
ORACLE®

Oracle Intelligent Bots Advanced Training

Guidelines for building a positive bots UX

ORACLE

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Topic Agenda

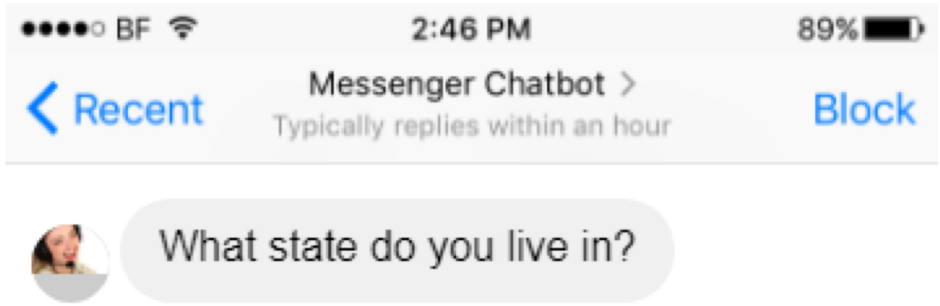
1. Chatbot personality
2. Guiding the user
3. When things go wrong
4. General good tips

A bit of fun before we start

If you talk to a man in a language he understands, that goes to his head. If you talk to him in his language, that goes to his heart.

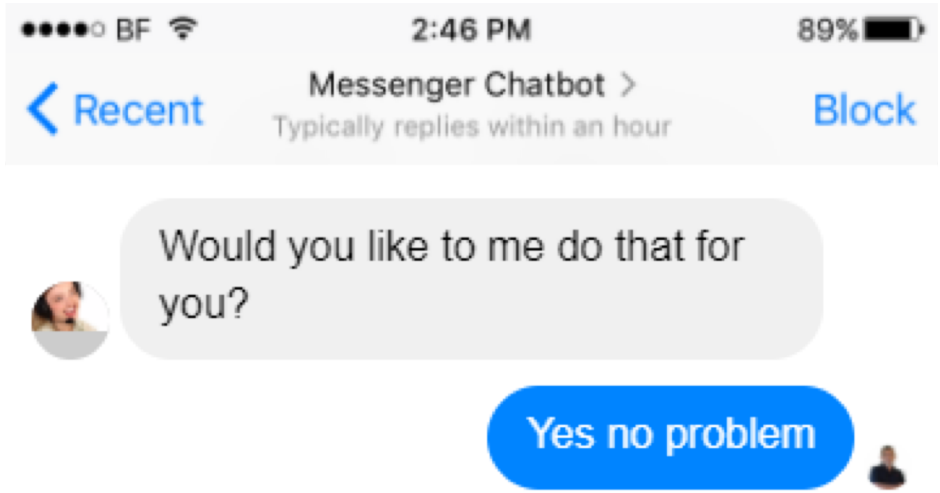
— Nelson Mandela

Why language can be difficult



Why language can be difficult

Why language can be difficult



There is more to meaning than words

There is more to meaning than words

Rollercoaster of emotions

There is more to meaning than words

He bottled up his anger

There is more to meaning than words

It’s a piece of cake

But now you have to be careful of translations

There is more to meaning than words

Kindergeburtstag

Words can be confusing

Words can be confusing

"Do you want to come to the movies"

Words can be confusing

"Don't you want to come to the movies"

Words have emotions

Word have emotions

Constructive criticism -> feedback

Word have emotions

I can’t complain-> Things are great thanks

Word have emotions

You neglected to supply> If you could tell us..

