

Dialogflow α (3 Points)

Experience Prototyping Conversational Interactions

In this assignment, you will start your work toward designing and developing your Dialogflow deliverable. We discussed in class that designing conversational interfaces has unique challenges and that ideation and prototyping methods that work very well in other design problems do not work well here. The good news is that we are also subject matter experts in conversation, but the bad news is that our expertise is encoded in our brains and is not readily available for us to use, what we called *tacit knowledge*. This is where *experience prototyping* comes into the picture: by simulating the social and/or the physical setting for the interaction and acting out the interactions using methods such as *bodystorming*, we unlock our expert knowledge and apply it to the design problem.

In this assignment, you will engage in experience prototyping for a *conversational shopping assistant*, which will serve as the basis for developing the intents and entities for the first prototype of your Dialogflow implementation. Specifically, the Dialogflow β tasks provided below should inform the development of the scenarios in Part 1, and the bodystorming of these tasks in Part 1 should be the basis for the specification of the intents, entities, and responses in Part 2.

Part 1. Experience Prototyping (1.5 Points). The first part of the assignment will involve engaging in *bodystorming* to generate ideas and specifications for your shopping assistant.

Part 2. Agent Specification (1.5 points). In the second part of the assignment, you will build on the outcome of your experience prototyping activity to develop specifications for the agent you will build in Dialogflow.

Submission Details

You will submit a completed version of this document in PDF format to Canvas.

Part 1. Experience Prototyping (1.5 Points). In this step, you will follow a process very similar to the process we followed for the in-class activity on experience prototyping, paying particular attention to *bodystorming* for idea generation. In the context of designing a shopping assistant robot, follow the steps below:

1. *Define context* — This is given to you: users interacting with a conversational shopping assistant embedded within a clothing retail website. There is no deliverable for this step.
2. *Develop scenarios* — Think about how the shopping assistant will help users. What are some tasks the shopping assistant can help users with? Develop 3 scenarios. The tasks from Dialogflow β (provided below) should be the basis of these scenarios. Reviewing the [WiscShop API Docs](#) will also be helpful in developing your scenarios.
3. *Identify design goals* — Determine what the shopping assistants can do to assist in these tasks. Consider aspects of the task where the assistant can bring added value. Our goal is not designing a fully autonomous assistant that could take care of everything with minimal input from the user, but what is called a *mixed-initiative design* where the assistant does what it's good at and the user does what the user is good at.
4. *Setup environment* — You can use the retail store interface provided in [the WiscShop source code](#) and/or another clothing retail store as your environment or prop during your acting.
5. *Act out interaction* — Ask a friend, family member, or another student in class to help you bodystorm user interactions with the shopping assistant to develop ideas and to more concretely define user and system behavior and interactions with the environment. Act out and record a transcript on at least one interaction for each scenario.
6. *Develop insight* — Capture the conversations from your bodystorming session and any other insight you have gained from the previous step in notes and translate them into a flowchart representation of the interaction.

Tasks that your Dialogflow β agent should support are listed below:

- **Login**
 - User is able to login with username and password. You do not need to handle account creation.
 - **NOTE:** It is sufficient if the user enters this information as a text query (typing), in case the username and/or password is hard to parse. It should still be in English, e.g. "Log in with username <username> and password <password>."
- **Queries**
 - *Categories:* User should be able to query about the types of products offered.
 - *Tags:* User should be able to inquire about the types of tags for a specific category.
 - *Cart:* User should be able to request information about what is in their cart (e.g. total number and type of items, total cost, etc.).
 - *Product Info:* User should be able to request information about a product. If the product has reviews, they should be able to inquire about reviews and average ratings.
- **Actions**
 - *Tags:* User should be able to narrow down the search results within a category by specifying tags, e.g. "Show me all the red ones".

- *Cart*: User should be able to add/remove items (or multiple of an item) to/from your cart. They should also be able to clear their cart.
- *Cart Confirm*: User should be able to review, then confirm their cart.
- **Navigation**
 - User should be able to navigate through the application with the voice assistant using natural language, e.g., "Take me to the home page" or "Show me the hats".
 - For a full breakdown of the various routes in the application, see the WiscShop readme.

