



master.of.code GLOBAL

# CHATBOT ANALYSIS FRAMEWORK

Test your conversational AI solution  
against the industry's best  
practices and benchmarks.

Get in touch via email:

[worldclass@masterofcode.com](mailto:worldclass@masterofcode.com)

Learn more:

[masterofcode.com](https://masterofcode.com)



## About This Whitepaper

There are an abundance of conversational solutions out there - leveraging voice or chat, on apps, websites, or a myriad of messaging channels. Brands continue to adopt these wonderful solutions to scale their customer service support, drive engagement, transactions and conversations, and reduce agent costs.

But how does one know their conversational AI solution is on the right path? That it's leveraging the best of the industry's leading practices, meeting users' increasing expectations, and fully taking advantage of the available technology to ensure frictionless and efficient experiences?

Typically businesses utilize scorecards or rating systems to ensure their tech-focused products are on an upward trajectory. However, in the conversational AI space, we are inundated with long-form best practice content that is difficult to digest at the speed at which bot teams are expected to optimize, develop, and launch on an ongoing basis.

Master of Code decided it was time to create a framework for businesses to use as a temperature check for their solution in terms of user experience and the solution's maturity and complexity.



# CONTENTS

<b>Introduction</b> .....	2
<b>Master of Code's Chatbot Analysis Framework overview</b>	
Chatbot Analysis framework components .....	4
How to use our scoring system .....	5
<b>Bot Analysis Framework</b> .....	6
<b>A deeper dive into the 8 framework components</b>	
Use case analysis .....	8
Bot persona and prescription .....	10
Personalization and context .....	11
Live agent integration .....	13
Conversation Design .....	14
Natural Language Processing (NLP) .....	15
Accessibility .....	16
Last but not least: CSAT .....	17
<b>Summary</b> .....	18



# MASTER OF CODE'S CHATBOT ANALYSIS FRAMEWORK

When we set out to create this analysis framework, we reviewed user-facing elements that would be critical to the overall [customer experience](#).

## Our Chatbot Analysis framework consists of the following 8 components:

### Use case analysis:

This addresses how 'large' the solution is, and how many and what types of use cases it covers and their complexity.

### Bot persona and prescription:

This analyzes how effective and prescriptive the virtual assistant's persona is.

### Personalization and context:

We look at how contextual the experience is.

### Live agent integration

How well the bot offers escalation, if at all.

### Natural Language Processing:

How well does the system handle natural language user inputs?

### Feedback:

Does the system allow the customer to share feedback?

### Accessibility:

Does the UI meet accessibility standards?

### Conversation Design:

An integral part of any bot's performance.

It should be noted this framework gives just a snapshot of how mature a conversational AI solution is. It acts as a starting point to identify where deeper analysis should be done. Every virtual assistant has its own set of metrics, benchmarks, and data points that can be unique to each industry or even the brand itself.

This framework only contains areas that would be viewable and tested by the bot's end users. After all, virtual assistants should be designed with user experiences at the forefront. If the user journey is poor, user adoption will always be stunted.





# HOW TO USE OUR SCORING SYSTEM

Using the below table, review the conversational AI system against each line item and based on its results, give a full or partial point to each element.