Ok, enough of the fun, you get the idea

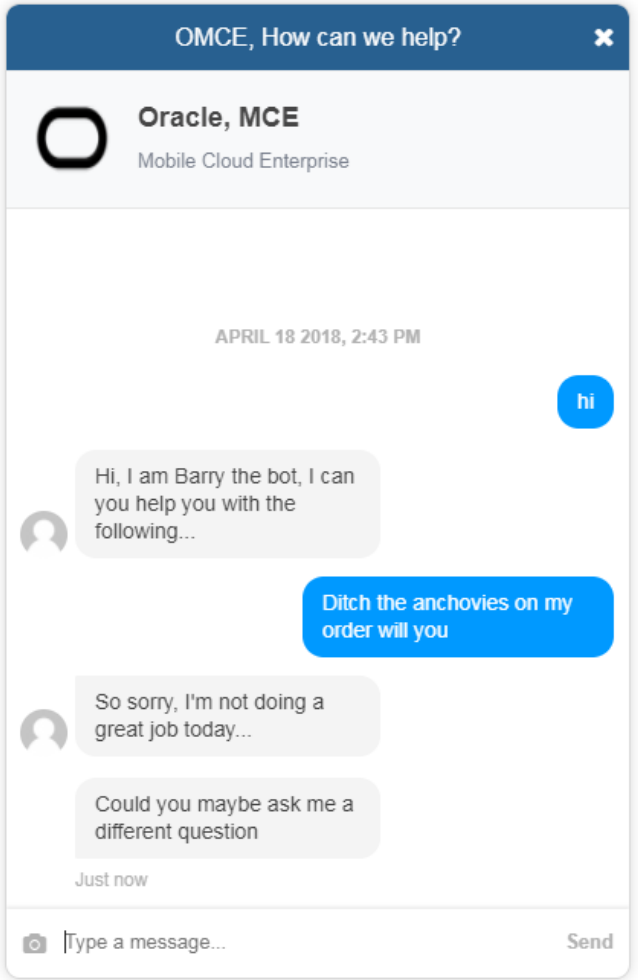
Topic Agenda

1. Chatbot personality
2. Guiding the user
3. When things go wrong
4. General good tips

It’s a bot, not a human, but give it personality

It’s a bot not a human, but give it personality

- Don’t try to pass of the bot as a human
  - If users know it’s a bot, expectations are set as such
  - Consider a name and avatar in line with your brand
    - But possibly not a human avatar
  - Anthropomorphism - human attributes without trying to be human
- Give a positive and welcoming introduction
  - Explain you are a bot, explain what services the bot offers *
  - Use the user’s name
- Handle failure gracefully*

