NoCap: Fact Checking with AI

Thomas Chamberlain tchamberlain2023@my.fit.edu
Anthony Ciero aciero2022@my.fit.edu
Josh Pechan jpechan2023@my.fit.edu
Varun Doddapaneni vdoddapaneni2023@my.fit.edu

1. Marius Silaghi msilaghi@fit.edu
2. Marius Silaghi msilaghi@fit.edu FIT CSE Professor
3. Progress of current Milestone (progress matrix)

Task	Completion %	Thomas	Anthony	Josh	Varun	To do
1. Design front end ● Home Screen ● Input Bar ● Report Screen ● Database Screen	40%	33%	33%	33%	0%	Article cards for home and database,add info for article, cleaner design
2. Set up AI model on AWS	100%	0%	0%	50%	50%	
3. Establish basic connection between AI and the API	75%	25%	25%	25%	25%	Establish the API route so frontend and backend can communicate
4. Build URL reader	90%	0%	0%	20%	80%	Connect to LLM functionality
5. Develop rudimentary backend and API structure	50%	0%	0%	50%	50%	Connect the text reading class and the ai class

4. Discussion (at least a paragraph) of each accomplished task (and obstacles) for the current Milestone:
   - Task 1: Started the creation of our front end design screens, including our home page, report page, and database page. These pages are easily accessible with our navigation side bar. Our home page consists of a search bar for inputting article URLS or raw text (and blank cards that we will fill with our article information). Our report page has text for meta data that we will fill later, a big number percentage currently 0 until we have our AI gather authenticity score, and a text area for our report abstract and full details, currently holding place

holder text. Our database page has a search bar for looking for articles or publishers. We still need to get report information from our AI for filling in the front end as well as creating a cleaner design, including max sizes for text areas and better scrolling. The sidebar is easily expandable if we do try to add more pages later, however that will most likely not be needed. There is a routing file as well as correct separation of all the pages and components to ensure easy expansion and readability.

- Task 2: We used the AWS Bedrock and IAM modules to set up our model. We chose the Nova Pro model as our primary model (additional options for models are planned for the future). We ensured the AWS account had the access keys set up, so each of us could access the AI model on our IDEs. We also changed specific permissions with the model so we could use it for the purpose we want it for.
- Task 3: We wanted to get the input prompt of the AI to be enterable from the front end. We have the AI ready to take an input, and the front page. We still need to connect them.
- Task 4: We wanted to have a way for the content of an article to be processed in the back end. We created a reader class that reads the relevant content of an article webpage and outputs it into the terminal.
- Task 5: We wanted to have a basic structure in our project environment and in our backend. We have a class that handles the AI input and output, and another that reads the content of a linked article, and outputs it in the terminal. In the future, we want to connect these two processes.

5. Discussion (at least a paragraph) of contribution of each team member to the current Milestone:
    - Thomas: Created the front page and text box to input text and URLs into. Set up basic front end structure.
    - Anthony: Helped with the front end development, specifically the report page and the database page, as well as helping with the design of the side navigation bar and its symbols, and making the input bar have different displays per page, later connecting to send the input to our AI for report making or our database for article searching.
    - Josh: Redid most of the front end, reorganized the files, fixed what should be components as well as separated the pages and styling for each page. Added routing and fixed scalability issues. Started work on setting up the backend database as well.
    - Varun: Set up the LLM model on AWS. Built backend files, such as getting the AI to generate a response, and a class that reads and saves article content from a given URL.

6. Plan for the next Milestone (task matrix) or

Task	Thomas	Anthony	Josh	Varun
1. Prompt engineering	0%	0%	50%	50%

2. Get a basic score of an article	0%	0%	50%	50%
3. Start process to break text down into tokens	0%	0%	50%	50%
4. Develop the backend database	0%	0%	100%	0%
5. Article Report/Publisher Cards	50%	50%	0%	0%
6. Article meta data connection	50%	50%	0%	0%

7. Discussion (at least a paragraph) of each planned task for the next Milestone or "Lessons Learned" if this is for Milestone 6
    - Task 1: Start on prompting the LLM model, so we can get the output to roughly be what we want. We want to ensure the model analyzes the text content in the way we want it to. Do basic prompting to format the output in the way we want it.
    - Task 2: Get a basic score from the AI reading an article's content. The model will give a basic score of an article after it reads the text. Connecting the back end classes and getting basic prompt engineering are tasks that must be completed first. This will be expanded upon later.
    - Task 3: We want to be able to break the text from an article into tokens. This will be the first step in getting an analysis of the text from the LLM model. We intend to use the LangChain module to help with this.
    - Task 4: The database using dynamoDB will need to be developed creating a data table and all large amounts of data will need to be stored in S3. The data table will have references to the data in S3 so that it is easily accessible.
    - Task 5: Create cards that store important report information per article. They will display the authenticity score, publisher, title, and image. 5 cards will show up on the home screen ranging from high to low authenticity for example. In the database screen, these cards will be in larger publisher cards, showing publisher name and authenticity average. These will be displayed by highest score first and show approximately 5 articles before expanding one.
    - Task 6: Once these cards are made, we will click on them to bring us to the report page for that specific article. We must fill out the report page with information like meta data such as title, author, and publication date, the authenticity score, and the report abstract and full details.
8. Date(s) of meeting(s) with Client during the current milestone: see Faculty Advisor meeting date below
9. Client feedback on the current milestone
    - see Faculty Advisor Feedback below
10. Date(s) of meeting(s) with Faculty Advisor during the current milestone:
    - Oct 24th
11. Faculty Advisor feedback on each task for the current Milestone

- Task 1: Advisor suggested to ensure the LLM model cannot be “tricked” with an article. Try giving the URL directly to the LLM and give it a prompt to strictly follow a certain process.
- Task 2: Advisor suggested score takes certain context, like how old the article is and on what subject it is about as factors that can affect the score.
- Task 3: Feedback related to task 1 feedback. The advisor suggested we ensure that only relevant info is given to the LLM and prevent any inputs that can “trick” the model. The advisor also suggested we take article images and videos into account.
- Task 4: The advisor suggested sorting the data stored on the database page which would in turn mean that the database would require good pk and sk to ensure quick look up and sorting.
- Task 5: Informed advisor about adding article cards and publisher cards for the home page and database page. Our advisor considered being able to filter publisher cards and information in those cards by categories other than highest authenticity score, such as publication date.
- Task 6: Informed advisor that our next plan is to get meta data for articles and display to the report screen when a specific report is generated from the home input bar or viewing an article card.

12. Faculty Advisor Signature: ______________________________ Date: ________

13. Evaluation by Faculty Advisor

- ○ Faculty Advisor: detach and return this page to Dr. Chan (HC 209) or email the scores to pkc@cs.fit.edu

○ Score (0-10) for each member: circle a score (or circle two adjacent scores for .25 or write down a real number between 0 and 10)

Thomas	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
Anthony	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
Josh	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
Varun	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10

- ○ Faculty Advisor Signature: ___________________________ Date: __________