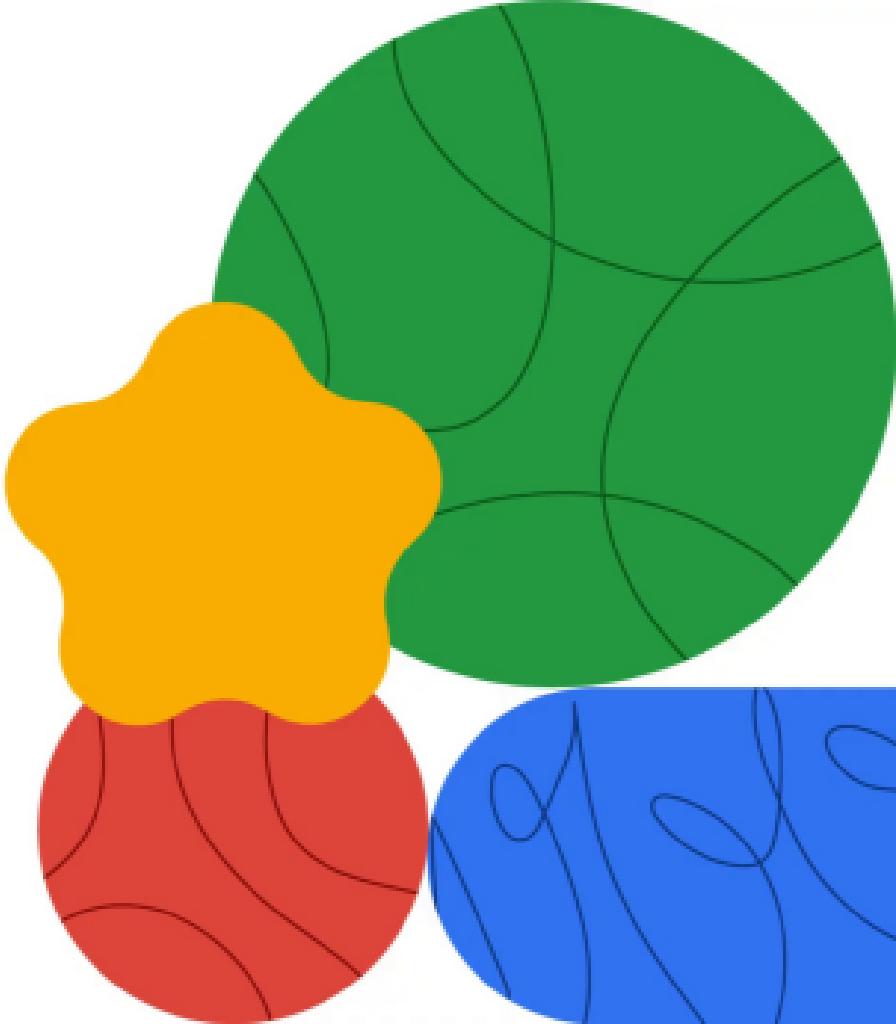
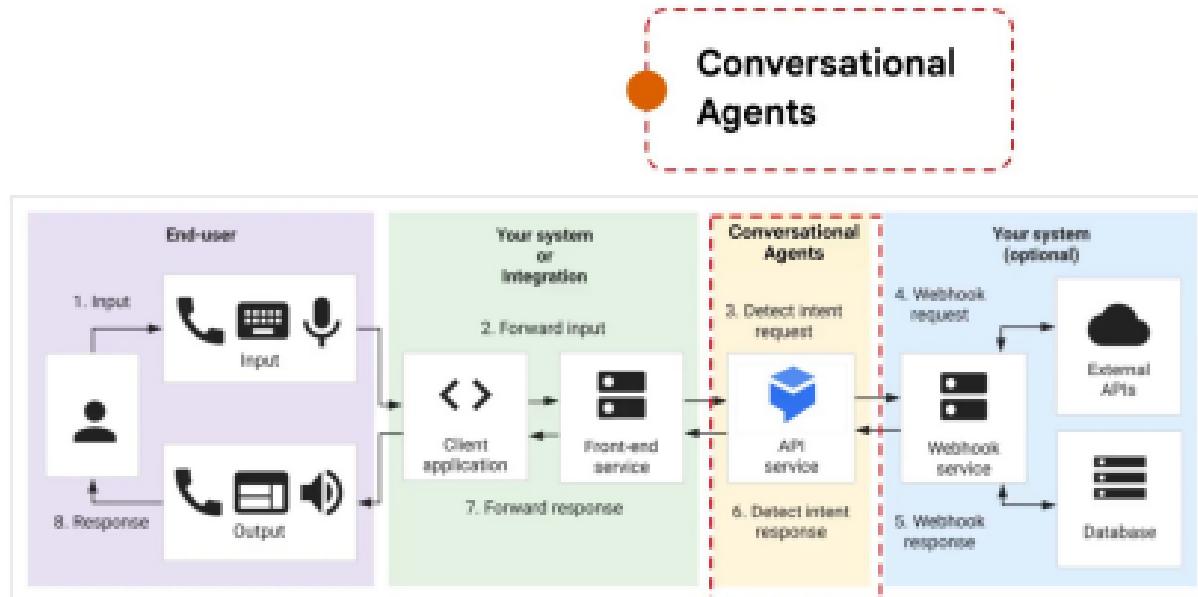


