

¹ Pankaj Sharma² Dr. Swati V. Chande

Design and Development of Pregnancy- Counselling Alexa Skill: Improving the Ux of Cui Applications



Abstract: - This research paper reflects a larger research effort focused on Application of Usability Engineering Principles to improve User Experience of Conversational User Interface (CUI) Applications. Conversational Artificial Intelligence (CAI) is the enabling technology to create human like interfaces in computers and several other devices. The paper discusses the challenges and concerns faced by pregnant women, highlighting the potential of technology, specifically chatbots, to assist and alleviate anxiety during pregnancy. This research paper outlines the design of a chatbot-based pregnancy companion aimed at addressing common concerns related to pregnancy and providing a tool for tracking and resolving queries through a new customized Alexa Skill.

Keywords: Usability, interfaces, chatbots

I. 1.0 INTRODUCTION

A conversational user interface (CUI) permits users to communicate with software, apps, and websites and chat bots through “intelligent” voice or text-based interface. This is very similar to the way they interact with real life human beings. The continuous growth and development in conversational user interface are the outcome of this continuous quest of making the machine intelligent enough to understand the intent of our commands and instructions. A complex set of algorithms facilitate the realization of these ideas [1].

A lot of research work is needed to improve the user experience of CUI applications [2]. This research work is a step in this direction. For conducting the usability engineering experiments to improve the user experience, we designed a new skill in the Amazon Alexa Ecosystem to provide counselling support to pregnant women. This CUI application was designed to act as a communication companion for the would be mother. This paper provides a brief description of the development environment, architecture and technology components, and the benefits of this new Alexa skill.

Usability Engineering Principles were applied for the continuous refinement of user experience for this skill. We named this skill as “Mother Mate Skill.” It can provide counselling support about the key milestones and important aspects of 40-week long pregnancy journey. It was designed to be an intelligent partner, with all knowledge about child birth, and ready to discuss anything, anytime just at the call of a voice “Open Mother Mate Skill” to any Alexa enabled device.

While working on the improvements in User Experience of Conversational User Interface Applications, there was a persistent need to build a spoken language understanding architecture that could convert human language queries into API calls. Another challenge was to ensure that the architecture goes along with Alexa’s prevailing user experience [3].

This research paper describes the design process and conception of this service architecture. Adding of customized skills permits researchers to experiment on UX of CUI applications without maintaining their own Speech Recognition Algorithms and Natural Language Processing systems. A new skill added to Alexa environment allows the user to define intents and utterances. The user requests that the new skill can handle are termed as intents, and the keywords that users speak to invoke those requests are termed as utterances [4].

II. 2.0 REVIEW OF RELATED WORKS

The research paper “Conversational Chatbot in Healthcare” emphasizes the importance of good health for a happy life and addresses the challenges of obtaining timely doctor consultations [2]. The proposed system seeks to build upon advanced Artificial Intelligence techniques to develop a Health chatbot that functions as an initial diagnostic tool and a repository of medical information. This AI-driven solution aims to reduce healthcare costs

¹Phd, Scholar Computer Science International School of Informatics & Management (ISIM), Jaipur, India

²Professor Computer Science International School of Informatics & Management (ISIM), Jaipur, India

Copyright © JES 2024 on-line : journal.esrgroups.org

by preemptively addressing user concerns before they necessitate direct physician involvement. Through natural language processing (NLP) and machine learning algorithms, the chatbot can analyze user inputs, offer preliminary diagnoses, and provide actionable health information, effectively functioning as an intermediary in the healthcare process[5].

Traditional methods of disseminating health information, particularly in the domain of maternal health, have proven insufficient in delivering personalized and contextually relevant guidance[6]. Recent advancements in Conversational User Interfaces (CUIs) offer a paradigm shift in this domain. Research has demonstrated that CUIs, powered by sophisticated AI models, are capable of delivering personalized counseling and health advice at scale [7]. The user preference for such interfaces is strongly correlated with the system's ability to provide context-aware, realistic responses, highlighting the necessity of AI integration in health communication platforms. Conversational agents embedded within CUI frameworks are essential for the accurate parsing and interpretation of health-related queries, especially in sensitive areas like pregnancy education. These agents utilize domain-specific knowledge bases and probabilistic models to identify and mitigate preconception health risks, thereby serving as critical tools for users with limited medical literacy. Furthermore, the automation capabilities of CUIs enable scalable dissemination of health information, presenting a cost-efficient solution for delivering critical healthcare education to underserved populations, particularly low-income pregnant women [8].

The "Thousand Days" concept, rooted in the Lancet series, underscores the significance of the initial thousand days from conception through early childhood as a determinant of long-term health outcomes for both the mother and child. AI-driven health platforms must account for the unique physiological and psychological stressors that characterize this period [7].

The systems designed to assist pregnant women must integrate data from epidemiological studies, such as those from the US Pregnancy Risk Assessment Monitoring System, which indicate a high prevalence of stress-related incidents among postpartum women, thereby informing the design of responsive and context-aware health interventions [9].

Pregnant women increasingly rely on digital platforms as primary sources of health information, with reported usage rates between 70% and 97%. To optimize the efficacy of AI-driven health tools, developers and healthcare practitioners must gain a granular understanding of the decision-making processes that govern this online behavior. [9]. This requires the application of data mining techniques to analyze user interactions with digital health platforms, enabling the development of more intuitive and responsive CUIs that cater to the specific informational and emotional needs of expectant mothers [10].

