

# Problem Statement

- **Introduction to GenAI and Simple LLM Inference on CPU and finetuning of LLM Model to create a Custom Chatbot**

# Unique Idea Brief (Solution)

**ParliamentGuru is a web application that enhances accessibility to political documents like legislative texts. Utilizing advanced natural language processing and AI, it allows users to upload PDFs and query them for specific information, providing accurate, context-based responses through the ChatGroq model. The app employs GoogleGenerativeAIEmbeddings for efficient document processing and similarity searches, all within a user-friendly Streamlit interface. Targeting politicians, researchers, and civic tech enthusiasts, ParliamentGuru simplifies the navigation of complex political**

**texts, promoting transparency and informed civic engagement by ensuring users can quickly access and understand relevant information.**

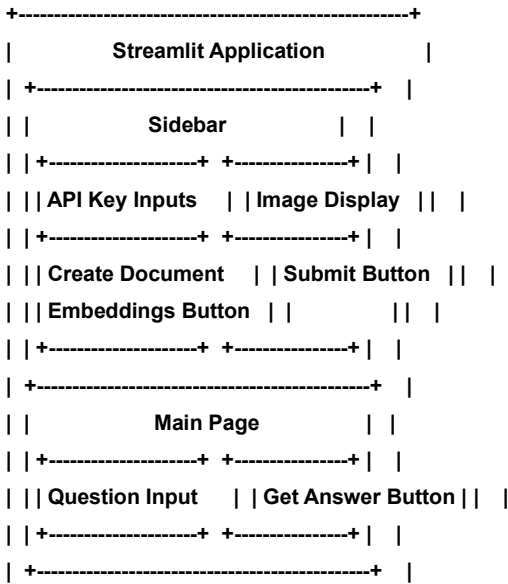
# Features Offered

- **API Key Management**
- **User Interface**
- **Document Embedding Creation**
- **Question and Answer System**
- **Response Display**

# Process flow

- **Environment Setup**
- **Sidebar Configuration**
- **Main Application**
- **Vector Embedding Creation**
- **Question Input and Answer Retrieval**

# Architecture Diagram



+-----+

+-----+

|           Backend Services           |

| +-----+ |

| |           Langchain           | |

| | +-----+ +-----+ | |

| | ChatGroq   | | FAISS   | |

| | +-----+ +-----+ | |

| | GoogleGenerativeAIEmbeddings       | |

| | +-----+ +-----+ | |

| | PyPDFDirectoryLoader   | | RecursiveCharacterTextSplitter | |

| | +-----+ +-----+ | |

| +-----+ |

| +-----+ |

| |           Environment Management       | |

| | +-----+           | |

| | dotenv   |           | |

| | +-----+           | |

| +-----+ |  
+-----+

+-----+

|           **Workflow**            |

- | 1. User inputs API keys and uploads documents.    |
- | 2. Documents processed to create embeddings.       |
- | 3. User queries handled by ChatGroq.               |
- | 4. Relevant responses retrieved based on context.    |

+-----+



# Technologies used

- **Streamlit**
- **Langchain**
- **Langchain Community**
- **Google Generative AI Embeddings**
- **Dotenv**

# Conclusion

**Parliament Guru effectively combines advanced AI with a user-friendly interface using Streamlit. It leverages Langchain, GoogleGenerativeAIEmbeddings, and FAISS for efficient document processing and retrieval. Users can input API keys, upload PDF documents, and query them seamlessly. The integration of the ChatGroq model ensures accurate, contextually relevant responses. Managing API keys with dotenv adds security. The app's ability to split documents into chunks and create a vector store enhances retrieval speed and precision. Future enhancements could include real-time updates, support for more document**

**formats, and advanced NLP features, making ParliamentGuru an invaluable tool for document querying.**