# Methods of testing

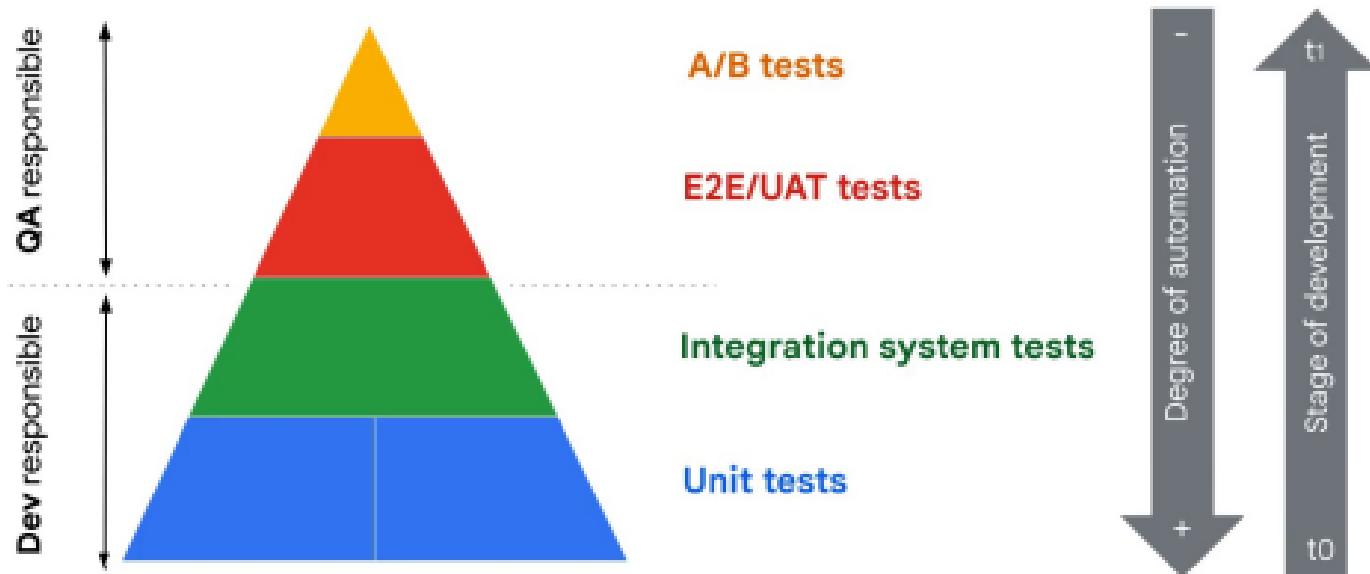


# Simplified architecture of Conversational Agents

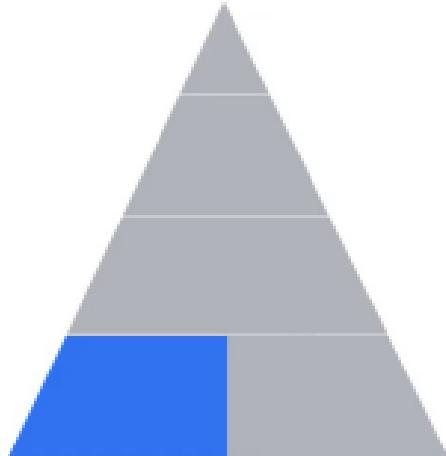


# Developing a testing strategy

The most successful testing strategies for Conversational Agents are based on a four tiered approach.