Chatbots, operating as AI-driven conversational agents, are pivotal in the transformation of health information dissemination. By employing deep learning models for natural language understanding (NLU) and generation (NLG), these systems can deliver context-sensitive and personalized health guidance. The integration of chatbots into healthcare ecosystems not only enhances the accessibility of medical information but also enables continuous, on-demand support for users. This AI capability is essential for bridging the gap between the growing demand for health information and the limited availability of direct medical consultation, particularly in resource-constrained environments [8] [9]. The functioning of medical chatbots relies on natural language processing, allowing users to submit health-related queries and receive relevant responses, and ultimately enabling individuals to inquire about healthcare matters without physically visiting a hospital [10].

The healthcare sector is grappling with challenges such as a growing demand for services and a shortage of trained professionals, leading to limited accessibility for patients [11]. Overcoming organizational, temporal, and geographical barriers is crucial for healthcare practitioners to provide timely and authentic treatment, especially in cases involving cholesterol, blood pressure, diabetes, and pregnancy [12]. The solution lies in Mobile Health (mHealth) services, which address these challenges by offering authentic healthcare remotely. These advancements contribute to overcoming healthcare challenges and improving accessibility to quality medical assistance.

The key insights derived from review of literature were utilized for identifying and refining the scope of work. There seems to be a unanimous agreement on the usefulness and benefits of conversational applications and

adding skills to existing CUI applications. Medical chatbots usage is on increasing trend but user experience feedback has not been studied adequately. The refinement and improvements based on user feedback is an open research area. Usability Engineering principles can be utilized for iterative feedback and refinement cycle.

Several research papers discuss the growing utilization of personalized conversational agents with natural language user interfaces in healthcare applications. This trend influences the content, structure, and purpose of dialogues between humans and conversational agents [13].

Further research is needed to understand how personalization may be applied to conversational applications in healthcare and the methods employed for its implementation. Notably, conversational technologies, particularly in healthcare, are gaining prominence. The literature review was focused on evaluating technical performance, user experience, and health-related outcomes associated with the use of conversational agents [14]. This review helped in better understanding of the evolving landscape of personalized interactions through chatbots in the healthcare sector.

III. 3.0 ARCHITECTURAL DESIGN, DEVELOPMENT AND DEPLOYMENT.

Amazon Alexa was designed with the vision of enabling ambient computing within the home environment. Ambient computing revolves around seamlessly integrating technology into our surroundings, making it an inherent and natural part of our daily lives. Alexa, Amazon's voice-activated virtual assistant, was engineered with the objective of seamlessly managing various facets of the domestic environment through sophisticated natural language processing (NLP) techniques. By enabling interaction with IoT (Internet of Things) devices via voice commands, Alexa has pioneered the transformation of traditional homes into intelligent, context-aware environments. This evolution epitomizes ambient computing, wherein the technology remains unobtrusive yet ubiquitously responsive. Alexa's capabilities extend beyond rudimentary command execution to encompass complex automation workflows, entertainment system orchestration, and communication facilitation, all through a highly refined Conversational User Interface (CUI) designed to deliver a frictionless user experience.

Developing new capabilities, or "skills," for Alexa involves a methodical process that begins with the formulation of an interaction model—a formal representation of the user-system dialogue. The interaction model is critical as it delineates the user's interaction flow and the corresponding system responses, ensuring that the skill's behavior aligns with user expectations. Utilizing the Alexa Skills Kit (ASK), developers are provided with a suite of APIs, SDKs, and documentation that facilitate rapid prototyping and deployment. This structured approach to skill development allows for the systematic design of user interactions, ensuring consistency, reliability, and scalability within the Alexa ecosystem [15]. The interaction model functions as the architectural blueprint for an Alexa skill, specifying the intents, slots, and utterances that define user interaction. Intents represent the high-level actions that the skill can perform, while slots capture specific data points relevant to those actions. The design of this model requires a deep understanding of natural language understanding (NLU) principles to accurately parse user inputs and map them to the correct system actions. The interaction model must be robust, accounting for a wide variety of phrasings and contextual nuances, to ensure that the system can handle diverse user requests seamlessly. This model not only drives the skill's functionality but also plays a crucial role in optimizing the user experience by making interactions intuitive and efficient. The first step was to define the skill's invocation name (Mother Mate), which users will use to activate the skill (Open Mother Mate Skill), and then design the intents, which represent the various tasks or actions the skill can perform. The development of an Alexa skill begins with the definition of its invocation name, which acts as the entry point for user interaction. This name must be carefully designed to be both unique and memorable, ensuring accurate recognition by the Alexa NLU engine. Following this, the creation of intents—the core actions the skill can execute—requires meticulous planning. Intents are coupled with sample utterances that guide the NLU model in associating user speech with the corresponding actions. This phase of development is critical as it directly influences the accuracy and efficiency of the skill, ensuring that user commands are correctly interpreted and processed by the underlying system.

With the interaction model defined, the next phase involves the implementation of the skill's logic and functional code. This is typically accomplished using server-side technologies such as Node.js or Python, which provide the necessary infrastructure for handling user requests, managing state, and interfacing with external APIs or databases. The coding process integrates the interaction model with backend logic, enabling the skill to

execute complex operations, process data, and generate contextually relevant responses. The code must be optimized for performance and scalability, ensuring that the skill can handle concurrent user interactions efficiently and maintain low-latency response times. The final stage of skill development involves the seamless integration of the interaction model with the implemented codebase to create a cohesive user experience. This involves rigorous testing to validate the skill’s performance across different scenarios and user inputs. The goal is to ensure that the skill responds accurately and promptly to all user interactions, delivering a user experience that is not only functional but also engaging and intuitive. The successful integration of these components results in an Alexa skill that leverages advanced AI and NLP capabilities to provide a sophisticated, yet user-friendly, interface within the broader smart home ecosystem. We actively utilized usability engineering principles and multi staged interviews with the pilot user group to achieve a well-defined interaction model. Due diligence at the skill definition and implementation lays the groundwork for a seamless and engaging user experience. This also makes Alexa's skills diverse and adaptable to an array of user needs and preferences thereby improving the usability of Alexa enabled devices.

