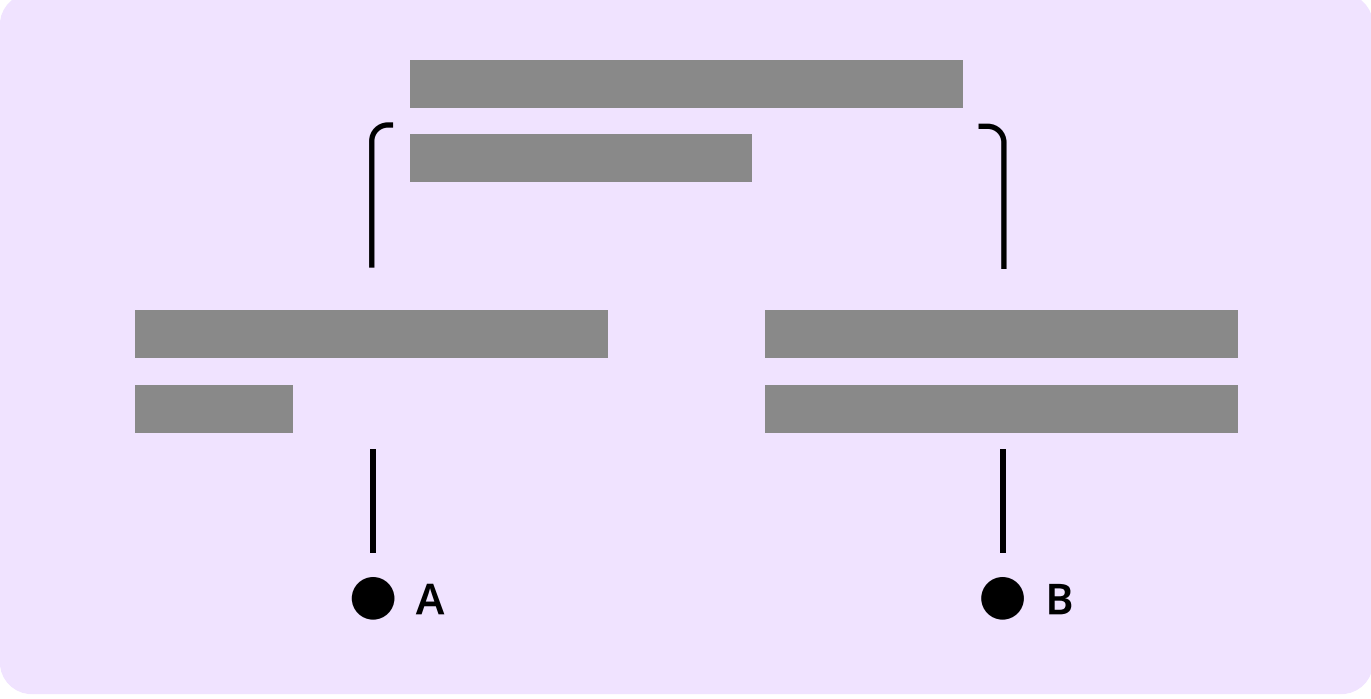


INTENT-BASED CONVERSATIONS

Defining clear conversational paths



Overview

Intent-based conversation focuses on understanding and responding to user intents. The system identifies what the user wants to achieve, such as designing a form or creating a workflow, and provides relevant responses or actions. This pattern enhances user experience and productivity by offering contextually relevant responses.

Common Issues

Issues with copilot intents often stem from the ambiguity in user input. Users tend to phrase their requests in various ways, creating ambiguities that simple keyword-based systems might not resolve accurately. Another significant problem is the lack of context; without a proper understanding of context, the copilot's responses can be generic and unhelpful, often forcing users to repeat or rephrase their queries multiple times. Additionally, managing complex tasks is challenging for copilot systems. Multi-step tasks or requests involving several related actions require a clear understanding of user intent, which can be difficult to achieve. This series of problems often culminates in user frustration, as receiving repeated irrelevant or incorrect responses decreases user satisfaction with the platform.

Solution

Design the copilot to recognise specific intents relevant to Nintex, such as generating a document or creating an app, to provide accurate and relevant responses.

