



Research article

Voice communication module for automotive instrument panel indicators based on virtual assistant open-source solution - Mycroft AI

Módulo de comunicación de voz para indicadores de un panel de instrumentos automotriz basado en la solución de asistente virtual de código abierto - Mycroft AI

Ricardo Hernández Mejía¹ , Francisco Javier Ibarra Villegas² , Caín Pérez Wences² 

¹Posgrado CIATEQ, A.C., Av. Nodo Servidor Público #165 Col. Anexa al Club de Golf, Las Lomas, 45131 Zapopan, Jalisco, México

²CIATEQ, A.C. Centro de Tecnología Avanzada, Santiago de Querétaro, 76150, Querétaro, México.

Corresponding author: Ricardo Hernández Mejía, Posgrado CIATEQ, A.C., Av. Nodo Servidor Público #165 Col. Anexa al Club de Golf, Las Lomas, 45131 Zapopan, Jalisco, México. E-mail: richernandezm@gmail.com. ORCID: 0009-0005-2384-1464.

Submitted: 19 de Septiembre del 2023

Accepted: 19 de Septiembre del 2023

Published: 5 de Octubre del 2023

Abstract. - *This work was originated from the increasing interest in several industries to implement voice based virtual assistant solutions powered by the Natural Language Processing field of study. This work is focused on the automotive industry Human Machine Interface related products, specifically the Instrument Panel. Nowadays people are constantly using virtual assistants like Google Assistant, Alexa, Cortana or Siri on their electronic devices. Furthermore, 31% of cars have a built-in virtual assistant, for example Ford uses Alexa, Mercedes-Benz and Hyundai use Google Assistant, BMW and Nissan use Cortana, GM uses IBM Watson, Honda uses Hana and Toyota uses YUI. Apart from the proprietary solutions described earlier, there are also contemporary open-source generic solutions available on the market, such as Mycroft AI which stands out from other technologies due to ready to deploy, well documented, simple installation on a Linux PC or RPI SoC, and simple execution. This paper presents a way to use Mycroft AI as an alternative to add artificial intelligence-based voice assistance to applications in the automotive domain. The voice communication module presented here drives notifications related to three different entities: seat belt, fuel level and battery level, all of them are telltales present in any automotive Instrument Panel. Since the Mycroft AI design approach is based on Human Centered Design (HCD), the voice communication module presented here provides real user experience (UX) based design. As a conclusion, Mycroft AI demonstrates great potential as an alternative to add voice assistance to automotive industry Human Machine Interface related products. About future work, due to the fact that Mycroft AI is based on Python, there are many possibilities for connecting and expanding the voice communication module by using countless Python libraries in order to import and process any type of information, in any format or source, for example the information from communication technologies like CAN, LIN, Ethernet, MOST, GPS or any other device or technology in order to create comprehensive automotive solutions.*

Keywords: Instrument panel; Virtual assistant; Voice communication module; Mycroft AI; Human centered design; User experience.

Resumen. - *Este trabajo se originó del creciente interés por parte de diferentes industrias para implementar soluciones de asistente virtual basado en voz impulsadas por el campo de estudio del Procesamiento del Lenguaje Natural. Este trabajo está enfocado en los productos relacionados a la Interfaz Humano Máquina de la industria automotriz, específicamente el Panel de Instrumentos. Hoy en día las personas usan constantemente asistentes virtuales como Google Assistant, Alexa, Cortana o Siri en sus dispositivos electrónicos. Más aún, 31% de los autos tienen un asistente virtual integrado, por ejemplo, Ford usa Alexa, Mercedes-Benz y Hyundai usan Google Assistant, BMW y Nissan usan Cortana, GM usa IBM Watson, Honda usa Hana y Toyota usa YUI. Aparte de las soluciones de marca registrada descritas anteriormente, también hay soluciones genéricas de código abierto contemporáneas disponibles en el mercado, tales como Mycroft AI que se hace notar por sobre otras tecnologías por características como listo para usar, bien documentada, instalación simple en una PC Linux o RPI SoC, y una ejecución simple. Este artículo presenta una manera de usar Mycroft AI como una alternativa para agregar inteligencia artificial basada en asistencia de voz a aplicaciones en el dominio automotriz. El módulo de comunicación de voz presentado aquí maneja notificaciones relacionadas a tres diferentes entidades: cinturón de seguridad, nivel de gasolina y nivel de batería, todos ellos son indicadores virtuales presentes en cualquier Panel de Instrumentos Automotriz. Dado que el enfoque de diseño de Mycroft AI se basa en Diseño Centrado en el Humano (HCD), el módulo de comunicación por voz presentado aquí provee un diseño basado en experiencia de usuario (UX) real. Como conclusión, Mycroft AI demuestra gran potencial como una alternativa para agregar asistencia de voz a los productos relacionados a Interfaz Humano Máquina de la industria automotriz. Acerca del trabajo a futuro, debido al hecho que Mycroft AI está basado en Python, existen muchas posibilidades para conectar y expandir el módulo de comunicación por voz a través del uso de innumerables bibliotecas de Python para importar y procesar cualquier tipo de información, en cualquier formato o fuente, por ejemplo la información proveniente de tecnologías de comunicación tales como CAN, LIN, Ethernet, MOST, GPS o cualquier otro dispositivo o tecnología para crear soluciones automotrices integrales.*

Palabras clave: Panel de instrumentos; Asistente virtual; Módulo de comunicación por voz; Mycroft AI; Diseño centrado en el humano; Experiencia de usuario.