A. 3.1 Overview of Technology Components.

The overview architecture of New Skill development technology components is as described below.

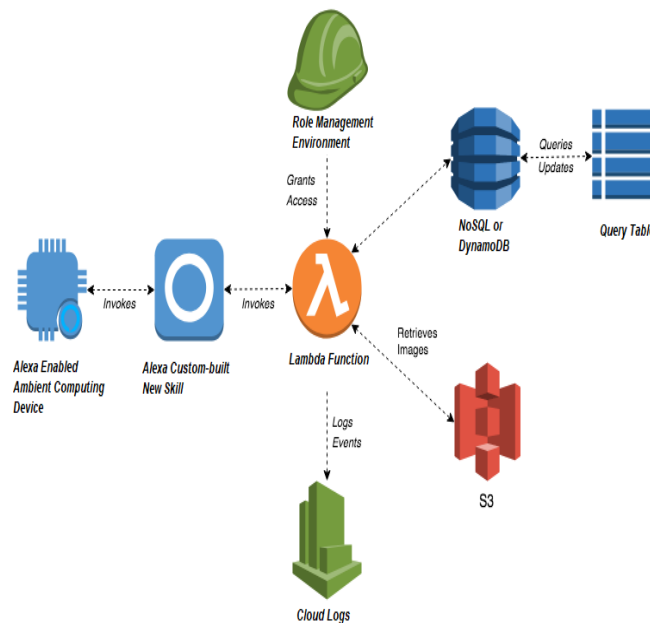


Figure1: Architectural Design and Technology components

1) 3.1.1 Alexa Skills Kit

Alexa Skills Kit is an assortment of APIs and tools that facilitate the addition of new skills to Alexa. This robust collection provides an accessible framework for integrating diverse skills into Alexa, enabling a broad spectrum of functionalities that can cater to the unique needs and preferences of users. By providing a user-friendly platform that simplifies the integration of skills into Alexa, ASK plays a pivotal role in fostering innovation and enriching the overall user experience within the Alexa ecosystem [16].

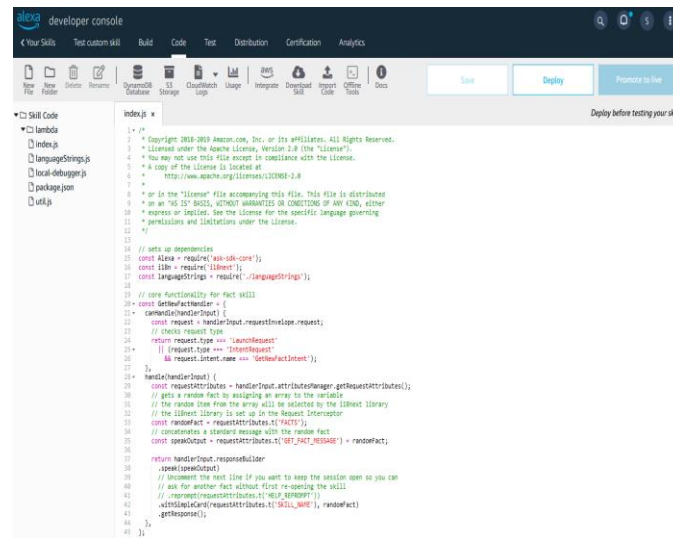


Figure 1: Developer Console for New Alexa Skill

2) 3.1.2 AWS Lambda

For implementing and running the application in a server-less environment we utilized AWS Lambda Function. AWS Lambda stands at the forefront of cloud computing, designed to meet the demands of modern software development by providing a robust platform for the creation and hosting of enterprise-grade serverless applications. At its core, AWS Lambda eliminates the need for developers to manage traditional server infrastructure, allowing them to focus exclusively on writing code and building functionality. This serverless architecture is not only cost-effective, as users only pay for the compute time their code consumes, but also inherently scalable, effortlessly handling varying workloads without the need for manual intervention [17].

3) 3.1.3 Node.js and Python for skill development

Python and Node.js emerge as excellent choices for developers seeking to add a new skill to Alexa, offering distinctive advantages that cater to different preferences. Node.js, built on the V8 JavaScript runtime, is a powerhouse for server-side development. Its event-driven architecture and non-blocking I/O operations make it particularly well-suited for handling multiple concurrent requests, a crucial aspect of building responsive and scalable Alexa skills. Node.js, with its asynchronous nature, enables developers to create efficient, high-performance skills that can handle real-time interactions seamlessly.

