Semester 2 Project

Unified Messaging Automation using Amazon Web Services

From

Yashwanth Pindi

Aditya Nisal

Namita Deshpande

Bhushan Mahajan

Professional IT Business and Digitalization (ProITD)

Supervisor: Prof. Nico Schönnagel

Hochschule für Technik und Wirtschaft Berlin

1 Index

2 Acknowledgement

We would like to express our deepest gratitude to several individuals who were instrumental in the successful completion of this semester project.

First and foremost, we extend our heartfelt thanks to our first supervisor, Prof. Nico Schönnagel. His exceptional guidance, insightful feedback, and constant support have been invaluable throughout this project. Prof. Schönnagel's expertise and dedication have significantly enhanced the quality of our work, and we are deeply appreciative of his time and effort.

We would also like to acknowledge and show our appreciation to all those who have remotely assisted us during this journey. This includes our family and friends, whose encouragement and support have been an important contribution to the completion of this research work. Their understanding and patience allowed us to focus on our project, while their moral support kept us motivated.

In addition, we are grateful to our peers and colleagues who provided constructive criticism and suggestions, helping us refine our ideas and approaches. The collaborative spirit and shared knowledge within our academic community have been a great source of inspiration and improvement.

Finally, we recognize the importance of the resources and facilities made available to us by our institution. The access to research materials, technical support, and a conducive environment for study and experimentation has been crucial in bringing this project to fruition.

To everyone who played a part, directly or indirectly, in the successful completion of this project, we extend our sincere thanks. Your contributions have been deeply appreciated, and we are grateful for your support.

Thank you.

3 Problem Statement

In today's fast-paced and information-driven environment, organizations face significant challenges in providing timely and efficient access to essential information and services. Our esteemed organization, OEV, currently lacks a centralized and user-friendly communication system to effectively address inquiries and facilitate interactions with stakeholders through chat or mobile phone. This gap results in inefficiencies, such as cumbersome navigation through complex phone trees, delays in connecting with appropriate departments, and difficulties in scheduling meetings.

The primary issues identified include:

1. Fragmented Information Access: Individuals seeking information about OEV's history, service offerings, or areas of specialization encounter difficulties in obtaining a concise and comprehensive overview.
2. Inefficient Departmental Routing: Callers often face challenges in being directed to the correct department, leading to frustration and wasted time.
3. Complicated Meeting Scheduling: The current process for scheduling meetings is not streamlined, leading to potential delays and scheduling conflicts.

To address these challenges, OEV needs a centralized and automated telephone line leveraging Amazon Connect's cloud-based contact center capabilities. This solution should offer an intuitive and efficient way for callers to access necessary information, connect with the appropriate departments, and schedule meetings. Additionally, future enhancements should include the implementation of Interactive Voice Response (IVR) technology, 24/7 availability, and integration with an internal knowledge base to further improve the user experience.

The goal of this initiative is to establish the OEV Information Access Line, transforming it into a comprehensive and user-friendly hub that streamlines communication and enhances interaction with OEV.

4 Introduction

The objective of this project is to establish a dedicated telephone line that will serve as a central information hub for inquiries pertaining to our esteemed organization, OEV. OEV is known for its innovative solutions and exceptional service offerings across various sectors. The organization consistently strives to enhance the experiences of its customers and stakeholders by adopting cutting-edge technologies and best practices. This initiative represents a significant step in that direction by leveraging the advanced capabilities of Amazon Web Services (AWS) and other communication tools to improve accessibility and streamline communication.

The primary AWS services utilized in this project include Amazon Connect, Amazon Lex, and Amazon Bedrock. In addition to AWS services, this project also incorporates Slack as a critical component of the communication strategy. The establishment of this dedicated telephone line is envisioned to transform how OEV handles inquiries, offering a more efficient, responsive, and user-friendly experience to all stakeholders. By centralizing information dissemination and leveraging advanced technological solutions, OEV aims to reduce response times, increase accuracy in information delivery, and improve overall satisfaction among customers and stakeholders.

Furthermore, this initiative aligns with OEV's broader strategic objectives of digital transformation and operational excellence. By adopting and integrating state-of-the-art technologies, OEV not only enhances its service offerings but also positions itself as a leader in leveraging technology to drive innovation and improvement in customer service.

