Assignments — Week 13 | Design
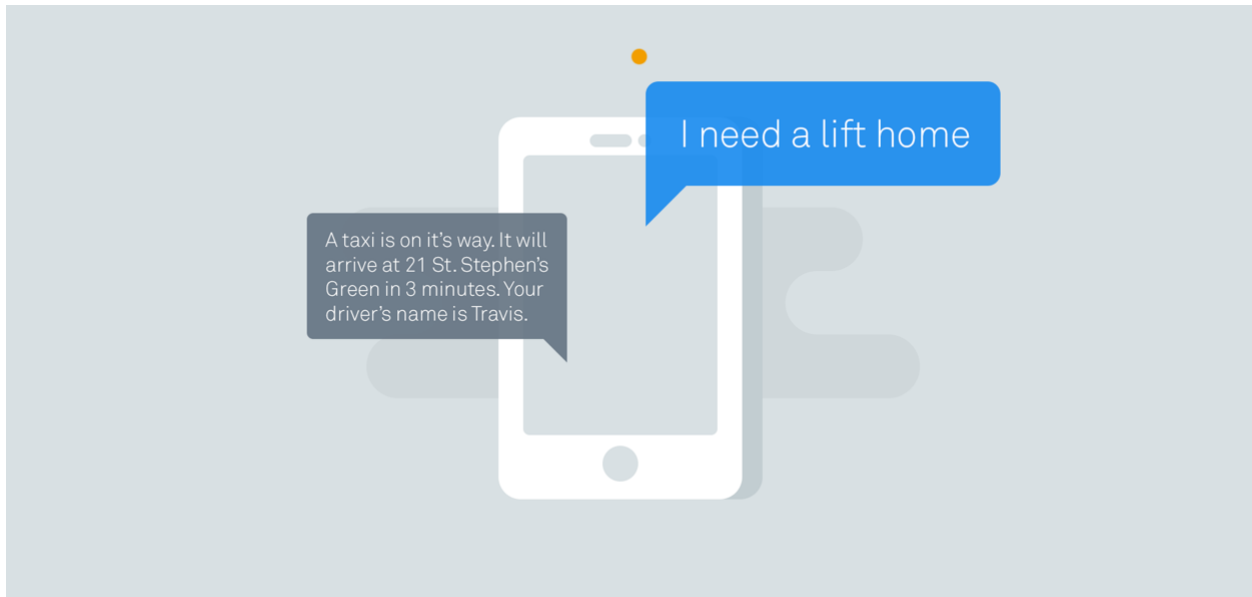# Experience Prototyping Conversational Interactions

Image source

In this assignment, you will start your work toward designing and developing your Module 3 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 locked into brain mechanisms 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 the first part of the assignment, you will engage in experience prototyping of a *conversational shopping assistant*, which will serve as the basis for developing the intents and entities for the first prototype of your Dialogflow implementation.

Part 1. Experience Prototyping. 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.

2. *Develop scenarios* — Think about how the shopping assistant will help users. What are some tasks the shopping assistant can help users with? Develop 2-3 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. *Set up environment* — You can use the retail store provided with Module 3 starter code and/or another clothing retail store as your environment.
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 concrete define user and system behavior and interactions with the environment. Act out 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.

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. Your flowcharts can be in the form of a graph where the nodes are system behaviors and arrows are user behaviors. You can use a flowcharting application such as [Smart Draw](#).

<scenarios-and-design-goals>
Scenarios:
1. Help user to find some products that fit to their preference
2. Purchase the products based on the information that user provided
3. Check the order history or shipping status

Design:
1. Search or retrieve the requested information from the online shopping database and report to users
2. Filter the information from database based on the users' requirements
3. Initial the transaction based the products and users' payment information
4. Lookup/search functions to find the order information based on the information that users provided

<bodystorming-transcripts>

I performed this bodystorming with my roommate Houqi Li who is a student double major in Math and Computer Science at University of Wisconsin – Madison.

Scenarios 1 - Help user to find some products that fit to their preference:

Shopping Assistant: Hello, I'm your voice shopping assistant, how can I help you?

Houqi: I want to buy a hoodie with Stadium Fleece on it.

Shopping Assistant: Gotcha, I found 5 products satisfying your requirements, which style do you want, men or women?

Houqi: I want the men's style.

Shopping Assistant: Ok, there are two products you many like it. What size do you want?

Houqi: I want the extra-large one.

Shopping Assistant: I found a product hoodie with Stadium Fleece extra-large size that fits to your need.

Houqi: Thanks.

Scenarios 2 - Purchase the products based on the information that user provided

Shopping Assistant: Hello, I'm your voice shopping assistant, how can I help you today?

Houqi: I want to buy a hoodie with Stadium Fleece extra-large size in men's style.

Shopping Assistant: Let me take a look on our storage. How many quantities do you need?

Houqi: I only need 1 hoodie.

Shopping Assistant: It seems we have enough stock for you. Do you want to continue with the shipping and payment information?

Houqi: Yes, sure.

Shopping Assistant: What's your shipping address?

