

Taller 2 — Introducción al Desarrollo Web Móvil (UCN)

1) Grupo y equipo

- Grupo 1 — Equipo: SmartCoders

2) Integrantes (Nombre — RUT)

- Bastian Salinas — 21.848.994-K
- Benjamín Cuello — 21.682.135-1
- Benjamín Salas — 21.758.667-4
- Tomás Guerra — 21.664.344-5

3) Tecnologías y bases de datos

- API Pokémon: NestJS (Node.js, TypeScript) + PostgreSQL
- API Países: Express.js (Node.js) + PostgreSQL
- API Clima y Feriados: FastAPI (Python) + MongoDB
- Frontend móvil: Apache Cordova + HTML/CSS (Tailwind CDN) + JavaScript

Estructura relevante del proyecto:

- `backend/nest/api-pokemon/` (Pokémon)
- `backend/express/` (Países)
- `backend/fastapi/` (Clima y Feriados)
- `frontend/www/` (frontend reutilizado por Cordova)
- `cordova/` (proyecto Cordova para generar APK)

4) Cómo instalar y ejecutar (resumen)

Requisitos: Docker Desktop, Node.js y npm, Cordova CLI, Java JDK + Android SDK.

1. Levantar las 3 APIs y el frontend (Docker Compose)

```
docker compose up --build
```

Servicios por defecto:

- Pokémon (NestJS): `http://localhost:3000`
- Países (Express): `http://localhost:4000`
- Clima/Feriados (FastAPI): `http://localhost:8000`
- Frontend (estático): `http://localhost:8080`

2. Generar APK Android con Cordova (en otra terminal)

```
npm run sync-www:cordova  
cd cordova  
cordova platform add android  
cordova build android
```

Archivo generado (debug):

```
cordova/platforms/android/app/build/outputs/apk/debug/app-debug.apk
```

La apk fue probada en BlueStacks, en la cual funciona correctamente.