# Testing strategy: Routing unit tests

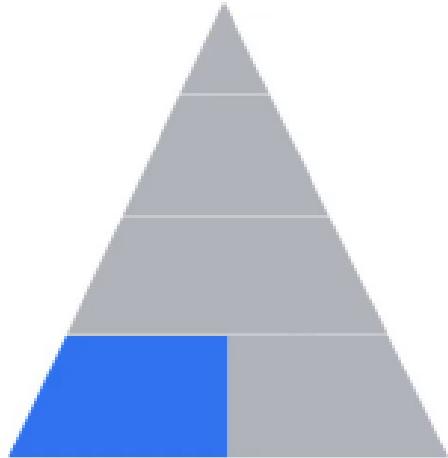


**Owners:** Development team

## **Objectives:**

- Unit testing proves that the logic of a specific unit functions correctly, and provides developers with feedback.
- Unit testing begins in the sandbox environment, testing one module, segment, or application in isolation from other applications or solution components.
- There's an expectancy for happy paths that cover all work done by the dev team.
- Tests may include interactions with other units, in an application, or limited testing with other systems.

# Testing strategy: Routing unit tests

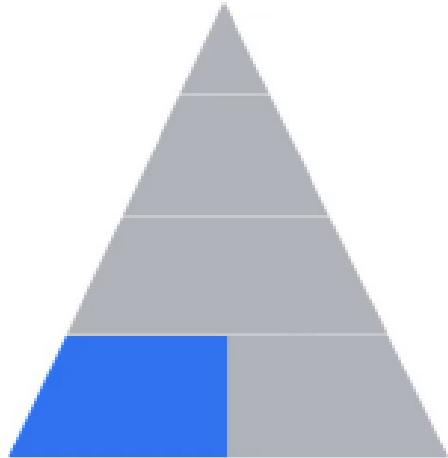


**Owners:** Development team

## **Objectives:**

- Unit testing proves that the logic of a specific unit functions correctly, and provides developers with feedback.
- Unit testing begins in the sandbox environment, testing one module, segment, or application in isolation from other applications or solution components.
- There's an expectancy for happy paths that cover all work done by the dev team.
- Tests may include interactions with other units, in an application, or limited testing with other systems.

# Testing strategy: Routing unit tests

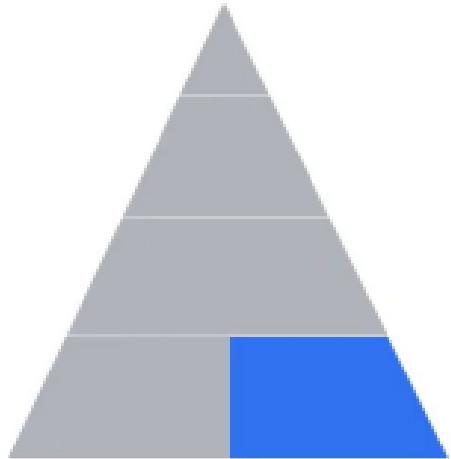


**Owners:** Development team

## **Objectives:**

- Unit testing proves that the logic of a specific unit functions correctly, and provides developers with feedback.
- Unit testing begins in the sandbox environment, testing one module, segment, or application in isolation from other applications or solution components.
- There's an expectancy for happy paths that cover all work done by the dev team.
- Tests may include interactions with other units, in an application, or limited testing with other systems.

