

CS 466/566 – Voice Assistants

Final Project

Designing, Building and Testing a Voice Application

- A Dialogflow agent in [Dialogflow](#)

NOTE : I have used more than 3 intents because I thought taking each of these components as individual smaller components instead of one, less clear bigger component, would be better. Clubbing smaller components into one intent would have made the fulfillment code more complex, hence I have taken them as 6 nuclear intents.

Step 1 - Describe the Application

Purpose: The purpose of this application is to provide users with real-time weather updates, forecasts, and weather alerts for specific locations. Users can interact with the voice assistant to get current weather conditions, future weather forecasts, and any severe weather alerts.

Features:

1. Check Current Weather:

- Users can ask for the current weather in any specified location.
- The assistant will provide information such as temperature, humidity, wind speed, and weather conditions (e.g., sunny, cloudy, rainy).

2. Weather Forecast:

- Users can ask for a weather forecast for the next few days in a specified location.
- The assistant will provide forecasted weather conditions, including temperature highs and lows, and any expected precipitation.

3. Weather Alerts:

- Users can ask if there are any weather alerts for a specified location.

- The assistant will provide information on any active weather warnings or alerts, such as severe storms, hurricanes, or flood warnings.

4. Good Morning Weather Update:

- Users can ask for a morning weather update for a specified location.
- The assistant will provide weather conditions for the morning, helping users to plan their day.

5. Good Night Weather Update:

- Users can ask for a night weather forecast for a specified location.
- The assistant will provide weather conditions for the night, helping users to prepare for the next day.

6. Weather Tomorrow:

- Users can ask for the weather forecast for tomorrow in a specified location.
- The assistant will provide forecasted weather conditions for the next day, including temperature, expected precipitation, and overall weather summary. This helps users to plan their activities for the upcoming day effectively.

API: The application will use the OpenWeatherMap API to fetch real-time weather data, forecasts, and alerts.

Step 2 - Interaction Model

The interaction model for this application includes five intents, each designed to handle specific user requests related to weather information.

1.) CheckWeatherIntent:

Purpose: To provide the current weather conditions for a specified location.

Slots/Parameters:

- **geo-city:** The city for which the user wants to know the weather.
- **date-time:** The date and time for which the weather is requested (if applicable).
- **AlertType (Custom Slot):** Specific type of weather alert if the user is inquiring about severe weather conditions.

Sample Utterances/Training Phrases:

1. How's the climate in San Diego
2. What's the weather status in Phoenix
3. Is it windy in Philadelphia
4. What's the humidity in Dallas
5. Tell me the current weather in Atlanta
6. What's the weather like in Boston
7. Is it raining in Denver
8. How warm is it in San Francisco
9. Do I need an umbrella in Seattle
10. What's the temperature in Miami
11. Is it sunny in Houston
12. How's the weather tomorrow in Portland
13. How's the weather today in Portland
14. Tell me the weather in Mumbai
15. What's the weather in Portland?

2.) WeatherForecastIntent:

Purpose: To provide a weather forecast for the next few days in a specified location.

Slots/Parameters:

- **geo-city:** The city for which the user wants the weather forecast.
- **date-period:** The range of dates for the weather forecast.

- **number:** The number of days for the forecast.

Sample Utterances/Training Phrases:

1. How's the weather looking in **Seattle** over **the next few days**?
2. Give me the weather forecast for **Orlando** for **the next 3 days**.
3. How will the weather be in **Nashville** over **the next 6 days**?
4. Can you provide the weather forecast for **Washington, D.C.** for **the next 4 days**?
5. Tell me the weather forecast for **Phoenix** for **the next few days**.
6. What's the weather forecast for **Houston** **this week**?
7. Can you provide the weather forecast for **Dallas** for **the next 4 days**?
8. What will the weather be like in **Miami** in **the coming days**?
9. Give me the weather forecast for **San Francisco** for **the next 7 days**.
10. How's the weather looking in **Los Angeles** over **the next 5 days**?
11. Tell me the weather forecast for **Chicago** for **the next week**.
12. What's the weather forecast for **New York** for **the next 3 days**?
13. Give me the weather forecast for **Portland** for **the next 5 days**
14. How's the weather looking in **Portland** **this week**?
15. What's the weather forecast for **the next 5 days** in **Portland**?

3.) WeatherAlertsIntent:

Purpose: To provide information on any active weather alerts for a specified location.

