HOLY SPIRIT UNIVERSITY OF KASLIK

SCHOOL OF ENGINEERING

ROBOCHEF

BRING PEPPER BACK TO THE KITCHEN

Completed by
THE PEPPER MILLS

Anthony El Chemaly (202100079)
Catherina El Khoury (202101204)
Rodaina Fayad (202000680)
Elie El Rayess (202000006)
Fawzi El Khoury (202100741)
Garo Margossian (202000133)

Presented To: Br. Elie SAAD
Course: GIN456- Human-Robot Interaction

USEK – Fall 2023-2024

Table of Contents

Table Of Figures: 3
Acknowledgement: 4
1. Introduction: 5
1.1. Why pepper mills? 5
1.2. Project Description: 6
1.3.Project Objectives: 6
2. Implementation Plan: 8
3. Research Paper: 9
3.1. Research Paper Analysis: 9
3.1.1. Questions and Hypothesis: 9
3.1.2. Methods and Procedures: 10
3.1.3. Results and Findings: 13
3.2. Future Prospects 14
4. Use Cases and Personas: 15
4.1. Use Cases: 15
UC-01 Request a Recipe 16
UC-02 Recommend a Recipe 18
UC-03 Create Account 21
UC-04 Provide Nutritional Information for a Specific Recipe 24
UC-05 Play music or Tell a joke 26
UC-06 Monitor kitchen appliance 28
4.2 Personas: 30
Persona 1: Samira, Stay at home Mom. 31
Persona 2: Sally, a Professional Chef Dreamer 33
Persona 3: Anastasia, an Untrained Chef 34
5. Technical Part: 35
5.1. Folder Structure 36
5.2. Init Project Block 37
5.3. Clear Memory Block 38
5.4. Subscribe to Event Block 39
5.5. Dialogue 43
6. Discussion and Conclusion: 44
7. Prototype: 46

Table Of Figures:

Figure 1 Pepper Mills 5
Figure 2 RoboChef 6
Figure 3 Work Flowchart 9
Figure 4 Cooking Assistant Robot: conversation history between the cooking person and the robot 12
Figure 5 Sample of Recipe Data 13
Figure 6 Structure of Recipe Data 13
Figure 7 Evaluation Items of System Viewpoint 13
Figure 8 Evaluation Items of solve issues during cook 14
Figure 9: Project Root: Blocks Used 35
Figure 10: Folder Structure 37
Figure 11: Init Project Block 38
Figure 12: Clear Memory Data 39
Figure 13: Subscribe to events block structure 40
Figure 14: ALMemory Data Fetching in Python WebView Script 40
Figure 15: Code Snippet Responsible for WebPage Navigation based on AL Memory Changes 41
Figure 16: Cuisine and Mealtype QiChat Snippet 41
Figure 17: JS Function to raise an event 42
Figure 18: QiChat Dialogue Snippet 42
Figure 19: Display Info JS Function Used to Display All Information 42
Figure 20: Entertainment Proposal 43
Figure 21: Play Song Behavior used in ^run 43
Figure 22: Step Proposal 44

Acknowledgement:

We would like to express our sincere gratitude to the Holy Spirit University, Kaslik, for providing us with the opportunity to undergo this course and for facilitating all the necessary requirements, including the Pepper robot - the key component of our learning experience.

A special thanks goes to Br. Elie Saad for generously sharing his experience and knowledge, guiding us through the essential steps to construct a complete interaction with Pepper.

We are deeply appreciative of the collaborative efforts of the Pepper Mills team, which worked as one consistent family, working together to achieve an outstanding end project.

Lastly, we sincerely hope that you find merit in our work, a result of the hard work invested throughout the semester.

1. Introduction:

1.1. Why pepper mills?

Figure 1 Pepper Mills

We chose the name "Pepper Mills" for our team because it beautifully reflects our unique vision of integrating technology and culinary artistry. Just as a pepper mill grinds and enhances the flavor of a dish, our robot Pepper will assist and elevate the kitchen experience. It symbolizes the perfect blend of innovation and tradition, making every meal a masterpiece. "Pepper Mills" embodies our commitment to redefining the future of cooking with a touch of creativity and automation.

1.2. Project Description:

RoboChef is our project for the Pepper Robot combining technology with the art of cooking. It allows the user to enjoy cooking again with Pepper's assistance. Our RoboChef offers various recipes, step-by-step instructions, cooking tips, and an interactive human-robot relationship which makes her a flexible friendly assistant that aims to change the way chefs approach cooking, making it an entertaining, enjoyable, and interactive journey.

Figure 2 RoboChef

1.3.Project Objectives:

- ✓ *User-Friendly Robot:* Pepper will display options to search for recipes, view cooking tips, and adjust serving sizes. By that, users can easily navigate through the application with minimal effort.

- ✓ *Recipe Database and Step-by-Step Instructions*: RoboChef will own an extensive recipe database, categorized by cuisine, meal type, dietary preferences, and cooking time where Pepper will guide

chefs through each recipe with clear voice-guided instructions and relevant images ensuring an easy-to-follow cooking process.

- ✓ *Voice-guided Instructions and Serving Sizes:* Users can interact with Pepper through natural language commands or questions related to the recipe, and Pepper will respond accordingly. They can also provide Pepper with the number of servings they intend to prepare and RoboChef will automatically recalculate ingredient quantities to match the selected serving size.

- ✓ *Ingredient Substitution Suggestions:* RoboChef will suggest ingredient substitutions based on the ingredients available in the kitchen, allergies or dietary restriction. Chefs will receive recommendations for suitable replacements without affecting the recipe's essence.

- ✓ *Cooking Tips, Techniques and Nutritional Information:* The Project will include a section where Pepper shares cooking tips and techniques to improve users culinary skills and knowledge. In addition, RoboChef will provide nutritional information for each recipe, including calories, macronutrients, and micronutrients, allowing users to make informed dietary choices.

- ✓ *Favorites and History:* Users can mark recipes as favorites for easy access and view their cooking history, making it simple to revisit and recreate their preferred dishes.

- ✓ *Integration with Smart Devices and Assistance:* We are currently studying the feasibility of designing Pepper to interact with smart kitchen devices, such as ovens and blenders as needed, directly through the application and help with basic cleaning tasks like sweeping up crumbs or spills and meal preparation by doing some physical tasks.

- ✓ *Entertainment:* While cooking, Pepper can entertain the user by playing music and telling jokes or other ways we might consider in the future.

2. Implementation Plan:

1. *Application Development:*

Engage our team of computer engineers and one biomedical engineer to design and develop the RoboChef

project, supporting the outlined objectives.

2. *Recipe data collection:*

Collaborate with online culinary experts and chefs resources to verify and provide various, high-quality

content and accurate recipe database.

3. *Voice Integration:*

Work with Pepper voice recognition to ensure smooth interaction between users and PepperChef.

4. *Testing:*

Apply testing phases to ensure the project's functionality, its usability, accuracy, and performance. Address any identified issues and optimize the application accordingly.

5. *User Feedback for Improvement:*

Get feedback from a group of users who experienced RoboChef to gather insights. Utilize this feedback to

enhance and refine our project before presenting.

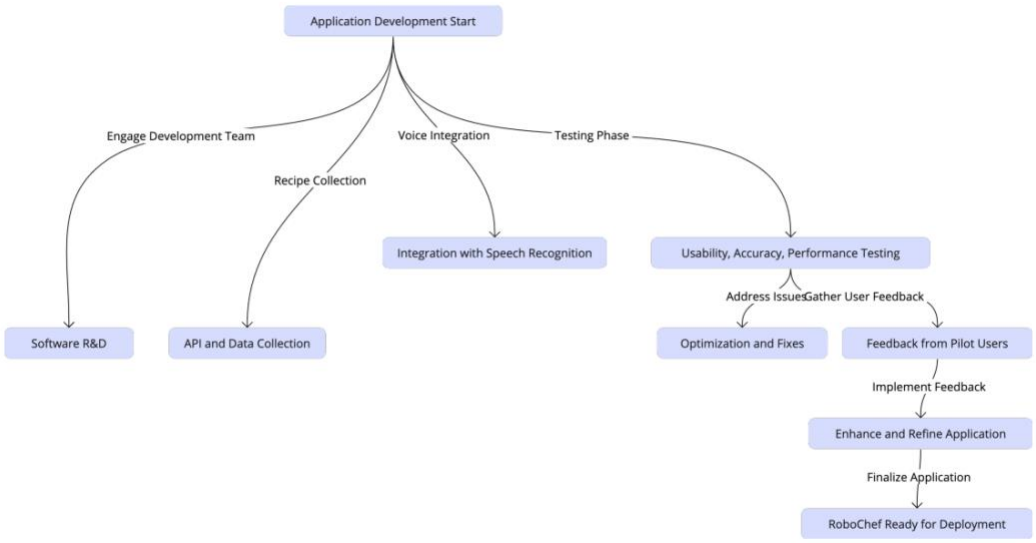


Figure 3 Work Flowchart

3. Research Paper:

We chose the paper *"Cooking Assistant Service Utilizing an Interactive Robot"* by S. Kumagai et al. (2017 IEEE/SICE International Symposium on System Integration) for its direct relevance to our project.