01

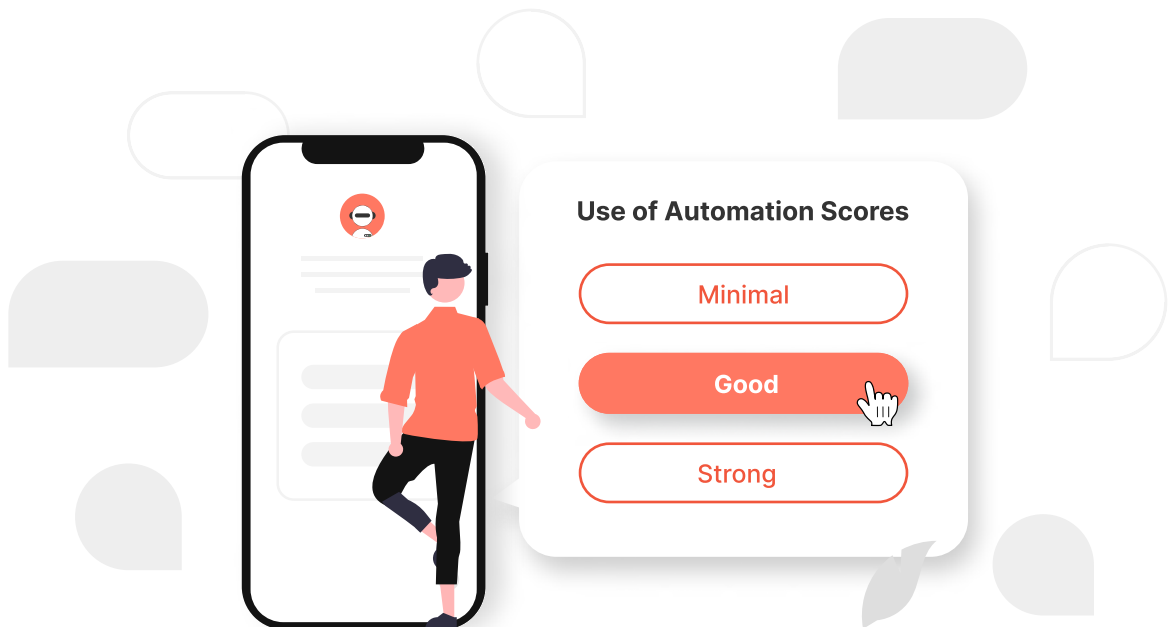
Give one point (unless otherwise noted) if the chatbot achieves the required criteria. Give a partial point if the chatbot does some or part of the described feature or element, and zero points if such feature is not currently included in the solution at all.

02

After scoring each section, add up the points out of a total of 20 and review it against our scoring system table on the following page.

03

There are three score categories your bot may fall into; minimal, good, and strong uses of automation. This will give you a better picture of where your chatbot solution stands in providing an engaging, effective, conversational experience that drives user adoption and containment.





# CHATBOT ANALYSIS FRAMEWORK

( Yes = 1 pt, Partial = 0.5 pt, No = 0 pt unless noted)

<b>Use case analysis</b>	Approximate # of use cases <i>(If less than 5 = 1 pt, Between 6-10 = 3 pts, If 11+ = 5 pts)</i>	
	Does the bot have transactional (end to end) use cases?	
	Is it clear that the bot offers use cases that are faster and more efficient than web/app user flows?	
<b>Bot persona and prescription</b>	Does the bot identify itself as a bot/virtual assistant?	
	Does the bot have a trustworthy, personable and consistent tone of voice?	
	Does the bot clearly inform the user of its scope and capabilities?	
<b>Personalization and context</b>	Does the bot leverage personalization and/or slots?	
	Does the bot utilize APIs, SDKs or integrations? (i.e: does the bot know and use information about its users)	
<b>Live agent integration</b>	Does the bot have live agent handover integration?	
	If after hours does the bot offer other options for users to get support? (i.e: after hours flow)	
<b>Conversation Design</b>	Is the content clear, concise, and digestible?	
	Does the bot leverage its channel capabilities? (structured content etc)	
<b>Natural Language Processing</b>	Does the bot handle free form NLP responses?	
	Does the bot allow for auto escalation based on natural language?	
<b>Accessibility</b>	Does the chatbot UI meet accessibility standards?	
<b>CSAT</b>	Does the bot allow for a customer feedback or offer a survey?	
<b>Total Score</b>		___ /20

Print this page and add your scores!

Print this page and add your scores!



# REVIEWING FRAMEWORK SCORES

Use the following table to understand how the score of the framework speaks to the analyzed conversational AI's complexity and sophistication.



## Minimal use of automation

Between **1-10**

This chatbot has minimal features, use cases and/or technologies to drive automation and efficiencies. There is a large opportunity for the business to take further advantage of the technology and abilities of conversational AI.