4) 3.1.4 Application of usability engineering principles

Usability Engineering Principles were used for design development and refinement of the prototype during the development phase. UE places the user at the Centre of the design process. It involves understanding the needs, behaviours, and preferences of users to create products that align with their expectations. Usability refers to how easy and efficient it is for users to achieve their goals with a product. Our intension was to create a usable design which is intuitive and provides a pleasant experience. There usability dimensions were closely monitored using feedback mechanisms and also by keeping a close watch on performance and agile responses by the skill application. A quick response time contribute to a smoother and more engaging user experience. We have briefly mentioned the Usability Engineering interventions with the relevant Skill Development and Implementation Steps 4.1 to 4.11

B. 3.2 Development Objectives

Leveraging a Conversational User Interface (CUI) through an Alexa Skill application for Pregnancy Counselling presents immense scope for providing personalized and interactive support to users. With the increasing popularity of voice-enabled devices, such as Amazon Echo, integrating a pregnancy counselling application as an Alexa Skill allows users to engage in natural and intuitive conversations. This approach caters to a diverse audience, including those who may find voice interactions more accessible or convenient, especially during moments when hands-free communication is preferred, such as while multitasking or attending to household

chores. The conversational nature of Alexa Skills enables a dynamic exchange of information, empowering users to ask questions, seek advice, and receive personalized counselling in a manner that mimics natural conversation.

As the technology evolves, integrating new features and capabilities into the Alexa Skill further enhances its scope, making it a valuable resource for expectant mothers seeking comprehensive and interactive support throughout their pregnancy journey. The Key Objectives for developing this CUI based Alexa Skill are:

1. Design, develop and deploy, Customised Alexa Skill “Mother Mate”
2. Use DynamoDB as the backend storage for the New Alexa Skill voice responses
3. Design Display cards for new skill
4. Creating a data bucket in Amazon S3 for the storage and retrieval of Display Cards.

C. 3.3 Authentic Knowledge source

Authentic sources of information were identified to create the right knowledgebase for the Mother mate CLI application. Since any consultation is valuable only if the information it provides is relevant and accurate, so special care was taken for utilizing authentic data sources. The NHS database on Pregnancy week-by-week describes as to what to expect from pregnancy and child development stages on a week-by-week basis. The personal details capture module also asks the client on expected date of delivery. This date is then used to compute the current week and extracting the related information [18].

Several other informational resources such as mayoclinic.org, webmd.com were also consulted. Experts of the field were actively involved to validate the content and verify the quality of counselling. The intent was that the CLI application outputs should be better than or at least as good as any other web resource and at the same time it should provide a seamless dialog between the user and the New Alexa Skill.

The skill definition was so designed that it initially provides a succinct knowledge-based response and then goes by the flow of the conversation. The user can steer the conversation to any week as desired for understanding what all happens during the growth of child in the womb.

IV. 4 SKILL DEVELOPMENT AND IMPLEMENTATION STEPS

The new skill definition for Alexa starts with skill definition and designing an interaction model. The designing of the interaction model includes designing the intent schema, predicting probable utterances, and then describing the variable intents through slots. Building the Mother Mate Skill can be outlined through following steps.

1) 4.1: Design the skill's voice interaction model.

The designing of new Alexa Skill includes as the base step an intent schema and the related utterances in a simple grammar. Given below are a few sample intents and the related utterances to illustrate the interaction model

Intent1 PregnancyGuidelines: The example Utterances for this intent are "Give me pregnancy advices." "Tell me something about being pregnant." "What should I know about pregnancy care?"

Intent 2 AskSymptoms for symptoms based consultation: The Utterances are "What are the general symptoms of week number X?" "Tell me about common pregnancy conditions." "What can I expect during pregnancy in terms of symptoms?"

Intent 3 WeightDiary. The related Utterances for WeightDiary are, "Log my weight today as {Weight}." "Update my weight to {Weight} kilograms." "Record that I weigh {Weight} pounds now."

Intent 4 MoodTracker: The example Utterances for this use-case are "Record my mood as {Mood} today." "Update my emotional state to {Mood}." "Log that I'm feeling {Mood} right now."

Intent 5 NutritionAdvice: The example Utterances are "Give me nutrition advice for pregnant women." "Tell me about a healthy diet during pregnancy." "What foods should I focus on for a balanced pregnancy diet?"

Usability Engineering intervention for this stage was to understand the target audience through user research. This involved discussions with the users and conducting interviews to gather insights into actual user needs and behaviours. The intents were designed to align with these research results. The collected data was analysed to identify patterns and pain points. This UE step facilitated the creating of user personas and defining the problem space.

The sample code for the intent utterance programming is given below

Intent Schema, Sample Utterances

```
{ "intents": [ { "intent": "PregnancyGuidelines", "slots": [ { "name": "Weight", "type": "AMAZON.NUMBER" } ] }, { "intent": "AskSymptoms", "slots": [ { "name": "Mood", "type": "CUSTOM_MOOD" } ] }, { "intent": "NutritionAdvice", "types": [ { "name": "CUSTOM_MOOD", "values": [ { "id": "1", "name": { "value": "happy" } }, { "id": "2", "name": { "value": "stressed" } }, { "id": "3", "name": { "value": "calm" } } ] } ] } ] }
```

PregnancyGuidelines, { "PregnancyGuidelines": ["give me pregnancy advices", "what should I know about being pregnant", "tell me about pregnancy care"] }

AskSymptoms, { "AskSymptoms": ["what are the symptoms of pregnancy", "tell me about common pregnancy symptoms", "what can I expect during pregnancy in terms of symptoms"] }

WeightDiary, { "WeightDiary": ["log my weight today as {Weight}", "update my weight to {Weight} kilograms", "record that I weigh {Weight} pounds now"] }

MoodTracker, { "MoodTracker": ["record my mood as {Mood} today", "update my emotional state to {Mood}", "log that I'm feeling {Mood} right now"] }

NutritionAdvice, { "NutritionAdvice": ["give me nutrition advice for pregnant women", "tell me about a healthy diet during pregnancy", "what foods should I focus on for a balanced pregnancy diet"] }