The paper explores the hypothesis that integrating an interactive robot as a cooking assistant service can address challenges faced by users of digital devices for cooking, including loneliness during cooking, difficulties operating devices with dirty hands, and challenges in recording cooking progress. The hypothesis suggests that such integration can enhance the cooking experience, making it more engaging and enjoyable, while also simplifying device interaction and progress recording, ultimately improving user satisfaction and kitchen engagement, aligning closely with our project goals.

3.1. Research Paper Analysis:

3.1.1. Questions and Hypothesis:

The hypothesis of this research is that utilizing an interactive robot as a cooking assistant service can address the following issues commonly experienced by individuals who use digital devices for cooking recipes:

1. Issue of Loneliness and Lack of Fun:
   The presence of an interactive robot during cooking

will enhance the cooking experience, making it more engaging and enjoyable.

2. Difficulty in Operating Devices with Dirty Hands:
   An interactive robot can provide a hands-free and voice-activated interface, eliminating the need for users to touch their devices with dirty hands while cooking.

3. Difficulty in Recording During Cooking:
   The robot can assist users in recording their cooking progress and making notes, simplifying the process of documenting the recipe and cooking steps.

The hypothesis suggests that the introduction of such a cooking assistant service will result in a more positive and efficient cooking experience, ultimately improving user satisfaction and engagement in the kitchen.

3.1.2. Methods and Procedures:

- ✓ Preliminary Experiment**:**

  - The preliminary experiment was conducted to confirm the functionality of the cooking assistant service and the interactive robot.

  - A specific cooking procedure was created where the cooking person and the interactive robot converse while preparing dishes. The cooking procedure for the egg sandwich was demonstrated.

  - Evaluation items were established to confirm the system's functionality and to address issues during cooking. Some of these evaluation items included checking if the user saw the recipe from the robot, whether the robot started the cooking instruction, and if the robot continued instruction until the end.

  - One evaluator cooked using the robot for this experiment.

  - After the cooking was finished, the evaluator evaluated specific items set for the experiment.

- The researchers first created a cooking procedure in which the cooking person and the interactive robot converse while preparing dishes.

- Evaluation items were established to confirm whether the system functioned properly and whether issues during cooking were resolved.

- The preliminary experiment was carried out using the above scenario and evaluation items.

✓ Implementation of Communication:

- The robot and the web server communicate using RSNP (Robot Service Network Protocol).

- The robot processes the recipe data in a structured manner to provide cooking instructions to the user.

- The robot receives voice interaction from the cooking person and provides cooking instructions. In the future, it is also assumed that the camera can be operated by voice and can move the camera by voice operation.

- The service has two access patterns:
  - The cooking person refers to the web service using a PC or mobile phone.
  - The cooking person refers to the web service using the robot.

"We use Speech Recognizer to implement the function of speech recognition. Speech Recognizer is a library of speech recognition…"

"We developed our robot's text-to-speech function by Google's text-to-speech engine. The reasons why we choose Google's engine are as follows: Firstly, text-to-speech engine works faster than text-to-speech web API because there is no network traffic. Secondly, it can work offline."

	Cooking Assistant Robot: "Make an egg sandwich"
	Cooking Person: "Next"
	Cooking Assistant Robot: "These ingredients are for one person. Please prepare a boiled egg"
	Cooking Person: "Next"
	Cooking Assintast Robot: "Please prepare 2 sandwich breads"

Figure 4 Cooking Assistant Robot:
conversation history between the cooking person and the robot

- ✓ Recipe Data:

  - The structure of the sample recipe data includes columns for the recipe name, image file name, description, number of people, ingredients, and how to cook.

  - Cooking recipe data was made using recipe data from the "Meal to Wake Up" campaign published by the Ministry of Agriculture, Forestry, and Fisheries on the web. The data was part of a campaign by the Ministry of Agriculture, Forestry, and Fisheries to expand rice consumption. For this research, 9 items were extracted and used for verification from a total of 1999 items.

  - This recipe data is stored in the database in the six columns.

    - The following table shows an **unstructured** sample of the recipe data that is to be structured before being used by the robot:

column name	contents
Ingredients	Autumn eggplant, 1 bottle, meat sauce, 50 grams
How to cook	1. Slice the eggplant thinly and heat it side by side without wrapping the heat-resistant container. (40 seconds with 600-watt microwave oven), Add meat sauce to 1. and mix well, then heat for a further minute, ready!

Figure 5 Sample of Recipe Data

- The following table shows the **structure** of the recipe data used by the cooking assistant robot:

column name	contents
Ingredients	Autumn eggplant, 1 bottle
	meat sauce, 50 grams
How to cook	1.Slice the eggplant thinly.
	2.Arrange the heat resistant containers without wrapping them.
	3.Heat for 40 seconds in a 600-watt microwave oven.
	4.Add meat sauce to heated eggplant and mix well.
	5.After heating for another minute, it is completed!

Figure 6 Structure of Recipe Data

3.1.3. Results and Findings:

✓ The next table shows whether the robot was able to correctly follow the instructions given to it and read the recipe aloud.

Evaluation Item	Result
1.Did you see the recipe from the robot	Yes / No
2.Whether the reading start button is displayed	Yes / No
3.Whether cooking instruction by the robot is started	Yes / No
4.Did the robot instruct cooking till finished cooking	Yes / No

Figure 7 Evaluation Items of System Viewpoint

- ✓ The upcoming table shows whether the robot was able to help the user solve any problems that they encountered while cooking.

Evaluation Item	Result
1.Whether loneliness is relieved by creating dishes while talking with the robot	Yes / No
2. Were you able to reduce the effort of referring to the recipe	Yes / No

Figure 8 Evaluation Items of solve issues during cook

3.2. Future Prospects

- ✓ For the experiment of actual production, the researchers plan to flow an animation of the specific cooking procedure and to change the contents instructed by the robot.

- ✓ The researchers also plan to implement a function that asks the robot "Are you going well?" and to implement cooking tips according to the reply.

4. Use Cases and Personas:

4.1. Use Cases:

Use cases are essential in such project as they illustrate practical scenarios showing how our cooking assistant robot, RoboChef, addresses specific challenges and enhances user experiences in the kitchen.

These scenarios guide development, ensuring features like hands-free and voice-activated interfaces meet real-world needs, making our technology a practical tool in the kitchen.

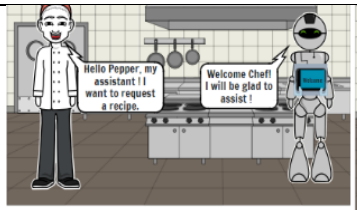
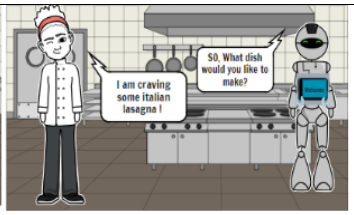
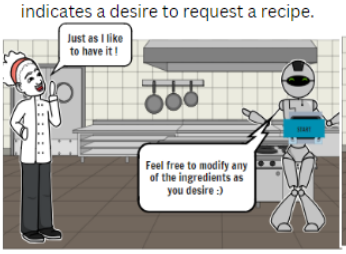
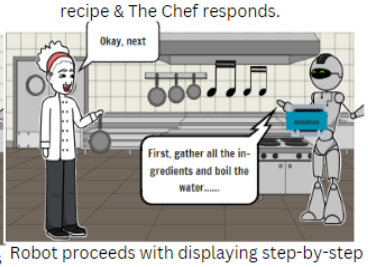
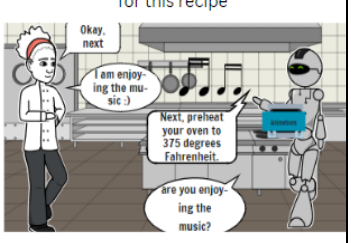
Each use case is organized with necessary components. It starts with a *use case ID and its name,* followed by *the acknowledgment (Author and Creation Date). Actors*, including users and RoboChef, are identified, and *triggers* define the initiating event. The *description* outlines the goals, while *preconditions* and *postconditions* set the state before and after the execution of the use case.

The *action sequence* details the scenario step by step. *Extensions* cover alternative paths, and *requirements* list essential functionalities. *Storyboards* visually illustrate the scenario to better understand the execution of the use case. *Priority* shows the importance of each use case, and *related use cases* show dependencies. *Assumptions* is what we expect from the users from this use case and *open issues* address what issues may be faced.

This structure ensures that every use case serves as a valuable guide, aligning RoboChef with practical kitchen needs and user experiences.

