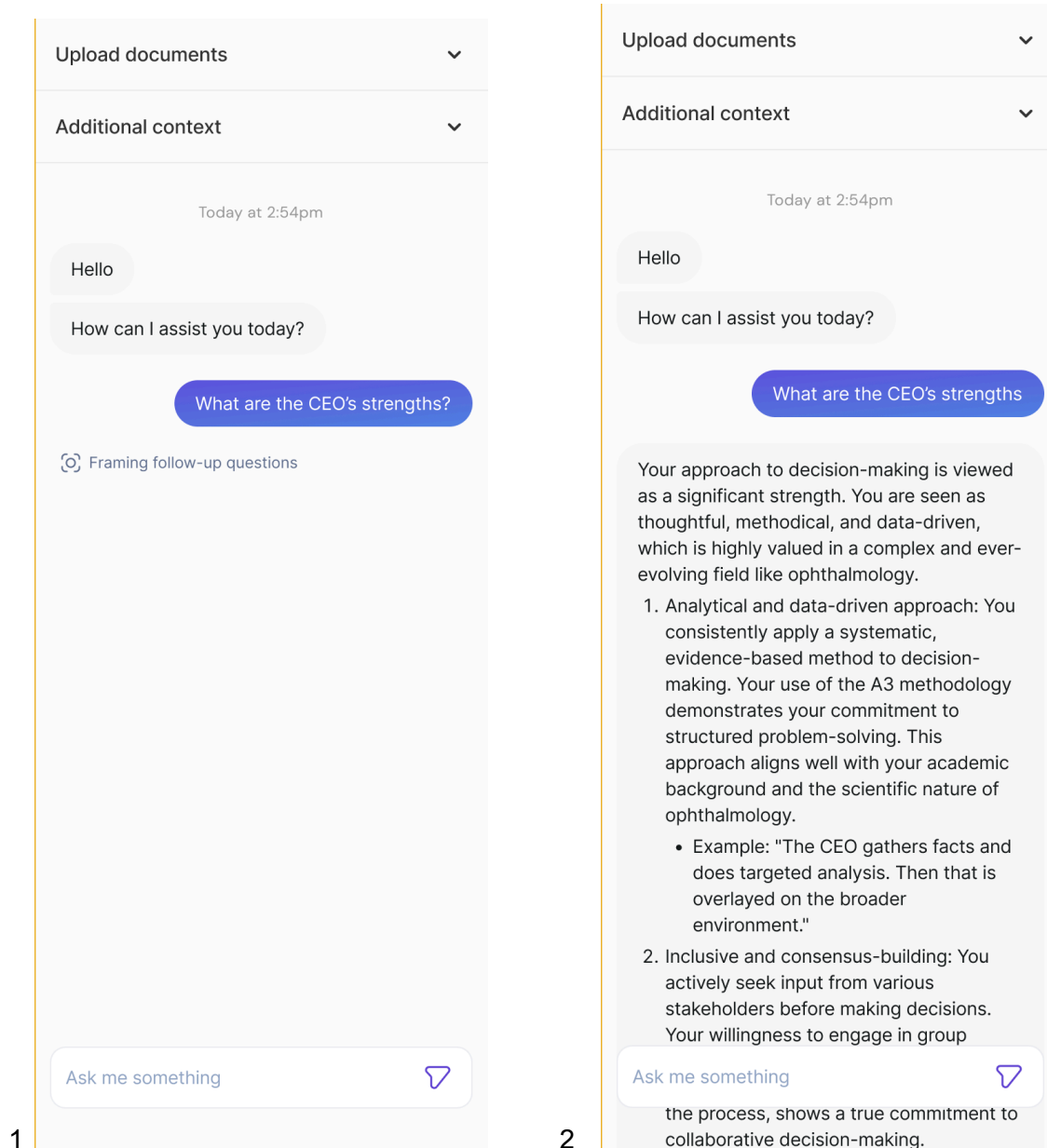


AI + Full stack or mainly Full stack

Problem statement:

Create an application from which users can chat with PDF and SQLite databases simultaneously.

Screens (for Inspiration):



3

Upload documents

org chart.docx

CEO self-evaluation.pdf

Note: A.I. uses these documents to frame all answers to your specific needs. Examples may include org charts, what you are looking for, meeting notes, etc.

Tech stack:

Backend:

Fast API

Frontend:

React/Next JS

Model:

OpenAI/Gemini/llama

Details:

There is a static database which has the following three tables:

Users table:

```
user_id, survey_id
```

```
Example:
```

```
user_id,survey_id
```

```
1,101
```

```
2,101
```

```
3,102
```

```
4,102
```

```
5,103
```

Survey table:

```
survey_id, question_id, question
Example:
survey_id,question_id,question
101,1,"How satisfied are you with our service?"
101,2,"Would you recommend our service to others?"
102,3,"How easy was it to use the product?"
102,4,"How likely are you to use the product again?"
```

Responses table:

```
user_id, survey_id, question_id, response
Example:
user_id,survey_id,question_id,response
1,101,1,"Very satisfied"
1,101,2,"Yes"
2,101,1,"Satisfied"
2,101,2,"Maybe"
3,102,3,"Easy"
3,102,4,"Very likely"
4,102,3,"Difficult"
4,102,4,"Unlikely"
5,103,1,"Neutral"
```

Also, there is an option to upload a pdf. You can store the pdf in the server directory only for simplicity or in the cloud like AWS s3 etc.

The user should be able to chat with the chatbot in the natural language. The chatbot can fetch important data from the pdf and the SQLite data to respond to the user's query depending upon the context.

- Use open AI or any open-source models to create the chatbot. The chatbot should be able to respond to the user's query corresponding to both the SQL data and PDF data.
- You can use SQLite data and keep the file in the server folder structure
- You can keep uploaded PDF files in the server folder or use any cloud storage provider like S3, etc.

Evaluation:

1. How efficiently you can create the APIs
2. How efficiently do you come up with the flow and the architecture
3. Accuracy of the chatbot is not important, but the flow of the architecture is.
4. Quality and cleanliness of the code and overall code structuring.

Examples.

- What are the top five amazing facts mentioned in the pdf?
- How many people took the survey with survey id 2
- What are the general answers of the people for the survey with question id 4 of survey id 4?

Deadline:

24th Saturday EOD

Deliverable:

Send the GitHub repo to the following email:

saurabh@evva360.com

And CC to:

asifrahaman162@gmail.com

You can create a mono repo or a separate repo both end and backend