**HornerosBot**

| **Team Name:** | ***Horneros Team*** |
| --- | --- |
| **Team Members:** | *Ruben Alvarado*  *Jose García*  *Boris Palacios*  *Marcos Egatz*  Derik Daza  *Juan Ignacio Cardona*  *Leonor Hernandez*  *Ignacio Ibarra* |
| **Role:** Team Lead | *Leonor Hernandez* |

**Scope**

Nowadays, companies and institutions must resolve the increasingly accelerated support of their clients and employees. It’s common to see the same kind of support following some kind of pattern that is repeatedly resolved by some resource from the organization. To name a few examples of this, we’ve identified "I did not receive the coffee maker that I ordered on the website" or "How many days of vacation do I have available" or "I have a problem with the payment of my salary" . Resolving a large number of interactions in time requires more human resources. To address these repeated problems in an organization, occuring in between company/client or company/employee, a ChatBot has been proved to be suitable to provide an automated instant messaging processing system. These software artifacts are employed for intelligent, agile and effective communication. In doing so, communication is done through texts which could have arrived syntactically different from another, but they share the same meaning, falling within some category of a known problem. Because of this they can be later structured, achieving a successful outcome to the customer, and providing upstream insight to the organization.

**Description**

The idea is to create a ChatBot that assists people with frequently asked questions (FAQ) and processes requests form permits and vacations, learning from the interactions made through instant text messaging, thus reducing the need for human agent intervention.

**Objective**

The ChatBot must:

* Be able to interact in a normal conversation flow in the case of requests for vacation licenses
* Provide user information by querying a database
* Be trained so that it learns the correct answers.

# Industry

Internal areas such as HR, Retail, Banks, others

# Technology

**Stack**: JavaScript, Dialogflow, Cloud Firestore, Telegram API

# Architecture

The architecture consists in:

1. Dialogflow service:

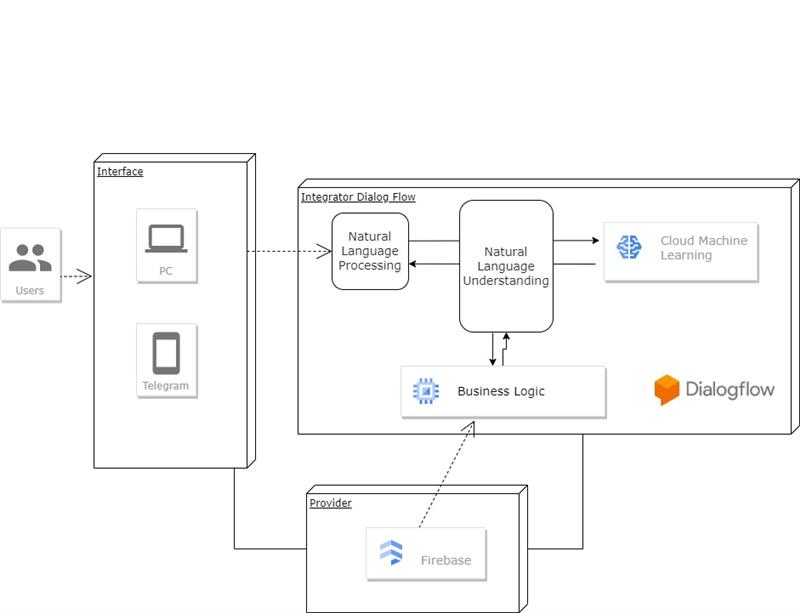
* As interface for natural language processing of input text messages
* For natural language understanding of the text messages
* Handling text messaging interactions.
* The application layer is a server based in NodeJs assigned by Fulfillment.

1. Firebase is used as:

* NoSQL database service

1. Telegram or web browser is used:

* User interface to interact with ChatBot.



