ISSN: 2754-6659

Journal of Artificial Intelligence & Cloud Computing

SCIENTIFIC Research and Community

Review Article

Open Access

Chat Bots as the Future Multi Channel Customer Support user Experience Strategy : An Alternative View Point on Reusability Across end Customer and Customer Support Channels

Tanmaya Gaur

Bachelor of Engineering (Electronics and Telecommunication), Birla Institute of Applied Sciences, USA

ABSTRACT

Organizations that sell products or services need to provide their customers with experiences that allow them to buy, renew, upgrade, service, or troubleshoot their products or services. These customer support experiences are built using a combination of end-customer experiences, often provided through web or native applications, and additional retail or care experiences where customer support agents access CRM, CSM, and ERP applications to address customer concerns. Traditionally, retail or care agent experiences have been different from web customer experiences, as they are purpose-built and optimized for quicker reuse by the organization's pre-trained staff, whereas end-customer experiences are often simple and intuitive. Historically, this distinction has applied even when reuse was technically possible. In the modern AI-driven world, many customer-facing experiences are being replaced by AI chatbots. This paper explores the concept of chatbot experiences being the sole interface across end-customer, care and retail, leveraging the same core to power both advanced user and simplified end-customer conversations. This approach promotes reusability, eliminating the need to develop additional UI-based experiences for both customers and agents.

***Corresponding author**

Tanmaya Gaur, Bachelor of Engineering (Electronics and Telecommunication), Birla Institute of Applied Sciences, USA.

Received: November 07, 2023; **Accepted:** November 11, 2023; **Published:** November 25, 2023

Keywords : CRM, Call Center, Chatbots, Conversational AI

Introduction

Traditional end-user experiences have typically depended on visual web or mobile interfaces made available to end-customers. These interfaces allow the customers to 'self-serve' their needs without needing to contact customer support. When customers do reach out to an organization's support teams, the support employees use CRM, CSM, or ERP user interfaces to assist them. For most organizations, a high-level view of their enterprise capability architecture reveals a similar layout.

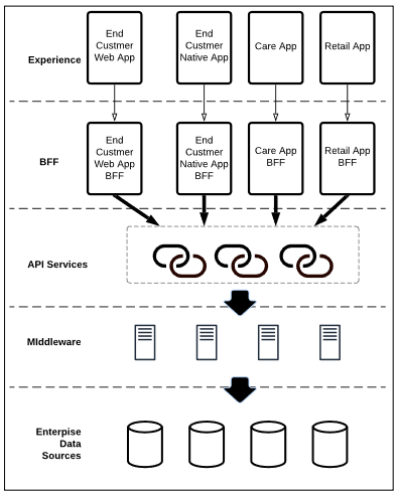

Figure 1: Traditional Organizational Architecture Layout

Traditional end user experiences have long relied on visual web or mobile experiences presented to the customers. It is recommended to use a Backend for Frontend (BFF) pattern for a variety of reasons which provides a dedicated backend service for each specific frontend interface. This tier can help optimize performance and helps UI tailor API response to the unique needs of different application interface. The key benefits are the agility and abstraction it provides for backend as well as security. This layer can handle data aggregation, be used for caching as well as to filter and hide unnecessary data to solve security and efficiency. Given these advantages and interface specific concerns, all UI interfaces often have a supporting BFF.

Native applications provide significant performance advantages when compared to web apps. They are faster, responsive, and can be more interactive. However, they do come with the additional overhead of having to manage technical codebase. While it is technically possible for organizations to not build a native app OR avoid building full native by re-using the web application wrapped in a native container for code reuse, for the purpose of this document, we will consider that being a separate app as is often the case.

Figure 2: Traditional Customer Care

For customers reaching out to Customer care or seeking help at a retail location, if applicable, the employees use an employee facing application or group of applications to aid. The applications are traditionally built to solve the employee experience and for KPIs like resolution time, first call resolution, MTTR etc. This is different from end customer experiences which are simpler and prioritize ease of use.

There are additional factors that may drive care and retail experiences to be different. In a telecom company for example, Retail solutions may be more sales focused compared to care experiences being service focused. There are at times different access user agents between the two as well. There may be a third group of back-office user agents with a completely alternate set of needs and criterion. There is often also different security, fraud and risk profiles which drive the implementation.

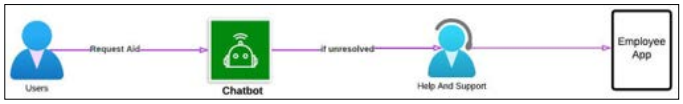

Figure 3: Modern Implementations with Chatbots Driving Cotainment

Another evolving aspect is the chatbot solutions. Initially, the customer support interfaces initially allowed customers to directly connect with employees which is no longer the case. Most organizations have now started incorporating smarter capabilities like IVR and chatbots in the mix which field the initial customer interaction. These chatbot experiences may be used to try and resolve the issue outright without involving a human employee but are also used at times to capture additional metadata for better routing or assignment of the customer interaction. Companies often have a KPI which tracks chatbot based call containment, often for tasks which can be easily automated. These automation chatbots or IVR layers drive yet another interface integration into the overall company ecosystem as laid out in Figure 4.

