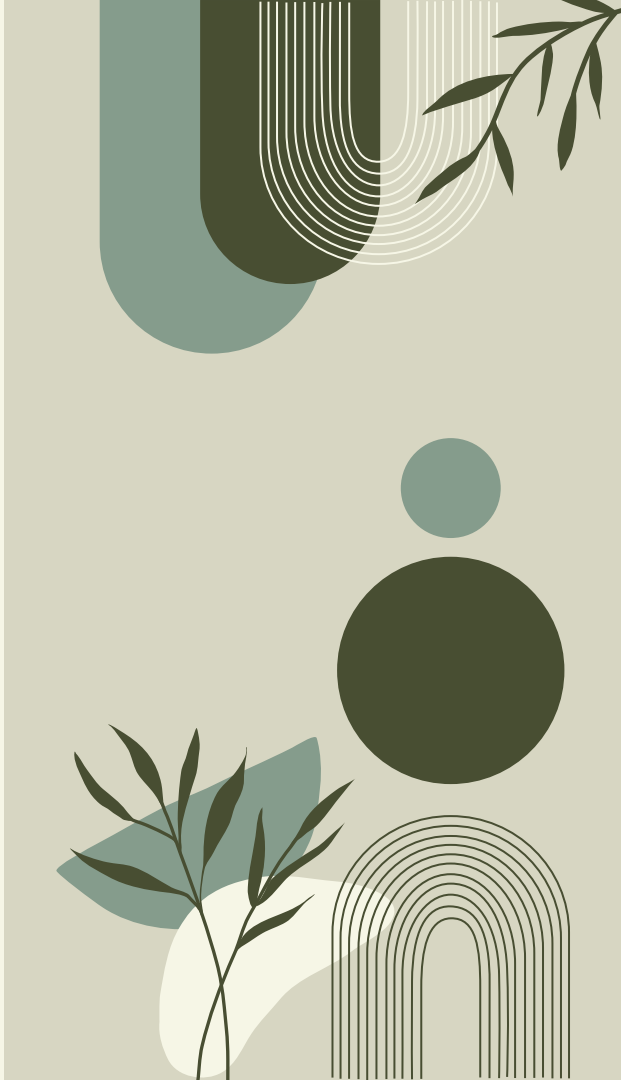
Intelligent QA agent about crisis events

By Mikail Syed, Patrick Cross, Sean Scott, Aditya Singh, Maokun Zhana
CS4624: Multimedia/Hypertext
Dr. Farag
Virginia Tech, Blacksburg VA 24061
10/23/2024



Outline

01 | Motivation & Purpose

02 | Features

03 | Live Demo!

