Premier League Chat - Proof of Concept

Premier League Chat - Proof of Concept 1
- **Objective** 2
- **Requirements** 2
  - High-level requirements 2
  - Assumptions 2
  - Features & Ideas 2
  - Other Requirements 3
- **Approach** 3
- **Solution** 5
  - Architecture & Technologies 5
  - Application / Agent Flow 7
- **Results & Evaluation** 8
  - Example Results 9
- **Deliverables** 11
- **References** 11

Objective

Implement a simple POC that can take a natural language prompt from a user and return accurate squad information for Premier League teams upon a natural language request.

Requirements

High-level requirements

- Input: natural language user request
- Output: the First name, Surname, Date of Birth, and playing position of each player on the requested squad from the Premier League
- Integrate with the external API
- use LLMs

Assumptions

- The app will return the squad only for the current season, 2024/2025
    - correctly returned by the API
- The app will return only senior team members

Features & Ideas

- imple chat user interface - *Done*
- The app should only answer questions about the squads of Premier League clubs - *Done*
- The app should handle typos and unclear club names - *Done*
- Users can interact with the app in different languages - *Partially Done*
- The user can ask questions about specific squad groups, for example: “Who are the defenders of Manchester United?” - *Done*
- The user can ask about the squads of two teams - *Not Implemented*
- The user can ask about a squad using a player’s name, for example: “Who are Harry Maguire’s teammates?” - *Not Implemented*

All ideas for improvement are collected in the GitHub board: https://github.com/users/Laxmar/projects/1

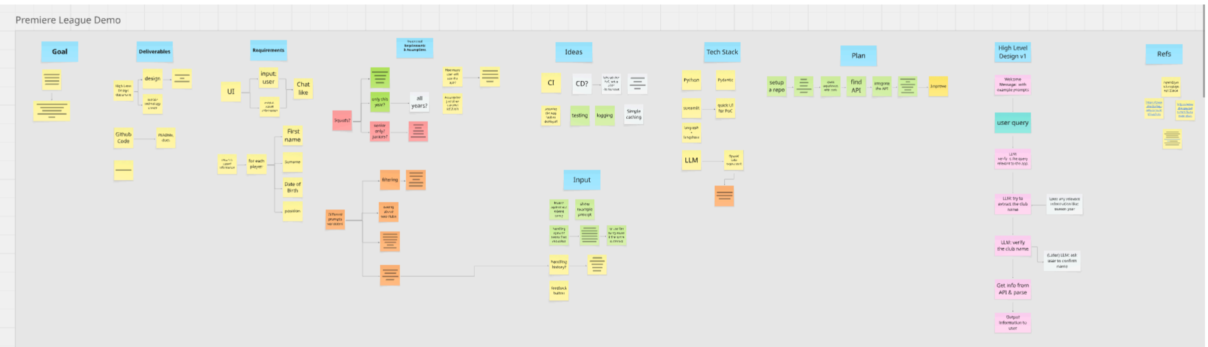
Other Requirements

For the purpose of the proof of concept, other requirements and architectural drivers were not considered. No technical constraints or quality attributes were taken into account.

