

PROFESSIONAL TRAINING REPORT

at

**Sathyabama Institute of Science and Technology
(Deemed to be University)**

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Technology Degree in Information Technology

By

**KIRTHIGA V
(40120046)**



**DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA**

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI, TAMILNADU – 600119**

April 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KIRTHIGA V (Reg. No. 40120046)** who carried out the project as a "Pizza Deliver Chat-bot using AWS Lex" under our supervision from January 2023 to April 2023.

V. Nirmalrani
27/04/2023
Internal Guide

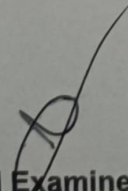
Dr. V.Nirmalrani

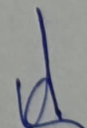
R.S. Subhashini
27/4/23

Head of the Department

Dr. R. Subhashini, M.E., Ph.D.,

Submitted for Viva voce Examination held on 28.04.23


Internal Examiner


External Examiner

DECLARATION

I, KIRTHIGA V (Reg. No. 40120046) hereby declare that the Project Report entitled "Pizza Delivery chat-bot using AWS Lex" is done by me under the guidance of Dr. V.NIRMALRANI and is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in **Information Technology**.

DATE: 28.04.2023 .

PLACE: Chennai

V. Kirthiga

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T Sasikala, M.E., Ph.D.**, Dean, School of Computing **Dr. R Subhashini, M.E., Ph.D.**, Head of the Department of Information Technology for providing me with necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. R Subhashini, M.E., Ph.D.** for her valuable guidance, suggestions and constant encouragement that paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Information Technology who were helpful in many ways for the completion of the project.

ABSTRACT

Because of breakthroughs in machine learning and deep learning, which are causing a change in every industry area and managing various types of activities better than people. The majority of monotonous jobs that were formerly performed by humans are now replaced by AI. Every firm is aiming to replace the least skilled labour with AI robots that can do comparable tasks more efficiently, especially when it comes to chat-bots. A chat-bot is computer software that mimics human interaction by using voice instructions, text dialogues, or both. Chatbots are being employed to address consumer concerns or problems in food delivery app businesses such as Zomato and Swiggy, but are chatbots truly useful in that business model? This business model's target customers are those who don't have time to go outside to obtain food, prefer convenience at home, or are unwilling to endure discomfort, thus their concerns should be resolved in the most convenient way possible. To fulfil the user's request, a chatbot is employed. It is critical for the chatbot to plan how to carry out the task that the user has asked. New tools are available now to create and deploy chatbots; Amazon Lex by AWS is one of them. This project focuses on creating a Pizza Ordering Chatbot using Amazon Lex to help the user order pizza.

TABLE OF CONTENT

Chapter No	TITLE	Page No.
	ABSTRACT	
	LIST OF FIGURES	
1	INTRODUCTION	1
	2.1 AIM	1
	2.1.1 SCOPE	1
2	LITERATURE SURVEY	3
3	AMAZON LEX OVERVIEW	6
	4.1 CHATBOT	6
	4.1.1 AMAZON WEB SERVICE	7
	4.3 OBJECTIVES	8
	4.3.1 NEED FOR AMAZON	11
	LEX	
4	METHODOLOGY	12
	5.1 SIGN IN AWS CONSOLE	13
	5.2 INTENT OF PIZZA ORDERING	14
	5.3 INTENT DETAILS	14
	5.4 CREATION OF INTENT	14
	5.5 INITIAL RESPONSES	15
	5.6 CREATION SLOT TYPE	15
	5.5 PROMPT FOR SLOT	16
	5.8 CLOSING RESPONSE	16
	5.9 BUILDING THE CHATBOT	17
5	RESULT AND DISCUSSION	18
6	SUMMARY AND CONCLUSION	21
	6.1 FUTURE ENHANCEMENT	
7	REFERENCES	24

LIST OF FIGURES

FIGURE NO	FIGURE DESCRIPTION	PAGE NO
4.1	ADVENT OF CONVERSATIONAL INTERACTIONS	5
4.2	ARCHITECTURE OF AMAZON LEX	7
4.3	FLOW OF CONVERSION	10
4.4	WORKFLOW OF LEX	11
5.1	SIGN IN CONSOLE OF AWS	13
5.2	INTENT OF PIZZA ORDERING	14
5.3	INTENT DETAILS	14
5.4	CREATION OF INTENT	15
5.5	INITIAL RESPONSES	15
5.6	CREATION OF SLOT TYPE	16
5.7	PROMPT FOR SLOTS	16
5.8	CLOSING RESPONSES	17
5.9	BUILDING THE CHATBOT	17
6.1	CHATBOT CONVERSATION	18

CHAPTER 1

INTRODUCTION

Amazon Lex is a service provided by Amazon Web Services (AWS) that allows developers to build conversational interfaces, also known as chatbots, for their applications. It uses natural language understanding (NLU) and automatic speech recognition (ASR) technologies to interpret and respond to user inputs in a conversational manner. Amazon Lex is a service that allows developers to build conversational interfaces using natural language processing and speech recognition technology. The service can be used to create chatbots, voicebots, and virtual assistants that can interact with users in a conversational manner. In this project, we explore the capabilities of Amazon Lex and build a chatbot that can answer users' questions about a fictional product called "Cool gadgets."

AIM AND SCOPE OF THE PRESENT INVESTIGATION

2.1 AIM

To create pizza delivery chatbot using Aws lex.

2.1.2 SCOPE

1. Natural Language Processing: AWS Lex provides a natural language understanding (NLU) service that enables the chatbot to understand and interpret customer requests and respond accordingly. This allows for a more conversational experience for the user.
2. Integration with other AWS Services: AWS Lex can be integrated with other AWS services such as Amazon Connect, Amazon S3, Amazon DynamoDB, and Amazon Lambda, allowing for seamless integration with existing systems and services.
3. Multi-channel Support: AWS Lex supports multiple communication channels, including text, voice, and messaging platforms, making it easy to reach customers through their preferred communication channel.
4. Scalability: AWS Lex is a highly scalable service that can handle large volumes of customer requests without any disruption in service.

5. Analytics: AWS Lex provides built-in analytics that allows businesses to track user interactions, understand user behaviour, and optimize the chatbot's performance over time.
6. Security: AWS Lex provides enterprise-grade security and compliance, including encryption, access control, and audit trails, ensuring that customer data is kept secure and confidential.

Overall, the scope of using AWS Lex chatbot is extensive, and it can help businesses to create an intelligent and scalable conversational experience for their customers, while also providing valuable insights into customer behavior and preferences.

