NoCap: Fact Checking with AI

Software Design Document

Names:
Anthony Ciero,
Varun Doddapaneni,
Joshua Pechan,
Thomas
Chamberlain

Advisor:
Dr. Marius Silaghi
Date: (09/29/2025)

TABLE OF CONTENTS

1. Introduction

1.1 Purpose

The purpose of the “NoCap” project is to evaluate media sources and texts, and provide an evaluation rating to the user. The purpose of this software design document is to showcase the architecture and design of the project.

1.2 Scope

The software will be a website that allows users to figure out how credible a source is. The website also allows other viewers to see the authenticity rating of these articles stored in our database on the website. Our software is for any user who wants to evaluate any article or any piece of text, and obtain a score on how credible it is.

1.3 Overview

This document provides a description for the architecture of our software, as well as our architecture diagram and visualization of our user interface.

1.4 Definitions and Acronyms

AI: Artificial intelligence

API: Application Programming Interface

AWS: Amazon Web Services

URL: Uniform Resource Locator

WCAG: Web Content Accessibility Guidelines

2. System Overview

General Description:

The system is a web-based application that is complemented by a chrome extension. It is designed to evaluate the authenticity of text or web articles. Users can submit raw text or a URL, and the system returns an authenticity score along with a detailed report explaining the reasoning. The platform also maintains aggregate credibility scores for publishers, allowing users to see broader trends in information credibility. A database that the score and reasoning will be stored in will be created and will be stored based on the URL so any raw text analysis will not be stored in the database.

Context and Design:

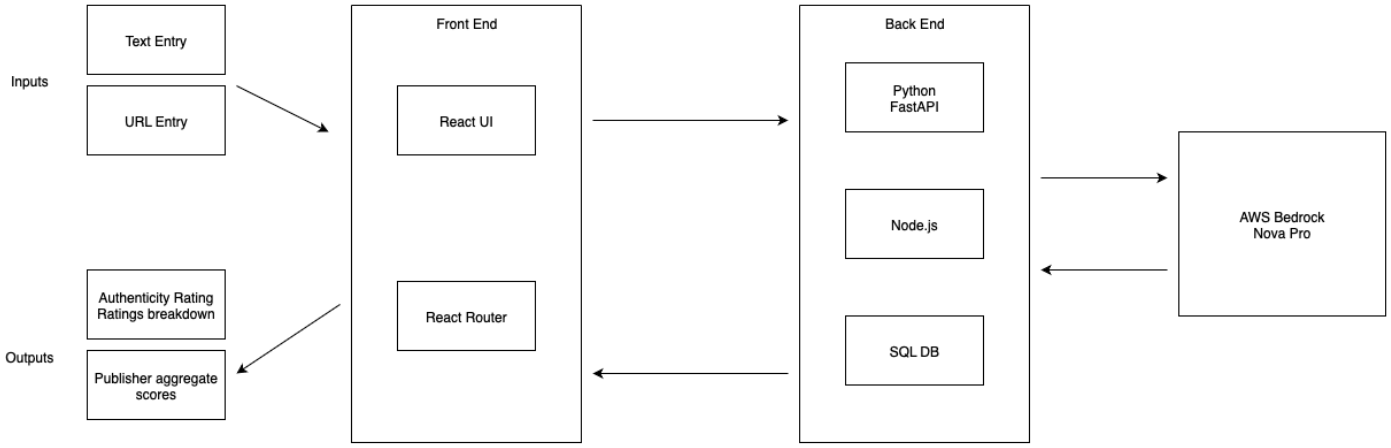
The system relies on Amazon Bedrock for AI-based text analysis and uses cloud infrastructure to process and return results quickly. It is designed to be accessible across all major browsers and devices, complying with WCAG accessibility standards. The chrome extension allows for seamless in-context analysis without leaving the users browsing experience. The chrome extension will solely be on google chrome.

Background Information:

With the growing challenge of misinformation online, users need a fast and reliable way to assess article credibility. This system combines a detailed assessment of articles showing the location and explanation of misinformation and a convenient and easy to use interface.