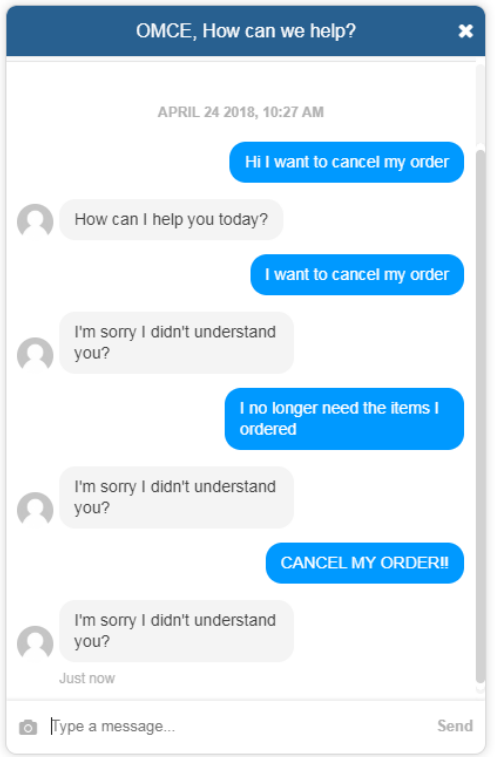


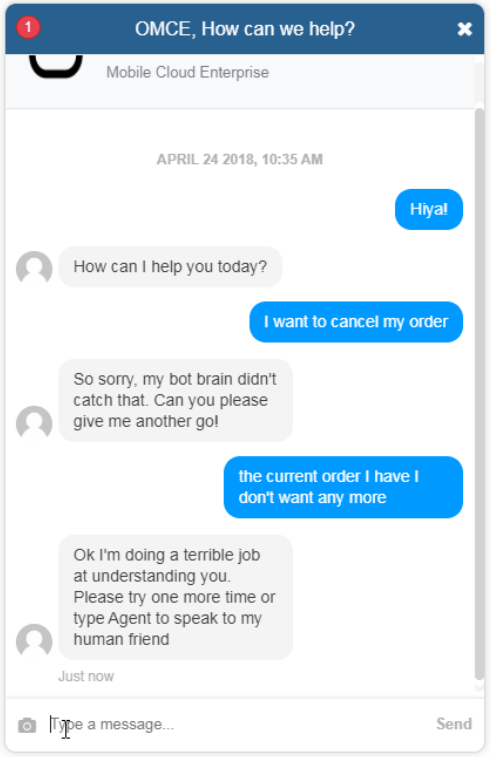
* More on this later

Vary bot responses

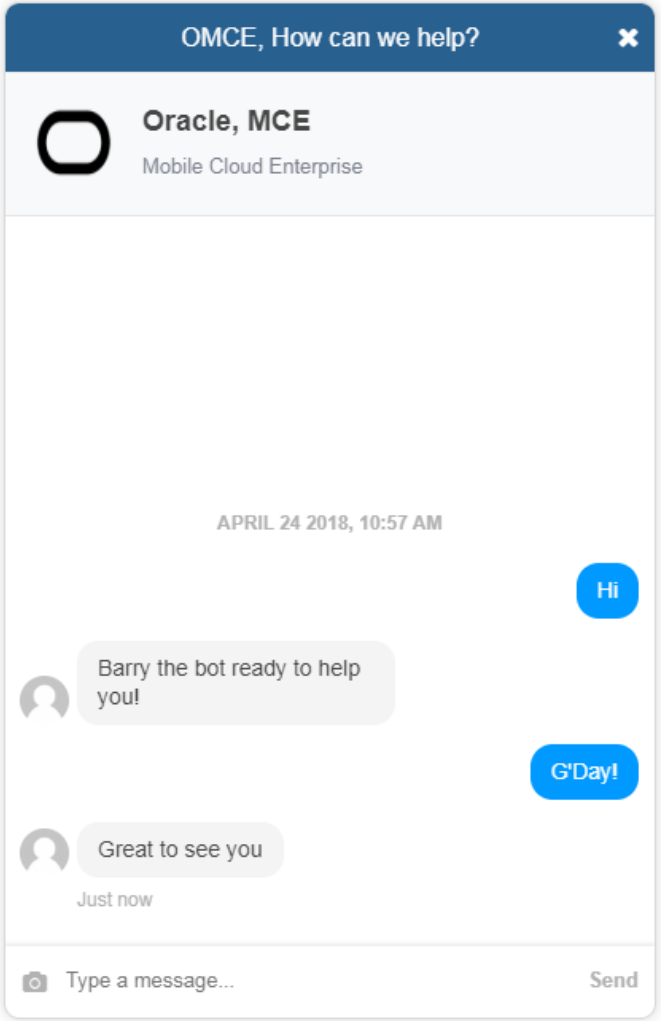
Vary the bot responses

- Avoid repetition and circuitous replies
  - No one is going to warm to the same "*I don't understand*" message
- Randomize greetings, goodbye, error and entity slotting messages
  - After all isn't that what a conversation is really like?
- You could possibly vary answers based on sentiment
  - "Great to hear that" vs "Sorry to hear that"





Vary the bot responses



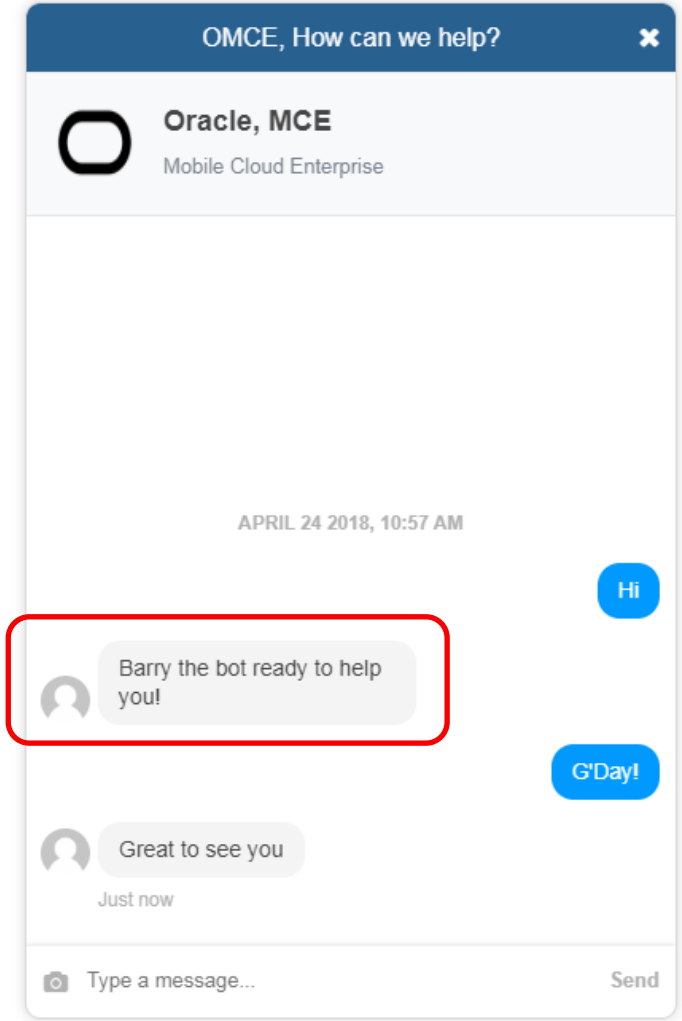

```
states:
  setArray:
    component: "System.SetVariable"
    properties:
      variable: "messages"
      value:
        - welcome: "Hey, what's up"
          goodbye: "See you later"
          confused: "I'm sorry I didn't understand"
        - welcome: "Hi there, ready to help!"
          goodbye: "Thanks for trying me out"
          confused: "Oops, I'm only a bot and I didn't quite get that"
        - welcome: "How can I help you"
          goodbye: "Thanks, see you soon"
          confused: "Sorry today I really suck!"
        - welcome: "Barry the bot ready to help you!"
          goodbye: "Bye!"
          confused: "Sorry I'm not doing a great job today"


# Generate a random number
  generateRandom:
    component: "System.SetVariable"
    properties:
      variable: "idvar"
      value: "${.now?long?string}"
  setRandomNumber:
    component: "System.SetVariable"
    properties:
      variable: "idxs"
      value: "${idvar.value?substring(idvar?length-1, idvar?length)}"

#Say a Random Welcome Message
  sayAWelcomeMessage:
    component: "System.Output"
    properties:
      text: "${messages.value[idxs.value?number].welcome}"
      keepTurn: true

#Say a Random Dont Understand Message
  sayAUnresolvedMessage:
    component: "System.Output"
    properties:
      text: "${messages.value[idxs.value?number].confused}"
      keepTurn: true
```