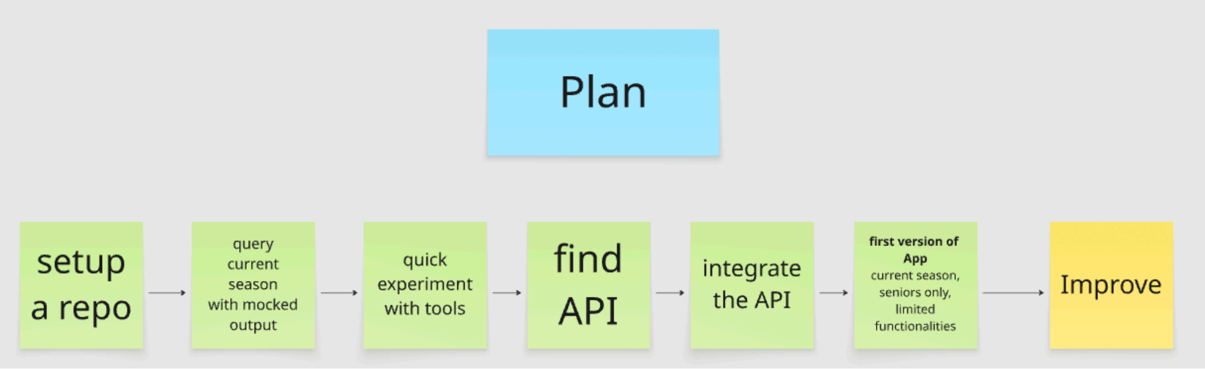
Approach

First, I gathered the requirements and my own ideas, and created a plan along with a high-level design. For this, I used a Miro board. The image is shown below, and the PDF file is available in the repository:

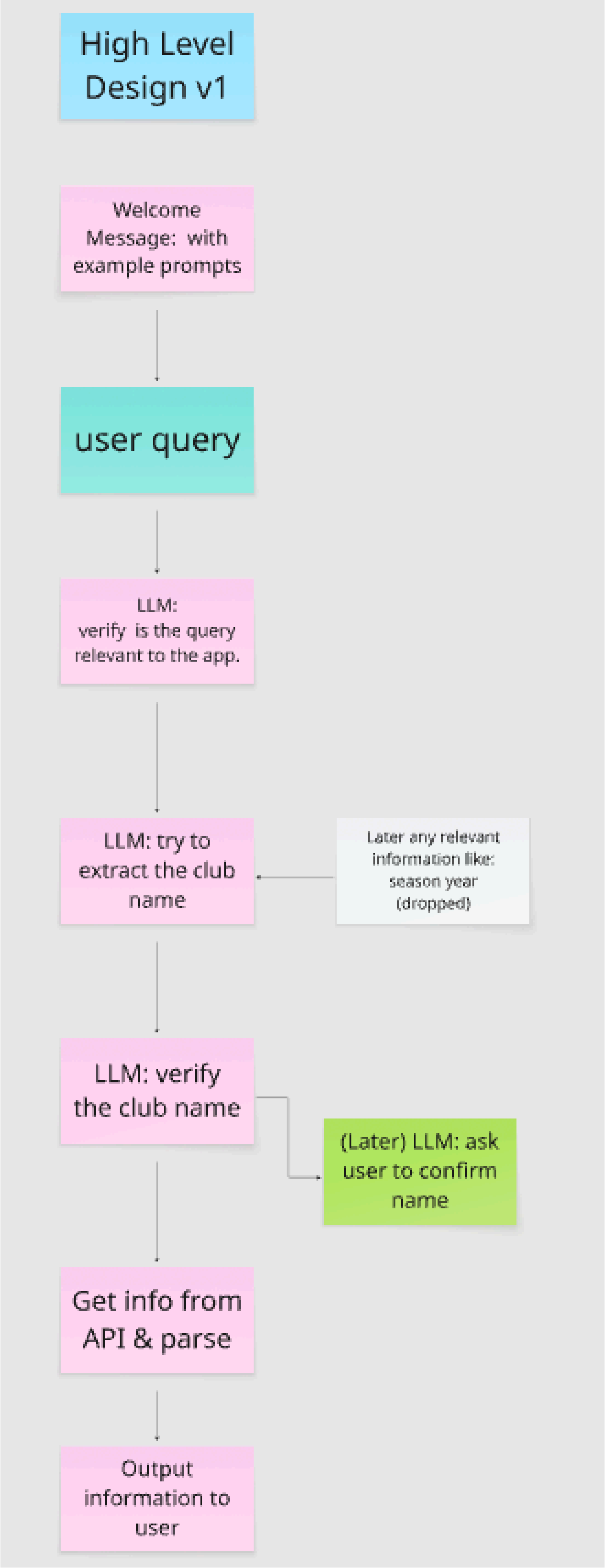
https://github.com/Laxmar/PremierLeagueChat/tree/main/docs/miro_approach.pdf



The initial high-level plan is shown below. Later, I defined and tracked tasks using the project's GitHub board. https://github.com/users/Laxmar/projects/1.

