

FACULTAD DE INGENIERÍA

Escuela Académico Profesional de Ingeniería de Sistemas e Informática

Tesis

**Implementación de un asistente virtual para la atención
al cliente en Electrocentro S. A. de Huancayo**

Edwing Cristian Artica Llacta

Para optar el Título Profesional de
Ingeniero de Sistemas e Informática

Huancayo, 2020

Repositorio Institucional Continental
Tesis digital



Esta obra está bajo una Licencia "Creative Commons Atribución 4.0 Internacional" .

ASESOR

DR. WAGNER ENOC VICENTE RAMOS

AGRADECIMIENTOS

A mi familia y amigos siempre presentes en mi vida, siento un completo agradecimiento hacia ellos por su constante apoyo y dedicación.

Este logro va dedicado a ellos, debido a que gracias a su constante motivación se ha culminado este proyecto el cual abre nuevas puertas que me permitirán tener nuevas experiencias y oportunidades para futuras metas.

Muchas gracias por todo siempre tendrán mi aprecio y consideración en futuros emprendimientos.

Edwing Cristian Artica LLacta

DEDICATORIA

A mi hermano menor, compañero inquebrantable y cómplice de travesuras por su madurez y apoyo siempre comportándose como el hermano mayor y atento.

A mi abuelo que en paz descansa por motivarme a salir adelante y siempre creer en mí.

A mi familia por siempre tenerme mucha estima y consideración en todo.

Edwing Cristian Artica LLacta

INDICE

AGRADECIMIENTOS	3
DEDICATORIA.....	4
INDICE	5
RESUMEN	13
ABSTRACT.....	14
INTRODUCCIÓN	15
CAPITULO I.....	17
PLANTEAMIENTO DEL ESTUDIO.....	17
1.1. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA	17
1.1.1. Problema General	25
1.1.2. Problemas Específicos.....	25
1.2. OBJETIVOS.....	26
1.2.1. Objetivo General.....	26
1.2.2. Objetivos Específicos.....	26
1.3. JUSTIFICACIÓN E IMPORTANCIA	27
1.3.1. Justificación Práctica	27
1.3.2. Justificación Tecnológica	28
CAPITULO II.....	30
MARCO TEÓRICO	30
2.1. ANTECEDENTES DEL PROBLEMA	30
2.1.1. Artículos Científicos	30
2.1.2. Tesis	32
2.2. BASES TEÓRICAS.....	34
2.2.1. Procesamiento de Lenguaje Natural.....	34
2.2.2. Computación Serverless.....	35
2.2.3. Servidores de Despliegue Continuo.....	36
2.2.4. Restful APIs.....	36
2.2.5. Webhooks	37

2.2.6. Chatbots	38
2.2.7. Machine Learning	39
2.2.8. Asistente Virtual	39
2.2.9. Atención al Cliente.....	40
2.2.10. Electrocentro S.A. Huancayo	40
2.2.11. Scrum	41
2.3. DEFINICIÓN DE TÉRMINOS BÁSICOS.....	46
CAPITULO III.....	50
METODOLOGÍA	50
3.1. METODOLOGÍA APLICADA PARA EL DESARROLLO DE LA SOLUCIÓN	50
3.1.1. Metodología SCRUM.....	50
3.1.2. Personas y roles del proyecto.....	51
3.1.3. Fase de Desarrollo y Entregables.....	51
CAPITULO IV	56
ANÁLISIS Y DISEÑO DE LA SOLUCIÓN.....	56
4.1. IDENTIFICACIÓN DE REQUERIMIENTOS.....	56
4.1.1. Estructura organizacional basado en procesos	56
4.1.2. Mapa del proceso de Atención al Cliente	58
4.1.3. Requerimientos Funcionales	59
4.1.4. Especificación de Requerimientos Funcionales.....	62
4.1.5. Requerimientos no funcionales.....	66
4.1.6. Conformación del equipo de trabajo	66
4.2. ANÁLISIS DE LA SOLUCIÓN	67
4.2.1. Análisis morfológico.....	67
4.2.2. Arquitectura de la Solución	85
4.2.3. Análisis tecnológico.....	90
4.2.3.1. Herramientas de Hardware	90
4.2.3.2. Herramientas de Software	90
4.2.4. Análisis económico	91
4.2.4.1. Estructura de Costos	91
4.3. DISEÑO DE LA SOLUCIÓN	93
4.3.1. Categoría e interacción de Usuarios	93

4.3.2.	Mapa de Navegación del Asistente Virtual	94
4.3.3.	Plantilla de la Página Principal	95
4.3.4.	Plantilla de las páginas secundarias	96
4.3.5.	Modelo Físico de la base de datos	107
CAPITULO V		109
CONSTRUCCIÓN DE LA SOLUCIÓN		109
5.1.	CONSTRUCCIÓN.....	109
5.1.1.	Desarrollo del Agente Conversacional en Dialogflow	109
5.1.2.	Desarrollo del Webhook para Extensión de Funcionalidad del Asistente Virtual ..	112
5.1.3.	Desarrollo de funciones Serverless.....	114
5.2.	PRUEBAS DEL PRODUCTO TECNOLÓGICO	118
5.2.1.	Pruebas de componentes.....	118
5.2.2.	Pruebas de aceptación	125
5.3.	RESULTADOS	126
6.3.1.	DATOS GENERALES DE LOS ENCUESTADOS.....	126
6.3.2.	RESULTADOS OBTENIDOS PRUEBAS DE ACEPTACIÓN	129
6.3.2.1.	Implementación del Servicio que realice consultas al API de Electrocentro	129
6.3.2.2.	Implementación de Consulta de Último Recibo	130
6.3.2.3.	Implementación de Consulta de Facturación de Últimos 4 Meses.....	131
6.3.2.4.	Implementación de Búsqueda de Lugares Cercanos	132
6.3.2.5.	Implementación de Consulta de Pasos Para Reclamos y Números de Contacto de la Empresa	133
6.3.2.6.	Implementación de Consulta de Interrupción de Servicio	134
6.3.2.7.	Implementación de Reporte de Incidencias	135
6.3.3.	RESULTADOS OBTENIDOS POR PRUEBAS DE FIABILIDAD	136
6.3.3.1.	Implementación del Servicio que realice consultas al API de Electrocentro	136
6.3.3.2.	Implementación de Consulta de Último Recibo	138
6.3.3.3.	Implementación de Consulta de Facturación de Últimos 4 Meses.....	139
6.3.3.4.	Implementación de Búsqueda de Lugares Cercanos	141
6.3.3.5.	Implementación de Consulta de Pasos Para Reclamos y Números de Contacto de la Empresa	142
6.3.3.6.	Implementación de Consulta de Interrupción de Servicio	144
6.3.3.7.	Implementación de Reporte de Incidencias	145

6.3.4. RESULTADOS FINALES	147
CONCLUSIONES.....	150
TRABAJOS FUTUROS	153
REFERENCIAS BIBLIOGRÁFICAS.....	155
ANEXOS	160

ÍNDICE DE FIGURAS

Figura 1. Calidad del servicio de Electrocentro 2018	19
Figura 2. Colas en la oficina principal de electrocentro	21
Figura 3. Colas en la oficina principal de electrocentro	21
Figura 4. Colas en la oficina principal de electrocentro	22
Figura 5. Banner informativo de lugares de pago colocado en la oficina de Electrocentro	23
Figura 6. Reclamos de servicio eléctrico Electrocentro 2018.....	24
Figura 7. Captura de la página web de Electrocentro sección reclamos	25
Figura 8. Horario de atención Página Facebook oficial de Electrocentro	28
Figura 9. Arquitectura de un chatbot.....	39
Figura 10. Organigrama estructural de Electrocentro	57
Figura 11. Mapa del proceso de atención al cliente a través del Asistente Virtual.....	58
Figura 12. Arquitectura de la solución	89
Figura 13. Interacción de usuarios asistente virtual	93
Figura 14. Mapa de Navegación Asistente Virtual Ada Electrocentro	94
Figura 15. Asistente Virtual Pantalla Principal.....	95
Figura 16. Asistente virtual listado de opciones	96
Figura 17. Listado de Opciones del Asistente Virtual.....	97
Figura 18. Listado de Opciones del Asistente virtual	98
Figura 19. Menú de opciones rápidas	99
Figura 20. Consulta de recibos del Asistente.....	100
Figura 21. Resultado de consulta de recibos.....	101
Figura 22. Consulta de lugares de pago	102
Figura 23. Enviar ubicación actual	103
Figura 24. Listado de lugares más cercanos	104
Figura 25. Información de reclamos	105
Figura 26. Números de reclamo	106
Figura 27. Consulta de interrupciones	107
Figura 28. Diagrama de Base de Datos para el Asistente Virtual	108
Figura 29. Perfil del Agente en Dialogflow	110
Figura 30. Listado de intenciones del Agente en Dialogflow	111
Figura 31. Integración con Facebook Messenger	112
Figura 32. Activación de cumplimiento para de intención.....	113
Figura 33. Integración de cumplimiento con servicio serverless.....	113
Figura 34. Aplicación serverless desplegada en Netlify	114
Figura 35. Listado de funciones serverless del proyecto	115
Figura 36. Consulta al Servicio REST API de Electrocentro.....	116
Figura 37. Consulta a Servicio del Asistente Virtual con Data Formateada	117
Figura 38. Género del participante	126
Figura 39. Edad promedio del participante.....	127
Figura 40. Lugar de procedencia.....	128
Figura 41. Frecuencia de uso del asistente	128

Figura 42. Funcionalidades más usadas	129
Figura 43. Información brindada acerca de los recibos de los clientes	130
Figura 44. Información brindada de la Facturación de los últimos 4 meses	131
Figura 45. Información brindada acerca de los lugares de pago cercanos	132
Figura 46. Información de pasos para los reclamos	133
Figura 47. Información de números de contacto	134
Figura 48. Interrupciones de servicio.....	134
Figura 49. Reportes de incidencias	135
Figura 50. Canal de referencia.....	136
Figura 51. Test de fiabilidad webhook de consultas al API Principal	137
Figura 52. Test de fiabilidad Api Consulta del último recibo.....	138
Figura 53. Prueba de fiabilidad api consulta facturación últimos 4 meses	140
Figura 54. Prueba de fiabilidad api consulta lugares de pago cercano.....	141
Figura 55. Prueba de fiabilidad api consulta pasos para reclamos y números de contacto	143
Figura 56. Prueba de fiabilidad api consulta interrupción de servicio.....	144
Figura 57. Prueba de fiabilidad api reporte de incidencias	146
Figura 58. Valoración de la utilidad del asistente.....	147
Figura 59. Resultados pruebas de fiabilidad Servicios del Asistente virtual	148
Figura 60. Resultados pruebas de fiabilidad códigos de respuesta obtenidos.....	149

ÍNDICE DE TABLAS

Tabla 1. Horario de atención de las oficinas de Electrocentro	22
Tabla 2. Personas involucradas en el proyecto del Asistente Virtual	51
Tabla 3. Fases de desarrollo scrum	51
Tabla 4. Requerimientos Funcionales	59
Tabla 5. Especificación de Requerimientos Funcionales.....	62
Tabla 6. Requerimientos no Funcionales	66
Tabla 7. Equipo de trabajo	67
Tabla 8. HU-RF-001 - Asistente Virtual en Messenger	68
Tabla 9. HU-RF-002 - Consulta de Recibos	69
Tabla 10. HU-RF-003 – Fecha de vencimiento de recibos.....	70
Tabla 11. HU-RF-004 - Consultar interrupciones	71
Tabla 12. HU-RF-005 - Lugares de pago cercano	72
Tabla 13. HU-RF-006 - Pasos para reclamos.....	74
Tabla 14. HU-RF-007 - Deudas anteriores.....	75
Tabla 15. HU-RF-008 - Números de contacto.....	76
Tabla 16. HU-RF-009- Reportar incidencias.....	77
Tabla 17. HU-RF-010 - Números de servicio.....	78
Tabla 18. HU-RF-011 - Saludo del asistente	78
Tabla 19. HU-RF-012 - Despedida de asistente	79
Tabla 20. HU-RF-013 - Listar opciones del asistente.....	80
Tabla 21. HU-RF-014 - Mostrar imágenes animadas	82
Tabla 22. HU-RF-015 - Asistente no comprende intención	83
Tabla 23. HU-RF-016 - Asistente conversación variada	84
Tabla 24. HU-RF-017 - Escoger institución bancaria	84
Tabla 25. Endpoints de la aplicación.....	86
Tabla 26. Endpoints publicados por Electrocentro para la aplicación	87
Tabla 27. Herramientas de Hardware.....	90
Tabla 28. Herramientas de Software	90
Tabla 29. Costos de personal.....	91
Tabla 30. Costos de infraestructura tecnológica	91
Tabla 31. Costos de infraestructura y mobiliario.....	92
Tabla 32. Otros costos.....	92
Tabla 33. Resumen de costos del proyecto.....	92
Tabla 34. Pruebas de Componentes 1	118
Tabla 35. Pruebas de Componentes 2	119
Tabla 36. Pruebas de Componentes 3	120
Tabla 37. Pruebas de Componentes 4	121
Tabla 38. Pruebas de Componentes 5	122
Tabla 39. Pruebas de Componentes 6	123
Tabla 40. Pruebas de Componentes 7	124
Tabla 41. Prueba de fiabilidad webhook de consultas al API Principal.....	137

Tabla 42. Test de fiabilidad consulta del último recibo	139
Tabla 43. Prueba de fiabilidad api consulta facturación de los últimos 4 meses	140
Tabla 44. Prueba de fiabilidad api consulta lugares de pago cercanos	142
Tabla 45. Prueba de fiabilidad api consulta pasos para reclamos y números de contacto	143
Tabla 46. Prueba de fiabilidad api consulta interrupción de servicio	145
Tabla 47. Prueba de fiabilidad api reporte de incidencias	146

RESUMEN

En la presente investigación se implementó el asistente virtual ADA para la empresa Electrocentro S.A., con el propósito de generar una atención más óptima con los clientes ayudando a realizar consultas acerca de sus recibos, historial de deudas, pasos para la realización de reclamos, información de números de contacto, además conocer lugares de pago cercanos al cliente basado en su ubicación actual, reportes de incidencias del servicio eléctrico, entre otros.

Para el desarrollo del Asistente Virtual, se utilizó la metodología SCRUM así como las herramientas Dialogflow para la implementación del agente que se encarga de la detección de intenciones mediante el Procesamiento de Lenguaje Natural y que luego se comunica mediante el uso de webhooks con los servicios desarrollados con una arquitectura de Computación Serverless basada en funciones alojado en Netlify el cual permitió un despliegue continuo de la aplicación, a su vez se conectó con el respectivo endpoint del Restful API de la empresa para la realización de consultas basadas en el intención que se detectó mediante la comunicación con los usuarios.

Como resultado de la implementación, se logró medir el impacto de uso del asistente el cuál fue valorado muy positivamente por el 77.6% de los usuarios que lo considera realmente útil.

Se concluye que el Asistente Virtual para Electrocentro facilita y mejora la atención brindada al usuario optimizando la forma en la que se comunican y se atienden las consultas, además de atenderlas a todas horas del día.

Palabras clave: Chatbots, atención al cliente, electrocentro, asistente virtual, SCRUM.

ABSTRACT

This investigation implemented a Virtual Assistant or chatbot called ADA developed for the company Electrocentro S.A, with the purpose of offering a better customer service to their clients helping them get information about their monthly bills, historical debts, the right steps to make a claim, contact numbers, also find the closest banking agencies to make their payments based on their current location, make reports about a failure on their service and so on.

For the development of the Virtual Assistant, it was used Scrum as a methodology of development and tools such as Dialogflow to develop an agent that understand the intentions of the users using Natural Language Processing and then communicate this intentions to the services implemented based on a Serverless architecture deployed on Netlify, service that helps with a continuous deployment, also all this services make request to a restful API developed by the company that retrieves all the data of the users.

As a result of the implementation, it measured an impact on the way people value the customer service previously offered by the company.

Finally, Electrocentro's virtual assistant ADA improves and facilitates the customer service provided to their clients, optimizing the way they communicate and all their consults are solved.

r, optimizing the way in which consultations are communicated and attended, in addition to attending them at all hours of the day.

Keywords: Chatbots, customer service, electro center, virtual assistant, SCRUM.

INTRODUCCIÓN

Machine Learning es un concepto muy importante en la tecnología actual, muchos nuevos servicios, productos y startups están centrando el núcleo de su negocio principalmente en esta tecnología. Bajo el Machine Learning existen varias sub disciplinas y aplicaciones tales como el Data Mining, Deep Learning, Natural Language Processing, entre otros.

Precisamente el Procesamiento de Lenguaje Natural es una tecnología que está siendo aprovechada por muchas compañías ya que permite reconocer previo a un entrenamiento del modelo de machine learning con un dataset con diversas expresiones, la intención que el usuario puede estar infiriendo y de esta forma ofrecer una respuesta preparada que pueda permitir una conversación más fluida con éste y a su vez poder realizar algunas consultas u ofrecer algunos servicios.

En la actualidad empresas como Google, Facebook, IBM, SAP, Amazon han desarrollado plataformas que realizan el Procesamiento de lenguaje natural, de esta forma se ha vuelto muy sencillo crear agentes que pueden interpretar los mensajes de los usuarios de aplicaciones web y que utilizan procesamiento de lenguaje natural por debajo, de esta forma la creación de modelos y aplicación de Machine Learning es completamente transparente para el desarrollador.

Electrocentro, empresa que se encarga de la generación y distribución de energía eléctrica en la región centro del Perú posee como canales de comunicación con sus clientes su centro de llamadas, página de Facebook y número de WhatsApp atendidos principalmente en horarios de oficina, debido a esto las consultas que se hacen fuera de ese horario no pueden ser atendidas, además muchas de las consultas realizadas a su página de Facebook son respondidas luego de entre 7 y 15 minutos y algunos casos al día siguiente generando así molestias entre los usuarios. Algunas de las consultas de los clientes pueden ser resueltas a través de la página web o un aplicativo móvil recientemente desarrollado pero el inconveniente que se observa es que se requiere de datos móviles o poseer una conexión Wifi.

Debido a ello se desarrolla ADA, el Asistente Virtual de Electrocentro implementado utilizando el Procesamiento de Lenguaje Natural y una arquitectura Serverless desplegado sobre la plataforma de Facebook Messenger que permite a los clientes de la empresa comunicarse a toda hora para la realización de sus consultas acerca de su servicio eléctrico.

Es así como la investigación está compuesta por:

El capítulo I, donde se muestra el planteamiento y formulación del problema. También los objetivos y justificación de la investigación. En el capítulo II, que presenta antecedentes científicos que nos brindan datos previamente expuestos para nuestro tema de investigación. En el capítulo III, se expone la metodología aplicada para el desarrollo de la solución. En el capítulo IV, se aborda sobre el análisis y diseño de la solución y finalmente en el capítulo V se describe el proceso de construcción del producto tecnológico y las pruebas respectivas.

El Autor

CAPITULO I

PLANTEAMIENTO DEL ESTUDIO

1.1. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA

A lo largo de los últimos años empresas Google, Microsoft y Facebook han enfocado grandes esfuerzos en desarrollar plataformas de Interfaz de usuario conversacional que permiten con una gran facilidad la creación de chatbots y bots que se comunican a través de mensajes de texto y voz (1). Estas plataformas utilizan Machine Learning y Natural Language Processing (NLP) para su funcionamiento y entrenamiento.

En nuestro país empresas como el Banco de Crédito del Perú (BCP), la Universidad Continental (UC) y SUNAT han adquirido aplicaciones de chatbots con la intención de mejorar la atención de sus clientes, en el caso del Banco de Crédito del Perú, tienen la aplicación llamada **Arturito**, un asistente virtual que funciona sobre Facebook Messenger y desarrollada por IBM con ayuda de su Plataforma NLP Watson que se encarga de responder consultas de a sus clientes a cerca de sus saldos, movimientos, tarjetas entre otros (2). En el caso de la Universidad Continental posee la aplicación **Contibot** que también funciona sobre Facebook Messenger y el NLP LUIS desarrollado por Microsoft, se encarga de brindar información a los estudiantes acerca de sus

cursos, horarios, notas, asistencias y trámites (3). También se tiene otra aplicación importante que es **Sofía**, chatbot de SUNAT que resuelve consultas relacionadas al Sorteo de Comprobantes de Pago y devolución automática de 5ta categoría, funciona sobre la plataforma web de SUNAT y está desarrollada con el NLP Watson de IBM y con la que SUNAT planea reducir sus colas en un 20% y reducir el costo de las operaciones de los clientes (4). Además, el 2018 Papa John's Perú presentó su chatbot o asistente virtual llamado **Papa Bot** que se encarga de tomar pedidos de los clientes a través de Facebook Messenger (5).

En nuestro país Electrocentro es la empresa proveedora de energía eléctrica en la mayor parte de la región centro del país, actualmente cuenta con alrededor de 739,024 clientes (6).

A largo de los años y con el creciente crecimiento de sus clientes no se ha logrado mantener una eficiente atención a estos, especialmente en cosas sencillas y por las que se pierde mucho tiempo en colas generando un malestar entre sus clientes tales como:

a) Molestias por el servicio, largas colas para realizar consultas de recibos, facturación y lugares de pago

Como parte de la medición de su servicio eléctrico Electrocentro realiza mediciones constantes para verificar su correcto funcionamiento, en la Figura 1 se puede visualizar la calidad del servicio ofrecida por la empresa, el año 2018 se realizaron 2288 mediciones de baja tensión de las cuales se obtuvo como resultado un 53.89% de buena calidad, 22.64% de mala calidad es decir 518 mediciones y fallidas 537, alrededor del 23.47% (7), de ello se podría rescatar que solo poco más del 50% de su servicio es correcto reflejándose así disconformidad del servicio por parte de los usuarios. Por otro lado, cuando un cliente desea realizar consultas acerca de su servicio eléctrico tiene que acercarse a las oficinas de Electrocentro, actualmente Electrocentro habilitó una línea telefónica para la realización de estas consultas, pero

a mayoría de los clientes la desconoce y sigue acercándose a las oficinas, lo cual supone una pérdida de tiempo, además de un incremento de las colas.

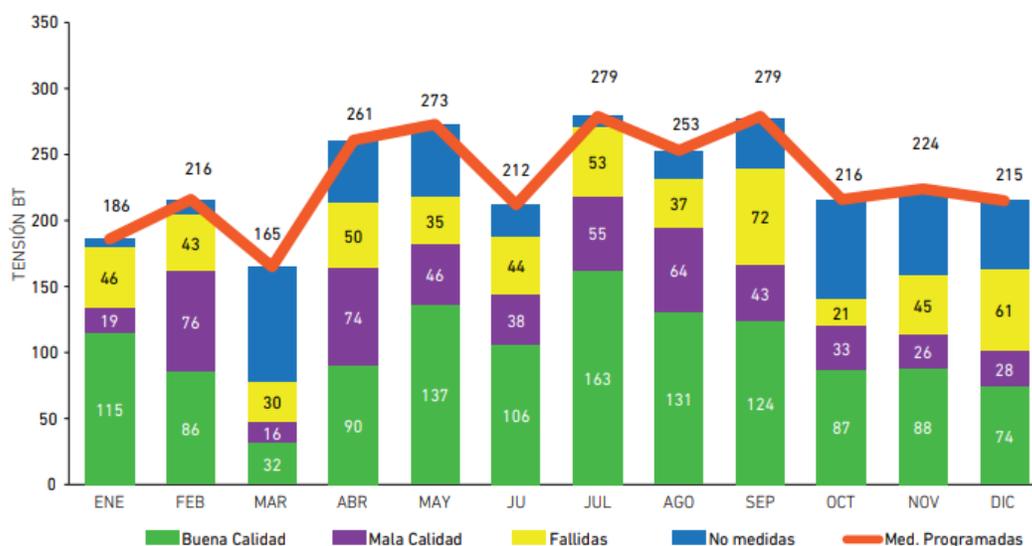


Figura 1. Calidad del servicio de Electrocentro 2018
Fuente: Memoria Anual de Electrocentro 2018

Para la consulta de recibos se tiene que esperar a que este documento llegue al hogar del cliente y de esta forma se pueda conocer el monto de su recibo y si por alguna razón este documento se perdiera el cliente no sabría exactamente cuál es el monto que debería pagar hasta acercarse a las oficinas, además Electrocentro recientemente habilitó un servicio en su página web que permite conocer este monto, pero igualmente es prácticamente desconocido por la mayor cantidad de sus clientes.