# Testing strategy: NLU unit tests

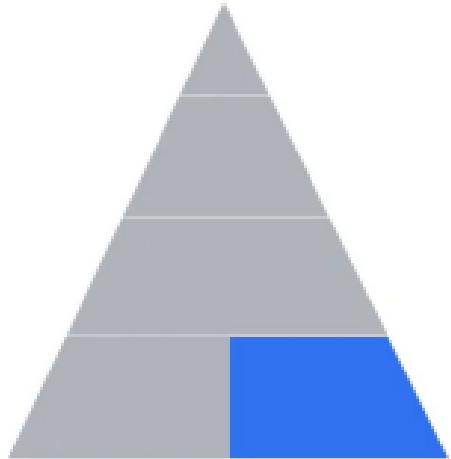


**Owners:** Development team

## **Objectives:**

- Tests ensure that the NLU model correctly identifies the intent of diverse user utterances, including variations in phrasing, synonyms, and informal language.
- Tests validate the NLU model's ability to consistently extract relevant entities (for example, dates, locations, product names) from a wide range of user input.
- Tests leverage the large volume of utterances to identify areas where the NLU model can be fine-tuned, enhancing its overall performance over time.

# Testing strategy: NLU unit tests

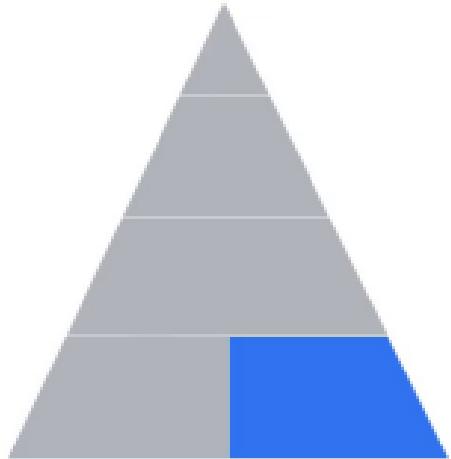


**Owners:** Development team

## **Objectives:**

- Tests ensure that the NLU model correctly identifies the intent of diverse user utterances, including variations in phrasing, synonyms, and informal language.
- Tests validate the NLU model's ability to consistently extract relevant entities (for example, dates, locations, product names) from a wide range of user input.
- Tests leverage the large volume of utterances to identify areas where the NLU model can be fine-tuned, enhancing its overall performance over time.

# Testing strategy: NLU unit tests



**Owners:** Development team

## **Objectives:**

- Tests ensure that the NLU model correctly identifies the intent of diverse user utterances, including variations in phrasing, synonyms, and informal language.
- Tests validate the NLU model's ability to consistently extract relevant entities (for example, dates, locations, product names) from a wide range of user input.
- Tests leverage the large volume of utterances to identify areas where the NLU model can be fine-tuned, enhancing its overall performance over time.

# Testing strategy: NLU unit tests

## Case study: Multiple device selection

Situation

A caller has multiple devices and Conversational Agents needs to specify which device the user is calling about.

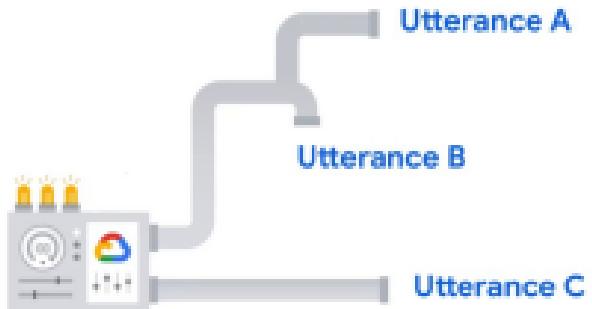
Conversational Agents expects the customer to input the last 4 digits of that given device. However, the caller could also say "I don't know," "None of them," or "All of them."

Consider

Develop intents to reduce agent escalation. Direct non-recognized Conversational Agents-related utterances to training datasets for the NLU algorithm.

Test

Agent enhancements, both new and regression utterances, should be tested to ensure recognition throughout the interactive voice experience for any given iteration.



# Testing strategy: NLU unit tests

## Case study: Multiple device selection

Situation

A caller has multiple devices and Conversational Agents needs to specify which device the user is calling about.

