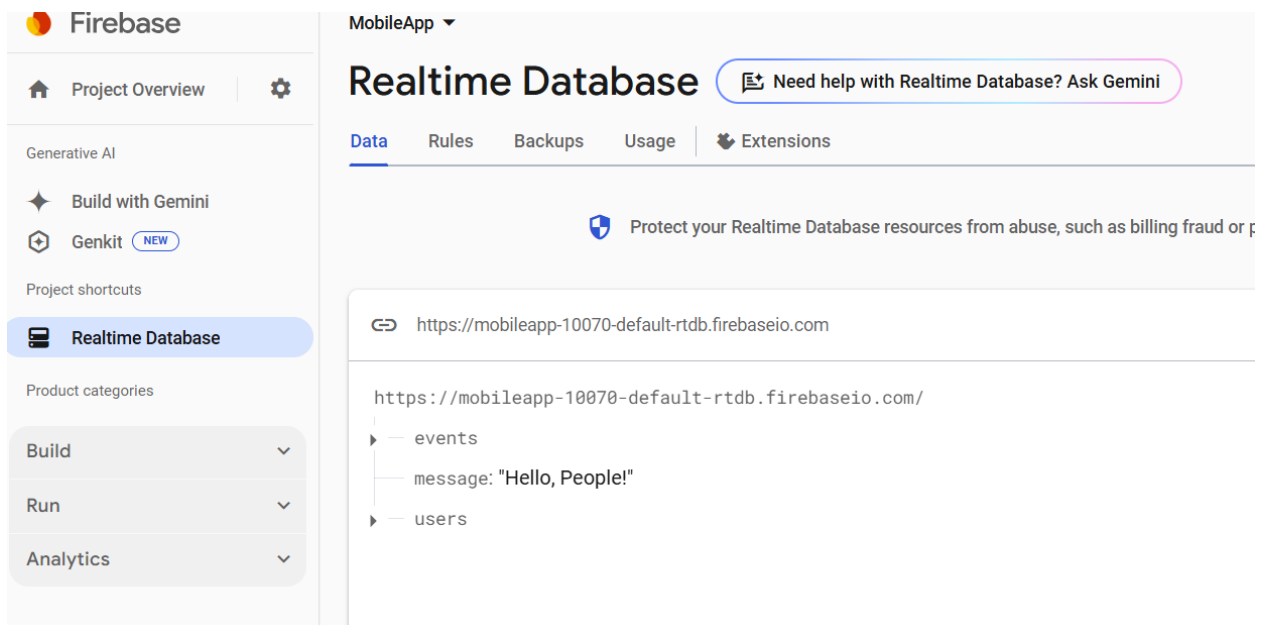
Binaya Rimal

02/09/2025

## Database

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.

Below is the firebase database that was created:



I selected this artifact for my portfolio because it highlights my ability to develop mobile applications with 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.

Changing the database had a significant impact on the overall app, requiring adjustments to ensure compatibility with Firebase. One of the main challenges was understanding Firebase's NoSQL structure, which differed from SQLite's relational format. This required restructuring data storage and modifying queries to align with Firebase's real-time synchronization capabilities.

Below is the new code of the `getEvents` function that is changed to match firebase:

```
2 usages  Binaya Rimal
public void getEvents(final OnEventsRetrievedListener listener) {
    // Use the default order by push key (this is implicitly done)
    Binaya Rimal
    databaseEvents.addValueEventListener(new ValueEventListener() {
        2 usages  Binaya Rimal
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            List<Event> eventList = new ArrayList<>(); // List of Event objects

            for (DataSnapshot snapshot : dataSnapshot.getChildren()) {
                Event event = snapshot.getValue(Event.class);
                if (event != null) {
                    String eventId = snapshot.getKey();
                    event.setEventId(eventId);
                    eventList.add(event); // Add Event object directly to the list
                }
            }
            Collections.reverse(eventList);

            // Call the listener with the retrieved events
            listener.onEventsRetrieved(eventList);
        }

        Binaya Rimal
        @Override
        public void onCancelled(DatabaseError databaseError) {
            Log.e(tag, "FirebaseError", msg, "Failed to retrieve events: " + databaseError.getMessage());
        }
    });
}
```

My primary goal was to improve my ability to integrate cloud-based services into mobile applications, and by implementing Firebase, I successfully achieved 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.

Link to the video demo of this enhancement:

<https://www.youtube.com/watch?v=nuiu8mVwNRE>

Instructions to download the app:

**Pre work:**

Download Android Studio

Clone the Repository

### **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.