Building User Interfaces

Design Principles for

Voice User Interfaces

Professor Bilge Mutlu

What will we learn today?

— Design principles for conversational interfaces

— Usability heuristics for conversational interfaces

Design Principles for Conversational Interfaces

Elephant in the Room

Recap: Definition of Usability: The effectiveness, efficiency, and satisfaction with which a specified set of users can achieve a specified set of tasks in a particular environment. — ISO 9241-11

Conversational interfaces are almost always less effective, less efficient, and less satisfactory than graphical user interfaces.

Effectiveness: Conversational interfaces are more error prone due to technology, ambiguities, and environmental influences.

Efficiency: Using conversational interfaces is almost never as fast as using graphical user interfaces.

Satisfaction: Interacting with conversational interfaces can be awkward, socially inappropriate, and frustrating.

So, what is the point of conversational interfaces?

Where do these interfaces deliver value?

1. Streamlining app installation, login, payment, notifications, and and so on in a conversational paradigm.[1]

2. In some contexts, e.g., while driving, CIs are more effective, efficient, and satisfactory due to resource constraints.

3. CIs address many accessibility problems, including vision (e.g., blindness), motor (e.g., tremor), and cognitive (e.g, dyslexia) deficiencies.

[1] Grover, 2016, Bots won't replace apps. Better apps will replace apps.

Design Principles

Gricean Maxims[2]

Definition: Proposed by Paul Grice, conversations follow the cooperative principle and four key maxims:

— Maxim of quality (truthful and accurate communication)

— Maxim of quantity (just the right amount of information)

— Maxim of relevance (appropriate and relevant information)

— Maxim of manner (clear, cooperative communication)
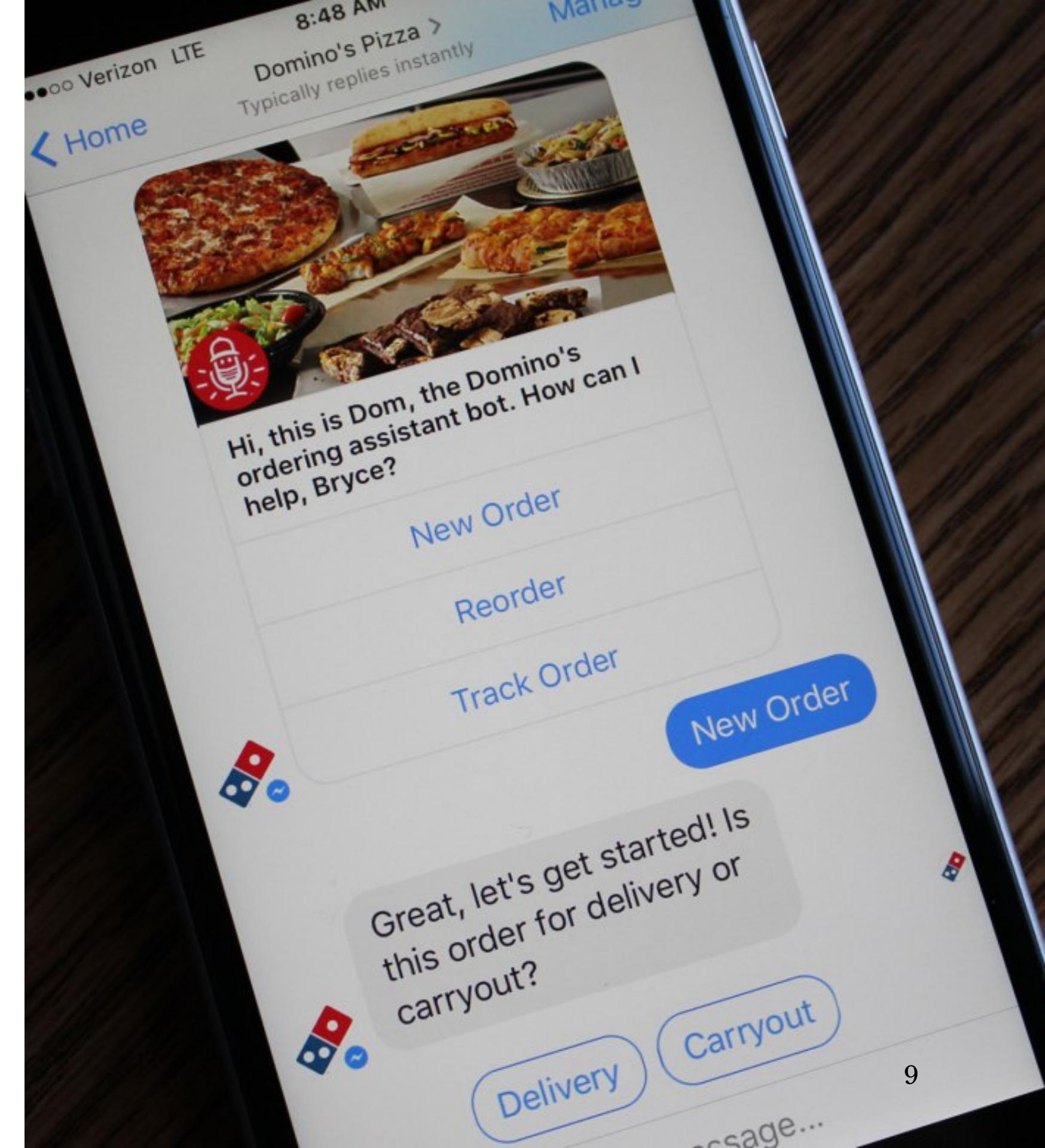
[2] Grice, 1975, Logic and Conversation