UC-01 Request a Recipe

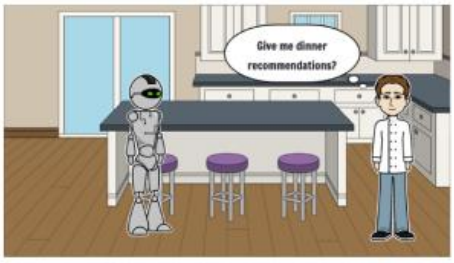
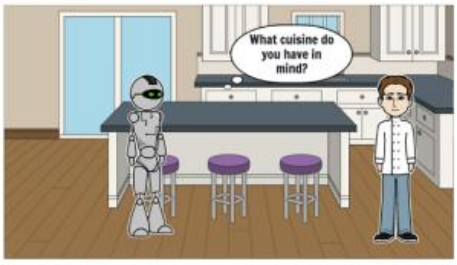
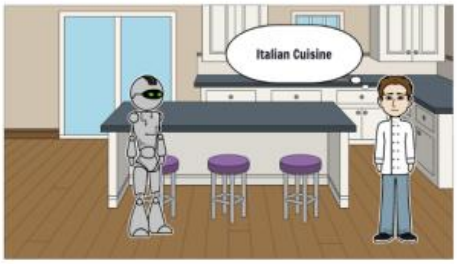
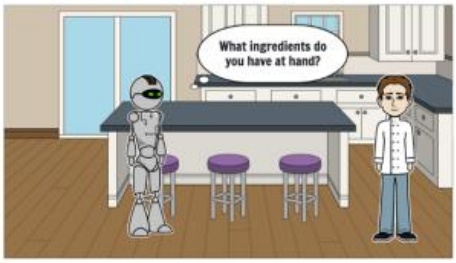
UC-ID and Name	**UC-01 Request a Recipe**		
Created By	Rodaina Fayad	**Creation Date**	12/10/2023
Actors	The main actors are Pepper the Robot (Culinary Cooking assistant) & person (Chef or cook).		
Trigger	The Chef indicates a desire to request a recipe of a specific food or dish from Pepper, The Robot.		
Description (Objectives/Goals)	*The user wants to receive cooking assistance from the Robot, seeking detailed, step-by-step instructions for a specific recipe, to enhance the chef's cooking experience.*		
Preconditions	PRECOND-1. The Robot is powered on, functional and connected to the recipe database. PRECOND-2. The Chef is recognized by the robot culinary assistant. PRECOND-3. The Chef logs into his account and is already registered with all required info using the registration form displayed on the screen.		
Postconditions	POSTCOND-1. A step-by-step instruction (with animation illustration) is returned and displayed on the screen accompanied with voice indications from the Robot.		
Action Sequence (Success Scenario)	1. The Chef indicates requesting a recipe from the Robot. 2. The Robot asks which specific dish and related cuisine are required. 3. The Chef then provides the name of the dish and related cuisine for the recipe. 4. The Robot displays all required ingredients for this recipe. 5. The Robot asks the Chef if they agree with the recipe or if they would like to change any of the ingredients. 6. The Chef confirms if they agree with the recipe. 7. Robot proceeds with displaying step-by-step instruction based on user input. 8. The Robot further explains each instruction using voice interaction. 9. Use case ends		
Extensions	**3a. No matching recipe is found:** • 3a1. The Robot alerts the Chef that there is no matching recipe available and suggests a similar recipe. • 3a2. The Chef chooses from suggested similar recipes or selects a different cuisine and returns to step 2 of the normal flow. **4a. The Chef wants to modify the ingredients:** • 4a1. The Robot allows the Chef to remove or replace specific ingredients and adjusts the recipe accordingly.		
Requirements	**Functional Requirements:**		

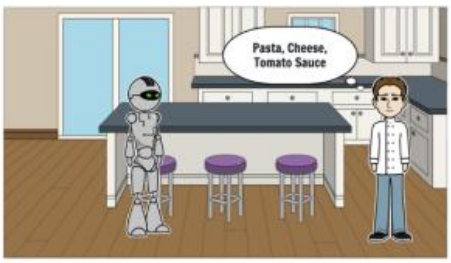
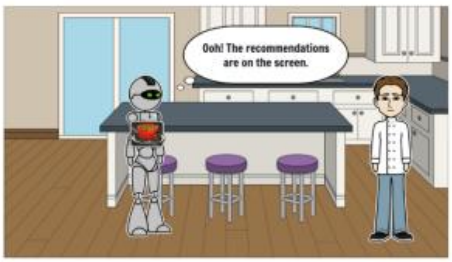
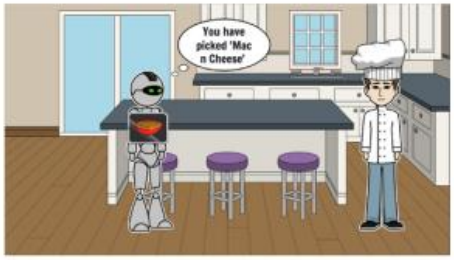
	1. **Recipe Database Interaction:** • Fetch recipes from a recipe database based on Chef's request. 2. **Displaying Ingredients and Instructions:** • Display required ingredients and step-by-step instructions with animations and voice indications. 3. **Recipe Modification:** • Allow Chef to modify recipe ingredients (remove/replace) and adjust the recipe accordingly. 4. **Handling No Matching Recipe:** • Alert Chef if no matching recipe is found and suggest alternatives **Non-Functional Requirements:** 1. **Usability:** • Provide a user-friendly interface for easy interaction. 2. **Performance:** • Respond to recipe requests swiftly and handle a high volume of requests. 3. **Reliability:** • Ensure system availability with minimal downtime and robust error handling.
Storyboards	 The Chef approaches Pepper and indicates a desire to request a recipe.  Robot asks about the required recipe & The Chef responds.  The Robot displays all required ingredients for this recipe  The Chef approves on given ingredients and presses start  Robot proceeds with displaying step-by-step instruction based on user input while playing some enternaining music  The Robot further explains each instruction using voice interaction.  All steps are done, the dish is ready to be served .
Priority	1

Related Use Cases	• Use case of creating an account and setting all health conditions • Use Case of Recipe Recommendation • Use Case of creating an enjoyable environment while cooking
Assumptions	• The Chef has a basic understanding of culinary terms and techniques. • Pepper the Robot has access to a comprehensive recipe database.
Open Issues	• Integration with external recipe databases for a wider range of cuisines. • Improving voice interaction for users with varying accents or speech patterns.

UC-02 Recommend a Recipe

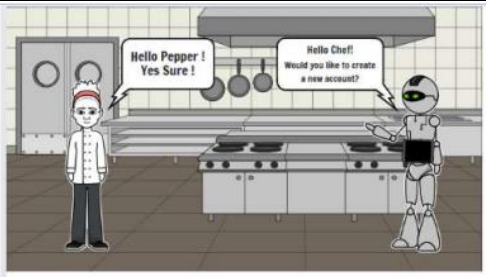
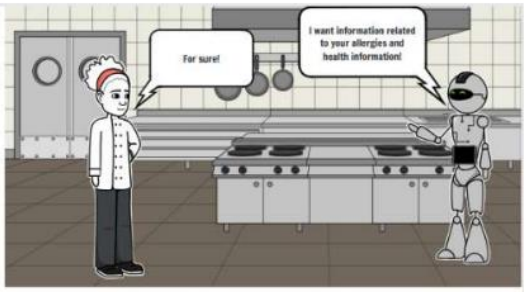
UC-ID and Name	**UC-02 Recommend a Recipe**		
Created By	Anthony El Chemaly	**Creation Date**	12/10/2023
Actors	The actors are Pepper, the culinary assistant robot, and the user (cook or chef).		
Trigger	User vocally expresses the need for a recipe suggestion.		
Description (Objectives/Goals)	*The user wants Pepper's recommendations to determine what to cook, considering specific requirements for their culinary preferences.*		
Preconditions	PRECOND-1: User is present in the vicinity of Pepper. PRECOND-2 Pepper is powered on, functional, and connected to the recipe database. PRECOND-3. User is logged into his/her account. PRECOND-4 User's preferences, saved previously, are accessible to Pepper		
Postconditions	POSTCOND-1. User is presented with a personalized recipe suggestion, along with details and alternative options. POSTCOND-2. User's feedback on the recommendation may be saved for future interactions.		
Action Sequence (Success Scenario)	1. User verbally asks Pepper for a recipe suggestion. 2. Pepper acknowledges the request. 3. Pepper scans its database, considering user preferences and dietary restrictions (filtering). 4. Pepper verbally communicates suggested recipes and displays them on its screen. 5. User can accept a recipe, reject, or ask for other suggestions.		

