www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 17(2), 2026, 190-205. ISSN: 2007-1558. https://doi.org/10.61467/2007.1558.2026.v17i2.1267

Conversational agent with integrated tracking based on generative AI considering the effect of document order and structure

Ruy Rodrigo Gutiérrez Tafoya[1], Maricela Quintana López[1], Asdrúbal López Chau[2], Saturnino Job Morales Escobar[1].
[1]Universidad Autónoma del Estado de México UAEM. CU Valle de México, Universidad Boulevard S/N Valle Escondido, Río San Javier, 54500 Estado de México, México.
[2]Universidad Autónoma del Estado de México UAEM. CU Zumpango, leva. Viejo a Jilotzingo S/N, 55600 Zumpango, Estado de México, México.
E-mails
[1]rgutierrezt001@uaemex.mx, [1]mquintanal@uaemex.mx, [1]sjmoralese@uaemex.mx, [2]alchau@uaemex.mx

Abstract. This article describes the development of a conversational agent for managing and tracking academic procedures. Based on requirements analysis, the essential components were defined to provide real-time updates on processes and improve the user experience. The development phases included dialogue flow design, training with Generative Artificial Intelligence (GAI), implementation of a data warehouse, generators, database integration, a web interface, and a module.
Follow-up. A crucial aspect was the importance of data order and structure. Standardization and organization of training content are fundamental to ensuring the accuracy and relevance of responses. The solution coherently articulates the components, ensuring robust interconnections. The follow-up module expands the agent's capabilities, offering comprehensive, efficient, and personalized interaction.

Keywords: chatbot, conversational agent evaluation, education

Article Info
Received Dec 2, 2025
Accepted Jan 11, 2026

1 Introduction

Human-machine interaction has been the subject of study since 1950, when Alan Turing published his article “Computing Machinery and Intelligence” in the journal MIND, in which he introduced the test that bears his name, establishing a theoretical basis for the development of technologies that seek to imitate and improve communication between humans and computer systems (Bender et al., 2021). As technology advances, human-machine interaction has undergone a significant evolution.

Thanks to natural language processing, conversational agents, also known as chatbots, have emerged. These systems are designed to maintain natural dialogues with users, facilitating communication through human-like language. Their ability to manage interactive conversations and generate coherent and contextual responses has transformed how organizations interact with clients and users, establishing them as indispensable tools in various sectors, including customer service (Tran, 2019), healthcare (Romero et al., 2020), and, the focus of this work, education.

In this latter context, recent advances in Generative Artificial Intelligence (GAI) have expanded the capabilities of conversational agents, enabling them to produce flexible and adaptive responses. Unlike traditional systems based solely on rules or coincidences, GAI allows the generation of original content based on context, offering more natural and personalized interactions (Andia et al., 2024).

Educational institutions often face challenges in providing personalized attention and resolving student inquiries regarding academic and/or administrative processes. Inadequate follow-up can lead to incomplete or erroneous procedures and ultimately negative consequences for the student. Among the problems observed are the following:

The demand for assistance with academic and administrative procedures exceeds the capacity of the available staff. This is because, at times, staff must repeat the same information to different students, resulting in long wait times or delayed responses if the request was submitted electronically. This demonstrates the lack of an automated and accessible mechanism that provides immediate and personalized assistance.

Furthermore, some students leave their procedures incomplete due to a lack of proper follow-up. Having a tool that allows users to track the progress of a procedure, identify outstanding requirements, or send timely reminders would help mitigate the consequences of delays or incomplete applications.

Considering the above, the need arises to create a conversational agent that uses generative artificial intelligence (GAI), trained with documents related to academic and administrative processes and information.

Regarding the academic environment, this is to ensure that the responses provided are relevant, reliable, and aligned with the institution's academic and administrative realities. Some studies focus on evaluating the quality of responses based on the prompt delivered to the agent, while others consider response time and user traffic.

connected to the agent. This research analyzes how the order and structure of input data impacts the generated response. The goal is to determine if a specific data order can optimize the quality of care provided to students.

In addition to measuring agent responses, a tracking module is needed to allow students to clearly check the status of their applications in real time. This module should display the overall progress of the application, detailing completed steps, outstanding requirements, and relevant dates associated with each stage.

In this way, the service would be automated, and the student experience would be optimized through intelligent and adaptive support, guaranteeing monitoring of their procedures.

2 Theoretical framework

This section addresses the theoretical foundations of conversational agents, enabling an understanding of both their technological and evaluative dimensions. First, the concept of a conversational agent and the main platforms available for its development are described. Then, the most commonly used metrics for measuring its performance are presented.

A conversational agent can be defined as a software system designed to interact with users using natural language, either written or spoken. These agents are capable of answering questions, performing automated tasks, and holding structured dialogues, making use of technologies such as artificial intelligence, natural language processing, and, in some cases, machine learning (Andia et al., 2024).

The evolution of conversational agents has been marked by the shift from rigid, pre-programmed response systems to more sophisticated models based on artificial intelligence. This process has been driven by the emergence of specialized platforms, including IBM Watson Assistant, Microsoft Bot Framework, Rasa, and Dialogflow CX (Cloud, 2025; DeepMind, 2023; IBM, 2025). These platforms contain architectures that allow for the design of complex conversation flows, the management of intents and entities, and the integration of data from external stores or services.

To determine the most robust and complete platform for developing a conversational agent based on Generative Artificial Intelligence (GAI) and specializing in structured processes, a comparative evaluation of the aforementioned platforms was conducted, focusing on their architecture and native integration capabilities with GAI. The results of this analysis are as follows: Google Dialogflow CX uses a State Machine (Flows)-based architecture. This approach is designed to offer granular and visual control of dialogue, making it ideal for managing complex, linear processes, such as tracking student paperwork (Cloud, 2025). IBM Watson Assistant employs architecture based primarily on Intents and Actions. This design aims for efficient automation in enterprise and contact center environments, although the management of complex dialogues is handled through contexts, which can be less intuitive than a workflow system (IBM, 2025). Microsoft Bot Framework operates under a Code Framework architecture. This means that development requires the use of programming languages (such as C# or Python) to define the dialogue logic, offering maximum control and customization at the expense of greater complexity in development and maintenance (Services, 2025).