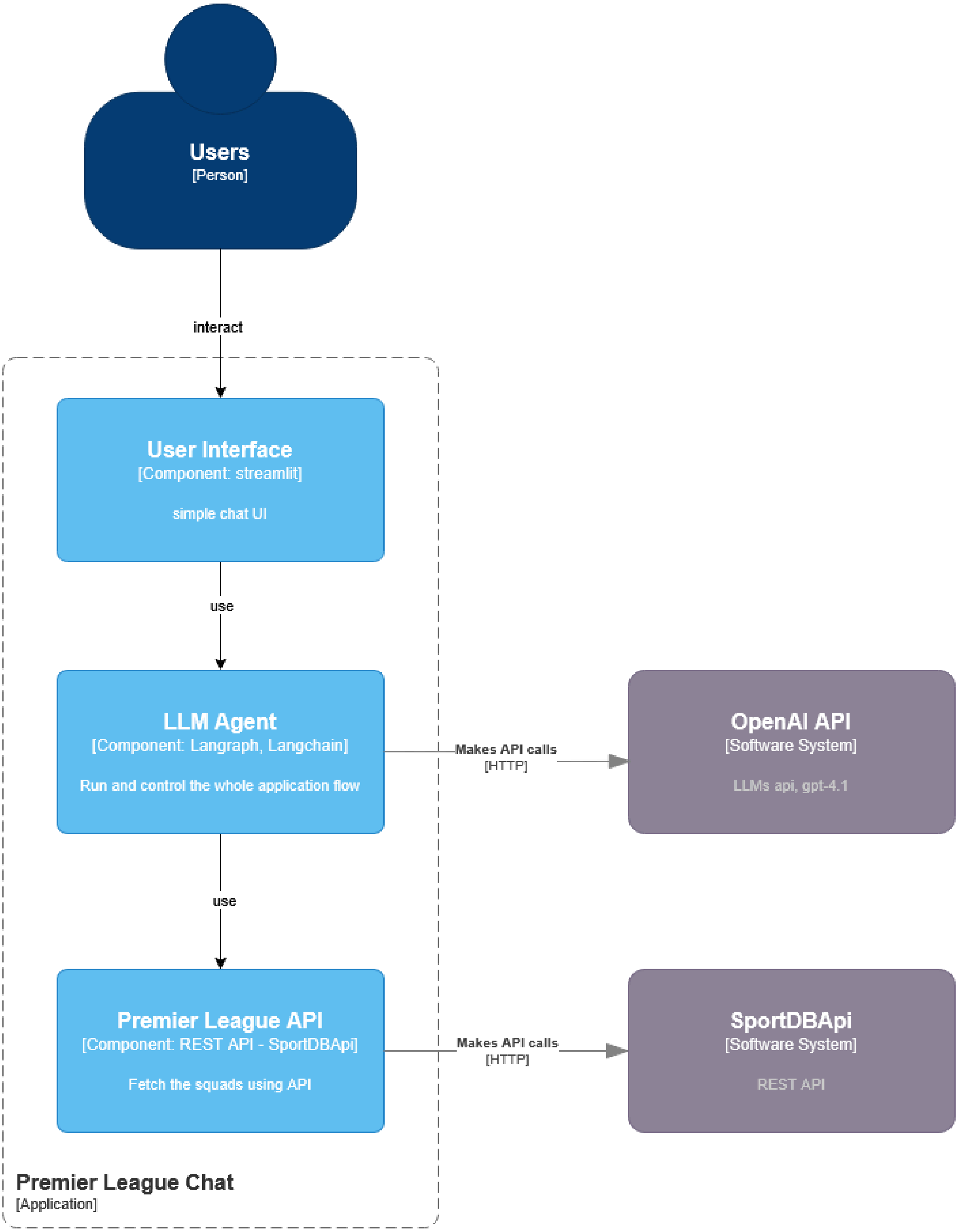

Plan
setup a repo
query current season with mocked output
quick experiment with tools
find API
integrate the API
first version of App
current season, seniors only, limited functionalities
Improve

After very quick experimentation with the tools. I decided on to simpler approach, which gives me more control over LLM and is easier to test. I decided to quickly implement the simplest approach and iteratively improve it. The high-level idea is to split the whole flow into small, simple stages. (The details will be explained in the next section.)


