Enhancing the MDX Employability Support via a Voice User Interface

Dereck Lam Hon Wah
Middlesex University Mauritius
Flic en Flac, Mauritius
Dl661@live.mdx.ac.uk

INTRODUCTION

The job market has become competitive, and having a degree is no longer enough to secure a job, so graduates with the support of the MDX Employability service need to maximize their chances by standing out through their achievements in university by building their CVs and developing their soft skills through various workshops. With a voice-activated personal assistant like Amazon Alexa or Apple Siri[1], we believe that the employability service will leverage its support to students with quicker access to information and services with the benefits of voice commands and dialogue-style conversations, allowing hands-free interactions [2].

DESIGN CONCEPT

The proposed Voice User Interface (VUI) system is a chatbot called MDX Employability Service Career Voice, which will feature speech recognition and synthesis with natural language understanding (NLU) to understand and respond to voice input with appropriate responses [2], like speech, visual output, or both. It will first help answer questions about the employability service, and responses containing URLs will be redirected to the desired page, for example, the employability service LinkedIn profile. The system will also assist students in finding a list of companies with internship opportunities based on their field of study and provide a map and auditory directions to any company of interest. Users can also book 1:1 appointments with advisors and enhance their current CVs. During the design concept in Appendix A Figure 1, we used five VUI guideline principles different from generic heuristics to improve its usability: guidance, workload, user control, consistency, and error management. Firstly, our system will have an advantage with its visual component, enabling users to interact using voice commands and the screen, known as a multimodal interface [3]. The available input box and conversation log will promote contextual awareness and memorability while allowing the users to use it at their own pace[3]. Guidance is the means to advise, inform, and guide the users throughout their interaction using prompting to communicate the conversation status to the users or immediate feedback, increasing user confidence and satisfaction [4]. Many existing systems, like Apple Siri, use prompting while requiring the users to do something explicit to instruct the system that they will speak [5]. In Appendix A Figure 1, the user will press the mic button to enable the listening state, and the system responds with visual feedback by changing the input box placeholder message from “Message” to “Speak to me”. Incorporating linguistic markers at the end of dialogues also helps orient users to the end of a conversation, allowing them to decide whether to continue or not [4]. For example, in Appendix A Figure 2, ending each successful intent with “Can I provide you with additional assistance?” ensures the system remains user-centric, allowing further interaction based on the needs. Responses should remain consistent through tone and speaking style [4], with some synonyms in response wording also encouraged to break the monotony and keep the user's attention, reducing the risk of the system seeming too robotic [3]. In Appendix A Figure 2, telling the users when tasks were successful or unsuccessful with proper messages and avoiding using a lot of “yes/no” responses does not demonstrate good conversational design and rewording them to something clear will increase the user trust [5]. The workload criterion focuses on the brevity of the user commands and the system responses conciseness [4]. Conciseness in shorter responses without extra words will decrease users’ listening time and better direct the information toward what the user wants to know or do. Brevity in user commands is to limit the number of steps a user must take to achieve a goal. For example, in Appendix Figure 1, booking a 1:1 appointment will require the user to perform several actions: opening Google Calendar, clicking on add an event, setting the time and date, adding a subject and confirming the event, actions that will annoy the user, which are grouped into one single command in our chatbot like "Book an appointment at 1 AM tomorrow". User control is vital in allowing users to interrupt the system or initiate another command at any moment without waiting for it to finish its sentence [3]. In Appendix A Figure 1, the stop button will act as an emergency exit command, improving user experience. Error management is to prevent errors by developing error prevention strategies and eliminating error-prone conditions [4]. In Appendix Figure 2, when booking a 1:1 appointment, the system will check if the user has provided the required parameters: date and time, and if not, it sends a prompt to request back these. Error message quality is essential, as a message only saying that the system did not understand the user correctly will only promote user frustration. The system should add information or questions that help users express their requests and avoid this error happening again [4]. For example, when the date is missing, the system will prompt: “On which date would you like to book your appointment”, that helps the user identify the problem and prevent the system from crashing. The system will be an AI-based chatbot [6] that uses NLU, a branch of natural language processing (NLP), to identify the intention (intent classification) and extract attribute values conveyed (slot-filing) from the user utterance [7], to engage in contextual discussion and give a more interactive experience. Examples of intent classification and slot filing can be depicted in Appendix A Table 1.