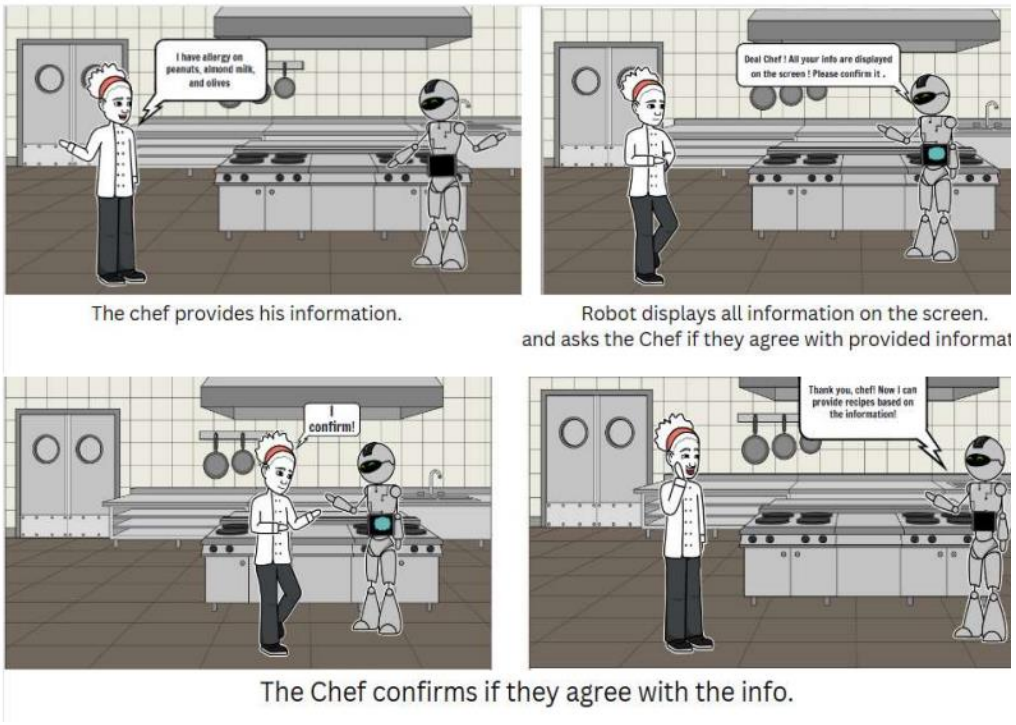
	6. If accepted, Pepper provides detailed instructions. 7. Use Case Ends.
Extensions	**A1. Unfavorable Recipe:** 1. User expresses dissatisfaction with the suggested recipe. 2. Pepper apologizes, seeks additional preferences, and provides other alternative recipes. **A2. Insufficient Data for Personalization:** 1. If user preferences are not clear, Pepper asks targeted questions to understand the user's demand. 2. Pepper provides a suggestion based on the newly acquired information. **A3. External Noise Interference:** 1. External noises (like TV, music, etc.) interfere with Pepper's ability to comprehend user requests. 2. Pepper requests the user to repeat the instruction or move to a quieter location.
Requirements	Providing data-driven recipe choices and suggestions based on needs, dietary preferences, and restrictions.
Storyboards	 1. Chef asks Pepper for dinner recommendations  2. Pepper asks for more information in order to give a more personalized answer.  3. Chef answers Pepper with relevant information  4. Pepper asks for more information in order to give a relevant answer.

	 5. Chef answers Pepper again with more relevant information  6. Pepper shows multiple options on the screen, of which the user should pick one.  7. Chef presses one of the suggested options listed on the tablet.  8. Pepper displays chosen dish on the screen and narrates to the chef what dish he has chosen.
Priority	1
Related Use Cases	• Use case of creating an account. • Use Case of setting all health conditions and dietary restrictions. • Use case of recipe request. • Use Case of creating an enjoyable environment while cooking
Assumptions	• The Chef has a basic understanding of culinary terms and techniques. • Pepper the Robot has access to a comprehensive recipe database. • The user is willing to engage in a vocal interaction with Pepper. • The user can provide feedback on suggestions for system improvement.
Open Issues	• Integration with external recipe databases for a wider range of cuisines or with an AI chatbot tool that eliminates the need. • Improving voice interaction for users with varying accents or speech patterns. • Robust and diverse database to cater to varied user tastes and preferences. • Ethical considerations and compliance with data protection regulations regarding storing and using user data.

UC-03 Create Account

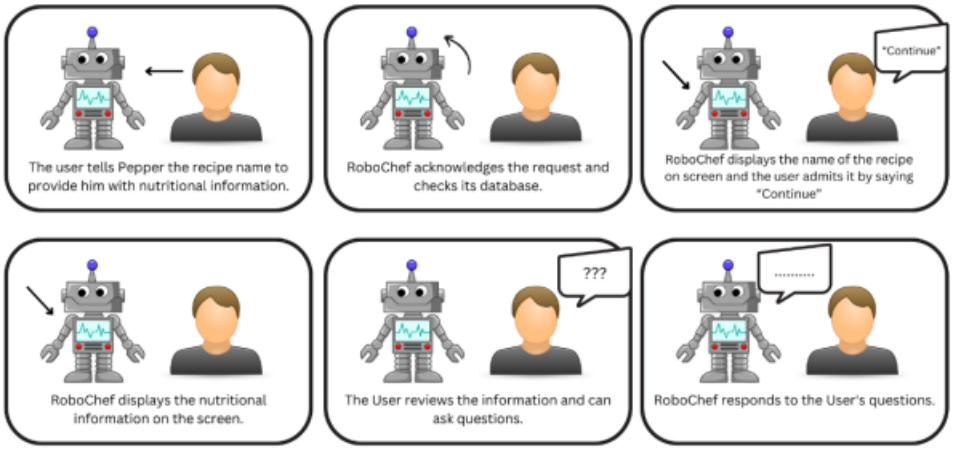
UC-ID and Name	**UC-03 Create Account**		
Created By	Elie El Rayess	**Creation Date**	12/10/2023
Actors	The main actors are Pepper the Robot (Culinary Cooking assistant) & person (Chef)		
Trigger	User wants to set up a new account on the Pepper Robot for personalized recipes.		
Description (Objectives/Goals)	*User wants to create Pepper Robot accounts to personalize their recipes, based on his information registered in his account.*		
Preconditions	PRECOND-1. the Chef is recognized by the robot culinary assistant. PRECOND-2. The Pepper Robot application is running and ready to accept user commands. PRECOND-3. the Chef starts to create his account		
Postconditions	POSTCOND-1. User has a registered account on the Pepper Robot with their preferences and settings.		
Action Sequence (Success Scenario)	1. User approaches the Pepper Robot. 2. Pepper Robot greets the user and asks if they would like to create a new account. 3. User expresses their desire to create an account. 4. Pepper Robot displays a registration form on its screen. 5. User enters their personal details, such as name, email, and password. 6. During the account creation, the user is asked to input their allergy and health information. 7. Pepper Robot validates the entered information. 8. The information is stored in the user's account profile. 9. The Robot displays all information on the screen. 10. The Robot asks the Chef if they agree with provided information. 11. The Chef confirms if they agree with the info. 12. Use case ends.		
Extensions	- If the user enters invalid information during registration (e.g., an invalid email address or password not meeting the criteria), Pepper Robot prompts the user to correct the errors and resubmit the form. - If the Pepper Robot experiences technical issues during the registration process, it informs the user and suggests trying again later.		
Requirements	-The Pepper Robot application should have a user registration system. -A user database should be maintained to store registered user details.		

	-The registration process should include validation of user-provided information. -The Pepper Robot should provide a user-friendly interface for account creation and preference setting. - The robot must be able to identify and recommend recipes that do not contain ingredients to which the user is allergic or restricted.
Storyboards	 Pepper Robot greets the user and asks if they would like to create a new account,and user expresses their desire to create an account.  Pepper Robot displays a registration form on its screen, and User enters their personal details, such as name, email, and password.  Pepper asks the chef to input his allergies and health information.

	
Priority	1
Related Use Cases	• Use case of Recipe Recommendation
Assumptions	• Users are interested in personalizing their recipes with the Pepper Robot Chef Assistant • Pepper Robot has an internet connection to validate user email addresses during registration.
Open Issues	• The security of user data and account information should be considered and addressed. • Data privacy and consent for user information storage should be clarified.

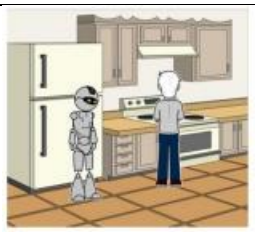
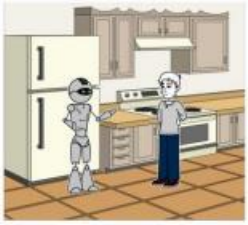
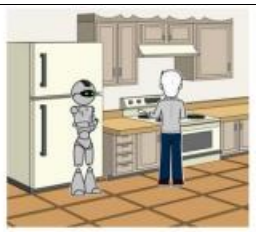
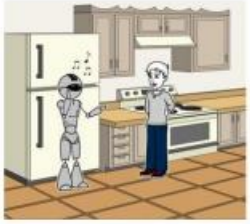
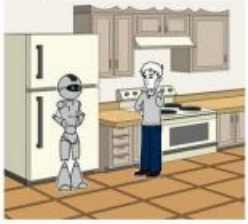
UC-04 Provide Nutritional Information for a Specific Recipe