3. System Architecture

3.1 Architectural Design



3.2 Decomposition Description

The overall architecture of the project can be divided into the following: Inputs, Outputs, Front End, Back End, and external connections. The inputs are the text entry or URL to be entered. These inputs are what the user will want to be evaluated by the application. These inputs are entered into the front end UI, built by React.

The entered data is routed to the back end, which is built using Python, and utilizes the FastAPI. Node.js enables the JavaScript code that the front end uses to be executed outside of the web browser. The backend will have functions that pass off data to the external connections on AWS, including the Nova Pro AI model. The back end utilizes prompt engineering, which includes the LangChain and LangGraph modules, to instruct the model on what to do.

The resulting data is the authenticity rating and insights on the input. If a URL was entered, the results are stored into the database under the corresponding publisher. The results are passed back to the front end, and results in the outputs that the user can see.

3.3 Design Rationale

The way the architecture in 3.1 is organized is intended to reflect a single "flow" or "process" that the user will go through. They will enter an input, which is entered into the front end. This is then processed by the back end with the help of the AWS external connections, then the transformed data is passed back to the front end, which is displayed as the output. This specific architecture was chosen because of its relative simplicity and ease of use. We want an application that is fast and easy to use, and our choices reflect this.

4. Data Design

4.1 Data Description

Our AI will either read the raw text provided by the user or will take the URL and read the article and take the text. It will then divide it into tokens, where the AI will analyze which tokens can be considered false or misinformation. The website will then store the articles along with their authenticity score and authenticity report.

4.2 Data Dictionary

Our input data will consist of two possible types. Raw text can be inputted by the user and our AI will take this and tokenize it into words. This is used to determine the authenticity score and report, but raw text data will not be saved in our database. More commonly, a URL can be inputted. When this happens, our AI will write the authenticity scores and reports, but unlike with raw text, if the specific inputted URL is not in our database, we save it along with its score and report and save it in our database, where users will be able to view articles previously inputted by other users and categorize them by publisher.

5. Component Design

5.1 Front End (React UI and Router)

- **Purpose:** Collects user input (text or URL) and displays the authenticity rating and breakdown.

5.2 Back End (Python, FastAPI, Node.js)

- **Purpose:** Processes requests from the front end, routes them to external services, and returns results.

5.3 Database

- **Purpose:** Stores URL results and publisher authenticity data for later reference. Sorted under publisher.