Chapter 2

LITERATURE SURVEY

M.Hariyadharshini et al. [3] designed a conversational chatbot that can interact with the users by addressing their services e.g., ordering a pizza. The configuration is added to the bot which can understand the user's requirement. To enable a serverless connection the Lambda service was utilized. Similarly, the Amazon RDS service was used to build backend database for the system. Various database engines are provided by RDS like Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle, and Microsoft SQL Server. Here MySQL workbench was chosen. The virtual console of the database helps the administrators to efficiently manage the database. The system has proven its error handling capacity by answering irrelevant questions of the user.

Poongothai.M. et al. [4] developed a chatbot that is personalized conversational assistant that can help the students to raise questions and clear their doubts on ongoing as well as completed IoT projects. Few features like auto scaling, real time notifications and quick response are advantages of this system. Later, this chatbot was integrated into their laboratory system.

Pinaki Chakraborty et al. [5] conducted a survey among undergraduate students about their opinion on online education during the COVID-19 pandemic. On receiving responses from 358 students, it was observed that the students preferred physical education and learning through MOOC platforms rather than online education. Yet, the results also indicate that the faculty had improved their teaching techniques about online learning. Jayakumar et al. [6] studied the problems faced by students in learning a new programming language and developed a chat bot that teaches students languages like java, cpp, python and conducted tests at each level of the learning path. Naeun Lee et al. [7] built a custom bot that creates personalized news based on the personal preference of the user. They have used concepts like text segmentation and custom tagging for the construction of the system so that the system can cover a broad field of spectrum.

Prashanth S et al. [8] built a college management service by using chat bot. The bot helps all users with their needs, for example leading the way to library, and providing assistance to the day to day needs on the college premises. Nitiraj Singh Sandu et al. [9] studied the adoption of AI-Chat bots to enhance the student learning experience in higher education, they spoke about how a chat bot can improve the productivity, learning, assistance and fill the communication gap and how the ai chat bot can be incorporated as a platform for learning.

Winkler et al. [10] emphasized the potential and impact the chat bots can have in learning process of the student and how these chat bots are helping the users by providing the assistance. Pedro Tamayo et al. [11] built a chat bot which is an assistant for distance learning i.e., EconBot by explaining how a chat bot can be simulated as teaching assistant to all the students who are attending college through distance learning program and how the bot improves the learning and helps them resolve the technical aspect of the subject.

Fabio Clariza et al. [12] spoke about how chat bot can act as support system in the education system and how they have used natural language processing techniques and how the right set of answers are important to answer the questions of the students. Gyorgy Molnar et al. [13] spoke about the role of chat bots in formal education and how the technology has evolved and used widely in the industry and in educational institutions and benefits of leveraging the technology.

A S S K Sreeharsha et al. [14] designed a chatbot system that can help people with hotel reservations using Amazon Lex and Facebook Messenger. Here, amazon lex was used for bot's configuration with utterances and Lambda function was used to validate the responses. Advantage of this system is providing real time notification system and login using the Facebook page.

Shivangi Viral Thakker et al. [15] proposed a holistic approach by performing data analysis through a survey of 364 engineering students to understand the bottlenecks in online learning. The metric was set out on a 5-point Likert scale where the answers were ranked from 2.81 to 3.46. The findings showed that Google Meet was the most used platform whereas Microsoft Teams was most preferred platform according to the students.

Yet, there are few limitations for this analysis, where the research was only limited to students in the urban areas which is prone to cause a lot of geographical and educational bias.

CHAPTER 3

AMAZON LEX OVERVIEW

4.1 Chatbot

A chatbot is a computer program that conducts a conversation in natural language via text or speech, understands the intent of the user and sends a response based on business rules and data of the organization. One of the first chatbots was ELIZA. It was also an early test case for the Turing Test, which determines if a machine can demonstrate cognition that is comparable to, or indistinguishable from, that of a person. The computer used "pattern matching" and replacement methods to provide prepared replies, giving early users the impression that they were speaking with someone who understood their input. The scripts in the software limited the program's capabilities. Joseph Weizenbaum, a researcher at MIT's Artificial Intelligence Laboratory, created an early natural language processing algorithm in the mid-1960s. It was allegedly made to show how shallow human-computer connections were at the time. Humans, on the other hand, found it highly entertaining when it was put on computers. In today's world, chatbots have many applications such as online shopping, ticket booking, news reports, food ordering, etc.



1st Generation:
Punch card and memory registration



2nd Generation:
Pointers & Sliders



3rd Generation:
Conversational Interface

Fig:4.1 Advent of conversational interactions

Amazon Lex is a robust discussion tool that enables developers to incorporate interactive experiences into new and existing applications by integrating voice and text interfaces. Alexa is powered by this behind-the-scenes service. Amazon Lex is a service that allows you to create these conversational user interfaces.

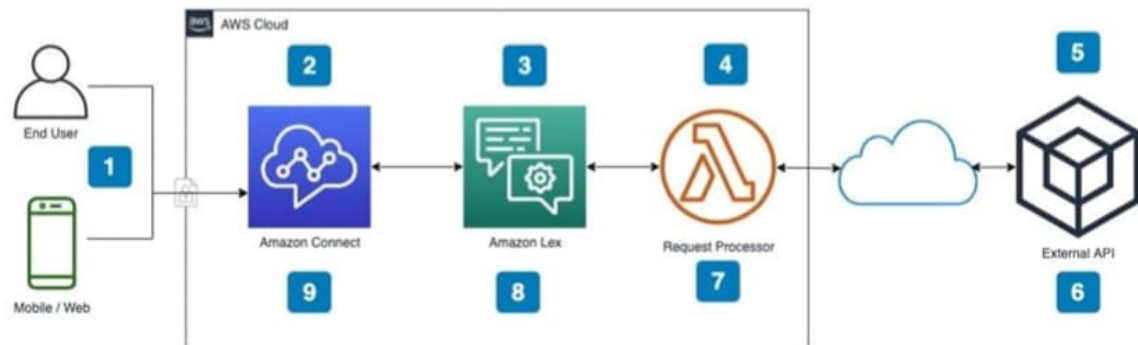


Fig:4.2 Architecture of Amazon lex

4.2 Amazon Web Services

AWS (Amazon Web Services) is a sophisticated, developing cloud computing platform offered by Amazon that comprises infrastructure as a service (IaaS), platform as a service (PaaS), and integrated software as a service (SaaS) solution. AWS services can offer organisation tools such as compute power, database storage and content delivery services. AWS was founded in 2006 as an extension of the internal operations established by Amazon.com to manage its online sales activities. AWS was among the first firms to provide a pay-as-you-go cloud computing model, which grows to give users with computation, storage, or networking as required.

AWS provides a wide range of tools and solutions for businesses and software engineers that may be utilised in cloud services in up to 200 countries and territories. AWS services are available to government agencies, educational institutions, charities, and commercial organisations.