Slots/Parameters:

- **geo-city:** The city for which the user wants to know about weather alerts.
- **AlertType (Custom Slot):** Specific type of weather alert (e.g., storm, flood).
- **date-time:** The date and time for which the weather alert is relevant.

Sample Utterances/Training Phrases:

1. Update on extreme weather alerts for Dallas, please.
2. Do we have any weather threats in Los Angeles?
3. Report any hurricane warnings in New Orleans.
4. Update me on any storm warnings in Chicago.
5. Let me know if there are any storm warnings in Chicago.
6. "Is there a hurricane warning in Portland?"
7. What's the weather alert status in Portland?
8. Give me any weather alerts for Seattle
9. "Is there any severe weather in Portland?"
10. Are there any weather emergencies in Syracuse?
11. Tell me if there are any weather emergencies in Seattle.
12. Are there any storm warnings in Portland?
13. Tell me about any weather warnings in Portland
14. Are there any weather alerts for Portland?
15. "Is there a hurricane warning in Seattle?"

4.) GoodMorningIntent:

Purpose: To provide the morning weather update for a specified location.

Slots/Parameters:

- **geo-city:** The city for which the user wants the morning weather update.
- **date-time:** The specific morning for which the weather update is requested.
- **time-period:** The time period for the weather update (morning).

Sample Utterances/Training Phrases:

1. Can I get the weather report for Denver this morning?
2. Tell me the morning forecast for Seattle.

3. How's the weather looking this morning in Miami?
4. What's the weather in Los Angeles for this morning?
5. Do I need an umbrella this morning in Portland?
6. Morning weather update for Portland, please.
7. Is it sunny this morning in Portland?
8. What's the temperature this morning in Portland?
9. Morning weather update of Seattle, please
10. Can you tell me the morning weather in Portland?
11. What's the weather in Mumbai this morning?
12. How's the weather this morning in Los Angeles?
13. Tell me the weather in California this morning?
14. What's the weather in Portland this morning?
15. Good morning

5.) GoodNightIntent:

Purpose: To provide the night weather forecast for a specified location.

Slots/Parameters:

- **geo-city:** The city for which the user wants the night weather forecast.
- **date-time:** The specific night for which the weather forecast is requested.

Sample Utterances/Training Phrases:

1. What's the night forecast in Vancouver?
2. Give me the night weather update for Toronto.
3. How's the weather tonight in Rome?
4. What's the temperature tonight in Boston?
5. Is it going to rain tonight in Miami?

6. How's the weather in **Mumbai** tonight?
7. Tell me the night forecast for **Berlin**.
8. What's the forecast for tonight in **Los Angeles**?
9. How's the weather tonight in **Sydney**?
10. Can you give me the night weather update for **London**?
11. Night weather update for **Tokyo**
12. Good night, how's the weather in Paris
13. Tell me the night weather in **New York**
14. What's the weather tonight in **Portland**?
15. Good night

6.) WeatherTomorrowIntent:

Purpose: To provide the weather forecast for the specified location for the next day.

Slots:

- **geo-city:** The city for which the user wants the weather forecast.
- **Entity:** @sys.geo-city

Sample Utterances/Training Phrases:

1. What's the weather prediction for **tomorrow** in **Orlando**?
2. Give me the weather update for **tomorrow** in **Las Vegas**.
3. Will it rain **tomorrow** in **Austin**?
4. How warm will it be **tomorrow** in **Philadelphia**?
5. Can you give me the weather for **tomorrow** in **Seattle**?
6. Provide the weather forecast for **tomorrow** in **Boston**.
7. Give me the weather forecast for **tomorrow** in **San Francisco**.
8. Can you tell me the weather for **tomorrow** in **Chicago**?
9. What's the weather like in **Houston** **tomorrow**?

10. How will the weather be tomorrow in Atlanta?
11. Tell me the weather tomorrow in Berlin
12. How's the weather looking tomorrow in Paris?
13. "Tell me the weather for tomorrow in New York
14. What's the weather tomorrow in Portland?
15. Can you tell me the weather for tomorrow in Berlin?

Step 3 - Fulfillment

1.) CheckWeatherIntent

The first intent I created is CheckWeatherIntent.