1. Introduction

There is a growing interest in the field of NLP study in academia, as several industries are deploying virtual assistant solutions. [1] reports 27% of people have whether Google Assistant, Alexa, Cortana or Siri, the smartphone is as 85% high in adoption over intelligent speakers, tablets, laptops, smart TVs, wearable technology, and home automation, nevertheless, a remarkable 31% of cars nowadays have a virtual assistant. In the automotive industry, [2] shows pairs of manufacturers and virtual assistants, this is Ford uses Alexa [3], Mercedes-Benz [4] and Hyundai [5] use Google Assistant, BMW [6] and Nissan [7] use Cortana, GM uses IBM Watson [8], Honda uses Hana [9], Toyota uses YUI [10]. From the academy and research communities, related relevant patents provide innovation trends, on the one hand, Automotive Virtual Personal Assistant [11] which is a system that actively monitors the car state to provide relevant notifications, and on the other hand, Proactive Virtual Assistant [12] which evaluates user's information to provide suggestions and perform actions in advance. Finally on worldwide events like the Consumer Electronics Show, technology and innovations from carmakers are presented as well, like the Mercedes-Benz User Experience Hyperscreen [13] which apart from multiple displays a virtual assistant is integrated, by Hey Mercedes command car information is retrieved, and taking into consideration GPS location, information of nearby restaurants, parking lots, between others is provided to the driver. It is well known that open-source products have characteristics such as low or non-existing cost for usage and distribution depending on the license type, high quality, security, open access and flexibility to modify its components, furthermore collaboration and innovation are present due to development communities' support; so, the virtual assistant solutions are not the exception to the rule. Based on the earlier, this work presents an investigation result in a

comparison table of relevant features of contemporary open-source virtual assistant solutions, besides a detail of the chosen solution Mycroft AI's components, algorithms, and methods is provided. Afterwards, the steps to create a Mycroft AI skill or application are presented, followed by the customization of intent, dialog and entity files for automotive instrument panel's indicators seat belt, fuel level and battery level. The main application design relies on dynamic behavior diagrams, a state machine, and a sequence diagram, to create a base product of a voice communication module for automotive instrument panel indicators based on Mycroft AI as the final goal. Finally, achievements, contributions and future work are listed and discussed.

2. Background

A wide range of open-source tools and technologies related to virtual assistance are available in the market, investigation was performed on sites such as makezine [14] where free and private voice assistants are compared based on open source architecture components, medevel [15] in which open-source technologies and platforms of popular voice assistants are analyzed, yourtechdiet [16] which lists project's origin and up-to-date status of best open source voice assistants, and finally libhunt [17] which reports virtual assistant solutions' popularity based on activity, commits on corresponding repositories and mentions from development communities, based on the earlier, the "Table 1. Contemporary open-source virtual assistant solutions" was created, the table shows a comparison between different contemporary open-source virtual assistant solutions and their most relevant characteristics. Mycroft [18] stood from the crowd due to ready to deploy, well documented, simple installation on a Linux PC, and straightforward execution.



Table 1. Contemporary open-source virtual assistant solutions.

| Assistant | Os/Hw | Prog. Language | Popularity | Internet required | Privacy | Customization | Documentation |
|----------------------|---------------------|---------------------------------|------------|-------------------|---------|---------------|---------------|
| Mycroft | Linux, RPI | Python, Bash | 😊 | ☁️ | ✅ | 😊 | 😊 |
| Leon | Windows, Mac | Node.js, Python, Http | 😊 | ☁️ | ✅ | 😐 | 😐 |
| Rhasspy | RPI | Docker, Python, Shell | 😐 | 🏠 | ✅ | 😊 | 😞 |
| Jasper | RPI | Python | 🙋 | ☁️ | ❌ | 😐 | 😞 |
| Almond | Linux, Web | Javascript | 😞 | ☁️ | ✅ | 😐 | 😞 |
| OpenAssistant | Windows, Linux, Mac | Own SDK | 😞 | 🏠 | ✅ | 😐 | 🙋 |
| LinTO | RPI | Docker, Python, Bash, C++, Java | 😐 | ☁️ | ✅ | 😐 | 😊 |
| Aimybox | Android, iOS | Apache 2.0 | 😐 | ☁️ | ❌ | 😐 | 😞 |
| Kalliope | RPI, Linux, Android | Python, Rest, Bash | 😐 | 🏠 | ❌ | 😐 | 😊 |

2.1 Why use Mycroft?

Mycroft is presented on IEEE's Entrepreneurs in consumer Electronics [19] as an open-source software platform which integrates technologies that have significantly improved in recent years such as speech recognition, text to voice, command processing, etc., all these technologies

allow to add voice assistance powered by artificial intelligence to any application executed on laptops, speakers, Raspberry PIs and cars.

A successful deployment of Mycroft is [20] in which an intelligent robot assistant is created to handle smart homes for the elderly.



Over other solutions Mycroft [21] presents itself as:

- **Open source:** The Mycroft code can be analyzed, copied, modified, and distributed, these cannot be said for Alexa, Google Assistant, Cortana, Siri which are black boxes, contents is hidden and protected by commercial licenses.
- **Respectful of users' privacy:** Voice recording works only if user grants permission.
- **Multiple hardware compliant:** RPI, Android, Linux PC.
- **Light:** Designed to be executed on low cost, low power, and low resources hardware.
- **Community oriented:** Vibrant, committed, and helpful community.

2.2 Modular Mycroft

Mycroft implements Voice Stack components [14] as part of an open-source virtual assistant architecture, these components can be configured, personalized, started and stopped independently, their openness and flexibility is the main advantage of Mycroft against the commercial and other open-source counterparts.