There is a large opportunity for MOC to help scale the brand's bot to grow ROI, user engagement, retention and reduce agent costs.

[Get in touch](#)



## Good use of automation

Between **11-15**

This chatbot shows good uses of automation, whether it be use of APIs and integrations or a larger list of use cases - there are good examples of useful features here.

There is an opportunity for MOC to take the brand's bot from simple to robust with further consultation on use case prioritization, expansion of transactional use cases and integrations.

[Get in touch](#)



## Strong use of automation

Between **16-20**

This chatbot shows strong uses of automation in a conversational AI solution. Many best practices and benchmarks outlined in the Framework have been implemented and there are prescriptive use cases that offer value.

MOC can help further scale your AI strategy, prioritize use cases, develop roadmaps, and grow ROI by offering our consulting services.

[Get in touch](#)



## A DEEPER DIVE INTO THE 8 COMPONENTS

Here's a breakdown of why each component is critical to a virtual assistant's overall success:

### USE CASE ANALYSIS

The larger the number of use cases, the more powerful the solution. By handling more flows, questions, and transactions, the speed and level of support a brand can offer is exponential.

With conversational AI technology continuing to mature, its upward trajectory results in additional opportunities for automation and use cases that a conversational AI solution can handle without human intervention.

If a brand's virtual assistant can handle 4-6 use cases, that's just scratching the surface on the level of automation it could achieve. We've designed chatbots with more than 40+ use cases that are a combination of transactional, instructional, and wayfinding.

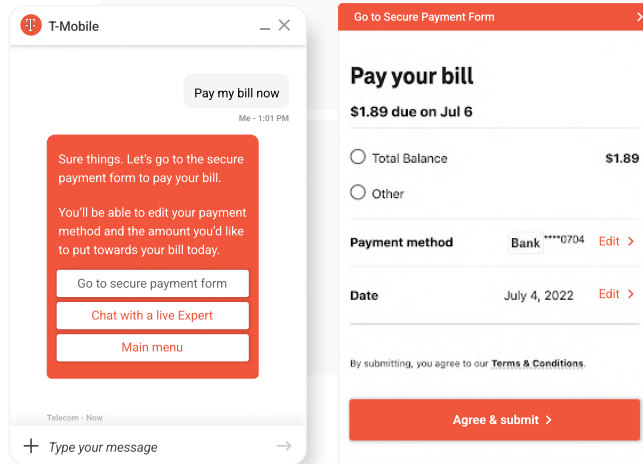
**Why does this metric matter to our framework?**

**It gives a good indication of how mature the brand's solution is.**



We also examine if the solution contains transactional use cases. We define these as: flows where the bot 'does' the action for you in the bot experience, without human intervention nor leaving the messaging window. These can include but are not limited to:

- Payment use cases
- Account updates
- Subscription or plan modifications



We've seen the rise in simple 'FAQ bots'. At Master of Code, what we call 'FAQ bots 2.0' include transactional use cases where instead of the chatbot telling the user how to complete the action, it does it for them.

The final factor during use case analysis is determining if the user flows within the automated experience appear to be faster and more efficient than current channel experiences. In other words: can the user accomplish a task, get information, or have questions answered in a faster fashion than if they tried the same action on the brand's website or app (if available)?

This is critical for virtual assistants to be successful, as users will not switch to a conversational channel unless it's fewer steps than the current state.



## BOT PERSONA AND PRESCRIPTION

Conversational AIs shouldn't be robotic. People prefer personable interactions so virtual assistants need to employ a helpful, personable and consistent persona for the best user experience.

It's also critical that a virtual assistant identifies itself as one to the user, to mitigate confusion that they're chatting with a human. The bot not only needs to be transparent to the user that it's a conversational AI, but also needs to share what it can do or help with.

Expect poor bot performance from prompts that ask the user general, open-ended questions on what it can assist with, making the user think that the bot is all-knowing and can help them with a myriad of questions. If the bot can't, the user will be greeted with fallback or error messages, causing them to most likely exit the experience and be wary of coming back to it.