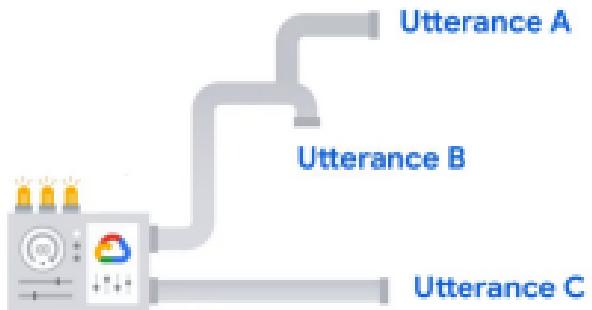
Conversational Agents expects the customer to input the last 4 digits of that given device. However, the caller could also say "I don't know," "None of them," or "All of them."

Consider

Develop intents to reduce agent escalation. Direct non-recognized Conversational Agents-related utterances to training datasets for the NLU algorithm.

Test

Agent enhancements, both new and regression utterances, should be tested to ensure recognition throughout the interactive voice experience for any given iteration.



# Testing strategy: NLU unit tests

**Pass**

# of passed test cases

100

**Failed**

# of failed test cases

0

# Testing strategy: NLU unit tests

**Pass**

# of passed test cases

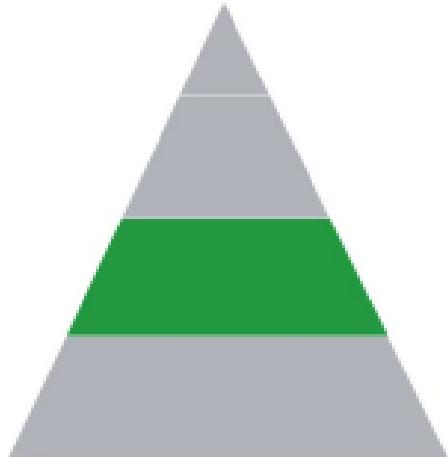
100

**Failed**

# of failed test cases

0

# Testing strategy: Integration system tests

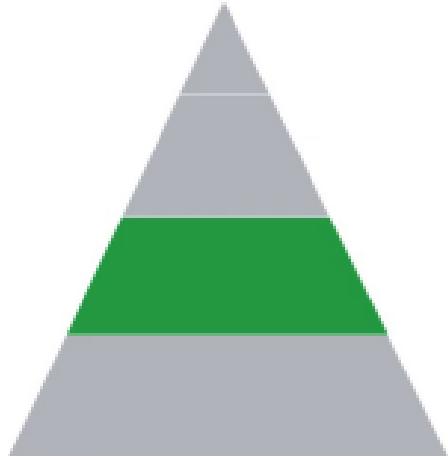


**Owners:** Developers

## **Objectives:**

- Integrated System Testing (IST) discovers errors in the interfaces between components and verifies that the modules work together correctly as one function.
- Developing in an integrated environment significantly eases IST testing.
- IST testing should be done by developers where possible, especially for functionalities where there's inconsistencies between Conversational Agents and the customer's environment.

# Testing strategy: Integration system tests

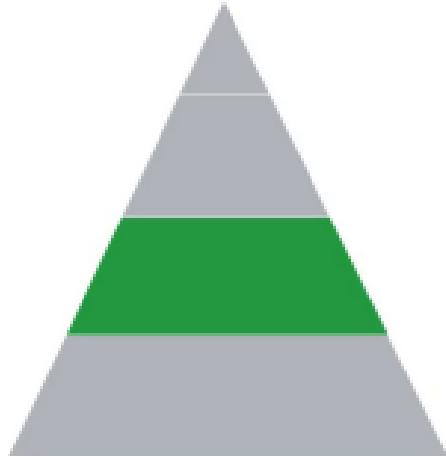


**Owners:** Developers

## **Objectives:**

- Integrated System Testing (IST) discovers errors in the interfaces between components and verifies that the modules work together correctly as one function.
- Developing in an integrated environment significantly eases IST testing.
- IST testing should be done by developers where possible, especially for functionalities where there's inconsistencies between Conversational Agents and the customer's environment.