Your deliverables will be the scenarios and design goals you have focused on, the transcripts of the bodystorming sessions, and a flowchart representation of the conversational capabilities suggested by your experience prototyping through your 3 scenarios. Your flowcharts can be in the form of a graph where the nodes are system behaviors and arrows are user behaviors. To generate flowcharts, you can use [SmartDraw](#) (using your NetID login) or free versions of other tools, such as [LucidChart](#) or [Creatly](#).

<scenarios-and-design-goals>

Scenario 1: Find products in the clothing retail website

Design goals:

- The shopping assistant should be able to provide the existing categories to the user.
- The shopping assistant should be able to find and show products based on the categories provided by the user, including bottoms, hats, leggings, plushies, sweatshirts, and tees.
- The shopping assistant should be able to provide the existing tags to the user.
- The shopping assistant should be able to find and show products under a certain category based on the tags provided by the user. For example, the tag may include women, pants, gret, etc.
- In these cases, the shopping assistant can bring added value to user experience. It first gives a quick introduction to the shop by informing the user about the existing categories and tags. Moreover, the user can more than likely find their desired items quickly based on the categories and tags.

Scenario 2: Obtain information of a certain product

Design goals:

- The shopping assistant should be able to show the details page of a product on the request of the user. The user should be able to obtain information such as the price, image, and description of that certain item.
- The shopping assistant should be able to show the reviews of the product on the user's request. The reviews contain information including stars and review contents.
- In these cases, the shopping assistant can bring added value since the user can directly access the information of a certain product by only providing its name. Instead of clicking through the website and trying to find the item manually, the assistant can save much time and effort.

Scenario 3: Buy the product(s) in the cart

Design goals:

- The shopping assistant should be able to list out all the current items in the cart before the user buys any products in it.
- The shopping assistant should be able to add/remove items in the cart. This also includes having multiple numbers of the same item and adding or removing them.
- The shopping assistant should be able to place an order on the user's request based on the items in the cart.
- Before finally placing the order, the shopping assistant should be able to let the user confirm if they want to place the order. If the answer is yes, then the assistant should proceed. If the answer is no, then the assistant provides an exit for the user to return and perhaps double-check some information.

<bodystorming-transcripts>

The environment is the retail store interface provided in [the WiscShop source code](#).

Scenario 1 transcripts:

Agent: Hello, welcome to WiscShop!

User: What are the categories in the shop?

Agent: We have six categories currently in the shop. They are bottoms, hats, leggings, plushies, sweatshirts, and tees.

User: What are the tags in sweatshirts?

Agent: We have four tags for sweatshirts. They are red, logo, long sleeve, and sweats.

User: Show me all the sweatshirts in the shop.

Agent: Showing all the sweatshirts in the shop.

[Agent navigates the website to the sweatshirts category]

User: Show me the long sleeve sweatshirts.

Agent: Showing the long sleeve sweatshirts.

[Agent filters the sweatshirts category with the long sleeve tag]

Scenario 2 transcripts:

Agent: Hello, welcome to WiscShop!

User: Show me the details of the Wisconsin Football Hat.

Agent: Showing the details of the Wisconsin Football Hat.

[Agent navigates the website to the Wisconsin Football Hat details page]

User: What is the price of it?

Agent: The price of the Wisconsin Football Hat is 30 dollars.

User: Give me some reviews on it.

Agent: The rating for the Wisconsin Football Hat is 5 stars. One review writes: Perfect! Love this hat!

Scenario 3 transcripts:

Agent: Hello, welcome to WiscShop!

User: Show me my cart.

[Agent navigates the website to the user's cart]

Agent: Here is your shopping cart.