En el proceso de consulta de deudas el cliente puede revisar la deuda pendiente que tiene con Electrocentro sólo en el recibo de luz, pero si este no llegara a su destino o el mismo cliente lo perdiera, este tendría que acercarse a las oficinas, generando molestias.

La continuidad del suministro es muy importante para medir la calidad del servicio eléctrico. Las interrupciones del servicio afectan directamente a los clientes e influyen en las múltiples actividades que estos realizan, ya sean comerciales, industriales o domésticas (7). La medición de interrupciones anuales se realizan mediante 2 indicadores SAIDI (duración) y SAIFI (frecuencia) (7).

En el año 2018 el SAIDI alcanzó un valor de 39.40 horas en promedio, de las cuales 8.30 horas que equivalen al 21.07% son responsabilidad de los generadores y transmisores de fuerza mayor que son justificadas por la autoridad, el restante 31.10 horas es decir el 78.93% son responsabilidad de Electrocentro (7).

Además en el año 2018 el SAIFI alcanzó un valor de 17.50 veces en promedio por suministro, de ese total 3.10 veces (17.71%) es de responsabilidad de generadores y transmisores los cuales son declaradas fundadas por la autoridad, el restante 14.40 veces (82.29%) es responsabilidad de Electrocentro (7). Además, tomando en cuenta los cortes programados en algún área de la ciudad, estos se comunican a través del periódico y anuncios en las emisoras radiales locales, pero muchas veces estos anuncios pasan inadvertidos por gran parte de la población ya que algunos de los ciudadanos no consumen este tipo de servicios, por esta razón además de estos medios la empresa permite consultas a través de su web, la cual no es muy utilizada por los usuarios, además que requiere un previo registro e identificación. En la Figura 2 y Figura 3 se observan las colas realizadas para la realización de pagos en las oficinas de la empresa. Además, en la Figura 4 se pueden apreciar la cola para la realización de reclamos acerca del servicio eléctrico.



Figura 2. Colas en la oficina principal de electrocentro.
Fuente: Elaboración propia. Tomada el 14 de octubre de 2019 por el autor



Figura 3. Colas en la oficina principal de electrocentro. *Fuente:* Elaboración propia. Tomada el 14 de octubre de 2019 por el autor



Figura 4. Colas en la oficina principal de electrocentro.
Fuente: Elaboración propia. Tomada el 14 de octubre de 2019 por el autor

b) Desconocimiento de Centros de Pago Alternativos

Los principales lugares de pago para los recibos de luz de los clientes son:

Tabla 1. Horario de atención de las oficinas de Electrocentro

Ubicación	Horario de Atención
Jr. Pichis Nro. 255 – Huancayo	Lunes a viernes: 8:00 a 18:30 hrs. Sábados y domingos: 8:00 a 18:00 hrs.
Av. Ferrocarril Nro. 622 - El Tambo	Lunes a viernes: 8:00 a 13:00 hrs; 14:30 a 18:30 hrs. Sábados y domingos: 8:00 a 18:00 hrs.
Av. Bruno Terreros Nro. 401 - Jauja	Lunes a viernes: 8:00 a 17:30 hrs. Sábados y domingos: 8:00 a 12:30 hrs.
Jr. Alonso Mercadillo Nro. 261 - Chupaca	Lunes a viernes: 8:00 a 17:30 hrs. Sábados y domingos: 8:00 a 12:30 hrs.
Av. Ramón Castilla Nro. 1012 - Concepción	Lunes a viernes: 8:00 a 17:30 hrs. Sábados y domingos: 8:00 a 12:30 hrs.

Fuente: Tomado de la página oficial de Electrocentro

La mayoría de los usuarios únicamente conocen estos lugares y por ende estos llegan a estar muy congestionados, especialmente los fines de mes cuando todos los clientes tratan de pagar sus recibos, a eso se pueden agregar los clientes que se acercan a las oficinas a realizar consultas, debido a todo esto se genera un malestar en el público, para remediar este problema Electrocentro habilitó otros medios de pago de recibos de luz, actualmente se pueden realizar los pagos en agentes de los bancos de la Nación, Interbank y la Caja Huancayo, entre otros, pero los clientes desconocen su ubicación debido que la única forma que utilizan para promocionarlo es un banner ubicado en la entrada de sus instalaciones tal y como se aprecia en la Figura 5.



Figura 5. Banner informativo de lugares de pago colgado en la oficina de Electrocentro.

Fuente: Elaboración propia. Tomada el 27 de setiembre de 2019 por el autor

c) Dudas con respecto a Reclamos

El año 2018, tal y como se observa en la Figura 6, tuvo como resultado un registro de un total de 31932 reclamos, y como promedio mensual 2661, lo cual representa alrededor de 33.45 por cada 10000 clientes. El número de reclamos fundados es de

9578 que representa el 29.99% del total, con un índice de 10.04 reclamos fundados por cada 10000 clientes, sumado a esto se tiene mucha desinformación acerca de la manera práctica para realizar sus reclamos, gran parte de los clientes tienen dudas acerca de los pasos necesarios para realizar los reclamos correspondientes a sus servicios y realmente no conocen un canal por el cual puedan informarse de los pasos para realizar sus reclamos más que acercarse directamente a las oficinas de la empresa, en la Figura 7 se muestra uno de los medios para reclamos al cual se puede acceder a través de la web de la empresa.

Mes	Año 2018				
	Reclamos	Reclamos fundados	Número de clientes	Índice de reclamos x cada 10 000 clientes	Índice de reclamos fundados x cada 10 000 clientes
Enero	2 291	769	781 531	29,31	9,84
Febrero	1 929	766	783 862	24,61	9,77
Marzo	2 312	814	787 233	29,37	10,34
Abril	3 498	972	789 617	44,30	12,31
Mayo	3 055	827	792 363	38,56	10,44
Junio	2 887	829	794 586	36,33	10,43
Julio	2 604	890	797 233	32,66	11,16
Agosto	2 971	957	799 301	37,17	11,97
Setiembre	2 761	858	801 772	34,44	10,70
Octubre	2 633	736	804 759	32,72	9,15
Noviembre	2 923	710	805 775	36,28	8,81
Diciembre	2 068	450	807 495	25,61	5,57
Total	31 932	9 578	807 495		
Promedio	2 661	798		33,45	10,04

Figura 6. Reclamos de servicio eléctrico Electrocentro 2018.

Fuente: Tomado de (7)

Figura 7. Captura de la página web de Electrocentro sección reclamos.
Fuente: Tomado de la página oficial de Electrocentro

El desarrollo de la aplicación beneficiaría a los clientes de la empresa, debido a que no necesitan aproximarse a sus oficinas a realizar sus consultas, en lugar de eso sólo necesitan acceder al Asistente virtual a través de Facebook Messenger a través de su Celular o Computadora y empezar a realizar sus consultas de manera rápida y sencilla.

1.1.1. Problema General

¿Cómo desarrollar un asistente virtual para la atención al cliente en Electrocentro Huancayo?

1.1.2. Problemas Específicos

- a) ¿Cómo obtener los datos por cliente de la base de datos de Electrocentro?
- b) ¿De qué manera pueden los clientes conocer el monto de su último recibo a través del asistente virtual?

- c) ¿De qué manera pueden los clientes conocer su facturación de los últimos 4 meses a través del asistente virtual?
- d) ¿De qué forma pueden los clientes conocer los lugares más cercanos en los que pueden realizar los pagos de sus recibos de consumo eléctrico a través del asistente virtual?
- e) ¿De qué manera pueden los clientes saber cuáles son los pasos para hacer un reclamo y los números de contacto de la empresa a través del asistente virtual?
- f) ¿De qué forma pueden los clientes saber si se efectuaran interrupciones del servicio eléctrico a través del asistente virtual?
- g) ¿De qué forma pueden los clientes reportar incidencias en su servicio de Electrocentro a través del asistente virtual?

1.2. OBJETIVOS

1.2.1. Objetivo General

Desarrollar un asistente virtual para la atención al cliente en Electrocentro Huancayo.

1.2.2. Objetivos Específicos

- a) Implementar un servicio que realice consultas REST al API de Electrocentro para obtener los datos de cada cliente.
- b) Implementar la opción de consulta de último recibo en el asistente virtual.
- c) Implementar la opción de consulta la facturación de los últimos 4 meses en el asistente virtual.
- d) Implementar la opción de búsqueda de lugares cercanos de pago en el asistente virtual.

- e) Implementar la opción de que brinde información sobre los pasos para realizar reclamos y los números de contacto en el asistente virtual.
- f) Implementar la opción de consulta de interrupciones del servicio eléctrico en el asistente virtual.
- g) Implementar la opción de reporte de incidencias del servicio eléctrico de los clientes.

1.3. JUSTIFICACIÓN E IMPORTANCIA

1.3.1. Justificación Práctica

La implementación del Asistente Virtual se realiza porque existe la necesidad de mejorar la atención al cliente que realiza la empresa Electrocentro, ya que los canales actuales no son suficientes, debido a no estar disponibles todo el día tal y como se aprecia en la Figura 8, y al estar sujetos a la atención por parte de un personal el cual tiene sus propias limitaciones. El Asistente Virtual permite solucionar esos problemas al encontrarse disponible todo el tiempo al ser un ente ubicuo y por ende puede atender a los clientes en todo momento, además de disponer una amplia funcionalidad, y al no estar sujeto a las limitaciones humanas puede atender a un amplio número de clientes al mismo tiempo, tarea que por caso contrario requeriría emplear más colaboradores.



Figura 8. Horario de atención Página Facebook oficial de Electrocentro
Fuente: Tomado de la página oficial de Facebook de Electrocentro

1.3.2. Justificación Tecnológica

En la actualidad los Asistentes Virtuales (Chatbots) se están volviendo entes imprescindibles para los negocios en particular los que lidian con gran cantidad de clientes, en nuestro país esta tendencia ha sido adoptada por grandes empresas como el Banco de Crédito del Perú, SUNAT, Universidad Continental por mencionar algunos, pero al mismo tiempo no son muchas las empresas que se atreven a implementar este tipo de tecnología además de no tener muchos profesionales capacitados en el tema. Cabe mencionar que en la región Junín son sólo 2 empresas que han apostado por innovar tecnológicamente para desarrollar sus Asistentes Virtuales que valiéndose de tecnologías como Machine Learning y lenguajes modernos compatibles con una arquitectura serverless como lo es Javascript. Contibot es uno de ellos, desarrollado por la Universidad Continental que utiliza tecnologías novedosas como la nube de Azure, el sistema de procesamiento de lenguaje natural como servicio LUIS.AI de Microsoft y Node JS, lenguaje que está adquiriendo una gran popularidad entre diversas empresas dedicadas al desarrollo web. El segundo Asistente Virtual de la región Junín es ADA desarrollado para

Electrocentro que precisamente al igual que su predecesor utiliza tecnologías nuevas tales como Dialogflow, servicio desarrollado por Google para el procesamiento de lenguaje natural, Netlify, servicio para permite el despliegue de aplicaciones serverless para el despliegue en este caso de los webhooks que conectan el agente procesador con la página de Facebook y con los servicios web de Electrocentro. Cabe destacar que la aplicación se encuentra en producción y brindando la atención a los clientes de la empresa.

CAPITULO II

MARCO TEÓRICO

2.1. ANTECEDENTES DEL PROBLEMA

2.1.1. Artículos Científicos

- a. Según el artículo científico de (1) titulado “**Building a Serverless Messenger Chatbot**” aborda el problema de cómo construir un asistente con arquitectura serverless que pueda permitir a los usuarios estar al tanto de las últimas noticias de su interés. Propuso la construcción de un chatbot para Messenger con arquitectura serverless que permita conocer las últimas noticias de su interés desarrollado mediante el uso de la metodología el Proceso Unificado Ágil, obteniéndose como resultado que la arquitectura funcionó como se esperaba, permite fácil extensión, a pesar de tener varias capas funcionando no influyó en su desempeño. Se concluyó que el chatbot desarrollado para Messenger resultó ser exitoso, además se planea que en futuras versiones brindar mejoras en el procesamiento de lenguaje natural para añadir mayor funcionalidad al chatbot.

De esta forma el mencionado trabajo de investigación contribuye a la presente tesis porque aporta un diseño de la arquitectura que pueden tomar aplicaciones de chatbot, así como su escalabilidad.

- b. El artículo científico (8) titulado **“Vote Goat: Conversational Movie Recommendation”** afronta el problema de cómo implementar un chatbot que pueda realizar recomendaciones de películas, para ello se propuso la implementación de un chatbot utilizando Dialogflow que permita recomendar películas desarrollado mediante la metodología el Proceso Unificado Ágil, obteniéndose como resultado que el chatbot fue desplegado en su totalidad utilizando la plataforma de Google Assistant y pudiendo así lograr recomendaciones a través de las plataformas Youtube y Google Play Movies. Finalmente se concluyó que el chatbot provee un enfoque conversacional a un sistema de recomendación de películas basado en tareas.

El mencionado trabajo de investigación aporta a la presente tesis porque aporta una detallada arquitectura, así como el enfoque que debe tener el agente de chatbot con respecto al usuario, además del manejo de diálogos.

- c. Según el artículo científico (9) titulado **“Bottester: Testing Conversational Systems with Simulated Users”** Aborda el problema acerca de cómo implementar una herramienta de testeo de agentes conversacionales. En esta investigación se propuso la implementación de una herramienta de testeo de agentes conversacionales de forma automatizada usando la metodología de Proceso Unificado Ágil, obteniéndose como resultado una herramienta que permite testear agentes conversacionales enfocándose en las respuestas correctas e incorrectas además del tiempo de respuesta para identificar preguntas que toman un tiempo significativo, además se realizó una validación y limpieza de errores de escritura al momento de ingresar los textos. Luego se concluyó que testear un chatbot no es trivial debido a la amplia posibilidad de textos a ingresar que existen que representan la misma intención, cada dialogo es interactivo lo que significa que es probable que nunca más se vuelva a repetir.

El mencionado trabajo de investigación aporta a la presente tesis porque aporta conceptos generales acerca de la testeabilidad de un asistente virtual al momento de realizar un test automatizado, así como el testing que puede realizarse a través de usuarios reales.

- d. El artículo científico (10) titulado **“Chatbot as an Intermediary between a Customer and the Customer Care Ecosystem”** brinda información de cómo hacer que un Chatbot puede actuar como intermediario mediante un cliente y un Ecosistema de atención al cliente. De esta forma se propuso la implementación de un chatbot permita que un este actúe como intermediario mediante un cliente y un ecosistema de atención al cliente, donde se investiga que el uso de chatbots como intermediarios no es una tarea trivial considerando varios niveles de complejidad asociados, siendo una de ellas la composición semántica de los términos usados por los usuarios. En conclusión, se determina que con un uso incremental de celulares se tiene un escenario preparado para desarrollar relaciones más significativas entre proveedores de servicios digitales y sus clientes. Esta investigación aporta a la presente tesis porque con los conceptos generales acerca de cómo un asistente virtual puede llegar a tener una mayor y relevante colaboración e interactividad mediante los clientes y el ecosistema digital que tenga una empresa.

2.1.2. Tesis

- a. Según (11) en su Tesis titulada **“Diseño de un Sistema Web de Búsqueda Inteligente Conversacional para Ubicación de Empresas y Servicios”**. Aborda el problema que se tiene con el extenso tiempo y la falta de integridad de resultados al buscar empresas y servicios laborales y empresariales. Propuso el diseño de un sistema web de búsqueda inteligente conversacional para la ubicar empresas diseñado mediante la metodología

Russell, obteniéndose como resultado una mayor sencillez y facilidad de uso al realizar búsquedas. En conclusión, el chatbot desarrollado ofrece una búsqueda inteligente y sencilla además de permitir búsquedas más firmes y precisas.

La mencionada tesis aporta al proyecto porque el estudio refleja la alta capacidad de certeza y facilidad que ofrecen los chatbots a los usuarios finales.

- b. Según (12) en su Tesis titulada **“Desarrollo de un Asistente Virtual Utilizando Facebook Messenger para la mejora del Servicio de Atención al Cliente en la Universidad Privada de Tacna en el 2017”**. Aborda la necesidad de tener un asistente las 24 horas del día que pueda brindar información exacta y oportuna a los estudiantes de la universidad. Se propuso el desarrollo de un asistente virtual para la plataforma Facebook Messenger para mejorar el servicio de atención al cliente de la Universidad Privada de Tacna desarrollado mediante el uso de la metodología el Proceso Unificado Ágil. Finalmente se obtuvo como resultado que la implementación del asistente virtual influye de manera positiva en el servicio de atención al cliente además de brindar respuestas oportunas y exactas.

El mencionado trabajo de investigación aporta a la presente tesis porque el estudio muestra la implementación y especificaciones técnicas del asistente virtual.

- c. Según (13) en su Tesis titulada **“Design and Implementation of a Chatbot in the Context of Customer Support”**. Aborda la dificultad que tienen los usuarios al servicio de atención al cliente y su largo proceso de atención. Por ende, se propuso desarrollar una solución de chatbot que sirva de interfaz entre el sistema de atención al cliente actual y los usuarios desarrollado mediante la metodología de Proceso Unificado Ágil, obteniéndose como RESULTADO que la implementación del chatbot funciona para los lenguajes francés, inglés y holandés clasificando tickets de forma correcta. En

consecuencia, se obtuvo que la solución desarrollada permite un fácil despliegue y desarrollo de cambios si son necesarios.

Debido a ello se aporta a la presente tesis porque el estudio muestra el diseño e implementación del chatbot, así como sus etapas de desarrollo y despliegue.

- d. Según (14) en su Tesis titulada **“Chatbots as a Mean to Motivate Behavior Change”**. Afronta el problema el cómo motivar el cambio de comportamiento de las personas con la ayuda de modernas interfaces de usuario. Propuso el desarrollo de una solución de chatbot para motivar a las personas a cambiar su comportamiento y llevar un estilo de vida más sostenible desarrollado mediante el Proceso de la Rueda de Hartson y Pyla, obteniéndose como resultado que la implementación del chatbot permitió que las personas interactúen con el asistente virtual con el que se sentían más relacionados que con sólo información de tipo informativa. Finalmente, en conclusión, se obtuvo que la solución desarrollada puede mejorar la recepción de información que tienen las personas al ser el chatbot una entidad que puede representar un rol humano.

El mencionado trabajo de investigación contribuye a la presente tesis porque el estudio muestra la influencia que puede tener un chatbot al ocupar el rol de una persona.

2.2. BASES TEÓRICAS

2.2.1. Procesamiento de Lenguaje Natural

Procesamiento de Lenguaje Natural (NLP) es un tipo de inteligencia artificial que permite a las máquinas escrudiñar mensajes imitando la habilidad humana de la comprensión de lenguaje. NLP utiliza un conjunto de técnicas de aserción que incluyen lingüística, semántica y reconocimiento de la máquina, todo esto para realizar mejores conexiones de esta forma mejorar su capacidad de comprensión y reconocimiento (15).

Técnicas de NLP han sido largamente aplicadas para la automatización de varias tareas de requerimientos de ingeniería incluyendo síntesis de modelos, categorización, trazabilidad, detección de equivalentes, extracción de información y detección de defectos, entre otros (16).

El uso de NLP se ha vuelto bastante popular en la actualidad es por eso que varios proveedores cloud lo ofrecen como Plataforma como Servicio (PaaS) entre los que tenemos a Dialogflow de Google, LUIS de Microsoft, Alexa de Amazon, Watson de IBM entre otros.

2.2.2. Computación Serverless

Computación Serverless es principalmente definida en el contexto de Funciones como Servicio. Los usuarios registran sus funciones, es decir código, en un proveedor de una nube serverless. Estas funciones son desencadenadas basadas en eventos, como por ejemplo una ruta HTTP que es invocada o un objeto añadido o removido del almacenamiento, entre otros. Cuando una función es desencadenada, el código es cargado en un contenedor donde es ejecutado e aislado en un sandbox. El contenedor puede ser borrado o cacheado para futuros re-usos luego de que la función haya sido ejecutada. Por ende, los contenedores son cruciales para el paradigma serverless. (17)

En la actualidad el termino Serverless se refiere a la nueva generación de PaaS (Plataforma como Servicio) ofrecida por la mayoría de proveedores Cloud, inicialmente promovidas por Amazon Web Services Lambda la cual fue anunciada en el 2014 y recibió un gran crecimiento en su adopción a mediados del 2016. La gran mayoría de proveedores de servicios cloud ofrecen servicios similares como Google Cloud Functions, Azure Functions, IBM OpenWhisk, entre otros. (18)

Por otro lado, un nuevo proveedor de Serverless Functions en Netlify Functions que es precisamente el servicio que es usado para la creación del

webhook del Asistente Virtual, así como las distintas funciones serverless creadas para consumir el API de Electrocentro.

2.2.3. Servidores de Despliegue Continuo

La implementación continua toma un paso adelante comparado con la entrega continua. Mediante la cual, los cambios que pasan por todas las fases del canal de producción son publicados para los clientes. De esta forma no hay intervención humana y solo una prueba errónea o con deficiencias evitará implementar un nuevo cambio sea desplegado en producción.

Despliegue continuo es un excelente medio de acelerar el ciclo de retroalimentación de los usuarios y quitar presión del equipo de desarrollo ya que no hay un día de despliegue. Los Desarrolladores pueden enfocarse en construir el software, y de esa forma ver su trabajo funcionando minutos después de que hayan terminado de trabajarlo (19).

El despliegue continuo es una estrategia para versiones de software para cada porción de código que pasa la fase de testing automático es automáticamente subido al ambiente de producción, haciendo de esta forma que los cambios sean visibles para los usuarios del software.

De esta forma se elimina protecciones humanas ante código sin probar en software funcional. Sólo debería ser implementado cuando el equipo de desarrollo rigurosamente agrega prácticas de desarrollo de “lista para producción” además de pasar por un riguroso testing (20).

2.2.4. Restful APIs

API (Application Programming Interface) en desarrollo web es un programa que permite recuperar información de un servicio. REST (Representational State Transfer) al ser un tipo de API se refiere principalmente es un conjunto de guías y reglas estandarizadas que permiten que 2 o más aplicaciones se comuniquen entre ellas.