2) 4.2: *Semantic parsing of Intents using AlexaMRL*

AlexaMRL (Alexa Meaning Representation Language) stands out as a decomposable semantic parsing formalism within the realm of natural language processing (NLP). This formalism serves as a structured and systematic way to represent the meaning of user utterances in a format that is both interpretable and analyzable by Alexa-enabled devices. Decomposability in the context of natural language processing (NLP) refers to the ability to break down complex queries or commands into distinct, manageable components, each representing a specific semantic unit. This approach is particularly vital in systems like Alexa, where user queries can range from simple instructions to intricate, multi-part requests. By decomposing these queries, the system can isolate individual elements, allowing for a more granular understanding of the user's intent. This decomposition is not merely a technical convenience but a fundamental strategy for enhancing the precision and relevance of responses in conversational AI systems, making interactions more intuitive and effective. In Alexa, semantic decomposition plays a crucial role in interpreting user commands by breaking down complex queries into their constituent parts. This process allows the system to better understand the underlying intent, leading to more accurate and contextually appropriate responses. The Alexa Multi-Representation Language (AlexaMRL) semantic parsing algorithms are designed to handle the complex task of modeling a wide range of user interactions. These algorithms enable Alexa to interpret natural language inputs more flexibly by mapping user queries to a structured representation that the system can process. AlexaMRL enhances the system's adaptability, allowing it to handle variations in phrasing and syntax while maintaining a high level of accuracy in understanding user intent.

In the development of Alexa skills, Finite-State Transducers (FSTs) are carefully designed to perform specific transformations between input sequences (e.g., spoken commands) and output sequences (e.g., responses or actions). This involves defining the states that the FST can occupy, the transitions between these states based on the input received, and the corresponding output symbols. The design of these FSTs is crucial, as it directly impacts the skill's ability to accurately interpret and respond to user inputs. By tailoring the FSTs to the specific needs of the skill, developers can create more responsive and reliable interactions, ensuring that the skill

operates effectively under various conditions. This encoding mechanism provided a formal and efficient way to model and process sequential data, making Finite-State Transducers a versatile tool in the realm of computational linguistics and automata theory.

Usability Engineering Reviews and testing were conducted to understand the intents and designing the parsing modules of AlexaMRL. For this we were doing the things that the CUI application users would want to do with it. This helped us in identifying the utterances where system is likely to get stuck and the other things that may generate a confused response. These usability review were supplemented with usability testing through actual trials by the users and observing these trials to watch where they are getting stuck.

3) 4.3: Design skill's Display Cards;

Designing effective display cards for Alexa-enabled products is crucial to enhance the overall voice experience and provide users with supplementary visual information. Display cards are visual elements that accompany Alexa voice responses on devices with screens, such as Echo Show and Fire TV. Careful design of display cards enhanced the overall voice experience by providing users with a comprehensive and multi-modal interaction that leverages both voice and visual elements. Advanced Usability Engineering Principles such as emotional design, gamification, and persuasive design were used for continuous improvement of skill display cards.

Consistency in these design elements was ensured by designing cards suggesting navigation support through standard terminology. This helps the users in building a mental model of how the product works, making it easier for them to navigate.

4) 4.4: Define the structure for DynamoDB table

Next development step is to create a DynamoDB table for storing the static knowledge base of a pregnancy-counselling Alexa skill in a set of database tables. These tables provide a structured way to organize the static knowledgebase of a pregnancy-counselling Alexa skill, allowing for efficient retrieval and management of information during skill interactions. Each table is associated with the responses the CUI application will return;

Pregnancy Tips Table: This table stores a collection of pregnancy tips, advice, and general information. Each record can include a unique tip identifier, the actual tip content, and potentially additional metadata such as the publication date or category.

Symptoms Table: Stores information about common pregnancy symptoms in this table. Each record may include a symptom identifier, the description of the symptom, and any relevant guidance or recommendations for managing or alleviating the symptom.

Nutrition Table: This table stores details about recommended nutrition during pregnancy. Records may include information on specific foods, dietary guidelines, and nutritional benefits for both the mother and the developing baby.

Checkup Schedule Table: Keep track of recommended checkup schedules during pregnancy in this table. Each record could contain details about the recommended timing for prenatal checkups, the purpose of each checkup, and any specific preparations or considerations.

Emotional Wellness Table: Store information related to emotional well-being during pregnancy. Records may include advice on managing stress, maintaining a positive mindset, and resources for emotional support. Metadata like the mood categories and associated tips could be included.

Mood Record Table: Manage a database to store the records of mood changes as described by the user for a particular date. This can be used to track the users emotional states and mood changes.

Personal Information Table: A database to capture personal user details like name, mobile number, email, pregnancy due date, preferences, nature of response that generated satisfaction, preferred conversation language etc.

User-Centred Design (UCD) was use to places the user at the centre of the design process to describe a user centric structure of the database. It involved understanding the user needs, behaviours, and preferences of users to create a skill that align with their expectations. Adobe XD was used the wireframing tool for creating basic design layouts and structures of the pregnancy support skill.