# Database

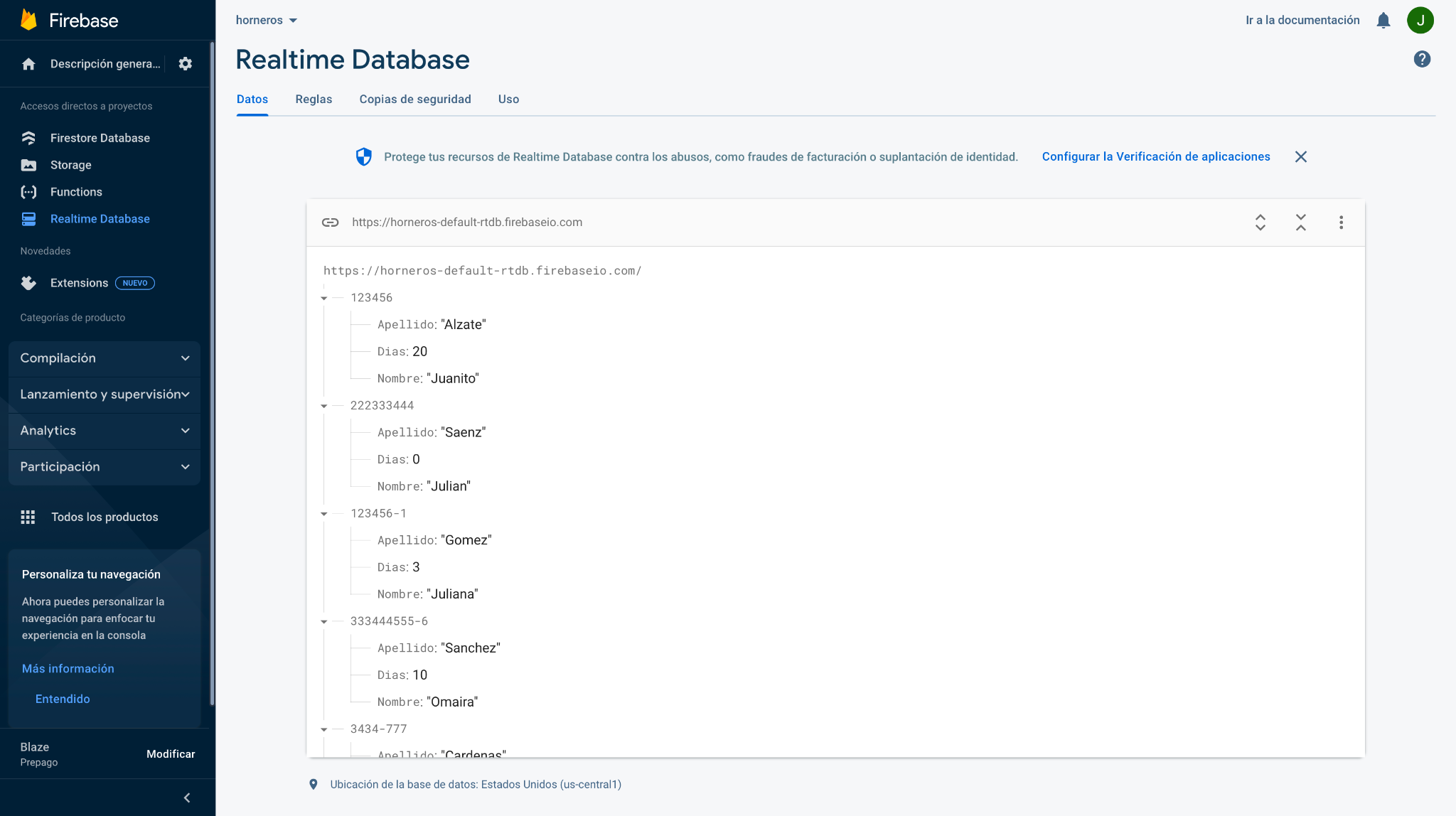
According to the unstructured nature of the different incoming requests from Users, and also given the flexibility needed for our project, it’s straightforward to think of using a non-relational database model. Firebase “RealTime Database” tool provides a non relational db engine solution to our goals.