Vary the bot responses




```
states:
  setArray:
    component: "System.SetVariable"
    properties:
      variable: "messages"
      value:
        - welcome: "Hey, what's up"
          goodbye: "See you later"
          confused: "I'm sorry I didn't understand"
        - welcome: "Hi there, ready to help!"
          goodbye: "Thanks for trying me out"
          confused: "Oops, I'm only a bot and I didn't quite get that"
        - welcome: "How can I help you"
          goodbye: "Thanks, see you soon"
          confused: "Sorry today I really suck!"
        - welcome: "Barry the bot ready to help you!"
          goodbye: "Bye!"
          confused: "Sorry I'm not doing a great job today"

# Generate a random number
  generateRandom:
    component: "System.SetVariable"
    properties:
      variable: "idvar"
      value: "${.now?long?string}"
  setRandomNumber:
    component: "System.SetVariable"
    properties:
      variable: "idxs"
      value: "${idvar.value?substring(idvar?length-1, idvar?length)}"

#Say a Random Welcome Message
  sayAWelcomeMessage:
    component: "System.Output"
    properties:
      text: "${messages.value[idxs.value?number].welcome}"
      keepTurn: true

#Say a Random Dont Understand Message
  sayAUnresolvedMessage:
    component: "System.Output"
    properties:
      text: "${messages.value[idxs.value?number].confused}"
      keepTurn: true
```

Handle smalltalk gracefully

Handle smalltalk gracefully

- You WILL have users asking about
  - The weather, jokes, are you human, what is happening the world
- Prepare to handle smalltalk
  - Either with a generic reply "*Sorry, that kind of stuff is not really in my job description*"
  - Or specifically
    - Link to a joke of the day web service, weather services, news etc
- Every step in the conversation is a chance to reinforce the bot personality
- A chance to manage the conversation BACK to the known use cases
  - "*Ok, so that's the best joke I've got, how about I get back to helping with your order*"