UC-ID and Name	**UC-04 Provide Nutritional Information for a Specific Recipe**		
Created By	Catherina El Khoury	**Creation Date**	12/10/2023
Actors	RoboChef, the Culinary Assistant (Pepper) & User (Chef)		
Trigger	The User seeks additional information and wants to understand the nutritional content of a specific recipe prepared by RoboChef (Pepper).		
Description (Objectives/Goals)	*The User wants to get nutritional information of a specific recipe, and Pepper provides the nutrition information needed.*		
Preconditions	1. The User must be physically present to specify the recipe for which they seek nutritional information from RoboChef. 2. RoboChef must be functioning correctly, and connected to the necessary databases to provide specific nutritional information.		
Postconditions	1. The User has received detailed nutritional information about the specified recipe from RoboChef. 2. The User may have a clearer understanding of the nutritional content of the dish.		
Action Sequence (Success Scenario)	1. The User asks for nutritional information about a specific recipe. 2. RoboChef acknowledges the request and checks its database. 3. Robochef displays the name of the recipe on screen. 4. The User admits it and says “Continue” to proceed. 5. RoboChef displays the nutritional information. 6. The User reviews the information and can ask questions. 7. RoboChef responds to the User's questions. 8. The interaction ends.		
Extensions	• **Ingredient Substitutions:** ⇨ The User can suggest using different ingredients if some are missing. ⇨ RoboChef explains how these substitutions affect the nutrition of the dish. • **Recipe Suggestions:** ⇨ If the User wants more recipe ideas that suit their dietary preferences, they can ask RoboChef for suggestions. ⇨ RoboChef will show a list of recommended recipes on its screen that fit the User's dietary requirements.		
Requirements	**1. Functional Requirements:** - Recipe Database: Pepper must have a database with recipes, listing their ingredients and nutritional details.		

	- Integration with Health Apps: It should be able to connect with health and fitness apps to track nutritional information. **2. Nonfunctional Requirements:** - Performance: Pepper should speak clearly so users can easily understand the information it provides. - Reliability and Accuracy: Pepper must work consistently, and any calculations or ingredient substitutions should be dependable and precise.
Storyboards	
Priority	2
Related Use Cases	• Use case of creating an account and setting all health conditions • Use case of Recipe Recommendation
Assumptions	• RoboChef has a good understanding of nutritional ingredients and how to calculate nutrition facts. • RoboChef is connected to a recipe database, and these recipes include current nutritional information.
Open Issues	• Ensuring clear and noise-free communication, adjusting Pepper's volume for better understanding by everyone. • Ensuring the accuracy and reliability of real-time nutritional calculations. • Connecting with external recipe databases for a wider range of nutritional information for various cuisines.

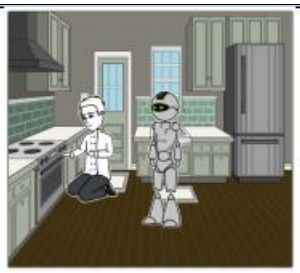
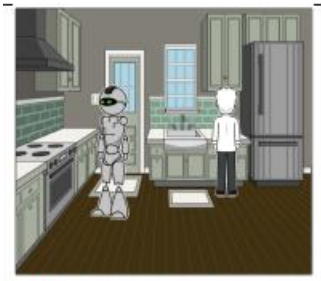
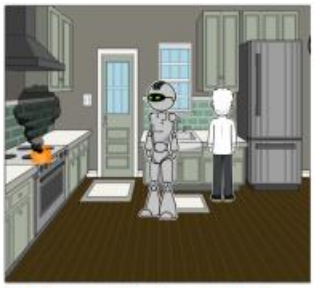
UC-05 Play music or Tell a joke

UC-ID and Name	**UC-05 Play music or Tell a joke**		
Created By	Garo Margossian	**Creation Date**	12/10/2023
Actors	The main actors are Pepper the Robot (Culinary Cooking assistant) and the user who wants to cook (Chef)		
Trigger	The chef indicates a desire to request entertainment in the form of music or a joke, while cooking		
Description (Objectives/Goals)	*The user wants to avoid boredom by engaging in entertainment, such as music or jokes, facilitated by the robot.*		
Preconditions	PRECOND-1. the user has requested a recipe from the robot and is being provided with instructions		
Postconditions	POSTCOND-1. The robot tells a joke or plays a song desired by the user or a random one in its libraries if not specified		
Action Sequence (Success Scenario)	1. The user indicates boredom and requests entertainment. 2. The robot asks the user if they would like to listen to a song or to a joke. 3. The user confirms the choice. 4. If a joke is chosen, the robot tell a joke. 5. If music is chosen, the robot asks the user if they want a specific song. 6. The user confirms specification. 7. The robot proceeds with the previous request until the next step of instructions is given. 8. Use case ends		
Extensions	**3a. No matching song is found:** 3a1. The Robot alerts the user that the requested song is unavailable. 3a2. The user chooses a different song. **4a. The user wants to modify the song settings** 4a1. The use requests to pause the song, the song will stay paused until receiving a play request. 4a2. The user requests to change the song, current song will stop. 4a3. The user chooses a different song.		
Requirements	*Having access to a vast array of high-quality song options.*		

Storyboards	 1. While following the given cooking instructions the user requests entertainment  2. The robot asks whether the user would like to listen to a song, the user confirms  3. The robot asks what it should play, and the user specifies a song  4. The robot starts playing the requested song  5. The user ask the robot to pause the song to choose a different one  4. The robot plays newly requested song
Priority	3
Related Use Cases	• Use case of creating an account and setting all health conditions • Use case of Recipe Recommendation
Assumptions	• The robot has access to a library of songs or the internet.
Open Issues	• Improving voice interaction for users with varying accents or speech patterns.

UC-06 Monitor kitchen appliance

UC-ID and Name	UC-06 Monitor kitchen appliance		
Created By	Fawzi El Khoury	**Creation Date**	14/10/2023
Actors	The main actors are Pepper the Robot (Culinary Cooking assistant) & person (Chef)		
Trigger	Pepper is triggered whenever an appliance in the kitchen detects an alarm or warning and whenever the user starts cooking a recipe.		
Description (Objectives/Goals)	*The user wants to monitor kitchen appliances to ensure safety, seeking to prevent potential dangers or malfunctions.*		
Preconditions	PRECOND-1. Pepper must be connected to all of the electric appliances in the kitchen, including the oven, refrigerator, and stove. PRECOND-2. Pepper must be programmed to monitor the status of the appliances.		
Postconditions	POSTCOND-1. Pepper has responded to any alarms or warnings from appliances.		
Action Sequence (Success Scenario)	1. The chef starts cooking a recipe. 2. Pepper automatically adjusts the temperature of the oven to the desired setting. 3. Pepper displays the ingredients available in the refrigerated for the recipe. 4. Pepper monitors the temperature of the food and the cooking time. 5. Once the food is cooked, Pepper alerts the user and turns off the oven. 6. Pepper prompts the user to remove the food from the oven. 7. Use case ends.		
Extensions	**3a. Event of a fire:** **1.** Alert the user: Pepper detects smoke in the kitchen so it alerts the user. 2. Turn off the oven. 3. Evacuate the kitchen: Pepper leads the user out of the kitchen to safety. 4. Call for help: Pepper calls the fire department or 911 to get help.		
Requirements	*Pepper should be able to interact with kitchen appliances, follow recipes, and ensure the safety of the user*.		

Storyboards	 1. Pepper is displaying the ingredients available in the refrigerated for the recipe  2. The chef is adding the ingredients to the oven. Pepper automatically adjusts the temperature of the oven to the desired setting.  3. Pepper monitors the temperature of the food and the cooking time.  4. Once the food is cooked, Pepper alerts the user that the food is cooked, and the chef removes it from the oven.  5. Pepper detects smoke in the kitchen.  6. Pepper alerts the user that there is a fire in the kitchen and the chef leaves the kitchen.  7. Pepper turns off the oven.  8. Pepper calls the fire department or 911 to get help.
Priority	2

Related Use Cases	• Use case of Request a Recipe • Use case of Recipe Recommendation
Assumptions	• The Chef can provide Pepper with accurate information about the recipe they want to cook. • Pepper can understand and follow recipes.
Open Issues	• Pepper should be affordable for the target market.

4.2 Personas:

Personas in our project are fictional user profiles representing diverse needs and behaviors in the kitchen. They are crucial for understanding and empathizing with our user base, guiding the design of our cooking assistant robot, Pepper. By crafting personas, we tailor features to meet specific user expectations, ensuring Pepper becomes a personalized and intuitive assistant, enhancing overall user satisfaction in the kitchen.

Each persona includes essential *demographic* details like the user's *name, age, cooking experience, occupation, and health.* Then we have the *motivations* section that includes *hobbies, cultural background, and community ties* to get to know user preferences more. *Personal challenges, such as social dynamics, environmental factors, and family structure*, are considered to ensure that Pepper aligns with users' lifestyles.

Persona 1: Samira, Stay at home Mom.