5) 4.5: Add raw facts to Database tables using AWS CLI

To initiate the data import, we prepared the raw facts data in a suitable format, such as JSON files or CSV files, and then used the AWS CLI to execute the batch write operation. By incorporating AWS CLI commands into the skill development workflow, we streamlined the setup and maintenance of the knowledgebase, ensuring that the skill is equipped with the latest and most relevant information. This approach automates the process of populating tables and streamlining the data import procedures for the pregnancy counseling Alexa skill. At the same time it significantly enhances the scalability and reproducibility of these operations. By automating table population, the system can efficiently handle large volumes of data, ensuring that the knowledgebase remains up-to-date without manual intervention. Additionally, this method improves scalability by allowing the system to adapt to increasing data demands as the knowledgebase expands. The reproducibility of the process ensures that updates can be consistently applied, maintaining the integrity and accuracy of the information over time, ultimately leading to a more reliable and easily manageable system.

6) 4.6: Create Amazon S3 storage for the Display Cards

Integrating visual content into Alexa Skills is essential for creating a more immersive and engaging user experience. Amazon Simple Storage Service (S3) provides a reliable and scalable storage solution for managing and serving display card images that complement voice interactions. Setting up an S3 bucket to store display card images involves several key steps. Initially, developers must create the bucket through the AWS Management Console, ensuring that it is configured to handle image files efficiently. Integrating an S3 bucket with the Alexa Skill's backend involves configuring the skill to access and retrieve images stored in the bucket. This process typically includes setting up API calls or SDKs to fetch image URLs from the S3 bucket dynamically based on user interactions. Incorporating visually rich content into Alexa Skills significantly boosts user engagement by providing a more interactive experience. Display card images serve as a visual complement to voice-driven interactions, making it easier for users to understand and follow through with commands.

7) 4.7: Map the CUI input to intents and cloud-based service

Building a customized Alexa Skill begins with a critical step: mapping the user's spoken input to predefined intents. This process is essential for transforming user commands into actionable responses within the skill. Defining intents is a crucial part of Alexa skill development, as it establishes the core functionality that the skill will provide. Intents are essentially the actions that the skill can perform based on user input. Once intents are defined, the next step involves creating sample phrases or utterances that users might use to activate these intents. These sample phrases serve as examples of how users are likely to phrase their commands, and they are used to train Alexa's natural language understanding (NLU) algorithms. Usability engineering plays a significant role in the mapping process by ensuring that the skill's design aligns with user needs and preferences. UX-focused interviews and testing are conducted to gather insights into how users interact with the skill and what phrases they naturally use. This user-centered approach actively used in this research helps refine the intent definitions and sample phrases, making them more intuitive and user-friendly. This resulted in creating a seamless and intuitive conversational experience, as it enables the skill to comprehend a wide range of user inputs and deliver relevant responses. This step is crucial to for good quality User Experience (UX). The focus was to design a system that is enjoyable, easy to use, and meet the needs of the users. The mapping of the implementation logic for handling Alexa requests was implemented using JSON request and response formats. The focus was to enhance the overall experience that users have when interacting with the new skill.

8) 4.8 Design and deploy the Lambda function

The Lambda function was used to design and generate responses to the user's utterances. These responses are generated as appropriate voice responses supplemented with display card for better visualisation. When a user issues a voice command or query, the utterance is sent to the associated Lambda function, where we defined the business logic to interpret the intent and generate a suitable response. For this customized Alexa Skill Mother Mate, the Lambda functions serve as the backend, allowing us to seamlessly integrate with external services, databases, or APIs to craft dynamic and context-aware replies. By leveraging Lambda functions, were able to build and encapsulate the intelligence required to process diverse user utterances, enabling the skill to provide meaningful and personalized interactions.

Usability Engineering Principles were utilised in designing the information architecture for the Lambda function. User-centric methodologies such as card sorting and tree testing were utilised. Card sorting is a valuable technique that engages users in organizing content by categorizing information into meaningful groups. In developing a pregnancy support chatbot, aligning the architecture with users' mental models is essential for delivering an effective and engaging experience. User-centered design approaches, such as card sorting and tree testing, are employed to ensure that the chatbot's information structure resonates with users' natural ways of conceptualizing and organizing information. This approach enhances the overall user experience by making the chatbot more intuitive and responsive to users' needs, ultimately facilitating better engagement and satisfaction.

Card sorting is a crucial method used to organize and prioritize content in the development of a pregnancy support chatbot. It involves presenting users with a set of topics or concepts and asking them to group these items into categories that make sense to them. This process helps identify how users naturally conceptualize and categorize information related to prenatal care, nutrition, emotional well-being, and other relevant subjects. Once the card sorting process is completed, the next step involves organizing the content based on the insights gained. For the pregnancy support chatbot, this means grouping and structuring information about prenatal care, nutritional guidelines, and emotional support in a way that mirrors users' natural categorizations. This user-centered organization ensures that the chatbot's content is logically arranged and easily accessible, making it simpler for users to find the information they need and improving their overall experience with the chatbot. Tree testing is a method used to refine the information architecture of a chatbot by evaluating how users navigate through a simplified representation of the chatbot's structure. This technique involves presenting users with a mock-up of the chatbot's information hierarchy and asking them to locate specific pieces of information. The goal of tree testing is to assess the effectiveness of the information structure and identify areas where users may struggle to find the information they need. For a pregnancy support chatbot, tree testing helps ensure that the structure supports efficient and intuitive navigation. During tree testing for the pregnancy support chatbot, users are tasked with finding specific information, such as advice on postpartum exercises or details about pregnancy-related feelings for a particular week. This process involves observing how users interact with the chatbot's structure, noting their navigation patterns, and evaluating their success rates. The results of tree testing reveal valuable insights into user pain points and navigation challenges. By carefully analyzing where users struggle or fail to find information, developers can iteratively refine the chatbot's structure to address these issues.