In summary, this thesis explores the development and implementation of a centralized telephone information hub for OEV, detailing the integration of Amazon Connect, Amazon Lex, Amazon Bedrock, and Slack. Through this project, OEV aspires to set new standards in customer and stakeholder engagement, reaffirming its commitment to excellence and innovation in every aspect of its operations.

5 Theoretical Background

Effective communication systems are crucial for the efficient operation of modern organizations. The integration of advanced technologies such as cloud-based contact centers and natural language processing can significantly enhance the accessibility and responsiveness of information services. This section explores the theoretical foundations that justify the implementation of an automated telephone line using Amazon Connect and AWS Lex at OEV.

Cloud-based contact centers represent a paradigm shift from traditional on-premise systems, offering enhanced scalability, reliability, and cost-effectiveness. According to the theory of cloud computing, leveraging cloud infrastructure allows organizations to dynamically adjust resources based on demand, ensuring optimal performance even during peak call volumes. Amazon Connect, as a cloud-based contact center, provides a robust framework for managing customer interactions with minimal infrastructure overhead.

Automated menu systems and IVR technology are grounded in the principles of user-centered design and system efficiency. Automated menus simplify navigation by presenting users with a clear set of options, reducing the cognitive load and minimizing the need for manual intervention. IVR systems enhance this experience by enabling natural language understanding, allowing users to interact with the system conversationally. This aligns with the theory of human-computer interaction, which advocates for intuitive and responsive interfaces.

6 Technical Terms

6.1 Amazon Lex

Amazon Lex is a service for building conversational interfaces into any application using voice and text. It enables the creation of chatbots that can engage with users in natural language. Lex uses deep learning technologies for automatic speech recognition (ASR) to convert speech to text, and natural language understanding (NLU) to recognize the intent of the text, enabling developers to build sophisticated, natural language conversational bots. Lex integrates seamlessly with AWS services, allowing for easy deployment and scaling of chatbots across various platforms.

6.2 Amazon Connect

Amazon Connect is a cloud-based contact center service that simplifies customer service management. It enables businesses to set up and manage a customer contact center with minimal infrastructure setup and maintenance. Amazon Connect offers features like skills-based routing, voice recognition, and real-time and historical analytics. It integrates with other AWS services, enabling a comprehensive solution for customer interaction, data analysis, and customer relationship management (CRM). The service is highly scalable, allowing businesses to grow their contact centers according to demand.

6.3 Amazon Bedrock

Amazon Bedrock provides a managed environment for building, training, and deploying generative AI models. It aims to simplify the process of leveraging large-scale machine learning models, particularly for tasks involving natural language processing (NLP), computer vision, and other generative AI applications.

6.4 Slack

Slack is a collaborative messaging platform designed for teams to communicate and work together more efficiently. It offers channels for organizing conversations, direct messaging for private communications, and integrations with numerous third-party applications and services. Slack enhances productivity by centralizing communication and providing tools for file sharing, search, and notifications. It supports real-time messaging, voice and video calls, and integrates with other productivity tools like Google Drive, Trello, and GitHub, making it a versatile solution for team collaboration and project management.

7 Requirements Analyses

Upon dialing the designated OEV Information Access Line, callers will be greeted by a user-friendly automated menu featuring a selection of pre-recorded options. This intuitive menu system is designed to empower callers, allowing them to efficiently navigate to their desired information or service without unnecessary complexity or delay.

The incorporation of AWS Lex further enhances the system's functionality, enabling it to handle a variety of queries, including detailed company information, human resources-related questions, and requests for the company’s address.
The implementation of this dedicated telephone line is anticipated to significantly improve the efficiency and effectiveness of our communication channels.

By offering a centralized and automated platform for information retrieval and inquiry resolution, we aim to foster a more streamlined and satisfying experience for all individuals interacting with OEV. This project underscores our commitment to leveraging cutting-edge technology to meet and exceed the expectations of our stakeholders.

