

ChatRecipeey

Advanced Programming

By:

Seif Muhammed Wafikk 22-101153

Abdelrahman Emad Shahwan 22-101034

Yousief Mohamed Khadr 22-101148

Ali Wael Hallaba 22-101000

Contents

Contents

| | |
|----------------------------------------|-----------|
| Introduction..... | 3 |
| What is Chatbot?..... | 3 |
| Historic Outline | 3 |
| Importance/Advantages of Chatbot..... | 4 |
| How do Chatbots work?..... | 4 |
| Types of Chatbots..... | 4 |
| Motivation | 5 |
| Analysis and Design | 6 |
| Modules..... | 6 |
| Hierarchy Chart | 6 |
| Description..... | 7 |
| Theory..... | 7 |
| Explanation of rules | 7 |
| Workload distribution..... | 8 |
| Seif Wafik | 8 |
| Abdelrahman Shahwan..... | 8 |
| Yousief Khadr | 8 |
| Ali Hallaba | 9 |
| Test Cases | 10 |
| Sample Screens..... | 11 |
| Challenges & Problems | 12 |
| Conclusion | 13 |
| References | 14 |

Introduction

What is Chatbot?

A chatbot is a computer program that simulates and processes human conversation (either written or spoken), allowing humans to interact with digital devices as if they were communicating with a real person.

Historic Outline

1960s - 1970s:

- Early chatbots like ELIZA and PARRY emerge, laying the foundation for conversational agents.

1980s - 1990s:

- Advancements in expert systems and natural language processing lead to more sophisticated chatbots like A.L.I.C.E.

2000s - Present:

- Rapid AI and machine learning advancements drive the development of intelligent chatbots.
- Major companies adopt chatbots for customer service and virtual assistance.
- Voice-activated virtual assistants like Siri and Alexa become mainstream, further popularizing conversational interfaces.

Importance/Advantages of Chatbot

Chatbots offer businesses a multitude of advantages. They provide 24/7 availability, ensuring users can access support anytime, and their scalability allows them to handle multiple conversations simultaneously, making them ideal for large user bases. By automating tasks, chatbots reduce the need for human agents, resulting in cost savings. Instant responses improve customer service, leading to faster issue resolution and higher satisfaction. Additionally, chatbots analyze user data to provide personalized interactions, delivering tailored responses and recommendations. They efficiently retrieve information from databases, helping users find answers faster, and engage users in interactive conversations, guiding them through processes. Moreover, chatbots are accessible across platforms and devices, catering to a wide range of users.

How do Chatbots work?

A chatbot is an automated conversational AI that pretends to be human and carries out programmed tasks based on specific triggers, responding through a web or mobile app. These triggers can be defined as “Pattern Matching” in Functional Programming terms

Types of Chatbots

Rule-Based Chatbots: These chatbots follow predefined rules and patterns to generate responses. They are relatively simple and are often used for specific tasks, such as answering frequently asked questions or providing basic customer support.

AI-Powered Chatbots: These chatbots utilize artificial intelligence (AI) and natural language processing (NLP) techniques to understand user input and generate more intelligent responses. They can learn from interactions with users and improve over time.

Motivation

We chose the topic of cooking assistance and recipe generation because, as four individuals with limited experience in cooking, we recognize the need for guidance and support in the kitchen. With our lack of expertise, we often find ourselves unsure about what to cook or how to prepare meals. Therefore, developing a chatbot dedicated to cooking will not only benefit others who share our culinary challenges but also provide us with the necessary tools and resources to explore and enjoy cooking in our own homes.

Analysis and Design

Modules

All of the following are “.scala” modules that are the building blocks for the main.scala

