

Genesys Code Challenge

A frontend application built with Angular to display the top and latest news from the Hacker News API.

Features

Display Top News: Fetches and displays the top 500 news stories.

Display Latest News: Fetches and displays the latest news stories.

Pagination: Implemented a "Load More" button to display articles in batches of 10 for better performance.

Dynamic UI: Visually distinguishes between different news types (e.g., stories, jobs) to improve readability.

More details about the code challenge are available in the document - **Frontend Coding Challenge - Hacker News Updated 2024.pdf**

Technologies Used

- Frontend : Angular 19 , TypeScript , SCSS/CSS, FontAwsome
- Backend : Hacker News API (provided) (<https://github.com/HackerNews/API>)

Prerequisites

Make sure the following tools, with specific versions or later, are installed on your machine.

```
Angular CLI: 19.2.17
Node: 23.5.0 (Unsupported)
Package Manager: npm 10.9.2
OS: win32 x64

Angular: 19.2.15
... animations, common, compiler, compiler-cli, core, forms
... platform-browser, platform-browser-dynamic, router

Package                                Version
-----
@angular-devkit/architect              0.1902.17
@angular-devkit/build-angular          19.2.17
@angular-devkit/core                   19.2.17
@angular-devkit/schematics             19.2.17
@angular/cli                           19.2.17
@schematics/angular                   19.2.17
rxjs                                    7.8.2
typescript                             5.6.3
zone.js                                0.15.1
```

Installation and Setup

- Clone the repository - git clone [makarandvichare2/genesysNewsTest](https://github.com/makarandvichare2/genesysNewsTest)
- Navigate to the project directory - cd your-repository
- Install the dependencies - npm install
- Run the application - npm start
- Run the unit tests - npm run test

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	98.6	96	96.96	98.47	
app	100	100	100	100	
app.component.html	100	100	100	100	
app.component.ts	100	100	100	100	
app/common/enviornment	100	100	100	100	
enviornment.dev.ts	100	100	100	100	
app/dashboard/components/news-dashboard	97.43	50	100	97.29	
news-dashboard.component.html	100	100	100	100	
news-dashboard.component.ts	97.36	50	100	97.22	69
app/dashboard/components/news-item	90	100	50	88.88	
news-item.component.html	100	100	100	100	
news-item.component.ts	88.88	100	50	87.5	24
app/dashboard/components/top-menu	100	100	100	100	
top-menu.component.html	100	100	100	100	
top-menu.component.ts	100	100	100	100	
app/dashboard/constants	100	100	100	100	
api-endpoints.const.ts	100	100	100	100	
app/dashboard/enums	100	100	100	100	
news-selection.enum.ts	100	100	100	100	
news-type.enum.ts	100	100	100	100	
app/dashboard/models	100	100	100	100	
news-response-model.ts	100	100	100	100	
pagination.model.ts	100	100	100	100	
app/dashboard/pipes	100	100	100	100	
date-in-words.pipe.ts	100	100	100	100	
news-title.pipe.ts	100	100	100	100	
app/dashboard/services	100	100	100	100	
news.service.ts	100	100	100	100	

- Run the cypress tests(e2e) - npm run e2e
- Run the storybook tests - npm run storybook

Development assumptions

- **Pagination** -
 - The provided API returns all top 500 news items in a single request. Client-side pagination was implemented to handle subsequent API calls for individual news details and improve application performance. The page size is set to 10 news items per page.
- **News Entity** - The “score” property from the API was assumed to correspond to the “points” property on the existing Hacker News UI.
- **Scope** - Only the "Top News" and "Latest News" features were implemented; other features were excluded from this challenge.
- **API Response** – The API is expected to return the top and new stories in sorted order, so no additional sorting logic is needed on the client side.
- **Local Memory** – Technologies like NgRx or local storage are not used since RxJS caching is sufficient for this purpose.

Design and Architecture

Sequence diagram for the load more news feature -