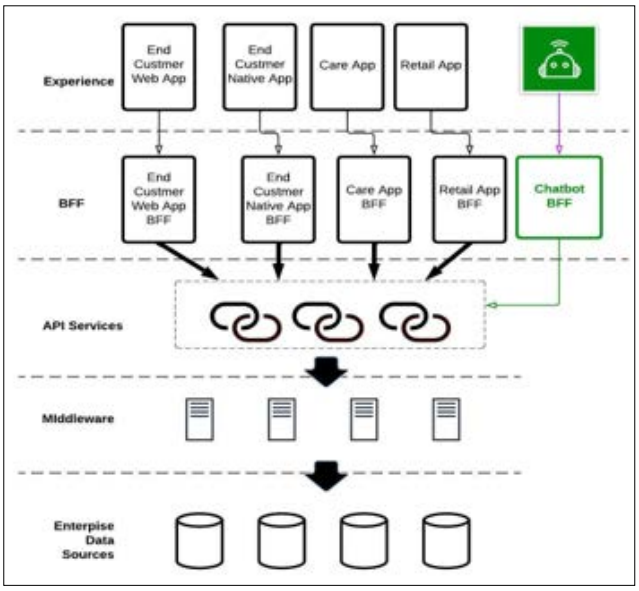

Figure 4: Modern Organizational Architecture Layout including Automation Bots

From now on, this paper will refer to the various engagement experiences as different 'channels. In our example, care and retail are different channels even though they are both internal employees facing. Having to develop and maintain different experiences and technology stacks across these channels comes at costs. Every time a downstream dependency changes or a new enhancement is required, the organization needs to make changes in multiple applications. Additionally, if these applications are developed in different technologies, that increases complexity of the change and maintenance.

While chatbots are being introduced to build customer support experiences, all the other channels continue to be developed. This paper will discuss utilizing these chatbots as an interface option across all these interfaces, pros and cons of the approach. The paper will discuss how chatbots, especially powered by generative AI can solve cross cutting concerns across the stack while significantly lowering costs. The paper will also go into details of chatbots as a technology and possible issues with going all in on them.

Understanding Chatbots

What are Chatbots

IBM defines a chatbot as a computer program that simulates human conversation with an end user. While not all chatbots are 'AI', it has become more prevalent given recent advances in conversational AI technology, especially around generative AI and Natural language processing.

Generative AI Chatbots

Generative AI chatbots are trained on large amounts of data from the transaction they aim to mimic. Generative AI-powered chatbots then create original content by predicting the most likely word to follow based on patterns in their training data. These chatbots are open domain chatbots that can perform a variety of tasks like summarizing, translating data. creating content and driving conversational flows and routing. Generative AI chatbots are often used in customer service to reduce wait times and improve the customer experience. They are also often used to help human contact center workers by allowing them to focus on more complex cases.

It is important to note that these chatbots are not search engines and do not provide a set of results in response to a specific search. Their responses may include a mix of correct and incorrect information. We will discuss this concern in the section on understanding limitations of chatbots.

Chatbots Compared to Humans

As mentioned, many organizations are replacing functions usually performed by a human employee with chatbots.
Listed below are some Key Differentiators

- **Emotional Intelligence & Empathy:** Humans possess emotional intelligence and empathy. While chatbots can be trained to mimic them in the interactions, they lack empathy and emotional intelligence.
- **Complexity:** Humans can handle complex queries whereas chatbots are currently more suitable for specific, pre-defined type of queries.
- **Context Aware:** Bots are not always able to understand the context of the conversation. As an example, they may not be able to understand that the conversation is a specific regional dialect and appropriate understanding.
- **Consistency:** Humans interactions lack consistency unless systematically enforced. Bot interactions in comparison are highly consistent.
- **Scale and Costs:** Bots are highly scalable and may provide

higher ROI at scale. Humans can't always provide scalability beyond a threshold.

- **Error Prone:** This may be debatable given AI powered bots have downsides like hallucination, but if they are well scripted with proper guard-rails, bots are less error prone than humans.

Chatbot and Customer Support

Chatbots are starting to become more common in customer support than other areas. The reasons vary by type of organization and support needs, but certain consistent and common themes are driving this adoption.