En la arquitectura REST la implementación del cliente, así como la implementación del servidor puede ser realizada independientemente sin que uno conozca al otro. Esto significa que la aplicación del cliente puede ser cambiada o actualizada en cualquier momento sin que esto afecte el funcionamiento del servidor. Uno de los paradigmas de REST es que es stateless, es decir no tiene estado, por ende, no tiene que saber nada sobre el estado del cliente y viceversa, de esta manera ambos pueden conversar sin conocer los mensajes previos (21).

Para el funcionamiento de REST el cliente debe realizar llamadas (request) al servidor para de esta manera poder recibir data o poder modificar data si fuera necesario, para hacer request es necesario tener el verbo HTTP a usar, el header que permite definir qué tipo de información será enviada, del mismo modo es necesaria la data que va a ser enviada o los parámetros y finalmente la ruta que responde a los recursos a obtener.

Entre los verbos HTTP que usa REST tenemos:

- GET: que permite obtener data de una determinada ruta
- POST: que permite enviar data para poder ser almacenada en la base de datos
- PUT: que permite actualizar data dado un generalmente un ID
- DELETE: que permite eliminar data dado un ID

Básicamente al realizar todas estas operaciones estas realizando un mantenimiento CRUD.

2.2.5. Webhooks

Es un concepto de API que está creciendo en popularidad a medida que más y más de las cosas que se realizan en la web pueden ser descritas como eventos. Son muy útiles y consumen pocos recursos a la hora de implementar reacciones a eventos.

Un webhook permite a una app proveer información en tiempo real a otras aplicaciones, a diferencia de un típico API en el que se necesita realizar consultas para obtener data (22).

Según (23), webhooks son callbacks HTTP definidas por el usuario y son desencadenadas por algún evento en una aplicación web y puede facilitar la integración con distintas aplicaciones o APIs de terceros.

Para el asistente virtual de Electrocentro fue necesaria la utilización de webhooks para poder extender la inteligencia del agente de Dialogflow, permitiéndole así, poder acceder a APIs externos como el API de los servicios de Electrocentro y de esa manera poder realizar consultas correspondientes a los clientes.

2.2.6. Chatbots

Un chatbot es un software que ha sido programado para proveer con respuestas automáticas para las que fue entrenado dependiendo de la intención que haya captado al momento de interpretar el mensaje del usuario, por lo general son usados dentro de aplicaciones de mensajería instantánea, en la Figura 9 se puede apreciar una arquitectura propuesta.

Como mencionan Baudart y Hirzel (24), un chatbot es un agente conversacional virtual que se comunica con los usuarios a través del lenguaje natural. Recientemente las compañías encuentran beneficioso el ofrecer servicios a sus clientes a través de chatbots, porque soportan diversos canales de entrega tales como páginas web, celulares, etc.

Chatbots están basados generalmente en texto para funcionar sobre plataformas como Messenger, WhatsApp, Telegram, entre otros, y emular una conversación con un humano para brindar servicios informativos, transaccionales y conversacionales (25).

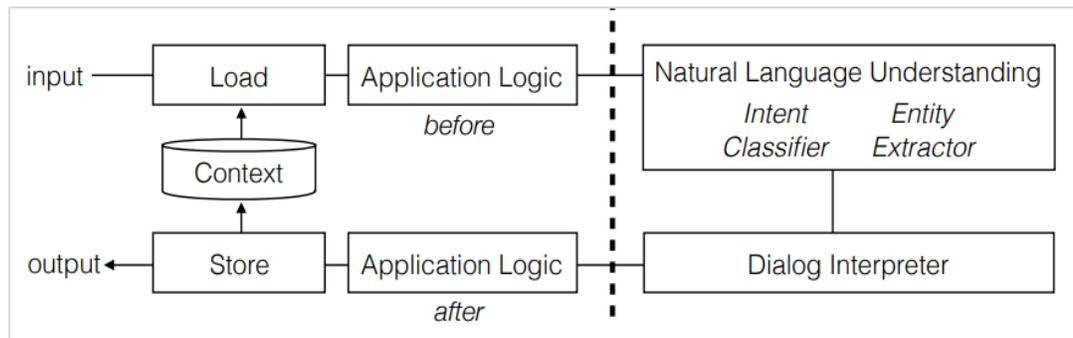


Figura 9. Arquitectura de un chatbot
Fuente: Tomado de (25)

2.2.7. Machine Learning

La tecnología de Machine Learning enseña a las computadoras y sistemas a cómo realizar tareas aprendiendo de datos, en lugar de ser programadas explícitamente. Se utiliza algoritmos sofisticados para aprender dado un gran volumen de datos, por ende, a mayor cantidad de datos mejora su capacidad de aprender. Los algoritmos de machine learning permiten priorizar y automatizar la toma de decisiones, además pueden señalar oportunidades y acciones que deberían de tomarse inmediatamente. Debido a ello se pueden plantear procesos de negocio más inteligentes y con una mejora de resultados (26).

2.2.8. Asistente Virtual

Los Asistentes Virtuales son aplicaciones que asisten a los usuarios a través de la ejecución de diversas tareas computacionales. Generalmente estas aplicaciones en su mayoría están desarrolladas con tecnologías tales como Procesamiento de lenguaje natural además que se pueden encontrar diversas implementaciones de estos en aplicaciones como Samsung Bixby, Apple Siri, Microsoft Cortana, entre otros. El número de usuarios de asistentes virtuales

a nivel mundial espera un crecimiento de 390 millones de usuarios en 2015 a 1.9 billones de usuarios en 2021 (27).

El crecimiento de Asistentes Virtuales en el mercado está dirigida principalmente por el creciente uso de smartphones así como el crecimiento de tráfico de redes sociales, lo que dirigido a la concientización de los beneficios del uso de asistentes virtuales por parte de los usuarios (28).

2.2.9. Atención al Cliente

Es la metodología que emplean diversas empresas para el contacto constante que tienen con sus usuarios o clientes para garantizar de esta forma que los productos o servicios que ofrecen estén correctos. La atención al cliente es un punto clave de contacto cuando los usuarios experimentan problemas en la interacción con los productos o servicios de la empresa (29).

2.2.10. Electrocentro S.A. Huancayo

“Empresa Regional de Servicio Público de Electricidad del Centro S.A. Electrocentro S.A. es la empresa encargada de brindar el servicio público de electricidad dentro de su zona de concesión (Región Centro del país) mediante la distribución y comercialización de energía eléctrica adquirida a empresas generadoras. Electrocentro S.A. fue creada de acuerdo a las leyes de la República del Perú, y su domicilio legal es Jirón Amazonas No. 641, Huancayo, Departamento de Junín. Electrocentro S.A. está presente en siete regiones del país: Junín, Huánuco, Pasco, Huancavelica, Ayacucho, Lima (parte de las provincias de Yauyos y Huarochirí) y Cusco (parte de la provincia de La Convención - VRAEM). Para efectos operativos y administrativos está organizada en seis Unidades de Negocio. El área de concesión en la que la empresa lleva a cabo sus operaciones es de 6 528 km², con 16 401 Km de redes en baja tensión, 17 496 en media tensión, 16 300 Sub estaciones de distribución, 776 Km de Líneas de transmisión y trece Centrales Hidroeléctricas propias y tres Centrales de propiedad de ADINELSA” (30).

2.2.11. Scrum

Scrum es un marco de trabajo creado para el desarrollo y mantenimiento de productos complejos, la cual consiste de diversas partes tales como los roles, eventos y artefactos del Scrum, además de sus reglas que las relacionan. Sus creadores fueron Ken Schwaber y Jeff Sutherland (31). Los pilares que soportan la teoría de la implementación de Scrum son la transparencia, debido a que los aspectos significativos del proceso deben ser visibles para todos los responsables del resultado además de ser definidos bajo un estándar común. Otro muy importante es la inspección, debido a que los usuarios deben inspeccionar frecuentemente los artefactos de Scrum y el progreso hacia el objetivo para detectar variaciones no deseadas. Y finalmente el último pilar es la adaptación, dado que, si se determina que algún proceso se desvía mucho más de los estándares aceptables, el proceso debe ajustarse para que el producto sea aceptable (31).

El Equipo Scrum

El equipo scrum está conformado por un Dueño del Producto (Product Owner), el Equipo de Desarrollo (Development Team) y un Scrum Master, otra característica resaltante es que los equipos son autoorganizados y multifuncionales, debido a esto los equipos eligen la mejor forma de llevar a cabo su trabajo y no intervienen personas externas al equipo, además al ser multifuncionales tienen todas las habilidades necesarias para llevar a cabo las diversas tareas sin depender de otras personas ajenas al equipo (31).

- **El Dueño del Producto (Product Owner)**

Es el responsable de maximizar el valor del producto y el desempeño del Equipo de Desarrollo, por otro lado, la forma en la que se desempeña este rol varía de acuerdo a la organización en la que se encuentre (31).

El Dueño del Producto es el único encargado de gestionar la pila del producto la cual incluye:

- Expresar claramente los elementos de la Pila del Producto.
- Ordenar los elementos en la Pila del Producto para alcanzar los objetivos de la mejor manera posible.
- Optimizar el valor del trabajo que el Equipo de Desarrollo realiza.
- Asegurar la visibilidad y transparencia de la Pila del Producto para todos los miembros del Scrum.
- Asegurar que todos los miembros del Scrum entienden los elementos de la Pila del Producto.

▪ **El Equipo de Desarrollo (Development Team)**

Equipo de profesionales encargados de entregar los incrementos del producto que potencialmente se puede poner en producción al final de cada Sprint. Son los únicos miembros que participan en la creación de los incrementos (31). Los equipos de desarrollo tienen las siguientes características:

- Son autoorganizados.
- Son multifuncionales, es decir todos cuentan con todas las habilidades para desarrollar el incremento.
- No se les asigna ningún título adicional aparte del de desarrolladores.
- No está considerada la existencia de subequipos dentro de los equipos de desarrolladores

▪ **Scrum Master**

Es el responsable de asegurar que Scrum se entienda y adopte. Se encargan de asegurar que el equipo trabaje ajustándose a la teoría, prácticas y reglas de Scrum (31). Algunas de sus funciones son:

- Encontrar técnicas para la gestión efectiva de la Pila del Producto.
- Facilitar los eventos del Scrum según su necesidad y prioridad.
- Ayudar a los equipos a generar productos de alto valor.
- Eliminar impedimentos e imprevistos en el progreso del Scrum.

Eventos del Scrum

Existen diversos eventos definidos en el Scrum. Cada evento tiene un bloque de tiempo o time-box de forma que se tiene una duración máxima. Una vez iniciado el Sprint no es posible acortarse o alargarse.

- **Sprint**

Es un bloque de tiempo definido previamente con mes de extensión máxima durante el cual se crea un incremento del producto final que debe ser utilizable y posiblemente desplegable. Es de recomendación que la duración de cada Sprint sea consistente a lo largo del proyecto., además, los Sprints son consecutivos (31).

- **Planificación del Sprint (Sprint Planning)**

Sirve para planificar todo el trabajo a realizarse a lo largo del Sprint, es creado mediante un trabajo colaborativo del Equipo del Scrum. Este evento tiene una duración máxima de 8 horas. El Scrum Master debe encargarse que este evento se lleve a cabo y que los participantes entiendan su propósito además de recalcar que el Equipo debe mantenerse dentro del bloque de tiempo (31).

- **Objetivo del Sprint (Sprint Goal)**

Es una meta establecida para el Sprint que se alcanzable a través de la implementación de la Pila del Producto. Ayuda al Equipo de Desarrollo a conocer lo que se está construyendo a través del incremento. Es

creado durante la Planificación del Sprint y brinda al equipo cierta flexibilidad con respecto a la funcionalidad a implementar a lo largo del Sprint (31).

- **Scrum Diario (Daily Scrum)**

Es una reunión de dura aproximadamente 15 minutos con la finalidad de que el Equipo de Desarrollo sincronice sus actividades y desarrolle un plan a ejecutar durante las siguientes 24 horas. Es llevado a cabo analizando el trabajo realizado desde el último Scrum diario y haciendo una proyección del trabajo que se podría realizar hasta el siguiente Scrum (31).

- **Revisión del Sprint (Sprint Review)**

Es llevado a cabo al final de cada Sprint para analizar el incremento y adaptar la Pila del Producto si es necesario. A lo largo de la revisión del Sprint, el equipo Scrum y los stakeholders debaten acerca de lo que se hizo durante el Sprint, tomando en cuenta lo anterior y en algún cambio de la Pila del Producto, los asistentes colaboran para determinar los cambios que se podría realizar para optimizar el valor (31). Se incluyen los siguientes elementos:

- Los asistentes son el Equipo Scrum y los interesados clave invitados por el Dueño de Producto
- El Dueño de Producto explica qué elementos de la Lista de Producto se han “Terminado” y cuales no se han “Terminado”
- El Equipo de Desarrollo habla acerca de qué estuvo bien durante el Sprint, qué problemas aparecieron y cómo fueron resueltos esos problemas
- El Equipo de Desarrollo hace una demostración del trabajo que ha “Terminado” y responde preguntas acerca del Incremento.

- **Retrospectiva del Sprint (Sprint Retrospective)**

Permite al Equipo del Scrum a inspeccionarse a sí mismo y crear un listado de mejoras que pueden incrementar el valor del producto en el siguiente Sprint. Es realizado después de la Revisión del Sprint y antes de la siguiente planificación del Sprint. Tiene una duración máxima de 3 horas, además tiene como propósito revisar del desempeño de las personas, relaciones, procesos y herramientas durante el último Scrum, también permite identificar, ordenar e implementar mejoras en la forma en la que el Equipo del Scrum desempeña su trabajo (31).

Artefactos de Scrum

Son diseñados específicamente para maximizar la transparencia de la información clave, y que todos los miembros del Scrum tengan el mismo entendimiento (31).

- **Pila del Producto (Product Backlog)**

Es una pila ordenada de todo lo que posiblemente sea necesario para el desarrollo del producto, además de ser la única fuente de requisitos para cualquier cambio a realizarse en el producto.

La pila del producto nunca está completa debido a que al inicio del desarrollo del proyecto solo que describen los requerimientos comprendidos hasta ese momento y esta irá evolucionando a medida que el producto y el entorno que hará uso de él cambie. Por ende, la pila del producto es dinámica, es decir cambia constantemente para identificar lo que el producto necesita para ser adecuado, útil y competitivo (31).

- **Pila de Pendientes del Sprint (Sprint Backlog)**

Es un conjunto de elementos de la Pila del Producto seleccionados específicamente para el Sprint, además de un plan para entregar el incremento del producto y alcanzar el objetivo del Sprint. Es una predicción realizada por el Equipo de Desarrollo de la funcionalidad que será parte del siguiente incremento (31).

- **Incremento**

Es la suma de todos los elementos de la Pila del Producto completados durante un Sprint. Al finalizar un Sprint el incremento de tiene el valor de terminado, lo cual significa que cumple las condiciones necesarias para ser desplegado y utilizado (31).

- **Definición de “Terminado” (Definition of “Done”)**

Se da cuando un elemento de la Pila del Producto o un incremento se describe como “Terminado”. Este debe de ser utilizable de tal forma que el Dueño del Producto debe poder elegir Desplegarlo inmediatamente si fuera necesario.

2.3. DEFINICIÓN DE TÉRMINOS BÁSICOS

- Dialogflow

Plataforma NLP de Google que permite crear agentes conversacionales, concebida el 2010 e inicialmente llamada API.ai, en la actualidad permite la creación de agentes conversacionales de voz y texto, además de poder ser desplegada en la mayoría de plataformas de mensajería instantánea, así como para incrementar la funcionalidad de otro producto de Google llamado Google Assistant (32).

- Agentes

Los agentes son módulos de entendimiento de lenguaje natural, dichos módulos pueden trabajar dentro de una aplicación de mensajería instantánea, páginas web, entre otros (33).

- Intenciones

Una intención representa lo que el usuario quiere decir y la manera o acciones en las que nuestra aplicación debería responder.

Según, las intenciones son definidas dentro de un agente y estas son mapeadas a unas respuestas programadas, cada intención definida tiene que tener varios ejemplos de cómo el usuario podría desencadenar esa intención. También se podría decir que una intención representa un turno de dialogo en la conversación (34).

- Entidades

Las entidades son mecanismos para identificar y extraer data valiosa de ingresos de lenguaje natural. Mientras que las entidades te ayudan a entender la motivación detrás de una entrada en particular, las entidades son usadas para obtener partes específicas de esa entrada, ya sean cantidades productos, tipos de productos, direcciones, unidades, entre otros (32).

- Contextos

Los contextos son referidos al estado actual en el que se encuentran las peticiones de los usuarios pudiéndose así transmitir piezas de información entre distintas intenciones y momentos de la conversación, en el caso de Dialogflow puede haber contextos de entrada y salida, esto significa que la data puede ser usada al inicio de una intención o al final de otra para ser enviada a una nueva (35).

- Eventos

Los eventos permiten invocar intenciones basadas en algo que ha pasado en lugar de lo que el usuario comunica.

En el caso de Dialogflow se soportan eventos de diversas plataformas como Facebook, Google Assistant, Slack entre otros (36).

- **Fulfillment (Cumplimiento)**
Es un programa que es desplegado como un webhook y que permite al agente de Dialogflow hacer consultas a la lógica del negocio, durante una conversación el fulfillment permite usar información extraída mediante el procesamiento de lenguaje natural de Dialogflow y de esa forma generar respuestas dinámicas o también poder desencadenar acciones en el backend (37).
- **Netlify**
Plataforma todo en uno que permite la automatización de proyectos web modernos, permitiendo reemplazar la infraestructura de alojamiento, integración continua y despliegue en un solo flujo de trabajo (38).
- **Request**
Llamada a un API para la obtención de data o en todo caso su modificación dado un ID (39).
- **Javascript**
Lenguaje de programación interpretado, orientado a objetos y con funciones de alto nivel, es principalmente usado en navegadores web pero también es usado en el servidor con su implementación llamada Node JS (40).
- **Node JS**
Node JS es un entorno de ejecución de Javascript que funciona sobre el motor V8 de Chrome, este incluye todo lo necesario para poder ejecutar JavaScript en el servidor.
Apareció cuando los desarrolladores originales de Javascript extendieron algo que sólo podía funcionar en el navegador web a algo que pueda funcionar en una maquina como si fuera una aplicación. Además, Node JS tiene un amplio ecosistema de paquetes llamado npm (41).
- **Express**
Es un framework web escrito en Javascript y que funciona sobre el entorno de ejecución Node JS (42).

- **Node Modules**
Son archivos o directorios que pueden ser importados ya sea por la implementación de Node JS o JS en el navegador para añadir funcionalidad extra, en otras palabras, son las librerías de Javascript (41).
- **Dialogflow Node JS Fulfillment**
Librería para Node JS que permite conectar un Agente de Procesamiento de lenguaje natural de Dialogflow con la aplicación que el usuario desarrolle, de tal modo que puede realizar consultas a APIs y bases de datos. Además, provee funcionalidad extra como manejo de texto, cartas, imágenes, botones de sugerencia y payloads personalizados basados en el API de Facebook Messenger (43).
- **Vue JS**
Es un framework progresivo para la construcción de interfaces de usuario, está diseñado para ser adoptado incrementalmente (44).
- **Firebase Firestore**
Es una base de datos flexible y escalable pensada para la programación de servidores y dispositivos móviles a través de Firebase, mantiene la base de datos sincronizada entre apps a través de agentes de escucha en tiempo real, además ofrece asistencia sin conexión para dispositivos móviles y la web (45).
- **Base de Datos no Relacional**
Una Base de Datos no relacional es cualquier base de datos que no sigue el modelo relacional provisto por los tradicionales sistemas de manejo de bases de datos relacionales. Esta categoría de base de datos es referida también como bases de datos No SQL y en los últimos años aumentaron su popularidad debido al nacimiento de aplicaciones Big Data (46).

CAPITULO III

METODOLOGÍA

3.1. METODOLOGÍA APLICADA PARA EL DESARROLLO DE LA SOLUCIÓN

3.1.1. Metodología SCRUM

Scrum es un marco de trabajo de procesos que ha sido usado para gestionar el desarrollo de productos complejos desde principios de los años 90. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos. Scrum muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo, de modo que podamos mejorar. Los fundamentos tomados para la aplicación de Scrum en el Proyecto son:

- Algunas de las razones por las que se propone y utiliza un ciclo de desarrollo iterativo e incremental como SCRUM en el desarrollo de este proyecto vienen a ser:
- El sistema modular permite que los requerimientos del Asistente Virtual ADA para Electrocentro permiten desarrollar un esqueleto básico pero funcional y que según se vaya desarrollando el proyecto podrán ir incrementando según el proyecto crezca.

- Entregas frecuentes de los requerimientos del sistema permiten que la empresa disponga de la funcionalidad requerida obteniendo así un sistema incremental y mejora continua.
- También se considera un previsto cambio y afinamiento de requerimientos constantemente, debido a que el sistema planea incorporar funcionalidades más adelante en el proyecto, además de un cambio en el orden de desarrollo del proyecto.

3.1.2. Personas y roles del proyecto

A continuación, en la Tabla 2 se presenta el listado de los Involucrados en el proyecto.

Tabla 2. Personas involucradas en el proyecto del Asistente Virtual

Persona	Rol
Karina Laura León	Product Owner
Edwing Artica LLacta	Scrum Master
Edwing Artica LLacta	Equipo

Fuente: Elaborado por el autor

3.1.3. Fase de Desarrollo y Entregables

Como parte del uso de Scrum se detalla enseguida las Fases de desarrollo y entregables contempladas para este proyecto, además se especifica las historias de usuario asignadas a cada sprint, véase la Tabla 3.