5.4 External Connection (AWS Bedrock, Nova Pro)

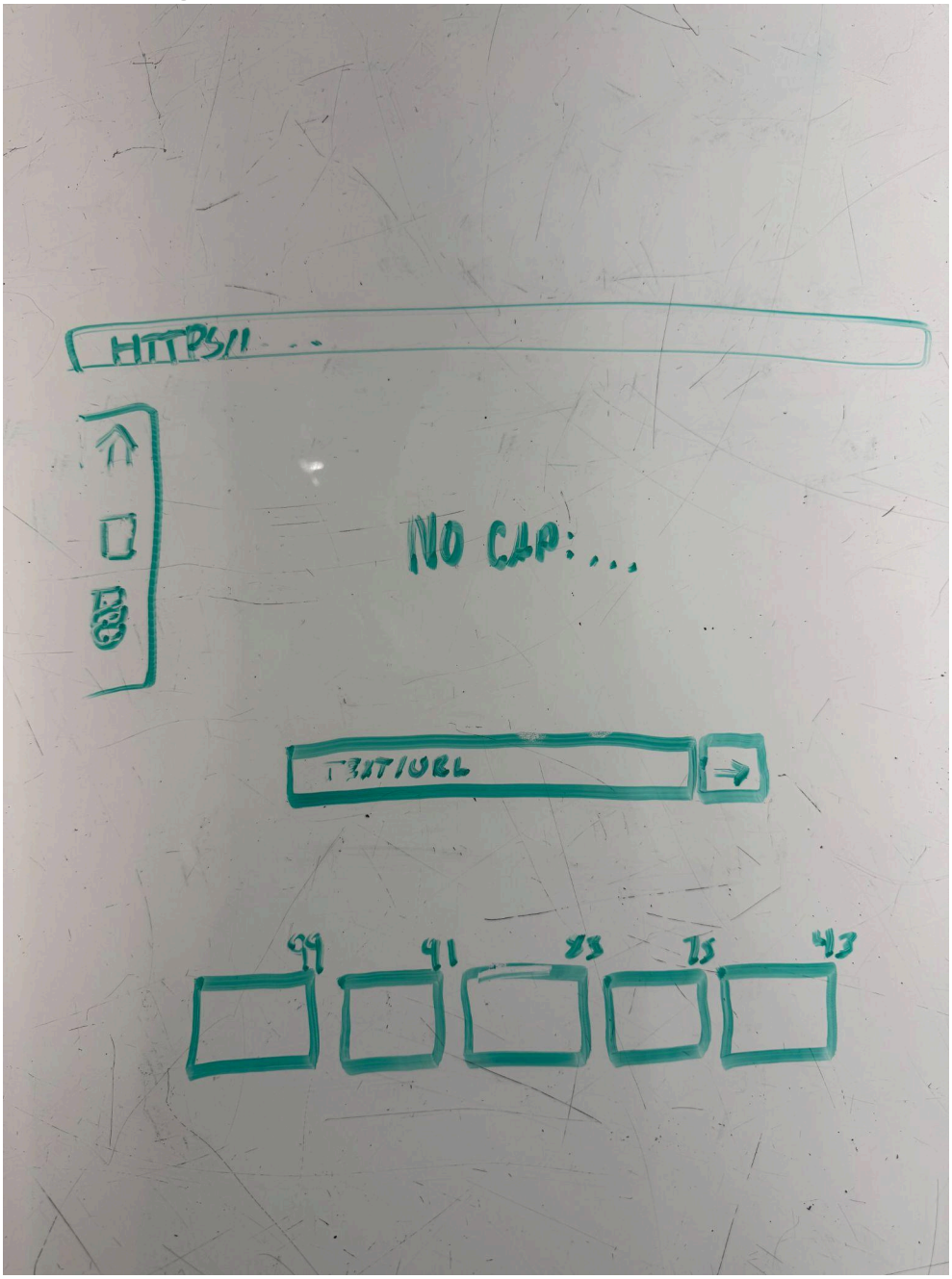
- **Purpose:** Performs the AI model inference to determine authenticity.

