

CAP4104Final

Introduction

Welcome to News API. The most innovative way to stay up-to-date with the latest news from around the world. Our web app is designed to make the process of consuming news faster, easier and more personalized to your interests.

We achieve this by utilizing a cutting-edge news API that allows us to gather breaking news from reliable sources and present it to you in an organized and user-friendly format. Our news API collects news from various sources and updates the content every minute so that you can be sure to get the latest news as soon as it happens.

Our web app provides you with various categories of news, such as business, sports, entertainment, politics, technology and more. Additionally, you can customize your news feed by selecting your preferred sources and topics.

We understand that every second counts when it comes to news, which is why our web app is designed to deliver news quickly, efficiently and accurately. Our goal is to provide you with a seamless experience and help you stay informed and up-to-date with the latest news from the United States(around the world to come).

Usability goals

Efficiency: Users should be able to quickly and easily find the news they're interested in, without having to spend too much time searching or navigating the site.

We did this by using a lot of builtin streamlit functions and by making the web app very intuitive and user friendly

Navigation: The web app should have clear, intuitive navigation that makes it easy for users to find the different categories of news and customize their news feed.

We did this by naming everything appropriately, and making everything fit into their proper tabs and headers. We also ensured that navigation was smooth by implementing features such as interactive tables and a map.

Personalization: Users should be able to personalize their news feed by selecting their preferred sources and topics, so that they only see the news that's relevant to them.

Users are able to sort through news that they want to see by a variety of methods, categories, relevance, and more, particularly through the ability to filter news stories by a selected state or region for more relevant, curated news content according to a user's preference.

Responsiveness: The web app should be responsive and load quickly, regardless of the device or internet connection speed.

We did this by choosing a fast and responsive API that we can request information from very fast

Accessibility: The web app should be accessible to users with disabilities, including those who use screen readers, keyboard-only navigation, or other assistive technologies.

We aimed to address accessibility throughout our project by keeping in mind this user base and how to make our interface as accessible as possible to any user, anywhere.

Readability: The web app should use clear, legible fonts and have a layout that's easy on the eyes, with appropriate use of white space, images, and other visual elements.

We did this by choosing an easy font to read, making headers and dividing the web app into proper, consistent sections.

Consistency: The web app should have a consistent design and layout across all pages, with a clear hierarchy of information and consistent use of colors, fonts, and other visual elements.

We did this by looking back on what we did and matching it to what we will do

Error prevention and recovery: The web app should be designed to prevent errors, such as broken links or missing images, and provide clear feedback and guidance if an error does occur.

We did this by testing the web app and seeing any errors that can occur and finding ways to either fix them or if it's an API error notifying the user of what is happening.

Design process

Define the problem: Finding reliable news becomes harder and harder as the internet gets bigger and bigger. We made this web app to make it easier to find trustworthy and reliable news.

Develop a concept: Using a News API we will make an intuitive and user friendly web app.

Sketches: we made mock up designs of what the web app should look like. Such mock ups were brainstormed by us and implemented through software and applications such as Figma.

Build the web app: We started building the web app in python using streamlit

Test and iterate: Conduct usability testing and gather feedback from users to identify areas of the web app that need improvement. Use this feedback to make changes to the design and functionality of the web app.

Launch and maintain: Once the web app is complete, launch it to the public and continue to monitor and maintain the web app.

API integration

We used the API to make news requests and display them to the user in different ways and formats, and sort them in specific ways based on the user's preferences.

Some problems we ran into:

One was that they have everything and top articles separated so you could only sort by categories in top articles and not in everything.

Another issue was reaching the limit on the api request we could request in a day

A future issue is if the api changes the website could become unusable, so someone must maintain it through any API changes.

Interactive widgets

For our web app, we implemented the use of various widgets, such as incorporating a checkbox function specifically for selecting and specifying a user's choice of state or region to receive curated news based on their selection. Similarly, the use of a map and interactive table allowed for greater interactivity between the user and the application by providing information about news based on region to a user.

HCI design principles

Our Web App follows primarily the principles of consistency, as we tried through our overall design to implement tables and other widgets and components which were evenly spaced, with equal font size and headings. Similarly, we also followed the principle of efficiency by sorting news and curating it based off a user's selection of state and/or region

Testing and feedback

We have made design improvements, as before we had everything on one page, and split it up into different pages and tabs based off what users have said, Another thing is we improved or added onto some of our things based off what users have said, for instance, on the search bar you can now limit it to articles that were released on a certain date. Finally, we made navigation improvements based on what users have said so now things are grouped together nicely and have a flow.

Conclusion

Overall, the process of creating this app involved a lot of brainstorming to figure out how to elevate the NewsAPI to make it more curated by categorizing news by states or region. We

were overall successful, and despite hardships and challenges faced along the way, successfully implemented a coherent and efficient project which achieves its purpose.