Tabla 3. Fases de desarrollo scrum

FASES DE DESARROLLO SCRUM	
INICIACIÓN	
PROJECT CHARTER	Elaboración del documento
	Verificación del documento

	Entrega del documento
	Aprobación del documento
PLANIFICACIÓN	
PLAN DE DIRECCIÓN DEL PROYECTO	Elaboración del documento
	Verificación del documento.
	Entrega del documento.
	Aprobación del documento.
	Elaboración de la pila del Producto
	Validación de la pila del producto.
EJECUCIÓN	
HISTORIAS DE USUARIO	Identificación de Historias de Usuario
	Descripción de Historias de Usuario
	Priorización de Historias de Usuario
	Verificación de Historias de Usuario
PROCESOS FUNCIONALES	Identificación de Procesos
	Diseño de procesos funcionales
	Verificación de procesos
BASE DE DATOS	Diseño de diagrama de Base de Datos
	Diseño de Tablas
	Verificación de diseño

PRODUCTO - SPRINT	
SPRINT 1	HU-RF-001 - Como usuario final, requiero acceder al asistente virtual desde Messenger de Facebook, con la finalidad de tener un acceso fácil al sistema
	HU-RF-013 - Como usuario final, requiero poder conocer las opciones que me ofrece el asistente virtual, con la finalidad de poder saber que funcionalidad tiene el asistente
	HU-RF-014 - Como usuario final, requiero que el asistente puede poder enviar imágenes animadas, con la finalidad de darle una personalidad más amigable
	HU-RF-015 - Como usuario final, necesito saber si el asistente no entendió mi intención, con la finalidad de poder preguntar de forma más adecuada mi consulta
SPRINT 2	HU-RF-009 - Como usuario final, requiero poder reportar incidencias, con la finalidad obtener soluciones a los problemas que presenta mi servicio
	HU-RF-010 - Como usuario final, requiero saber si la ubicación de mi código de servicio, con la finalidad de poder realizar consultas sobre mi servicio
	HU-RF-011 - Como usuario final, necesito obtener un saludo cordial del asistente virtual, con la finalidad de saber si está listo para interactuar
SPRINT 3	HU-RF-004 - Como usuario final, requiero saber si habrán interrupciones de servicio, con la finalidad prever el corte y evitar inconvenientes

	<p>HU-RF-005 - Como usuario final, requiero saber los lugares de pago más cercanos, con la finalidad de saber dónde pagar mis recibos</p>
	<p>HU-RF-006 - Como usuario final, requiero saber los pasos para un reclamo, con la finalidad de tener en cuenta que documentos y a qué oficinas asistir para realizar los reclamos</p>
	<p>HU-RF-008 - Como usuario final, requiero saber los números de contacto de la empresa, con la finalidad de poder realizar consultas más complejas</p>
SPRINT 4	<p>HU-RF-007 - Como usuario final, requiero saber el monto de mi deuda anterior, con la finalidad de poder prever el monto de dinero a pagar</p>
	<p>HU-RF-016 - Como usuario final, requiero poder tener una conversación coloquial con el asistente, con la finalidad de poder llevar una conversación más amena</p>
	<p>HU-RF-017 - Como usuario final, poder escoger la institución bancaria en la que quiero realizar mis pagos, con la finalidad de obtener sus oficinas cercanas</p>
SPRINT 5	<p>HU-RF-002 - Como usuario final, requiero saber el monto de mi recibo por el servicio eléctrico, con la finalidad de mantenerme informado sobre mis deudas</p>
	<p>HU-RF-003 - Como usuario final, requiero saber la fecha de vencimiento de mi recibo de luz, con la finalidad de realizar el pago a tiempo</p>
	<p>HU-RF-012 - Como usuario final, requiero obtener una despedida del asistente virtual, con la finalidad de finalizar la conversación con el asistente</p>

SEGUIMIENTO Y CONTROL DEL PROYECTO	
SPRINT 1	Reuniones con los miembros del área de TI de Electrocentro
SPRINT 2	
SPRINT 3	
SPRINT 4	Entrega de Proyecto y accesos a las plataformas de despliegue
SPRINT 5	
CIERRE – MEMORIA FINAL	
MEMORIA FINAL	Elaboración de Documentos (Lecciones Aprendidas, Oportunidades de Mejora)
	Verificación de Documento
	Entrega de Carta de Aceptación del Proyecto

Fuente: Elaborado por el autor

CAPITULO IV

ANÁLISIS Y DISEÑO DE LA SOLUCIÓN

4.1. IDENTIFICACIÓN DE REQUERIMIENTOS

4.1.1. Estructura organizacional basado en procesos

En la Figura 10 se muestra el diagrama estructural de la Empresa Electrocentro, en el cual se detallan las diversas gerencias, áreas y unidades de negocio que conforman la empresa.

ORGANIGRAMA ESTRUCTURAL DE ELECTROCENTRO S.A.

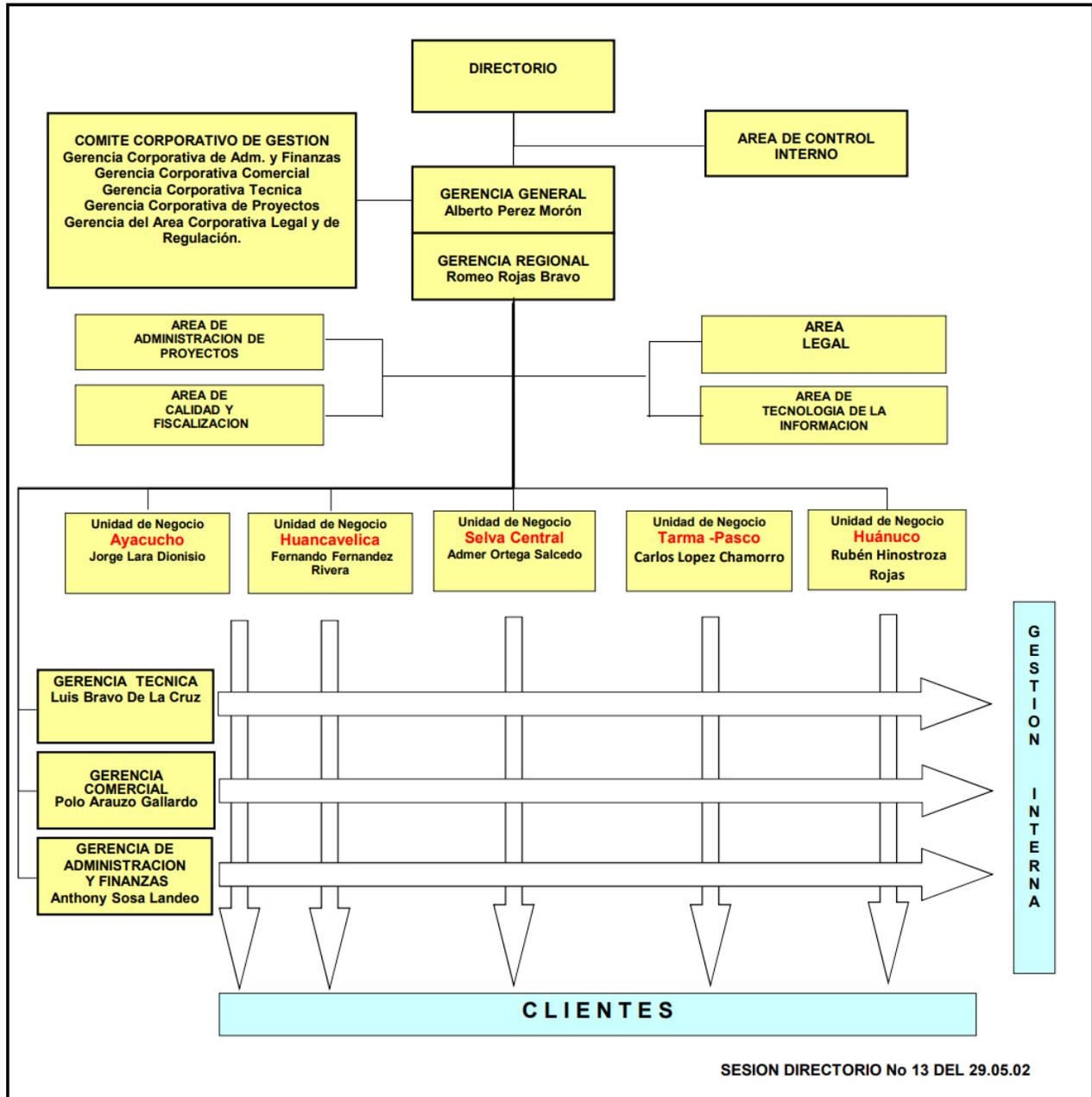


Figura 10. Organigrama estructural de Electrocentro.
Fuente: Tomada de (6)

4.1.2. Mapa del proceso de Atención al Cliente

En el proceso de atención al cliente se prevé que los clientes contacten directamente al asistente virtual y se realicen las validaciones correspondientes para detectar su intención, este proceso podría llevar a un error o la búsqueda de información no disponible, si este fuera el caso el cliente es derivado al community manager de la empresa, véase la Figura 11.

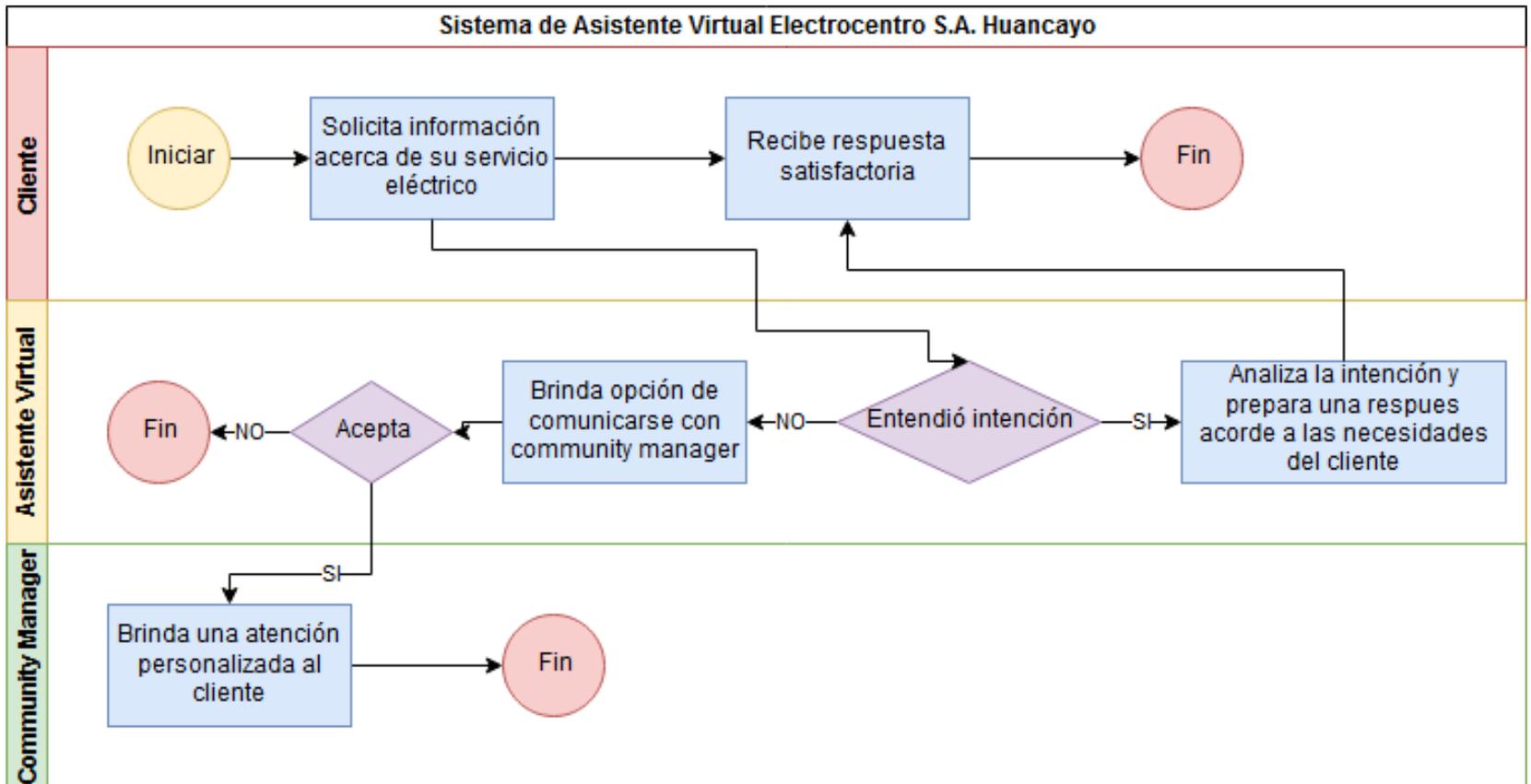


Figura 11. Mapa del proceso de atención al cliente a través del Asistente Virtual
Fuente: Elaborado por el Autor

4.1.3. Requerimientos Funcionales

Siguiendo la metodología Scrum se detallan los requerimientos funcionales los cuales detallan los servicios que prestará el sistema y la forma en la que se comporta dado una situación o evento, para el proyecto se tiene un listado de historias de usuario, véase la Tabla 4.

Tabla 4. Requerimientos Funcionales

ID	Enunciado de la historia	Alias	Estado	Dimensión/Esfuerzo	Iteración	Prioridad
HU-RF-001	Como usuario final, requiero acceder al asistente virtual desde Messenger de Facebook, con la finalidad de tener un acceso fácil al sistema.	Messenger	Planificado	Bajo	1	media
HU-RF-002	Como usuario final, requiero saber el monto de mi recibo por el servicio eléctrico, con la finalidad de mantenerme informado sobre mis deudas.	MontoRecibo	Planificado	alto	5	alta
HU-RF-003	Como usuario final, requiero saber la fecha de vencimiento de mi recibo de luz, con la finalidad de realizar el pago a tiempo.	FechaDeVencimiento	Planificado	alto	5	alta
HU-RF-004	Como usuario final, requiero saber si habrán interrupciones de servicio, con la finalidad prever el corte y evitar inconvenientes.	ConsultaInterrupciones	Planificado	medio	3	media
HU-RF-005	Como usuario final, requiero saber los lugares de pago más cercanos, con la finalidad de saber dónde pagar mis recibos.	LugaresDePagoCercano	Planificado	bajo	3	media

HU-RF-006	Como usuario final, requiero saber los pasos para un reclamo, con la finalidad de tener en cuenta que documentos y a qué oficinas asistir para realizar los reclamos.	PasosParaReclamo	Planificado	medio	3	media
HU-RF-007	Como usuario final, requiero saber el monto de mi deuda anterior, con la finalidad de poder prever el monto de dinero a pagar.	DeudaAnterior	Planificado	alto	4	media
HU-RF-008	Como usuario final, requiero saber los números de contacto de la empresa, con la finalidad de poder realizar consultas más complejas.	NumerosDeContacto	Planificado	medio	3	baja
HU-RF-009	Como usuario final, requiero poder reportar incidencias, con la finalidad obtener soluciones a los problemas que presenta mi servicio.	ReportarIncidencias	Planificado	medio	2	baja
HU-RF-010	Como usuario final, requiero saber si la ubicación de mi código de servicio, con la finalidad de poder realizar consultas sobre mi servicio.	NumeroDeServicio	Planificado	medio	2	baja
HU-RF-011	Como usuario final, necesito obtener un saludo cordial del asistente virtual, con la finalidad de saber si está listo para interactuar.	AsistenteSaludo	Planificado	medio	2	baja

HU-RF-012	Como usuario final, requiero obtener una despedida del asistente virtual, con la finalidad de finalizar la conversación con el asistente.	AsistenteDespedida	Planificado	alto	5	alta
HU-RF-013	Como usuario final, requiero poder conocer las opciones que me ofrece el asistente virtual, con la finalidad de poder saber que funcionalidad tiene el asistente.	AsistenteListarOpciones	Planificado	bajo	1	baja
HU-RF-014	Como usuario final, requiero que el asistente puede poder enviar imágenes animadas, con la finalidad de darle una personalidad más amigable.	AsistenteImagenAnimada	Planificado	bajo	1	alta
HU-RF-015	Como usuario final, necesito saber si el asistente no entendió mi intención, con la finalidad de poder preguntar de forma más adecuada mi consulta.	AsistenteNoComprende	Planificado	bajo	1	media
HU-RF-016	Como usuario final, requiero poder tener una conversación coloquial con el asistente, con la finalidad de poder llevar una conversación más amena.	AsistenteConversaciónVariada	Planificado	alto	4	media
HU-RF-017	Como usuario final, poder escoger la institución bancaria en la que quiero realizar mis pagos, con la finalidad de obtener sus oficinas cercanas.	EscogerInstituciónBancaria	Planificado	alto	4	media

Fuente: Elaborado por el autor

4.1.4. Especificación de Requerimientos Funcionales

Las especificaciones concernientes a cada uno de los requerimientos funcionales, es decir el contexto, eventos y resultados esperados son detalladas en la Tabla 5.

Tabla 5. Especificación de Requerimientos Funcionales

ID	Rol	Características	Razón/Resultado	Criterio de Aceptación	Contexto	Evento	Resultado
HU-RF-001	Como usuario final	Requiero acceder al Asistente Virtual desde Messenger de Facebook	Con la finalidad de tener un acceso fácil al sistema.	App Facebook	En caso ingrese desde un móvil.	Cuando se ingrese desde Messenger app	El Asistente Virtual tendrá en mismo funcionamiento.
				Messenger desde dispositivo móvil o PC	En caso entre desde una desktop o laptop	Cuando se ingrese desde el sitio web de Facebook	El Asistente Virtual tendrá en mismo funcionamiento.
HU-RF-002	Como usuario final	requiero saber el monto de mi recibo por el servicio eléctrico	Con la finalidad de mantenerme informado sobre mis deudas.	Recibir información	En caso ingrese número de servicio.	Cuando responda las preguntas.	El sistema será capaz de recibir la data del usuario
				Procesar información	En caso se tenga información.	Cuando se tenga la información del cliente.	El sistema analizará toda la información.
				Brindar una respuesta	Una vez haya realizado la consulta.	Cuando se tenga una respuesta concisa.	El sistema brindará la información al cliente.
HU-RF-003	Como usuario final	requiero saber la fecha de vencimiento de mi recibo de luz	con la finalidad de realizar el pago a tiempo	Brindar una fecha.	Una vez haya entregado la información de suministro.	Cuando pregunte cuál es la fecha de mi deuda.	El asistente me escribirá una fecha límite de

							pago de mi recibo de luz
HU-RF-004	Como usuario final	requiero saber si habrán interrupciones de servicio	con la finalidad prever el corte y evitar inconvenientes	Brindar información de interrupciones	Una vez haya dicho mi número de suministro.	Cuando pregunte si habrán interrupciones de servicio por mi zona.	El asistente debe brindar la información de interrupciones de servicio si las hubieran
HU-RF-005	Como usuario final	requiero saber los lugares de pago más cercanos	con la finalidad de saber dónde pagar mis recibos.	Brindar lugares de pago cercanos	Una vez haya dicho enviado mi ubicación.	Cuando pregunte por lugares de pago cercanos	El asistente debe mostrarme la bancos o agencias bancarias cercanas
HU-RF-006	Como usuario final	requiero saber los pasos para un reclamo	con la finalidad de tener en cuenta que documentos y a qué oficinas asistir para realizar los reclamos.	Informar los pasos para realizar un reclamo.	-	Cuando consulte los pasos para un reclamo.	El asistente debe dar la información de los pasos para realizar un reclamo
HU-RF-007	Como usuario final	requiero saber el monto de mi deuda anterior	con la finalidad de poder prever el monto de dinero a pagar	Informar crédito paralelo	Una vez haya ingresado mi número de suministro	Cuando consulte cuales son mis deudas anteriores.	El asistente debe mostrarme las deudas que tengo pendientes con la empresa
HU-RF-008	Como usuario final	requiero saber los números de	con la finalidad de poder realizar	Informar los número de		Cuando consulte	El asistente debe

		contacto de la empresa	consultas más complejas	contacto disponibles de la empresa		cuales son los números de contacto de la empresa.	informarme un listado de los números de contacto disponibles de la empresa
HU-RF-009	Como usuario final	requiero poder realizar reclamos.	con la finalidad obtener soluciones a los problemas que presenta mi servicio	Poder obtener una solución a los problemas de mi servicio	Una vez haya dicho mis datos de cliente.	Cuando consulte que quiero reportar reclamos	El asistente debe requerir los datos del reclamo e informarme su estado.
HU-RF-010	Como usuario final	requiero saber si la ubicación de mi código de servicio	con la finalidad de poder realizar consultas sobre mi servicio.	Informar la ubicación del código de suministro		Cuando consulte donde puedo conseguir mi número de suministro	El asistente debe mostrarme una imagen que detalle la ubicación del número de suministro
HU-RF-011	Como usuario final	necesito obtener un saludo cordial del asistente virtual.	con la finalidad de saber si está listo para interactuar	Brindar un saludo ameno		Cuando salude al asistente	El asistente mostrara un saludo ameno hacia el cliente
HU-RF-012	Como usuario final	requiero obtener una despedida del asistente virtual.	con la finalidad de finalizar la conversación con el asistente.	Brindar una conversación más amena con el cliente		Cuando el cliente se despida del asistente virtual	El asistente debe mostrarme una despedida amena
HU-RF-013	Como usuario final	requiero poder conocer las	con la finalidad de poder saber	Brindar todas las opciones		Cuando consulte sobre	El asistente responderá con

		opciones que me ofrece el asistente virtual.	que funcionalidad tiene el asistente.	que dispone el asistente virtual		las opciones disponibles del asistente virtual	un listado de todas las opciones disponibles para su funcionamiento.
HU-RF-014	Como usuario final	requiero que el asistente puede poder enviar imágenes animadas.	con la finalidad de darle una personalidad más amigable	Lenguaje coloquial	Una vez escriba.	Cuando escriba al asistente	El asistente se comunicará usando imágenes animadas amenas.
HU-RF-015	Como usuario final	necesito saber si el asistente no entendió mi intención.	con la finalidad de poder preguntar de forma más adecuada mi consulta	Lenguaje coloquial	Una vez escriba algún mensaje al asistente.	Cuando empiece una conversación.	El asistente responderá si no fue capaz de entender la intención del cliente.
HU-RF-016	Como usuario final	requiero poder tener una conversación coloquial con el asistente.	con la finalidad de poder llevar una conversación más amena.	Lenguaje coloquial	Una vez escriba algún mensaje al asistente.	Cuando empiece una conversación.	El asistente llevará una conversación cotidiana con el cliente
HU-RF-017	Como usuario final	poder escoger la institución bancaria en la que quiero realizar mis pagos.	con la finalidad de obtener sus oficinas cercanas.	Información de campañas crediticias.	Una vez haya ingresado mis datos.	Cuando pregunta las el tipo de banco	El asistente brindará un listado de opciones de instituciones bancarias

Fuente: Elaborado por el autor

4.1.5. Requerimientos no funcionales

Los requerimientos no funcionales describen funcionalidades y características del proyecto tales como rendimiento, disponibilidad, apariencia, entre otros, estos se detallan en la Tabla 6.

Tabla 6. Requerimientos no Funcionales

ID	Enunciado de la historia	Atributo
HU-RNF-001	El Asistente Virtual debe estar disponible las 24 horas del día y los 365 días del año, sujeto al hosting enlazado al sistema.	Disponibilidad
HU-RNF-002	El Asistente Virtual debe tener una respuesta fluida a los usuarios.	Fluidez
HU-RNF-003	El Asistente Virtual debe tener un nombre relacionado al nombre de la empresa.	De negocio
HU-RNF-004	El Asistente Virtual debe tener un banner relacionado a los colores de la empresa.	De negocio
HU-RNF-004	El Asistente Virtual debe iniciar los saludos con el nombre de la empresa.	De negocio

Fuente: Elaborado por el autor

4.1.6. Conformación del equipo de trabajo

En el equipo de trabajo del proyecto participan miembros de la empresa Electrocentro para validar y dar la conformidad de la finalización del proyecto, en la Tabla 7 se muestra los elementos participantes.

Tabla 7. Equipo de trabajo

Rol	Persona	Área
Product Owner	Karina Laura León	Jefe del Área de Tecnologías de la Información de Electrocentro Huancayo
SCRUM Manager	Edwing Cristian Artica LLacta	Desarrollador
Team	Edwing Cristian Artica LLacta	Desarrollador

Fuente: Elaborado por el autor

4.2. ANÁLISIS DE LA SOLUCIÓN

4.2.1. Análisis morfológico

En la HU-RF-001 se muestra la pantalla de inicial al contactar con el Asistente virtual, en ella se muestra el banner informativo y la descripción del asistente, véase la Tabla 8.