	"I want to feed my family"	
Demographics	Name	Samira
	Age	52
	Years with cooking experience	25
	Occupation	Stay at home Mom
	Health	Wears hearing aid
Motivation	Hobbies	Watching soap operas and cooking shows. Enjoys weekend family outings (local park, museum, or cinema). Loves reading, particularly cookbooks.
	Culture	Loves travelling and learning about foreign food cultures. Has not been doing so due to lack of time.
	Community	Part of the local Church community. Has taken cooking classes in the local area. Met with Chef Antoine.
	Personal challenges	Time constraints (between family and housekeeping). Finding a balance between meals her kids love and meals that are nutritious. Keeping up with everchanging food trends.

Social Environment	Background	Living in a suburban area with her husband and two children, ages 12 and 15. Had a tough time conceiving her children. Grown up with traditional cooking methods but is always keen to find new and efficient ways to feed her family. Juggles between local Church activities, her children's school and extracurricular activities, and her desire to accommodate for her children's and husband's health needs.
	Husband	Samira's husband Ibrahim has high blood pressure and chronic cardiovascular problems. These health issues have made him particularly sensitive to sodium and saturated fats in his diet. Samira often finds herself exploring heart-healthy recipes, opting for fresh herbs over salt, and choosing lean meats or plant-based proteins.
	Children	Samira's pride and joy. However, 12-year-old Issam comes with his unique health challenges that make cooking a tough job for Samira. Issam was diagnosed with Type 1 diabetes at a young age, requiring careful monitoring of his carbohydrate intake and ensuring a balanced diet to manage his blood sugar levels. Her other son Nehme is healthy but likes to eat a lot (2 regular servings).

Persona 2: Sally, a Professional Chef Dreamer

	"I want to be a Professional Chef"	
Demographics	Name	Sally
	Age	27
	Years with cooking experience	5
	Class	Sally graduated from a renowned culinary institute, earning a degree in Culinary Arts with honors. She is committed to continuous learning, attending advanced culinary classes and workshops to refine her skills.
Motivation	Hobbies	In her free time, Sally enjoys exploring food markets, experimenting with exotic ingredients, and writing her own recipes. She's also an avid reader of cookbooks and food blogs.
	Sports	She is an enthusiastic yoga practitioner. She also enjoys leisurely hikes.
	School	Sally's culinary journey began at Le Cordon Bleu, where she honed her skills under the guidance of some of the finest chefs in the industry.
	Personal challenges	As a chef in the culinary world, Sally is no stranger to the challenges of the industry. Balancing long, boring hours in the kitchen with personal life can be demanding. She also faces the challenge of pushing the boundaries of her creativity while respecting the classics of cuisine.
Social Environment	Background	Sally's passion for cooking was ignited by her Italian grandmother's Sunday feasts and her mother's delicious homemade pasta. Growing up in a family that celebrated food as the cornerstone of togetherness, she was naturally drawn to the culinary world.

	Parents	Sally's parents run a family-owned restaurant, and their deep love for Italian cuisine greatly influenced her culinary journey. They continue to be her biggest supporters, celebrating her achievements in the culinary world with pride.

Persona 3: Anastasia, an Untrained Chef

	"I want to learn how to cook"	
Demographics	Name	Anastasia
	Age	25
	Years with cooking experience	0.5
	Occupation	Part time teacher
	Health	Generally healthy but has been struggling with her diet since getting engaged.
Motivation	Hobbies	Reading, yoga, and hiking
	Sports	Swimming and running
	Personal challenges	Managing stress, balancing work, and her personal life, learning to cook healthy meals for two
	Culture	Keep the family recipes alive

Social Environment	Background	Anastasia has always been interested in cooking, but she never had the time or energy to pursue it seriously. Now that she's engaged, she wants to learn how to cook for her future husband.
	Parents	Anastasia grew up in a loving home with two supportive parents that were passionate about healthy eating and always cooked for her.

5. Technical Part:

In our project we made use of different blocks to reach the required functionality. The functionality we aimed to achieve was that of the three use cases **UC01- Request a Recipe, UC02 – Recommend a Recipe, and UC-05 Play music or Tell a joke.** In terms of events, the chef should first approach pepper who will guide him through the interaction. The chef then takes suggestions from pepper who in its turn reads out the recipe step carefully and suggests entertainment in times of waiting (baking time, refrigerating time, etc…).

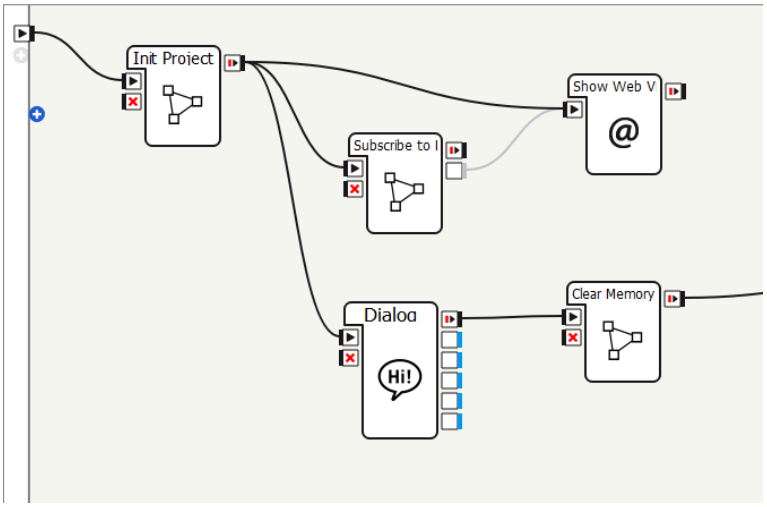


Figure 9: Project Root: Blocks Used

5.1.Folder Structure

As shown in the figure below, our interaction relies on behavior.xar in the behavior_1 folder. It is accompanied with a web page created in the html folder and its respective css for design, js for functionality such as raiseEvent, and pics where we stored all of the images we used (the recipe images included) and a webfonts folder.

Moreover, the play song usecase is implemented in its respective behavior under the play_song directory and uses the music files indicated under the music directory.

Furthermore, our dialogue files are under RS which stands for Recipe Suggestion and include a concepts.top file (responsible for our word dictionary – including cuisine type, mealtype, and other significant terms used in this project). We have also made use of the lexicon_enu.top concept file which is an open-source dictionary provided for daily words used. Our project dialogue is in RS_enu.top. We will discuss this in more detail in a separate section.

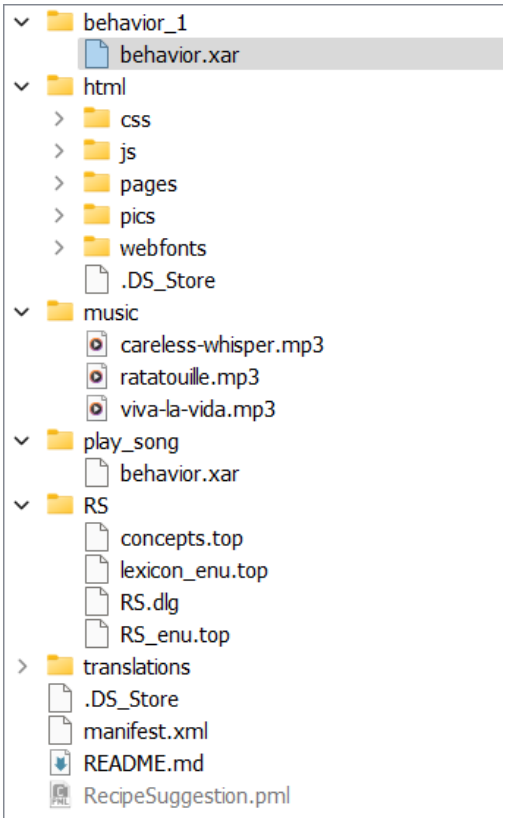


Figure 10: Folder Structure

5.2.Init Project Block

In our ***Init Project*** block, we use the Create Memory Block 7 times – once for every field we want to create in the AL Memory. The respective fields are:

- **cuisine**: This field represents the type of cuisine we are asking pepper for. E.g.: Italian, Chinese, French…
- **mealtype**: This field represents the type of meal. E.g.: Breakfast, Lunch, Dinner or Dessert
- **option1**: This field represents one of the options suggested by pepper after the chef picks the respective cuisine and mealtype mentioned above.
- **option2**: This field represents one of the options suggested by pepper after the chef picks the respective cuisine and mealtype mentioned above.
- **nxtProp**: This field is used to indicate to pepper to move to the next step in the recipe.
- **prvsProp**: This field is used to indicate to pepper to move back to the previous step in the recipe.