Multimodality[3]

Definition: Multimodal interfaces utilize multiple modalities, including visual information, speech, touch, and so on, in user experiences they afford.

Most conversational interfaces are multimodal interfaces.

[3] Image source

Multimodality Principle: Take advantage of other modalities, e.g., visual information, vibrations, etc., wherever appropriate.

Using multimodal components, you can provide users with breaks for decision making, interruptions, etc.

Potential caveats:

— Ask, "does my interface still support a speech-only interaction?"

— The conversational and other components must be designed together to fit within the conversation.

Interaction Paradigm
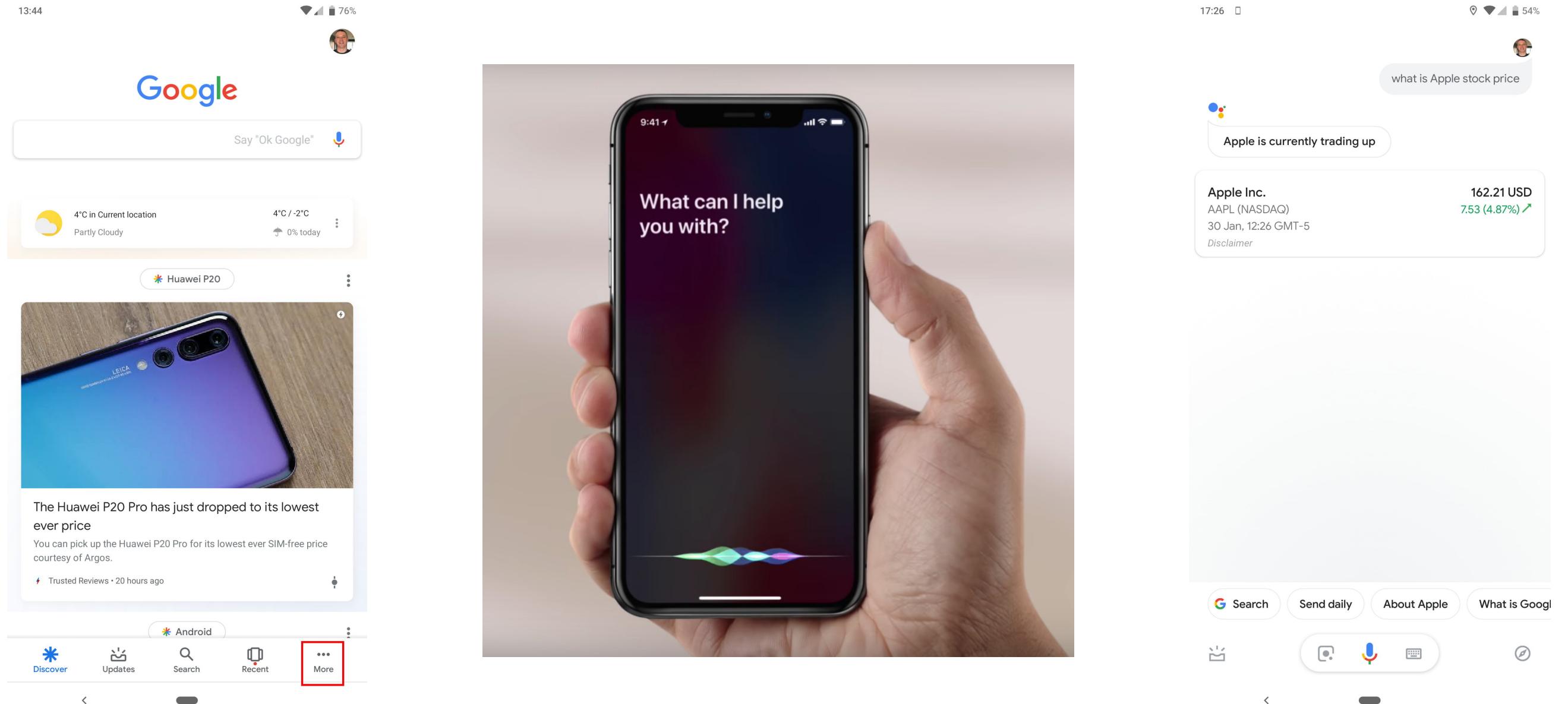
Conversational interfaces can follow different paradigms depending on the context of use and the design of the application:

— Command-and-control interfaces (always-on voice assistants)

— Conversational interfaces (chatbots, task assistants, social robots)

Command-and-Control Interfaces

Definition: Interfaces where speech input is mapped to specific system functions that are called immediately. These interfaces commonly utilize:

1. Expressing user intent using a wake word (e.g.,"OK, Google") or the pressing of a button (e.g., home button in the iPhone)

2. Indicating listening and understanding

3. Executing the mapped function

© Building User Interfaces | Professor Mutlu | Lecture 19: *Design Principles for Voice User Interfaces*

Conversational Interfaces

Definition: Interfaces where the interaction with the system has the characteristics of human conversations, including turn taking, theoretically infite depth, conversational markers, etc.

Turn-taking

Definition: Speaking turns is the core, cooperative structure of conversations that involves one speaker at a time and an explicit exchange of tokens.

Principles:

1. One speaker at a time — *transparency in who is speaking*

2. Turn exchanges — *explicit signaling of what will speak next*

3. Interruption handling — *very difficult with CIs*

Conversational Markers[5]

Definition: Speech cues that indicate the state or the direction of the conversation. Types of conversational markers:

1. Timelines ("First," "Halfway there," "Finally")

2. Acknowledgements ("Thanks," "Got it,", "Alright," "Sorry about that")

3. Positive feedback ("Good job," "Nice to hear that")

[5] Pearl, 2016, Designing Voice User Interfaces: Principles of Conversational Experiences

Assistant: I'll be asking you a few questions about your health. First, how many hours of sleep did you get last night?

User: About seven.

Assistant: Good job. And how many serving of fruits and vegetables did you eat yesterday?

User: Maybe four.[5]

[5] Pearl, 2016, Designing Voice User Interfaces: Principles of Conversational Experiences

Assistant: Got it. Last question—were you able to take your medication last night?

User: Yes.

Assistant: All right. That's it for now. I'll talk to you again tomorrow. Goodbye.[5]

[5] Pearl, 2016, Designing Voice User Interfaces: Principles of Conversational Experiences

Confirmations[5]

Definitions: CIs are designed with explicit forms of confirmation to improve system usability and transparency.

Can be explicit vs. implicit and speech-based vs. non-speech based (visual, action).

[5] Pearl, 2016, Designing Voice User Interfaces: Principles of Conversational Experiences

Explicit confirmation: Requiring the user to confirm: "I think you want to set a reminder to 'buy insurance before going skydiving next week.' Is that right?"[5]

Implicit confirmation: Letting user know what was understood: "Ok, setting a reminder to buy insurance..."