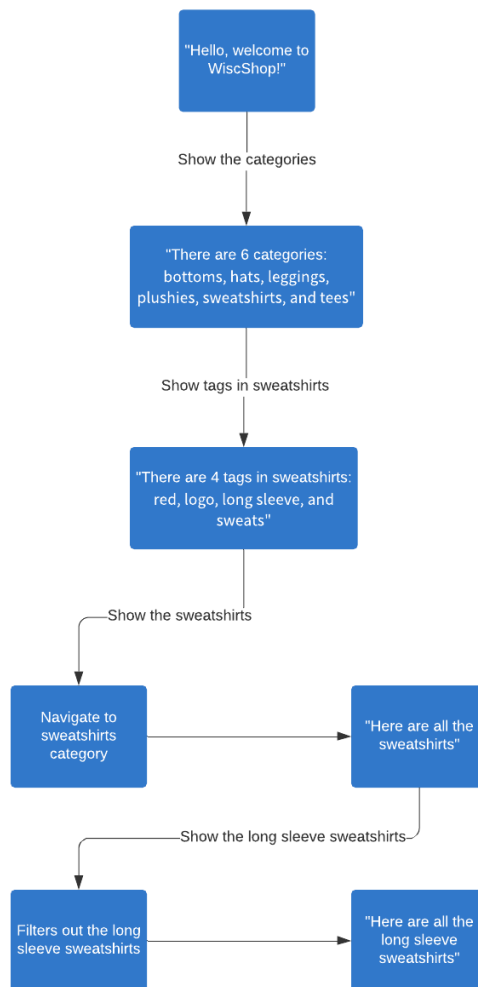
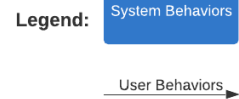
User: Remove the Wisconsin Football Hat.

[Agent removes the Wisconsin Football Hat from the user's cart]
 Agent: The Wisconsin Football Hat has been removed from your cart.
 User: Add one more of the Bucky Crew Neck Sweatshirt.
 [Agent adds one more Bucky Crew Neck Sweatshirt]
 Agent: One more Bucky Crew Neck Sweatshirt has been added to your cart.
 User: I want to place an order.
 [Agent navigates the website to the order review page]
 Agent: Here are all the items in your cart. Do you confirm placing an order?
 User: Yes, please.
 [Agent navigates the website to the order confirmation page]
 Agent: Thank you! Your order has been placed.

<flowchart>

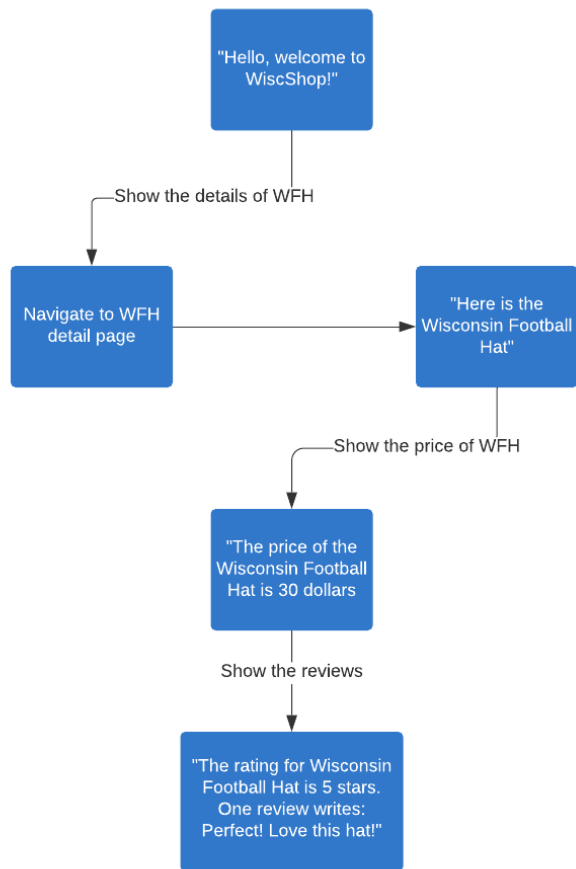
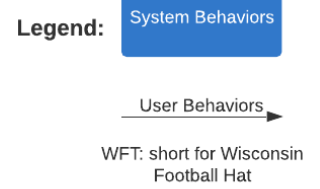
Scenario 1: Find products in the clothing retail website