4.2.1 How does AWS work?

1) AWS is separated into different services; each can be configured in different ways based on the user's needs. For an AWS service, users are allowed to access configuration choices as well as specific server mappings.

2) More than 100 services comprise the Amazon Web Services portfolio, including those for compute, databases, infrastructure management, application development and security. These services, by category, include:

- a) Compute
- b) Storage databases
- c) Data management
- d) Migration
- e) Hybrid cloud
- f) Networking
- g) Development tools
- h) Management
- i) Monitoring
- j) Security
- k) Governance
- l) Big data management
- m) Analytics
- n) Artificial intelligence (AI)
- o) Mobile development
- p) Messages and notification

4.3 Objectives

- Simplify the pizza ordering process: The chatbot should be designed to make it easier and faster for customers to place their pizza orders without having to navigate a complex website or menu.
- Improve customer experience: The chatbot should be able to provide prompt and accurate responses to customers' questions and concerns, making the ordering process more convenient and enjoyable.
- Increase sales: The chatbot can upsell customers by suggesting additional toppings or menu items that complement their orders. Provide personalized recommendations: The chatbot can learn customers' preferences over time and suggest pizzas and toppings that they are likely to enjoy based on their previous orders.
- Enhance customer loyalty: The chatbot can provide exclusive deals and promotions to loyal customers, helping to foster a long-term relationship between the customer and the pizza delivery service.

4.3.1 Preview of flow of conversation

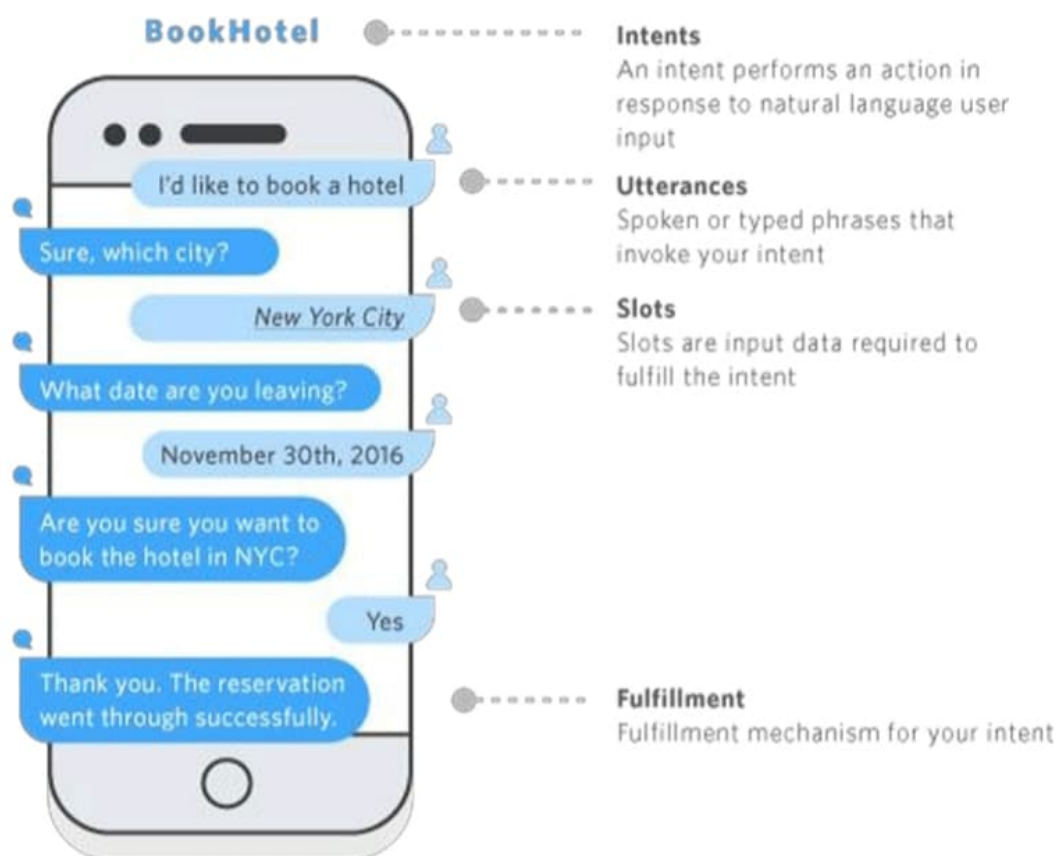


Fig:4.3 Flow of conversation

AWS Lex provides automatic speech recognition to convert audio input into text, enabling users to interact with applications through voice commands.

4.4 The Need for Amazon Lex

AWS Lex is a popular service that helps businesses build conversational interfaces using natural language processing (NLP). There are several reasons why businesses might need AWS Lex:

- **Improved customer experience:** Conversational interfaces powered by AWS Lex can provide a seamless and personalized experience to users. They can understand natural language queries and respond accordingly, which can lead to improved customer satisfaction and engagement.

- Automation of customer support: AWS Lex enables the creation of chatbots that can handle common customer inquiries and provide self-service support. This can help businesses reduce support costs and improve response times by automating repetitive tasks.
- Enhanced user engagement: Chatbots built with AWS Lex can engage users in interactive conversations, providing information, recommendations, and assistance. This can result in increased user engagement and retention, leading to better business outcomes.
- Scalability and flexibility: AWS Lex is a cloud-based service, which means it can scale automatically to handle varying levels of user interactions. It can be integrated with other AWS services and customized to suit specific business requirements, making it flexible and adaptable to different use cases.

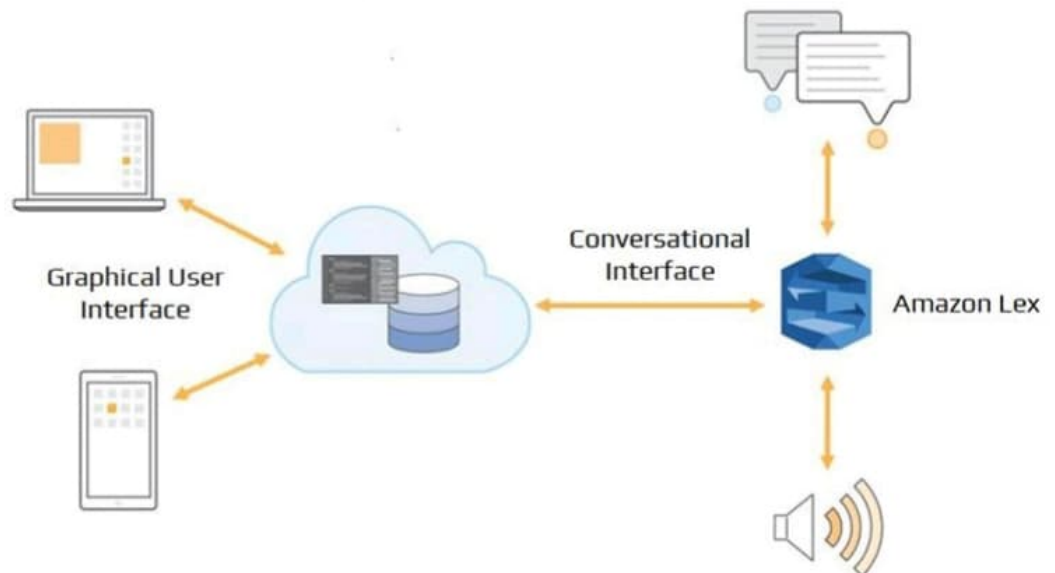


Fig:4.4 Workflow of lex

CHAPTER 4

METHODOLOGY

When it comes to building a project using Amazon Lex, there are several steps you follow to ensure a successful outcome. Here is a methodology you can use:

1. Define the project scope: Clearly define the project's objectives, goals, and scope. Identify the key stakeholders and their requirements.
2. Design the conversation flow: Determine the conversation flow that your bot will follow. Identify the various intents and slots that your bot will handle. Map out the conversation paths and the various responses that your bot will provide.
3. Build the bot: Use the Amazon Lex console to build your bot. Define the intents and slots that you identified in the previous step. Create sample utterances and slot values. Test the bot to ensure that it is functioning correctly.
4. Integrate with other services: Amazon Lex can integrate with other AWS services, such as Lambda, to perform actions based on user input. Determine which services you need to integrate with and set them up.
5. Test and refine: Test the bot thoroughly to identify any issues. Refine the conversation flow, intents, and slots as necessary.
6. Deploy the bot: Once you are satisfied with the bot's performance, deploy it to your desired platform, such as Facebook Messenger, Slack, or your own website.

Monitor and improve: Monitor the bot's performance regularly and make improvements as needed. Analyse user feedback and adjust the conversation flow and responses accordingly.

The below mentioned are the methodologies used to enter lex interface,



Sign in

☒ Root user

Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☐ IAM user

User within an account that performs daily tasks. [Learn more](#)

Root user email address

senthilgunal900@gmail.com

Next

By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

— New to AWS? —

Create a new AWS account