Rasa is an open-source platform whose architecture is based on machine learning, stories, and rules. This structure is specifically designed for dynamic dialogues trained with examples, requiring a high level of technical knowledge in ML for its correct implementation (Technologies, s. f.).

In the specific case of the Dialogflow CX platform, developing an agent requires defining dialogue flows to establish the stages of the conversation and the transitions between topics, configuring a datastore for data management (knowledge base for IAG), and implementing generators responsible for providing dynamic and contextualized responses, as well as training the agent. This modular structure governs the methodological design of this work.

Conversational agents have proven efficient in automating procedures and inquiries, contributing to the optimization of organizational processes (Rodriguez & Deudor, 2022). However, simply implementing these systems is insufficient without evaluation mechanisms to assess their performance in terms of both linguistic quality and operational efficiency.

In this regard, the specialized literature has identified various metrics applied to the evaluation of natural language processing in chatbots and generative models. Among the most widely used are ROUGE (Recall-Oriented Understudy for Gisting Evaluation) and METEOR (Metric for Evaluation of Translation with Explicit ORdering), which allow for the analysis of aspects such as accuracy, coherence, and similarity between the responses generated by the systems and the reference human responses (Akter et al., 2022).

Additionally, in the context of chatbots deployed in the cloud, performance metrics have been used that consider factors such as response time, concurrent processing capacity (throughput), and load testing with real or simulated users (Blagec et al., 2021). These metrics allow for the identification of strengths and limitations in the operation of the systems under intensive use conditions, which is essential to guarantee their scalability and stability.

It is worth noting that, in recent years, alternative metrics have been proposed that seek to overcome the limitations of traditional ones. One example is Sem-nCG, a semantically sensitive metric that has shown higher correlations with human judgment when evaluating automated summaries. (Croxford et al., 2025) This type of proposal represents a significant step towards more accurate evaluations that are aligned with human perception.

Finally, in the educational field, intelligent systems have been developed that integrate AI technologies with pedagogical approaches, such as the AIDET Intelligent Tutor System (Then et al., 2024). These types of applications require, in addition to linguistic and performance metrics, indicators that assess the relevance and pedagogical effectiveness of the responses, thus establishing a framework for validating intelligent educational systems and improving their impact on teaching and learning processes.

In conclusion, conversational agents are a constantly evolving technology that combines natural language interaction with process automation. However, their widespread adoption depends on both the advancement of the platforms that support them and the development of reliable metrics to evaluate their quality, performance, and relevance in specific contexts.

3 Development

This work focuses on the development of a conversational agent in Spanish with an integrated tracking module, implemented using Google Dialogflow CX. The development is organized into several key stages, notably the agent's training based on the order and structure of documents. This optimizes intent recognition and improves the understanding of user queries. Furthermore, the system is designed to guide students progressively through the submission and verification of requirements, ensuring efficient, consistent, and accurate interaction within the context of academic tutoring.

The following needs were identified, organizing the requirements that must be met to ensure the correct functioning of the system.

The system must allow for user creation and authentication, enabling the monitoring of user processes through a clear and adaptable interface across different devices. It must also guarantee database updates and availability, integration with the Dialogflow CX API, and efficient real-time response.

Furthermore, it requires compatibility with various platforms, data protection through encryption, and an architecture that facilitates continuous monitoring.

Based on the aforementioned requirements, the key components that the architecture will use were identified, including the external services that the agent must integrate, along with the deployment channels. As a result of the requirements analysis, the following components were defined: a database, a web interface, and a tracking module. These components allow for the establishment of a robust architecture, ensuring that the system operates efficiently, provides accurate responses, and facilitates the management of procedures for users.

The need was identified to implement a conversational agent based on Generative Artificial Intelligence (GAI), capable of interacting in Spanish and adapting to the students' academic context. To ensure its effectiveness, the agent requires a structured training process, which is carried out through the organization and classification of institutional documents, guaranteeing proper recognition of intent and correct interpretation of queries. This training allows the agent to offer coherent, consistent, and personalized responses, thus optimizing the student experience in resolving doubts related to academic and administrative procedures.

The database responsible for storing, organizing, and managing information, recording the history of interactions, user profiles, the status of procedures, and other relevant data; a web interface that will serve as the main channel, accessible from any browser, through which users will interact with the conversational agent and the other components; and finally, a tracking module whose function is to maintain detailed control of all the processes and procedures that users carry out within the platform.

3.1 Agent Development

This section presents the structure and operation of the conversational agent, ensuring that each component is properly integrated. It covers the creation of dialogue flows, the data store, and generators, as well as how these interact within Dialogflow CX.

The dialogue flow was designed to structure user interaction in a clear and efficient way, guiding them through different topics of interest and facilitating access to information. As they progress through the conversation, the system redirects them according to their needs until they reach the final nodes, where Generative AI (GA) is integrated. This allows for more open and personalized responses to specific questions, resulting in a more dynamic and natural interaction without disrupting the initial flow structure. Furthermore, this structure improves the user experience by making navigation more intuitive, preventing conversation blocks, and providing precise, context-based responses, leading to a smoother and more satisfying interaction, as shown in Figure 1.

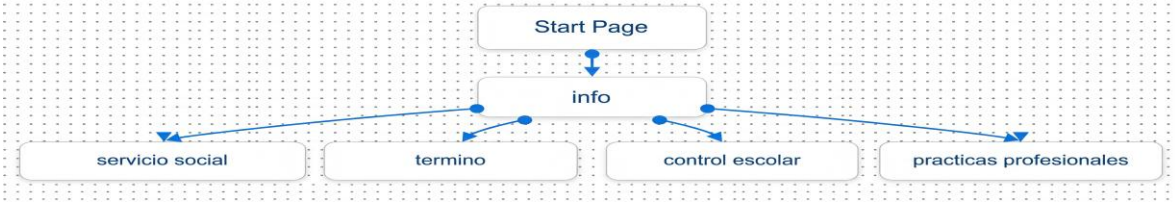

Fig 1. Dialogue flow

To ensure effective interaction, the following elements were considered in the dialogue flow: Nodes (pages) represent points where the conversation changes, including community service, term, student records, homepage, and internships. Intents identify the user's intentions to generate appropriate responses, while entities extract key information such as names or dates. Finally, contexts maintain the continuity of the conversation, ensuring that the agent remembers relevant information within a session.