- This intent will take the user's input and provide the current weather information for a specified location.
- The Training Phrases of the Intent will consist of one parameter which is an input entity.
- The inbuilt entity is the Location which is of entity type `@sys.geo-city`.
- The logic for training phrases is written in such a way that if the parameter is missed while the user is giving the input, then a prompt will be given by the agent as a response to provide the missing parameter value.
- If the parameter is provided, then the agent will move to the fulfillment part to give the response to the user, which here is enabled as a webhook via Cloud Functions.
- In the fulfillment part, we will enable the inline editor Node.js fulfillment code where we will give the custom responses.
- The logic to select the custom responses is written in such a way to provide a detailed weather report based on the information retrieved from the OpenWeatherMap API.
- It is also made sure to extract the values of the parameters from the JSON file in the interaction model using the `agent.parameters` method so that the response will include the reply based on the user's specifications.

2.) GoodMorningIntent

The second intent I created is GoodMorningIntent.

- This intent will take the user's input and provide the morning weather update for a specified location.
- The Training Phrases of the Intent will consist of one parameter which is an input entity.
- The inbuilt entity is the Location which is of entity type `@sys.geo-city`.

- The logic for training phrases is written in such a way that if the parameter is missed while the user is giving the input, then a prompt will be given by the agent as a response to provide the missing parameter value.
- If the parameter is provided, then the agent will move to the fulfillment part to give the response to the user, which here is enabled as a webhook via Cloud Functions.
- In the fulfillment part, we will enable the inline editor Node.js fulfillment code where we will give the custom responses.
- The logic to select the custom responses is written in such a way to provide a morning weather report based on the information retrieved from the OpenWeatherMap API.
- It is also made sure to extract the values of the parameters from the JSON file in the interaction model using the `agent.parameters` method so that the response will include the reply based on the user's specifications.

3.) GoodNightIntent

The third intent I created is GoodNightIntent.

- This intent will take the user's input and provide the night weather update for a specified location.
- The Training Phrases of the Intent will consist of one parameter which is an input entity.
- The inbuilt entity is the Location which is of entity type `@sys.geo-city`.
- The logic for training phrases is written in such a way that if the parameter is missed while the user is giving the input, then a prompt will be given by the agent as a response to provide the missing parameter value.
- If the parameter is provided, then the agent will move to the fulfillment part to give the response to the user, which here is enabled as a webhook via Cloud Functions.
- In the fulfillment part, we will enable the inline editor Node.js fulfillment code where we will give the custom responses.
- The logic to select the custom responses is written in such a way to provide a night weather report based on the information retrieved from the OpenWeatherMap API.
- It is also made sure to extract the values of the parameters from the JSON file in the interaction model using the `agent.parameters` method so that the response will include the reply based on the user's specifications.

4.) WeatherAlertsIntent

The fourth intent I created is WeatherAlertsIntent.

- This intent will take the user's input and provide any weather alerts for a specified location.
- The Training Phrases of the Intent will consist of one parameter which is an input entity.
- The inbuilt entity is the Location which is of entity type `@sys.geo-city`.
- The logic for training phrases is written in such a way that if the parameter is missed while the user is giving the input, then a prompt will be given by the agent as a response to provide the missing parameter value.

- If the parameter is provided, then the agent will move to the fulfillment part to give the response to the user, which here is enabled as a webhook via Cloud Functions.
- In the fulfillment part, we will enable the inline editor Node.js fulfillment code where we will give the custom responses.
- The logic to select the custom responses is written in such a way to provide information on any active weather warnings or alerts based on the information retrieved from the OpenWeatherMap API.
- It is also made sure to extract the values of the parameters from the JSON file in the interaction model using the **agent.parameters** method so that the response will include the reply based on the user's specifications.

5.) WeatherForecastIntent

The fifth intent I created is WeatherForecastIntent.

- This intent will take the user's input and provide the weather forecast for the next few days for a specified location.
- The Training Phrases of the Intent will consist of one parameter which is an input entity.
- The inbuilt entity is the Location which is of entity type **@sys.geo-city**.
- The logic for training phrases is written in such a way that if the parameter is missed while the user is giving the input, then a prompt will be given by the agent as a response to provide the missing parameter value.
- If the parameter is provided, then the agent will move to the fulfillment part to give the response to the user, which here is enabled as a webhook via Cloud Functions.
- In the fulfillment part, we will enable the inline editor Node.js fulfillment code where we will give the custom responses.
- The logic to select the custom responses is written in such a way to provide detailed weather forecast information based on the information retrieved from the OpenWeatherMap API.
- It is also made sure to extract the values of the parameters from the JSON file in the interaction model using the **agent.parameters** method so that the response will include the reply based on the user's specifications.