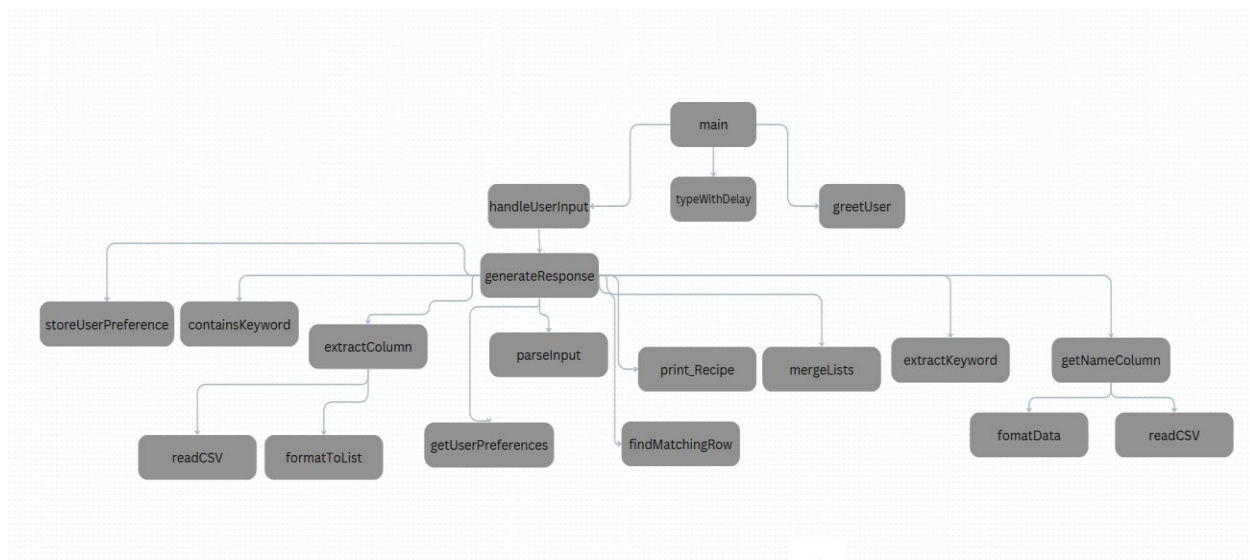
CSVHandler : Contains functions related to reading and processing CSV file “Recipes.csv” and convert what it’s read to suitable data structures that will be used in the other modules.

RecipeHandler : Contains functions related to processing and printing recipes.

UserInputHandler : Contains functions related to parsing user input and generating responses.

PreferencesManager : Contains functions related to storing and retrieving user preferences.

Hierarchy Chart



Description

Theory

In our chatbot's theory, the process begins with parsing the user's input, which involves breaking down the input into understandable components. These components are then compared with keywords extracted from our dataset file, which contains information about various recipes. By matching the parsed query with keywords associated with different recipes, we can identify the most relevant recipe to suggest to the user. The chatbot then retrieves and presents the recipe that best matches the user's input. Additionally, our chatbot has the capability to save user preferences if they express interest in a particular recipe, allowing for personalized recommendations in future interactions.

Explanation of rules

In our chat rules, we've implemented pattern matching to handle specific user interactions. For instance, if the user greets the chatbot with "hello" or "hi", the chatbot responds with a greeting. If the user's input contains keywords related to recipes, the chatbot generates a recipe suggestion. Additionally, if the user expresses liking or loving a recipe, the chatbot saves their preference for future interactions. These rules ensure that the chatbot responds appropriately to various inputs and enhances user engagement.

Workload distribution

Seif Wafikk

- Cleaning the Dataset “Recipes.csv” file.
 - Using python kernel
- Modularized the main source file to separate “.scala” files.
- Created all the functions in the CSVHandler.scala
 - **readcsv** : reading the csv file
 - **formatToList** and **formatData** to change the dataset style to List() “scala list type”
 - **extractColumn** and **getNamesColumn** : extracting the data from csv file by columns
 - **mergeLists** : to merge the keywords and the name for the recipe to ease the search
- Created the main.scala

Abdelrahman Shahwan

- Created all functions in the RecipeHandler.scala
 - **findMatchingRow**: finds the best matching row with keywords given
 - **typeWithDelay**: for the keyboard typing style
 - **print_recipe**: to print the ingredients and the steps
- Updated **generateResponse** function
- Updated **handleUserInput** function
- Updated main.scala

Yousief Khadr

- Created all the functions in the PreferencesManager.scala
 - **storeUserPreference** : to store user preferences
 - **getUserPreferences** : to retrieve user preferences
- Updated **handleUserInput** function

- Updated **parseInput** function

Ali Hallaba

- Created all functions in UserInputHandler.scala
 - **generateResponse** : it returns the recipe and save preferences
 - **parseInput**: breaks the query to words
 - **handleUserInput** : handles the chatbot's conversation.
 - **containsKeywords**: checks if the user's query contain keyword used before
 - **extractKeyword**: extract the keyword from the query

Acknowledgments

A special thanks to Eng. Nourhan El Naggar for her invaluable guidance and support throughout this project.

Test Cases

Test Case Title: Verify Chatbot Response to Valid Input

Input: "What are the ingredients for lasagna?"

Expected Response: Chatbot provides a list of ingredients for making lasagna.

Input: "Can you recommend a vegetarian recipe?"

Expected Response: Chatbot recommends a vegetarian recipe.