Houqi: My shipping address is 777 University Ave, Madison, WI, 53715

Shopping Assistant: Gotcha, what's your phone number?

Houqi: My phone number is 608-888-8888.

Shopping Assistant: Sounds good. What's your credit card number and expiration date?

Houqi: My credit card number is 8888-8888-8888-8888, and it expires on July 2025.

Shopping Assistant: Hold on. Let me put the information into the system. Congratulations, you order has been placed and your order number is 1234-5678. You can chat with me to check your order status anytime you like.

Houqi: Thanks.

Scenarios 3 - Check the order history or shipping status

Shopping Assistant: Hello, I'm your voice shopping assistant, how can I help you today?

Houqi: I ordered a hoodie last week and I want to check my order status.

Shopping Assistant: Ok, I can definitely help you with that. Do you have your order number with you?

Houqi: Sorry I forgot it.

Shopping Assistant: No worries. Let me try the other ways to help you out. What is your shipping address?

Houqi: My shipping address is 777 University Ave, Madison, WI, 53715

Shopping Assistant: I found an order associated with your shipping address. In order to verify your identity, what is your phone number?

Houqi: My phone number is 608-888-8888.

Shopping Assistant: Gotcha. What is your last 4 digits of your credit card associated with your order?

Houqi: It should be 8888.

Shopping Assistant: Ok, I found your order. Your order number is 1234-5678. Currently, the order status is shipped with UPS and the shipping number is 1234-5678-9012. The package is expected to deliver by the end of the day tomorrow.

Houqi: Thank you for your help.

<flowchart>

Please refer to the following graph about my flowchart

Part 2. Dialogflow Implementation. In this step, you will apply what you learned in your Experience Prototyping 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. 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 5 examples for each entity) you will be using with your agent.
3. For each *intent*, write a brief description of what the agent will *say in reply* 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 Canvas Assignment](#) and [API readme](#).

<intents-and-training-examples>

1. Request Product Info
- Please let me know the information about *Jump Around Shirt*.
- How about the *Jump Around Shirt*?
- What do you know about the *Jump Around Shirt*?
- Do you know anything about *Jump Around Shirt*?
- What is *Jump Around Shirt*?
- What are the reviews for *Jump Around Shirt*?
- What are the average ratings *Jump Around Shirt*?
- Tell me about the product descriptions for *Jump Around Shirt*?
- I want to know the details about *Jump Around Shirt*.
- Let me know the product information about *Jump Around Shirt.*

2. Inquiry about Category
- What kind of products are your offering?
- What kind of clothes do you have?
- Let me know the category that you are offering
- Tell me the about the category that you have
- What are the different groups of products do you have?
- Do you have different type of the products that I can choose from?
- I need to know the category
- I want to know the variety you have on your products
- Could you please provide me the category of your product?
- How do you know about the category of your product?

3. Inquiry About Tags
- What different styles do you have for *tees*?
- What variety do you have of the *tees*?
- How do you know about the *tees*?
- Let me know different product you have of *tees*?
- What kind of *tees* do you have?
- Which *tees* do you have in stock?
- What do you know about the variety of *tees*?
- I want to know the tags for *tees*.
- Please tell me about the tags for *tees*.
- What are the tags for *tees.*

4. Narrow Down the Tag
- Show me all the *red* ones
- I want to buy *red tees.*
- Do you have any *red tees*?
- What kind of the *red tees* do you have?
- Let me know the *red tees* that are available in your website.
- Could you please tell me the *red tees* that you have?
- I'm wondering if you have any *red tees*.
- I really like wearing *red tees*.
- *Red tees* are my favorite type of clothes.
- Search the item for *red tees*.

5. Review Cart
- What are the items in my cart?
- How much do I plan to buy?
- How many items do I plan to buy?
- Do I have any items in my cart?
- Do you have any information about my cart?
- Cart Summary
- Tell me about my cart
- Let me know the items in my cart
- Do you have many items in my cart?
- What do you have in my cart?

6. Add Item to Cart
- Add this one to my cart
- Add the *red tees* to my cart
- Could you please help me grab this item?
- Help me add this one to the cart please.
- I would like to buy this one
- Please help me bring it to my cart.
- I need this item to be added to my cart
- Why not bring this *red tee* to my cart?
- I want to put this item to my cart
- Try to help me carry this item to my cart.

7. Remove item from Cart
- Remove this one to my cart
- Remove the *red tees* to my cart
- Could you please help me grab this item out of my cart?
- Help me remove this one from my cart please.
- I'm not planning to buy this one anymore.
- Please help me bring it out of my cart.
- I need this item to be removed to my cart
- Why not bring this *red tee* out of my cart?
- I want to take this item away from my cart
- Try to help me take this item out of my cart.

8. Confirm Cart
- Those are all the things that I'm planning to buy
- The items in shopping cart look goods to me
- I would like to confirm the item in my shopping cart
- I'm ready to start to check out
- I have finished shopping
- I decide to purse to the next step after reviewing my cart
- I'm planning to buy all the items in my cart
- I need to check out the items in my cart
- I want to proceed to the payment
- Buy all the items in my cart

9. Navigation Around the App
- Take me to the *home page.*
- Direct me to the *home page.*
- Where is my *home page*?
- How can I get to the *home page*?
- Show me the *home page*.
- I want to go to *home page*.
- I need to go to *home page*.
- Going back to *home page*.
- Could you please lead me to the *home page*?
- Please guide me to the *home page*.