6.) WeatherTomorrowIntent

The sixth intent I created is WeatherTomorrowIntent.

- This intent will take the user's input and provide the weather forecast for the specified location for the next day.
- The Training Phrases of the Intent will consist of one parameter which is an input entity.
- The inbuilt entity is the Location which is of entity type **@sys.geo-city**.

- The logic for training phrases is written in such a way that if the parameter is missed while the user is giving the input, then a prompt will be given by the agent as a response to provide the missing parameter value.
- If the parameter is provided, then the agent will move to the fulfillment part to give the response to the user, which here is enabled as a webhook via Cloud Functions.
- In the fulfillment part, we will enable the inline editor Node.js fulfillment code where we will give the custom responses.
- The logic to select the custom responses is written in such a way to provide a detailed weather forecast for the next day based on the information retrieved from the OpenWeatherMap API.
- It is also made sure to extract the values of the parameters from the JSON file in the interaction model using the **agent.parameters** method so that the response will include the reply based on the user's specifications.

Step 4 – Persona

Name: Anya

Gender: Female

Background:

Anya is envisioned as a dynamic and reliable weather assistant who combines meteorological expertise with a passion for technology. She is always ready to provide the most accurate and timely weather updates to help users plan their day efficiently. Anya's background in environmental science and her enthusiasm for innovative technology make her an ideal guide for weather-related inquiries. Whether it's staying prepared for sudden weather changes or planning activities for the coming days, Anya ensures users are always well-informed.

Accent and Language:

- **Accent:** Neutral American English
- Anya's neutral American accent makes her easily comprehensible to a broad range of English-speaking users around the world.

Personality Traits:

- **Informative:** Anya delivers precise and comprehensive weather information, earning users' trust in her accuracy.
- **Friendly:** Anya interacts with users warmly, making every interaction pleasant and engaging.
- **Responsive:** Anya quickly and efficiently addresses user questions, providing the needed information without delay.

- **Reassuring:** Anya maintains a calm demeanor, particularly when communicating weather alerts or severe weather conditions, to ensure users feel secure.
- **Youthful:** Anya appeals to a younger demographic with her energetic and spirited manner of communication.

Interaction Style:

- **Engaging:** Anya uses conversational language and encourages user interaction with follow-up questions and prompts.
- **Clear and Concise:** Anya communicates weather information succinctly, ensuring users receive the most critical details without superfluous information.
- **Personalized:** Anya tailors her responses based on user preferences and past interactions, providing a personalized experience.

Sample Dialogues:

1. Check Current Weather:

- **User:** "What's the weather like today?"
- **Anya:** Sure, I can help with that. Which city's weather would you like to know?
- **User:** "What's the weather like today in Portland?"
- **Anya:** The current weather in Portland is few clouds with a temperature of 22.69°C and humidity of 40%.

2. Weather Forecast

- **User:** Can you give me the weather forecast for the next 3 days in Los Angeles?
- **Anya:** "The weather forecast for Los Angeles on: Thursday, June 13: clear sky, Avg Temp: 21.36°C Friday, June 14: few clouds, Avg Temp: 21.67°C Saturday, June 15: clear sky, Avg Temp: 25.18°C Sunday, June 16: clear sky, Avg Temp: 24.86°C Monday, June 17: clear sky, Avg Temp: 21.80°C"

3. Weather Alerts:

- **User:** Are there any weather alerts for Chicago?
- **Anya:** There are 12 weather alerts for Chicago: - light rain expected on 6/13/2024, 3:00:00 AM - light rain expected on 6/13/2024, 6:00:00 AM - light rain expected on 6/13/2024, 3:00:00 PM - light rain expected on 6/13/2024, 9:00:00 PM - light rain expected on 6/14/2024, 12:00:00 AM - light rain expected on 6/14/2024, 3:00:00 AM - light rain expected on 6/16/2024, 9:00:00 AM - moderate rain expected on 6/16/2024, 12:00:00 PM - light rain expected on 6/16/2024, 3:00:00 PM - light rain expected on 6/17/2024, 3:00:00 AM - light rain expected on 6/17/2024, 3:00:00 PM - light rain expected on 6/17/2024, 6:00:00 PM.

4. Good Morning Weather Update:

- **User:** Good morning!
- **Anya:** Good morning! Which city's weather would you like to know?
- **User:** Tell me the morning forecast for Seattle.
- **Anya:** Good morning! I couldn't get the weather information for Seattle on today morning. Please try again later.