## PERSONALIZATION AND CONTEXT

If there's a lack of [personalization and context in your chatbot](#), chances are the user is doing the heavy lifting of providing information in order to get the answers they need.

The bot will have to ask the user questions instead of using APIs or integrations to anticipate their needs. For example, if a user is inquiring about an order, a non-personalized nor unauthenticated experience would mean the virtual assistant will have to ask what issue they're having with their order, request a lengthy order number, and then give the customer an answer. The alternative would be a user asking where their order is, and the bot would check in the backend via integration which orders are active and haven't been delivered and relay the information, anticipating the user's need.

A well designed solution ensures the conversational experience is more efficient and faster than the current state otherwise it's not an optimal user experience. Using the above example, it's likely faster for a user to log into their account on the website, go to their orders to see the latest update, not needing to enter an order number to do so.

The illustration shows two user profiles, Mark and Olivia, each with a list of banking services. Mark's profile is on the left, and Olivia's is on the right. Both profiles have a 'Banking' icon above their service lists.

**Mark**  
Has a chequing account at the bank

- Online Banking
- Check account balance
- Pay a bill
- Apply for a credit card

**Olivia**  
Uses multiple services from the bank including a credit card, savings account, and mortgage.

- Online Banking
- Check account balance
- Make a credit card payment
- Make a mortgage payment



Another factor we analyze is how the virtual assistant leverages the use of slots. If the user enters information, does the bot save it and use it later on, to dictate the user's experience? Or regardless of the information provided to the solution, the bot states the same, static information, causing a less than optimal user experience.

If you're investing in conversational AI, ensure your technical roadmap includes **integrations**, leveraging **APIs** and **SDKs** for a more personalized, efficient user experience as well as the implementation of **slots** and **entities** to offer further contextualization of conversations.







## LIVE AGENT INTEGRATION

Today, there are still limitations to what virtual assistants can help customers with. For more complex inquiries or situations, human-to-human conversations are the best user experience.

While we understand that not all brands with conversational AIs may have the technical systems in place or budgets to employ live chat agents, it's a UX goal to work towards.

If there are live agents standing by for support, the chatbot should have an 'after hours' flow for users who escalate outside of the hours agents are online. It's not a **user-centric experience** if the bot offers human help, transfers them to the queue, and then the user is greeted with a 'sorry, our agents are offline' message. The optimal experience here is the bot confirming they're not the best solution to help and let the user know their agents are offline before escalation, and offer a more appropriate option.

### The escalation option could range, depending on the bot platform you're employing:

- **SUFFICIENT:** Inform the customer of when the agents will be back online before escalation so they can reach out at that time.
- **GOOD:** Ask the user to leave their email address and message in the bot window that will be sent to a customer support email channel and further dealt with via email (if the above recommendation is not technically available and if the query can be answered in a timely manner).
- **BETTER:** If live chat is unavailable but there is another customer support channel that is available, offer that one for immediate assistance. For example, when phone support is 24/7 but live chat is available during only business hours.
- **BEST:** Ask the user to leave a message which will go into a queue and will be answered by an agent as soon as they're back online.

Until virtual assistants are all-knowing and can handle the most complex of customer issues, it's essential we can offer a seamless handoff to a human when a customer needs it.