<entities-and-examples>

1. Color
- Blue
- Green
- Yellow
- Black
- White
- Red
- Purple

2. Object
- Cart: Shopping Cart, Bag, Shopping Bag, Trolley, Basket
- Category: Group, Type, Class, Section, Kind
- Tags: Style, Label, Mark, Tally, Stamp
- Items: Products, Cloth, Commodity, Thing, Merchandise
- Product Info: Product Detail, Product Description, Product Information, Product Review
- Price: Dollar, Bill, Fee, Fare, Expense, Payment
- Quantity: Bulk, Capacity, Volume, Amount

3. Tags
- men: Male
- women: Female
- red: dark red, light red
- white: dark white, light white
- logo: icon
- printed: engraved
- cotton
- polyester
- shortsleeve

- longsleeve
- shorts
- pants
- sweats
- hoodies
- grey: dark grey, light grey
- youth: teen
- adult
- embroidered: embellish
- black: dark black, light black
- fuzzy: downy
- badger: UW-Madison, UW, Madison, Wisconsin
- pillow: Cushion
- keychain
- visor: Cap, Hat

4. Category
- hats: Cap, Visor
- sweatshirts:
- plushes
- leggings: yoga pants
- tees: shorts
- bottoms: pants

5. Navigation
- Home page: home, main page, default
- Shopping Page: Product Page, Item Page
- Shopping Cart: Cart, Shopping Bag
- Payment Page: payment, check-out
- Order Page: order status, order history

6. Actions
- Take: Direct, Guide, Lead
- Show: Tell, Depict, Speak
- Add: Bring, Put, Increase
- Remove: Decrement, Delete, Clear Away, Pull Out
- Continue: Next step, Move next, Show next, Show More
- Back: Go Back, Move Backward, Undo
- Confirm: Decide, Make Sure, Affirm, Approve, Certify

<intent-responses-and-procedures>

1. Request Product Info
- Response: "Gotcha, here is the information I found about the product you are looking for. The descrption for product is xxxx. Currently, it sells as xxx dollars. Do you want to continue to make the purchase?"
- Action: Through checking the tag and category information that user provided. It will find the best match item id by fetching the product lists. Then it will fetch the detail product information from the server and read the information to user.

2. Inquiry about Category
- Response: "I'm here to help you out. We are the largest online clothing retail website. We have categories various from xxx, xxx, xxx, to, xxx. Please let me know which category that you want to explore.
- Action: Through fetching the categories server, it will have a list of categories for the products. Then it will read the list to user and wait their response on which one they decide to go.

3. Inquiry About Tags
- Response: "I'm glad to help you. I found xxx items that fit to your preference. They are: xxx, xxx, xxx, and, xxx. Do you want to narrow down your result by saying the tags or styles that you like?
- Action: Through checking the category information that user provided. It will find a list of items with the products name and id by fetching the products list. Then read the product information to user and wait to user's response on tags in order to narrow down the result.

4. Narrow Down the Tag
- Response: "It seems I have found a product that satisfy your needs. The product name is xxx with a description of xxx. The unit price is xxx. Would you like to continue add to the cart?
- Action: Through checking the detail tags information that user provided. It will find the best match item from the list of items that generate before. Then it will fetch the detail item information and read them to users. In addition, it will wait user's response in order to add them to the cart.

5. Review Cart
- Response: "I hope you have a wonderful shopping day. Currently, there are xxx items in you cart. They are xxx, xxx, xxx, and xxx. The total dollars are xxx. Would you like to proceed to make the payment?"
- Action: It will traverse the array which store the object information for each item that user added to the cart before. Then it will add up all the dollars for each item and speak the amount to the users.

6. Add Item to Cart
- Response: "I have just added your item xxx to the cart. Currently, there are xxx items in your cart and the total amount is xxx dollars. Do you want to add more items?"
- Action: It will push the current item to the array that store the object information that you have added to the cart. Then it will traverse the array and find the price and quantity information.

7. Remove item from Cart
- Response: "I have just removed your item xxx to the cart. Currently, there are xxx items in your cart and the total amount is xxx dollars. Do you want to add more items?"
- Action: It will remove the current item from the array that store the object information that you have added to the cart. Then it will traverse the array and find the updated price and quantity information.

8. Confirm Cart
- Response: "Congratulations! You have found all the products that you need. Currently, there are xxx items in your cart and the total amount is xxx dollars. Do you want to continue to make the payment?"
- Action: It will traverse the array that store the object information that you have added to the cart. Then it read the final price and quantity information to the users.

9. Navigation Around the App
- Response: "Gotcha. I'm on the cart review page right now. Please let me know if you plan to visit another page and I can take you over there.
- Action: It will fetch the information of the page based on the user request. Then it will read the information on the new page to the users.