8 Concept

This project aims to create a scalable appointment scheduling system using various AWS services, including Amazon Bedrock, Amazon S3, Amazon Lex, Amazon Connect, and Amazon CloudFront, integrated with Slack for communication and an audio call feature through a web-based chatbot. Users can request appointments via a web interface or mobile app, with requests routed securely through Amazon CloudFront.

Amazon S3 stores necessary data, while Amazon Bedrock serves as the knowledge base, processing this data for accurate responses. Amazon Lex provides a conversational interface for text and voice interactions, linked to Amazon Connect for managing voice requests. Slack integration ensures timely notifications, and the audio call feature adds convenience. This system delivers high availability, security, and flexibility, making it an efficient solution for appointment management.

To address the identified challenges and enhance the efficiency of our communication processes, we propose the implementation of a dedicated telephone line that will serve as a central information hub for OEV. This solution is designed to streamline the way callers interact with our organization, leveraging the advanced capabilities of Amazon Connect and AWS Lex. The key components of this solution include:

8.1 Amazon Connect Integration:

a. Cloud-Based Contact Center: Amazon Connect will serve as the backbone of our new telephone line, providing a scalable and reliable cloud-based contact center solution. This integration ensures robust performance and accessibility, accommodating varying call volumes with ease.

b. Automated Menu System: Upon dialing the OEV Information Access Line, callers will encounter an intuitive, user-friendly automated menu. This menu will feature pre-recorded options that guide users to the information or services they seek, reducing the need for manual intervention and minimizing wait times.

8.2 AWS Lex Implementation

a. AWS Lex uses Natural Language Processing will be employed to enhance the system's interactivity and responsiveness. Utilizing NLP capabilities, AWS Lex will allow the system to understand and process natural language queries, providing accurate and contextually relevant responses.

b. Comprehensive Query Handling: The system will be equipped to handle a wide range of inquiries, including requests for detailed company information, human resources-related queries, and directions to the company’s address. This ensures that callers receive precise and comprehensive answers to their questions.

8.3 Interactive Voice Response (IVR)

The system's capabilities can be further augmented with the implementation of an Interactive Voice Response system. This advanced technology facilitates natural language understanding, enabling the system to engage in more conversational interactions, directly addressing callers' inquiries.

8.4 24/7 Availability

Envision an Information Access Line that operates continuously, offering uninterrupted access to essential information and basic assistance regardless of time zone. This expansion ensures unparalleled accessibility.

8.5 Amazon Bedrock Knowledge Base Integration:

Future iterations can explore the integration of the Information Access Line with our internal knowledge base. This would empower the system to provide real-time responses to frequently asked questions, further enhancing the efficiency of information retrieval.

By implementing this innovative project, we aim to foster a more streamlined and efficient communication experience for all parties interacting with OEV. The OEV Information Access Line promises to transform into a constantly evolving central hub, providing a user-friendly and comprehensive gateway to information and interaction.

8.6 Website Integration

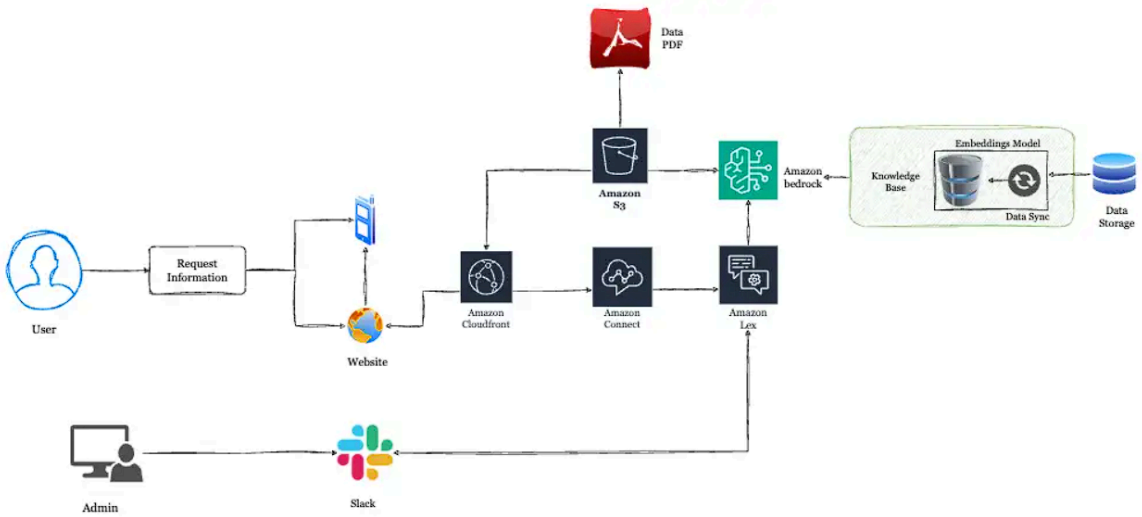
In order to see the Amazon Connect Chatbot on our website, we are using a website hosted on our Amazon S3 bucket. To deploy a chatbot on a website using Amazon Connect's Communication Widget, a series of methodical steps are employed to ensure seamless integration and functionality. Initially, the Communication Widget's configuration tool within Amazon Connect is utilized to define the chatbot's appearance and features, such as enabling call and chat functionalities.