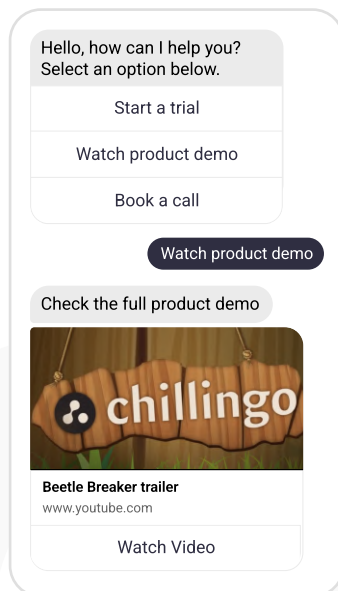


## CONVERSATION DESIGN

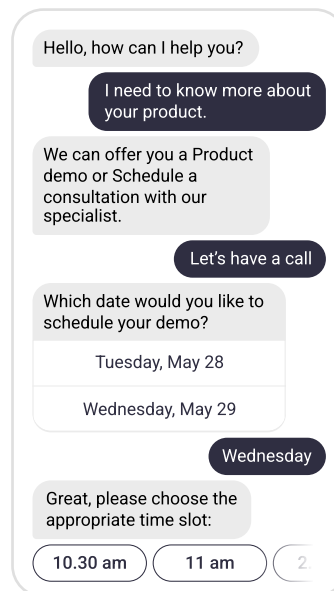
Although [conversation design](#) is part of all the aforementioned factors, we look specifically here at how clear, concise, and digestible the bot content is and how well it leverages channel capabilities.

We know that on web and mobile, users scan more than they read. To mitigate misclicks and mistaps, content needs to be succinct and clear for the best user experience. Too often we see conversational solutions that essentially copy and paste web content into a chat window - one that is a fraction of the size of a browser window. Lengthy content requires undesirable scrolling just to answer a user's question.

### Rule-Based Chatbots



### Conversational AI



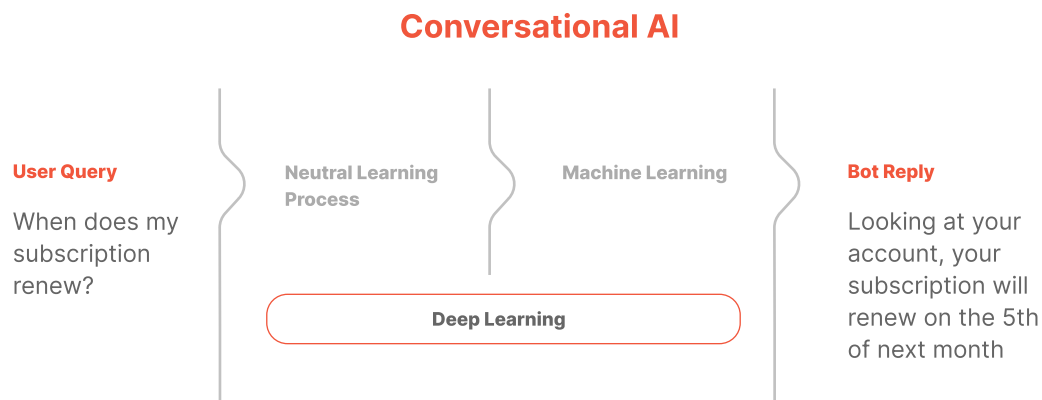
Another conversation design best practice we look for in chat solutions is whether the channel-specific structured content is effectively leveraged. For example, in a web chatbot, are quick replies and static buttons being used? Does the button text wrap properly or does it get cut off if too long? In an SMS bot, does it offer A) B) C) options for easy selection? Or just Yes and No options, which can be limiting if used exclusively.

These factors all play into providing a clean, simple, and frictionless user experience, mitigating errors and fallbacks.



## NATURAL LANGUAGE PROCESSING (NLP)

As chatbot technology continues to become more sophisticated, so do user expectations on how conversational AIs should function. Users now expect [chatbots to understand natural language](#) and will often prefer to type responses instead of clicking through buttons and links to get the information they need. If your bot doesn't employ NLP today, chances are your users are often met with error or fallback messages. By using NLP, the chatbot is able to have more natural conversations with humans. By conversing, users are able to get help faster than if a bot requires a user to click through menus to get where they need to go.