In our current case, the database collection’s document has the following squema, i.e.:

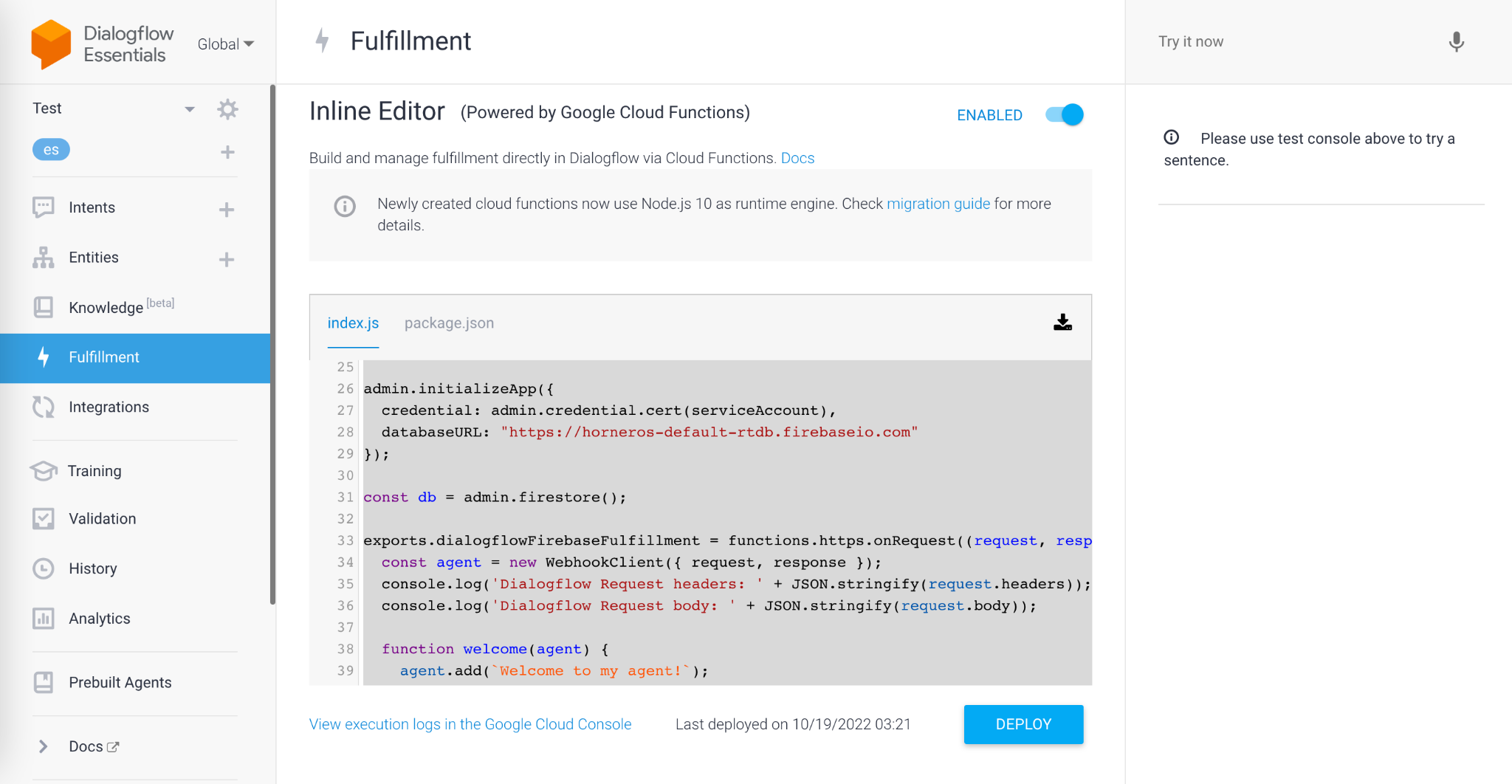
| PersonId : {  Nombre: string,  Apellido:string,  Dias: number  } | 123456: {  Apellido: “Alzate”  Dias:20  Nombre: “Juanito”  } |
| --- | --- |

For our project, there’s implemented a db, consumed by the ChatBot, in the following resource:

DB URL: <https://horneros-default-rtdb.firebaseio.com/>



**Fullfillment**

****

Fulfillment is the layer for the applications to integrate with, by the usage of webhooks we can tell NodeJs to run custom actions like:

* Page entry fulfillment.
* DB operations.
* Routes.
* Event handlers.
* Initial prompts for forms.
* Reprompt handlers for forms.