# Testing strategy: Integration system tests

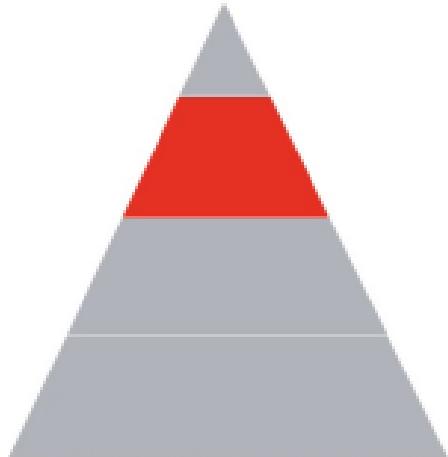


**Owners:** Developers

## **Objectives:**

- Integrated System Testing (IST) discovers errors in the interfaces between components and verifies that the modules work together correctly as one function.
- Developing in an integrated environment significantly eases IST testing.
- IST testing should be done by developers where possible, especially for functionalities where there's inconsistencies between Conversational Agents and the customer's environment.

# Testing strategy: End to End (E2E) user acceptance testing (UAT)

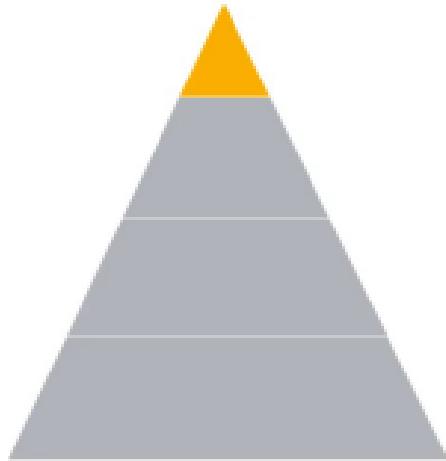


**Owners:** Quality Assurance (QA)

## **Objectives:**

- UAT allows stakeholders to certify and ensure the bot can accommodate and react to real world scenarios.
- The UAT phase must ensure both correct E2E functionality (both from an integration and data integrity standpoint) as well as a satisfactory customer experience.
- Developing in an integrated environment eases UAT/E2E testing. If this is not feasible, an alternative is to perform UAT tests in a Conversational Agents sandbox.

# Testing strategy: A/B tests

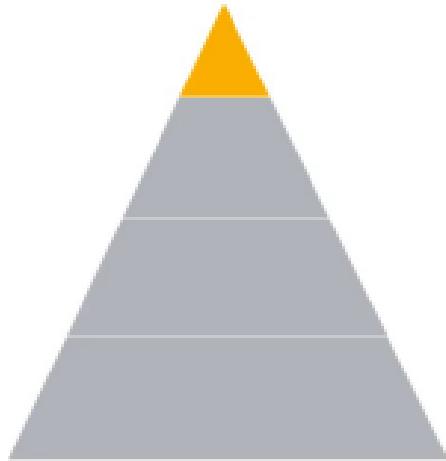


**Owners:** Quality Assurance (QA)

**Objectives:**

- A/B Testing is the final phase of the software certification process prior to exposure to live customers.
- It is a UX methodology that involves in a randomized experiment to compare the performance of two systems or conversational experiences. It verifies which versions better meets the functional requirements outlined in the project scope or specifications.

