



AI Powered Document Summarizer

The project is aimed at creating a AI-based web application which enables users to enter lengthy documents and receive and concise summary of the content.

Key features:

- **Document input** : The application allow users to paste or input text documents of varying lengths.
- **Auto summarization** : NLP algorithms will process the text and then automatically generate the summary.
- **User friendly Interface** : Designing an intuitive and user-friendly interface where users can easily input their documents and view the generated summaries.
- **Interactive Output**: Separate output page will show the summary as well as input document. User can copy the generated summary to clipboard, ask for the more short or concise summary.
- **Number of words and Characters** in input text by the user and in the summary generated by the application. This feature will help to get the required summary in the pre-defined word limit for the user.
- **Error Handling** : Implement error handling to gracefully manage scenarios where users input invalid documents, unsupported format or if any processing issue encountered.
- **Real-time Processing** : Summarization process will occur in real-time, enabling users to receive their summaries quickly.
- **Responsive design** : The platform web application that functions seamlessly on various devices, including desktops, tablets, and smartphones.

Optional Features:

- **Multiple document input and output** : User can choose how many summaries he wanted and then he will decide the best of all. Also, allowing multiple files input will benefit user in generating summary for many files simultaneously.
- **Document Format Handling** : Support various document formats, including plain text and common document formats like PDF and DOCX. Implement the necessary parsing and extraction mechanisms.
- **Feedback system** : User may provide feedback on the output summary generated.
- **Extractive and abstractive summarization**

Key Technologies and requirements in the project:

- Python for backend development, NLP processing, NLTK or spacy and flask setup
- React-JS and CSS for front-end development
- File handling libraries
- Axios library in React for asynchronous HTTP request
- Deployment platforms
- Design tools like figma or freeform for designing the UI/UX
- Version control system like Git - Github.