Test Case Title: Verify Chatbot Response to Invalid Input

Description: Verify that the chatbot handles invalid input gracefully and provides appropriate error messages.

Steps:

Input: "Xyz123"

Expected Response: Chatbot responds with a message indicating that no recipes with this input.

Input: ""

Expected Response: Chatbot responds with a message indicating that he didn't understand.

Test Case Title: Verify Chatbot Performance Under Load

Steps:

Simulate concurrent user interactions with the chatbot.

Measure the response time for each interaction.

Ensure that the chatbot responds to each interaction within a specified time threshold.

These are just a few examples of test cases. Depending on the complexity of your chatbot and the requirements of your project, you may need to include additional test cases covering various scenarios, error conditions, and performance metrics. Each test case should clearly outline the input, expected response, and any specific conditions or requirements for validation.

Sample Screens

```
User : X+2
No Recipes Found
ChatRecipee: Anything else ?
User :
ChatRecipee: I'm sorry, I didn't understand that. Can you please try again?
User : i want recipe for bamia
```

Recipe: Bamia

Ingredients:

- 1 1/2 teaspoons olive oil
- 2/3 cup sliced white onion
- 2 teaspoons minced garlic
- 1 1/2 cups chopped fresh tomatoes
- 1 teaspoon fresh lemon juice
- 2 teaspoons chopped fresh parsley
- 2 teaspoons coriander
- 1/2 teaspoon ground cardamom
- 1/4 tablespoon fresh ground black pepper
- 1 pinch salt
- 1 1/2 lbs small okra pods (fresh)

Steps:

1. Heat oil in a medium sized heavy saucepan over medium heat.
2. Sauté onions for 2 minutes.
3. Add the garlic and sauté for 1 minute.
4. Stir in all the remaining ingredients except the okra.
5. Bring the mixture to a hard simmer

6. Fill the okra and slice into 1/2 inch rounds and stir in.
7. Simmer for 10 minutes-serve hot over rice.

ChatRecipee: Anything else ?

User : thank you

ChatRecipee: You're welcome! How can I help you further?

User :

User : goodbye

ChatRecipee: Goodbye! Have a great day!

PS D:\University\Spring 2024\Advanced Programming\Project\chatbot\src>

Challenges & Problems

In our project, we encountered several challenges and problems that affected our development process. One significant issue was the limitation of available libraries and resources in Scala 3, making it difficult to import and utilize external libraries for tasks like reading CSV files. As a result, we had to resort to using the basic reader provided by Scala, which presented challenges in handling CSV data effectively. We found ourselves creating numerous functions dedicated to extracting specific information from the CSV, leading to a more complex and cumbersome codebase.

Another challenge we faced was managing user preferences, particularly in saving and loading this data efficiently. We struggled to implement a robust solution for persisting user preferences across sessions, which impacted the personalization aspect of our chatbot.

Additionally, we encountered difficulties with recursion when processing large datasets. Due to performance concerns, we opted for using loops instead of recursion, especially for operations involving significant amounts of data. While this approach helped mitigate performance issues, it introduced its own set of complexities and trade-offs.

Overall, these challenges posed significant obstacles during the development of our project, requiring us to devise creative solutions and workarounds to overcome them. Despite these difficulties, we persevered and successfully delivered a functional chatbot solution, albeit with some limitations and compromises.

Conclusion

While we're proud of our ChatRecipeey project, time constraints prevented us from implementing a graphical user interface (GUI). We faced challenges in finding and importing external libraries compatible with Scala 3, which impacted our development process. However, despite these setbacks, our project has highlighted the potential of chatbot technology in the culinary domain. ChatRecipeey offers personalized recipe suggestions and cooking assistance, showcasing the value of chatbots in simplifying tasks and sparking creativity in the kitchen. Moving forward, we're inspired to further explore and refine the chatbot concept, building on the lessons learned from this project to drive future innovations in conversational AI.

References

Oracle, "What is a Chatbot?," [www.oracle.com](https://www.oracle.com/eg/chatbots/what-is-a-chatbot/), 2022

<https://www.oracle.com/eg/chatbots/what-is-a-chatbot/>

Goetz, E. (2017). Chatbots: The Ultimate Guide. Chatbots Magazine

<https://chatbotsmagazine.com/the-complete-beginner-s-guide-to-chatbots-8280b7b906ca>

Tomašev, N., & others. (2019). Chatbot: Friend or Foe? In *AI and Nature*. Springer.

Beneteau, E. (2019). What Are Chatbots and How Do They Work? HubSpot.