# Testing strategy: A/B tests



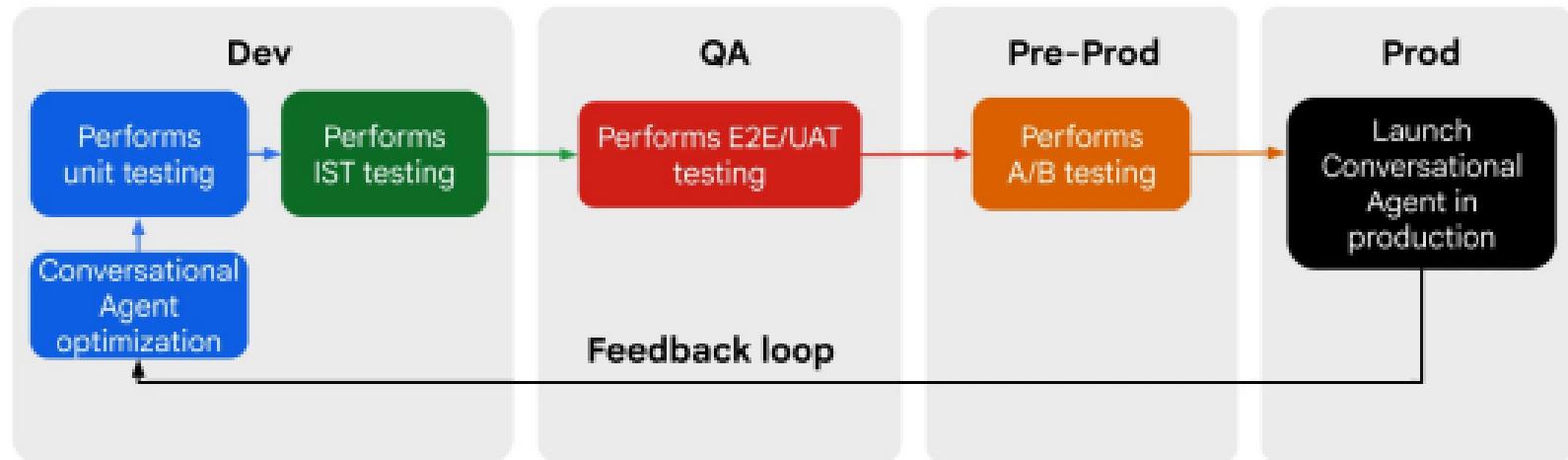
**Owners:** Quality Assurance (QA)

**Objectives:**

- A/B Testing is the final phase of the software certification process prior to exposure to live customers.
- It is a UX methodology that involves in a randomized experiment to compare the performance of two systems or conversational experiences. It verifies which versions better meets the functional requirements outlined in the project scope or specifications.

# Testing setup

The following flow is based on the assumption that all environments are in the customer's project and Dev, QA, and Pre-Prod environments are configured like the Prod environment.



# Additional testing strategies to consider

Activities	Objective	Test execution	Environment
Design testing (usability, accessibility, etc.)	<ul style="list-style-type: none"><li>Intent classification and entity recognition.</li><li>Conversational designs with specific, highlighted scenarios.</li><li>User experience/Usability tests.</li></ul>	Conversation design team	Dev
Performance testing	<ul style="list-style-type: none"><li>Assess the performance of the Conversational Agent and other infrastructure components under varying loads. This includes testing response times, accuracy under load, and the system's ability to handle multiple simultaneous interactions.</li></ul>	Telecom Telephony	QA / Pre-Prod
Regression testing	<ul style="list-style-type: none"><li>Retest previously delivered features to ensure the solution still works as expected after new code is introduced.</li></ul>	QA team	QA / Pre-Prod
Adversarial testing	<ul style="list-style-type: none"><li>Deliberately exposes the agent to confusing, malicious, or unexpected inputs in order to uncover vulnerabilities and areas for improvement.</li></ul>	QA team	QA / Pre-Prod / Prod
Failover and recovery	<ul style="list-style-type: none"><li>Test how the conversational agent handles system failures, how it recovers, and how it maintains data integrity in such events.</li></ul>	QA team	QA / Pre-Prod
Beta testing	<ul style="list-style-type: none"><li>Launch the completed Product into production and collect data from actual users to ensure that the Product is working as expected prior to go-live.</li></ul>	Subset of member-clients	Prod

# Tracking the results of testing

Test cases > [View all](#) [Report](#) [Import](#) [Coverage](#)