Example 1 - Creating a workflow

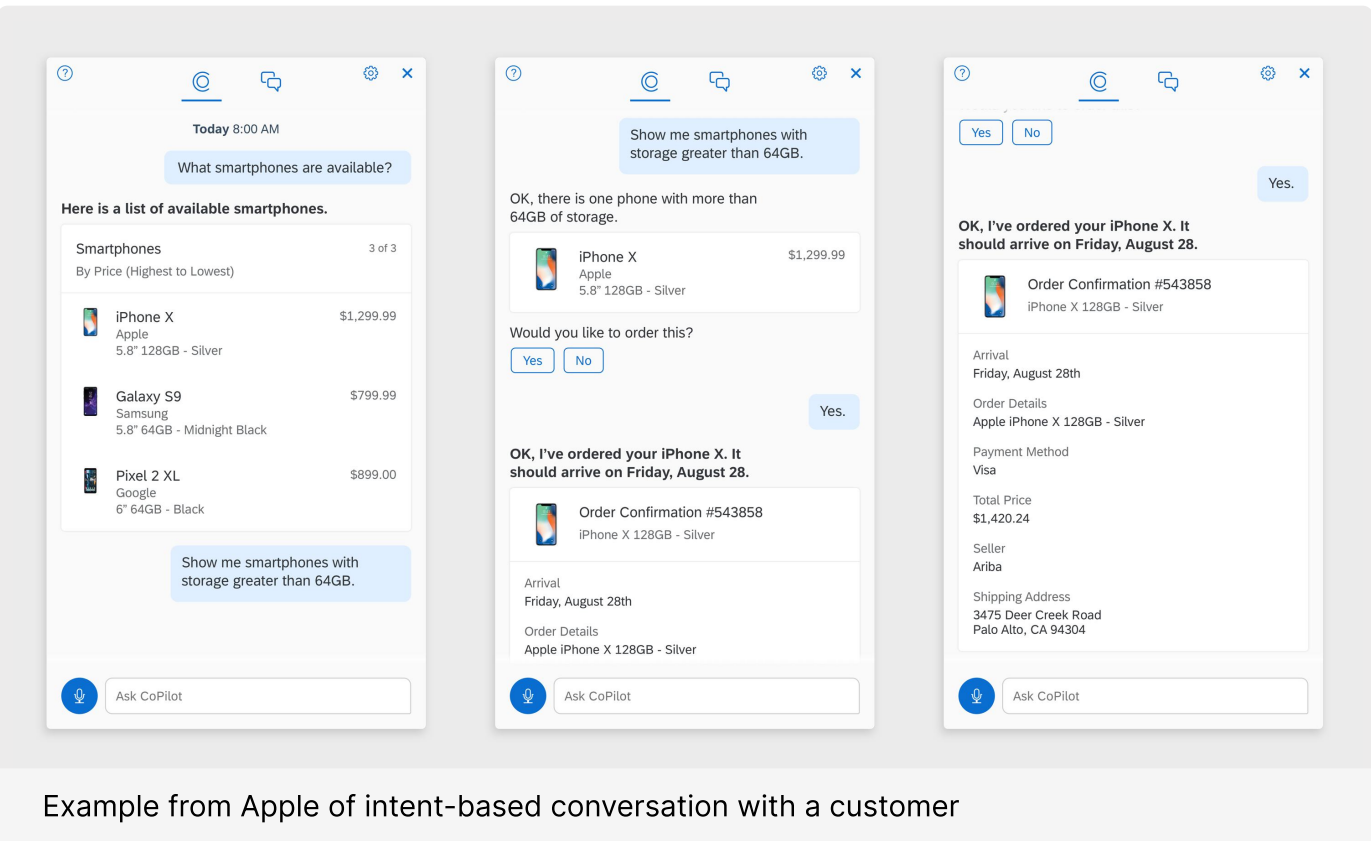
Intent
User wants to create a workflow.

Interaction flow
User: "I need to create a new workflow."
Copilot: "Sure, would you like to start with a template or create from scratch?"
User: "Create from scratch."
Copilot: "Great, what would you like to name this workflow?"
User: "Sales Process Automation."
Copilot: "Excellent. What are the key steps involved in your sales process?"
...