6. Human Interface Design

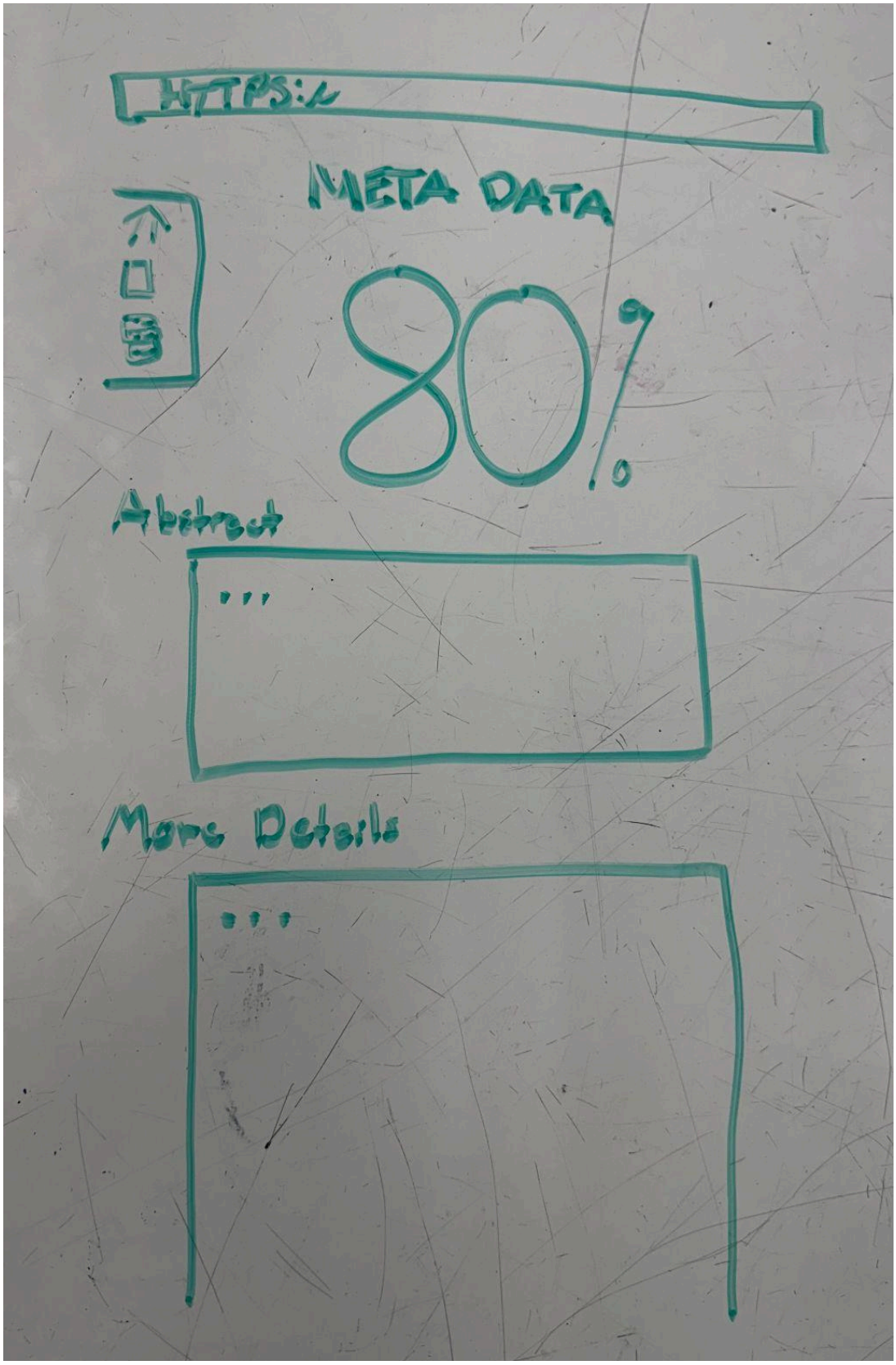
6.1 Overview of User Interface

1. The user navigates to the NoCap website and pastes an article URL or raw text into a submission box.
2. Click the submit button.
3. The user is then redirected to the report page that shows:
    a. Article metadata - title, author, publication date, domain
    b. the authenticity score - score for the site determined by the AI
    c. analysis report - report containing any and all reasoning for the authenticity score
4. Users can then navigate to the database page to find other articles or see aggregate scores from submitted authors.