PROTOTYPE

This prototype is built using Vue 3 and Tailwind CSS for the front end, with TypeScript leveraging type safety. Vue 3 is a progressive JavaScript framework with efficient Document Object Model (DOM) reactivity and compiler-optimized rendering, contributing to a more efficient and responsive user experience. For the backend, the stack is Node.js with Express.js, utilizing JavaScript to create a RESTful API to communicate with various external APIs. The VueUse package has some utility functions around the Web Speech API, making integration of speech recognition (speech-to-text) and synthesis (text-to-speech) easier. The chatbot utilizes Google Dialogflow ES, a cloud NLU solution, making it easier to design and train a speech conversational agent with intent classification and slot-filling capabilities[7]. For intent classification, the developer provides a set of training sentences for each intent. For slot-filling, which refers to entities in Dialogflow, they can either be system default or custom ones created by the developer [6]. The training process is then fully automated and does not require many training phrases compared to CFG-based approaches, making it easier to develop and extend in the future [7]. For each conversational turn, the RESTful API backend calls the Dialogflow API to query the agent about user utterances. When Dialogflow matches an end-user expression, it will look for the best response between the developer intent configurable and the knowledge bases, which act as a cloud database containing an FAQ CSV about the employability service. Each intent will have multiple response variations to create a more natural conversation, and those containing URL links, like social media links, will trigger a tab redirection to the desired page. The browser's “localStorage” stores the conversation log and speech synthesis settings, ensuring data persistence across page reloads like ChatGPT. In Appendix B Figure 3, the users can clear the conversations by clicking the bin button or disable the speech synthesis by toggling the speaker button. The user can switch between voice commands or typing by clicking the mic button. Depending on the selected mode, the input box placeholder message will dynamically change from "Type a message" to "Speak to me", enforcing the guidance principle [4]. The user can interrupt the chatbot execution to initiate another command by clicking the stop button [3]. In Appendix B Figures 3 and 4, for intents that require a dynamic response, the chatbot uses Dialogflow fulfilment through a webhook and calls external API services. For booking a 1:1 appointment, the webhook calls Google Calendar API, which checks the user calendar for an existing event at the defined date and time, if none, it dynamically adds one. Finding an internship will consist of two key features. The first feature involves the webhook calling OpenAI API using a custom prompt to generate a list of 3 companies in the user's specified industry, bypassing ChatGPT standard context and summary response. The second feature uses a company name to call the Google Maps API. It retrieves an embedded map and text directions to the specified company, rendered on the front end in visual and auditory formats, providing an interactive and accessible user experience. Here is the demo video link: https://youtu.be/g95VgUhoBI4

EVALUATION METHOD

To evaluate the user experience and usability of the chatbot, we conducted behavioural patterns observation during user interaction [1] and a post-evaluation method using the UEQ+, which seems the most appropriate compared to existing VUI questionnaires that lack task-related usability aspects such as efficiency, controllability or learnability [8]. The UEQ+ is a modular framework derived from the User Experience Questionaire (UEQ), consisting of 16 scales to be combined to form a solid questionnaire [9]. From a VUI perspective, the UEQ+ will assess three specific UX aspects: response behaviour (i.e., communicates like a human conversationalist), response quality (i.e., its responses fulfil the user intention), and comprehensibility (i.e., correctly understands the user intention using natural language) [8], [9]. The evaluation took place on 15 March 2023 in the Middlesex University Mauritius library, for which we sent an email invitation to 8 computer science students. 5 students participated voluntarily, above the minimum requirement of two for this coursework. Before the evaluation, we ensure to adhere to ethical guidelines by soliciting uninformed consent from the participants about the purpose of this evaluation, their right to withdraw, and any data collection aspects. This information is also present at the start of the questionnaire, including a statement where they agree to participate, and we did not collect personal information like names and emails to protect participants' identities and ensure confidentiality. Before starting the evaluation, each participant received a task list to complete using the VUI, covering a range of functionalities, from asking questions about the employability service to booking a 1:1 appointment, and looking for internships. During the participant interaction, we observed any challenges faced or errors and recorded quantitative data like task success rate, completion time and number of errors. After the task’s completion, they filled out a Google Form, which included the UEQ+ and some open-ended questions about the VUI. Below we depicted the UEQ+ questionnaire with its construct and questions to be rated on a scale from 1 (strongly disagree) to 5 (strongly agree):

Response Behaviour:

- RB1 - How quickly does the VUI system respond to your commands or questions?

- RB2 - How predictable are the responses from the VUI system to your inputs?
- RB3 - How well does the VUI system adapt to your way of speaking (e.g., accent, speed)?

Response Quality:

- RQ1 - How accurate are the responses provided by the VUI system to your queries?
- RQ2 - How relevant are the VUI system's responses to your specific needs and inquiries?
- RQ3 - Do the responses from the VUI system provide you with complete information for your queries?