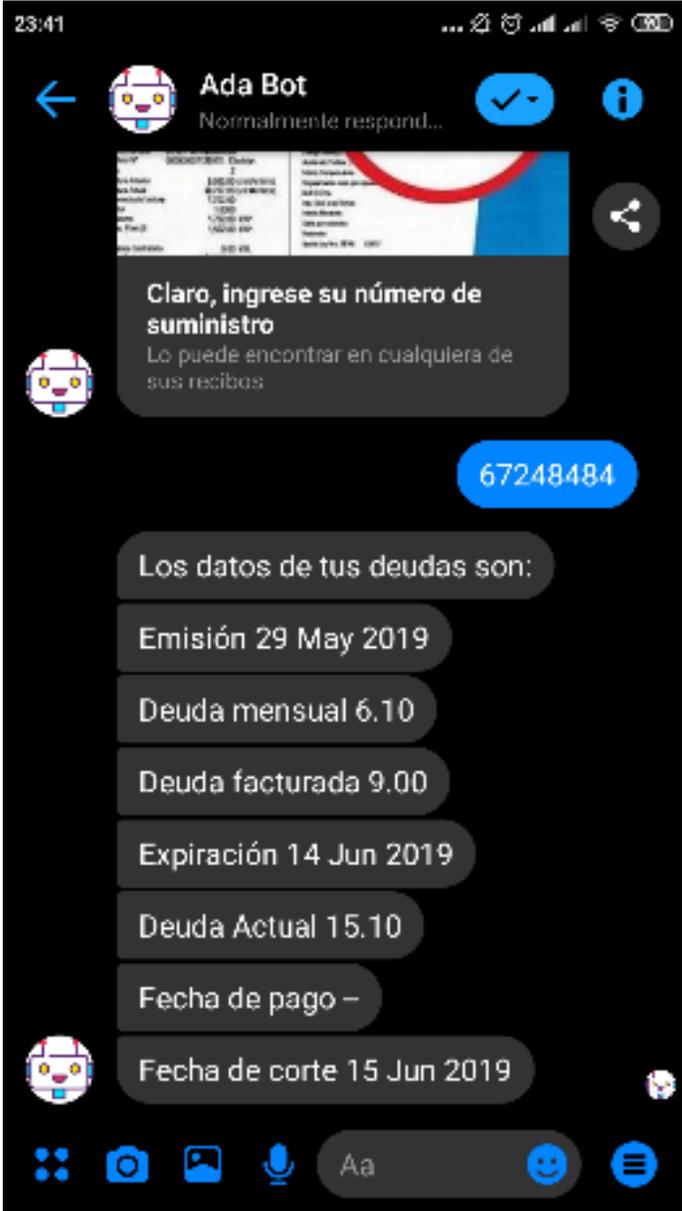
Tabla 8. HU-RF-001 - Asistente Virtual en Messenger

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-001	AsistenteMessenger	Asistente	Muestra el inicio de la conversación del asistente
			<p>El Usuario deberá presionar el botón empezar para dar inicio al funcionamiento del asistente virtual.</p>

Fuente: Elaborado por el autor

En la HU-RF-002 mostrada en la Tabla 9, son mostrados los datos recibidos de la consulta del recibo de luz enviando su número de servicio, se recibe la fecha de emisión, el monto de la deuda, entre otros.

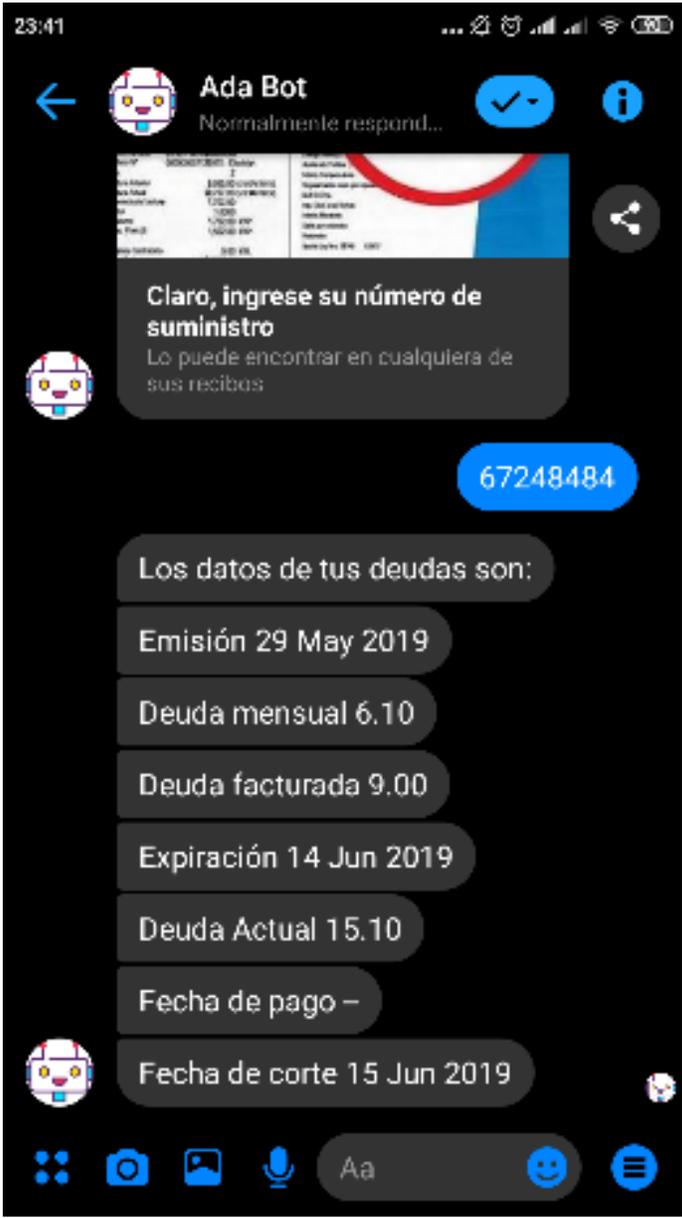
Tabla 9. HU-RF-002 - Consulta de Recibos

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-002	MontoRecibo	Asistente	Muestra los datos de su recibo de luz del presente mes.
			<p>El Usuario deberá poder visualizar el monto de su deuda del mes con la empresa</p>

Fuente: Elaborado por el autor

La HU-RF-003 puede ser apreciada en la Tabla 10, como parte de la consulta del recibo del mes es mostrada su fecha de vencimiento.

Tabla 10. HU-RF-003 – Fecha de vencimiento de recibos

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-003	FechaDeVencimiento	Asistente	Muestra la fecha de vencimiento de su recibo del mes
 <p>The screenshot shows a WhatsApp chat interface with a bot named 'Ada Bot'. The bot's profile picture is a white robot head with red and blue accents. The chat header shows 'Ada Bot' and 'Normalmente respond...'. The main content of the chat is a list of debt information presented in a dark grey bubble with white text. The list includes: 'Los datos de tus deudas son:', 'Emisión 29 May 2019', 'Deuda mensual 6.10', 'Deuda facturada 9.00', 'Expiración 14 Jun 2019', 'Deuda Actual 15.10', 'Fecha de pago -', and 'Fecha de corte 15 Jun 2019'. Above the list, there is a blue bubble containing the number '67248484'. The chat background is black, and the bottom of the screen shows the standard WhatsApp input area with icons for gallery, camera, microphone, text, emojis, and a menu.</p>			<p>El Usuario deberá poder visualizar un listado de opciones en forma de cartas con botones los cuales representan las opciones del asistente</p>

Fuente: Elaborado por el autor

La HU-RF-004 detallada en la Tabla 11, muestra si se tiene interrupciones programadas realizando la consulta enviando el número de servicio en su recibo

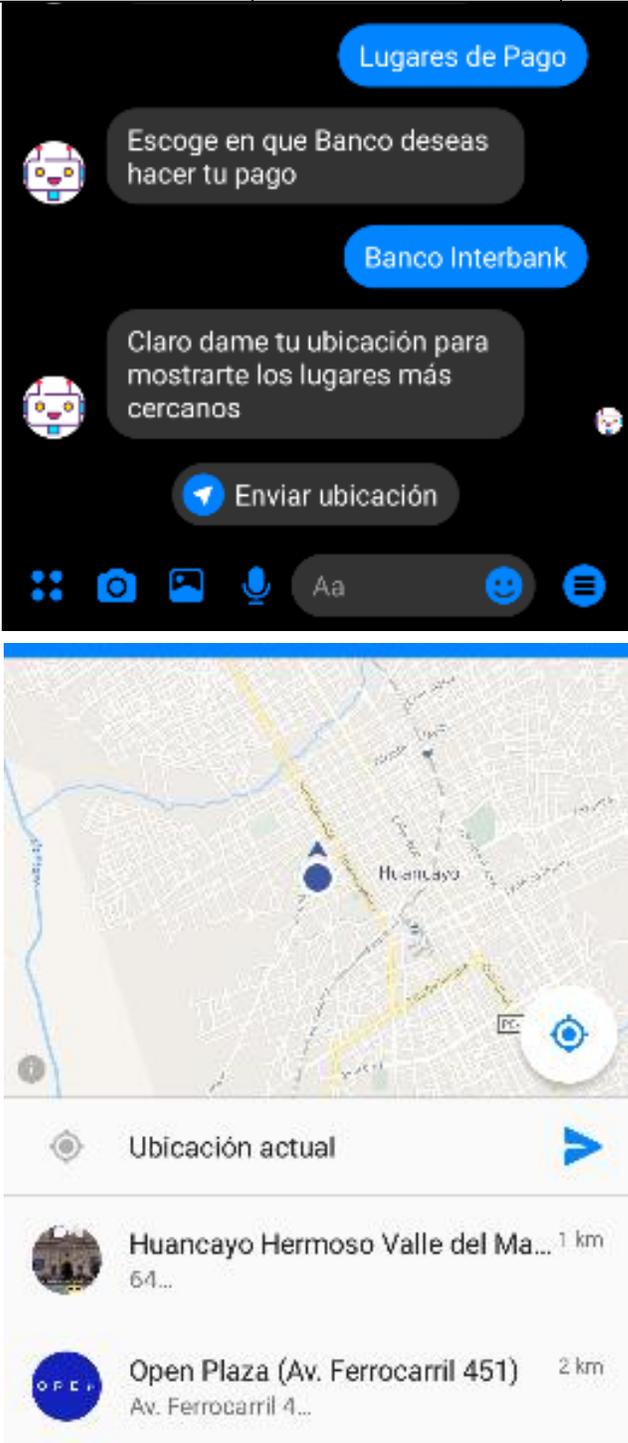
Tabla 11. HU-RF-004 - Consultar interrupciones

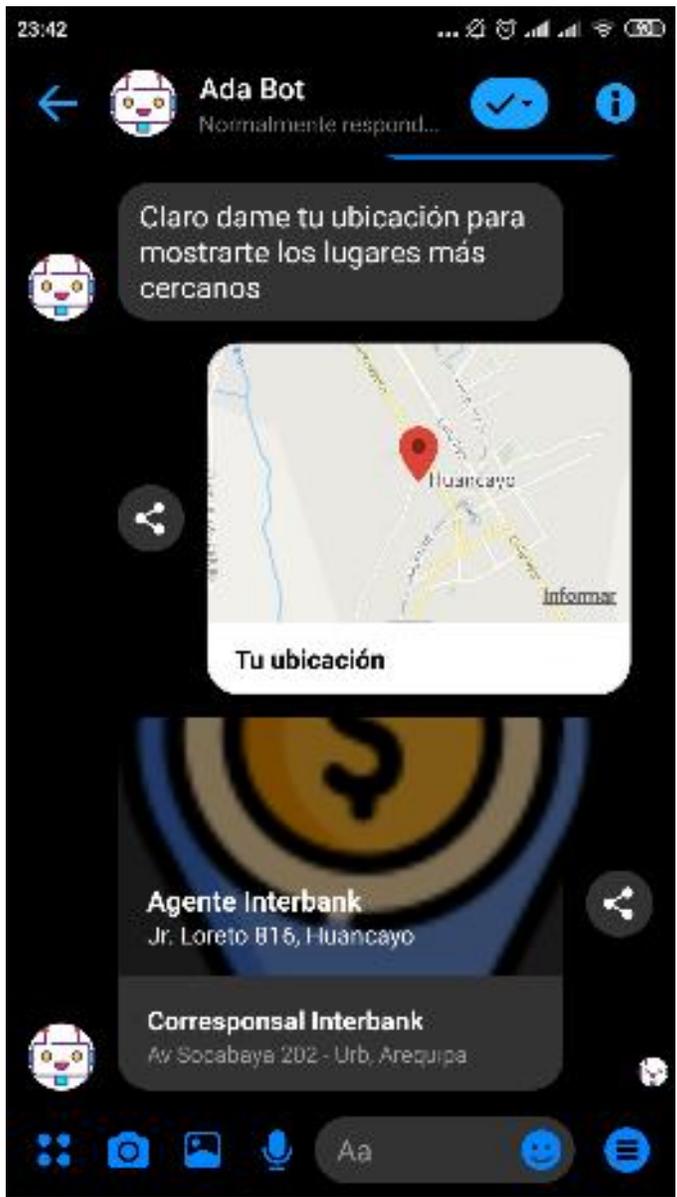
Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-004	ConsultaInterrupciones	Asistente	Muestra las interrupciones programadas de servicio para el área del cliente
			El Usuario deberá poder visualizar si se tiene una interrupción programada de su servicio

Fuente: Elaborado por el autor

La HU-RF-005 detallada en la Tabla 12, muestra los lugares de pago cercanos dados la ubicación actual del cliente y la institución bancaria ingresada

Tabla 12. HU-RF-005 - Lugares de pago cercano

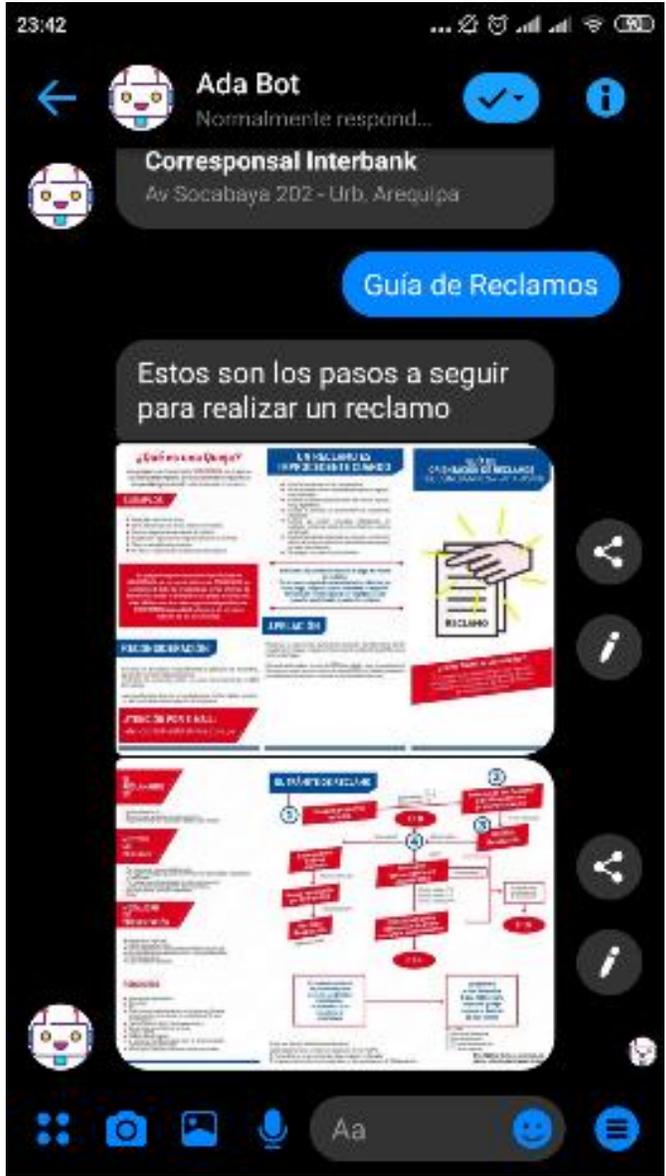
Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-005	LugaresDePagoCercano	Asistente	Muestra los lugares cercanos de pago
 <p>The screenshot shows a chat conversation on a dark background. At the top, a blue bubble says 'Lugares de Pago'. The user asks: 'Escoge en que Banco deseas hacer tu pago'. The assistant responds: 'Banco Interbank'. The user then asks: 'Claro dame tu ubicación para mostrarte los lugares más cercanos'. The assistant responds with a map of Huancayo and a list of nearby banks: 'Huancayo Hermoso Valle del Ma...' (1 km) and 'Open Plaza (Av. Ferrocarril 451)' (2 km). The chat interface includes a bottom bar with icons for gallery, camera, microphone, text input (Aa), emojis, and a menu.</p>		<p>El Usuario deberá poder visualizar dada su ubicación cuales son los lugares de pago más cercanos.</p> <p>El Usuario debe escoger la entidad bancaria que desea encontrar</p> <p>El Usuario debe enviar su ubicación actual</p>	



Fuente: Elaborado por el autor

En la HU-RF-006 detallada en la Tabla 13, recolecta las imágenes de la web y las muestra a través de la aplicación de Messenger, en estas se observan los pasos necesarios para los reclamos.

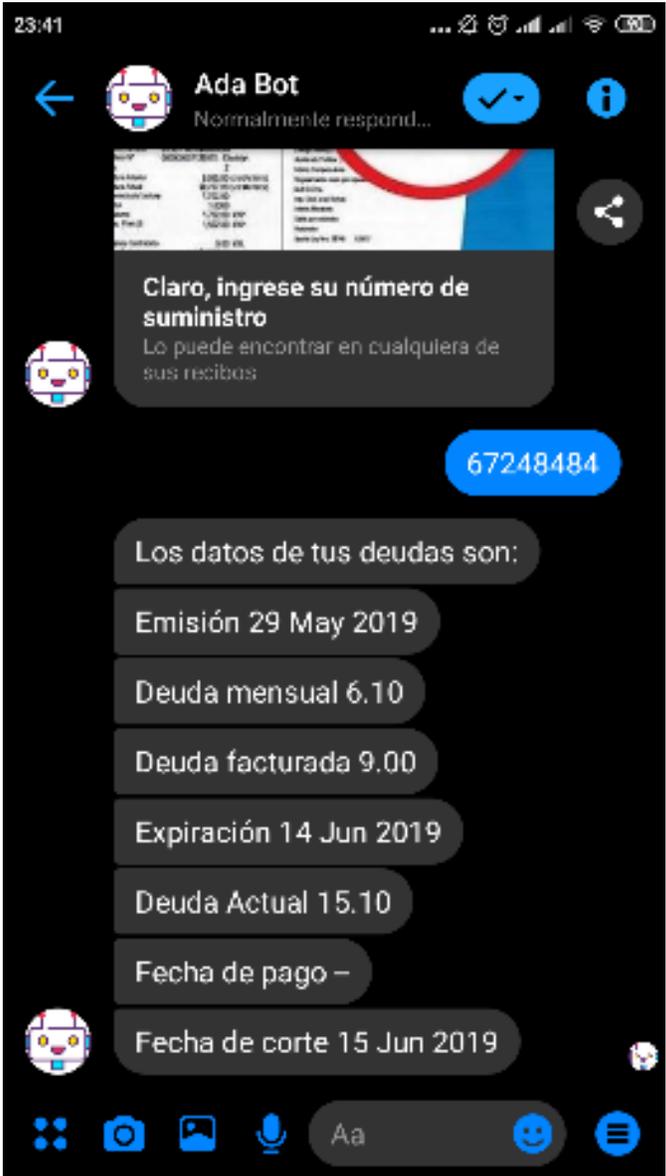
Tabla 13. HU-RF-006 - Pasos para reclamos

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-006	PasosParaReclamo	Asistente	Muestra los pasos para realizar los reclamos acerca del servicio eléctrico
 <p>The screenshot shows a chat interface with a bot named 'Ada Bot'. A blue button labeled 'Guía de Reclamos' is visible. Below it, a message contains the text 'Estos son los pasos a seguir para realizar un reclamo' followed by two images: one with text instructions and another with a flowchart. The flowchart details the steps for filing a claim, starting with 'RECLAMO' and moving through various stages like 'RECEPCIÓN', 'ANÁLISIS', and 'RESOLUCIÓN'.</p>			<p>El Usuario visualizará un listado de imágenes que detallan los pasos para realizar un reclamo</p>

Fuente: Elaborado por el autor

La Tabla 14, muestra el análisis de la HU-RF-007 que como parte de la consulta de los recibos de luz se muestra mediante Messenger la deuda pendiente del cliente.

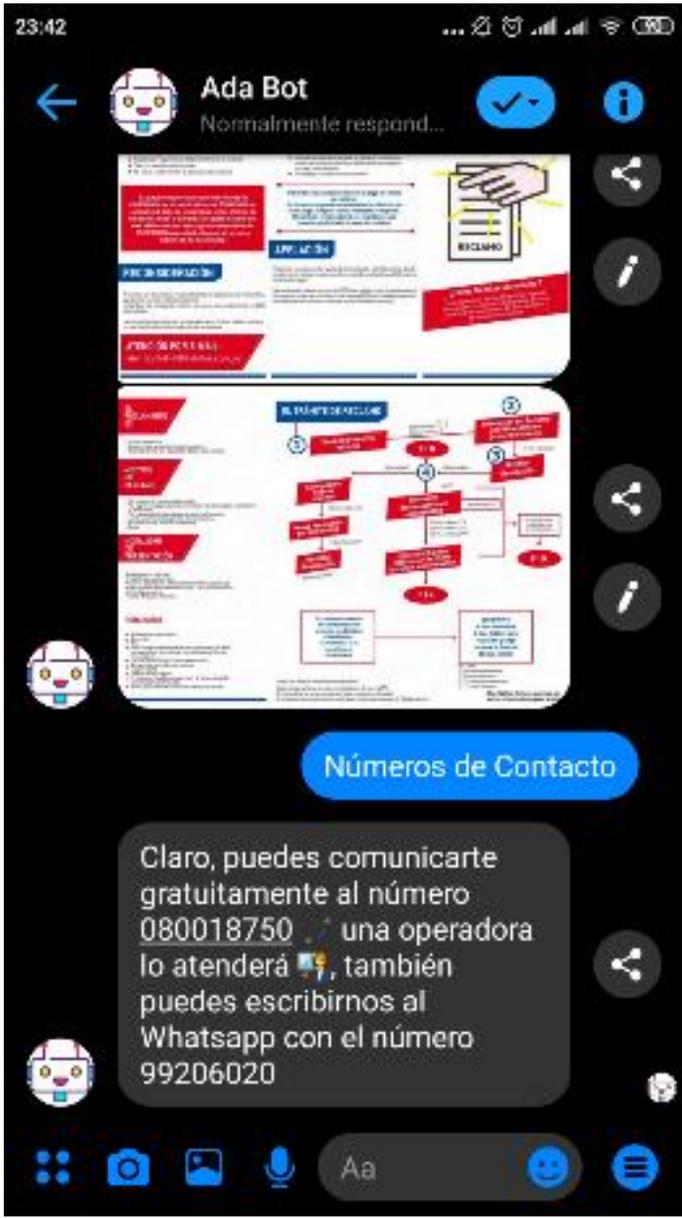
Tabla 14. HU-RF-007 - Deudas anteriores

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-007	DeudaAnterior	Asistente	Muestra el monto de la deuda anterior del cliente
 <p>The screenshot shows a chat interface with a bot named 'Ada Bot'. The bot has sent a message asking for the supply number and then provided a list of debt information:</p> <ul style="list-style-type: none"> Los datos de tus deudas son: Emisión 29 May 2019 Deuda mensual 6.10 Deuda facturada 9.00 Expiración 14 Jun 2019 Deuda Actual 15.10 Fecha de pago - Fecha de corte 15 Jun 2019 			<p>El Usuario visualizará cuanto es el monto de la deuda que tiene pendiente con Electrocentro</p>

Fuente: Elaborado por el autor

La Tabla 15, detalla la historia HU-RF-008 que retorna los datos de los números de contacto de la empresa recogidos de su web corporativa.