In this specific case, the initial phase of the conversation is presented, providing the user with an overview of the topics the agent can address. Through natural language processing, the user can select their area of interest, enabling a more flexible and targeted interaction. Once the dialogue reaches a certain point, interaction with the Generative AI (GAI), in this case Gemini, is activated. Gemini is trained to answer specific questions within the previously defined thematic
scope. To ensure the accuracy and relevance of the responses, the GAI has been pre-trained using structured documents, improving its ability to process and understand information.

The creation of this dialogue flow is based on a well-defined and organized structure, using visual or conversational diagrams to represent different scenarios and ensure smooth interaction. Thanks to this approach, the agent can guide the user effectively.

Data warehouse. It is a cloud-based document storage where structured data such as CSV, PDF, HTML, TXT or unstructured data such as JSONL or metadata can be uploaded.

Creating a data warehouse precisely tailored to the needs of the virtual support agent is a fundamental step in optimizing user interaction. By establishing a single source of truth, data redundancy is eliminated, and ambiguity caused by confusing terminology is prevented, which is critical for the accuracy of the response. This centralized repository allows the agent to quickly and efficiently access relevant and validated information, significantly improving the accuracy and effectiveness of each interaction and ultimately providing more personalized and higher-quality support for the user's specific needs.

To date, this knowledge base has been enriched with the initial compilation of 15 key documents covering a wide range of essential administrative and academic topics. This diverse content ensures that the agent can provide detailed assistance on crucial issues, including:

- Social service: Covering everything from the requirements to perform it and the steps to register with the IMSS, to the registration processes, available programs and the final release.
- Title: Complete guides to options, eligibility criteria and the process flow.
- School administration: Including the registration process, the request for official documents, and the process for dropping a subject (temporary, partial, or total withdrawals).
- Academic tutoring: Detailing the tutor assignment, the program's role, and how to request advice or schedule follow-up sessions.
- Frequently Asked Questions (FAQ): A compilation of common questions about academic and administrative procedures for immediate answers.

In addition, the data warehouse integrates a glossary of terms to standardize the language, defining key concepts such as "Credits," "Academic Record," and "Academic Load," thus ensuring uniformity and clarity in all interactions. This structure not only facilitates the agent's operation but also lays the foundation for future expansions and continuous learning.

Figure 2 shows a response generated by a generative AI, in this case Gemini. Its training involved modifying the documents in both structure and order so that it could extract accurate and relevant information to respond effectively to students. A robust cloud-based data storage system allows the agent to quickly access specific details, such as academic requirements and administrative processes. This is exemplified in the presented conversation, where the agent provides detailed information about the credits required to complete community service

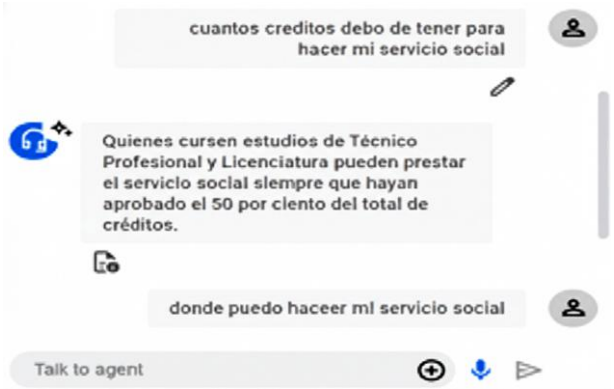

Fig 2. Gemini's example of generative artificial intelligence

To ensure the accuracy and relevance of the responses, the agent was trained using structured documents organized by order and content, as well as a set of representative phrases in Spanish. This training enables the agent to correctly recognize users' intentions and respond coherently within the context of academic tutoring and administrative procedures.

3.2 Agent training

From the data store, comprised of the initial collection of 15 documents covering key topics such as Community Service, Academic Records, Graduation, and Tutoring, a central document was selected to begin the agent's development and training. This choice allowed us to focus on a specific domain, ensuring high accuracy in the initial interactions. This document, used in the experiment, is a PDF file (which we call the base document) containing all the information in Spanish regarding the procedures, characteristics, duration, and requirements for students to complete their community service. It was observed that this document contains, in addition to text, elements such as images, lists, and tables. Some of these elements, such as images containing text and tables, prevented the conversational agent from finding answers to the questions initially posed. The first transformation performed on the base document was to remove all the images and manually extract all the text to add it to a new document, called "Ver1." This new document then contains all the text and tables from the base document, plus the text that was in the images. The tables and numbered and unnumbered lists were not modified. Based on document Ver1, documents Ver2 and Ver3, which are modified versions of the same document, were restructured and reorganized. All lists were numbered, and some processes that follow a specific order were added as numbered lists. These three documents were named as follows (called "Ver1", "Ver2", and "Ver3") and used to feed the intelligent agent. The three documents were modified in their structure and order, but not in their content; that is, no text was added or removed.

The characteristics of the three documents used to feed the conversational system are as follows:

1. Document Ver1. It has 2 pages, with a total of 651 words, it has 5 tables and 5 figures, it does not contain numbered lists, but it does have 1 unnumbered list.
2. Document Ver2. It has 2 pages, with a total of 712 words, it has 1 table, it does not contain figures, it has 8 numbered lists, and it does not have unnumbered lists.
3. DocumentVer3. It has 4 pages, with a total of 764 words, it does not contain tables or figures, it has 8 numbered lists, it does not have unnumbered lists.

Google Dialogflow CX was used as the platform for the conversational agent. This platform allows the use of various LLMs (Learning Management Tools) to create dialogue flows in user or customer service systems. The LLM used for this project was Gemini, due to its ease of compatibility with the platform and the lack of need for complex configurations, as it is also developed by Google. A Google Cloud bucket was used for document storage. The documents used to populate Gemini were the following: a) the process for performing the social service, in its three versions, and the Python code with the metrics.