Comprehensibility:

- C1 - How clear is the speech output (in terms of pronunciation and volume) from the VUI system?
- C2 - How simple and understandable is the language used by the VUI system?
- C3 - How easy is it to interpret and use the information provided by the VUI system?

EVALUATION FINDINGS

Based on the result in Table 2, we calculated the mean score and standard deviation for each question and the overall mean score for each construct to identify the strengths and weaknesses within the system.

Construct	**Question ID**	**S. D**	**Mean**
Response Behaviour	RB1	0.8	3.6
	RB2	0.9	4.1
	RB3	0.7	2.9
	Overall Mean Score:		3.5
Response Quality	RQ1	0.6	3.4
	RQ2	0.5	4.2
	RQ3	0.4	3.6
	Overall Mean Score		3.7
Comprehensibility	C1	0.5	3.9
	C2	0.7	3.8
	C3	0.6	4.0
	Overall Mean Score		3.9

Table 1 - UEQ+ Result table

In the Response Behavior construct, RB2 had the highest score, indicating that the users can easily anticipate the VUI responses, which is essential for building trust and ensuring ease of use. An above-average score for RB1 suggests the system responds quite efficiently to user inputs. However, we noted delays in responses involving external APIs or, in some cases, no response at all, which varied with internet quality, leading to user frustration. For RB3, we observed challenges in adapting to diverse accents, especially the Mauritian accent, and some participants expressed concerns over the system's language limitation to English. The Response Quality construct showed strong outcomes for RQ2, indicating the VUI system effectively interpreted user intents and provided information that met their needs, suggesting the features are well-aligned with the goals of undergraduate students, such as finding internships and booking appointments with employability advisors. However, RQ1 and RQ3 showed room for improvement, particularly in intent classification, compromised by a small set of training phrases in Dialogflow, leading to unexpected answers or errors, especially with direct company name inputs for location searches. We also observed a minor discrepancy in the system's responses for finding companies in a specified industry, where it uses OpenAi API and unexpectedly included ChatGPT context information. Lastly, we observed that in group settings, where multiple voices were present, the VUI system struggled to identify the primary speaker, leading to erroneous command captures. Comprehensibility achieved the highest overall mean score, with clear speech output and responses designed for ease of understanding in simple English, making the VUI accessible to a broader audience.

Following the evaluation findings, we elaborate on some recommendations to improve the user experience of the proposed VUI prototype. For external API delays, the developer can provide visual feedback such as "Checking for the latest information, please wait" to manage expectations during longer waits and for no response, the developer needs to develop a fallback mechanism with a generic response if an external API fails within a timeout. For language limitations to English and challenges in understanding diverse accents, the developer can integrate additional language models to support multilingual interactions and modify the interface to allow users to select their preferred language for interactions. Another suggestion from participants was adding a feature to adjust the speaking speed, benefiting users who are non-native English speakers or those with auditory processing challenges. For intent classification inaccuracy, the developer needs to enrich the training dataset with a broader set of training phrases, focusing on edge cases and variations in how users may phrase their queries, where additional user feedback and observed interactions will be encouraged to identify gaps in the training data. For the minor discrepancy with the OpenAI API response, the developer can perform prompt engineering to strengthen the current prompt to avoid unnecessary content. For the group setting problem, the developer could involve spatial filtering to isolate the primary speaker's voice speech signal from a mixture of signals from multiple simultaneously active talkers and background noise [10].

CONCLUSION

This was a great learning experience on the different stages of a VUI development from design concept, prototyping with new technology, and evaluation using different techniques.

REFERENCES

[1] C. M. Myers, A. Furqan, and J. Zhu, "The Impact of User Characteristics and Preferences on Performance with an Unfamiliar Voice User Interface," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, Glasgow Scotland Uk: ACM, May 2019, pp. 1–9. doi: 10.1145/3290605.3300277.

[2] F. Rakotomalala, H. N. Randriatsarafara, A. R. Hajalalaina, and N. M. V. Ravonimanantsoa, "Voice User Interface: Literature review, challenges and future directions," *Syst. THEORY CONTROL Comput. J.*, vol. 1, no. 2, Art. no. 2, Dec. 2021, doi: 10.52846/stccj.2021.1.2.26.

[3] S. Ross, E. Brownholtz, and R. Armes, "Voice user interface principles for a conversational agent," in *Proceedings of the 9th international conference on Intelligent user interfaces*, Funchal, Madeira Portugal: ACM, Jan. 2004, pp. 364–365. doi: 10.1145/964442.964536.