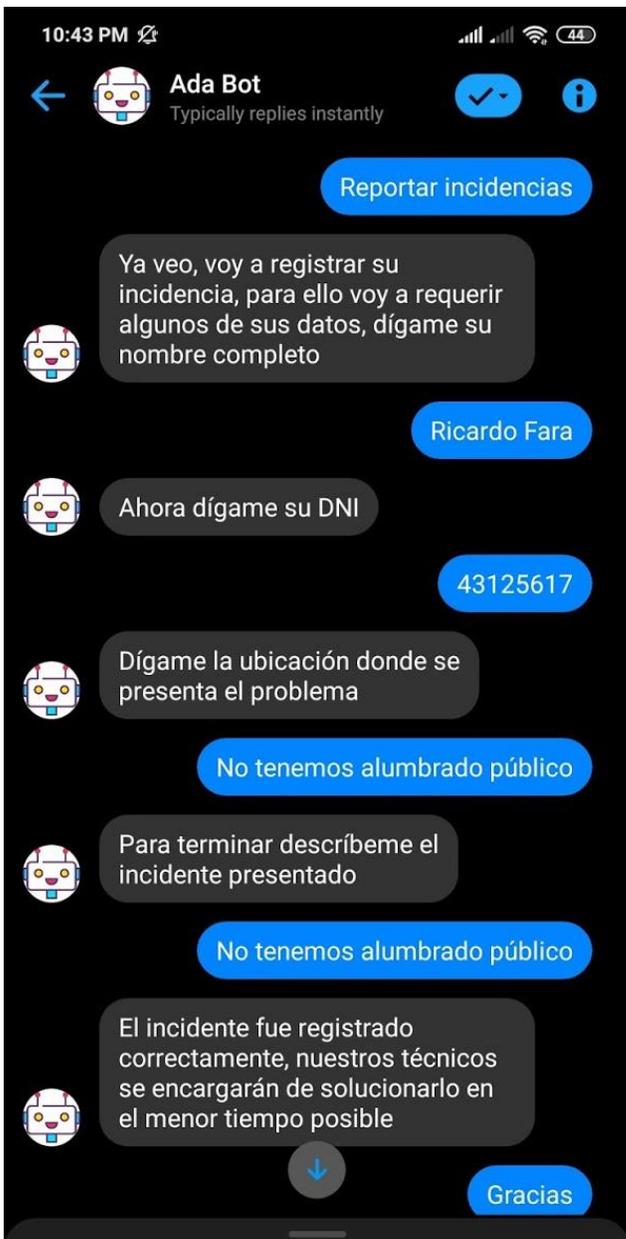
Tabla 15. HU-RF-008 - Números de contacto

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-008	NumerosDeContacto	Asistente	Muestra los números de contacto de la empresa
			<p>El Usuario visualizará en forma textual los números de contacto disponibles de la empresa</p>

Fuente: Elaborado por el autor

La HU-RF-009 da a conocer cómo es que se reportan las incidencias del servicio eléctrico de los clientes haciendo uso del asistente virtual, véase la Tabla 16.

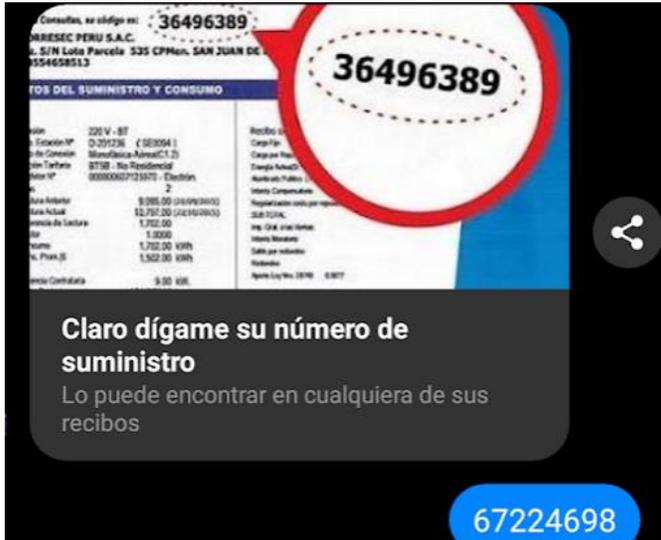
Tabla 16. HU-RF-009- Reportar incidencias

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-009	ReportarIncidencias	Asistente	Muestra los datos necesarios para reportar una incidencia
			<p>El Usuario ingresará sus datos personales</p> <p>El usuario enviará una imagen del reclamo o incidente</p> <p>Se registrará el resultado y será enviado a los miembros de soporte de la empresa</p>

Fuente: Elaborado por el autor

La HU-RF-010 da a conocer permite al cliente ingresar su número de servicio además le detalla la ubicación de este en su recibo, véase la Tabla 17

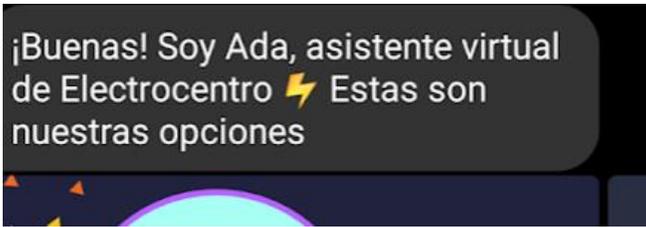
Tabla 17. HU-RF-010 - Números de servicio

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-010	NumeroDeServicio	Asistente	Muestra los datos necesarios para la visualización del número de servicio o suministro
			El Usuario visualizará mediante una imagen la ubicación en su recibo del número de servicio necesario para realizar las diversas consultas

Fuente: Elaborado por el autor

La HU-RF-011 muestra al usuario un saludo ameno registrado en la plataforma de Dialogflow, es posible mostrar emojis como parte de este, véase la Tabla 18.

Tabla 18. HU-RF-011 - Saludo del asistente

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-011	AsistenteSaludo	Asistente	Muestra un saludo del asistente
			El Usuario visualizará un saludo ameno por parte del asistente virtual

Fuente: Elaborado por el autor

Como se puede apreciar en la Tabla 19, que expone los detalles de la HU-RF-012, se hace presente una imagen animada de tipo GIF cuando el cliente se despide del asistente virtual.

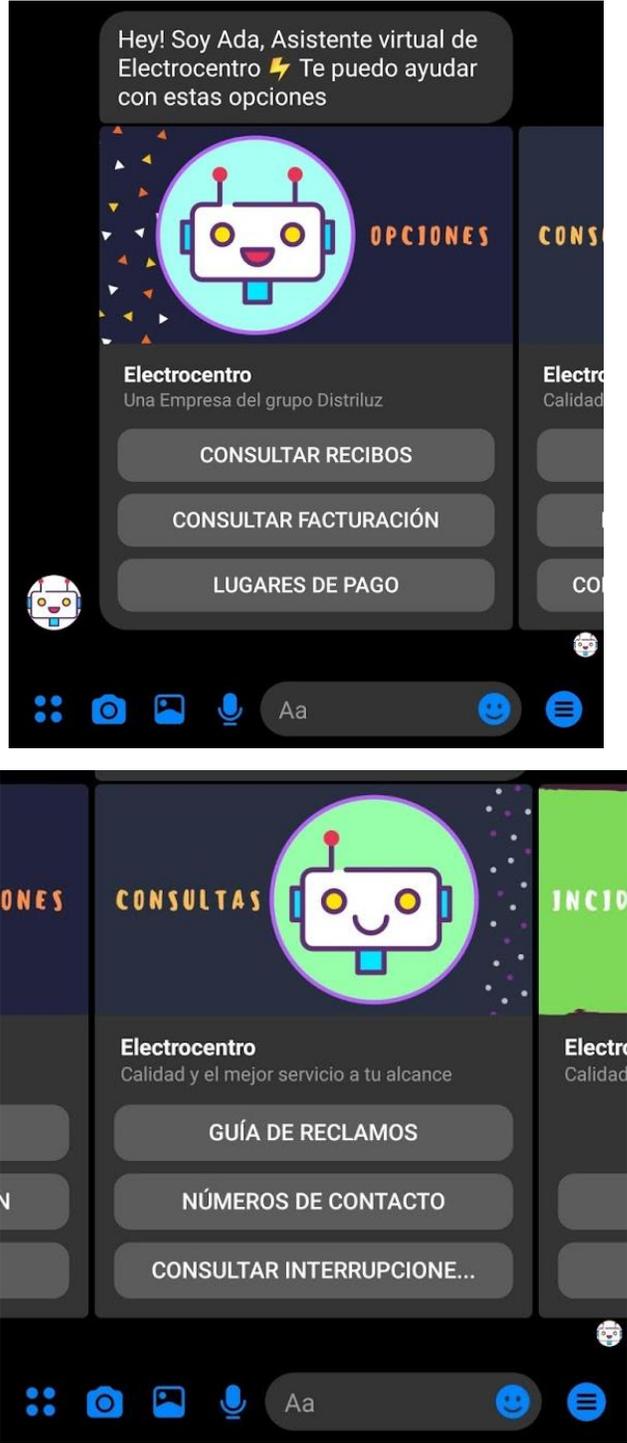
Tabla 19. HU-RF-012 - Despedida de asistente

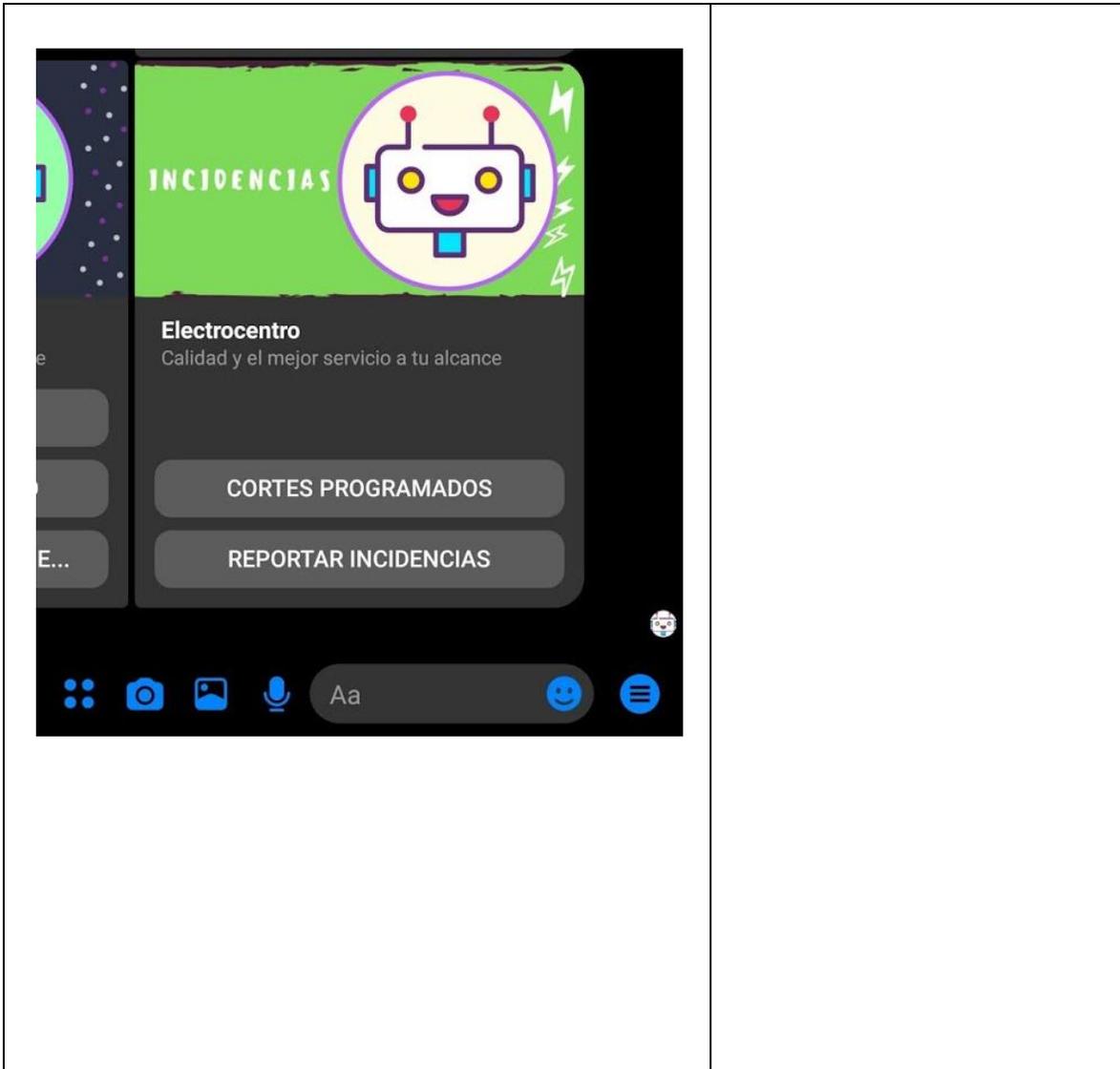
Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-012	AsistenteDespedida	Asistente	Muestra un despedida del asistente
			El Usuario visualizará una despedida amena por parte del asistente virtual

Fuente: Elaborado por el autor

En la Tabla 20, se muestra lo detallado en la historia HU-RF-013, en la que se listan las opciones disponibles en el asistente, las cuales son agrupadas mediante cartas utilizando el api de Facebook Messenger.

Tabla 20. HU-RF-013 - Listar opciones del asistente

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-013	AsistenteListarOpciones	Asistente	Muestra el listado de opciones disponibles del asistente virtual
			<p>El Usuario visualizará mediante un carrusel de cartas el listado de opciones en forma de botones organizados, estos pueden ser pulsados para su funcionamiento</p>



Fuente: Elaborado por el autor

La HU-RF-014, refiere que en diversos momentos de la comunicación entre el asistente y el cliente se mostraran imágenes animadas para lograr una conversación más amena estas imágenes son obtenidas a través del api de Tenor GIF, véase la Tabla 21.

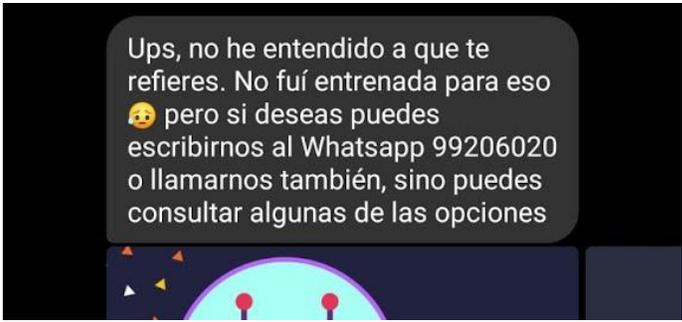
Tabla 21. HU-RF-014 - Mostrar imágenes animadas

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-014	AsistentelImagenAnimada	Asistente	Muestra una imagen animada en formato GIF
			<p>El Usuario visualizará imágenes animadas en ciertas partes de la conversación</p>

Fuente: Elaborado por el autor

En la Tabla 22, que detalla la HU-RF-015, se muestra el mensaje obtenido al no haber logrado identificar la intención del cliente.

Tabla 22. HU-RF-015 - Asistente no comprende intención

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-015	AsistenteNoComprende	Asistente	Muestra un mensaje que indica que no comprendió la conversación
			El Usuario visualizará mensajes que indiquen que la conversación no se entendió para que el usuario ingrese un mensaje válido

Fuente: Elaborado por el autor

La HU-RF-016 permite al asistente llevar una conversación variada con el cliente ya que incluye varios tópicos que comunes que a menudo se ven en una conversación, entre ellos se le puede preguntar al asistente sobre bromas, quién es su jefe, cómo se siente, entre otros, véase la Tabla 23.

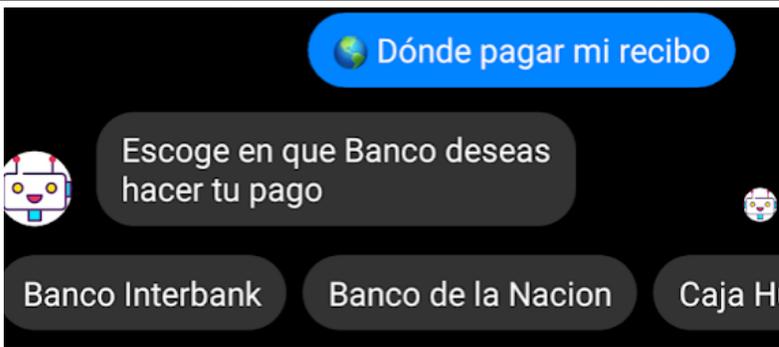
Tabla 23. HU-RF-016 - Asistente conversación variada

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-016	AsistenteConversaciónVariada	Asistente	Muestra mensajes variados de conversación coloquial
			El Usuario visualizará mensajes variados de acuerdo a la conversación que lleve con el asistente en un lenguaje coloquial

Fuente: Elaborado por el autor

La Tabla 24 refiere la historia HU-RF-017 en la cual se permite que el cliente escoja entre un listado de bancos para lograr hacer su consulta de lugares de pago.

Tabla 24. HU-RF-017 - Escoger institución bancaria

Código	Nombre Historia	Módulo	Criterio Aceptación
HU-RF-017	EscogerInstituciónBancaria	Asistente	Muestra un listado de opciones de instituciones bancarias
			El Usuario visualizará un listado de botones que le permitirán escoger la agencia bancaria más cercana

Fuente: Elaborado por el autor

4.2.2. Arquitectura de la Solución

El Asistente Virtual es desarrollado y presentado a los usuarios a través de la plataforma de Facebook Developers y la aplicación de Facebook Messenger Móvil y Web, de esta forma se logra obtener los datos ingresados por los usuarios a través de texto y de los menús integrados en la aplicación, gracias a esto se permite que la conversación sea procesada y entendida mediante el agente conversacional desarrollado en Dialogflow que funciona como Procesador de Lenguaje Natural a su vez el agente incrementa su inteligencia mediante el servicio serverless desplegado en Netlify que contiene el webhook principal conectado al agente y las diversas funciones serverless que alimentan las diversas intenciones, cada una de estas funciones se conectan al API de Electrocentro, el cual accede a la base de datos de Electrocentro y retorna toda la data de los usuarios concerniente a su servicio eléctrico, de esta forma, en cada consulta se puede obtener la data de los clientes al ingresar su número de suministro, además Netlify está conectado al repositorio de Github para un despliegue continuo, todo esto se observa a detalle en la Figura 12.

Además, en la Tabla 25, se describen los servicios desarrollados y desplegados para que el asistente sea funcional los cuales son llamados por el Webhook principal cuando el usuario lo requiera, estos servicios se comunican con el Api de Electrocentro para obtener la data requerida, procesarla y enviarla a través del webhook a la aplicación donde se encuentra el Asistente Virtual, en este caso Facebook Messenger

Tabla 25. Endpoints de la aplicación

Servicios	End Points de la Aplicación
Webhook Principal	Parámetros: Manejados internamente por el agente de Dialogflow Url: https://ada-bot.netlify.com/webhook
Deudas del cliente	Parámetros: nroservicio
	Url: https://ada-serverless-functions.netlify.com/debts
Respuesta animada por TenorGIF	Parámetros: intencion
	Url: https://ada-serverless-functions.netlify.com/gifResponse
Deudas de los últimos 4 meses del cliente	Parámetros: nroservicio
	Url: https://ada-serverless-functions.netlify.com/historicalBilling
Interrupciones de servicio programadas	Parámetros: nroservicio
	Url: https://ada-serverless-functions.netlify.com/interruptions
Lugares de pago cercanos alimentado por Google Maps Api	Query Params: longitude, latitude, bank
	Url: https://ada-serverless-functions.netlify.com/paymentLocations
Webscraper para obtener información de la web	Parámetros: nroservicio
	Url: https://ada-serverless-functions.netlify.com/scrapper

Reporte de Incidentes	Parámetros: fullname, dni, location, description
	Url: https://us-central1-ada-bot-1b3d7.cloudfunctions.net/storeIncidents

Fuente: Elaborado por el autor

Por otro lado, se detallan los servicios web expuestos por Electrocentro, los cuales al ser consultados retornan la información de los clientes, dados los parámetros especificados por la empresa, véase la Tabla 26.

Tabla 26. Endpoints publicados por Electrocentro para la aplicación

Servicios	Endpoints Api Electrocentro
Información del suministro consultado	Parámetros: {"ParametroControlTag":"idnroservicio 46118192"}
	Url: http://servicios.distriluz.com.pe:50000/Distriluzmobile/api/serviciodistriluzmobile/ConsultaSuministroInformacion
Validación el suministro	Parámetros: {"ParametroControlTag":"idnroservicio 46118192"}
	Url: http://servicios.distriluz.com.pe:50000/Distriluzmobile/api/servicioregistro/validarsuministroasociar
Consulta facturación de los últimos 4 meses	Parámetros: {"ParametroControlTag":"idnroservicio 46118192"}
	Url: http://servicios.distriluz.com.pe:50000/Distriluzmobile/api/serviciodistriluzmobile/consultaresumenultimafacturacion

Consulta Interrupciones de servicio programadas	Parámetros: {"Parametros":{"IdNroServicio":46118192}}
	Url: http://servicios.distriluz.com.pe:50000/SISAC/Atencion/Api/OficinaVirtual/ListarInterrupciones

Fuente: Elaborado por el autor

Como se puede apreciar en la Figura 12 la solución está conectada a través de diversos servicios web los cuales intercambian información relacionada al cliente que realiza sus consultas, cada servicio es independiente y está desplegado en su propio contenedor, por ende el consumo y facturación es optimizado dependiendo de su tiempo de ejecución y tomando en cuenta como única fuente de datos el API de Electrocentro que retorna esta data desde su base de datos principal.