Questions asked to the intelligent conversational agent

1. ¿Cuál es el link para llenar la información del SICDE?
2. ¿Ya que tengo mis formatos de SICDE que debo de hacer?
3. ¿Dónde puedo hacer mi servicio social?
4. ¿Cuándo me dan mi carta de termino de servicio social?
5. ¿Qué debo de hacer para obtener mi evaluación final?
6. ¿Qué documentos debo de entregar en el paso 2?
7. ¿Al trimestre, qué documentos debo de entregar?
8. ¿Cuántos días tengo para entregar mi informe?
9. ¿Cuándo debo de hacer el informe final?
10. ¿Cuánto tiempo tarda en llegar el certificado de servicio social?

The quantitative analysis applied to the conversational agent's responses was performed using metrics common in natural language processing. The objective was to evaluate the responses generated by different restructurings of the documents provided to the intelligent conversational agent. The responses generated between the different document versions were compared, that is, between the original document and the modified versions. For this evaluation, we used six widely recognized metrics in natural language processing: ROUGE 1, ROUGE 2, ROUGE L, METEOR BERTScore, and Sem-NCG.

A description of these follows.

- ROUGE (Recall-Oriented Understudy for Gisting Evaluation): Measures the overlap of n-grams between the generated text and the reference text. The variants ROUGE-1 (unigrams), ROUGE-2 (bigrams), and ROUGE-L (longest common subsequence) were used (Ganesan, 2018).
- METEOR (Metric for Evaluation of Translation with Explicit ORdering): Evaluates the accuracy and recall of unigrams, also considering semantic correspondence and word ordering (Rosario & Noever, 2023).

BERTScore measures the similarity between the generated text and the reference text based on the contextual embeddings of the tokens (words) generated by the BERT model. (Zhang et al., 2020).

Sem-NCG is an advanced metric that focuses on evaluating the semantic relevance and coherence of the language model's response in the context of the query or prompt (Croxford et al., 2025).

For the ROUGE metric, there are three types, these are the following: ROUGE-1. Evaluates the match of unigrams (individual words) between the generated text and the reference text. It is a simple metric that checks whether the keywords or important concepts of the reference text are present in the generated response.

The ROUGE-1 formula (eq. (1)) is based on the calculation of the recall metric:

ROUGE-1= Ui /U

Where Ui: Incident Unigrams (1)

U: Total number of unigrams in the reference text

This value will be between 0 and 1, where 1 indicates a total word match between the generated text and the reference text. For a conversational agent, a high score on ROUGE-1 indicates that the chatbot includes the most common words.
relevant or fundamental in their response, thus reflecting the main concepts and themes of the user's message, although without considering the word order.

ROUGE-2. Measures the matching of bigrams (pairs of consecutive words) between the generated text and the reference text. The ROUGE-2 score is particularly useful for assessing whether the generated response has basic coherence, since bigrams help maintain a relationship between words and preserve short sequences of ideas.

ROUGE-2 is calculated as follows:

ROUGE-2= Bi B

Where:

Bi: Incident Bigrams

B: Total number of bigrams in the reference text (2)

Similar to ROUGE-1, the value will be between 0 and 1. A value close to 1 suggests that the sequence of ideas and phrases in the generated text closely approximates that of the reference text. In a chatbot, a high score. ROUGE-2 indicates that the sequence of words generated by the agent aligns consistently with the user's intent.

A high score indicates that the order and structure of ideas are similar between the generated response and the source text, which is key to maintaining naturalness and clarity in a conversation. For a conversational agent, a high ROUGE-L score indicates that the chatbot has maintained the conversation structure appropriately, following a similar order and organization of ideas to the source text. This helps the generated responses appear more fluid and structured, improving the user experience. Together, these metrics assess the relevance, coherence, and structure of the generated response. For applications like chatbots, they help verify that the responses capture important concepts, follow the message's logic, and maintain a natural flow of ideas, resulting in a more accurate and satisfying interaction for the user.

BERTScore (Measures Similarity of Meaning)

BERTScore is calculated using the harmonic mean of Recall (R) and Accuracy (P). The final BERTScore formula is:

$$\text{BERTScore} = \text{F1 Score} = (2 \times P \times R) / (P + R)$$

Accuracy (P) measures how relevant the words generated by the model are, comparing each generated word to the best word in the reference. Recall (R) measures how well the model covers the important words in the reference, comparing each word in the reference to the best word in the model.

The BERTScore gives us an average that indicates how close the meaning of the generated response is to the meaning of the expected response. The metric does not use exact word matches, but rather the Cosine Similarity of the meaning vectors (embeddings) of the words.

Sem-NCG (Measures Similarity of Meaning + Order)

Sem-NCG evaluates the quality of a response by comparing it to an ideal response, penalizing it if important information is not presented at the beginning. The formula for calculating Sem-NCG is: Sem-NCG = Model Cumulative Gain / Ideal Cumulative Gain. The Gain in this formula is calculated using the semantic similarity of each phrase in the generated response to the reference, making it semantically aware. The Cumulative Gain is the sum of these scores in the order the model placed them. Dividing by the Ideal Cumulative Gain (the score that would be obtained in the perfect order) rewards models that place the most relevant information (with the highest semantic gain) at the beginning of the response.

In this way, Sem-NCG combines semantic similarity with order of importance, penalizing responses that are redundant or that place crucial information at the end.

3.3 Training methodology

The methodology applied was designed to achieve an objective and controlled comparison of the responses generated by the intelligent conversational agent.

The prompt shown in Figure 1 was designed, in which the i in question i corresponds to the question number, that is, i can take the value 1, 2, ..., 10. This prompt guides the agent to base its answers on the documents provided.

The methodology is summarized as follows:

1. Document upload: Each document (Ver1, 3.6 Qualitative Analysis Ver2, and Ver3) was uploaded separately to a Google Cloud bucket to ensure accessibility and proper processing by the agent. This way, the agent only has access to one version of the document during each experiment run. An experiment consists of asking the agent questions and collecting the responses.
2. Asking questions of the agent: For each experiment, the same prompt structure was used to ask each question of the conversational agent. Each question was asked separately, waiting for a response before asking the next question. In this way, the responses produced by the conversational agent are generated using only one version of the base document, that is, only document Ver1, Ver2, or Ver3.
3. Collection of responses: The responses generated by the agent for each question and for each document version were stored.
4. Quantitative and qualitative analysis: For each outcome of the experiments, the metrics explained above were applied to the stored responses, and the expert manually analyzed each response.
5. Results Analysis: The metrics obtained in each experiment were compared, as well as the expert's observations. Based on this, conclusions were drawn about the conversational agent's performance.