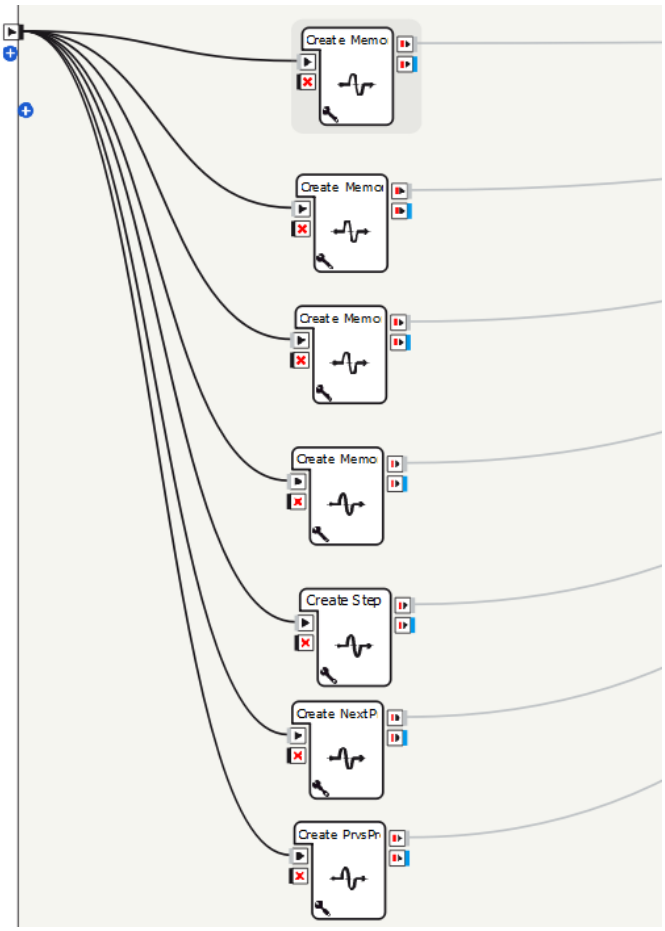


Figure 11: Init Project Block

5.3.Clear Memory Block

In our ***Clear Memory*** block, we use the Clear Memory Block 7 times – once for every field we previously created in the AL Memory.

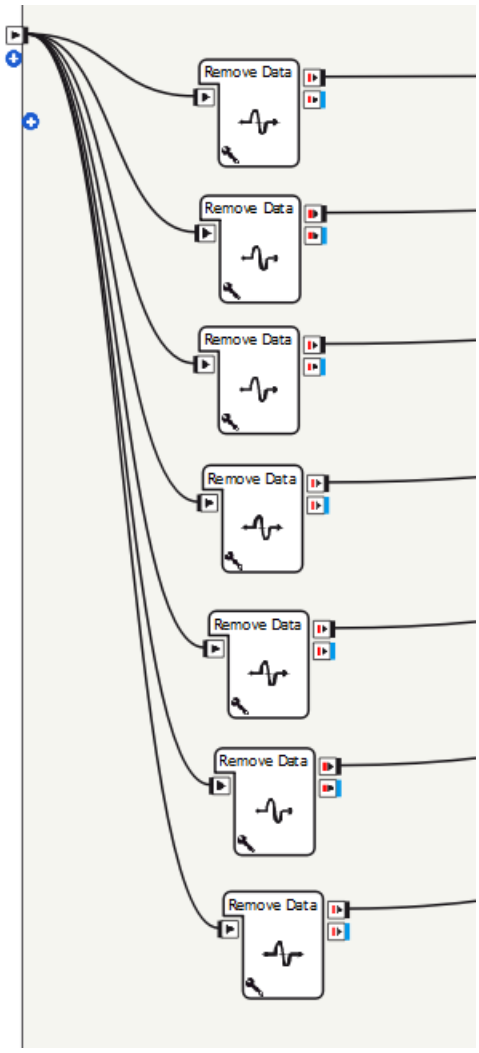


Figure 12: Clear Memory Data

5.4.Subscribe to Event Block

In our ***Subscribe to Event*** block, we use the Subscribe to Event block whose on event changes the parameters set for the web view and changes the display accordingly.

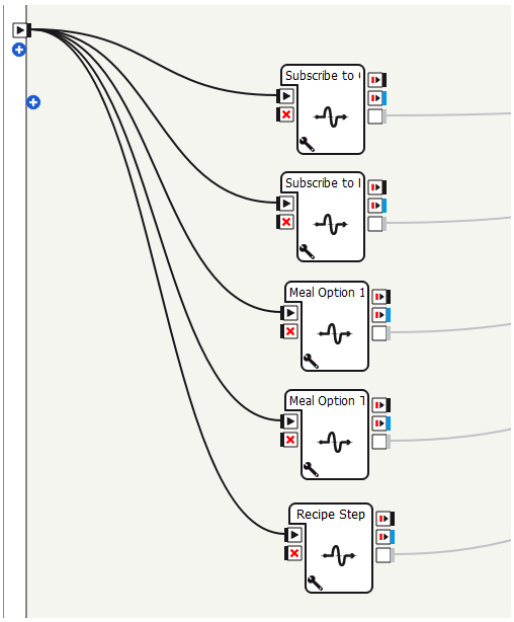


Figure 13: Subscribe to events block structure

We listen here to the specific events that are required to change the webview display as shown in the python script used in the show WebView Block and whose snippet is shown below:

```
# create proxy on ALMemory
memProxy = ALProxy("ALMemory","localhost",9559)

#get data. Val can be int, float, list, string
cuisine = memProxy.getData("cuisine")
mealtype = memProxy.getData("mealtype")
option1 = memProxy.getData("option1")
option2 = memProxy.getData("option2")
nextinstruction = memProxy.getData("nextinstruction")
```

Figure 14: ALMemory Data Fetching in Python WebView Script

```
        if tabletService:
            # use default robot IP address from the tablet
            robotIP = "198.18.0.1" # tabletService.getRobotIp()
            if nextinstruction:
                url = "http://{ip}/apps/{appName}/pages/step.html?nextinstruction={nextinstruction}".format(ip=robotIP, appName=appName, nextinstruction=nextinstruction)
            else:
                if option1 and option2:
                    url = "http://{ip}/apps/{appName}/pages/choice.html?cuisine={cuisine}&mealtype={mealtype}&option1={option1}&option2={option2}".format(ip=robotIP,
appName=appName, cuisine=cuisine, mealtype=mealtype, option1=option1, option2=option2)
                else:
                    if cuisine or mealtype:
                        url = "http://{ip}/apps/{appName}/pages/displayinfo.html?cuisine={cuisine}&mealtype={mealtype}".format(ip=robotIP, appName=appName, cuisine=cuisine,
mealtype=mealtype)
                    else:
                        url = "http://{ip}/apps/{appName}/pages/index.html".format(ip=robotIP, appName=appName)

            self.logger.info(url)

            # Show the web view on the tablet
            tabletService.showWebview(url)
        else:
            self.logger.warning("Couldn't find tablet service, so can't set application: %s" % appName)
```


Figure 15: Code Snippet Responsible for WebPage Navigation based on AL Memory Changes

Based on the code above, the webpage is redirected to the Index Screen where the user is welcomed when the AL Memory fields are all empty.

Moreover, the webpage is redirected to the intermediate page where the user's choice of cuisine or mealtype is displayed when the AL Memory cuisine and mealtype are filled with user input from QiChat Dialogue:

```
proposal: %cuisineprop Which cuisine are you in the mood for? I have Italian \pau=80\, French \pau=80\, Chinese \pau=80\ and Mexican!
    u1:(Italian) $cuisine=italian Great! \pau=200\ Now, do you have a specific type of dish or meal time in mind from the Italian cuisine?
        u2:(starter) \RSPD=40\ $mealtype=starter Perfect! For a cozy Italian starter, I'd recommend a Caprese Salad. For an adventurous one, how about Arancini Balls?
$option1=caprese-salad $option2=arancini-balls
        u2:(dessert) \RSPD=40\  $mealtype=dessert A quick Italian dessert would be Panna Cotta. Feeling cozy? Try Tiramisu. $option1=panna-cotta $option2=tiramisu
        u2:(breakfast) \RSPD=40\ $mealtype=breakfast A quick Italian breakfast would be a Cappuccino with a Cornetto. I also recommend a Frittata. $option1=cappuccino
$option2=frittata
        u2:(lunch) \RSPD=40\ $mealtype=lunch For a cozy Italian lunch, try a Margherita Pizza or maybe a Spaghetti Carbonara? $option1=margherita-pizza $option2=spaghetti-
carbonara
        u2:(dinner) \RSPD=40\ $mealtype=dinner Feeling adventurous for dinner? How about Lasagna al Forno? Feeling cozy? Try Osso Buco. $option1=lasagna-al-forno $option2=osso-
buco
```

Figure 16: Cuisine and Mealtype QiChat Snippet

Furthermore, the webpage is redirected to the choice page to pick between two recommended recipe choices when the option1 and option2 ALMemory fields are filled in the QiChat Dialogue as shown in the Figure 16 above.