(Bullet point 4 : If there is any issue in the API to fetch data for the intent this is how we handle an error for best user experience)

5. Good Night Weather Update:

- **User:** Tell me the night weather in New York?
- **Anya:** "Good night! Tonight in New York, it's going to be broken clouds with a temperature of 22.81°C and humidity of 61%.

6. Weather Tomorrow:

- **User:** What's the weather like in Houston tomorrow?
- **Anya:** The weather in Houston tomorrow is expected to be overcast clouds with an average temperature of 30.15°C.

This persona ensures that Anya is not only a weather information provider but also an engaging and reliable assistant who enhances the user experience with her friendly and informative demeanor.

Step 5 - User Testing

Part 1 - Description of Usability Testers

1. Tester 1: Kavya

- **Background:** Kavya is a 24-year-old Computer Science Engineer living in Portland. She has a strong technical background and frequently uses voice assistants like Alexa and Google Assistant in her daily life. Kavya was chosen because she matches the persona created for the application, Anya, and her frequent use of technology makes her an ideal candidate to provide insightful feedback on the usability and functionality of the weather assistant.
- **Reason for Selection:** Kavya's familiarity with voice assistant technology and her alignment with the persona of Anya make her a valuable tester. Her insights

will help identify any technical or user experience issues that may not be immediately obvious.

2. Tester 2: Aashritha

- **Background:** Aashritha is a 30-year-old parking enforcement officer in Portland. She uses this application to check daily weather updates to know whether to carry a jacket, hat, raincoat, umbrella, or shoes depending on the weather conditions. Aashritha was chosen to test the application because she represents a practical user of weather-related voice assistant technology.
- **Reason for Selection:** Aashritha's practical need for accurate and timely weather updates makes her an ideal tester for the application. Her feedback will ensure the application is useful and reliable for users who depend on it for their daily activities and work requirements.

Part 2 - Usability Testing

Expected Results

Task	Time to Complete	Expected Output
Task 1 How warm is it in San Francisco	5 sec	The current temperature in San Francisco is 17.54°C.
Task 2 Are there any weather emergencies in Syracuse?	7 sec	There are no emergencies alerts for Syracuse at the moment.
Task 3 Give me the weather forecast for Orlando for	10 sec	The weather forecast for Orlando on: Thursday, June 13: light rain, Avg Temp: 28.54°C Friday, June 14: light rain, Avg Temp: 27.95°C Saturday, June 15: overcast clouds, Avg Temp: 29.01°C Sunday, June 16: few

the next 3 days.		clouds, Avg Temp: 29.68°C Monday, June 17: overcast clouds, Avg Temp: 28.63°C
Task 4 Are there any weather alerts for Portland?	10 sec	There are 7 weather alerts for Portland: - light rain expected on 6/15/2024, 3:00:00 AM - light rain expected on 6/15/2024, 12:00:00 PM - light rain expected on 6/15/2024, 3:00:00 PM - light rain expected on 6/15/2024, 6:00:00 PM - light rain expected on 6/15/2024, 9:00:00 PM - light rain expected on 6/16/2024, 12:00:00 AM - light rain expected on 6/16/2024, 3:00:00 AM
Task 5 What's the weather like in Houston tomorrow?	5 sec	The weather in Houston tomorrow is expected to be clear sky with an average temperature of 30.75°C.

Results from Usability Testing:

Task	Time to Complete	User 1 Done?	User 1 - Time	User 2 Done?	User 2 - Time
Task 1 How warm is it in San Francisco	5 sec	Yes	7 sec	Yes	5 sec
Task 2 Are there any weather emergencies in Syracuse?	7 sec	Yes	8 sec	Yes	7 sec

Task 3 Give me the weather forecast for Orlando for the next 3 days.	10 sec	Yes	10 sec	Yes	12 sec
Task 4 Are there any weather alerts for Portland?	10 sec	Yes	12 sec	Yes	11 sec
Task 5 What's the weather like in Houston tomorrow?	5 sec	Yes	7 sec	Yes	7 sec

Part 3 - Likert Scale Testing:

Kavya's experience:

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
The system is easy to use.					X
Beginners can easily navigate the system without prior experience				X	
Is it easy for you to accomplish tasks using the system?					X

The voice assistant responded accurately to my requests.					X
I encountered no issues during my use of the application.			X		
The information provided was clear and helpful.					X
Have you found any aspects of the system to be difficult to use or understand?	X				
I was able to complete all tasks without difficulty.					X
The application's responses were timely.			X		
The application would be useful in my daily life.					X