- **Wake Word Detection:** "Hey Mycroft" is default, customized through [22]. Due to the simplicity to configure a new Wake Word through phonemes on a text configuration file based on CMU Dictionary of sound, it was decided to use PocketSphinx [23] part of CMUSphinx [24] originally based on the SPHINX-II [25] speech recognition system, this system achieves improved unified acoustic and language modeling through normalized feature representations, multiple-codebook semi-continuous hidden Markov models, between-word senones, and multi-pass search algorithms. Software Precise can be used to provide higher precision on Wake Word detection at the expense of demanding training a NN on large audio sequences.

- **Speech To Text:** Google STT [26] is the default engine, deep learning progress on voice transcription makes possible models like LSTM RNN [27] perform remarkably on the speech recognition domain and further text transcription.
- **Intent Parser:** Adapt [28] is the default software developed by Mycroft AI to identify utterances or commands as machine readable data structures after parsing the natural language text input. Software Padatious [29] can be used to provide higher precision on utterance detection at the expense of demanding training a NN on required phrases.
- **Text to Speech:** Mimic [30] is the default software, developed by Mycroft AI along with VocaliID, Mimic is based on the open-source text to speech synthesis engine CMU's Flite, voice synthesis is achieved by Classification and Regression Tree and Finite State Transducer algorithms. Google TTS [31] can be also used.
- **Mycroft Skills:** Mycroft AI applications, e.g., timers, alarms, whether, time, date, etc., custom skills can be developed with Mycroft Skills Kit (MSK) support and Python programming language.

2.3 User Centered Design

Mycroft's design and development approach is driven by user's needs, the philosophy also known as User Centered Design (UCD) a concise version of Human Centered Design (HCD) which turns around observe and understand users, assures useful, understandable, pleasant, and enjoyable products to interact with, essentially the final goal of the Norman's User Experience [32].

Mycroft makes use of the Design Thinking [33] method, which synthesizes the problem handling as follows:

When _____,
I want _____,
So, I am able to _____.



This follows the Mycroft's application flow.

- What words will be used?
- What will be the answer to provide?
- What information will be needed?
- What dependencies are required.?

3. Methodology

After the hardware to work with is chosen whether it is RPI, Android or Linux PC, in order to use Mycroft AI, steps in [18] must be followed

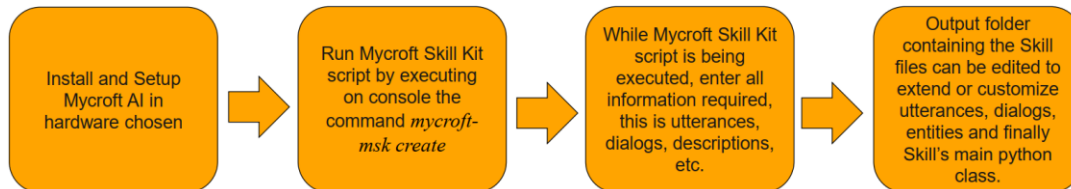


Figure 2. Mycroft AI flow to create a Skill.

3.1 Mycroft Skill Kit

Mycroft Skills Kit (MSK) utility is installed along with Mycroft with the objective to facilitate the creation, upload, and upgrade of skills in the corresponding local directories or repositories. *mycroft-msk create* is the console command used to run MSK and execute an interactive script to ask and answer information to generate a skill skeleton in the form of a template.

3.2 Creation of Automotive Telltales Skill

An automotive Instrument Panel (IP) presents to the driver different ECUs's signals visually or auditory, this work considers three main telltales

to install and set all the dependencies needed to successfully run the software solution, generic tests are provided to make sure the solution is properly installed and dependencies working well, this is especially relevant for the microphone and speakers peripherals and corresponding drivers, just in case of trouble, useful help is provided by Mycroft AI with a troubleshooting section [34].

The “Figure 2. Mycroft AI flow to create a Skill” shows the steps to create and modify the Skill to meet custom needs.

which are part of the safety relevant telltales according to the National Highway Traffic Safety Administration (NHTSA) [35]: seat belt, fuel level and battery level.

For example, for seat belt telltale, applying the Design Thinking approach previously presented, we get the following output:

When seat belt status is queried and unfastened,
I want to be provided a suggestion to fasten the seat belt,

So, I am able to achieve a trustful trip.

“Table 2. Mycroft Skill Kit script for Skill Telltales” shows the *mycroft-msk create* command result for seat belt telltale.



Table 2. Mycroft Skill Kit script for Skill Telltales.

| Script question | Answer given |
|--------------------------|---|
| Utterances | What is the car seat belt status |
| Dialog | Car seat belt is fastened |
| Short description | Telltale status / Indicator level retrieval |
| Long description | Skill to retrieve instrument panel's current telltale status or indicator level |
| Author | Hernandez Ricardo |
| Category | IoT |
| Tags | None |

Finally, MSK's script asks for GitHub's repositories creation for the skill, this will be required only if the skill will be published in the Mycroft's marketplace. In our case, it is enough

the MSK's output, this is the Skill Telltales located in local directory /opt/mycroft/skills, "Table 3. Mycroft Skill Telltales output" shows output files and directories.

Table 3. Mycroft Skill Telltales output.

| Item | Description |
|--------------------------|--|
| Locale | Directory containing files (intent and dialog) for every language supported, en-us for English as default. |
| __init__.py | The skill python-based core to import libraries, define class which inherits from Mycroft Skill to work with voice, define own methods to handle intents and dialogs, create skill's class instance and its execution. |
| README.md | Skill's human readable information, provided in MSK's script, short and long descriptions, author, category. |
| settingsmeta.yaml | Parameters of Mycroft's profile stored at https://sso.mycroft.ai/home.mycroft.ai , like date, time, time measured, location, voice type, etc. |
| manifest.yml | External software dependencies if any. |

3.3 Intents, Dialogs and Entity files

Within Locale directory it is found intent file and dialog file which contains phrases originated in

MSK's script, later by manually editing this pair of files and following the Mycroft's guidelines and rules, custom utterances and dialogs were added to these files. Besides, file telltales.entity



was manually created to provide flexibility, a wild-card to handle different kind of phrases for each telltale, the entities are used in both, intent and dialog files. Referenced files and corresponding contents are summarized in “Table 4. Intent, Dialog and Entity files”. Facts about dialogs are on the one hand, that they are

randomly chosen by Mycroft to simulate a more natural impression, and on the other hand in this work dialogs are intentionally incomplete to handle apart each telltale’s status or level through python code. The result is the user experience’s feedback functionality in action.

