Checkpoint 4: Build a Website for Fake Review Detection

Objective:

Develop an interactive website where users can input an product URL. The backend will scrape reviews from the provided link (using the script from Checkpoint 3) and classify each review as "Real" or "Fake," displaying the results on the frontend.

What is a Website for Fake Review Detection?

This website will enable users to analyze reviews from any product page. By entering the product URL, the website will scrape reviews, process them using the trained models from Checkpoint 2, and display predictions for each review in an intuitive interface.

Basic Components of the Website:

1. **Frontend Development:**
    - Provide a text input field where users can enter the product URL.
    - Add a button for users to initiate the review scraping and classification process.
    - Display each scraped review along with its classification result ("Real" or "Fake").
    - Use any frontend technology such as React, Angular, Vue.js, or basic HTML/CSS/JavaScript—whichever technology you are most comfortable with.
2. **Backend Development:**
    - Use the web scraping script from Checkpoint 3 to extract reviews from the product URL.

- Load the trained models from Checkpoint 2 to classify each scraped review as "Real" or "Fake."
- Create an API endpoint that accepts the product URL, performs web scraping, and returns the classified reviews in JSON format.

3. **Data Flow:**
   - User provides the product URL via the frontend.
   - Backend scrapes the reviews and classifies them using the trained model.
   - Backend sends classified reviews to the frontend, which displays them in a user-friendly format.

Tasks:

1. **Set Up Your Environment:**
   - Select technologies for frontend (React, Vue.js, etc.) and backend (Flask, Django, Node.js, etc.).
   - Install the necessary libraries and dependencies (e.g., Flask/Django for backend, React for frontend, BeautifulSoup for scraping).
2. **Frontend Development:**
   - Build an input form with a text field for the product URL and a submit button.
   - Create a section to display the scraped reviews along with their classification results.
   - Style the page using CSS frameworks like Bootstrap or Tailwind CSS.
3. **Backend Development:**
   - Implement the scraping logic from Checkpoint 3 to extract reviews from the product URL.
   - Integrate the trained model from Checkpoint 2 to classify reviews.
   - Create an API endpoint that takes the product URL as input, processes the reviews, and returns the results.

4. **Integration:**
    - Connect the frontend to the backend via API calls.
    - Ensure the user experience is smooth, with appropriate error handling for invalid URLs or failed scraping attempts.

Deadline:

16th February 2025, 11:59 PM

Deliverables:

1. **GitHub Repository with:**
    - Frontend code.
    - Backend code (including the integrated web scraping script and model loading logic).
    - README file documenting:
        - The website’s functionality.
        - Steps to set up and run the project locally.
        - Technologies used.
    - (Optional) Link to the deployed website.
2. **Checklist for Review:**
    - The website should successfully scrape reviews and classify them as "Real" or "Fake."
    - The frontend should display the results clearly.
    - Ensure all deliverables are uploaded to the GitHub repository titled Project_WoC_7.0_Fake_Review_Detection in the folder checkpoint 4.