Upon completion of this configuration, Amazon Connect generates an index.html script tailored to the specified parameters. This script is subsequently copied into an index.html document, which is then uploaded to an Amazon S3 bucket designated for static website hosting. By enabling static website hosting on the S3 bucket, the index.html document becomes accessible as a fully operational website.

This approach leverages Amazon S3's robust infrastructure to provide a reliable hosting environment, allowing users to interact with the chatbot seamlessly through their web browsers. This method efficiently facilitates the deployment of interactive chat solutions on any website.

9 Use Case Diagram

Link to the Diagram:
https://drive.google.com/file/d/1QelIJej4UvgKsiLPZWwZkh08h2DyCb4G/view



10 Future Scope and Enhancements

In the future, when a person interacts with the bot on the website, and then later would want to send a message to a person from OEV Online Dienste GmbH, this would be possible with the help of a lambda function. To send messages from your Amazon Lex chatbot to a Slack channel on demand (meaning only during certain events), you need to integrate both services using AWS Lambda.

Enabling communication between an Amazon Lex chatbot and a Slack channel on demand requires certain prerequisites and implementation steps. The primary prerequisites include having a fully functional Amazon Lex bot, access to a Slack workspace with the ability to create and configure a Slack App, and an AWS Lambda function to facilitate the interaction between Lex and Slack.

The first step in the implementation process involves creating a Slack App. This can be done by visiting the Slack API website and following the prompts to create a new application. During the configuration of this app, it is essential to grant the necessary permissions, such as the chat:write scope, which allows the app to send messages to Slack channels. Once the app is configured, it needs to be installed into the Slack workspace, at which point an OAuth Access Token is generated. This token is crucial for authenticating API requests from the Lambda function to Slack.

Following the creation of the Slack App, an AWS Lambda function must be set up. This can be accomplished by visiting the Slack API website and following the prompts to create a new application. The function needs to be configured with appropriate permissions to allow it to interact with the Lex bot. The core of this configuration is the function's code, which handles the extraction of messages from Lex events and their subsequent transmission to Slack via the Slack Webhook URL.

The Lambda function code can be written in Python, and it typically involves importing necessary libraries such as json and urllib3. The function captures the message from the Lex event and constructs a payload to be sent to Slack. This payload is then posted to the Slack Webhook URL using an HTTP POST request. To streamline the deployment and enhance security, the Webhook URL should be stored as an environment variable in the Lambda function's settings.

Once the Lambda function is operational, the Lex bot must be configured to invoke this function. This is done within the Lex bot's configuration settings, specifically under the intent that should trigger the Slack message. The fulfillment section of the intent should be set to use the Lambda function created earlier.

11 References

1. "Amazon Lex | Eu-Central-1." Accessed June 3, 2024. https://eu-central-1.console.aws.amazon.com/lexv2/home?region=eu-central-1#bot/3H17VOJVSP.

2. Live Coding with AWS: Building Call Centres with Amazon Connect, 2017. https://www.youtube.com/watch?v=WESBp_Sg7cg.

3. "Slack API: Applications | HTW Nico Slack." Accessed June 3, 2024. https://api.slack.com/apps/A075Q0HNKPX?created=1.

