**Project Proposal**

**Project Scope:**

The proposed project, the **Real-Time Social Media Sentiment Analysis Tool**, is designed to analyze sentiments of text inputs instantly. This tool aims to provide users, ranging from individuals to professionals, with immediate feedback on the emotional tone of textual data. The project will utilize Python for backend sentiment analysis and React for creating a dynamic and user-friendly frontend.

**Technologies and Methodology**

Our project will employ **Python** and **React** as the core technologies. **Python** will be used to handle backend operations, including sentiment analysis leveraging libraries such as **nltk** or **textblob** etc. A **Flask-based RESTful API** will facilitate communication between the frontend and the backend, managing data flow and analysis results. **React** will be used to develop the frontend interface, enabling real-time user interactions and display of sentiment analysis. The entire project will be hosted on **GitHub**, which supports **Agile** development practices by enabling incremental updates, feature tracking, and version control.

**User Stories and Development Timeline**

The development of this tool will be driven by user-centric stories to ensure it meets end-user expectations effectively. Key user stories include:

* A **user** who wishes to quickly understand the sentiment of written text for immediate interpretation or response.
* A **marketing professional** looking to analyze customer feedback on social media to gauge public sentiment towards a product or service in real time. These stories will shape the functionality and interface of the tool, ensuring practical utility in real-world scenarios.

The project timeline will be managed through an Agile burndown chart, which will help monitor the progress against planned deadlines and ensure timely completion of the project phases. The development process will be divided into several **two-week sprints**, each dedicated to developing and refining features such as the API setup, sentiment analysis processing, and the user interface.

**Tasks Breakdown:**

1. **Initial Planning and Setup:**
   1. Define user stories and project requirements.
   2. Set up development environment (Python, React, Flask).
2. **Backend Development:**
   1. Implement initial sentiment analysis module using Python libraries (nltk, textblob).
   2. Create a RESTful API using Flask to handle data flow.
   3. Integrate additional NLP tools.
3. **Frontend Development:**
   1. Build a basic React interface for user input and output.
   2. Implement real-time sentiment display on the frontend.
   3. Add styling and UI enhancements for user-friendliness.
4. **Testing and Optimization:**
   1. Test the sentiment analysis functionality for accuracy.
   2. Optimize performance of the backend and frontend components.
   3. Conduct user feedback sessions and make improvements.
5. **Documentation and Deployment:**
   1. Write a comprehensive README.md.
   2. Prepare deployment documentation.
   3. Finalize project hosting on GitHub.
6. **Final Review and Presentation:**
   1. Conduct a final review of all features.
   2. Prepare project presentation materials.

**Dataset Link : 🡺** [**dataset**](https://we.tl/t-foVbcgrJvJ)

**README.md 🡺** [**README**](https://we.tl/t-0ozjzI6MWT)