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

English ▾

Fig:5.1 Sign in console of AWS

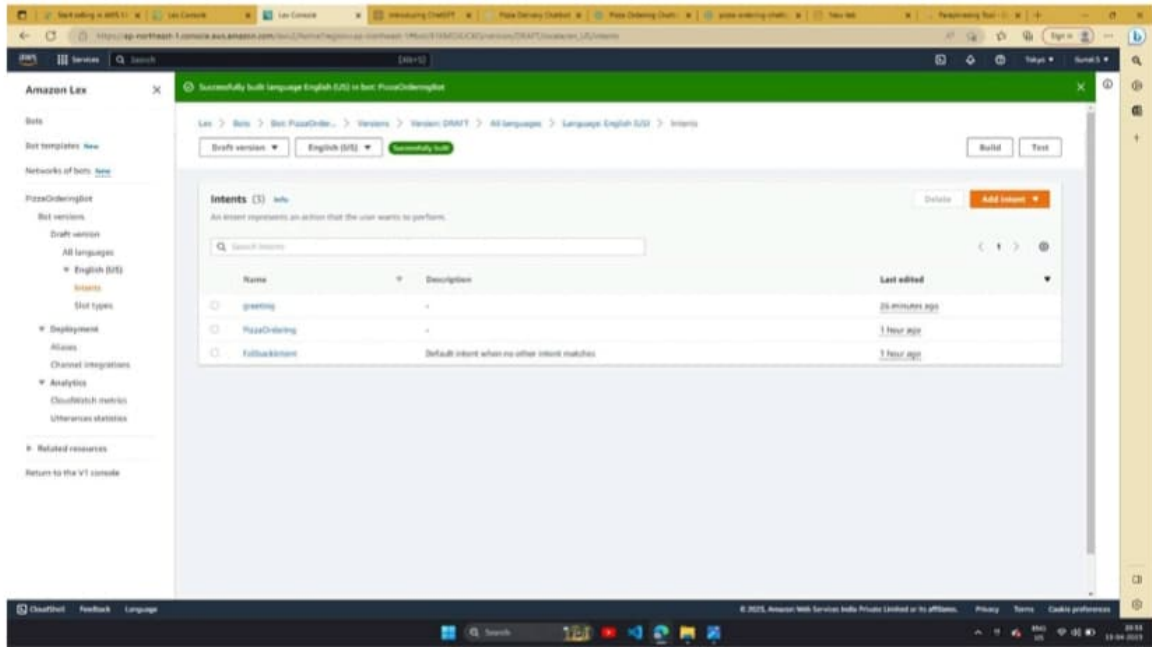


Fig:5.2 Intent of Pizza ordering

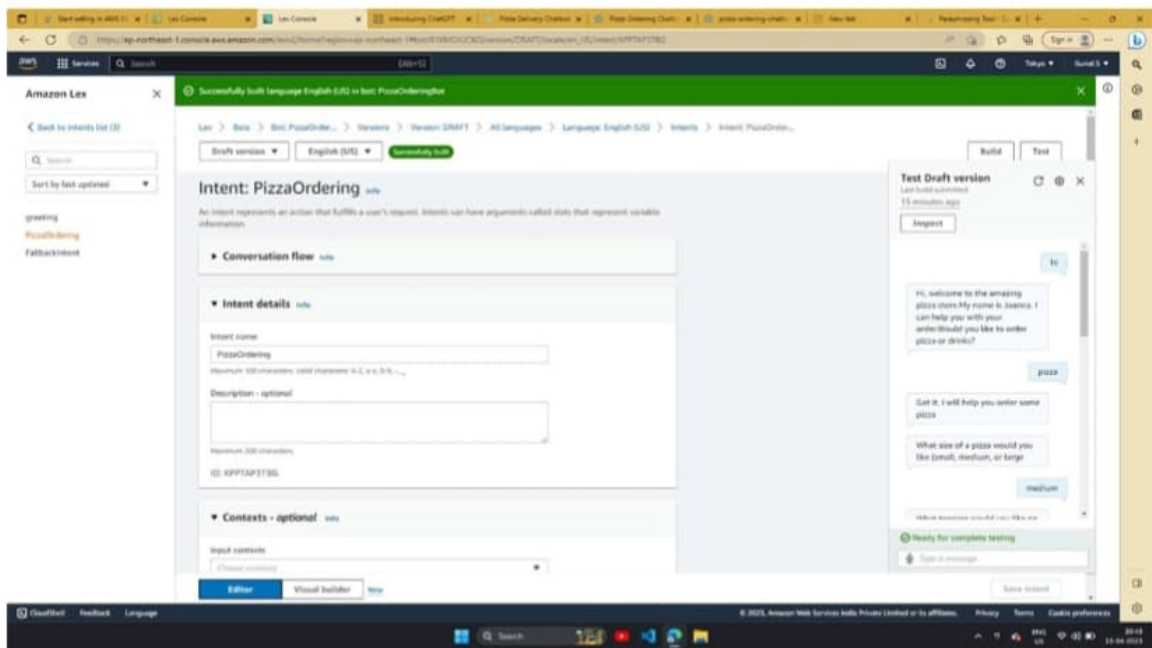


Fig:5.3 Intent details and contents to be provided

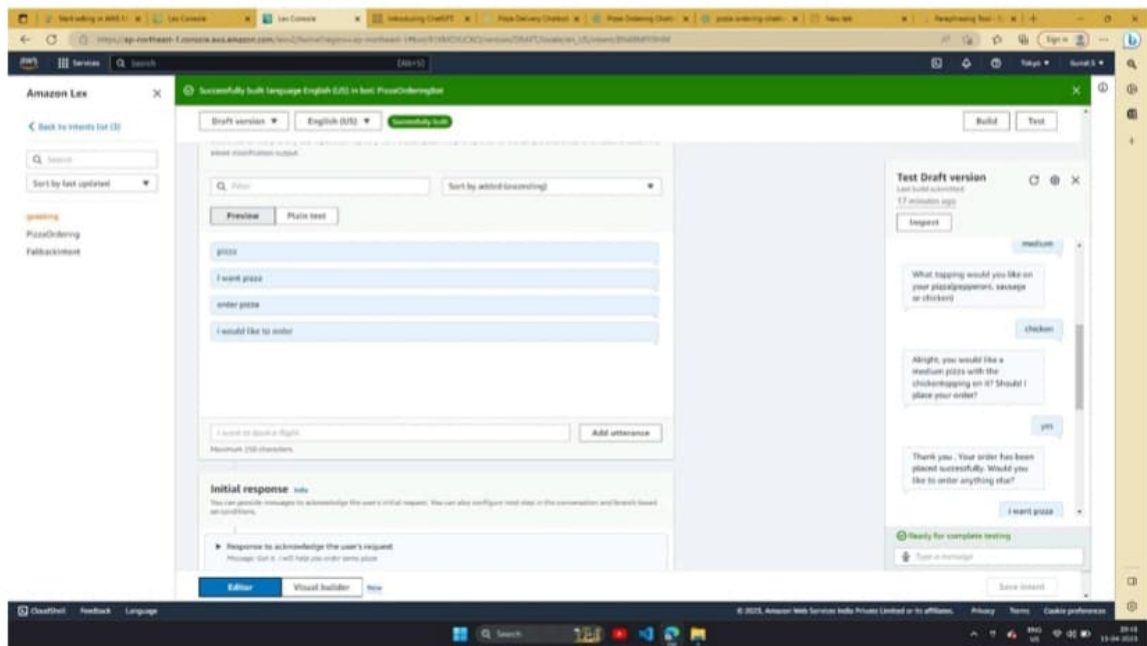


Fig:5.4 Creation of intent

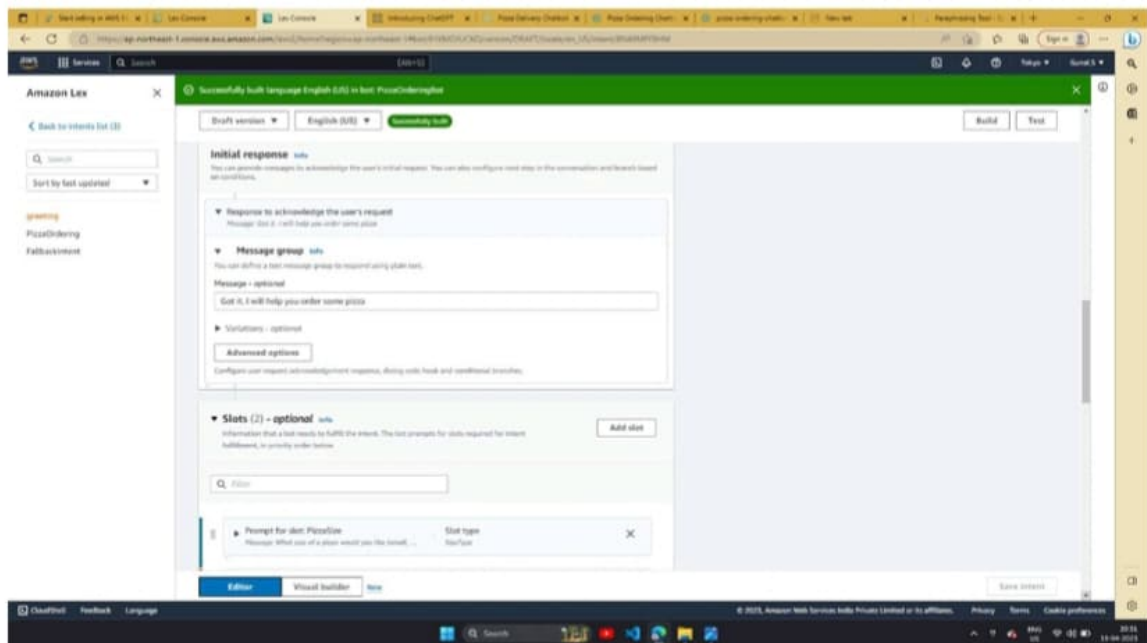


Fig:5.5 Initial Responses

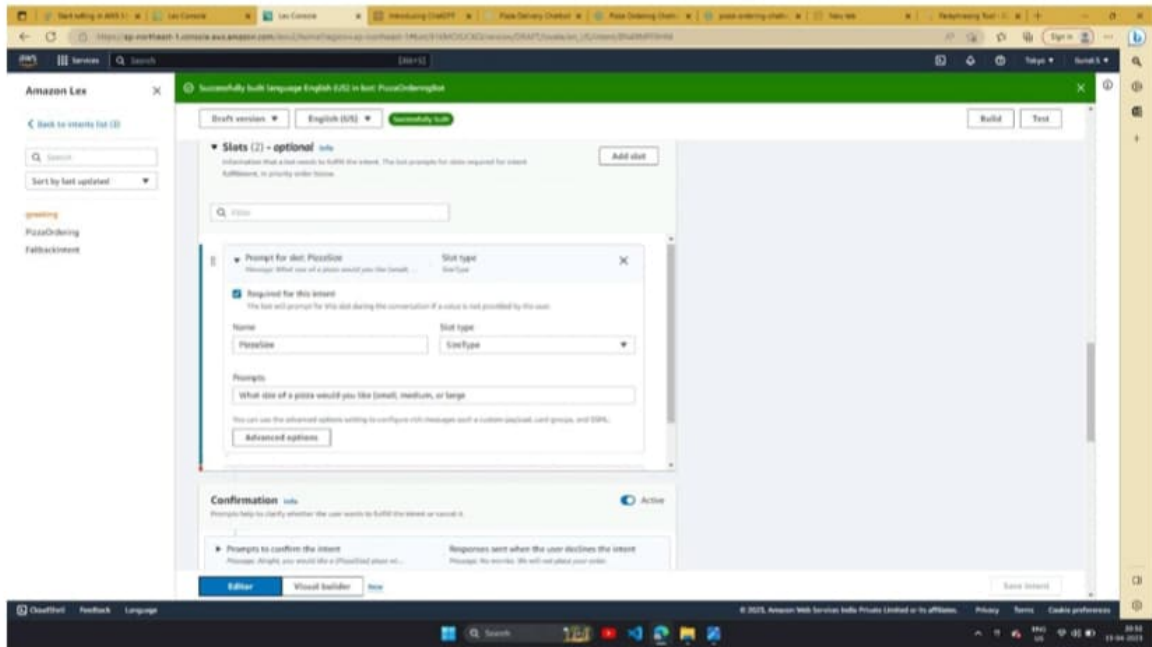


Fig:5.6 Creating slot type

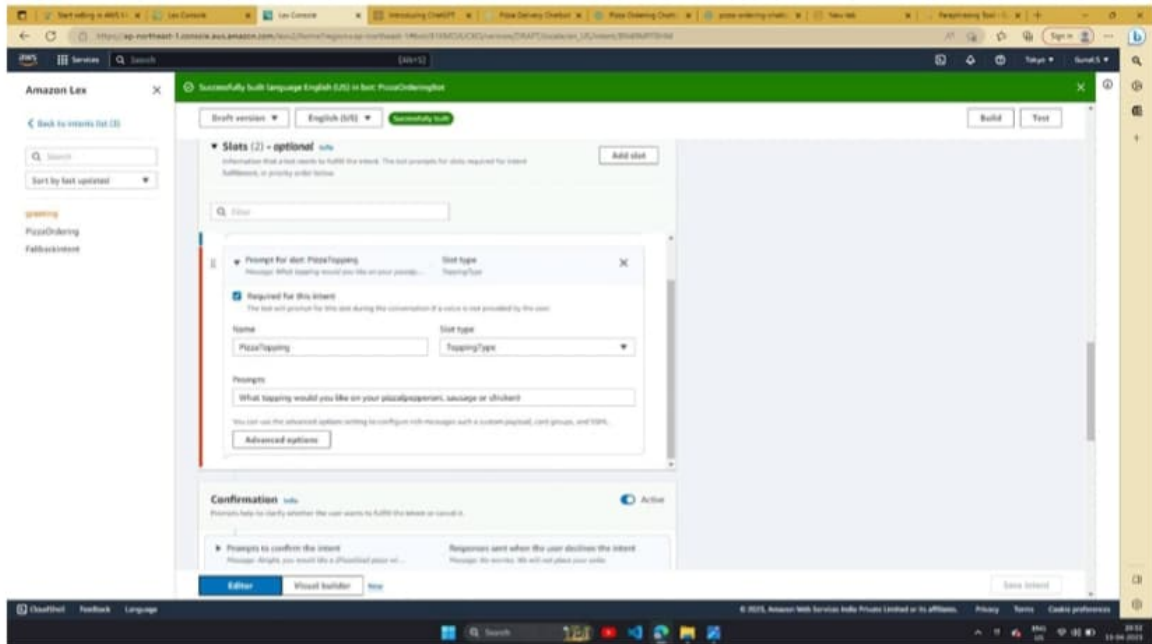


Fig:5.7 Prompt for slots

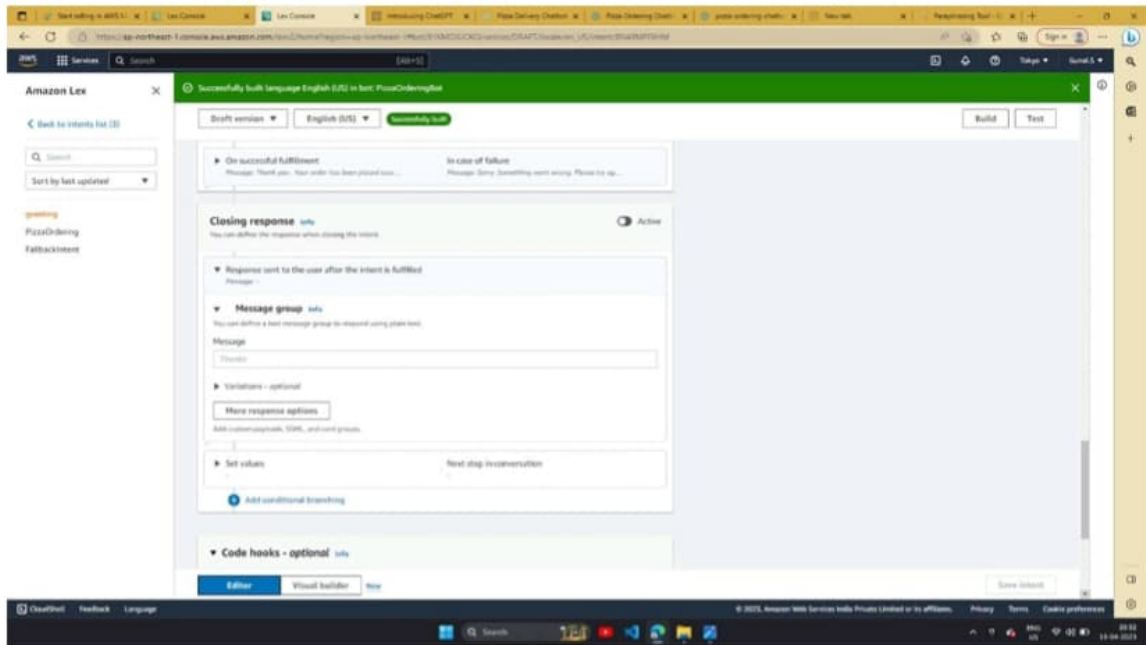


Fig:5.8 Closing response

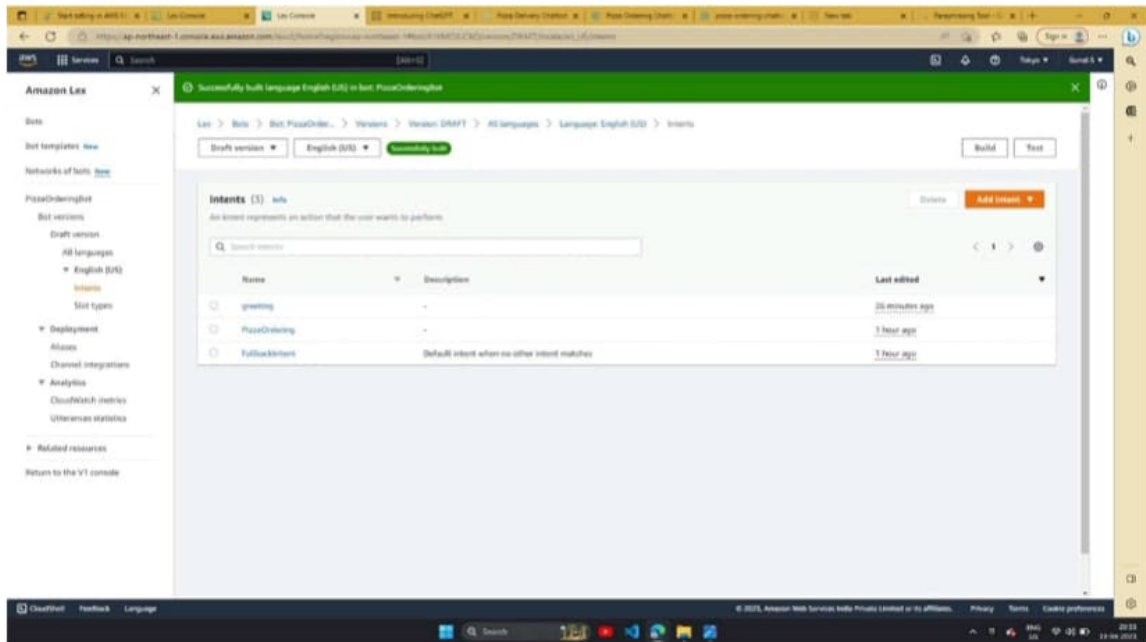


Fig:5.9 Building the chatbot

CHAPTER 5

RESULT AND DISCUSSION

The following figures are the responses given for the user based on the different queries it was asked.

