Ads Scraping and Dashboard Exercise

Objective

Build a POC system that **scrapes Facebook Ads Library data for Nike** and **displays it in a web dashboard**.

Part 1 - Scraping

Using the Facebook Ads Library page below, programmatically scrape Nike ads:

https://www.facebook.com/ads/library/?active_status=all&ad_type=all&country=US&is_targeted_country=false&media_type=all&search_type=page&view_all_page_id=15087023444

For each ad, store in a DB:

- Ad ID
- Status (active / inactive)
- Platform(s)
- Start date
- End date (if any)
- Ad asset (image/video file)

Retrieving **only the first 50 ads** is sufficient.

Please design and describe:

- Scraping approach and tools (in Python)
- Data storage and schema

Part 2 - Dashboard Web App

Build a simple **full-stack dashboard** to visualize the scraped ads.

Backend

- **Node** (framework of your choice)
- Dockerized

- Have it structured as a **full-blown large application**, even though the content it minimal

Frontend

- **React** (framework of your choice)
- Dockerized
- Have it structured as a **full-blown large application**, even though the content it minimal

Dashboard Requirements

- At least one visualization (chart/table)
- Show ads over time
- Distinguish active vs inactive ads
- Dates range filter
- Platform filter
- Ads assets

Make sure to have relevant data in the DB to demonstrate all functionality, even if edited after scraping.

Instructions

- Implement scraper, backend, frontend
- Submit via email to asaf.shai@adge.ai or via GitHub to user gargyas
- Include a short README explaining what it does, how it works, and how to run it

Aim for a quick implementation - it's ok to make assumptions and cut corners, to achieve an end-to-end working demo quickly.

Then let's meet for a review and code review.

Thank you and good luck!