- Chatbots are cheaper to operate at scale than human interactions. This can help lower costs.
- Well defined chatbots minimize errors and provide more accurate results.
- Chatbots may provide avenues to increase customer engagement beyond human scaling thresholds.
- Chatbots are highly available including 24 * 7 support
- Chatbots can improve response times versus humans, which may provide better customer satisfaction.
- Chatbots may be able to provide support across multiple channels. This is the crux of this paper, and we will deliberate more on this topic.
- If trained, Chatbots can provide multi-lingual support.

Chatbot Challenges

It is not like integrating chatbots into an organization is without challenges, this section list known challenges in chatbot adoption

- **Training and Setup:** Training data for chatbots is hard to come by and may require sanitization of existing data. There are often complex, regulatory or compliance aspects that need to be factored in given the nature of the enterprise. Simple FAQ bots are often easier than transactional bots that need access to customer data and hence need solution for authentication, authorization like concerns. It often takes significant time and effort to train the AI.
- **Security:** Organizations have a need to keep data secure. Ensuring only relevant data is requested from the customer or made available in responses and is securely transmitted when not at rest is key concerns aside from access security.
- **Adoption:** Chatbots may not be as likable as humans and hence need significant work to generate engagement and better the user experience.
- Vernacular Aspects like being able to support multiple languages, and ability generate relevant insights or understand the ask properly is often a concern.
- AI Chatbots often brings about additional complexity including figuring out the right mix with regular chatbots. Not all scenarios need AI chatbots and there need to be ways to seamlessly handoff between the various experiences. Additionally, there have been known instances of bias, unpredictability, ethical risks etc. with generative AI powered bots that need to be overcome.

Channel Variances and Chatbots

Form Factors

End-customers or employees access the applications needed on different devices based on availability, personal choice or due to it being enforced by the organization. The form-factor refers to this device's size, shape, and style. When we refer to form-factor of a device especially in context of websites and apps, it specifies the type of device it is such as a tablet, a smartphone, a PC etc. More importantly, these different devices have different breakpoints which refer to the screen width at which a website or application should adapt its layout and design to ensure optimal user experience. Optimal User-experience is important to be able to support the various 'form-factors' or 'breakpoints' in a usable fashion to gain adoption. Given the variations in these devices, especially across the different channels, different tech stacks or technologies become the solution norm.

Chatbot experiences which are like chat experiences do not necessary vary significantly by device form factors and breakpoints and hence do not need different implementations between device type variants.

User Experience and Navigation

With different channels and complex user experiences, a lot of effort goes in development. There is usually a process of planning an Information Architecture (IA). This basically means the structure of the content; how the menu or navigation system should set things out. For larger services with lots of content, this structure gets complicated. On a particularly information-heavy website, users might need to move through several layers of navigation to find a specific function or utility.

This precedes an often-costly design, development and maintenance cycle with these costs multiplied for the different user experiences across device and channel types.

Chatbots on the other hand work brilliantly for such scenarios. If the user can express that question, chatbots can be trained to answer them. No digging or diving complex navigation structures that were developed. This is more satisfying user experience for the customer and for the development team. Refer figure 5 below for a representation of this variance.

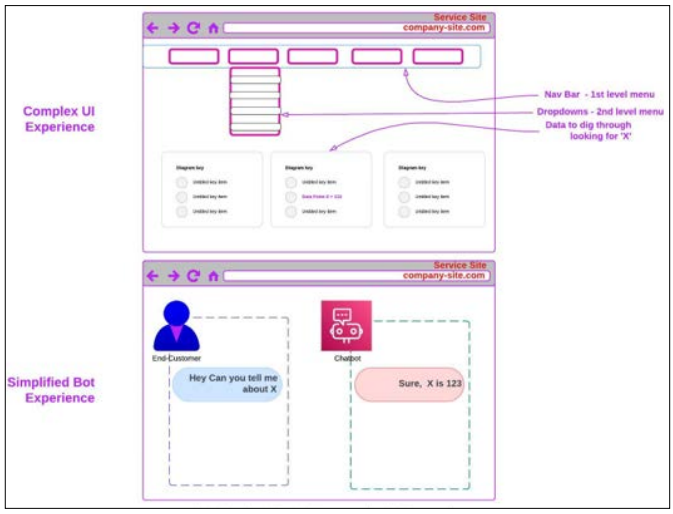

Figure 5: Complex UI Experience Versus Simplified Bot Experience

Functional Requirements

Different user types of teams often have different experience requirements or differing priority. As brought up earlier in this manuscript, a company's care operation may be heavily service focused while the retail operation may be more sales driven. These variations often lead to different technologies of development patterns getting used. Service flows are simpler and more modular with little to no reuse whereas sales flows are generally spread out across multiple screens and follow a similar cart and checkout strategy. There may be modules like payments which are expected to be reused. These decisions often drive not just the

user experience but also technology choices.

While there is the limitation of chatbots being not fit for GUI intensive tasks, something we will discuss later in this section. Outside of GUI intensive tasks, Chatbots are well abstracted between the user presentment and backend implementation and do not need to vary in implementation due to this variance.