[5] Pearl, 2016, Designing Voice User Interfaces: Principles of Conversational Experiences

Error Handling

Definitions: Deviations from expected conversational flow due to technical mistakes, unexpected user behavior, environmental influences, etc.
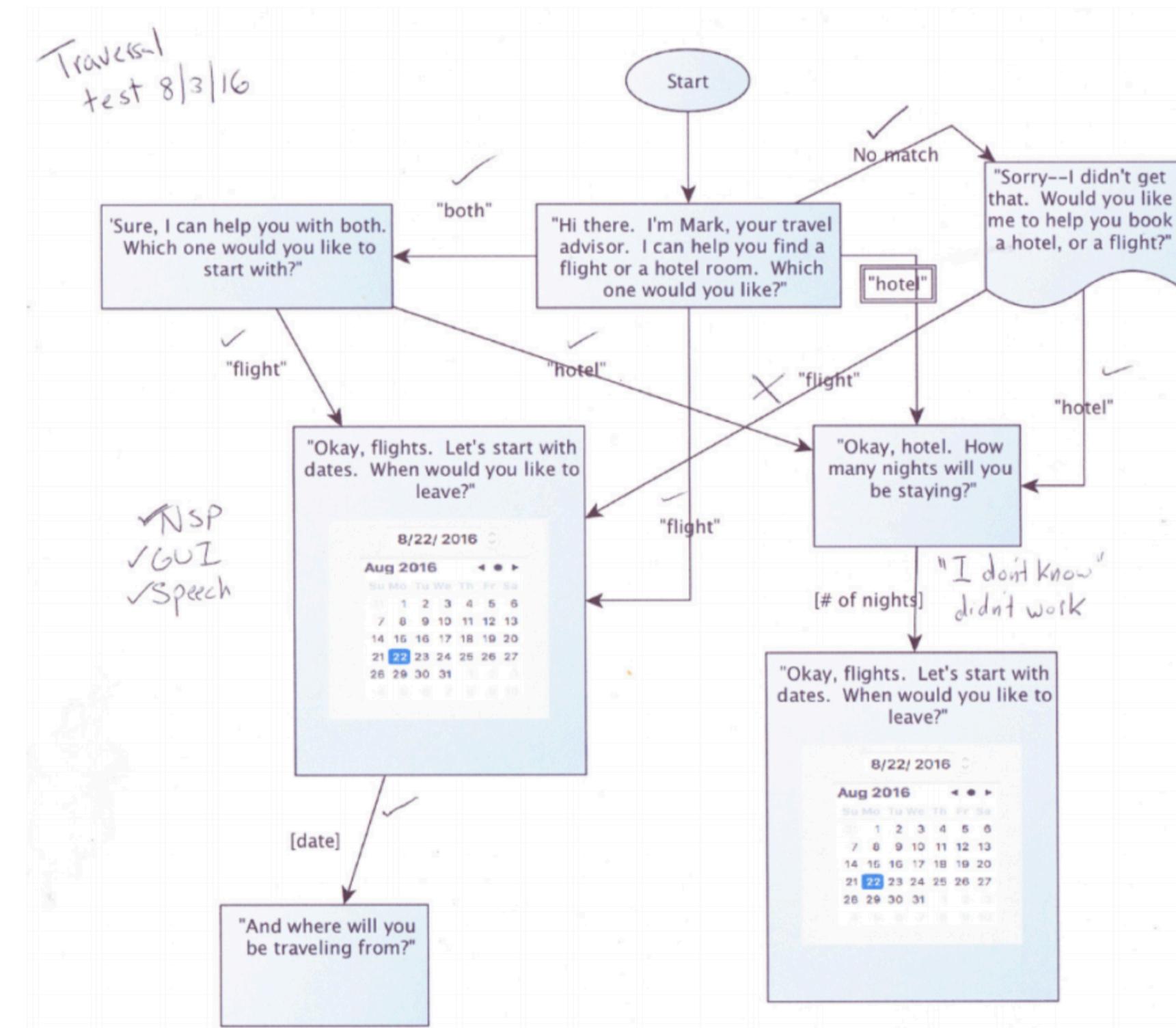
Types of errors:[5]