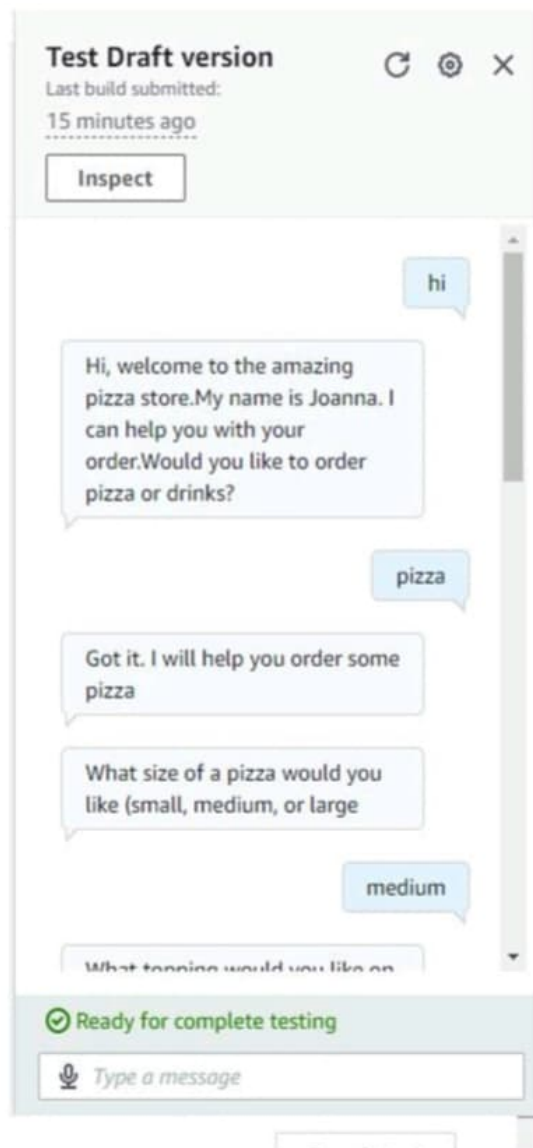


Fig:6.1 Chatbot conversation

A successful pizza delivery chatbot should be able to handle a wide range of user queries and requests, such as taking orders, answering frequently asked questions, providing information about menu items and prices, and handling payments and

delivery options. It should also be able to handle complex user requests, such as customizing orders or accommodating dietary restrictions.

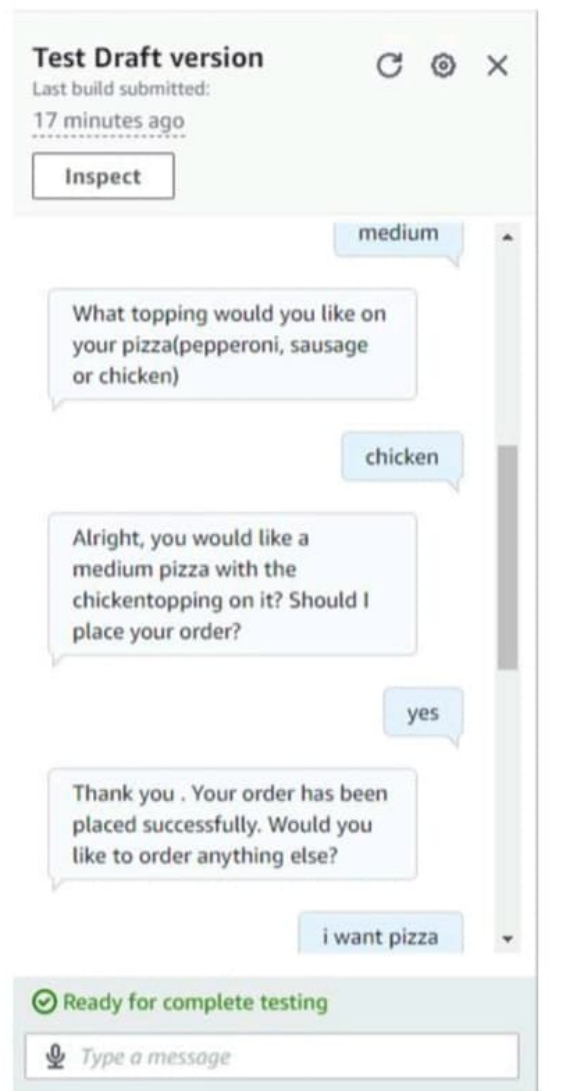


Fig:6.2.1 Conversation

Additionally, the chatbot should be able to provide a seamless user experience by being user-friendly and easy to navigate. This can be achieved by incorporating natural language processing (NLP) techniques, which enable the chatbot to understand and interpret user input and respond in a natural, human-like manner.

