Rox iOS take-home project

Project Overview

A core feature of the Rox iOS app displays news stories about the companies our users are following. Your task is to create a News Reader application that fetches and displays articles from NewsAPI.org.

API Information

- **API Provider:** NewsAPI.org
- **Documentation:** https://newsapi.org/docs
- **API Key:** You will need to register for a free API key

Core Requirements

1. News Feed View

- Display a list of news articles fetched from the https://newsapi.org/v2/top-headlines endpoint
- Each article in the list should show the following at the minimum:
  - Headline
  - Source name
  - Publication date (formatted appropriately)
  - Thumbnail image (if available)
- Handle loading states and network errors gracefully
- The feed should paginate through all available articles
- Layout & UI design is up to you

2. Article Detail View

- Navigate to an article detail view when a user taps on an article in the feed
- The screen should display the following at the minimum:
  - Headline
  - Featured image (if available)
  - Source name
  - Publication date (formatted appropriately)
  - Article content or description
  - Link to the original article

- Include a way to return to the news feed
- Layout & UI design is up to you

Extra Credit Options

These are completely optional and listed in no particular order

Category/Company Filtering

- Implement a mechanism to filter news by categories or by company
- Display currently selected category

AI Features

- Intentionally left open-ended…feel free to be creative!

Technical Requirements

- Interface implementation should be done predominantly in SwiftUI
- Include handling for loading, error, and empty states
- You may use any open-source libraries you like

Evaluation Criteria

Your submission will be evaluated on:

1. **Functionality**
   - Working implementation of core requirements
   - Proper handling of API responses and errors
2. **Code Quality**
   - Clean, readable, and well-organized code
   - Appropriate architecture and separation of concerns
     i. Your implementation decisions should roughly reflect how you would write code in a production environment
3. **User Experience**
   - Intuitive navigation and layout
   - Appropriate handling of view states
   - Visual design and attention to detail

Submission Guidelines

1. Create a GitHub repository for your project
2. Include a README with:

- ○ Screenshots of the Feed & Detail screens
- ○ Setup instructions
- ○ Brief explanation of your architecture choices
- ○ List of implemented features (including any extra credit)
- ○ Shortcuts or hacks you included. Note anything you'd do differently in a real production environment
- ○ Any notable challenges and how you solved them
3. Email the repository link to eric@rox.com when complete

Good luck!