1. No speech detected

2. Speech detected, but nothing recognized

3. Something was recognized correctly, buyt the system does the wrong thing with it

4. Something was recognized incorrectly

[5] Pearl, 2016, Designing Voice User Interfaces: Principles of Conversational Experiences

Flowcharting Conversational Interactions[5]
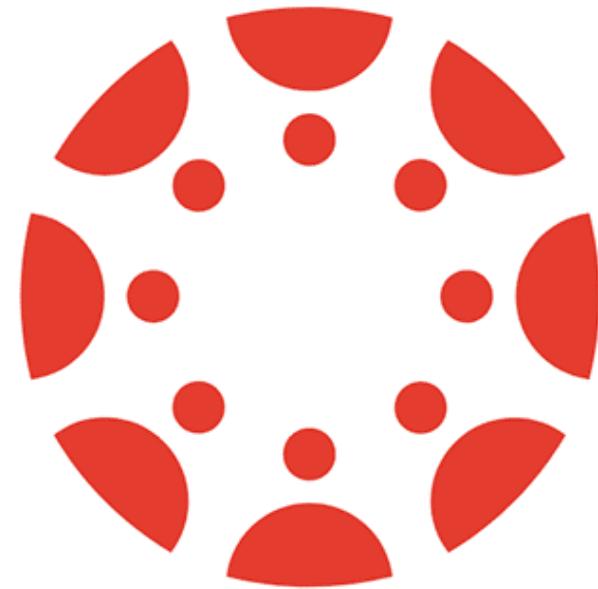
The most commonly used method of modeling and prototyping conversational interactions is defining flows that show how the interaction will flow depending on system state, user behavior, or external influences.

[5] Pearl, 2016, Designing Voice User Interfaces: Principles of Conversational Experiences

Quiz 1
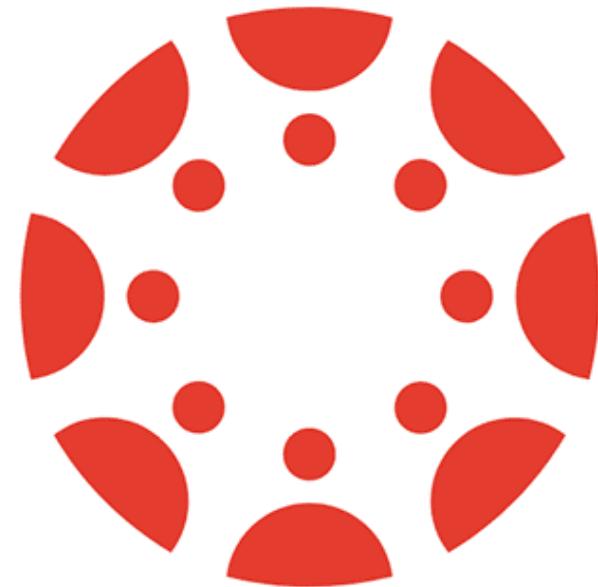
Complete the Canvas quiz.

Quiz 2

Complete the Canvas quiz.

Usability Heuristics for Conversational Interfaces

Recap: What are Usability Heuristics?[6]

Definition: Developed by Jacob Nielsen, heuristic evaluation involves having a small set of evaluators examine the interface and judge its compliance with recognized usability principles (the "heuristics").

[6] NN/g: How to conduct a heuristic evaluation

Heuristics for Conversational Interfaces[7]

Seventeen heuristics that fall into five broad categories:

1. General

2. Conversational style

3. Guiding, Teaching, and Offering Help

4. Feedback and prompts

5. Errors

[7] Wei & Landay, 2018, Speech-based Conversational Agent Heuristics

General Heuristics

Heuristic #1

S1: Give the agent a persona through language, sounds, and other styles.

— Create an illusion by being consistent.

— Make sure to do this without being distracting.

Heuristic #2

S2: Make the system status clear.

— Use verbal, sound, or multimodal feedback.

— Communicate delays immediately and give feedback while "busy".

Heuristic #3

S3: Speak the user's language.

— Use words, phrases and concepts familiar to end users, rather than system-oriented or technical jargon.

Heuristic #4

S4: Start and stop conversations.