The previous process was repeated identically for each version of the document, applying the same questions and conditions to each one. This ensured that the conditions were identical for all experiments. You are a conversational agent dedicated to being a virtual assistant for students at a public university in Mexico. Answer the following question based solely on the documents provided: question i.

Figure 1: Prompt used for the conversational agent. The responses generated by the conversational agent were evaluated using six quality metrics: ROUGE1, ROUGE2, ROUGE L METEOR BERTScore and Sem-NCG.

The results obtained for each metric and for each set of documents were evaluated to determine if there were differences, and if the metrics were higher for any set. In addition to the quantitative analysis using the metrics described above, a qualitative analysis was applied to the responses generated by the intelligent conversational agent. For this analysis, the assistance of an expert in the problem domain was sought; this expert is the administrative officer responsible for social service and student records processes at the public university campus where the experiments were conducted. The qualitative analysis focused on comparing the responses obtained from the documents.

Ver1, Ver2 y Ver3 were used for each question, evaluating differences in content, style, and accuracy. The expert was asked to verify whether the responses generated by the conversational agent were correct, complete, and in the original order of the documents. In addition, they noted whether the responses were generated with any changes, such as capitalization, parentheses, numbered lists, or bullet points. The number of characters in each response was also counted to compare which document configuration generated shorter or longer responses.

3.4 Quantitative Analysis

The values obtained with the ROUGE 1, ROUGE 2, ROUGE L METEOR, BERTScore, and Sem-NCG metrics are presented in Tables 1 and 2, respectively.

Table 1: ROUGE 1, ROUGE 2 and ROUGE L results

	ROUGE 1			ROUGE 2			ROUGE L		
Pregunta	Ver1	Ver2	Ver3	Ver1	Ver2	Ver3	Ver1	Ver2	Ver3
1	0.7000	0.9120	0.8182	1.0000	1.0000	1.0000	0.5871	0.7368	0.8571
2	0.4079	0.5306	0.4425	0.5957	0.6122	0.6576	0.7273	0.7698	0.7448
3	0.5225	0.6184	0.6018	0.4255	0.4583	0.4583	0.6515	0.7910	0.7761
4	0.7925	0.7925	0.8000	0.7778	0.7857	0.7857	0.4000	0.5484	0.5484
5	0.9563	0.9585	0.9583	0.8511	0.8980	0.8980	0.4396	0.3371	0.4396
6	0.2041	0.0851	0.1633	0.9643	0.9683	0.9643	0.1579	0.1672	0.1579
7	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.0000	0.0000	0.0000
8	0.0816	0.1000	0.0816	0.1875	0.2000	0.1875	0.1875	0.2667	0.1429
9	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
10	0.2857	0.2105	0.2857	0.7500	0.7533	0.7500	0.0090	0.1290	0.1290

Table 2: METEOR BERTScore and Sem-NCG Results

	METEOR			BERTScore			Sem-NCG		
Pregunta	Ver1	Ver2	Ver3	Ver1	Ver2	Ver3	Ver1	Ver2	Ver3
1	0.0000	0.9375	0.0000	0.5871	0.7368	0.8571	0.5010	0.6540	0.8010
2	0.3379	0.6801	0.7921	0.7273	0.7368	0.7448	0.6120	0.6680	0.6790
3	0.3398	0.5763	0.6545	0.6515	0.7761	0.7910	0.5890	0.6850	0.7010
4	0.6523	0.9478	0.9418	0.4000	0.5484	0.5484	0.3550	0.5050	0.5050
5	0.9090	0.9073	0.5324	0.3371	0.4396	0.4396	0.3000	0.4010	0.4010
6	0.1469	0.9496	0.1238	0.1572	0.1579	0.1679	0.1200	0.1250	0.1400
7	0.9815	0.9815	0.0000	0.6500	1.0000	1.0000	0.6000	1.0000	1.0000
8	0.0543	0.0909	0.1925	0.1429	0.1875	0.2667	0.1000	0.1500	0.2200
9	0.9998	0.9998	0.9998	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
10	0.1500	0.8085	0.0758	0.0090	0.1290	0.1290	0.0050	0.1000	0.1000

3.5 Training results

The comparative analysis of the different versions of the document used to train the conversational agent revealed a progressive and significant improvement in the quality of the responses. Version 3 (Ver3) of the document demonstrated the best performance from the agent, achieving the highest scores in the advanced metrics (ROUGE-2, ROUGE-L, BERTScore, and Sem-NCG). This suggests that the optimization performed in Ver3 resulted in a knowledge base that allows the agent to achieve greater structural coherence and organization of ideas (ROUGE-L and ROUGE-2), as well as greater fidelity to the semantic meaning and contextual relevance of the information (BERTScore and Sem-NCG). In contrast, Version 1 (Ver1) of the document consistently ranked as the lowest performer.

Despite the overall superiority of document Ver3, a slight exception was observed in METEOR and ROUGE-1, where Version 2 (Ver2) of the document resulted in marginally better scores. Since these metrics focus on the exact overlap of keywords (unigrams), this suggests that the lexical structure of document Ver2 may have been slightly more effective for recalling exact terms. Nevertheless, the overall results validate that the structure and final content of document Ver3 (which was the agent's input) successfully transferred knowledge with greater semantic accuracy and structural fluency to conversational interaction.

3.6 Limitations and scope of the experiment

The agent was developed using Dialogflow CX and Gemini 1.5 generative artificial intelligence, strategically selected for its native compatibility and advanced functionalities. These features, such as the wide contextual window and the model's inherent reasoning capabilities, proved crucial for processing complex documents and generating accurate responses with low latency, enhancing the agent's functionality in managing procedures. Despite these technological strengths, experimental validation faced limitations inherent to the scope of the domain and the evaluation methodology. The study focused exclusively on analyzing a single base document in its three versions regarding Social Service, which restricts the generalizability of the results to the entire universe of academic procedures. Furthermore, challenges were identified in standardizing the evaluation: the use of a fixed set of ten specific questions extracted from general user queries.