Yushun Chen | December 2, 2021



Scenario 2: Obtain information of a certain product

Yushun Chen | December 2, 2021

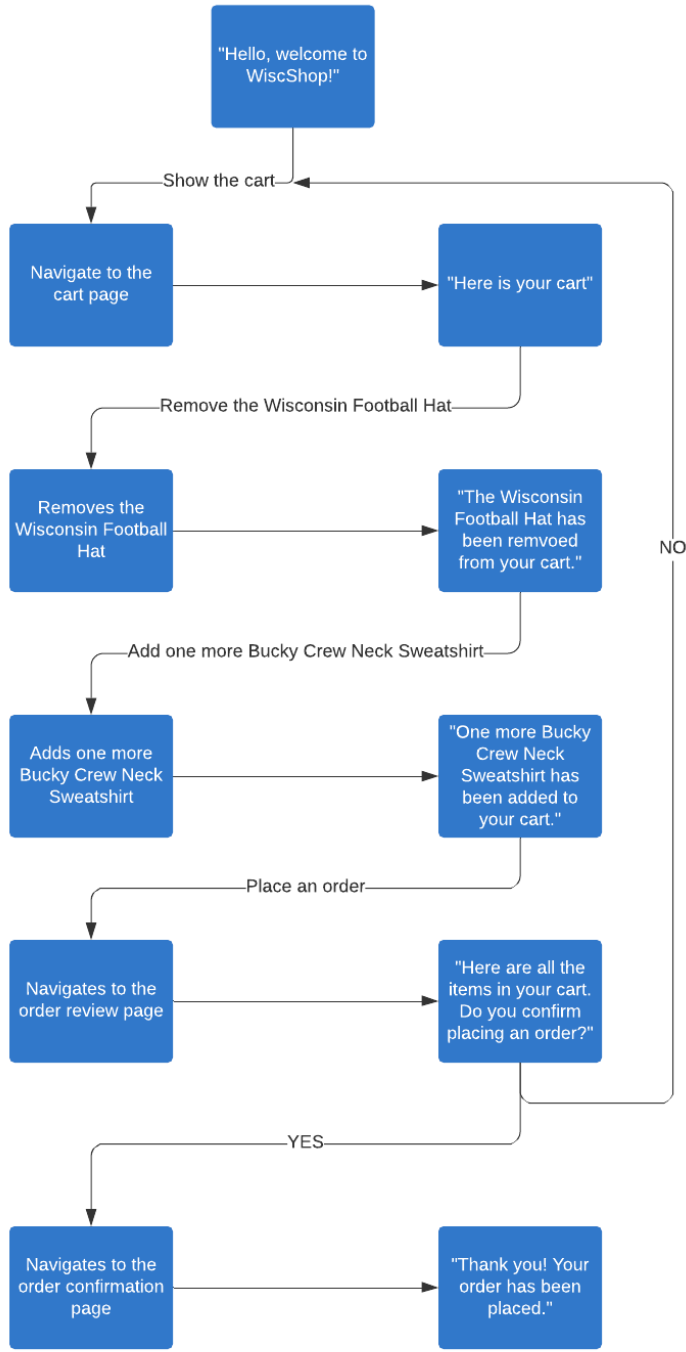


Scenario 3: Buy the product(s) in the cart

Yushun Chen | December 2, 2021

Legend: System Behaviors

User Behaviors →



Part 2. Agent Specification (1.5 Points). In this step, you will apply what you learned in your experience prototyping activity to the design of the agent you will be creating. More specifically, you will draw on the outcome of your bodystorming session to determine the *intents* and *entities* that your agent will utilize in its conversation, and consider how you will use them and server data to provide responses.

If a particular intent or response is infeasible to implement the way you imagined in your bodystorming session, explain why, and propose an intent or family of intents which can be realistically implemented using the Dialogflow framework that will support the same functionality.

In this part, you will provide three main deliverables:

1. A list of all *intents* you will use (provide 10 training examples for each intent).
2. A list of all *entities* (provide at least five examples for each entity) you will be using with your agent.
3. For each *intent*, develop agent responses, specifically what it will *say in reply* (at least three responses to avoid repetition) and what it will *do* to change the GUI.

