**Event Finder Android App**

**Team members:**

**X00185050 – Waiho Wong**

Worked on Android, Frontend, App testing (50%)

**X00186372 – Dawid Ratajczyk**

Worked on Web API backend, Android (50%)

**GitHub Repository:** <https://github.com/DawidRatajczyk02/CA2_MobileProject.git>

**Service URI Scheme:** <https://eventfinder-api-a2efbwf9hyfrbqfb.westeurope-01.azurewebsites.net>

**Azure Deployment Settings**

Obraz zawierający tekst, numer, Czcionka, linia

Zawartość wygenerowana przez sztuczną inteligencję może być niepoprawna.Obraz zawierający tekst, numer, oprogramowanie, zrzut ekranu

Zawartość wygenerowana przez sztuczną inteligencję może być niepoprawna.Obraz zawierający tekst, numer, oprogramowanie, Czcionka

Zawartość wygenerowana przez sztuczną inteligencję może być niepoprawna.Obraz zawierający tekst, zrzut ekranu, oprogramowanie, numer

Zawartość wygenerowana przez sztuczną inteligencję może być niepoprawna.

**About**

This is an Event Finder app where you can browse events, create events, edit events and delete events. This app also has full support for English and Spanish speakers.

**Screenshots**

The backend of the Event Finder app was built using Azure and a RESTful API with ASP.NET Core. The API includes endpoints for event management (create, update, delete, retrieve) and user management (register, login). The app communicates with the backend via Retrofit in the Android frontend. The backend was deployed on Azure App Services, using SQL Database for data storage. API endpoints were tested and integrated with the mobile app for seamless functionality.

Obraz zawierający tekst, zrzut ekranu, numer, Czcionka

Zawartość wygenerowana przez sztuczną inteligencję może być niepoprawna.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Zawartość wygenerowana przez sztuczną inteligencję może być niepoprawna.

Main page app with a list of events from the database and a button to change to Spanish and an add event button.

Add event button will let you create an event by filling the empty fields and once submitted, the new event will be saved in the database and be displayed in the main page.

A screenshot of a cell phone

AI-generated content may be incorrect.A screenshot of a phone

AI-generated content may be incorrect.

Pressing on one of the events will bring you to the details page of that event and display all the information and the delete, edit and go back buttons.

Edit button will display all the information in fields which you can replace or add onto the text and then when you save the changes, the updated information will display and be saved in the database.

Delete button will give a confirmation message asking you if you are sure about deleting the event and once you press delete, event will be erased from the database.

Unfortunately, were were unable to implement the users. But the idea was that once a user creates an event it is assigned to him and only he can manage it (a user id would be assigned to an event).

A screenshot of a phone

AI-generated content may be incorrect.A screen shot of a phone

AI-generated content may be incorrect.

A screenshot of a phone

AI-generated content may be incorrect.

**Testing Report**

Tool Used: Espresso

There are 4 total tests created:

* Change language test
* Add event test
* Edit event test
* Delete event test

From creating the tests and having issues while creating, the language button has a weird bug where once you switch to Spanish, then you press an event then go back, and press the language button again to switch back, it won’t work unless you press you press it again. Because of this issue, I have made it so in the test, it presses twice to change back to English.

Add event test and Edit event test, I hardcoded what to type in the fields or replace in the fields for editing and works well. The Delete event button also works well, telling the test which event to delete by hardcoding the title text of the created event.  
  
Test is best executed altogether (not individually) because of the hardcoding.