Finally, the participation of several experts in the qualitative analysis, while valuable, introduced potential subjectivity into the validation of the accuracy and consistency of the agent's responses, despite the use of multiple metrics. Advanced quantitative NLP techniques (ROUGE 1, ROUGE 2, ROUGE L, BERTScore, Sem-NCG) were rigorously applied to mitigate this effect when comparing performance between different versions of the documents.

3.7 Training recommendations

This section evaluates the responses to ten questions posed to a conversational agent based on GIA, which uses Spanish-language documents as its primary knowledge base for generating responses. These documents relate to the procedures for completing community service at a public university in Mexico. The documents provided to the conversational agent were altered in their structure and format, but not in content, to identify the effect of these changes on the agent's responses. Applying the six metrics most frequently used in the literature to evaluate GIA, the following recommendations were made.

1. Removal of irrelevant content from tables: All tables were reviewed to ensure they contained only relevant information. Any unnecessary rows were removed, keeping only the essential data. The content of each column was clarified, avoiding any ambiguity and facilitating interpretation.
2. Clear headings in tables: The headings were reviewed to accurately describe the content of each column, avoiding any ambiguity and facilitating interpretation. 3. Image to text conversion: All information contained in images was converted to plain text. Subsequently, the images were removed to optimize processing.
3. Conversion of embedded tables to plain text: In cases where the text included embedded tables, the information was converted to a plain text format, removing the original tables.
4. Delimitation of the topic: The main topic of the document was clearly defined, eliminating any irrelevant information. This ensures that the content remains within the necessary context for the IAG to process the information correctly.
5. Numbering format: The use of periods in list numbering was avoided, as this type of format can interfere with the correct reading and processing of the IAG.

Generators

Generators are components responsible for processing user requests and generating dynamic responses based on up-to- date data. These generators interact with the database or external APIs to obtain relevant information and provide personalized responses based on the context of the conversation. Their proper implementation allows the agent to offer accurate information, improving the user experience and optimizing process automation.

Figure 3 shows the creation of a generator for the welcome message. To create a generator, a message is used that tells the generative AI how to respond. For example, "The agent's name is Masquito. You should give a friendly greeting in fewer than 10 words and explain what kind of procedures you can help the student with, such as school supervision, social services, and psychological counseling."

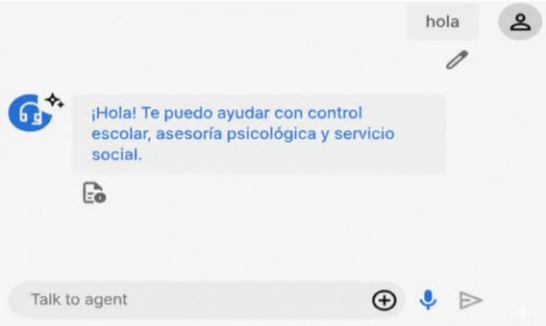

Fig 3. Example of an AI generator dialog flow

A total of 10 generators were developed for different situations, including welcome, farewell, presentation, confirmation, retry, denial, misunderstood entry, request for information, follow-up on procedures and handling off-topic questions.

At this point, the components used in the architecture are developed, which are: the database, the web interface, and the tracking module. Database. The database is related to the processes that the conversational agent must query and update. Its design allows for efficient integration with Dialogflow CX through the tracking module, ensuring that users receive accurate and up-to-date answers about the status of their processes. Furthermore, it facilitates data persistence, interaction traceability, and agent performance optimization.

Figure 4 shows the database diagram and then explains how it works with the agent and the tracking module. When the user requests information about a process, the agent consults the Student Procedures table to identify the current status of their procedures and what stage they are at

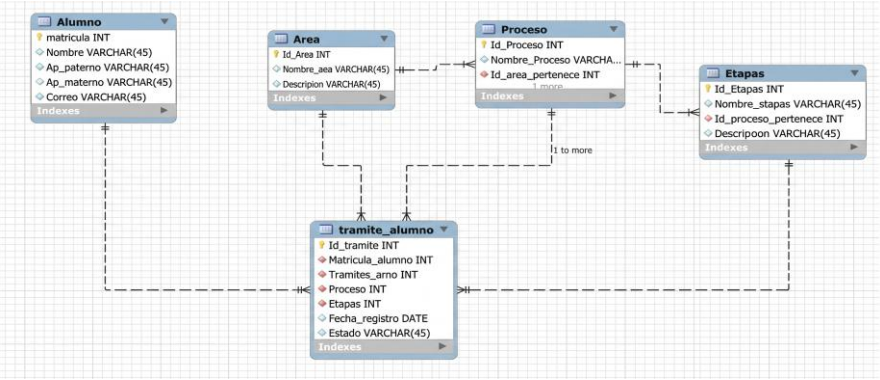

Fig 4. Tracking module database

The Process and Stages table allows the agent to provide specific details about the progress of the process and the next necessary actions. References to the Area tables allow you to determine which department or area is responsible for each process or stage.

The student table facilitates the identification and personalization of interaction with the student. This database model allows the conversational agent to access and update information in real time, providing detailed and accurate tracking of each student's procedures and processes. System Interface. The system interface is designed to provide an intuitive and visually appealing user experience. The main interface features a responsive design compatible with both mobile and desktop devices, enabling quick access to tracking modules, interaction with the conversational agent, and data management. Charts, tables, and interactive elements such as filters and dynamic forms facilitate navigation and access to relevant information, as shown in Figure 5.

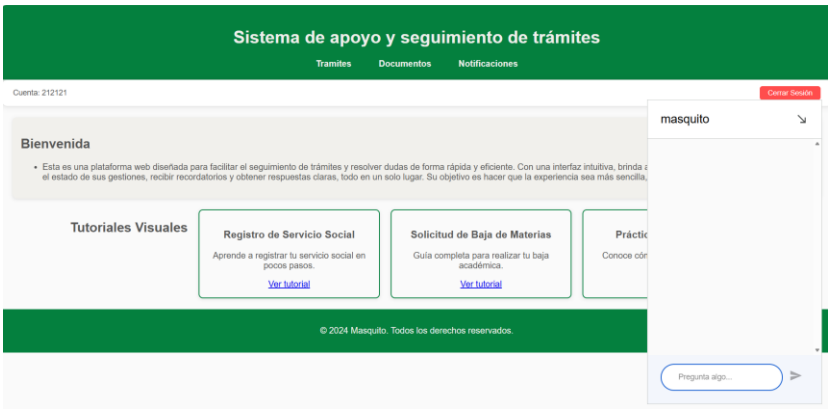

Fig 5. Web interface

Figure 6 shows the web interface designed so that each section of the page is optimized to minimize complexity, featuring organized panels with clear and accessible menus. The chat window for interacting with the agent is integrated into all main pages, ensuring real-time support. The interfaces prioritize personalization, displaying specific information for each user based on their permissions and roles defined in the system.