High Level Design v1
Welcome Message: with example prompts
user query
LLM: verify is the query relevant to the app.
LLM: try to extract the club name
Later any relevant information like: season year (dropped)
LLM: verify the club name
(Later) LLM: ask user to confirm name
Get info from API & parse
Output information to user

Solution

Architecture & Technologies



The application, shown above, consists of three main components. In the code, these are divided into the backend (Agent and Premier League API) and the frontend (UI app).

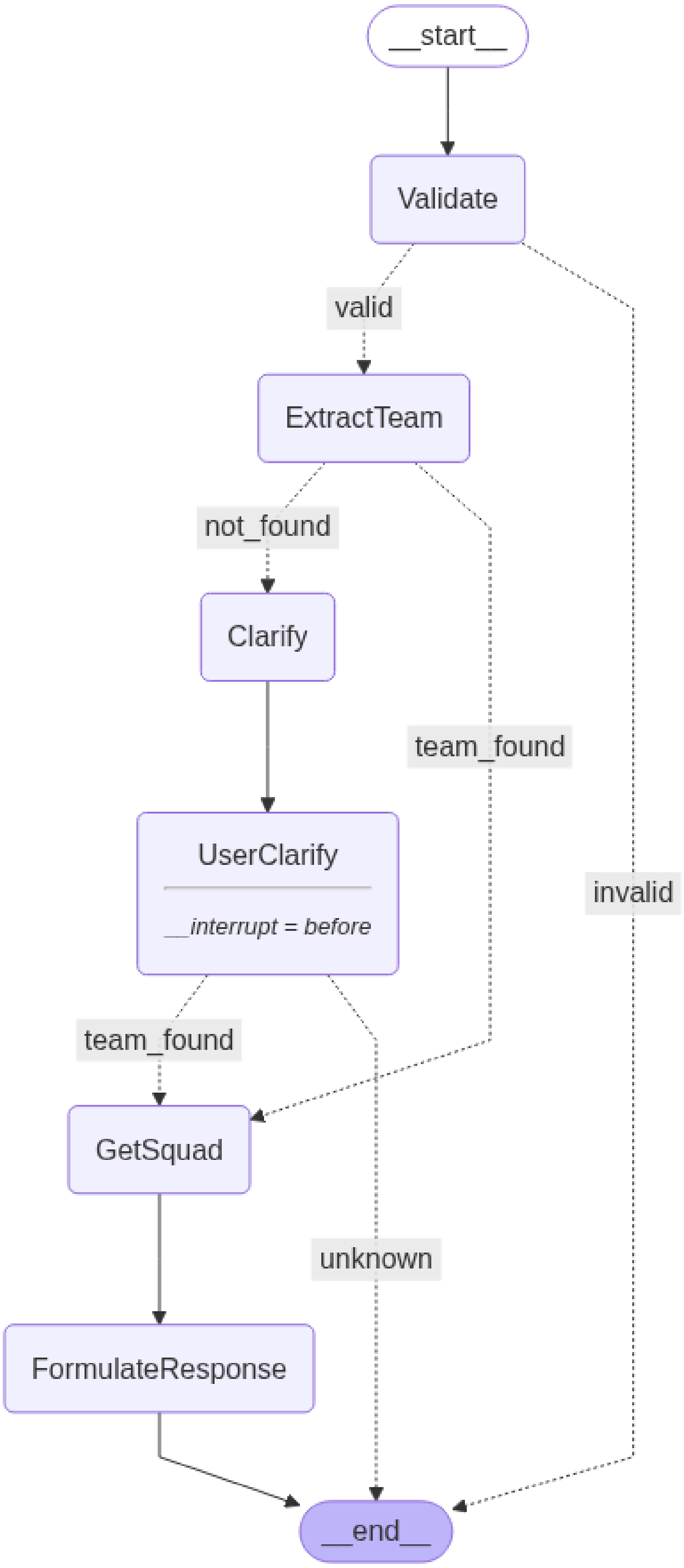
Main technologies:

- **Python** - my first choice of language.
- **Streamlit** - a simple UI framework for Python. For the proof of concept, it was my first choice as it allows quick implementation, easy improvements, and fast iterations.
- **Langraph & LangChain** - mature frameworks for handling LLM flows and agents.
- **OpenAI & GPT-4.1** - my first choice of api, but it can easily be swapped for another API. I also tested o4-mini, however, it sometimes hallucinates. I believe a smaller, faster model could be used, but I haven't focused on optimization. (to avoid premature optimization).
- **TheSportDB API** - I checked several APIs and also considered scraping data from websites, but based on the requirements, I decided to integrate this API.
    - Simple API & cost-effective - a good fit for the proof of concept
    - The API also provides endpoints to search for team and player names, which may be useful for future features. More: https://github.com/users/Laxmar/projects/1/views/1?pane=issue&itemId=115212492&issue=Laxmar%7CPremierLeagueChat%7C3

Other technologies:

- Pytest
- Pyright - type checking
- Ruff - formatter & linter
- Loguru - logger
- CI - GitHub actions (currently unused)
- deployment - probably streamlit platform (soon)

Application / Agent Flow



The implementation is chat-like, but it only handles messages up to retrieving a single team's squad. According to the Langraph architecture, the flow is represented as a graph consisting of nodes and edges. (presented above)

Nodes

- **Validate** - It validates if the user query is about a Premier League team squad.
- **ExtractTeam** - It tries to extract the team name from the user query. If the team is not found, it sets team_found to False.
- **Clarify** - If the team is not found, it asks for clarification. It tries to guess the most likely team name from the user query, knowing all Premier League clubs. It creates a clarification request, which is shown to the user.
- **UserClarify** - before this node, the app will show the clarification request. This node analyzes the user's answer. It tries to get the team name based clarification request and a user clarification response.
- **GetSquad** - get the squad using the API
- **FormulateResponse** - it creates the final answer using the squad information and the user query

Results & Evaluation

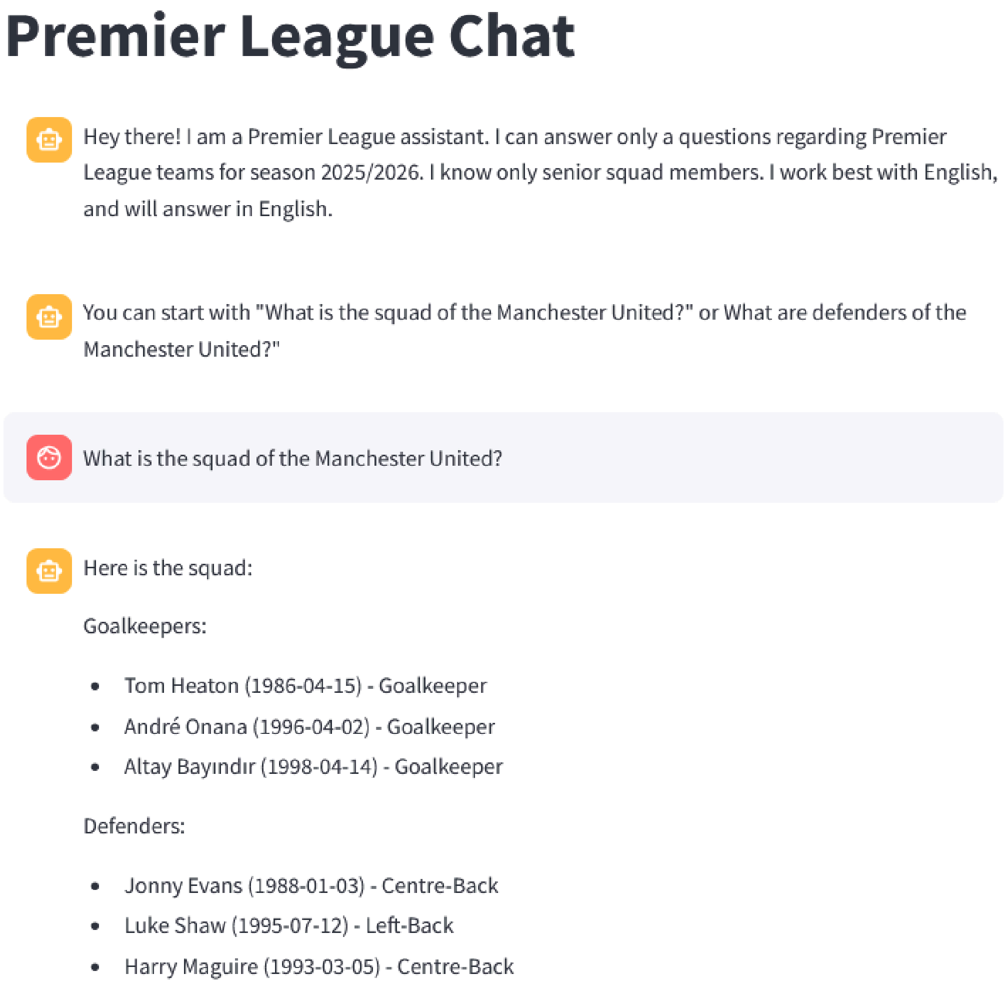
Based on the defined features, I created a set of evaluation queries. I split the queries into categories and defined a set of queries for each category. The categories are:

- BASE_USER_QUERIES
- IRRELEVANT_USER_QUERIES
- NOT_PREMIER_LEAGUE_TEAMS_QUERIES
- PRECISE_USER_QUERIES
- UNCLEAR_TEAMS_QUERIES
- DIFFRENT_LANGUAGES_QUERIES

(https://github.com/Laxmar/PremierLeagueChat/tests/evaluation_queries.py)

The results can be found in evaluation_results.csv and the log file evaluation.log located in the test folder. The evaluation checks whether the entire flow works as intended. The final model has not been fully validated; currently, validation can only be performed manually by reviewing the logs. I also tested basic queries for all Premier League clubs to verify the requirements.

Example Results



Premier League Chat

Hey there! I am a Premier League assistant. I can answer only a questions regarding Premier League teams for season 2025/2026. I know only senior squad members. I work best with English, and will answer in English.

You can start with "What is the squad of the Manchester United?" or What are defenders of the Manchester United?"

What is the squad of the Manchester United?

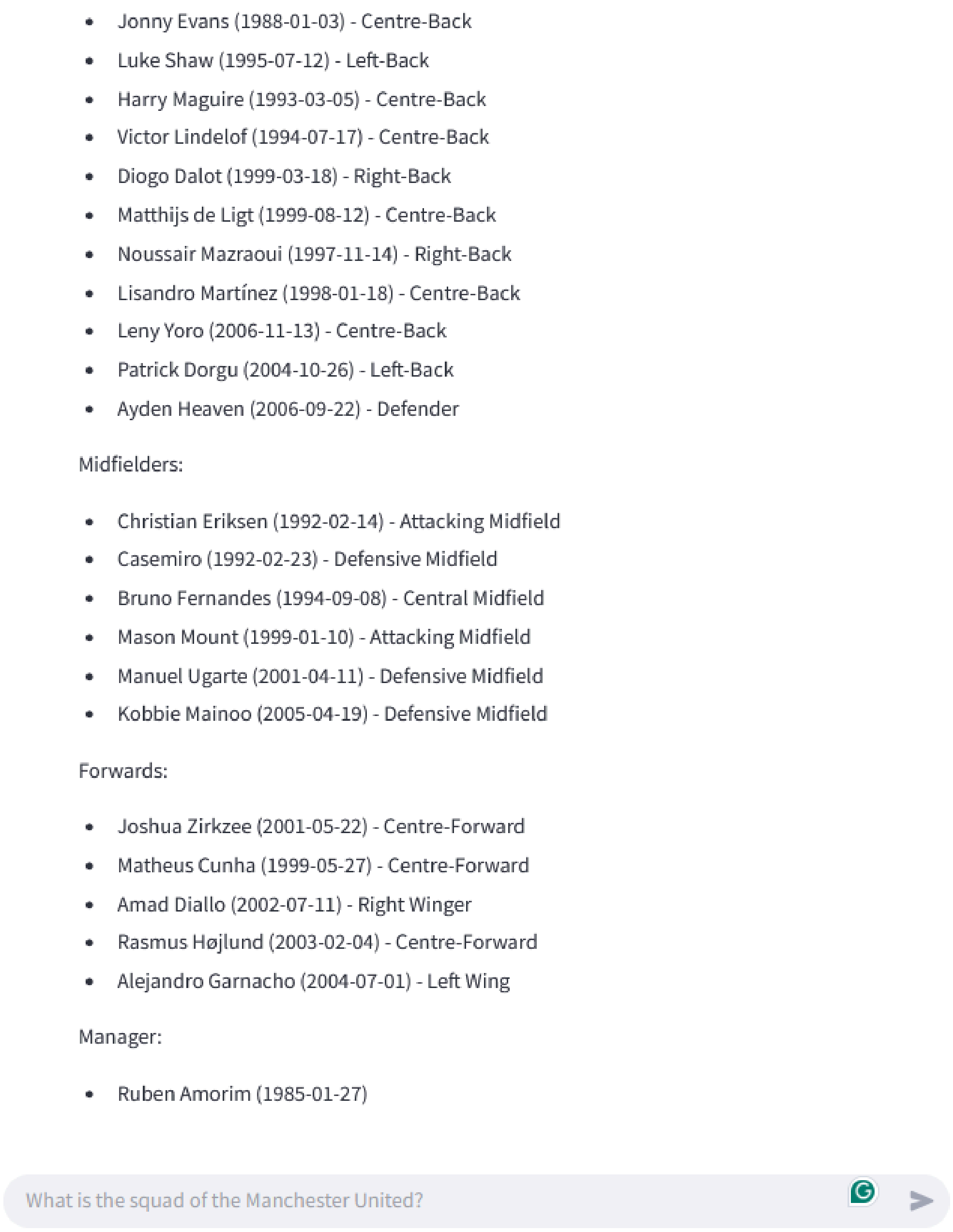