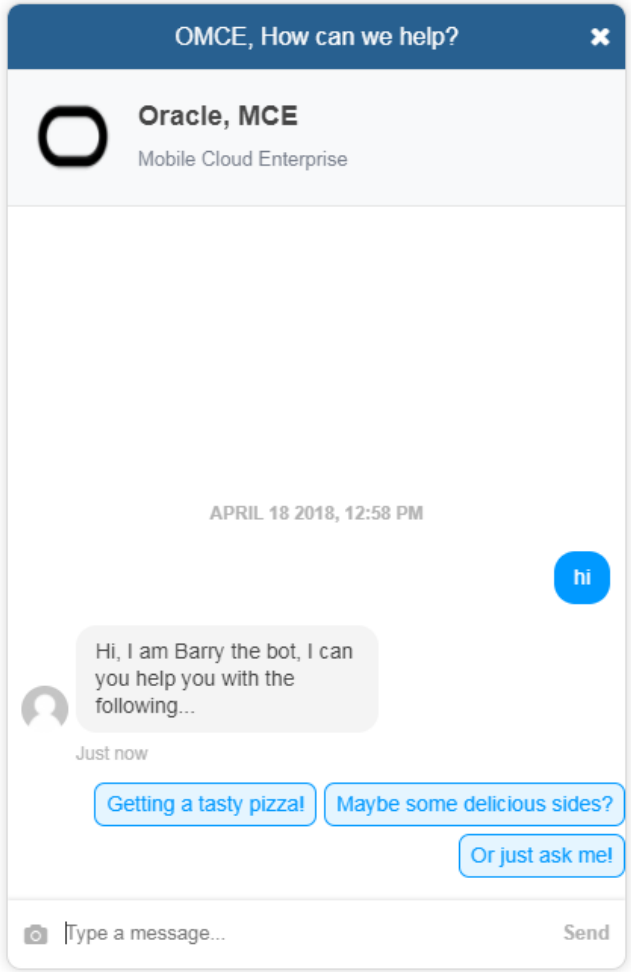
Topic Agenda

1. Chatbot personality
2. Guiding the user
3. When things go wrong
4. General good tips

Give guidance from the start

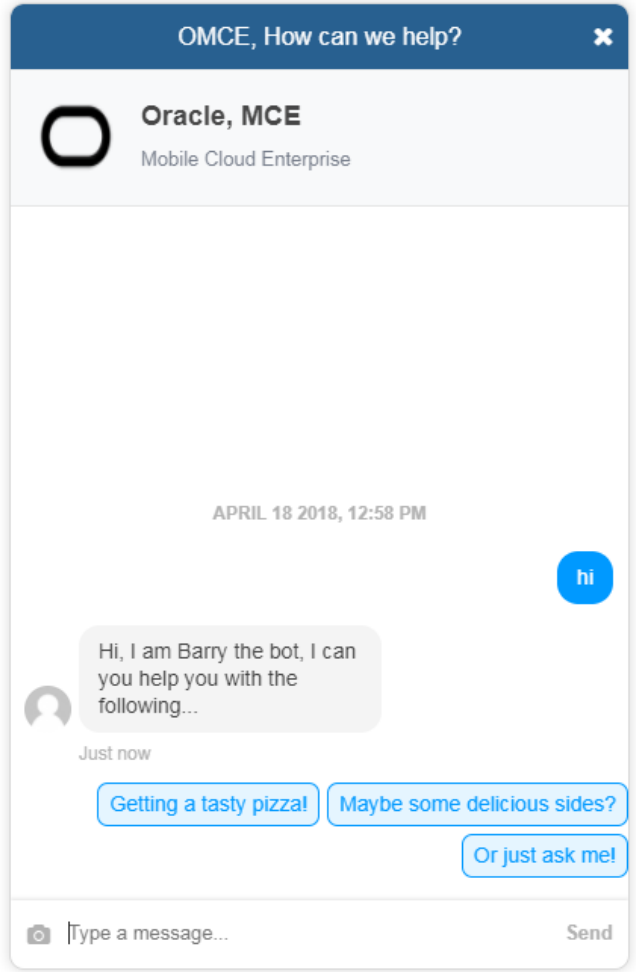
Give guidance from the start

- Not always clear how to start a conversation
- The bot should give guidance on what services it can support
- Use “quick replies” to indicate the most common options
- Remind and offer some sort of “help” or “reset” option



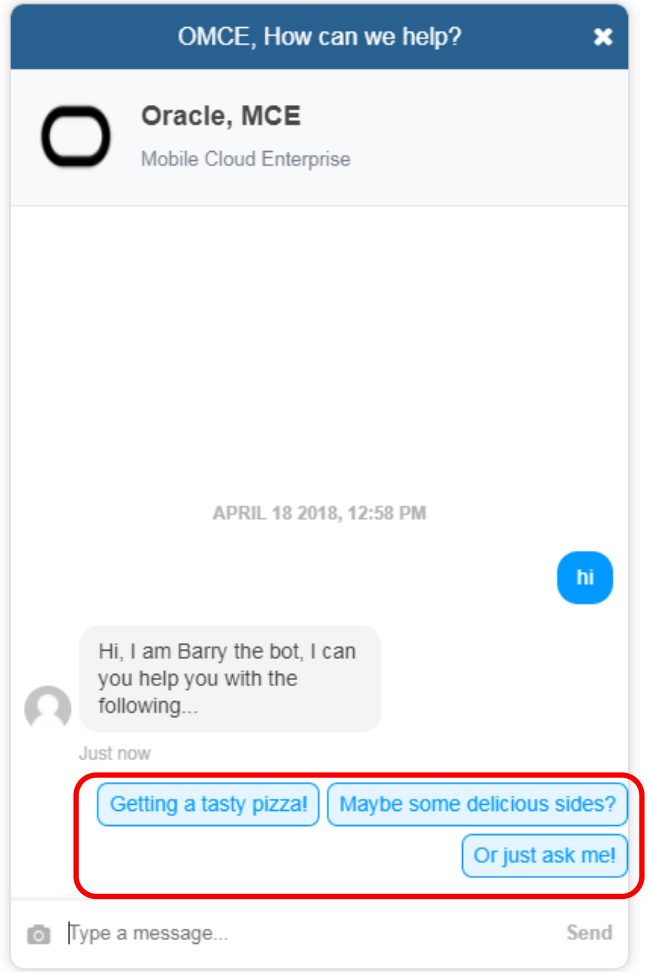
Give guidance from the start


```
WelcomeCRC:
  component: "System.CommonResponse"
  properties:
    processUserMessage: true
    keepTurn: false
    metadata:
      responseItems:
      - type: "text"
        text: ""
      globalActions:
      - label: "Getting a tasty pizza!"
        type: "postback"
        keyword: "pizza"
        payload:
          variables:
            openingQ: "pizza"
      - label: "Maybe some delicious sides?"
        type: "postback"
        keyword: "sides"
        payload:
          variables:
             openingQ: "sides"
      - label: "Or just ask me!"
        type: "postback"
        keyword: "something else"
        payload:
          variables:
            openingQ: "SomethingElse"
  transitions: {}
```