### An additional NLP consideration is how effectively the bot offers immediate escalation:

- If an intent is triggered that must require human intervention. E.g: canceling an account or subscription
- Human sentiment such as frustration or anger where a human is best to support
- If a user requests a live agent directly in the chat experience and the virtual assistant actions the request

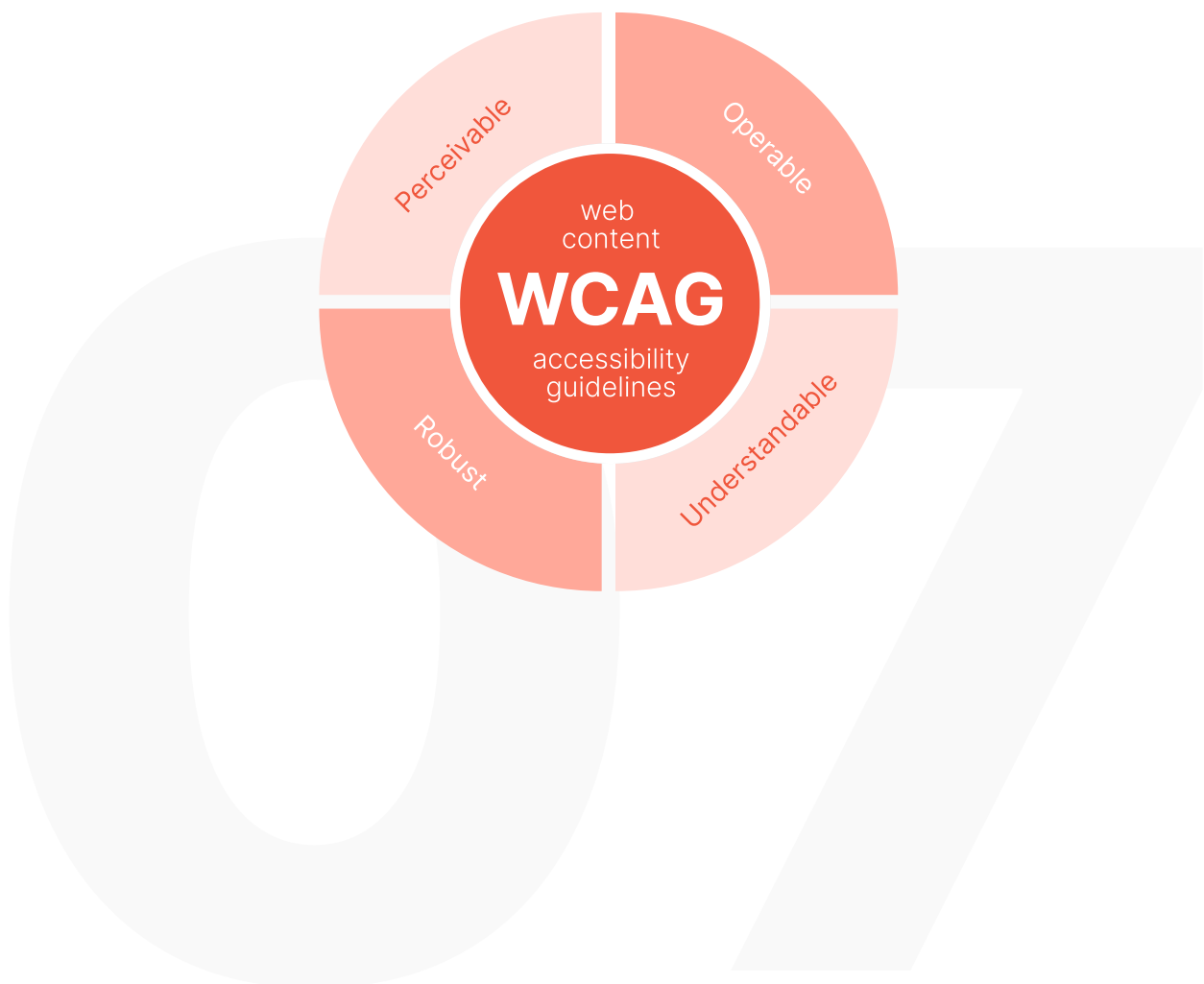


## ACCESSIBILITY

This has been a popular topic discussed over the last couple of years but often has not been practiced. Having chat windows and UIs that are colorful and engaging are great but the color scheme must follow WCAG guidelines pertaining to field labels, legends, error messaging, keyboard access, and color contrast ratios.