Table 4. Intent, Dialog and Entity files

| File | Content |
|-------------------------|--|
| telltales.intent | Car {telltale} status Current car {telltale} status What is the car {telltale} status Status of car {telltale} status (Give Tell) me the car {telltale} status |
| telltales.dialog | {telltale} is Car {telltale} is Current car {telltale} is |
| telltales.entity | seat belt fuel level battery level |

Furthermore, a sample of user experience’s feedforward functionality, this is, the helpful information provided to the user to decide what to do next, is exemplified as a simulated car trip

in three different modes. Every trip mode represents a different level of feedforward functionality detailed in “Table 5. Trip modes for feedforward functionality”.

Table 5. Trip modes for feedforward functionality

| Trip mode | System’s behavior description |
|--------------------------|---|
| Notification Mode | Telltale’s status or level is provided with simple sentence and no more. Decision of what to do next relies completely on the user. |
| Suggestion Mode | Telltale’s status or level is provided with context information sentence about the car’s location. Even decision of what to do next still relies on user, on this mode the decision is facilitated. |
| Action Mode | Telltale’s status or level is provided with context information sentence about the car’s location. Decision of what to do next relies completely on assumed autonomous car. |



A simple state machine model was created to depict available Trip Modes, this is presented in “Figure 3. Trip Modes state chart”. To start a trip

and enter one of the Trip Modes available, utterances are introduced by adding intent files presented in “Table 6. Trip modes intent files”.

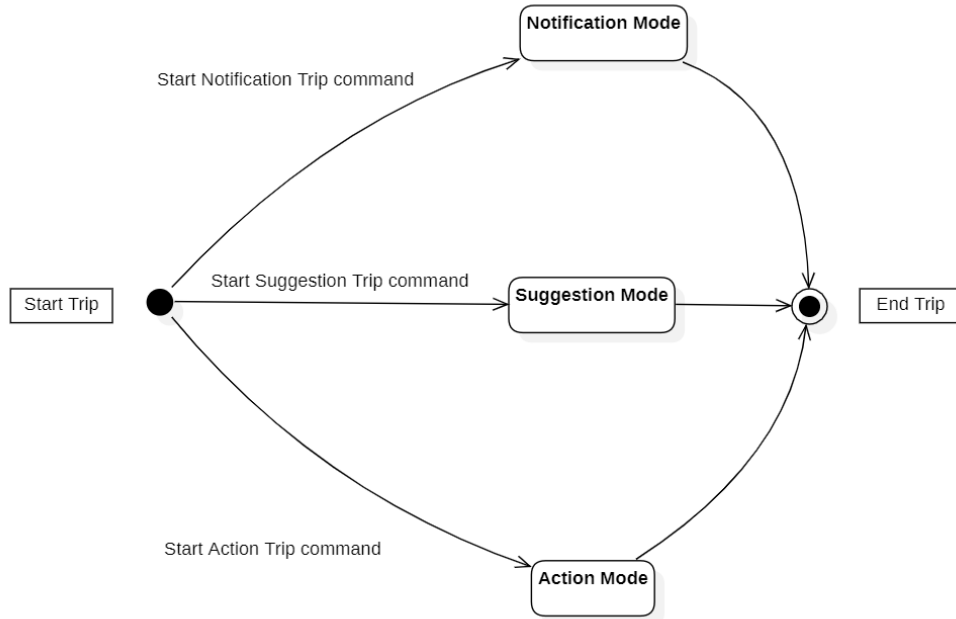


Figure 3. Trip Modes state chart.

Table 6. Trip Modes intent files

| File | Content |
|---------------------------------|-------------------------|
| trip_notification.intent | start notification trip |
| trip_suggestion.intent | start suggestion trip |
| trip_action.intent | start action trip |

Simulated trip’s sequence of actions for each Trip Mode which takes into consideration utterances, dialogs and simulated ECU’s messages are shown in corresponding sequence diagrams. On “Figure 4. Skill Telltales Notification Mode sequence diagram”, after welcome message is provided by the voice communication module running inside an UX

ECU, utterance from user is received to start a trip in Notification Mode, then simulated ECUs: Seat Belt, Fuel Level and Battery Level provide corresponding status to finally trigger notification messages with predefined simple sentence dialogs. Decision on what to do next relies on the user.