Aashritha's experience:

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
The system is easy to use.					X
Beginners can easily navigate the system without prior experience					X
Is it easy for you to accomplish tasks using the system?					X
The voice assistant responded accurately to my requests.				X	
I encountered no issues during my use of the application.			X		
The information provided was clear and helpful.					X
Have you found any aspects of the system to be difficult to use or understand?	X				
I was able to complete all tasks without difficulty.				X	

The application's responses were timely.				X	
The application would be useful in my daily life.					X

Part 4 - Open-ended questions:

Open-ended Questions for Kavya and Aashritha

1. What did you like most about the application?

Kavya: The application's ability to provide accurate and timely weather updates was impressive. I also appreciated the natural flow of interaction and the detailed information provided.

Aashritha: I liked that the application provided clear and concise weather updates that are very useful for my daily planning. The interaction felt smooth and natural.

2. What improvements would you suggest for the application?

Kavya: It would be great to have more variety in the responses to make the interaction feel less repetitive. Additionally, including some visual elements could enhance the user experience.

Aashritha: Adding more personalized suggestions based on the weather forecast, such as recommending specific clothing or accessories, would be helpful.

3. Did you find any part of the application confusing or difficult to use?

Kavya: Overall, the application was easy to use. However, there were a few moments where the response time was slightly delayed, which could be improved.

Aashritha: The application was straightforward to use. However, sometimes the responses felt a bit too robotic. Adding more variation in the dialogue could improve the user experience.

Step 6 - Review and Reflection

General Impressions:

- **What worked well in this assignment?** The structure and step-by-step approach provided a clear path for developing the voice assistant application. The integration with Dialogflow and the OpenWeatherMap API worked smoothly, allowing for accurate and timely weather information retrieval.
- **What did not work well?** Some issues were encountered with the fulfillment logic, particularly in handling specific time periods for weather forecasts. Debugging these issues took longer than anticipated, impacting the overall timeline.
- **Are there parts of the assignment you would want to change?** I would like to make changes related to the personal interaction phase where the user can get automatic prompts based on their daily schedule. This would be helpful for users to follow a routine and receive timely weather updates relevant to their daily activities.

Time:

- **How much time did you spend on this assignment?** I spent approximately 4 days on this assignment.
- **What steps took the longest to complete? Why?** The longest step was developing and debugging the fulfillment code, particularly for the GoodMorningIntent and WeatherTomorrowIntent. Handling different time periods and ensuring accurate weather data retrieval required careful logic and extensive testing. Also with the formatting for date and time display for user in the output.
- **Was there anything particularly difficult and time-consuming?** Debugging the fulfillment logic and ensuring the correct handling of slot values were both challenging and time-consuming. This required a deep understanding of both Dialogflow's capabilities and the OpenWeatherMap API.

Testing:

- **In terms of testing, what type of testing turned out to be the most useful?** User testing with real users provided the most valuable insights. It revealed issues that were not apparent during development and highlighted areas for improvement in user interaction.
- **What questions and tasks gave you the most insight into issues with your application?** Tasks related to specific time periods for weather updates (e.g., morning or tomorrow) highlighted issues with how the application interpreted and handled these slot values. Questions about the ease of use and the interactions also provided critical feedback.

- **What did you learn from your user testing?** I learned that while the application was generally easy to use, there were specific areas where the responses could be more natural and engaging. User feedback emphasized the importance of varied responses and more personalized interactions.
- **How would you continue testing this application if you had more time?** I would conduct more extensive usability testing with a broader range of users, including those with different levels of familiarity with voice assistants. Additionally, I would implement automated testing to ensure consistent performance and accuracy across different scenarios.

Future Work:

- **Are there parts of the application that need more work?** Yes, the handling of specific time periods for weather updates needs further refinement to ensure accuracy. Additionally, the variety and naturalness of responses can be improved.
- **What about testing?** More extensive user testing is needed to gather a broader range of feedback. Automated testing can also be implemented to ensure consistent performance and accuracy.
- **Are there features you would have liked to add?** I would like to add more personalized suggestions based on the weather forecast, such as clothing recommendations or activity suggestions. Integrating visual elements and notifications for severe weather alerts would also enhance the user experience.

Demo link: <https://bot.dialogflow.com/e1a72402-ae97-4b73-8349-1d115c327cca>