The UK government is a great example of implementing accessibility. They developed [accessible chatbot patterns](#) more than 6 years ago that follow WCAG guidelines.

Want to check if your virtual assistant has the right contrast ratios? Use this handy [WCAG compliant contrast checker](#) here.



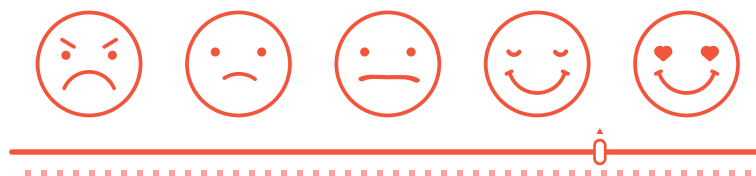


## LAST BUT NOT LEAST: CSAT

Without feedback, we can't improve. Virtual assistants should allow their users to give feedback so chatbot teams can easily review and continue to make improvements and optimizations.

Conversational solutions often have quick surveys that trigger when a user signals that they are wanting to end the conversation. It's a great way to quickly understand if there is an overall poor user experience, giving insight into where things may be breaking down that require tuning or redesign.

Survey designs can range from using a Medallia Promoter Score to leveraging emojis to understand user experience. If the bot receives a low score, it should prompt the customer to share more information either via free text or additional option selections to better understand their score.





## IN SUMMARY

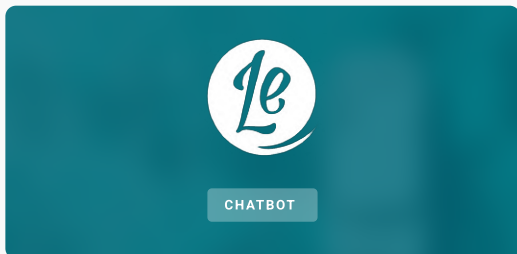
Having a framework to regularly reference and test your conversational solution against, will not only ensure you're following Conversational AI's best practices but will also highlight gaps in user experience and feature opportunities.

By partnering with Master of Code, we will use this framework to help you:

- Understand how mature your conversational AI solution is
- Validate best practice feature sets and functions
- Identify new and optimize existing use cases further engagement and containment.

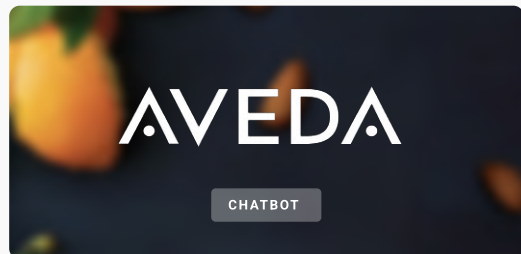
From there, with additional backend access, we can perform further analysis and audit the solution's performance based on analytics and metrics including containment, escalation rates, and goal funnels.

## WE BUILD CONVERSATIONAL AI CHAT AND VOICE SOLUTIONS FOR LEADING BRANDS



An integrated e-commerce chatbot that generated more than **\$300,000** of revenue in the first **90 days** of launch.

[Learn More](#)



A Messenger chatbot that increased the average weekly appointment bookings by **7.67x**.

[Learn More](#)



## About Master of Code Global

Master of Code empowers enterprises to reap the benefits of using conversational AI solutions to increase customer satisfaction, retention, and acquisition, and save time and money by automating their users' most common requests.

Our team of Conversational AI experts can build a strategic roadmap on how to leverage automation to solve customer pain points. Through our proprietary conversational design methodology and processes, we prioritize use cases that will provide the most impactful conversational experiences that convert and generate the highest ROI and CSAT for your customers.



master.of.code GLOBAL

# WE'RE HELPING BUSINESSES REDEFINE AND ELEVATE CUSTOMER EXPERIENCES WITH CONVERSATIONAL AI

Contact our team

Get in touch via email:

[worldclass@masterofcode.com](mailto:worldclass@masterofcode.com)

Learn more:

[masterofcode.com](https://masterofcode.com)