[4] C. Nowacki, A. Gordeeva, and A.-H. Lizé, "Improving the Usability of Voice User Interfaces: A New Set of Ergonomic Criteria," in *Design, User Experience, and Usability. Design for Contemporary Interactive Environments*, A. Marcus and E. Rosenzweig, Eds., Cham: Springer International Publishing, 2020, pp. 117–133. doi: 10.1007/978-3-030-49760-6_8.

[5] C. Pearl, *Designing Voice User Interfaces: Principles of Conversational Experiences*. O'Reilly Media, Inc., 2016.

[6] S. Singh and D. S. Singh, "Effective Analysis of Chatbot Frameworks: RASA and Dialogflow".

[7] O. Keszocze and I. G. Harris, "Chatbot-based assertion generation from natural language specifications," in *2019 Forum for Specification and Design Languages (FDL)*, Sep. 2019, pp. 1–6. doi: 10.1109/FDL.2019.8876925.

[8] A. M. Klein, K. Kölln, J. Deutschländer, and M. Rauschenberger, "Design and Evaluation of Voice User Interfaces: What Should One Consider?," in *Design, Operation and Evaluation of Mobile Communications*, G. Salvendy and J. Wei, Eds., Cham: Springer Nature Switzerland, 2023, pp. 167–190. doi: 10.1007/978-3-031-35921-7_12.

[9] A. M. Klein, A. Hinderks, M. Schrepp, and J. Thomaschewski, "Construction of UEQ+ scales for voice quality: measuring user experience quality of voice interaction," in *Proceedings of Mensch und Computer 2020*, Magdeburg Germany: ACM, Sep. 2020, pp. 1–5. doi: 10.1145/3404983.3410003.

[10] M. Taseska and E. A. P. Habets, "Informed Spatial Filtering for Sound Extraction Using Distributed Microphone Arrays," *IEEEACM Trans. Audio Speech Lang. Process.*, vol. 22, no. 7, pp. 1195–1207, Jul. 2014, doi: 10.1109/TASLP.2014.2327294.

APPENDICES

Appendix A

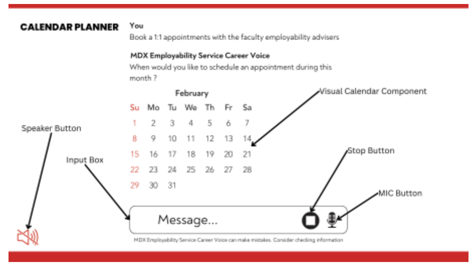

Figure 1 Wireframe.

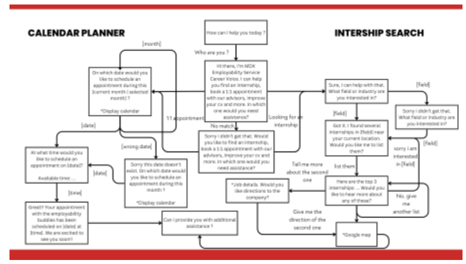

Figure 2 Conversation Structure

User Utterance	**Intent Classification**	**Slot Filling**
Set up a one-on-one meeting for **tomorrow** at **10 AM**.	Book 1:1 appointment	Date: Tomorrow Time: 10 AM
What's the location of **Ceridian**?	Request Company Location	Company: Ceridian
I am looking for an internship in **media**	Find an internship	Industry: Media

Table 2 NLP Intent Classification and Slot Filling Example

Appendix B

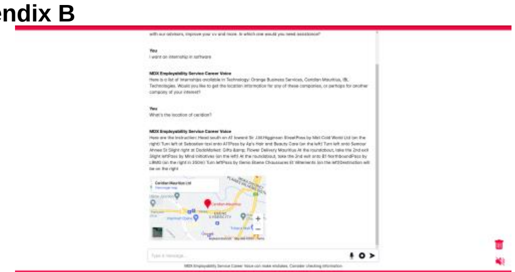

Figure 3 Prototype Map Implementation

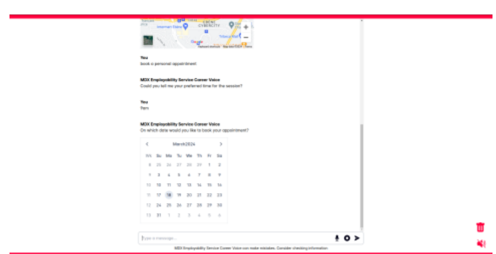

Figure 4 Prototype Calendar Implementation