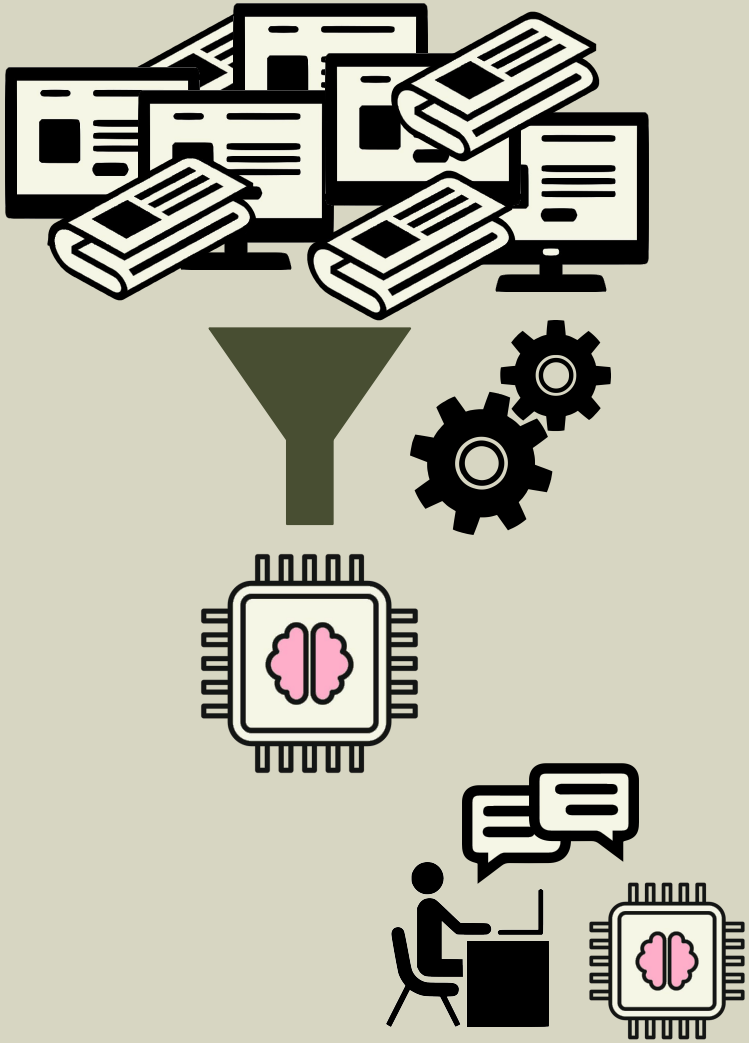
04 | System Walkthrough

05 | Challenges + Lessons Learned

06 | Future Features

07 | Acknowledgements

The Problem



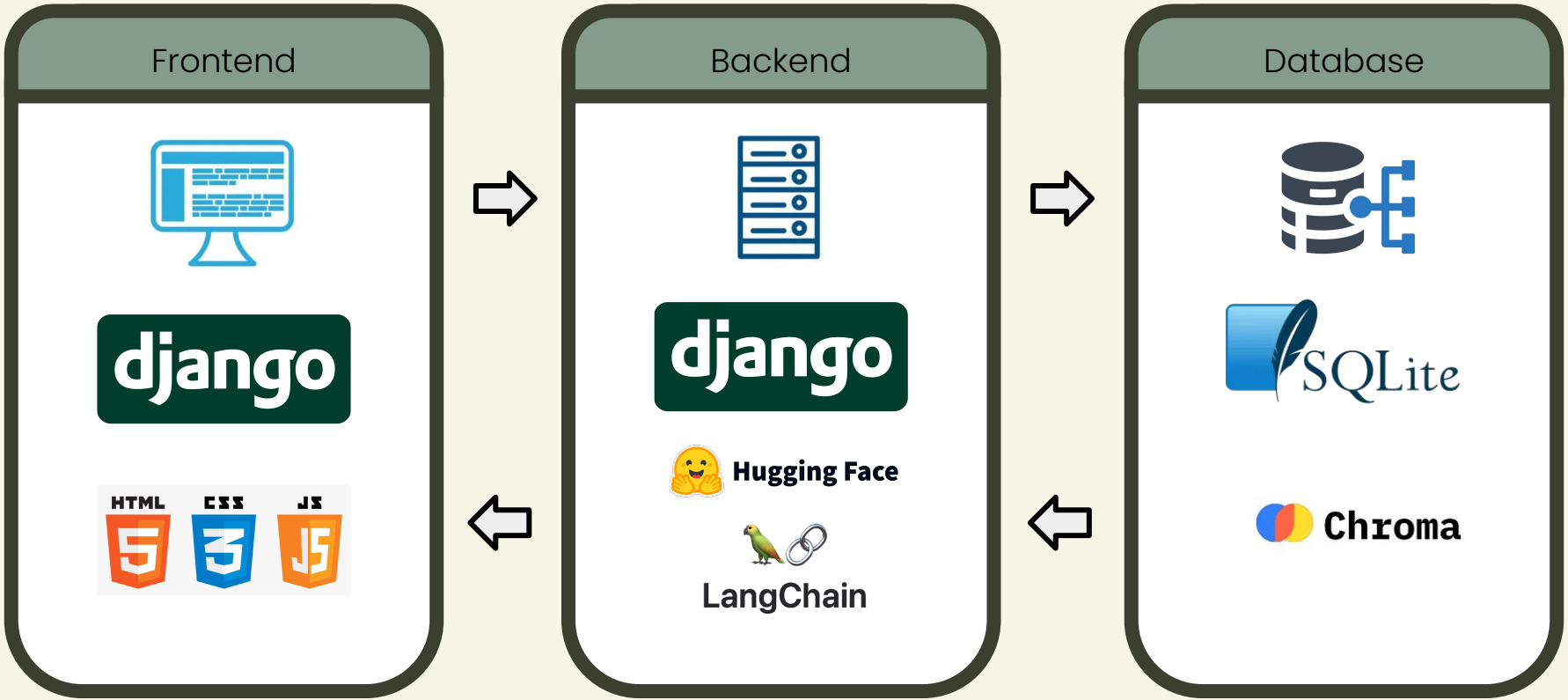
Our Solution

Features

1. User Accounts
2. Collections
3. Document + URL upload
4. Interactive Chatbot

LIVE DEMO

Tech Stack



Database Implementation

MySQL implementation

- Uses Django ORM (Object Relational Mapping)
- Key Models: Users, Collections, Documents, Chat Sessions

ChromaDB implementation

- Vector database for RAG implementation
- Uploaded documents' content processed into vector embeddings, which is stored in ChromaDB

Each collection can contain multiple documents, and each document is linked to its vector embeddings.

LLM Implementation

- Integrated multiple local LLM models (TinyLlama, RoLlama2, Capybara Hermes)
- RAG architecture (retrieval of relevant documents for answering user queries)
- Used Langchain for implementation, from document processing to LLM interactions
- Modular design allows for easy switching between models
- Maintains conversation history for future queries
- Models are downloaded only when first requested, making the system more efficient

Django Implementation

- Backend of the program manages all interactivity within the website
- Has a user authentication (login, signup, logout) functionality
- Allows for users to own Collections of Documents with full CRUD adherence and functionality
- Allows for users to go within the Collections and perform CRUD operations on individual Documents
- Documents and Collections are managed in SQLite, Django’s default database

Challenges

Technical

- Prompt Engineering
  - Small Model Sizes
  - Improper Responses
- Merging RAG and web issues – caused redesign of project

Non Technical

- Collaboration
  - Team Communication
  - Goal Setting

Future Features!

Small

- Containerization with Docker
- Improved UI
- Revised prompt engineering for better output
- More LLMs

Large

- Chat/Collection export functionality
- LLM download and selection management
- Add a web scraper to find relevant URLs

Acknowledgements

Client - Dr. Mohammed Farag

- Dr. Farag is our main point of contact for this project and laid out the guidelines and specifications for our design
- Dr. Farag's research interests include include intelligent transportation systems, connected/automated vehicles, C-V2X, machine learning, large-scale data analysis, large-scale system analysis and design, big data, and information retrieval, which will help us immensely for our project

Thank you!

Do you have any questions?