Example 2 - Generating a document

Intent
User wants to generate a document

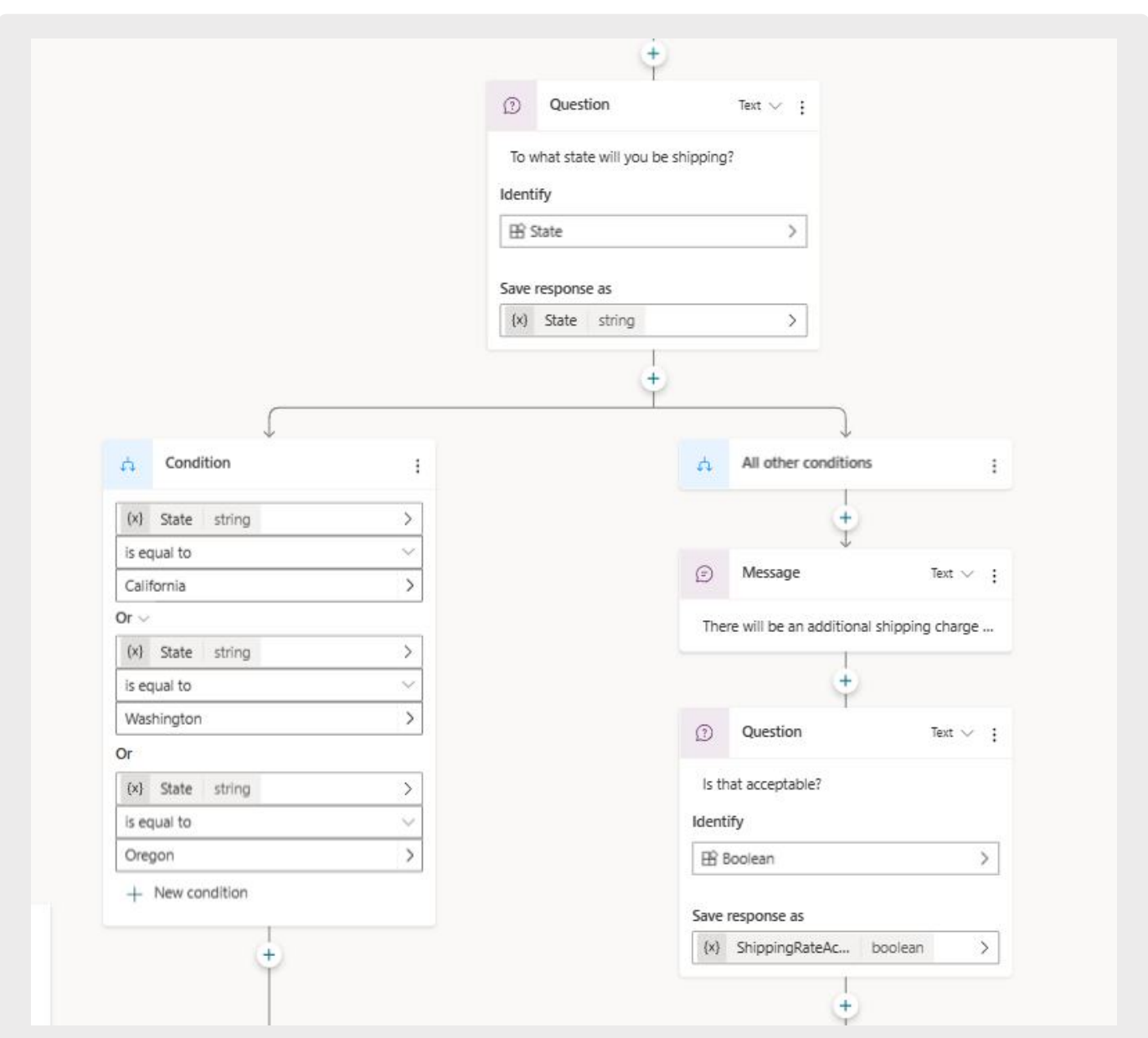
Interaction flow
User: "I need to generate a contract."
Copilot: "Absolutely, do you have a template you'd like to use, or should we create a new one?"
User: "Use the standard contract template."
Copilot: "Got it. Please provide the client's name and the contract start date."
User: "Client is ABC Corp, and the start date is June 1st."
Copilot: "Perfect. Your contract for ABC Corp starting on June 1st is being generated now."



Example from Apple of intent-based conversation with a customer

Key elements

- Intent recognition: Use NLP to accurately interpret user inputs and identify goals
- Response Generation: Provide clear, concise, and actionable replies tailored to intents
- Fallback mechanism: Handle unrecognised intents by offering alternative suggestions or rephrasing queries
- Integration: Ensure seamless execution of user intents with Nintex's tools
- User feedback: Continuously improve the system through feedback
- Accessibility: Ensure inclusivity



Example of designing for intent-based conversation

Best practices

- Clear and concise language: Avoid confusion and guide users efficiently
- Provide examples: Help users understand how to phrase their requests
- Continuous improvement: Refine intent recognition and response accuracy through feedback
- Consistency: Maintain a consistent design across all interactions to build user trust

Additional considerations

- Context awareness: Incorporate context from previous interactions to provide more relevant responses
- Proactive assistance: Anticipate user needs and offer proactive suggestions or actions.
- Personalisation: Customise responses based on user preferences and past behaviour to enhance the user experience.