9) 4.9: Create Identity and Access Management Role

In the context of Alexa Skills development, extending the default Identity and Access Management (IAM) Role is a critical step to enable Lambda functions to interact with DynamoDB. IAM roles define the permissions and access policies that Lambda functions possess, and extending these roles is essential for allowing Lambda functions to perform specific operations on DynamoDB tables. IAM roles in AWS provide a mechanism to manage permissions and access control for various AWS resources. In the case of Lambda functions, the IAM role specifies what actions the function can perform on resources like DynamoDB. To facilitate interactions with DynamoDB, specific permissions must be granted, such as the ability to insert (`dynamodb:PutItem`) or update (`dynamodb:UpdateItem`) items in a DynamoDB table. Properly defining these permissions is crucial for ensuring that the Lambda function operates within its intended scope and performs necessary tasks effectively. After locating the appropriate IAM role, the next step is to extend it by adding permissions for DynamoDB. This involves modifying the role's policies to include actions such as `dynamodb:PutItem` and `dynamodb:UpdateItem`. These actions allow the Lambda function to perform write operations on DynamoDB tables, such as inserting new records or updating existing ones.

10) 4.10: Deploy the Pregnancy Counselling skill

Deploying a customized Pregnancy Counselling Skill for Alexa-enabled devices necessitates a meticulous approach to ensure optimal user experience and functionality. This process involves several critical stages: packaging the skill components, configuring the skill within the Alexa Developer Console, and executing comprehensive testing protocols. The initial phase of deployment involves the systematic packaging of skill components into a deployable artifact. This package comprises the Lambda function code—responsible for backend processing and business logic—the skill manifest, which outlines the skill's metadata and configuration details, and any supplementary resources such as multimedia files. Following the packaging phase, the next step

is to upload the skill package to the Alexa Developer Console. This platform, integrated with the Alexa Skills Kit (ASK), provides a comprehensive interface for managing and deploying Alexa Skills. Post-upload, the skill requires configuration within the Alexa Developer Console. This configuration involves defining the invocation details—such as the specific phrases users will use to activate the skill—and setting up interaction modalities, which dictate the user interaction patterns. The final stage of deployment involves ensuring that the skill's functionality is in alignment with the intended user experience. This requires a thorough analysis of testing outcomes to verify that the skill provides a coherent and satisfactory interaction for users. Addressing any identified issues through iterative refinement is crucial for optimizing performance and user satisfaction.

11) 4.11: Test the Mother Mate skill

The new Alexa Skill was tested using the Alexa Skills Kit Development Console Test tab. The communication flow of the skill is started by invoke the skill using the voice invocation. The Console Test tab has a visual interface along with the voice response interface. It shows the response and simultaneously uses speech synthesis algorithms to generate voice output of Alexa skill response. The test console window is designed to display the request as well as the response body.

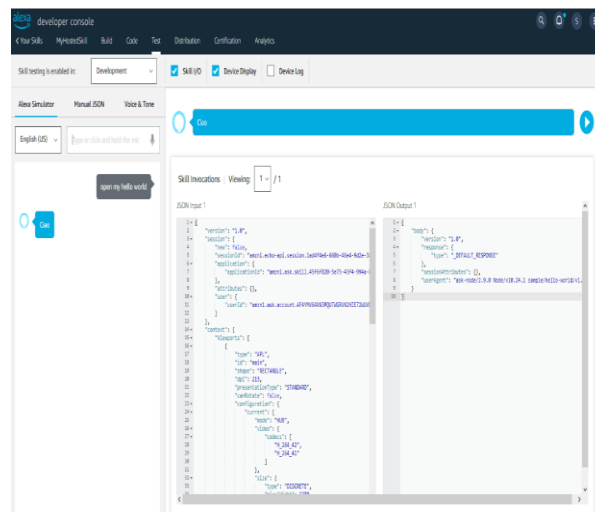


Figure 3: Test page on developer console for skill testing

Usability testing with real users play a pivotal role in assessing the effectiveness and usability of a pregnancy support chatbot. This involved direct interaction with individuals who are representative of the intended user demographic. We observed and analysed how users navigate and engage with the chatbot's features. Through realistic scenarios and tasks, such as seeking advice on prenatal health, managing nutrition during pregnancy, or addressing emotional well-being concerns, we were able to evaluate the chatbot's ability to provide relevant and helpful information. Real users' feedback during usability testing sheds light on aspects like user interface design, clarity of information, and overall user experience, helping identify potential challenges and areas for improvement.

This also provided an opportunity to evaluate this CUI application and gauge the chatbot's impact on user satisfaction and confidence. We collected qualitative and quantitative data on users' satisfaction levels, perceptions of the chatbot's trustworthiness, and their likelihood to rely on the tool for pregnancy-related guidance. The results of this evaluation of the proposed solution helped in refinement of the chatbot to better align with user expectations. This iterative testing process ensured that the pregnancy support chatbot not only meets technical specifications but also resonates with its users trust and reliability.

V. 5.0 CONCLUSION AND FUTURE WORK

The use of chatbots in healthcare applications is recognized for efficiently addressing patient queries with accurate advice. The paper introduces a model for analysing the potential services that a pregnancy companion chatbot can offer, emphasizing the implementation of probable use cases through suitable technology. While acknowledging that the proposed solution is not comprehensive, we suggest possibilities for extending features

and technology applications, such as incorporating alert messages, email notification services, and predictive analytics. The data captured by the chatbot can be utilized to enhance user engagement and monitor personal health, while security measures are necessary to restrict access to authorized users interacting with smart speakers. As a concluding statement we would like to reiterate the potential of chatbots in pregnancy companionship, offering both assistance and a platform for tracking and managing pregnancy-related concerns.