Functional Capabilities

Going back to the retail/care use-case, retail users may want access to hardware like scanners, printers etc. which is not the case for care integration. These integrations, especially hardware is often better solved by using native OS applications instead of web apps. The same case applies to end customers, where based on the capability, native OS based apps may be better suited or more optimized for certain tasks. Developing separate user experiences, one native and one web based in this example comes at significantly higher cost. There is a middle ground in building a hybrid app where the native app wraps a web app-based solution providing the best of both worlds. This however comes with the hidden costs of complex maintenance and limited troubleshooting available to such web wrapped apps.

Wrapping bots inside native or web shells does not have similar complexity as bots have minimal presentment logic and are mostly API driven. Hence bots are not affected by this variance.

NFR like Security, Fraud, Compliance

Security, Fraud and Compliance aspects often drive many of our technology choices. Fraud has become more prevalent, even via insider threats in many companies and hence internal apps too are often driven by these needs. Zero trust often dictates limited data presentment based on authorization and need of the specific interaction. This gets complex fast for large enterprises as the same user interface is overloaded by these concerns.

Bot implementation if well-structured are simpler to implement and operate when faced with multiple such guard-rails.

Ease of use and Multi-Modality

Chatbots are often easier to initiate interaction with, whether through text or voice, The user is not waiting for the website to load, or to navigate to specific section of the webapp.

The nature of voice-based chatbots also means they can be hands free. This may offer advantages where screen-based UI isn't appropriate. As an example, interacting with maps while driving is safer when done using voice assistant than by using the interface.

Where Bots don't Make Sense

Chatbots are not well suited for complex and UI intensive use-cases. They also don't work very well for scenarios where large amount of data needs to be entered. Also, as mentioned in the earlier section, they often lack understanding and empathy and may not be well suited for certain scenarios.

Additionally, as it comes to different user-scenarios, chatbots may not be well equipped on intuitive to enter and deal with large amount of data or data types like images.

Bot Centric Organization's Layout

Given how bots can be applied across concerns and user-types, an organization can significantly simplify its stack.
A Lean Bot-Based Stack is Depicted in the Figure 6 below

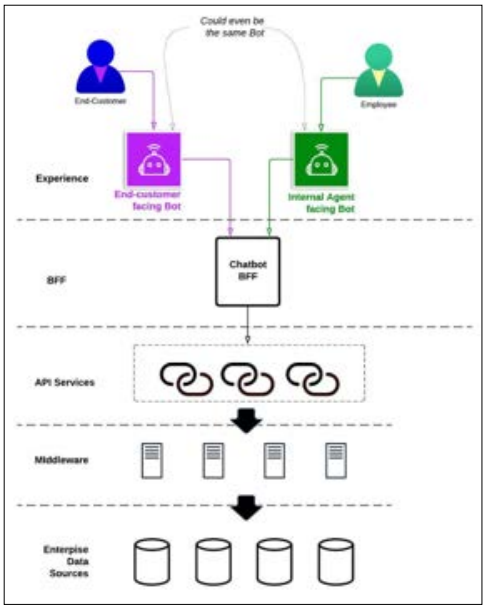

Figure 6: Modern Bot Centric Organizational Architecture

Conclusion

Chatbots present a unique opportunity not just to lower costs but also to simplify the overall company architecture, especially in the customer support space. As detailed in this paper, it can serve as not just a containment tool fronting the human support agents but can also be utilized as the only customer facing experience as well as the option available for customer care agents if a call does come through. The simple nature of the interaction and experience proves it easy to overload it with the various channel concerns. That said, chatbots are not well suited for all scenarios so it is important to understand these limitations when implementing such a solution [1-7].

References

1. 5 Blunt truth about chatbot limitations. https://www.iadvize.com/en/blog/limitations-ai-chatbots.
2. What's the Difference between Web Apps, Native Apps, and Hybrid Apps? https://aws.amazon.com/compare/the-difference-between-web-apps-native-apps-and-hybrid-apps/.
3. What is chatbots. https://www.ibm.com/topics/chatbots.
4. (2014) Responsive Web Design: Relying on the Form-Factor. https://www.red-gate.com/simple-talk/development/dotnet-development/responsive-web-design-relying-on-the-form-factor/.
5. Interactive Voice Response. https://www.ttec.com/glossary/interactive-voice-response.
6. (2017) 5 reasons why a chatbot is better than a website. https://medium.com/hijiffy/5-reasons-why-a-chatbot-is-better-than-a-website-75477cee7341.
7. (2023) The Role of Chatbots in CRM. https://medium.com/@shreesagarwani/the-role-of-chatbots-in-crm-0407fbf5a205#:~:text=CRM%20systems%20rely%20on%20data,updates%20on%20their%20order%20status.