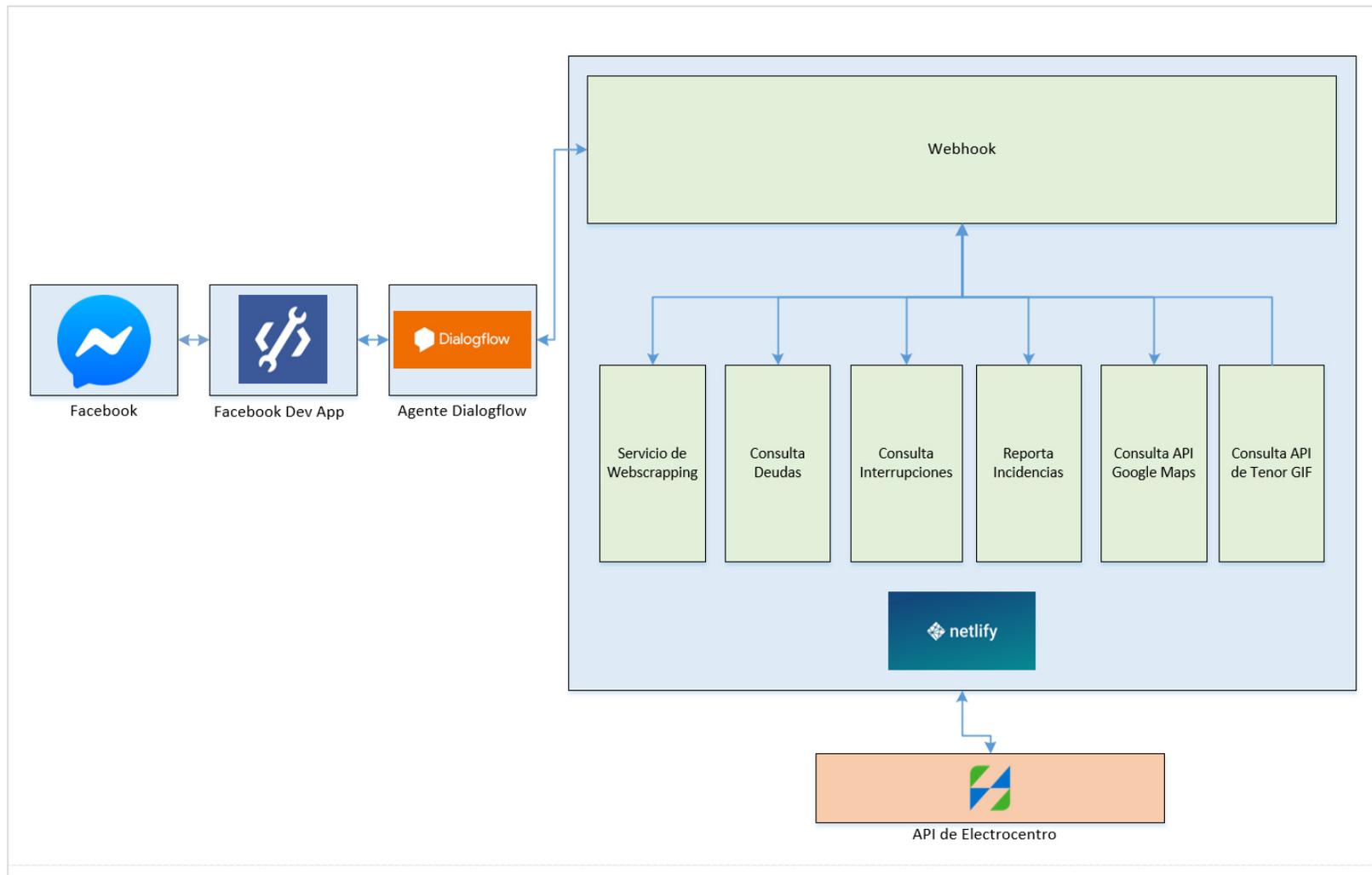


Figura 12. Arquitectura de la solución
Fuente: Elaborado por el autor

4.2.3. Análisis tecnológico

Se consideran materiales, técnicas y herramientas para construir el producto.

4.2.3.1. Herramientas de Hardware

Principalmente se toman en cuenta el hardware usado para el desarrollo del Asistente virtual, véase la Tabla 27.

Tabla 27. Herramientas de Hardware

Hardware	Especificación
PC	<ul style="list-style-type: none">- Memoria: 16GB- Disco Duro: 1TB- Procesador: Intel Core i7 6700k- Velocidad: 4.4GHz- Sistema Operativo: 64 bits, procesador x64

Fuente: Elaborado por el autor

4.2.3.2. Herramientas de Software

Para el desarrollo del Asistente se usaron las siguientes herramientas de software véase la Tabla 28, debido a ser muy populares y fáciles de usar y configurar además de que permiten un desarrollo muy ágil.

Tabla 28. Herramientas de Software

Software	Especificación
Herramientas para codificar programas	<ul style="list-style-type: none">- Visual Studio Code
Herramienta para gestionar la base de datos	<ul style="list-style-type: none">- Firebase Firestore
Framework de Desarrollo	Node JS, Dialogflow Node JS Library, Vue JS
Plataforma de Procesamiento de Lenguaje	Dialogflow API V2

Fuente: Elaborado por el autor

4.2.4. Análisis económico

4.2.4.1. Estructura de Costos

COSTOS DE PERSONAL

El costo de personal estimado para el proyecto se puede observar en la Tabla 29, se consideran los principales roles de la metodología Scrum.

Tabla 29. Costos de personal

Ítem	Cargo / Rol	Cant	Monto Mensual (S/.) (*)	Monto Total del Proyecto (S/.)
01	Scrum Master	1	3,000.00	3,000.00
02	Analista de Desarrollador	1	2,400.00	2,400.00
Total (Personal)				S/.7,400.00

Fuente: Elaborado por el autor

(*) El Monto que se está considerando para los sueldos, ya incluye los costos que la empresa tiene que pagar por seguros, AFP, entre otros)

COSTOS DE INFRAESTRUCTURA TECNOLÓGICA

En la Tabla 30 se muestra el costo de infraestructura tecnológica, se toma en cuenta servicios de despliegue continuo para mejorar la actualización del servicio, así como el bajo costo que presentan.

Tabla 30. Costos de infraestructura tecnológica

Ítem	Descripción del Servidor/Plataforma como servicio	Cant	Costo Mensual (S/.)	Costo Anual (S/.)
1	Netlify	1	35.00	420.00
2	Firebase Firestore	1	0.00	0.00
Total (Infraestructura Tecnológica)				S/.420.00

Fuente: Elaborado por el autor

COSTOS DE INFRAESTRUCTURA Y MOBILIARIO

En este apartado están considerados los elementos usados por el desarrollador por lo cual son sólo considerados elementos básicos y esenciales, véase la Tabla 31.

Tabla 31. Costos de infraestructura y mobiliario

Ítem	Descripción	Cant	Costo Unitario (S/.)	Costo Total (S/.)
1	Escritorios	1	700.00	700.00
2	Sillas	1	250.00	250.00
Total (Infraestructura y Mobiliario)				S/.950.00

Fuente: Elaborado por el autor

OTROS COSTOS

Se tienen muy pocos costos extra en el proyecto debido a que principalmente utiliza tecnologías web, véase la Tabla 32.

Tabla 32. Otros costos

Ítem	Descripción	Cant	Costo Unitario (S/.)	Costo Total (S/.)
1	Impresoras	1	950.00	950.00
Total (otros)				S/.950.00

Fuente: Elaborado por el autor

RESUMEN LOS COSTOS DEL PROYECTO (ANUAL):

Finalmente se muestran los costos totales del proyecto en la Tabla 33.

Tabla 33. Resumen de costos del proyecto

Tipo de Costo	Total (S/)
Personal	S/ 7400.00
Infraestructura Tecnológica	S/ 420.00
Infraestructura Mobiliaria	S/ 950.00
Otros	S/ 950.00
Total (Costos del Proyecto)	S/ 9720.00

Fuente: Elaborado por el autor

4.3. DISEÑO DE LA SOLUCIÓN

4.3.1. Categoría e interacción de Usuarios

La comunicación que existe entre los usuarios de la aplicación generalmente es directa entre el usuario directo y el Asistente Virtual, aunque en ciertas ocasiones en las que el Asistente no puede solucionar las dudas del cliente este es redirigido hacia el community manager de la empresa que se encarga de responder las preguntas de forma personalizada esto es representado en la Figura 13.

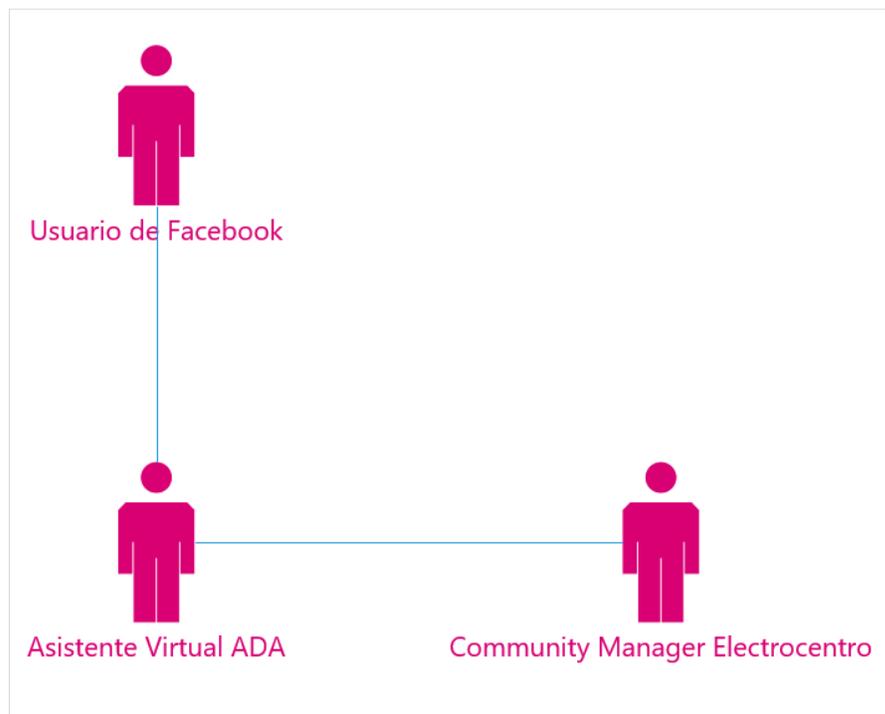


Figura 13. Interacción de usuarios asistente virtual

Fuente: Elaborado por el autor

4.3.2. Mapa de Navegación del Asistente Virtual

A continuación, en la Figura 14, se detalla la estructura de navegación del Asistente a través de todas las opciones que presenta

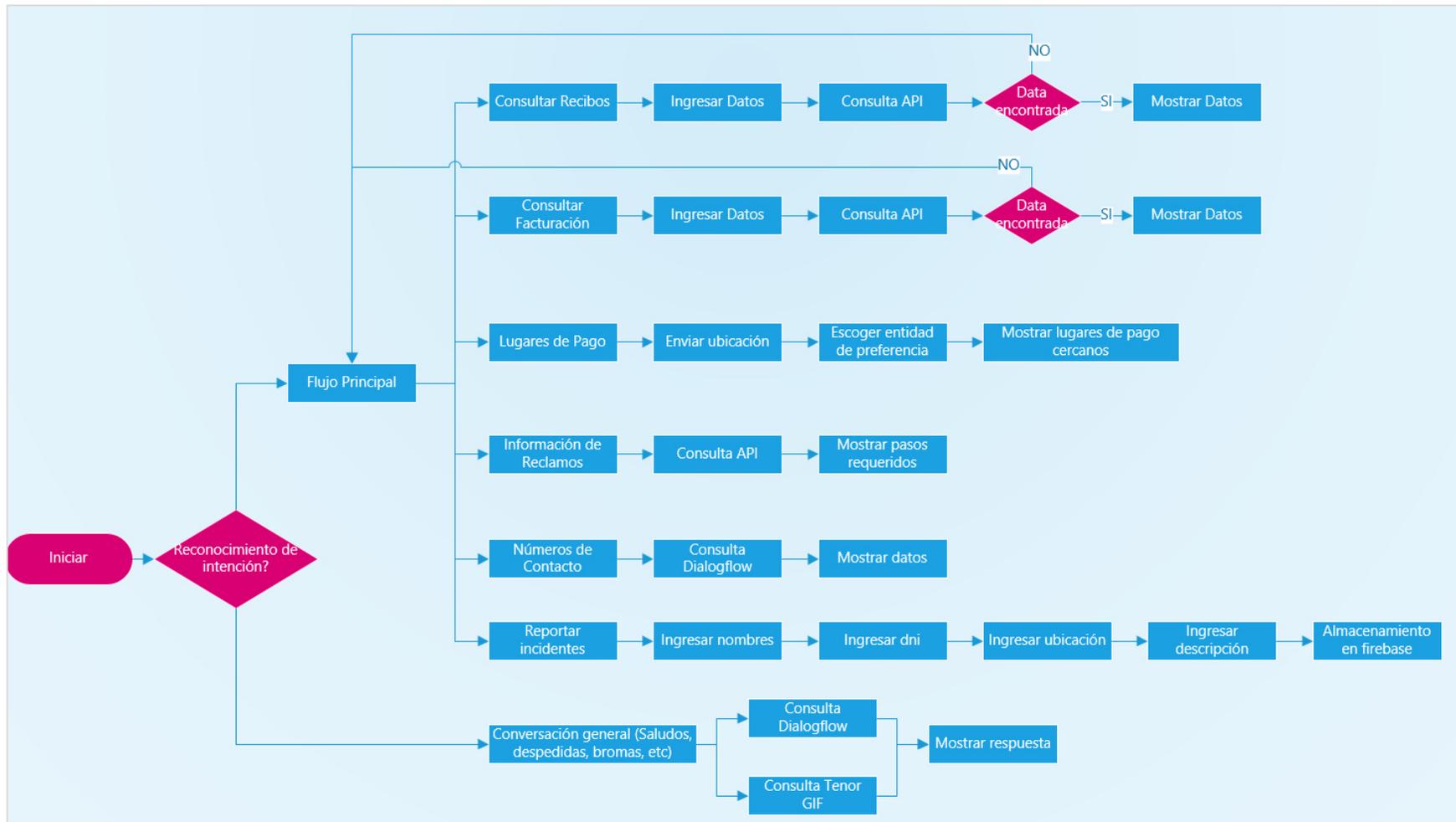


Figura 14. Mapa de Navegación Asistente Virtual Ada Electrocentro

Fuente: Elaborado por el autor

4.3.3. Plantilla de la Página Principal

El Asistente Virtual desarrollado utiliza la plataforma Messenger de Facebook para poder comunicarse con los usuarios de Electrocentro y de esta manera poder brindarles información realizando consultas al API de la empresa y retornando la información a través de mensajes textuales e imágenes.

Al inicio de la conversación se solicitará la aceptación de los términos y condiciones de la siguiente forma tal y como se observa en la Figura 15.



Figura 15. Asistente Virtual Pantalla Principal.
Fuente: Elaborado por el autor

4.3.4. Plantilla de las páginas secundarias

Luego de haber aceptado los términos de la aplicación se procederá a la presentación del Asistente Virtual y al mismo tiempo se mostrarán mediante cartas propias de Facebook Messenger las opciones disponibles para el usuario, véase Figura 16, Figura 17 y Figura 18.

En la primera carta se puede apreciar las opciones: Consultar Recibos, Consultar Deudas y Lugares de Pago.

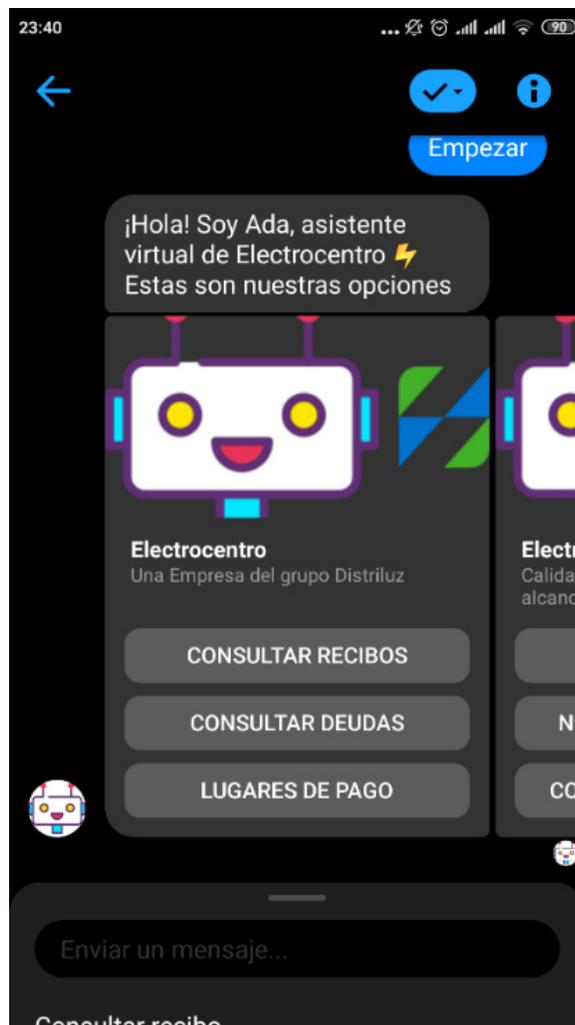


Figura 16. Asistente virtual listado de opciones
Fuente: Tomado del Asistente Virtual desplegado en Messenger

En la segunda carta se puede apreciar las opciones: Guía de Reclamos, Números de Contacto y Consulta de Interrupciones.

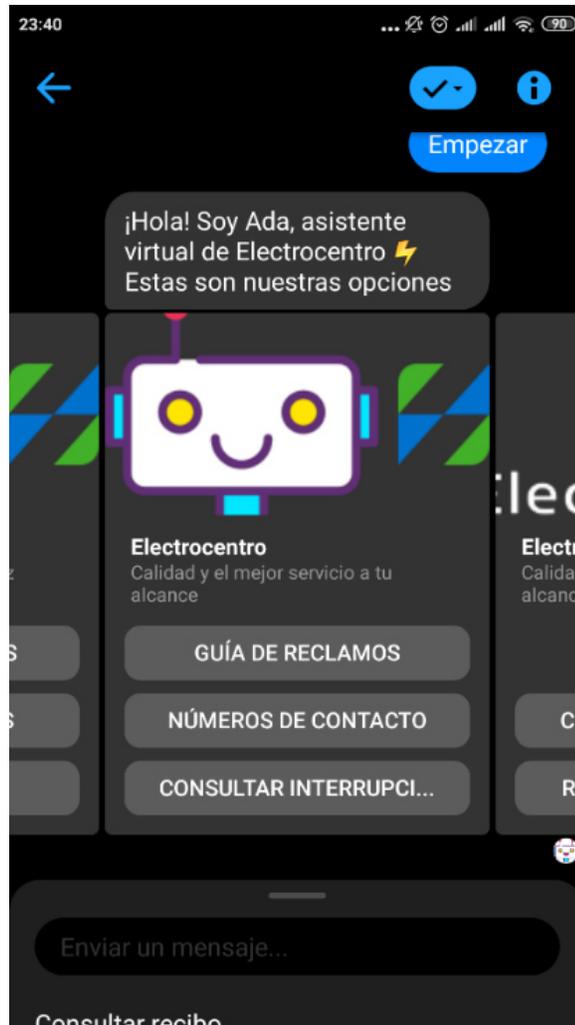


Figura 17. Listado de Opciones del Asistente Virtual
Fuente: Tomado del Asistente Virtual desplegado en Messenger

Finalmente, en la tercera carta se muestran las opciones de Cortes Programados y Reportes de Incidencias.



Figura 18. Listado de opciones del Asistente virtual
Fuente: Tomado del Asistente Virtual desplegado en Messenger

También se posee un menú de acceso rápido permanente en la parte inferior de la pantalla que tiene por defecto las opciones más usadas por los usuarios, el cual se observa en la Figura 19, además de la opción de desplegar el listado completo de opciones rápidamente.

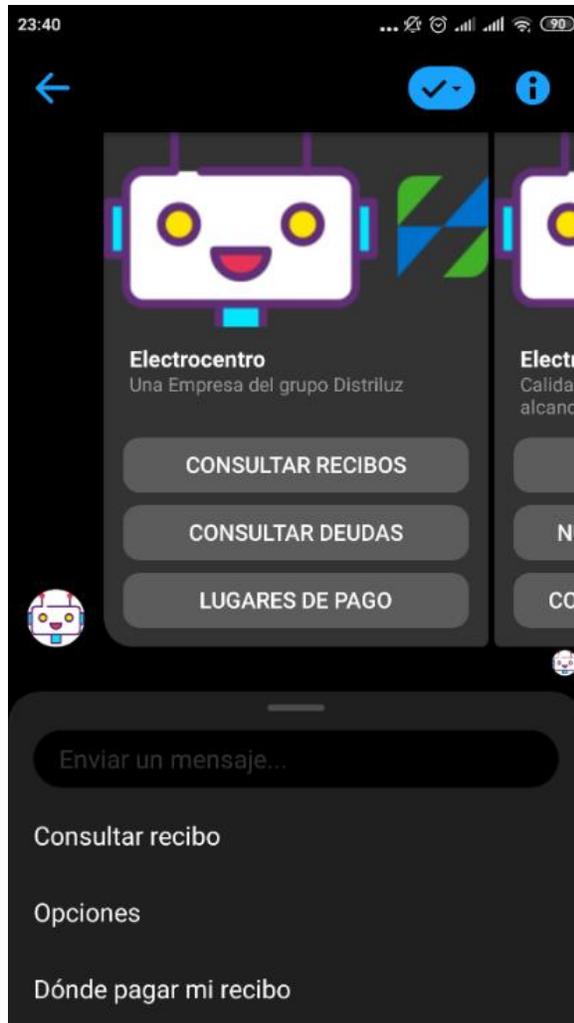


Figura 19. Menú de opciones rápidas
Fuente: Tomado del Asistente Virtual desplegado en Messenger

Cuando se ingresa un texto aleatorio o sin sentido el asistente está programado para responder con una de las tantas respuestas entrenadas si es que se detecta una intención que no se entiende.

A continuación, véase Figura 20, se hace una consulta acerca de los recibos del cliente, para realizar esta consulta el asistente requiere el número de servicio del cliente el cual se puede encontrar en cualquier recibo de los meses anteriores.

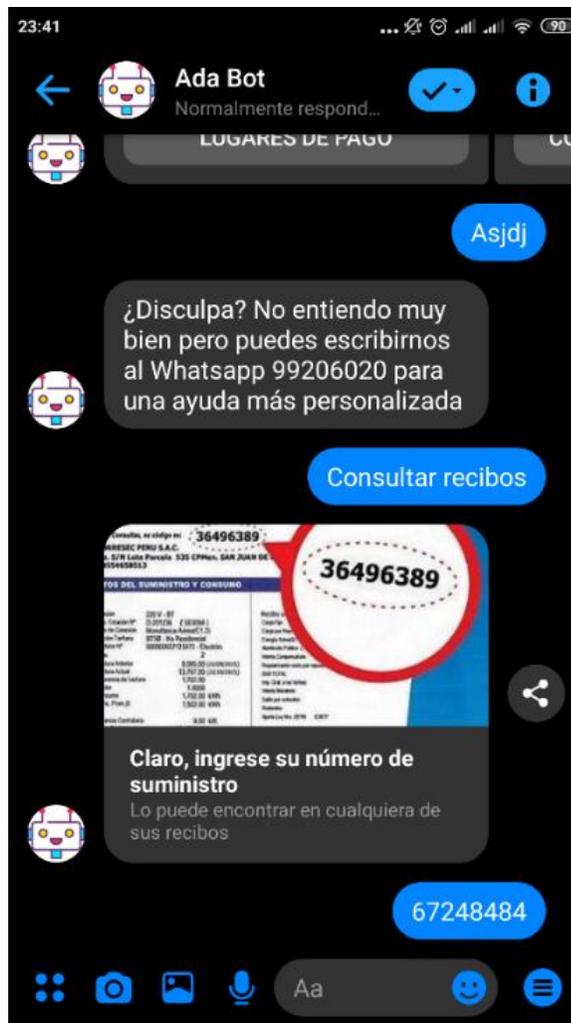


Figura 20. Consulta de recibos del Asistente
Fuente: Tomado del Asistente Virtual desplegado en Messenger

Luego el asistente realiza la consulta al API de Electrocentro la cual está conectada a su Base de datos que contiene la data de los clientes de la empresa y de esta forma obtiene los datos del recibo del cliente y se los muestra a través de una respuesta textual en Messenger, véase Figura 21.