sd Skill Telltales Notification Mode Sequence Diagram

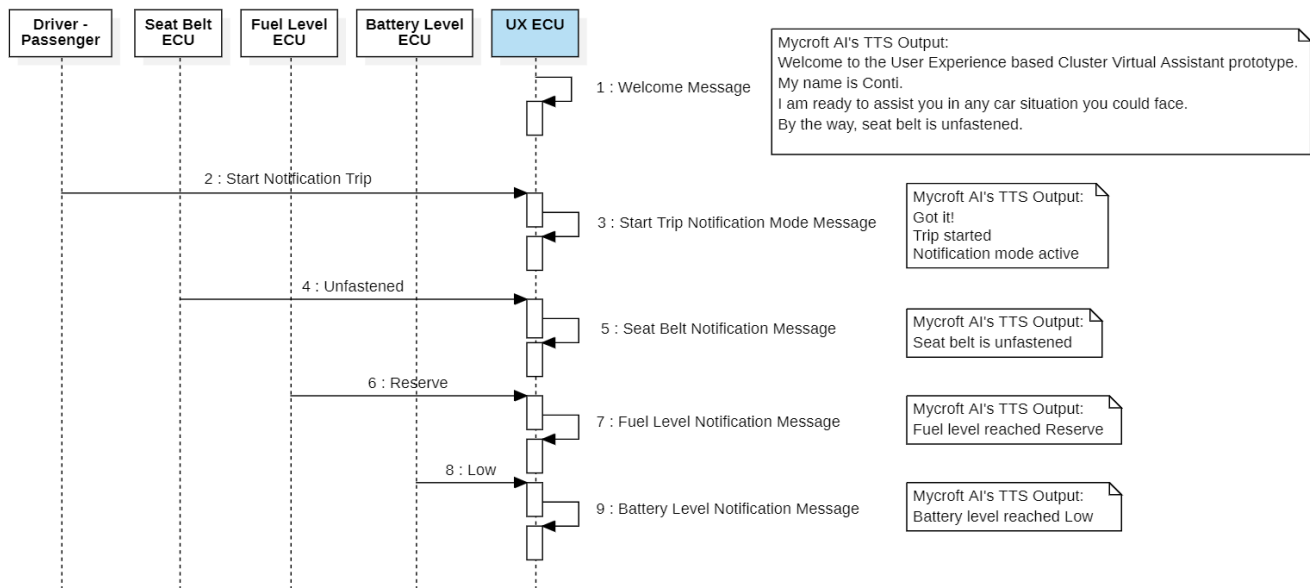


Figure 4. Skill Telltales Notification Mode sequence diagram.

On “Figure 5. Skill Telltales Suggestion Mode sequence diagram”, after welcome message is provided by the voice communication module running inside an UX ECU, utterance from user is received to start a trip in Suggestion Mode, then simulated ECUs: Seat Belt, Fuel Level and

Battery Level provide corresponding status to finally trigger suggestion messages with predefined sentence dialogs based on context information about the car’s location. Even decision of what to do next still relies on user, on this mode the decision is facilitated.

sd Skill Telltales Suggestion Mode Sequence Diagram

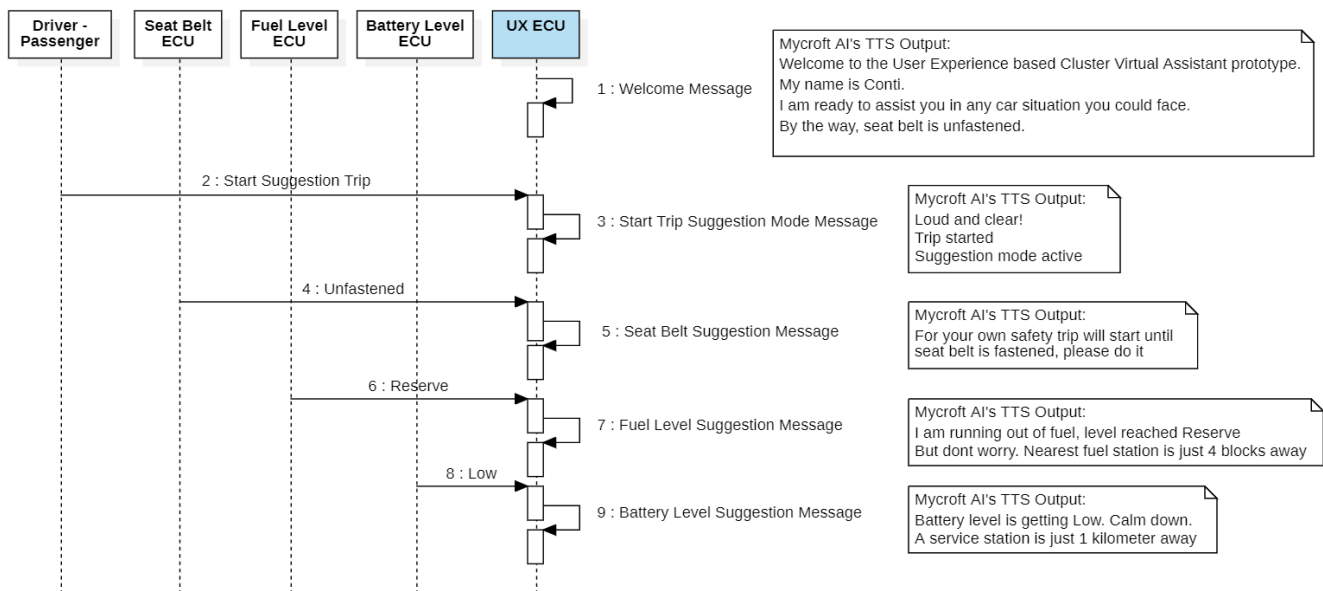


Figure 5. Skill Telltales Suggestion Mode sequence diagram.



On “Figure 6. Skill Telltales Action Mode sequence diagram”, after welcome message is provided by the voice communication module running inside an UX ECU, utterance from user is received to start a trip in Action Mode, then simulated ECUs: Seat Belt, Fuel Level and

Battery Level provide corresponding status to finally trigger action messages with predefined sentence dialogs based on context information about the car’s location. Decision of what to do next relies completely on assumed autonomous car.