— Use a wake word to start a conversation, but don't require it again in the same conversation.

— Gracefully end conversations when the user is done.

Heuristic #5

S5: Pay attention to what the user said and respect the user's context.

— Leverage user input when it can be used as a parameter to a command.

— Remember what the user has said in the current conversation.

— Use context you already know about the user to fill in fields, but confirm them.

— Use context to respond intelligently (e.g., location/environment, time constraints, # of users, user identity/age).

Conversational Style

Heuristic #6

S6: Use spoken language characteristics.

— Use discourse markers as part of confirmations and prompts to make conversation more natural (e.g., "next", "and", "so", "actually", "sure", "ok", "got it").

— Leverage prosody, including rhythm, tone, pauses, emphasis, discourse fillers (e.g., "uh", "uhm", "hmm", "ah", "like").

Heuristic #7

S7: Make conversation a back-and-forth exchange.

— Don't always prompt for everything all at once.

— Take turns and don't let instructions get in the way.

— Give users a chance before jumping in.

Heuristic #8

S8: Adapt agent style to who users are, how they speak, and how they are feeling.

— Users prefer agents that have conversational style similar to their own — match it.

— Match the user's emotion, gender, and personality.

Guiding, Teaching, and Offering Help

Heuristic #9

S9: Guide users through a conversation so they are not easily lost.

— Guide subtly using natural affordances rather than explicitly.

— Guide user towards desired response and cue the user what type of response is desired.

— Allow data to be naturally given in response to single or multiple prompts.

Heuristic #10

S10: Use responses to help users discover what is possible.

— Teach multiple possible ways of asking for a result.

— Use examples in a natural manner rather than teaching commands explicitly.

Feedback and Prompts

Heuristic #11

S11: Keep feedback and prompts short.

— Clear but succinct.

— Keep lists of items short (3-5 max.), and let people ask if they want to hear more.

— Let experienced users have faster and shorter prompts.

Heuristic #12

S12: Confirm input intelligently.

— Confirm input implicitly through results or next prompt. » Confirm irreversible or critical actions explicitly and even allow undo after confirmation.

Heuristic #13

S13: Use speech-recognition system confidence to drive feedback style.

— High: Do it and tell me

— Moderate: Confirm input

— Low: Re-prompt ("Say that again?")

Heuristic #14

S14: Use multimodal feedback when available.

— Lights

— Graphic displays

— Sounds

Errors

Heuristic #15

S15: Avoid cascading correction errors.

— Escalate detail in prompts when input is ambiguous or incorrect.

— If input results in multiple hypotheses, let user select from list with "yes" / "no".

— For error correction, use a different modality or voice response style (e.g., select from a list).