Give guidance from the start


```
WelcomeCRC:
  component: "System.CommonResponse"
  properties:
    processUserMessage: true
    keepTurn: false
    metadata:
      responseItems:
      - type: "text"
        text: ""
      globalActions:
      - label: "Getting a tasty pizza!"
        type: "postback"
        keyword: "pizza"
        payload:
          variables:
            openingQ: "pizza"
      - label: "Maybe some delicious sides?"
        type: "postback"
        keyword: "sides"
        payload:
          variables:
            openingQ: "sides"
      - label: "Or just ask me!"
        type: "postback"
        keyword: "something else"
        payload:
          variables:
            openingQ: "SomethingElse"
  transitions: {}
```



Bring the user back to the happy path

Bring the user back to the happy path

- When things are going wrong, offer the user a way back to the happy path
- State what you do know then drive the user through
  - "Ok so you want to book a holiday, let's first of all check what time of year"

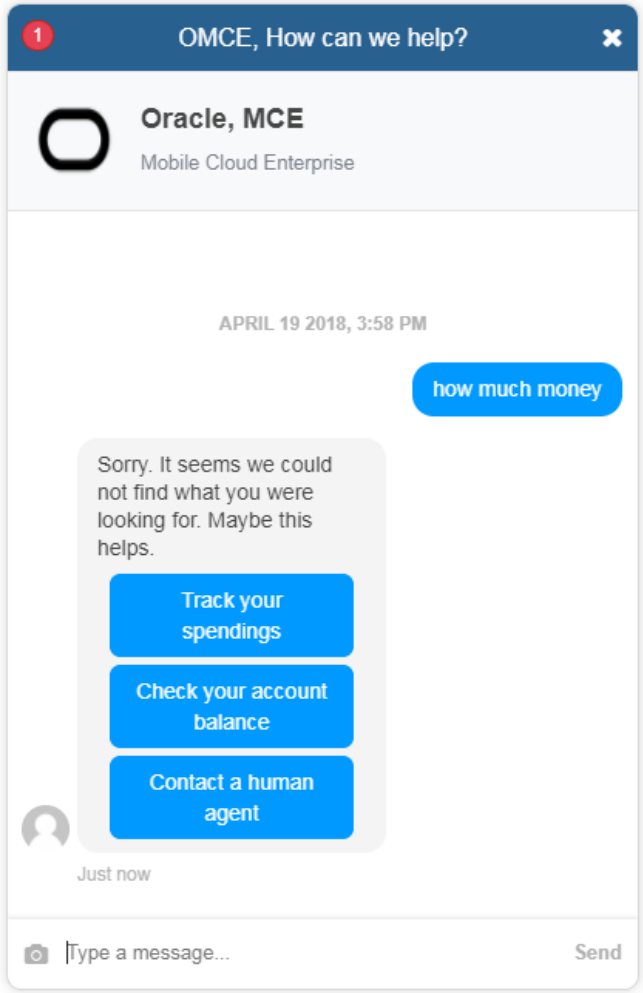
Topic Agenda

1. Chatbot personality
2. Guiding the user
3. When things go wrong
4. General good tips

When the bot fails, fail gracefully and give options to the user

When the bot fails, fail gracefully and give options

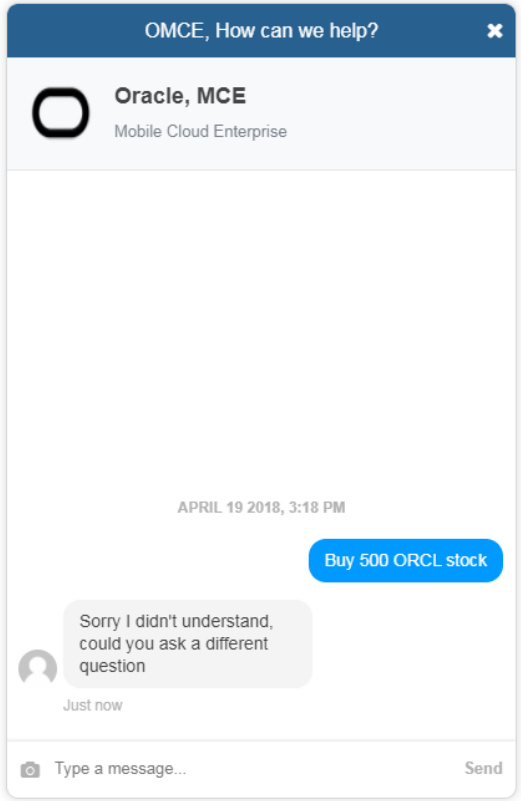
- Fail gracefully
  - Don't blame the user
    - "You didn't input a valid option" vs "Sorry I didn't understand"
  - An opportunity to re-enforce the bot personality
    - "Whoa, you lost me there! Can you ask again, but go easy on me!"
- Be careful with language
  - "*That order doesn't exist*" vs "*I couldn't find that order*"
- Give the user intelligent options to get back on track
  - Show the nearest resolving intents
  - Give an option to reset the conversation
  - Human agent hand-off