Tracking module. Allows users to monitor the progress of their procedures clearly and in real time. It includes a visual interface that displays the current status, completed steps, and pending actions through charts and progress bars. Each process is linked to a history that details status changes and their corresponding dates, as shown in Figure 6.

Fig 6. Tracking module

At a technical level, the module is integrated with the system's backend, which manages database updates, and with the conversational agent, which answers quick queries about the status of procedures. Users can interact with the module from any device, accessing their processes in a personalized way thanks to a system of roles and permissions, ensuring security and accessibility.

4 Results

So far, testing has evaluated the connection between the interface and the Dialogflow CX API, as well as the integration between the API and the data store. This progress has validated communication between the system's core elements before proceeding with the full agent implementation. Furthermore, the project manager has reviewed the system and given their approval, confirming that it meets the established requirements. To verify the correct creation of the API, it is possible to observe both the message sent and the response generated by the conversational agent. This process analyzes whether the agent correctly receives the request, interprets it appropriately, and responds as expected, as shown in Figure 7.

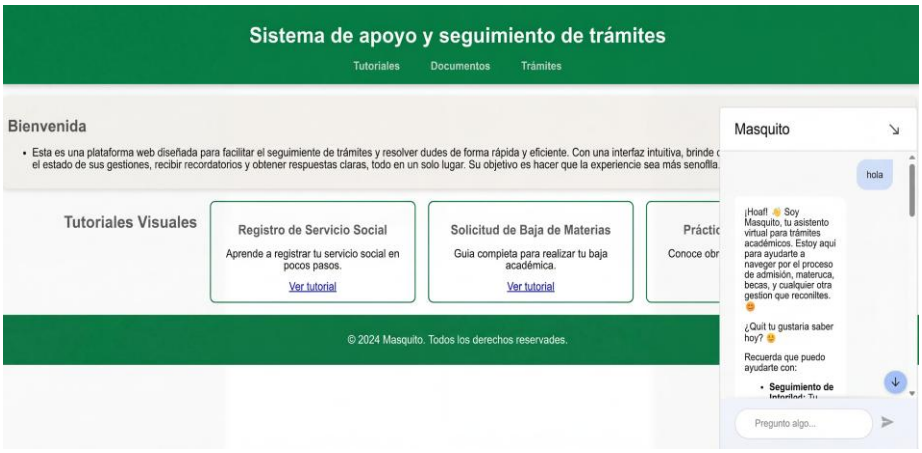

Fig 7. Conversational agent test

In this case, the test was performed by sending a conversation start message, after which the agent should provide a welcome response according to its configuration. In addition to validating that the greeting is appropriate, the test also verifies that the agent is operational, that data is flowing correctly between the client and the server, and that there are no communication errors.

In addition, the system interfaces underwent extensive testing, focusing on login functionality and the creation of new users. It was verified that users could authenticate correctly with valid credentials and that the system handled failed login attempts appropriately, displaying error messages in case of incorrect credentials. Furthermore, the new user registration flow was evaluated, ensuring that the entered information was stored correctly in the database and that the corresponding permissions were generated according to the assigned role.

User data protection is ensured by a combination of robust application and infrastructure security protocols. First, data transmission between your device and the conversational agent server is always protected using TLS/SSL protocols, ensuring that all communication is encrypted and unreadable to third parties. Your login credentials are handled with the utmost care; passwords are stored using one-way hashing with salting, meaning they are transformed into an irreversible code and are never stored in plain text in the database. Furthermore, when using the process management functionalities, the principle of least privilege is applied, guaranteeing that each user only accesses the information and processes strictly necessary for their assigned role. Regarding the sensitive information handled by the conversational agent, the Dialogflow CX platform and its components operate within the secure environment of Google Cloud, adhering to its rigorous security standards. These include data encryption both at rest and in transit, thus ensuring its permanent confidentiality. With regard to Generative Artificial Intelligence (GAI), sensitive data from administrative procedures or academic performance used for its training undergoes anonymization or pseudonymization processes. This practice mitigates the risk that GAI could identify or expose specific personal information, ensuring that all interactions with the agent are fully consistent with established data privacy and ethics policies. Tests were conducted on the system's web interfaces to ensure their proper functioning and usability. Access to the different sections via drop-down menus was verified, ensuring intuitive and smooth navigation. Additionally, the documents section was reviewed to ensure that information was displayed accurately and in an organized manner, as shown in Figure 8. These tests allowed for the identification and correction of potential errors, optimizing the user experience and ensuring the correct display and management of information.

Figura 8. Documents section interface

Student Survey

The conversational agent was evaluated through a survey administered to 50 students, including both current and former students, to measure its usability, user satisfaction, and impact on administrative processes, known as perceived efficiency. The aggregated results indicated high overall satisfaction (92%) and a strong perception of ease of use (90%), demonstrating that the interface is intuitive and has been well-received by its target audience. This evaluation, both qualitative and quantitative, confirms that students perceive the overall experience offered by the platform positively.

Despite these positive results, one area for improvement was identified: the perceived reduction in errors was the lowest metric, reaching 85%. This suggests that, while navigation is straightforward, it is necessary to review and optimize the validation flows or the clarity of the requirements to minimize errors in the submission of procedures. Nevertheless, the analysis of perceived efficiency was compelling, as 95% of students believe the system has reduced the total time required to complete their tasks. Finally, the high recommendation rate of 93% reinforces the success of the implementation from the end-user perspective.

Limitations and challenges

This section analyzes the possible limitations and difficulties when implementing the conversational agent.