Heuristic #16
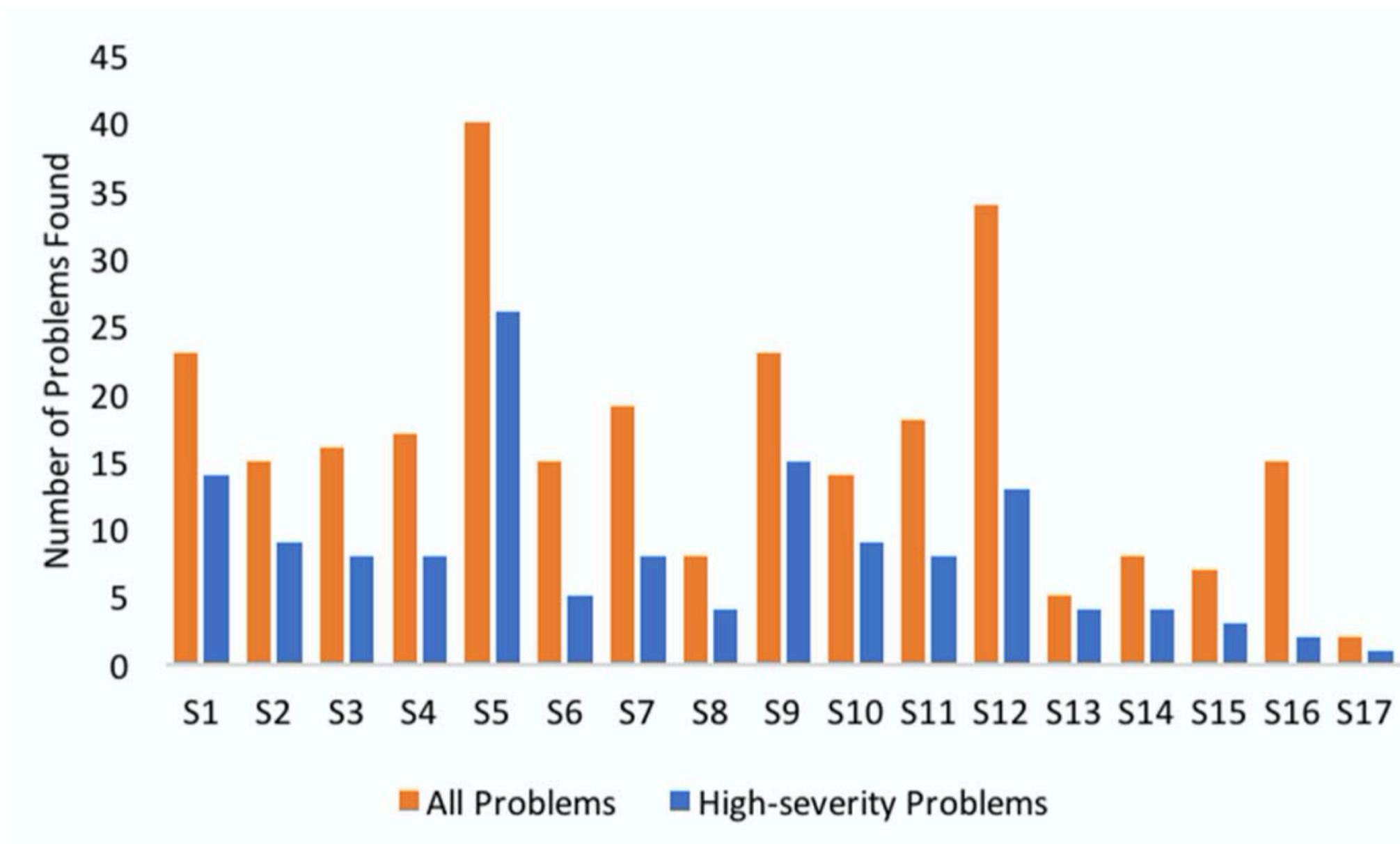
S16: Use normal language in communicating errors.

— Vary (error) prompt wording on re-prompts.

— Don't blame the user for errors (don't say: "that was not a valid response").