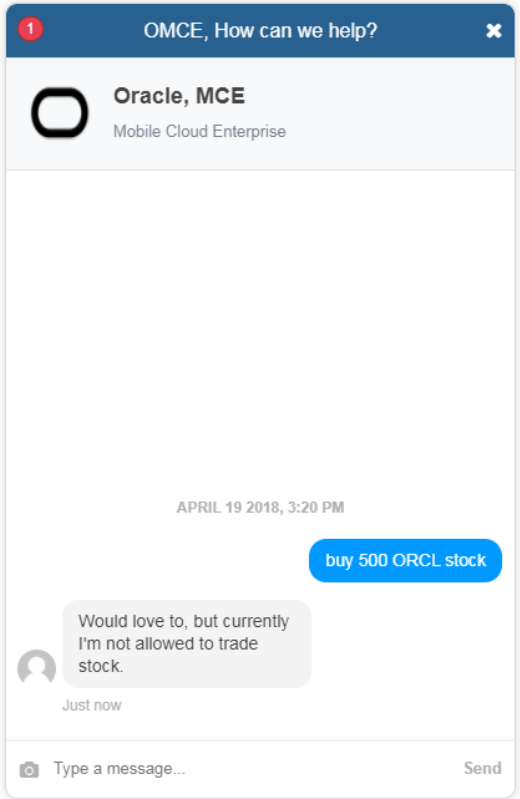


Handle the things you know you don’t know

Handle the things you know you don’t know

- Create intents for the use cases you know you can’t handle
  - Your bot likely handles a subset of business functions
  - Gracefully handle the business function NOT supported by the bot
- Allows the bot to more precisely handle failure
  - User knows input was understood
  - Prevents redundant retries

