Pre-Interview Coding Challenge

Challenge 3: UI - Chatbot

Objective

Create a chatbot that enables user to select any of 3 Stock Exchanges (LSEG, NASDAQ, NYSE) and for the selected exchange provides 5 stocks that are traded in that exchange.
User can then select any one of those stocks to view the latest stock price.

Requirements

- Create a chatbot using any preferred language/platform
- Home Menu must consist of 3 Stock exchanges (LSEG, NASDAQ, NYSE)
- Stock Menu - Upon selecting any of them, provide 5 stock names (different for each exchange)
- Upon selection of any stock name, display its current value, and again provide Stock Menu (options to select 5 stocks)
- Provide options to go to Home Menu at any point in time.

Data & Inputs

Input data of stock names and price values is provided in json format in the file “Chatbot – stock data.json”. An indicative layout is given here, but you are free to improvise.

Output Format

A chatbot UI with the above Home Menu and Stock Menu with intuitive navigation. You can ship your solution to us or host it in any of your preferred platforms and share the access details. Also provide the source code via a GitHub Repository as outlined in the Section “Submission” below.

Error Handling

The application should gracefully handle exceptions, such as no files, empty files etc., feel free to include as much exception handling as possible. It provides insights into your ability to anticipate what can go wrong.

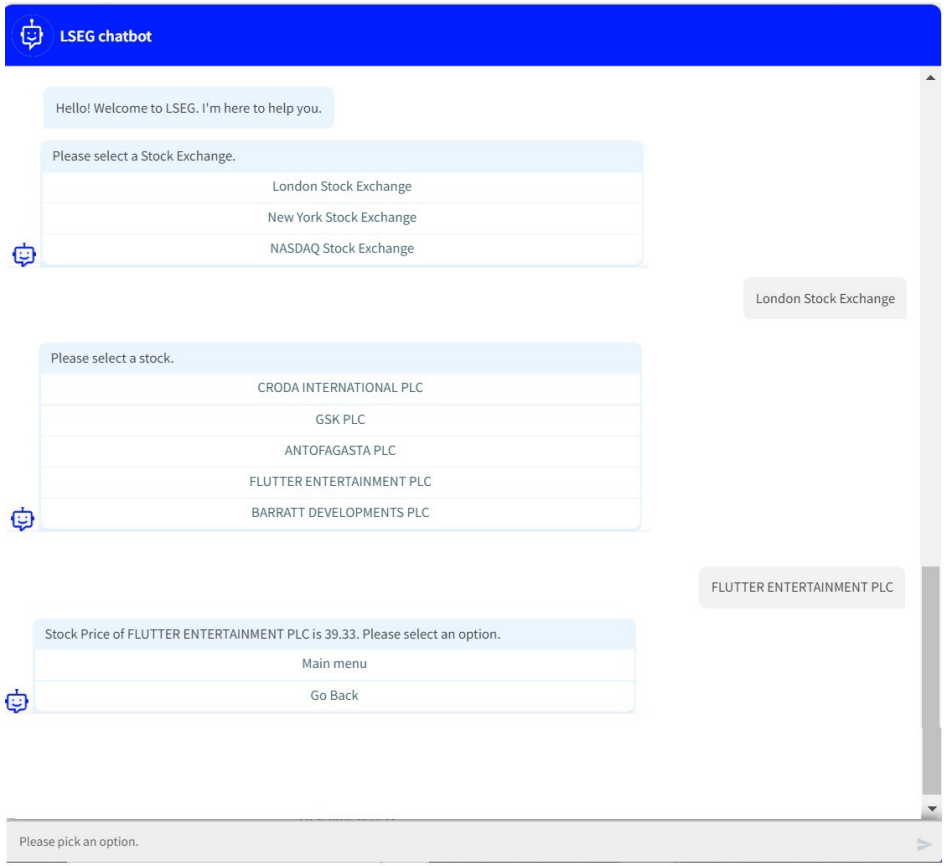
Documentation

Include a README file explaining how to set up and run your application.

Optional Enhancements

Feel free to add enhancements that could improve the extensibility/maintainability for future enhancement, user experience etc., Some suggestions include:

- additional functionality or checks
- value added user experience features
- optimizations for performance and scalability



Submission Instructions & FAQ

How to submit your challenge

Please submit your code via a publicly accessible GitHub repository created specifically for this challenge.

- It is expected that you will know how to create a repository on GitHub and check your code into it.
- Do not share your repository with anybody else other than the reviewer.
- You should ensure your repository is read-only to everybody except yourself, this is so nobody can interfere with your submission after you have completed it. .
- The code for the challenge should be the only code in the repository except for the README file, explaining how to set up and run your application, and providing any access related information.
- Estimated time spent on the challenge should be ~2hours, if you do not finish , include any additional information in the README file.
- Ensure your code is well-commented and follows good coding practices.

Frequently Asked Questions

Q: Can I use external libraries or frameworks?
A: Yes, you can use any libraries or frameworks you find appropriate.
Q: Is it required to deploy the application online?
A: No, it's not required. However, if you choose to do so, please include access details in your README.
Q: What if I have questions during the challenge?
A: Document any assumptions you make in your README file.
Q: Are there any specific coding standards or practices I should follow?
A: While there are no enforced coding standards, your code should be readable, well-organized, and demonstrate good software development practices.
Q: Is it okay to use code snippets or libraries from the internet?
A: Yes, but ensure you understand and can explain any code you use. Also, respect code licensing and give credit where due.
Q: What will this challenge be assessing?
A: We are looking at your coding skills, problem-solving abilities, creativity, and how you approach and structure a project.
Q: Can I use online resources for help?
A: Yes, you can use online resources but make sure you understand and can explain any code you use. Plagiarism will disqualify your submission.
Q. Can I use AI generated code to complete the challenge?
A. No, the challenges are designed to test your skills and understanding of the problem presented, if we detect AI generated solutions it will be treated the same as we would treat “Plagiarism”.
Q: For the UI related exercise, in case of time crunch should I prioritize look and feel or the functionality?
A: Functionality has higher precedence than look and feel, in case you have to compromise on one of them.