REFERENCES

- [1] Jennifer, S. Shalom, G. Deepayan, P. Aditya and B. Harshitha, "Survey on Chatbots in Healthcare," *International Journal of Research in Engineering, Science and Management*, vol. Volume 5, no. Issue 4, pp. 140-145, April 2022.
- [2] V. Kimbahune, D. Patil, S. Nimbalkar, S. Alekar, M. Yadav and S. Desai, "Conversational Chatbot in Healthcare," *International Journal of Research Publication and Reviews*, vol. Vol 3, no. No 5, pp. pp 2916-2921, May 2022.
- [3] A. B. Kocaballi, S. Berkovsky, J. C. Quiroz, L. Laranjo, H. L. Tong, D. Rezazadegan, A. Briatore and E. Coiera, "The Personalization of Conversational Agents in Health Care: Systematic Review," *Journal of Medical Internet Research*, vol. vol. 21, no. issue 11, p. e15360 p.1, 2019.
- [4] P. Utama, N. Weir, C. Binnig and U. C., "Voice-based dataexploration: Chatting with your database," in *Proceedings of the 2017 workshop on Search-Oriented Conversational AI*, 2017.
- [5] C. B. Eleonor, "Pharmabot: a pediatric generic medicine consultant chatbot," *Journal of Automation and Control Engineering*, vol. Vol 3, no. 2, pp. 33-41, 2015.
- [6] Sheth, H. Y. Yip, A. Iyengar and P. Tepper, "Cognitive services and intelligent chatbots: current perspectives and special issue introduction.," *IEEE Internet Computing*, vol. 23, no. 2, p. 6-12, 2019.
- [7] Patel, P. Garrison, M. Jesus, H. Minas, M. Prince and S. Saxena, "The Lancet's series on global mental health: 1 year," *The Lancet*, vol. 372, no. 2, p. 1354-1357, 2016.
- [8] A. Palwankar, P. A. Satpute, R. Dighe and R. Bhopale, "Artificial Intelligence Based Healthcare Chatbot System," *IEJRD - International Multidisciplinary Journal*, vol. vol. 5, no. no. 5, pp. 6-12, June 2020.
- [9] K. T. Wirawan, M. Sukarsa and P. A. Bayupati, "Balinese Historian Chatbot using Full-Text Search and Artificial Intelligence Markup Language Method," *International Journal of Intelligent Systems and Applications(IJISA)*, , vol. Vol.11, no. No.8, pp. pp.21-34, 2020.
- [10] S. Laumer, M. Christian and G. Fabian, "Chatbot acceptance in healthcare: Explaining user adoption of conversational agents for disease diagnosis," in *27th European Conference on Information Systems (ECIS)*, Stockholm & Uppsala, Sweden, June 8-14, 2019.
- [11] C. V. K. P. N. R. B. R. K. R. I. J. J. S. Anil Kumar, "Self-Diagnosing Health Care Chatbot using Machine Learning," *International Journal of Advanced Science and Technology*, pp. 29(05), 9323-9330, 2020.
- [12] M. Bali, S. Mohanty, S. Chatterjee, M. Sarma and R. Puravankara, "Diabot: A Predictive Medical Chatbot using Ensemble Learning," *International Journal of Recent Technology and Engineering*, vol. Volume 8, no. Issue 2, pp. 68-75, 2019.
- [13] G. Lyons, V. Tran, C. Binnig, U. Cetintemel and T. Kraska, "Making the case for query-by-voice with echoquery," in *Proceedings of the 2016 International Conference on Management of Data ACM*, 2016.
- [14] S. Mahalakshmi, T. Sharmila, S. Priyanka, R. Sastry, B. V. R. Murthy and C. K. K. Reddy, ""A survey on various chatbot implementation techniques",," *Journal of Applied Science and Computations*, Vols. Volume VI,, no. Issue I,, pp. 115-129, January 2019.
- [15] AWS Whitepapers, "Amazon Web Services, "AWS Serverless Multi-Tier Architectures," 2017.
- [16] AWS Whitepapers, "Amazon Web Services, "Amazon Lex Developer Guide," 2017.
- [17] A. W. S. (AWS), ""What Is AWS Lambda?," Amazon, 2018. [Online]. Available: <https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>. [Accessed 26 January 2023].
- [18] NHS, ""Week-by-week guide to pregnancy," National Health Service- Public Health England, 2019. [Online]. Available: <https://www.nhs.uk/start4life/pregnancy/week-by-week/>. [Accessed 22 January 2023].
- [19] A. W. S. (AWS), " "What Is Amazon DynamoDB?, Documentation," Amazon, 2018. [Online]. Available: <https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html>. [Accessed 20 January 2023].
- [20] P. Sayakhot and C. Olah, "Internet use by pregnant women seeking pregnancy-related information: a systematic review," *BMC Pregnancy and Childbirth*, vol. 16, no. 1, p. 1-10, 2016.
- [21] P. Orlow, Z. Z and B. TW, "Perceived organizational affiliation and its effects on patient trust: role modeling with embodied conversational agents.," *Patient Education Counselling*, vol. 100, no. 9, p. 1730-1737, 2017.
- [22] Bickmore, L. Pfeifer, D. Byron, S. Forsythe, L. Henault, B. Jack, R. Silliman and O. Paasche, "Usability of conversational agents by patients with inadequate health literacy: evidence from two clinical trials.," *J Health Communication*, vol. 15, no. S2, pp. 197-210, 2010.