For a full description of what the GUI can do, and the requirements of the agent, see the Dialogflow β assignment details and the [WiscShop API Docs](#).

<intents-and-training-examples>

Intent 1: Show the categories

Training examples:

1. "Show me the categories."
2. "Give me the categories."
3. "What are the categories in the shop?"
4. "What categories do you have?"
5. "What things in general do you have?"
6. "What does the shop have?"
7. "What can I choose from in the shop?"
8. "How many categories are there?"
9. "What kinds of items do you have?"
10. "What types of items are there?"

Intent 2: Show the tags

Training examples:

1. "Show me the tags of <category name>."
2. "Give me the tags of <category name>."
3. "Show me some details of <category name>."
4. "What are the tags in <category name>?"
5. "What tags do you have in <category name>?"
6. "Does <category name> have any tags?"
7. "What can I choose from <category name>?"
8. "How many categories are there in <category name>?"

9. "What kind of <category name> do you have?"
10. "What types of <category name> are there?"

Intent 3: Show item(s) under a categories or under tag(s)

Training examples:

1. Show me items of <category name>.
2. Give me all <tag description> items of <category name>.
3. List all items of <category name>.
4. Find the <tag description> items of <category name> for me.
5. Take me to the items page of <category name>.
6. Can you get the <tag description> items of <category name>?
7. What are the items under <category name>?
8. What can I buy from <category name>?
9. Are there any <tag description> items under <category name>?
10. What are the choices for <category name>?

Intent 4: Show details of an item

Training examples:

1. Give me the details of <item name>.
2. Show me more of <item name>.
3. Take me to the <item name> details page.
4. Give me more information about <item name>.
5. Show me <item name>.
6. More about <item name>.
7. Provide more details about <item name>.
8. What do you know about <item name>?
9. What about <item name>?
10. Show me anything of <item name>.

Intent 5: Show the price of an item

Training examples:

1. Show me the price of <item name>.
2. How much is <item name>?
3. Give me the price of <item name>.
4. How much does <item name> cost?
5. Price on <item name>?
6. How much do I need to pay for <item name>?
7. Tell me <item name>'s price.
8. How much is the cost of <item name>?
9. Would you please check the price for <item name>?
10. How much are the <plural item name>?

Intent 6: Show the reviews of an item

Training examples:

1. Show me the reviews of <item name>.
2. Check the reviews of <item name> for me.
3. Tell me people's opinion on <item name>.
4. Can you show me the reviews of <item name>?
5. What do people say about <item name>?
6. How is <item name> rated?
7. Are people satisfied with <item name>?
8. Any reviews on <item name>?
9. What is <item name>'s rating?
10. Are there any reviews of <item name>?

Intent 7: Show the cart

Training examples:

1. Show me the cart.
2. Take me to my cart.
3. What do I have in the cart?
4. List the items in my cart.
5. Show me the items in my cart?
6. Is my cart empty?
7. Are there any items in the cart?
8. What are the items in my cart?
9. Go to my cart.
10. How many items are there in the cart?

Intent 8: Add item(s) to the cart

Training examples:

1. Add <item name> to the cart.
2. Can you add <item name> to the cart?
3. I want <item name>.
4. Add one more of <item name> to my cart.
5. I want to buy <item name>.
6. Can I have <item name>?
7. I'll take <item name>.
8. I've decided to buy <item name>.
9. Can I add <item name> to my cart?
10. I want to add <item name> to the cart.

Intent 9: Remove item(s) to the cart

Training examples:

1. Remove <item name> to the cart.
2. Can you remove <item name> from the cart?
3. I don't want <item name>.

4. I don't want to buy <item name>.
5. Can I remove <item name> from the cart?
6. I'll pass on <item name>.
7. I've decided not to buy <item name>.
8. I want to remove <item name> from the cart.
9. I'll buy <item name> next time.
10. <Item name> is not something I want.

Intent 10: Place an order

Training examples:

1. Place an order.
2. I'd like to place an order.
3. Can I place an order?
4. I'd like to buy the items in my cart.
5. Can I check out?
6. I'd like to check out.
7. Can I buy the items now?
8. Where can I check out?
9. I'd like to proceed on my cart.
10. Purchase items in cart.