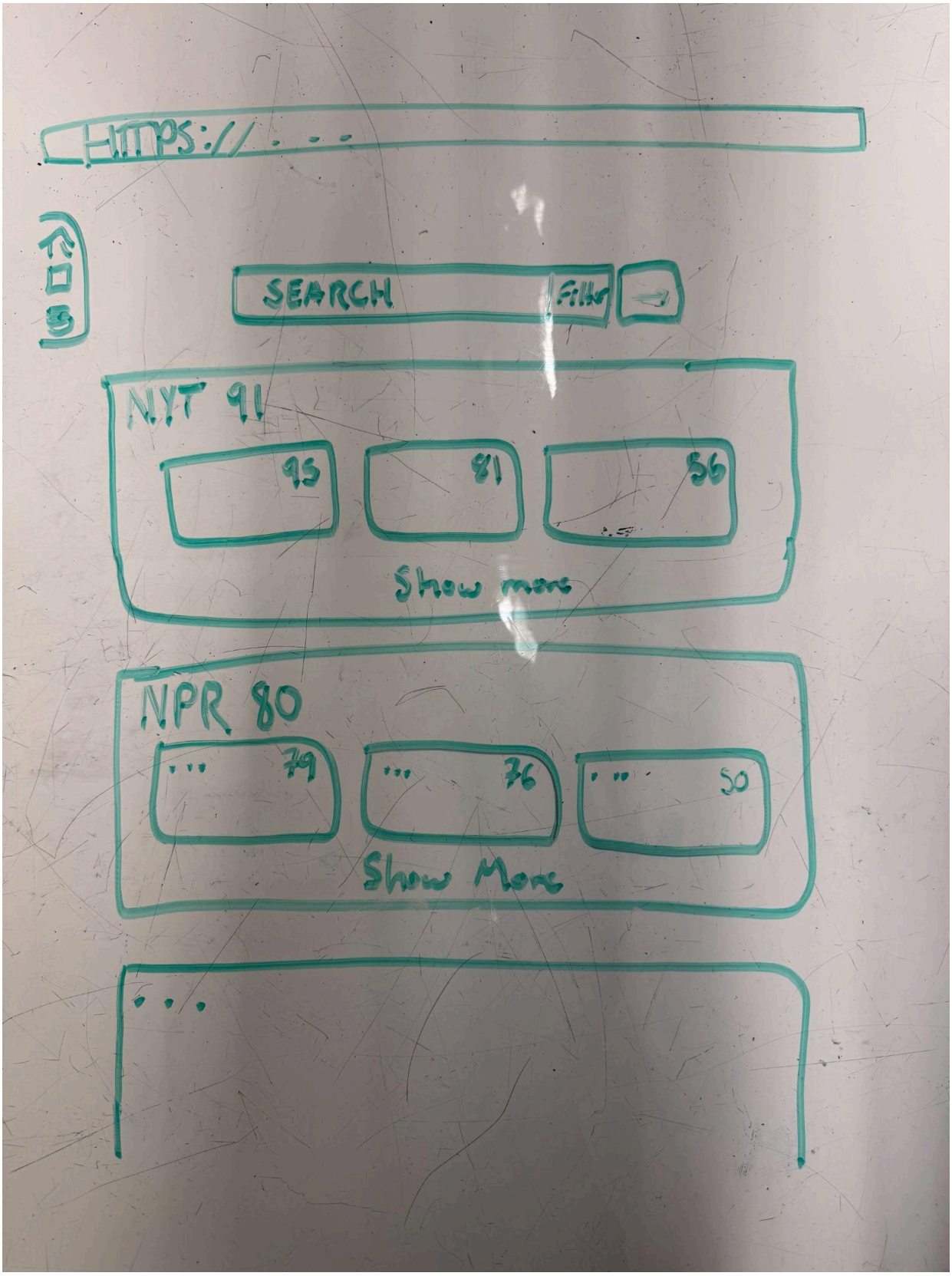
6.2 Screen Images



Home Page



Report Page



Database Page

6.3 Screen Objects and Actions

6.3.1 Home Page

- Text/URL Entry Box: Small box where users can paste their text or URL they want to fact check
- Submit Button: Button next to text/URL entry box used to enter the input in the box. Button will not work if the text/URL entry box is blank.
- Example Database Entries: On the home page below the entry box will show three cards of example articles and reports. These will be to show and explain to users the AI's scoring system and example reports. Users can click on these to view their full report in the report page.
- Page Navigation Center: Navigation bar on side used for going to other pages. Consists of Home button, Report button, and Database button. The Home button brings the user to this page. The Report button will allow users to go to the most recently viewed report. The button will not be available if the user has not generated a report, viewed one of the example article cards below the entry box, or entered the database to select an article. The Database button brings the user to the page where users can view the stored articles. This appears on all pages in order to navigate between them easily.

6.3.2 Report Page

- Article Meta Data: Information about the article like article title, publisher, author, and publication date.
- Authenticity Score: Score generated by AI, numerically ranking the authenticity of the given article.
- Authenticity Report Abstract: Small abstract about the authenticity report. In place for easy understanding of larger authenticity reports.
- Authenticity Report: Report generated by AI, reporting and explaining what parts of the article it deemed as fact or fiction, its reasoning behind the authenticity score, and the authenticity of the article as a whole.
- Page Navigation Center

6.3.3 Database Page

- Article Search Bar: Search bar at the top of the page that allows users to search specific articles or keywords to check our database for pre-authenticated information about topics.
- Article Filter Options: Allows users to filter by different options such as alphabetically, highest/lowest authenticity score, publication date oldest/newest.
- Search Button: Button next to the article search box used to enter the input in the box.
- Publisher Authenticity Cards: Cards that show the specific publishing company, their average authenticity score based on articles in our database, and the three articles with the highest authenticity score, as well as a show more option expanding the card to view more articles of the same publisher.

- Article Cards: Cards within the publisher cards for a specific article, showing article name and authenticity score. Clicking on the article will bring the user to the report page of the given article.
- Page Navigation Center