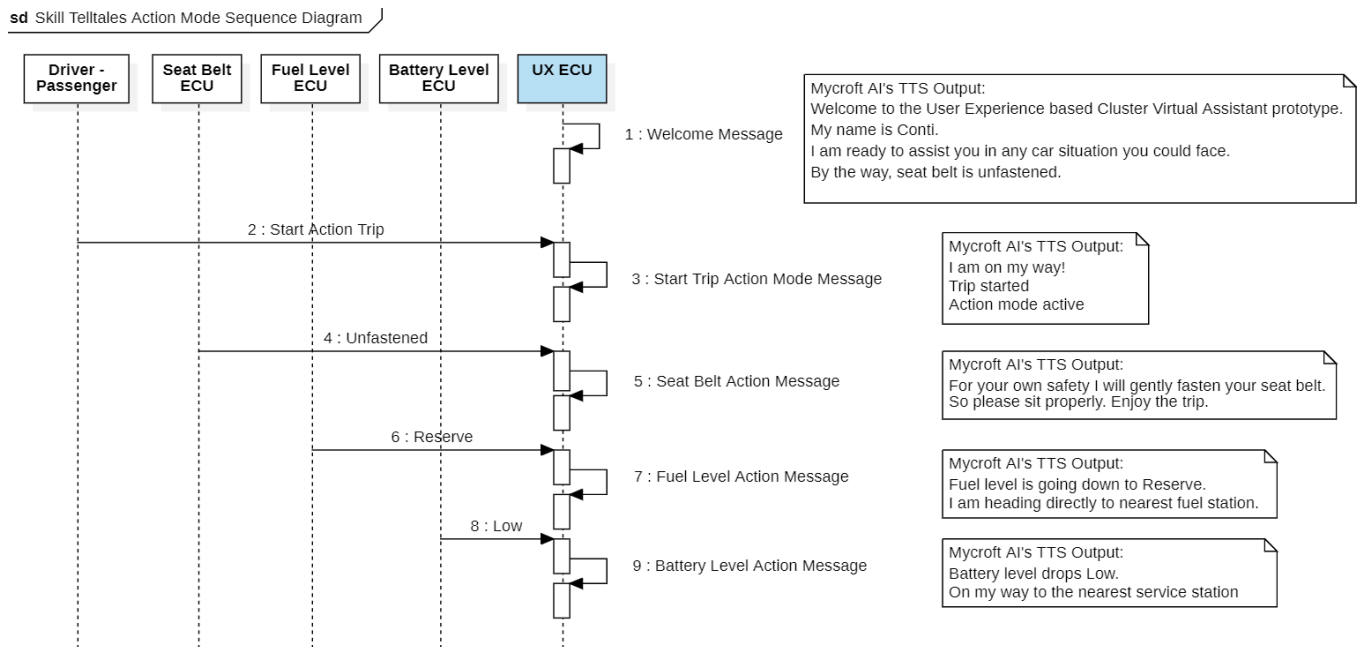


Figure 6. Skill Telltales Action Mode sequence diagram.

3.3 Skill Telltales class

The Mycroft generated and later customized Skill Telltales python class, through the methods **handle_telltales**, **handle_notification**, **handle_suggestion** and **handle_action** is the main responsible to process each telltale's status request and speaks a predefined output according to the dynamic behavior represented in “Figure 3. Trip Modes state chart”, “Figure 4. Skill Telltales Notification Mode sequence diagram”, “Figure 5. Skill Telltales Suggestion Mode

sequence diagram”, and “Figure 6. Skill Telltales Action Mode sequence diagram”.

3.4 Test cases

To provide a real example of the Skill Telltales class in action “Table 7. Test cases for Skill Telltales” summarizes meaningful test cases, this is the input called intent, utterance or command provided by a human with his voice to Mycroft AI and corresponding output called dialog provided Mycroft AI to the human.



Table 7. Test cases for Skill Telltales.

| Intent (Human) | Dialog (Mycroft AI) |
|--------------------------------------|--|
| Car seat belt status | Current car seat belt is Unfastened |
| What is the car fuel level status | Fuel level status is Reserve |
| Give me the car battery level status | Car battery level status is Low |
| Start Notification Trip | Notification messages from simulated trip in Notification Mode |
| Start Suggestion Trip | Suggestion messages from simulated trip in Suggestion Mode |
| Start Action Trip | Action messages from simulated trip in Action Mode |

4. Results and discussions

The main goals and results achieved in this work are listed as follows:

- Voice-Based Virtual Assistants investigation result reveals an increasing trend in the automotive industry, this is confirmed by their omnipresence in current and short-term future vehicles, also it is proof enough of their relevance as a big value asset that must be paid attention to.
- Contemporary Open-Source Virtual Assistant solutions presented here as investigation result, evidences the worldwide motivations to work and contribute to the improvement and refinement of methods, algorithms and techniques related to the NLP.
- Mycroft AI demonstrated to be the most complete Open-Source Virtual Assistant based on the facts of installation and setup on a Linux PC is quite simple, documentation and troubleshooting is available and helpful, and finally, deployment of voice applications or skills is straightforward aided by the MSK.
- The skill Telltales created by following the UCD approach and the Design Thinking method

over three telltales seat belt, fuel level and battery level, represents a voice communication module for automotive instrument panel indicators with the following characteristics: 1) **Simple**, because skill output folder and dependency files are reduced and concrete, 2) **Portable**, since it can be deployed in hardware like RPI, Android or Linux PC which has Mycroft AI instantiated, and 3) **Customizable**, though the adaption of the intent, dialog, entity files as well as the python based class of the skill.

5. Conclusions

The main contribution of this work is to lay the foundations of the evolution of contemporary Automotive Instrument Panels from a mere physical device to a comprehensive virtual device, this is, automobile users demand of future vehicles voice-based virtual assistance, which is easily achieved by using Mycroft AI, besides the unique footprint of Mycroft AI is the deployment of the UCD approach and the Design Thinking method that makes the difference against competitors.

