Progress made thus far:

Web Scraping:

So far, one of the main aspects of progress is web scraping of the Smogon forums using the Node.js library Puppeteer. This library provides tools to navigate through each thread in the forum, specifically targeting the team building discussion section, and extracting the text from each thread.

Extension Logic:

As for the chrome extension's logic, I've successfully implemented a basic template. It allows me to run the extension on my own browser, displaying a popup with basic UI elements such as the tool's name, icon, and a table. This table will be used to showcase the relevant posts for the user's query of interest.

Remaining tasks:

BM25 Ranking:

One of the main remaining tasks is the implementation of BM25 ranking for relevant articles. Currently, I am looking into PyScript to make use of the NLP libraries we've used in class. The goal is to assess the results by reviewing the resulting forum posts and confirming their relevance to the user's query.

UI Improvement:

To enhance the extension's user interface, the plan is to integrate relevant forum threads based on the user's query. The popup will feature a list of threads, allowing users to scroll through and find the information they need. The implementation will include a search bar for inputting the users specific query (something like “Rillaboom hyper offense” looking for a specific pokemon / strategy).

Challenges/Issues being faced:

Goal Reorientation:

Based on the remaining time, it’d be best to reorient the original goal. The focus would be to provide posts from the Smogon forum that are relevant to a specific Pokemon/playstyle. The idea of making specific suggestions has been moved to a stretch goal, to be explored based on available time.

Web Scraping Challenge:

One notable challenge encountered was implementing web scraping in a Chrome extension. Since the target sites are forums with multiple pages and posts, specific tools were required.

Puppeteer from Node.js provided the necessary functionality. However, due to Node.js not being able to run within the extension directly I used a different approach. A local Node server now handles the scraping, and the extension requests the forums' scraped data from it.