🎉[CODE](https://github.com/rubenalvarado1987/hornerosBot) OF THE TEAM HORNEROS.

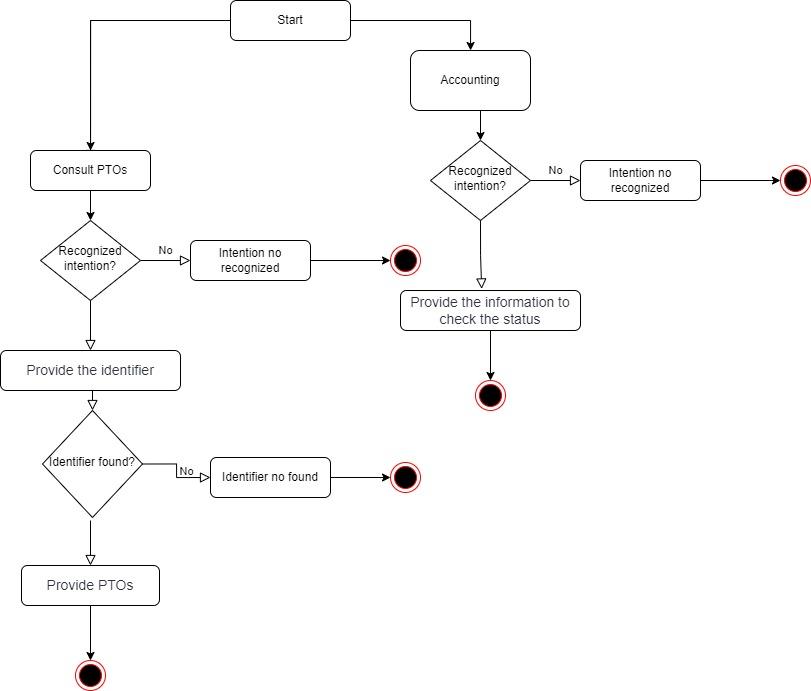
# Conversational Flows

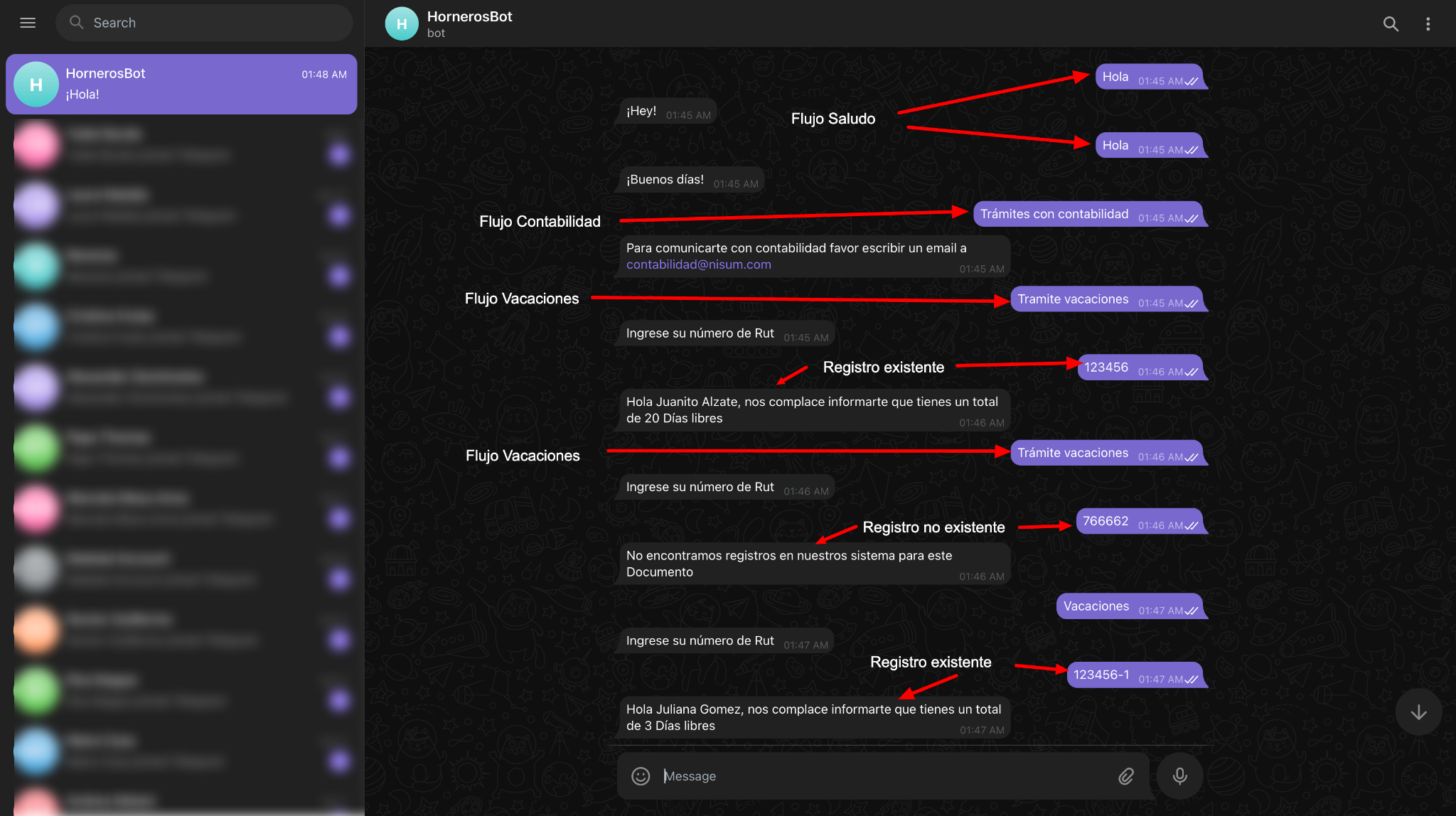
For our intended case, there are currently two main use cases. These two flows resolves two well known problems identified, i.e.:

“Consult PTOs”: Answer the employee how many days it has available.

“Accounting”: Provide point of contact related to Accounting matters.

As depicted in the following flow diagram, here there are the aforementioned 2 flows:





# Demo

Watch demo video in this [link](https://www.youtube.com/watch?v=UvvNSib3_64)

# SWOT

| **Strengths** | Similar to face-to-face experience, saving time doing a previous filter, easy to implement/use |
| --- | --- |
| **Weaknesses** | No empathy, unaccurate |
| **Opportunities** | Opportunity to scale to multiple conversational flows (salaries, formalities, accounting), accessibility improvement (adding audio interactions). |
| **Threats** | Security and confidentiality problems may be considered |

# Future Releases

This POC doesn’t cover aspects related to CI/CD, security, accessibility, monitoring and analytics of the chatbot. These tasks may be considered in future releases.

# Teamwork remarks

Over the 3 days implementation period, the team has been working remotely adopting Agile guidelines, which were adapted to timing and capabilities constraints. Unknowns were tackled down by spike stories, and through a few iterations the Team has been able to reach not only the goal, but also a significant level of continued improvement. Ceremonies proved to be of upmost importance in allowing to share and collaborate knowledge in a professional manner, levering team’s work to reach the goal.

# Team Spec

| Ruben Alvarado | SemiSr Frontend Developer Angular |
| --- | --- |
| Jose García | Senior Backend Developer Java |
| Boris Palacios | Senior Backend Developer Java |
| Juan Ignacio Cardona | SemiSr DevOps |
| Derik Daza | Senior Backend Developer Java |
| Marcos Egatz | Senior Backend Developer Java |
| Ignacio Ibarra | SemiSr. Data Engineer |
| [Leonor Hernandez](mailto:lehernandez@nisum.com)\* | Agile Coach - PO - SM |