Here is the squad:

Goalkeepers:

- Tom Heaton (1986-04-15) - Goalkeeper
- André Onana (1996-04-02) - Goalkeeper
- Altay Bayındır (1998-04-14) - Goalkeeper

Defenders:

- Jonny Evans (1988-01-03) - Centre-Back
- Luke Shaw (1995-07-12) - Left-Back
- Harry Maguire (1993-03-05) - Centre-Back



- Jonny Evans (1988-01-03) - Centre-Back
- Luke Shaw (1995-07-12) - Left-Back
- Harry Maguire (1993-03-05) - Centre-Back
- Victor Lindelof (1994-07-17) - Centre-Back
- Diogo Dalot (1999-03-18) - Right-Back
- Matthijs de Ligt (1999-08-12) - Centre-Back
- Noussair Mazraoui (1997-11-14) - Right-Back
- Lisandro Martínez (1998-01-18) - Centre-Back
- Leny Yoro (2006-11-13) - Centre-Back
- Patrick Dorgu (2004-10-26) - Left-Back
- Ayden Heaven (2006-09-22) - Defender

Midfielders:

- Christian Eriksen (1992-02-14) - Attacking Midfield
- Casemiro (1992-02-23) - Defensive Midfield
- Bruno Fernandes (1994-09-08) - Central Midfield
- Mason Mount (1999-01-10) - Attacking Midfield
- Manuel Ugarte (2001-04-11) - Defensive Midfield
- Kobbie Mainoo (2005-04-19) - Defensive Midfield

Forwards:

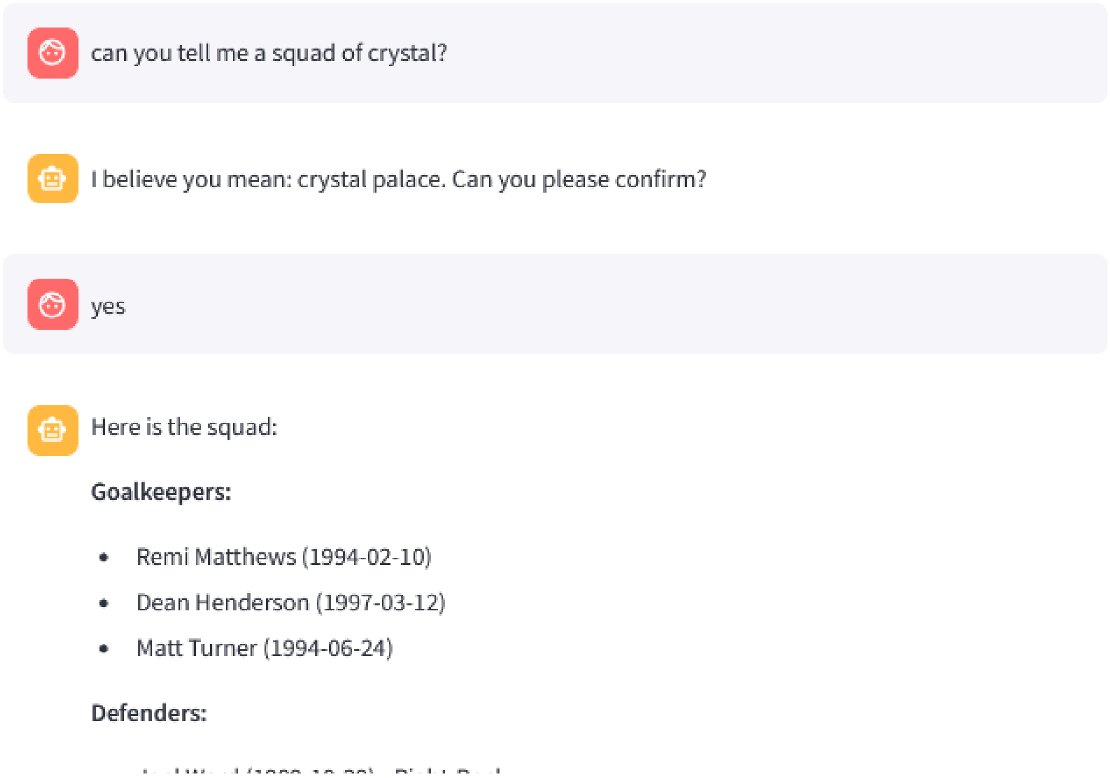
- Joshua Zirkzee (2001-05-22) - Centre-Forward
- Matheus Cunha (1999-05-27) - Centre-Forward
- Amad Diallo (2002-07-11) - Right Winger
- Rasmus Højlund (2003-02-04) - Centre-Forward
- Alejandro Garnacho (2004-07-01) - Left Wing

Manager:

- Ruben Amorim (1985-01-27)

What is the squad of the Manchester United?

2nd example with unclear name



can you tell me a squad of crystal?

I believe you mean: crystal palace. Can you please confirm?

yes

Here is the squad:

Goalkeepers:

- Remi Matthews (1994-02-10)
- Dean Henderson (1997-03-12)
- Matt Turner (1994-06-24)

Defenders:

Deliverables

- High-level design document (pdf)
  - This document, https://github.com/Laxmar/PremierLeagueChat/tree/main/docs
- Proof of concept (code) Public GitHub repository with all associated code, comments, documentation and diagrams related to the assignment for all parts of the assignment attempted.
  - https://github.com/Laxmar/PremierLeagueChat
  - https://github.com/Laxmar/PremierLeagueChat/tree/main/docs

References

- Repository: https://github.com/Laxmar/PremierLeagueChat
- The Github Board: https://github.com/users/Laxmar/projects/1/views/1
- The sports DB API: https://www.thesportsdb.com/
- Link and access to the demo will be shared after deployment.