<entities-and-examples>

Entity 1: Category

Examples:

1. What are the categories in the shop?
2. How many categories are there in the shop?
3. What are the tags of hats?
4. What are the items of leggings?
5. Are there any logo plushies?

Entity 2: Tag

Examples:

1. What are all the tags of bottoms?
2. Are there any grey pants?
3. Show me long sleeve sweatshirts.
4. What are the items of the tag red in tees?
5. Show me white hats in the shop.

Entity 3: Shop item

Examples:

1. Show me Bucky Crew Neck Sweatshirt.
2. Show me details of the Wisconsin Football Hat.
3. Add White Wisconsin Visor to the cart.

4. Remove Bucky Crew Neck Sweatshirt from the cart.
5. Do you have Game Day Bucky Plush?

Entity 4: Cart

Examples:

1. Show me the cart.
2. Add White Wisconsin Visor to the cart.
3. Remove Bucky Crew Neck Sweatshirt from the cart.
4. Empty the cart.
5. Place an order on the items in my cart.

<intent-responses-and-procedures>

Intent 1: Show the categories

Response:

1. We have six categories currently in the shop. They are bottoms, hats, leggings, plushies, sweatshirts, and tees.
2. Here are all the categories: bottoms, hats, leggings, plushies, sweatshirts, and tees.
3. Currently, we have bottoms, hats, leggings, sweatshirts, and tees in our shop.

Procedure: No GUI change here. The user stays on the home page.

Intent 2: Show the tags

Response:

1. We have four tags for <category name>. They are <tag names>.
2. Here are the tags for <category name>: <tag names>.
3. <Category name> has <number of tags> tags. They are <tag names>.

Procedure: No GUI change here. The user stays on the home page or on the category page.

Intent 3: Show item(s) under a categories or under tag(s)

Response:

1. Here are all the items of <category name>.
2. Great! I have found the <tag description> items under <category name>.
3. We have these items of <category name> in our shop.

Procedure: The shopping assistant should take the user to the page of items under their specified categories and possibly filter the items based on the tags provided by the user.

Intent 4: Show details of an item

Response:

1. Here are more details about <item name>.
2. I have found more of <item name> for you.
3. Here you go, <item name>.

Procedure: The shopping assistant should take the user to the detail page of their specified item.

Intent 5: Show the price of an item

Response:

1. The price for <item name> is <price>.
2. It costs <price> for you to purchase <item name>.
3. You would spend <price> on <item name> if you buy it.

Procedure: No GUI change here. The user stays on the detail page of a certain item.

Intent 6: Show the reviews of an item

Response:

1. Here are some reviews on <item name>. One review writes <review content>.
2. Here is what people say about <item name>. One user said <review content>.
3. This item has a rating of <review rating>. Some people reviewed that <review content>.

Procedure: No GUI change here. The user stays on the detail page of a certain item.

Intent 7: Show the cart

Response:

1. Sure, here is your cart.
2. Currently, you have <item names> in your cart.
3. Here you go, there are <number of items> in your cart.

Procedure: The shopping assistant should take the user to the cart page.

Intent 8: Add item(s) to the cart

Response:

1. <Item name> has been successfully added to cart!
2. I have added <item name> to the cart.
3. Success! <Item name> is in your cart now.

Procedure: The shopping assistant should update the cart page by adding item(s).

Intent 9: Remove item(s) to the cart

Response:

1. <Item name> has been successfully removed from the cart!
2. I have removed <item name> from the cart.
3. Success! <Item name> is no longer in your cart.

Procedure: The shopping assistant should update the cart page by removing item(s).

Intent 10: Place an order

Response:

1. Here are all the items in your cart. Do you confirm placing an order?
2. Please review your cart. Do you confirm placing the order?
3. Are you sure about placing your order?

Procedure: The shopping assistant should take the user to the order review page. If the user confirms the order, it should take the user to the order confirmation page. Otherwise, the shopping assistant should probably take the user back to the cart.