Another important consideration is the scalability of the chatbot. As the number of users increases, the chatbot should be able to handle multiple requests simultaneously without experiencing significant performance issues or downtime.

In terms of potential implications, a successful pizza delivery chatbot can lead to increased customer satisfaction, improved operational efficiency, and increased revenue for the business. By providing a convenient and efficient way for customers to order pizza, the chatbot can help to reduce wait times, minimize order errors, and increase the overall quality of service.

Overall, the success of your pizza delivery chatbot project will depend on a variety of factors, such as its features, user experience, scalability, and ability to meet the needs of your target audience. With the right approach and development team, however, a successful pizza delivery chatbot can provide significant benefits to both the business and its customers.

CHAPTER 6

SUMMARY AND CONCLUSION

Amazon Lex is a powerful tool for building conversational interfaces, allowing users to interact with applications using natural language. The platform offers a range of features and capabilities, including natural language understanding, intent recognition, and integration with other AWS services. When undertaking a project using Amazon Lex, there are several key considerations to keep in mind. First, it is important to carefully define the scope of the project and identify the specific use case or business problem that the conversational interface will address. This will help guide the design and development process and ensure that the end product meets the needs of the intended users.

Another important consideration is the design of the conversational flow and dialogue management. It is important to craft a natural and intuitive conversation that guides the user towards their desired outcome while handling any errors or misunderstandings that may arise. Integration with other AWS services such as Lambda, S3, and DynamoDB can greatly enhance the capabilities of an Amazon Lex bot, allowing for more complex and dynamic interactions. It is also important to consider security and privacy concerns, such as ensuring that user data is protected and that the bot is not vulnerable to attacks or abuse.

In conclusion, Amazon Lex is a powerful tool for building conversational interfaces, allowing developers to easily create chatbots, voice assistants, and other interactive applications. It offers natural language processing capabilities that enable bots to understand user inputs and respond appropriately. When designing a project using Amazon Lex, it is essential to have a clear understanding of the problem you are trying to solve and the user's needs. You should also consider the types of interactions your bot will have and the channels through which users will interact with it.

Additionally, it is important to carefully plan and design the conversation flow and user interface to ensure a smooth and efficient user experience. Testing and iterating on your bot's performance is also crucial to continually improve its functionality and accuracy.

Overall, Amazon Lex is a powerful tool for creating conversational interfaces that can improve user engagement and satisfaction in a variety of applications. With careful planning and execution, it can be an effective solution for a wide range of use cases.

7.1 Future enhancement for project using amazon lex

If you are working on a project using Amazon Lex, there are several future work options that you can explore to enhance your project's functionality and improve the user experience. Here are a few suggestions:

Multi-language support: Consider adding support for multiple languages to your Amazon Lex chatbot. This will enable users who speak different languages to interact with the chatbot in their preferred language, increasing the chatbot's reach and accessibility.

Integration with other services: Amazon Lex integrates with a wide range of AWS services, such as Amazon Lambda, Amazon Polly, and Amazon Connect. Consider integrating your chatbot with these services to provide additional functionality, such as text-to-speech, voice recognition, and live agent transfer.

Natural language processing: Explore ways to improve the chatbot's natural language processing capabilities. This could include adding more training data, refining the chatbot's dialogue management, or integrating third-party natural language processing services.

Personalization: Consider adding personalization features to your chatbot, such as user-specific preferences, personalized recommendations, and customized responses based on user history and behaviour

Analytics and metrics: Implement analytics and metrics to track user engagement and measure the success of your chatbot. This can help you identify areas for improvement and optimize the chatbot's performance over time.

Improving the User Experience: To make your chatbot more engaging, you can incorporate multimedia elements like images, videos, and audio files. You can also develop more sophisticated conversational flows that offer personalized recommendations, interactive games, or quizzes to users.

Adding Intelligence and Automation: Amazon Lex can be integrated with other Amazon services like AWS Lambda, Amazon SageMaker, and Amazon Comprehend to add intelligence and automation to your bot. For example, you can use Amazon SageMaker to build and deploy machine learning models that help your bot understand user behaviour or use AWS Lambda to automate backend processes like data processing, authorization, and validation.

Overall, the potential for a chatbot project using Amazon Lex is vast, and the future possibilities are limited only by your imagination and resources.

CHAPTER 7

REFERENCES

- [1] G.Daniel, J. Cabot, L. Deruelle, and M. Derras, "Multi-platform chat-bot modeling and deployment with the jarvis framework," in *Advanced Information Systems Engineering (Lecture Notes in Computer Science)*, vol. 11483, P. Giorgini and B. Weber, Eds. Rome, Italy: Springer, Jun. 2019, pp. 177–193, doi: 10.1007/978-3-030-21290-2_12.
- [2] O. Diaz and F. M. Villoria, "Generating blogs out of product catalogues: An MDE approach," *J. Syst. Softw.*, vol. 83, no. 10, pp. 1970–1982, Oct. 2010, doi: 10.1016/j.jss.2010.05.075.
- [3] D. Falessi, N. Juristo, C. Wohlin, B. Turhan, J. Münch, A. Jedlitschka, and M. Oivo, "Empirical software engineering experts on the use of students and professionals in experiments," *Empir Softw. Eng*, vol. 23, no. 1, pp. 452–489, Feb. 2018, doi: 10.1007/s10664-017-9523-3.
- [4] J. Hutchinson, J. Whittle, and M. Rouncefield, "Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure," *Sci. Comput. Program.*, vol. 89, pp. 144–161, Sep. 2014.
- [5] P. Jackson and I. Moulinier, *Natural Language Processing for Online Applications: Text Retrieval, Extraction and Categorization*, vol. 5. Amsterdam, The Netherlands: John Benjamins,
- [6] D. Kavalier, S. Sirovica, V. Hellendoorn, R. Aranovich, and V. Filkov, "Perceived language complexity in GitHub issue discussions and their effect on issue resolution," in *Proc. 32nd IEEE/ACM Int. Conf. Automated Softw. Eng. (ASE)*, Oct. 2017, pp.
- [7] A. Kleppe, *Software Language Engineering: Creating Domain-Specific Language Using Metamodels*. London, U.K.: Pearson, 2008.
- [8] L. C. Klopfenstein, S. Delpriori, S. Malatini, and A. Bogliolo, "The rise of bots: A survey of conversational interfaces, patterns, and paradigms," in *Proc. Conf. Designing Interact. Syst. (DIS)*, 2017, pp. 555–565