For this work, the scope of the Instrument Panel's indicators was reduced to only three telltales which are safety relevant: seat belt, fuel level and



battery level, however real Instrument Panel's indicators are numerous safety relevant or non-safety relevant, a short list of intents and dialogs was introduced in corresponding files to keep it simple, and finally a fixed status for each telltale was set in the corresponding python class to conclude generic dialogs.

About the applicability of the contribution of this work, the automotive industry whether big companies, mid-range companies or start-ups are the main interested parties and beneficiaries of this work, since voice communication modules are fully compatible with any kind of HMI products inside the car. Due to the fact Mycroft AI is an open-source solution, its usage brings economic benefits along with legal responsibilities.

Future work relies on the vast possibilities of connection and expansion of the voice communication module presented in this work, through the usage of countless python libraries to import any kind, format or source of information, for example, different systems interconnection through communication buses and technologies already available in the automotive domain like CAN, LIN, Ethernet, MOST, GPS location between others, in order to get, process and present relevant information to the final user.

6. Acknowledgment

To Posgrado CIATEQ A.C. due to the institutional support and guidance received to conclude this work in a professional and successful way. To Continental Automotive Occidente due to the sponsorship provided to perform the master's degree along with Posgrado CIATEQ A.C. which made possible this work. To Dr. Francisco Javier Ibarra Villegas due to their guidance and support on the process to shape and concrete this work.

7. Authorship acknowledgment

Ricardo Hernández Mejía: Conceptualization, Methodology, Software, Validation, Formal analysis, Research, Resources, Original draft, Visualization, Project administration. *Francisco Javier Ibarra Villegas*: Review and Editing, Supervision, Project Administration. *Cain Pérez Wences*: Review and Editing.

References

- [1] S. Arora, V. A. Athavale, H. Maggu and A. Agarwal, "Artificial Intelligence and Virtual Assistant—Working Model," in *Mobile Radio Communications and 5G Networks*, Singapore, 2020.
- [2] K. C. Majji and K. Baskaran, "Artificial Intelligence Analytics—Virtual Assistant in UAE Automotive Industry," in *Inventive Systems and Control*, Singapore, 2021.
- [3] Ford, "Amazon Alexa & Ford® | Alexa Built-In with Ford® Streaming," Ford Motor Company, 2023. [Online]. Available: <https://www.ford.com/alexa/>. [Accessed 17 May 2023].
- [4] M.-B. G. Media, "Mercedes-Benz delivers integration of the Google Assistant," Mercedes-Benz, 16 December 2016. [Online]. Available: <https://group-media.mercedes-benz.com/marsMediaSite/en/instance/print/Mercedes-Benz-delivers-integration-of-the-Google-Assistant.xhtml?oid=15181402>. [Accessed 17 May 2023].
- [5] "Send destinations, start or select the temperature of your Hyundai using Google Home and Bluelink.," Hyundai Motor America, [Online]. Available: <https://owners.hyundaiusa.com/us/en/resources/blue-link/using-blue-link-with-google-home.html>. [Accessed 17 May 2023].
- [6] H. Boeriu, "Microsoft CEO mentions the successful BMW integration of Alexa and Cortana," BMW Group, 24 September 2018. [Online]. Available: <https://www.bmwblog.com/2018/09/24/microsoft-ceo-mentions-the-successful-bmw-integration-of-alexa-and-cortana/>. [Accessed 17 May 2023].
- [7] "Nissan Intelligent Mobility: Nissan presenta su asistente virtual Cortana diseñado para facilitar la vida cotidiana de los conductores," Nissan, 17 April 2017. [Online]. Available: <https://mexico.nissannews.com/es->