— Don't show mock concern (don't say: "I'm sorry. I did not understand the response I heard.").
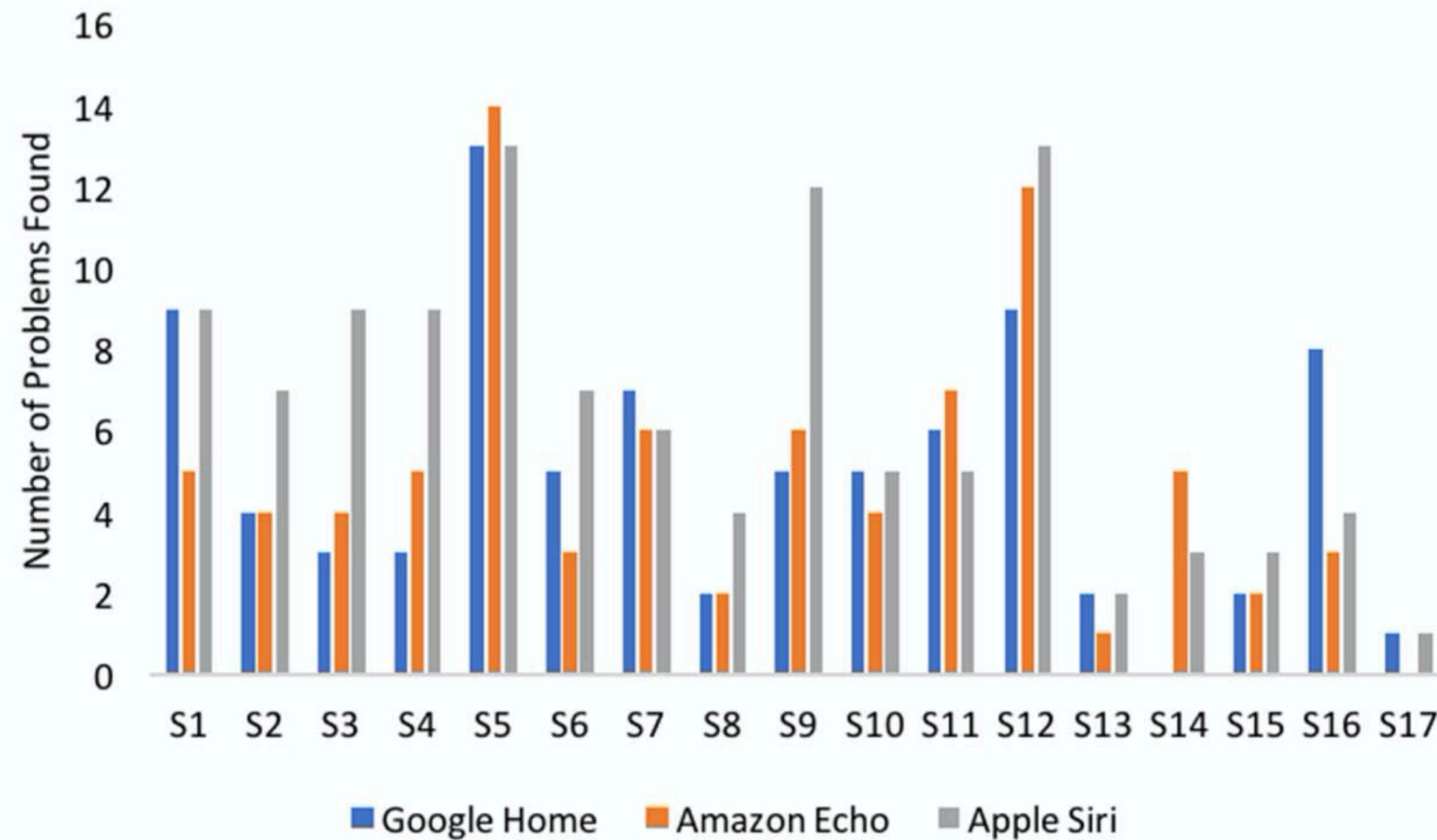
Heuristic #17

S17: Allow users to exit from errors or a mistaken conversation.

— Use a special escape word globally (e.g., "Stop").

— Use non-speech methods when speech fails (e.g., push a physical button).
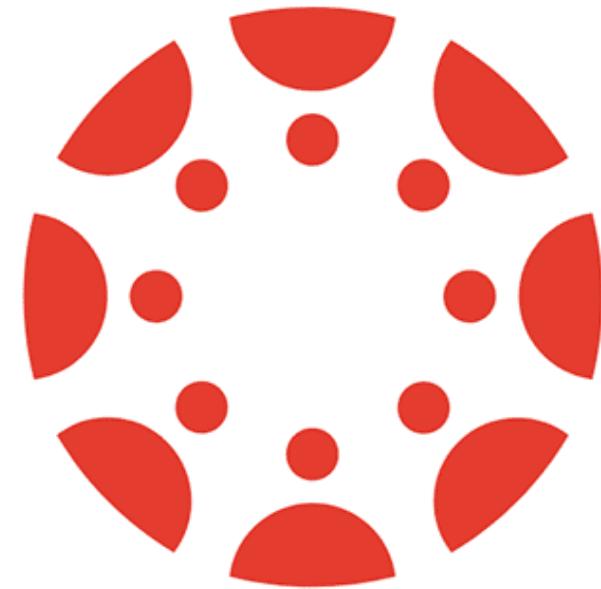
[7] Wei & Landay, 2018, Speech-based Conversational Agent Heuristics

[7] Wei & Landay, 2018, Speech-based Conversational Agent Heuristics

Quiz 3

Complete the Canvas quiz.

What did we learn today?

— Design principles for conversational interfaces

— Usability heuristics for conversational interfaces