The main limitations include the number of supported users, server capacity, and optimizing agent responses. Additionally, challenges include the need to properly configure the VPS, ensure data security, and improve the conversational agent's accuracy to provide a seamless user experience.

For this project, resources from the institution will be used and the implementation will be carried out on its official website.

The estimated monthly cost for a typical conversational agent with up to 1,000 users includes a dynamic website with VPS hosting ($10–$20 USD), Dialogflow CX ($10 USD for up to 10,000 requests), and a domain and SSL certificate ($1 USD), for a total cost of $31–$41 USD. This plan is recommended for sites with a higher volume of interactions and users, requiring increased performance. The main technical requirements for implementation are detailed below.

The technical requirements include a VPS hosting environment capable of handling moderate traffic, with two virtual CPUs and 2-4 GB of RAM. A 20-40 GB SSD is recommended for optimal performance, along with a minimum of 1 TB of monthly bandwidth for 1,000 users. The operating system can be Linux (Ubuntu, Debian, CentOS) or Windows Server, with security measures such as firewalls, SSL certificates, and backups. It must also support Dialogflow CX for API integration and agent communication. This analysis allows us to anticipate potential difficulties and develop strategies to mitigate them. From here, we can continue discussing how to overcome these challenges and ensure the project's successful development, guaranteeing compliance with the established requirements.

5 Discussion

Implementing a process tracking module in conversational agents using Dialogflow CX has a profound impact and a number of significant implications for human-computer interaction. The addition of this module radically transforms the functionality of conversational agents, enhancing their ability to offer a richer and more personalized interaction with users.

This module allows the conversational agent to provide real-time information on the status of different processes, which is essential in contexts where continuous monitoring and frequent updates are necessary. By integrating external databases and APIs, the agent can access user-specific information and tailor responses to individual needs, providing a level of personalization previously unattainable. This precision is crucial not only for enhancing the user experience but also for ensuring the correct structure, order, and format of the documents required for each procedure, as the agent can guide the student on the status of their documentation in real time. This integration improves the relevance of responses and guarantees that the information provided is accurate and up-to-date, enriching the user experience and preventing errors.

6 Conclusions and future work

This paper presents the development of a conversational agent whose architecture incorporates a process tracking module that significantly transforms its functionality. This module allows the agent not only to answer questions or perform basic tasks, but also to provide personalized tracking of each user's specific processes, substantially improving interaction and satisfaction. Integration with the database enables the storage of detailed information about the status of the monitored processes. Thus, when a user initiates a conversation with the conversational agent and expresses interest in a specific process, the agent uses Dialogflow cx to analyze the request and determine the most relevant response, generating a personalized reply. This response is then transmitted to the user through the conversational agent's interface, ensuring seamless communication. The key benefits of this architecture are numerous. First, it provides greater personalization based on the user's specific process information. Second, it facilitates automation, reducing the need for human intervention. Third, it enhances the user experience and satisfaction through real-time updates and personalized responses. Finally, the architecture is scalable, meaning it can adapt to a growing number of users and processes. It also opens the door to future improvements, such as the integration of data science, artificial intelligence, and expansion to other communication channels, such as mobile applications or messaging platforms.

Expressions of gratitude
The authors of this article thank the Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) and the Universidad Autónoma del Estado de México UAEMex Centro Universitario Valle de México for the support provided in carrying out this work

References

Akter, M., Bansal, N., & Karmaker, S. K. (2022). Revisiting automatic evaluation of extractive summarization task: Can we do better than ROUGE? In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 1547–1560). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.findings-acl.122

Amazon Web Services. (2025). *Amazon Lex*. https://aws.amazon.com/lex/

Andia, R. D. A., Chacaltana, G. G. P., & Farfan, R. A. M. (2024). *La inteligencia artificial generativa en el proceso de diseño y producción de recursos educativos digitales para la educación superior: Un estudio de caso del curso piloto «Designing Educational Innovation Projects» de la Universidad Peruana de Ciencias Aplicadas* (Tesis de maestría). Universidad Peruana de Ciencias Aplicadas.

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency* (pp. 610–623). https://doi.org/10.1145/3442188.3445922

Blagec, K., Dorffner, G., Moradi, M., & Samwald, M. (2021). A critical analysis of metrics used for measuring progress in artificial intelligence. *arXiv*. https://arxiv.org/abs/2008.02577

Croxford, E., Gao, Y., Pellegrino, N., Wong, K., Wills, G., First, E., Liao, F., Goswami, C., Patterson, B., & Afshar, M. (2024). Current and future state of evaluation of large language models for medical summarization tasks. *npj Health Systems, 2*(1), 6. https://doi.org/10.1038/s44401-024-00011-2

Ganesan, K. (2018). ROUGE 2.0: Updated and improved measures for evaluation of summarization tasks. *arXiv*. https://arxiv.org/abs/1803.01937

Google Cloud. (2025). *Dialogflow CX architecture*. https://cloud.google.com/dialogflow/cx/docs/concept/architecture

Google DeepMind. (2023). *Gemini: Our largest and most capable AI models*. https://deepmind.google/technologies/gemini

IBM. (2025). *IBM Watson Assistant*. https://www.ibm.com/watson/assistant

Rasa Technologies GmbH. (s. f.). *Rasa Open Source*. https://rasa.com/

Rodriguez, M. M. S., & Deudor, D. D. V. (2022). *Sistema de información para el servicio de posventa inmobiliaria usando chatbot* (Tesis). Universidad Peruana de Ciencias Aplicadas.

Romero, M., Casadevante, C., & Montoro, H. (2020). *Cómo construir un psicólogo chatbot* [Trabajo académico no publicado]. Universidad Autónoma de Madrid.

Rosario, G., & Noever, D. (2023). Grading conversational responses of chatbots. *arXiv*. https://arxiv.org/abs/2303.12038

Then, R., Espinal, L., Marte, E., & Cascante, G. (2024, July). Exploring technologies for intelligent tutoring systems in the development of AIDET: Integrating IAG and advanced pedagogical concepts for their design, overcoming challenges and their potential. In *Proceedings of the 22nd LACCEI International Multi-Conference for Engineering, Education, and Technology*.

Tran, A. (2019). *Artificial intelligence in e-commerce: Case Amazon* (Master's thesis). Centria University of Applied Sciences.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations (ICLR)*.