The user is then prompted to pick one of the choices on the screen with the use of two images to choose from. Technically, the images are shown using the option1 and option2 memory data as names in the **/pics** directory with a .png extension. We then proceed to use a raiseEvent function onClick to transmit the user's choice to the QiChat dialogue in order to deliver a consistent user experience.

```
function raiseEvent(name, value) {
  QiSession(function (session) {
    session.service("ALMemory").then(
      function (mem) {
        mem.raiseEvent(name, value);
      },
      function (error) {
        console.log("An error occurred:", error);
      }
    );
  });
}
```

Figure 17: JS Function to raise an event

Finally, the webpage is redirected to the STEP Screen where a recipe step is displayed when a nextinstruction memory is triggered which in turn is displayed on the tablet itself. This nextinstruction is passed from the QiChat Dialogue as shown in the code snippet below:

```
proposal: %step5 \RSPD=90\ $nextinstruction="Cover with foil, bake for 25 to 40 minutes until golden." Cover with foil, bake for 25 to 40 minutes until golden.
    u1:(next) ^gotoReactivate(step6)
    u1:("go back") ^gotoReactivate(step4)
    u1:(e:prvsProp) ^gotoReactivate(step4)
    u1:(e:nxtProp) ^gotoReactivate(step6)
```

Figure 18: QiChat Dialogue Snippet

```
function displayPageInformation() {
  document.getElementById("cuisine").innerHTML =
    "Cuisine: " + decodeURI(getUrlParam("cuisine", ""));
  document.getElementById("mealtype").innerHTML = decodeURI(
    "Meal Type: " + getUrlParam("mealtype", "")
  );

  imageName1 = getUrlParam("option1", "");
  if (imageName1) {
    document.getElementById("optionOneLabel").innerHTML =
      transformText(imageName1);
    document.getElementById("option1").src = "../pics/" + imageName1 + ".png";
    document.getElementById("option1").onclick = function () {
      raiseEvent("chosenOption", imageName1);
    };
  }
  imageName2 = getUrlParam("option2", "");
  if (imageName2) {
    document.getElementById("optionTwoLabel").innerHTML =
      transformText(imageName2);
    document.getElementById("option2").src = "../pics/" + imageName2 + ".png";
    document.getElementById("option2").onclick = function () {
      raiseEvent("chosenOption", imageName2);
    };
  }
}
```

Figure 19: Display Info JS Function Used to Display All Information

5.5.Dialogue

We have made use of our dialogue to deliver information to the webpage as shown above and to provide a consistent and clear flow of information to the user in more than one perspective (Tablet, Speech…). As we have mentioned earlier, we integrated the use of proposals accompanied with user input to deliver robot output based on user input.

This is shown clearly in our entertainment use case where the user is prompted to pick a type of entertainment (joke or music). This either invokes a play music behavior using the ^run(play_song) or a joke using the ^rand ~JokeList (JokeList is a concept in the concepts.top file).

```
proposal: %entertainmentM Would you like me to play some music or tell a joke?
    u1:(music) Which music do you want to hear?
        u2: (_~SongList) I will play the requested song $songPlay=$1 $songPlay ^run(play_song) ^gotoReactivate(end)
    u1:(joke)  ^rand ~JokeList ^start(animations/Stand/Emotions/Positive/Laugh_3) ^wait(animations/Stand/Emotions/Positive/Laugh_3) ^gotoReactivate(end)
```

Figure 20: Entertainment Proposal

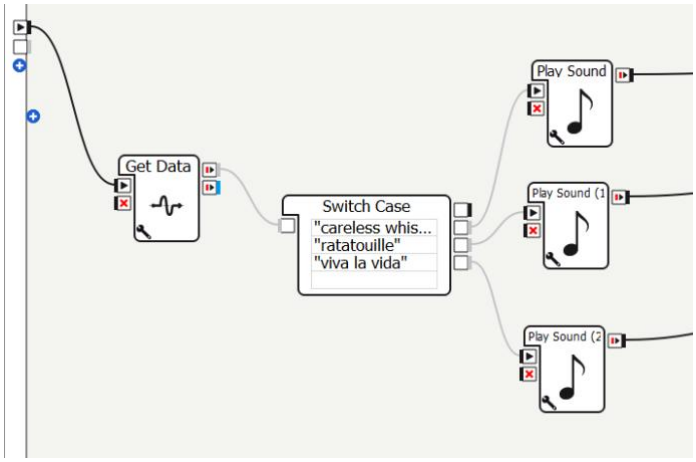


Figure 21: Play Song Behavior used in ^run

We have also made use of dialogue animations such as ^start(animations/Stand/Gestures/Hey_3) and ^start(animations/Stand/Emotions/Positive/Winner_1) as well as speech mode ^mode(contextual) ,tone ~joyful, and pauses defined in the concepts.top.

As we see here, we have made use of the ^goToReactivate(proposalName) in order to ensure that all steps can be revisited for the user to be able to go forwards and backwards again in the same proposal (read same step twice) -something that is very important for cooking instructions.

To make the robot's speech comprehensive during cooking, we have set the reading speed to 90 to accustom for a seamless user experience in his/her cooking journey.

```
proposal: %step3 \RSPD=90\ $nextinstruction="Combine ricotta, egg, and Parmesan for the cheese filling." Combine ricotta, egg, and Parmesan for the cheese filling.
    u1:(next) ^gotoReactivate(step4)
    u1:("go back") ^gotoReactivate(step2)
    u1:(e:prvsProp) ^gotoReactivate(step2)
    u1:(e:nxtProp) ^gotoReactivate(step4)
```

Figure 22: Step Proposal

6. Discussion and Conclusion:

In the beginning, selecting a project topic proved to be a bit challenging. We needed Pepper to be a valuable and necessary asset that makes people's lives easier. After brainstorming, we settled on the concept of creating RoboChef, an experienced cooking assistant. RoboChef does not only provide ingredients and instructions but also make cooking enjoyable by playing music and telling jokes while cooking. Additionally, RoboChef gives users' access to a variety of unique and needed recipe suggestions from kitchens around the world.

Our vision was to establish a full database storing recipes, instructions, and culinary types from diverse cultures, making RoboChef a repository of knowledge that every chef dreams of having in their kitchen.

Our project demonstrated Pepper's success in the kitchen, as mentioned in our slogan: *"Bring Pepper back to the Kitchen"*.

We named our team "Pepper Mills" to maintain a kitchen-themed atmosphere.

Once our idea is set, we created a plan to bring it to life. Starting with finding a research paper that aligns perfectly with our project. After a dedicated research, we identified one that provided the necessary insights "*Cooking Assistant Service Utilizing an Interactive Robot*". A key aspect that caught our attention in the paper was how to structure data in the database to enhance peppers' performance, allowing it to

deliver short instructions orally instead of presenting a large paragraph for understanding. This was illustrated in Figures 5 and 6 (in the report), showcasing the transition from unstructured to structured recipe data for Pepper.

Subsequently, we created various scenarios and characters for our project, including use cases and personas. The instructor provided us with a template for the use cases, and each team member created their own. While everything was clear, the storyboard section was a bit confusing. With the instructor's guidance, we filled this part by creating different storyboards using Canva.com and storyboardthat.com.

The personas comprised three different characters (users) representing 3 different ages from 3 life categories. Samira, a mother of a family, deals with diverse cases in her household, aiming to meet the needs of family members with various health problems. Sally, a professional chef from a family of chefs, seeks to explore recipes from different cultures worldwide. Lastly, Anastasia is new to cooking and relies on Pepper to guide her through the essential tips for cooking.

The most significant challenge started when attempting to implement the planned scenarios with RoboChef. We began with Choregraphe, an entirely new software allowing us to interact with Pepper. Understanding how to manipulate the boxes, create dialogs, and display images using web view took time and presented initial difficulties. After several trials, we successfully created a human-robot interaction using this software.

Going into the code proved more challenging than imagined. Addressing errors, identifying missing elements and getting used to a new language, QiChat was strongly time-consuming. But fortunately, our instructor guided us through chapters and provided us with the documentation needed.

Concerning HTML, CSS, and JavaScript, the implementation was not particularly challenging for us, due to our prior knowledge gained from the web programming course. However, despite this familiarity, some team members came from different engineering backgrounds by which applying these skills to Pepper and establishing interactions with Pepper's tablet demanded additional knowledge. Specifically, understanding event handling, a concept new to some of us, became critical for this task.

Testing presented additional challenges, especially in environments with noise, as Pepper struggled to understand spoken commands. This led us to use the dialog box provided by Choregraphe to input commands clearly.

We also faced an issue where, with each trial, the memory kept information from the previous attempt. Consequently, we had to find a way to clear the memory at the beginning of each trial, that was resolved successfully.

In the end, we successfully integrated our use cases with Pepper, and the success was met with cheers of "Hooray!" Reflecting on the project, it was not just about meeting goals; it showcased our team's ability to navigate challenges, learn new things and adapt successfully, transforming our idea into a tangible and enjoyable cooking assistant.

To sum up, the experience with the course was truly exceptional. Navigating through new technologies and engaging with robots proved to be challenging, yet it constituted the highlight of the entire journey. The course exceeded our expectations, introducing not only the expected content but also exploring the implementation of cutting-edge technologies. This milestone expanded our knowledge into an entirely new world—the fascinating world of human-robot interaction!

7. Prototype:

In order to test the validity of our work, we have set up a convenient cooking environment in the kitchen, dressed up our culinary assistant, pepper , with the cooking custom and filmed a considerably small video showcasing our whole, enjoyable journey in the HRI course !

Here is the link to our video:
Pepper Mills - Human Robotics Interaction - Final Demo - YouTube