4. "SlackbotMainFlow - Amazon Connect - Edit Flow." Accessed June 3, 2024. https://projecthtwwithnico.my.connect.aws/contact-flows/edit?id=arn%3Aaws%3Aconnect%3Aeu-central-1%3A590183876879%3Ainstance%2Fbaf6e6e6-c770-4c87-8efe-a022e2385be8%2Fcontact-flow%2Fcd8f8ee1-f296-4eb7-865c-d10a279ef5eb.

5. Pradip Nichite. (2022, July 22). *How to Make a Chatbot Using Amazon Lex and AWS Lambda (Python) | Conversational AI Part 2* [Video]. YouTube. https://www.youtube.com/watch?v=W6T-RFei6SY

6. PradipNichite. (n.d.). *Youtube-Tutorials/Amazon_Lex/Part2.py at main · PradipNichite/Youtube-Tutorials*. GitHub. https://github.com/PradipNichite/Youtube-Tutorials/blob/main/Amazon_Lex/Part2.py

7. Sameer-Goel. (n.d.). *Smart-HR-Bot-with-SentimentAnalysis-Integrated-with-SMS-and-other-services/Use Case 1. Greet the user with welcome message and HR menu options - Lambda Fulfilment.md at master · sameer-goel/Smart-HR-Bot-with-SentimentAnalysis-Integrated-with-SMS-and-other-services*. GitHub. https://github.com/sameer-goel/Smart-HR-Bot-with-SentimentAnalysis-Integrated-with-SMS-and-other-services/blob/master/Use%20Case%201.%20Greet%20the%20user%20with%20welcome%20message%20and%20HR%20menu%20options%20-%20Lambda%20Fulfilment.md

8. AWS Developers. (2022, April 1). *Create an Amazon Lex-based Question and Answer Chatbot with the AWS QnA Bot Solution* [Video]. YouTube. https://www.youtube.com/watch?v=Gyi85CJhBP0

9. *AWS Amplify MissingRequiredParameter userId error*. (n.d.). Stack Overflow. https://stackoverflow.com/questions/54148545/aws-amplify-missingrequiredparameter-userid-error

10. *FulfillmentCodeHook doesn't have an associated Lambda Function AWS*. (n.d.). Stack Overflow. https://stackoverflow.com/questions/76342541/fulfillmentcodehook-doesnt-have-an-associated-lambda-function-aws

11. *Amazon Lex: How It Works - Amazon Lex V1*. (n.d.). https://docs.aws.amazon.com/lex/latest/dg/how-it-works.html

12. Amazon Web Services. (2024, April 16). *Gen AI ChatBot – How to integrate Amazon Lex and Knowledge bases for Amazon Bedrock* [Video]. YouTube. https://www.youtube.com/watch?v=QEqZILuy8q8

13. Campus, O. D. (n.d.). *OEV Digital Campus 360 - Blog*. OEV Online Dienste GmbH. https://digitalcampus360.de/blog

14. *OEV Online Dienste GmbH - ein Unternehmen der S-Finanzgruppe: Information and Updates | XING*. (n.d.). XING. https://www.xing.com/pages/oev

15. *Integrating an Amazon Lex Bot with Slack - Amazon Lex V1*. (n.d.). https://docs.aws.amazon.com/lex/latest/dg/slack-bot-association.html

16. Amazon Web Services. (2024b, April 16). *Gen AI ChatBot – How to integrate Amazon Lex and Knowledge bases for Amazon Bedrock* [Video]. YouTube. https://www.youtube.com/watch?v=QEqZILuy8q8

17. AWS Partner Success. (2023, July 26). *Amazon Connect : Chat Widget Customization & Website Integration* [Video]. YouTube. https://www.youtube.com/watch?v=jvJEo4aEEOI

18. Amazon Web Services. (2019, April 2). *Introduction to Amazon CloudFront* [Video]. YouTube. https://www.youtube.com/watch?v=AT-nHW3_SVI

19. *Set up outbound caller ID - Amazon Connect*. (n.d.). https://docs.aws.amazon.com/connect/latest/adminguide/queues-callerid.html