- MX/releases/nissan-intelligent-mobility-nissan-presenta-su-asistente-virtual-cortana-dise-ado-para-facilitar-la-vida-cotidiana-de-los-conductores. [Accessed 17 May 2023].
- [8] "Hello, OnStar – Meet Watson," General Motors Company, October 2016. [Online]. Available: <https://news.gm.com/newsroom.detail.html/Pages/news/us/en/2016/oct/1025-watson.html>. [Accessed 17 May 2023].
- [9] "Honda Introduces “Cooperative Mobility Ecosystem” at CES 2017," Honda Global, 6 January 2017. [Online]. Available: <https://global.honda/newsroom/news/2017/c170106eng.html>. [Accessed 17 May 2023].
- [10] "Toyota Concept-i Makes the Future of Mobility Human," Toyota Newsroom, 4 January 2017. [Online]. Available: <https://pressroom.toyota.com/toyota-concept-i-future-of-mobility-human-ces-2017/>. [Accessed 17 May 2023].
- [11] S. A. Friedman, P. R. Remegio, T. U. Falkenmayer, R. A. Kyle, R. Kakimi, L. D. Heide and N. N. Puranik, "Automotive Virtual Personal Assistant". Estados Unidos Patent US 2019/0311241 A1, 10 Octubre 2019.
- [12] V. Aggarwal and D. Binay, "Proactive Virtual Assistant". Estados Unidos Patent US 2020/0175386 A1, 2021.
- [13] Georg Walthart, "Mercedes-Benz presents the MBUX Hyperscreen at CES: New MBUX generation with intelligent new features such as “Mercedes Travel Knowledge”," Mercedes-Benz Group Media, 11 Enero 2021. [Online]. Available: <https://group-media.mercedes-benz.com/marsMediaSite/en/instance/ko/Mercedes-Benz-presents-the-MBUX-Hyperscreen-at-CES-New-MBUX-generation-with-intelligent-new-features-such-as-Mercedes-Travel-Knowledge.xhtml?oid=48617114>. [Accessed 16 09 2023].
- [14] K. Reid, "Private By Design: Free and Private Voice Assistants," makezine, 17 Marzo 2020. [Online]. Available: <https://makezine.com/article/home/connected-home/private-by-design-free-and-private-voice-assistants/>. [Accessed 14 Octubre 2022].
- [15] H. Mousa, "10 Top Open-source Voice Assistants Projects for Developers," MEDevel, 19 Diciembre 2018. [Online]. Available: <https://medevel.com/10-open-source-voice-assistants/>. [Accessed 14 Octubre 2022].
- [16] "List of Top 7 Open Source Voice Assistants," YourTechDiet, 2022. [Online]. Available: <https://yourtechdiet.com/blogs/open-source-voice-assistants/>. [Accessed 14 Octubre 2022].
- [17] S. Bright, "Mycroft-core Alternatives," LibHunt, 2022. [Online]. Available: <https://www.libhunt.com/r/mycroft-core>. [Accessed 14 Octubre 2022].
- [18] J. Montgomery, "Mycroft AI," Mycroft AI, Inc, 2015. [Online]. Available: <https://mycroft.ai/>. [Accessed 14 Octubre 2022].
- [19] T. Wilson, "Entrepreneurs in Consumer Electronics: Steve Penrod of Mycroft AI [Professional Development]," *IEEE Consumer Electronics Magazine*, vol. 8, no. 4, pp. 74-75, 2019.
- [20] e. a. N. H. Abdallah, "Smart Assistant Robot for Smart Home Management," in *2020 1st International Conference on Communications, Control Systems and Signal Processing (CCSSP)*, El Oued, Algeria, 2020.
- [21] Mycroft, "Why use Mycroft AI?," Mycroft, 2020. [Online]. Available: <https://mycroft-ai.gitbook.io/docs/about-mycroft-ai/why-use-mycroft>. [Accessed 22 Enero 2023].
- [22] C. M. University, "The CMU Pronouncing Dictionary," Carnegie Mellon University, 20 Noviembre 2014. [Online]. Available: <http://www.speech.cs.cmu.edu/cgi-bin/cmudict>. [Accessed 14 Octubre 2022].
- [23] C. M. University, "Pocket Sphinx," Carnegie Mellon University, 25 Marzo 2014. [Online]. Available: <https://github.com/cmuspinx/pocketsphinx>. [Accessed 14 Octubre 2022].
- [24] C. M. University, "CMU Sphinx," Carnegie Mellon University, 25 Marzo 2014. [Online]. Available: <https://cmuspinx.github.io/>. [Accessed 14 Octubre 2022].
- [25] X. Huang, F. Allewa, M.-Y. Hwang and R. Rosenfeld, "An Overview of the SPHINX-II Speech Recognition System," *Proceedings of the workshop on Human Language Technology*, no. HLT '93, pp. 81-86, 1993.
- [26] Google, "Google Cloud STT," Google, 13 Noviembre 2013. [Online]. Available: <https://cloud.google.com/speech-to-text>. [Accessed 14 Octubre 2022].



- [27] Beaufays, Françoise, "The neural networks behind Google Voice transcription," Google, 11 August 2015. [Online]. Available: <https://ai.googleblog.com/2015/08/the-neural-networks-behind-google-voice.html>. [Accessed 06 June 2023].
- [28] "Adapt," Mycroft, 2020. [Online]. Available: <https://mycroft-ai.gitbook.io/docs/mycroft-technologies/adapt>. [Accessed 20 May 2023].
- [29] "Padatious," Mycroft, 2021. [Online]. Available: <https://mycroft-ai.gitbook.io/docs/mycroft-technologies/padatious>. [Accessed 20 May 2023].
- [30] "Mimic TTS," Mycroft, 2022. [Online]. Available: <https://mycroft-ai.gitbook.io/docs/mycroft-technologies/mimic-tts>. [Accessed 20 May 2023].
- [31] "Google Text-to-Speech," Google Inc, [Online]. Available: <https://cloud.google.com/text-to-speech?hl=es>. [Accessed 20 May 2023].
- [32] D. Norman, The Design of Everyday Things, New York: Basic Books, 2013.
- [33] R. F. Dam, "The 5 Stages in the Design Thinking Process," Interaction Design Foundation, Junio 2021. [Online]. Available: <https://www.interaction-design.org/literature/article/5-stages-in-the-design-thinking-process>. [Accessed 14 Octubre 2022].
- [34] Mycroft AI, "Audio Troubleshooting," Mycroft AI, 16 September 2022. [Online]. Available: <https://mycroft-ai.gitbook.io/docs/using-mycroft-ai/troubleshooting/audio-troubleshooting>. [Accessed 16 September 2023].
- [35] U. S. Government, "Federal Motor Vehicle Safety Standards; Controls, Telltales and Indicators," NHTSA, 17 August 2005. [Online]. Available: <https://www.federalregister.gov/documents/2005/08/17/05-16325/federal-motor-vehicle-safety-standards-controls-telltales-and-indicators>. [Accessed 04 September 2022].

Derechos de Autor (c) 2023 Ricardo Hernández Mejía, Francisco Javier Ibarra Villegas, Caín Pérez Wences



Este texto está protegido por una licencia [Creative Commons 4.0](https://creativecommons.org/licenses/by/4.0/).

Usted es libre para compartir —copiar y redistribuir el material en cualquier medio o formato— y adaptar el documento —remezclar, transformar y crear a partir del material— para cualquier propósito, incluso para fines comerciales, siempre que cumpla la condición de:

Atribución: Usted debe dar crédito a la obra original de manera adecuada, proporcionar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que tiene el apoyo del licenciante o lo recibe por el uso que hace de la obra.

[Resumen de licencia](#) - [Texto completo de la licencia](#)