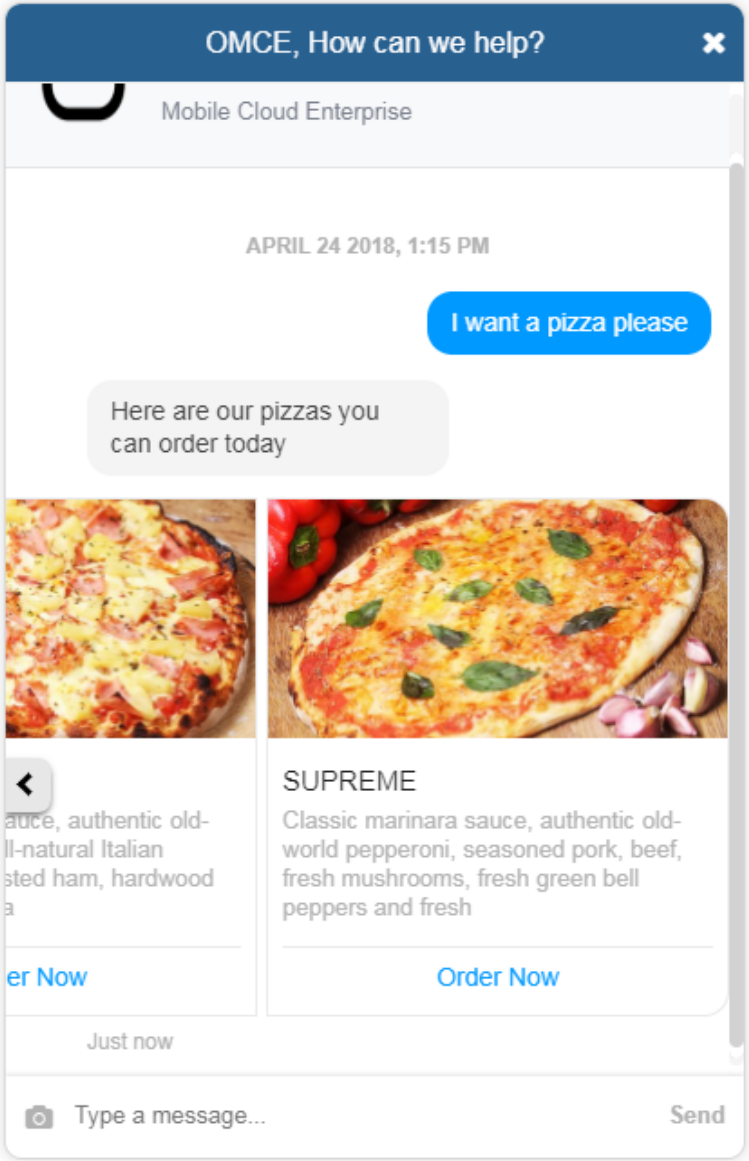




Limit the “surface area” for errors

Limit the surface area for errors

- The bot should try and guide the conversation back to the use case in hand
- Avoid open ended questions
  - *"What kind of pizza do you want"* vs *"Do you fancy meaty, spicy or veggie-licious?"*
- Use quick replies, cards and carousels to select
  - Easier to read
  - Quicker and more accurate to select, especially when mobile
- Ask for confirmation before committing a transaction



Topic Agenda

1. Chatbot personality
2. Guiding the user
3. When things go wrong
4. General good tips

General good tips

Subtitle

Context is king

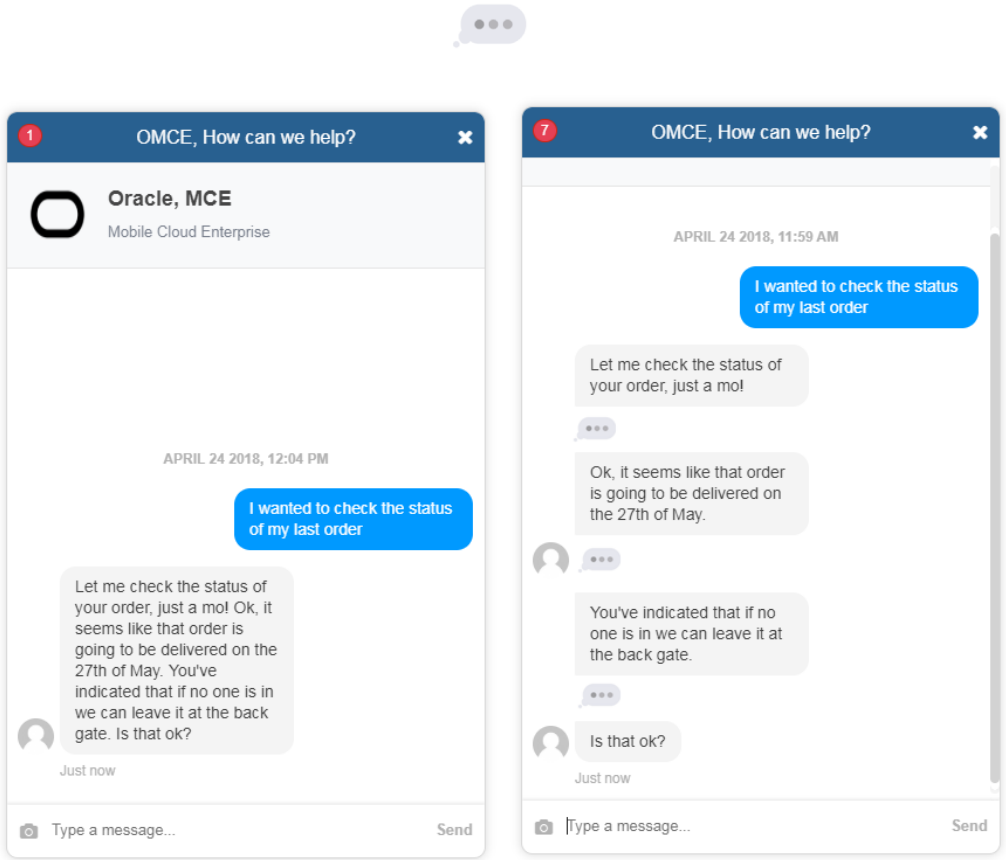
Context is king

- If you re-engage with a bot it should remember that context
  - Make the conversation more relevant
    - *"Shall I deliver to your office like last time?"*
- Offer options based on where the conversation currently is
  - Offer dips when ordering pizza, not when submitting your payment details
  - *"What's the weather like in London"* then *"what about this weekend"*
- Active listening – repeating back key points
  - *"Sorry to hear you want to cancel your membership"*
  - *"So you are looking for a skiing holiday in April in Europe, let's see what we can do!"*

Make bot responses as easy and quick to digest as possible

Make bot responses easily digestible

- Try to keep responses short and avoid tl;dr (too long; didn't read)
- Consider multiple responses
  - But don't split a response at an arbitrary point
- Use typing indicator to "queue up" responses from the bot
- Quick replies, card and carousels are often quicker to digest than written text



Advanced Bot Training Hands-On

Lab 8a and Lab 8b

Integrated Cloud

Applications & Platform Services


ORACLE®