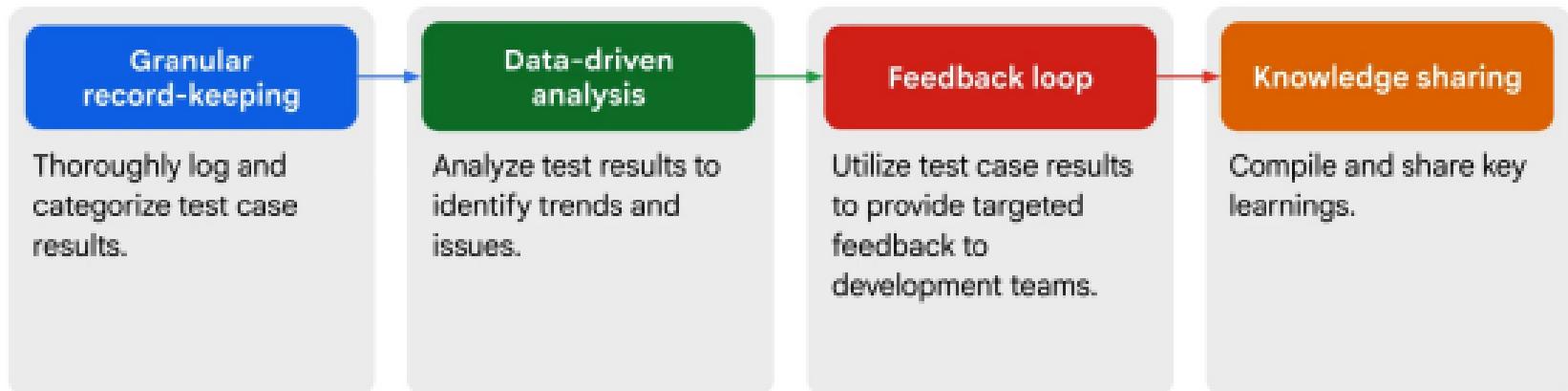
[S](#) [Publish](#) [Agent settings](#) [Test Agent](#)

Pass	Fail
# of passed test cases	# of failed test cases
13	10

[Filter](#) Filter test cases

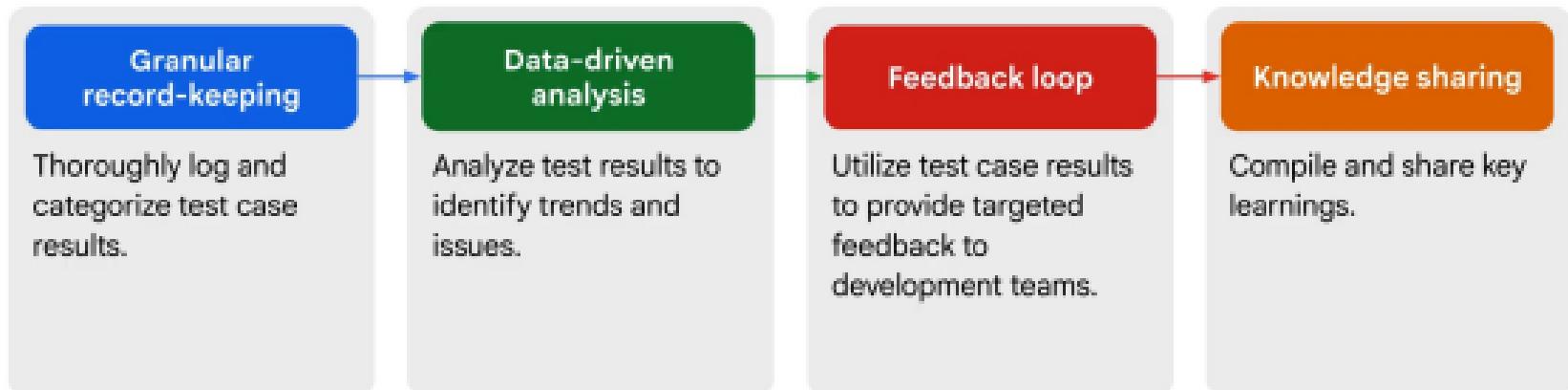
<input type="checkbox"/> Display Name	Tags	Tested	Result
<input type="checkbox"/> Happy Path - Add Plan Option 2	-	Feb 12, 2024 10:09 AM Draft	Passed
<input type="checkbox"/> Agent Transfer - Recurring or One Time No Match x)	-	Feb 12, 2024 10:09 AM Draft	Failed
<input type="checkbox"/> Happy Path - Line 1:012324, User specifies adding date just for this month	-	Feb 12, 2024 10:09 AM Draft	Passed
<input type="checkbox"/> Happy Path - Line 3:003333, False Notification	-	Feb 12, 2024 10:09 AM Draft	Passed

# Tracking the results of testing



Continuous Improvement

# Tracking the results of testing



Continuous improvement

**Google** Cloud