PROCEDURE FOLLOWED:

- First part involves data extraction. Used a feedparser to parse the RSS feeds and extract article details like title, content, publication date, and source URL.
- The second step involved the setting up of the data storage. SQLAlchemy to interact with a PostgreSQL database. Schemas are defined for the articles and store them without duplicates.
- Next steps involves processing the articles and categorizing them.
- Furthermore, spaCy for classifying articles into the provided categories: terrorism, natural disasters, uplifting, and others. Basically test classification is performed over here.
- Modification of the Celery worker to include classification after processing the article. This will allow the system to scale and handle articles asynchronously without blocking the main process.
- Inclusion of logging to track the progress and handle potential errors.
- Here the Redis would be used in backend for sending and receiving messages.

TECHNOLOGIES USED [WITH THEIR SPECIFIED VERSION]:

feedparser==6.0.10

SQLAlchemy==1.4.32

psycopg2==2.9.3

celery==5.2.3

nltk==3.6.5

spacy==3.1.3

redis==4.1.0

MODIFICATIONS:

[These were not implemented into the project but ideated for better handling of the project]

- Sentiment analysis can be included as it will be able to give better results as it not only goes by the gives the keywords but also by statements and gives an overall score which in return will help in better classification.
- Word2Vec can be incorporated for better contextual understanding and classification.
- Dynamic dashboards can also be implemented and incorporated into this.