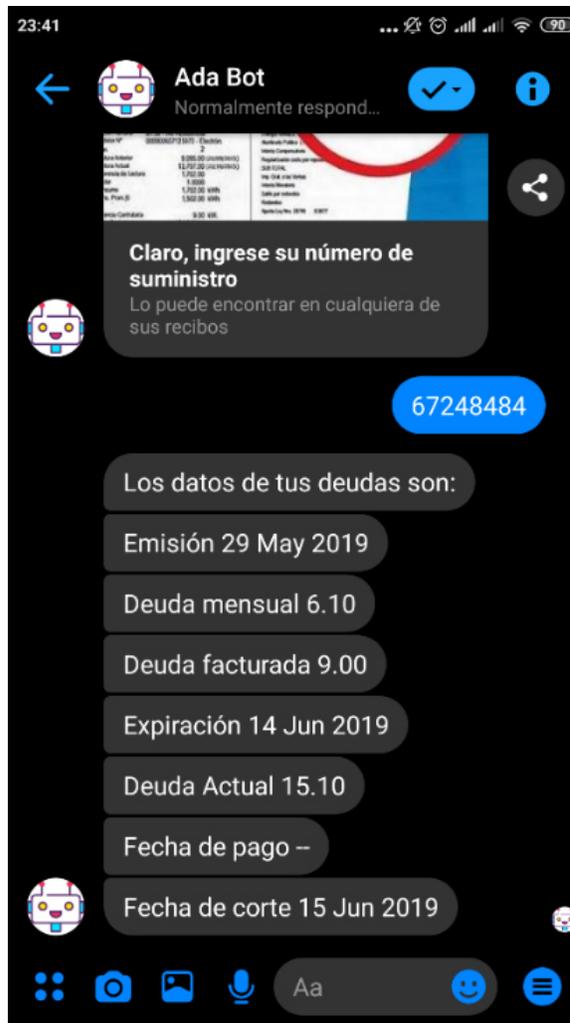


Figura 21. Resultado de consulta de recibos
Fuente: Tomado del Asistente Virtual desplegado en
Messenger

Para realizar la consulta de lugares de pago el cliente sólo tiene que ingresar un mensaje textual que represente esa intención, a continuación, se le pedirá que seleccione una institución bancaria de su preferencia, y para mostrarle los lugares más cercanos se le requerirá su ubicación actual que es tomada a través de la aplicación de Facebook Messenger, véase la Figura 22.

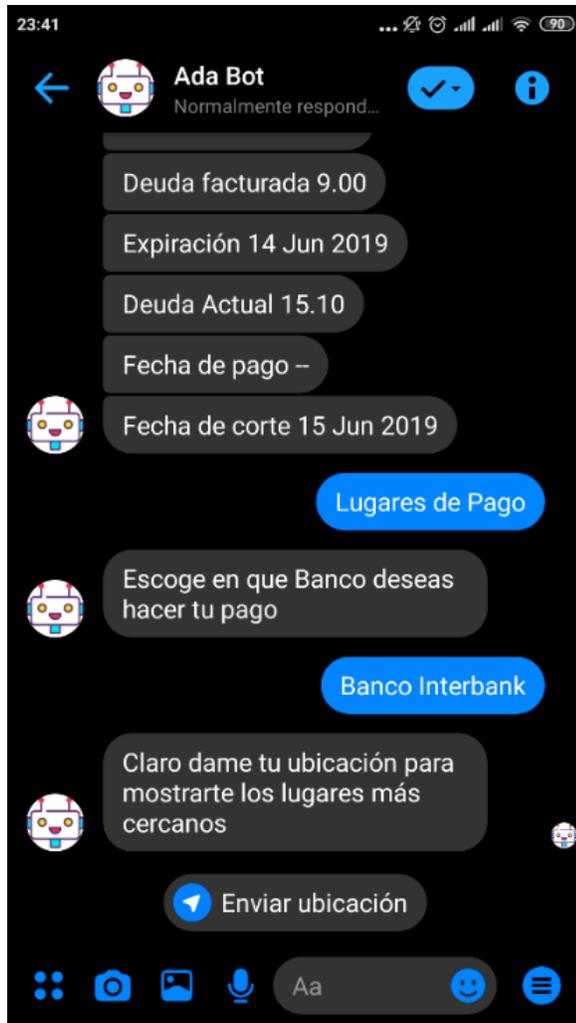


Figura 22. Consulta de lugares de pago
Fuente: Tomado del Asistente Virtual desplegado en Messenger

El cliente sólo debe enviar su ubicación, tal y como se observa en la Figura 23, para que el Asistente pueda realizar la consulta al API de Google Places.

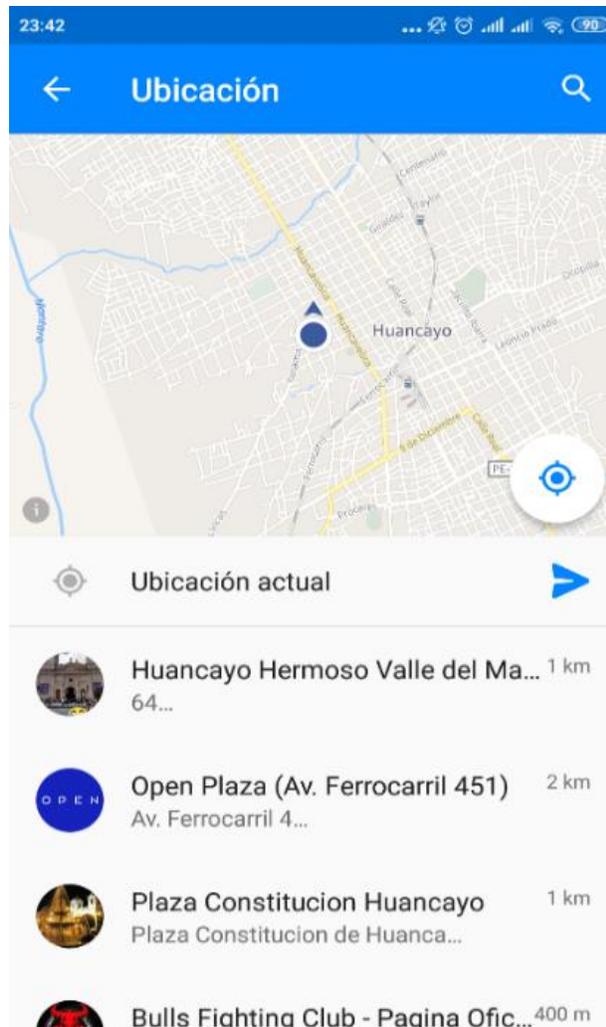


Figura 23. Enviar ubicación actual
Fuente: Tomado del Asistente Virtual desplegado
en Messenger

El cliente sólo debe enviar su ubicación para que el Asistente pueda realizar la consulta al API de Google Places y retorna un listado con los lugares de pago más cercanos, el resultado se puede apreciar en la Figura 24.

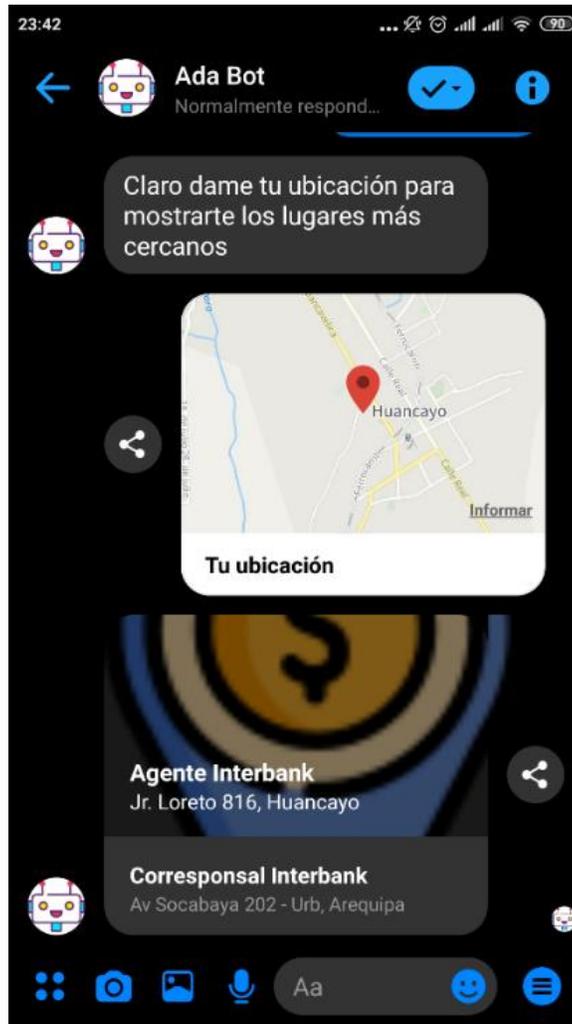


Figura 24. Listado de lugares más cercanos
Fuente: Tomado del Asistente Virtual desplegado en Messenger

Acerca de la consulta de guía de reclamos se puede ingresar una intención que represente que se quiere conocer los pasos para realizar un reclamo, luego el asistente pasa a mostrar las imágenes que detallan los pasos detallados para la realización de un reclamo, esto se puede observar en la Figura 25.

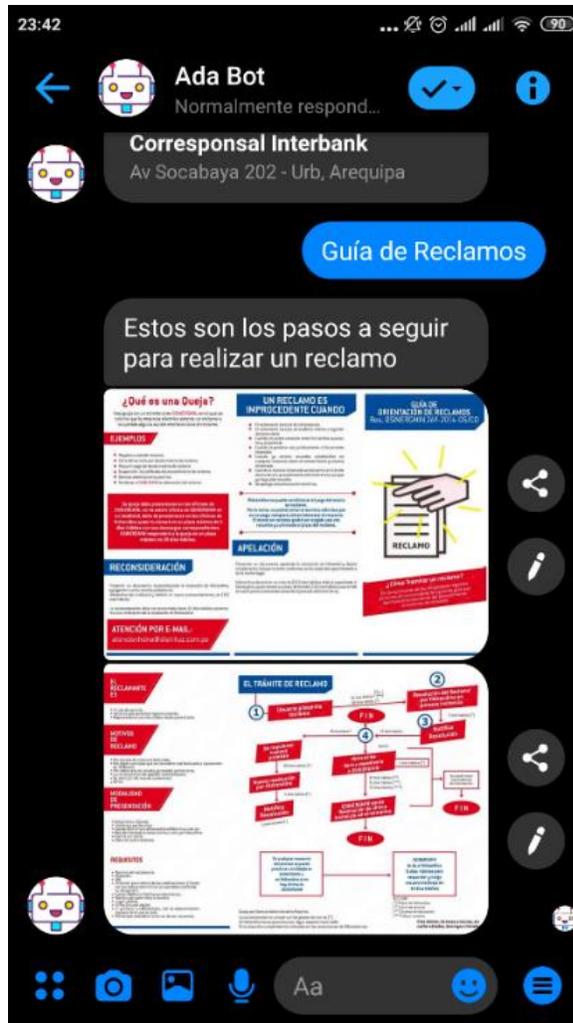


Figura 25. Información de reclamos
Fuente: Tomado del Asistente Virtual desplegado en Messenger

Del mismo modo para consultar los números de contacto se ingresa una intención que la represente y el asistente muestra los números de contacto obtenidos de Electrocentro, véase la Figura 26.

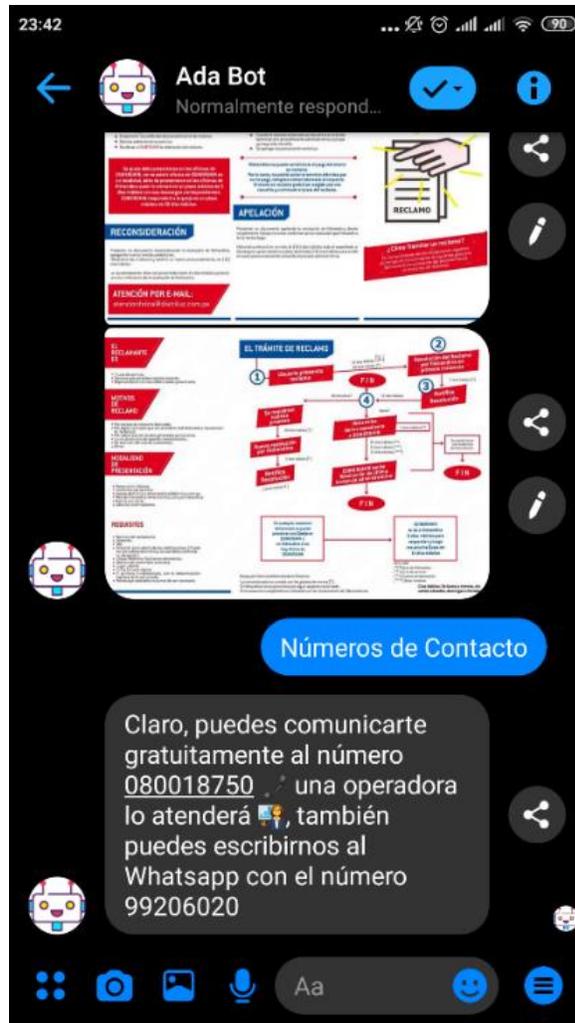


Figura 26. Números de reclamo
Fuente: Tomado del Asistente Virtual desplegado en Messenger

Para la consulta de interrupciones se ingresa una intención similar, como se muestra en la Figura 27, a continuación, el asistente requiere que el usuario ingrese su número de servicio con ello se realiza la consulta al API de la empresa y se puede mostrar al cliente si su servicio presentará interrupciones.



Figura 27. Consulta de interrupciones
Fuente: Tomado del Asistente Virtual desplegado en Messenger

4.3.5. Modelo Físico de la base de datos

Para la base de datos del Asistente Virtual se utilizó el servicio de Firebase Firestore que ofrece un sistema de base de datos no relacional, esta base de datos se utiliza para almacenar los reportes de incidencias de los clientes de Electrocentro y puedan así ser visualizados y evaluados por el personal a cargo véase la Figura 28.

En este modelo de Base de Datos se contempla principalmente la parte de incidencias que van a ser reportadas a través del asistente virtual, así como la autenticación y administración de etiquetas para cada incidencia, el resto de datos que son consumidos por la aplicación tales como datos de los recibos, deudas de los últimos meses, cortes programados, entre otros, se encuentran en la base de datos de la empresa y son consumidos a través de un servicio REST API, por ende, no se puede mostrar su modelo de base de datos.

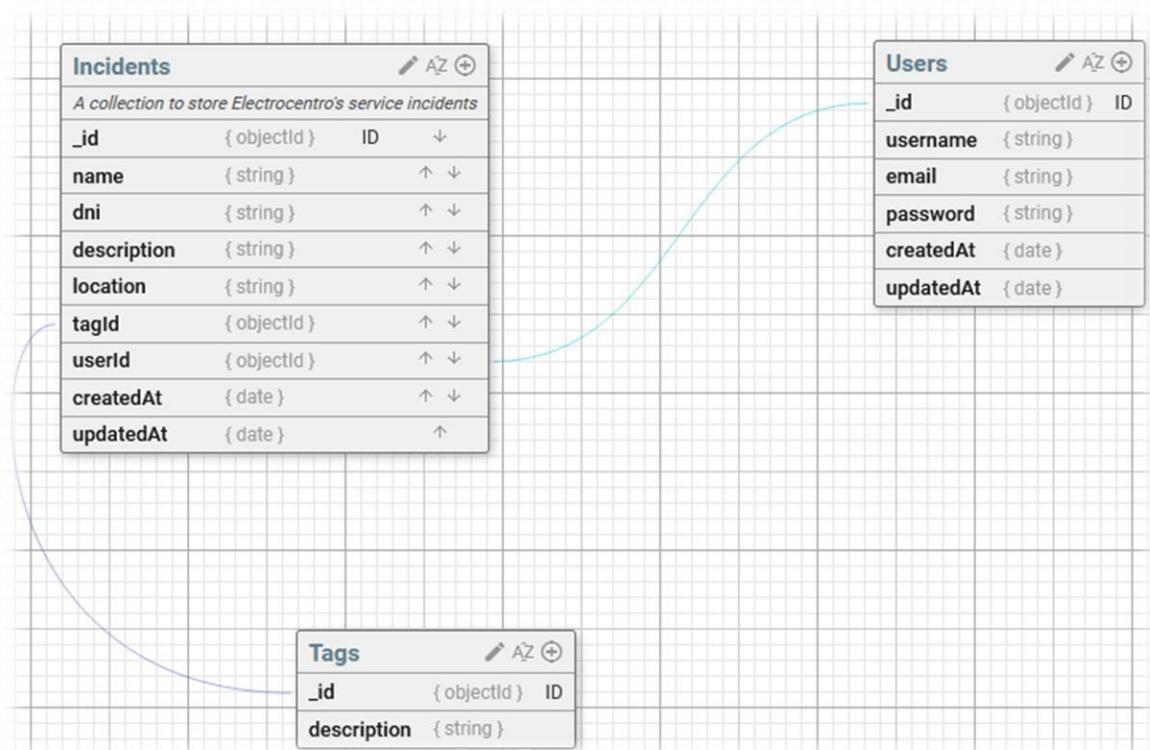


Figura 28. Diagrama de Base de Datos para el Asistente Virtual
Fuente: Elaborado por el autor

CAPITULO V

CONSTRUCCIÓN DE LA SOLUCIÓN

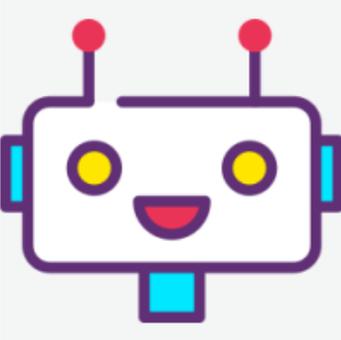
5.1. CONSTRUCCIÓN

5.1.1. Desarrollo del Agente Conversacional en Dialogflow

El Agente Conversacional se desarrolló utilizando la versión 2 de Dialogflow, herramienta desarrollada por Google para la implementación de agentes que pueden ser entrenados para funcionar en diversas plataformas de mensajería. En la Figura 29 se muestra los detalles del agente, tales como el ID del proyecto, su cuenta de servicio, el idioma que soporta, la zona horaria que va a soportar, así como la versión que se usa actualmente.

ada-bot

General Languages ML Settings Export and Import Environments Speech Share Advanced



DESCRIPTION

Asistente Virtual desarrollado para Electrocentro S.A.

DEFAULT TIME ZONE

(GMT-5:00) America/New_York

Date and time requests are resolved using this timezone.

GOOGLE PROJECT

Project ID	ada-bo [REDACTED]
Service Account ⓘ	dialogflow [REDACTED]

API VERSION

V2 API
Use [Cloud API](#) as default for the agent. Your webhook will receive and return [V2 format messages](#).

BETA FEATURES

Enable beta features and APIs
Be the first to get access to the newest features and latest APIs. ([Full V2-beta API reference](#))

Figura 29. Perfil del Agente en Dialogflow
Fuente: Tomado de la página de Dialogflow del proyecto

Las intenciones especificadas en la Figura 30, permiten el correcto funcionamiento del Asistente Virtual, además de las ventajas que ofrece la plataforma como son la expansión de intenciones nuevas para mejorar la funcionalidad del asistente, las cuales pueden ser desarrolladas en siguientes versiones del proyecto.

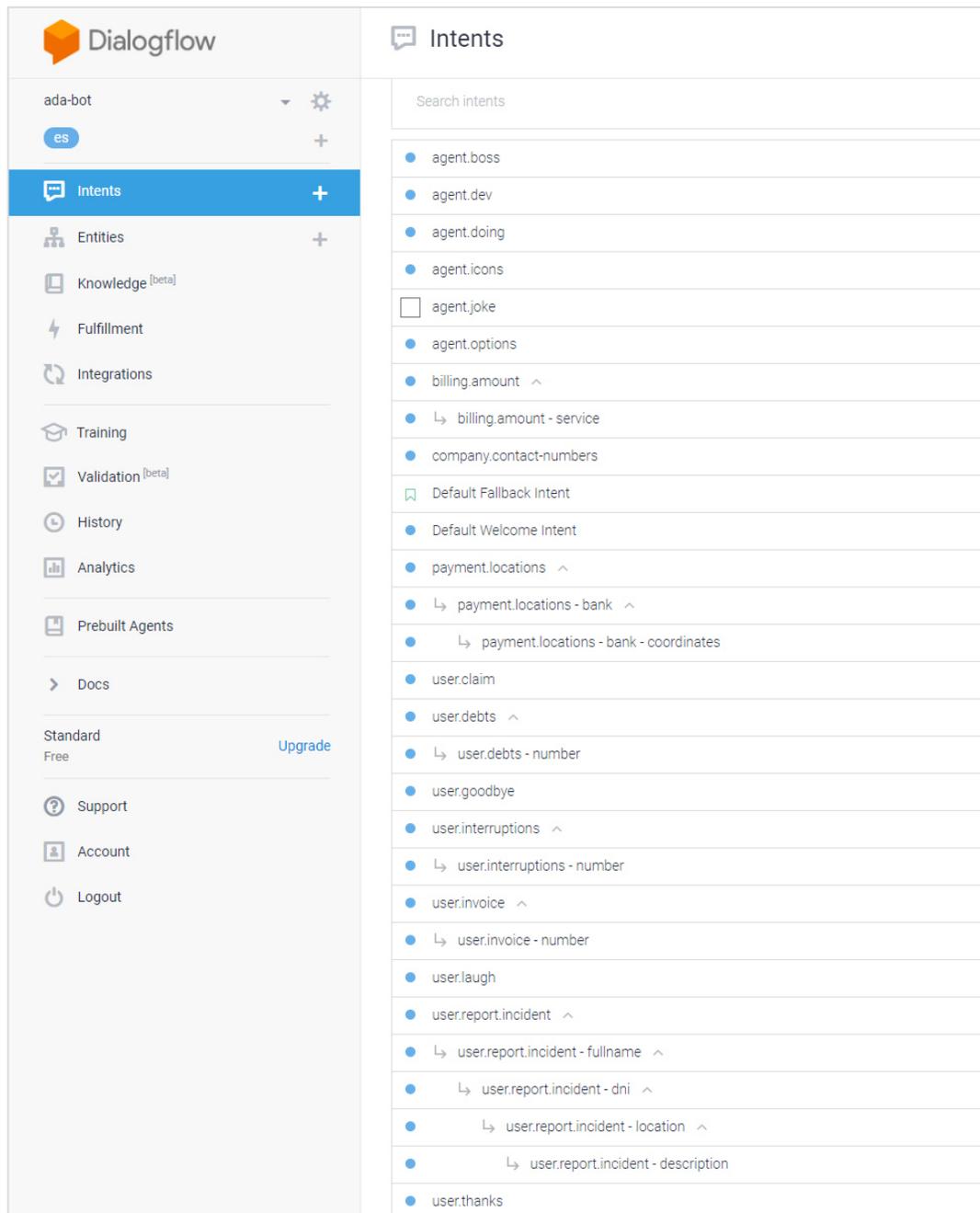


Figura 30. Listado de intenciones del Agente en Dialogflow
Fuente: Tomado de la página de Dialogflow del proyecto

Además, Dialogflow por defecto permite una integración mucho más rápida con servicios como Facebook Messenger que se utiliza para este proyecto y otras plataformas como Slack, Twitter, Skype, entre otros.

Este servicio de integración rápida genera un webhook que se conecta con la aplicación creada en Facebook Developers, véase Figura 31, la cual está directamente conectada a la página de Facebook del Asistente Virtual.

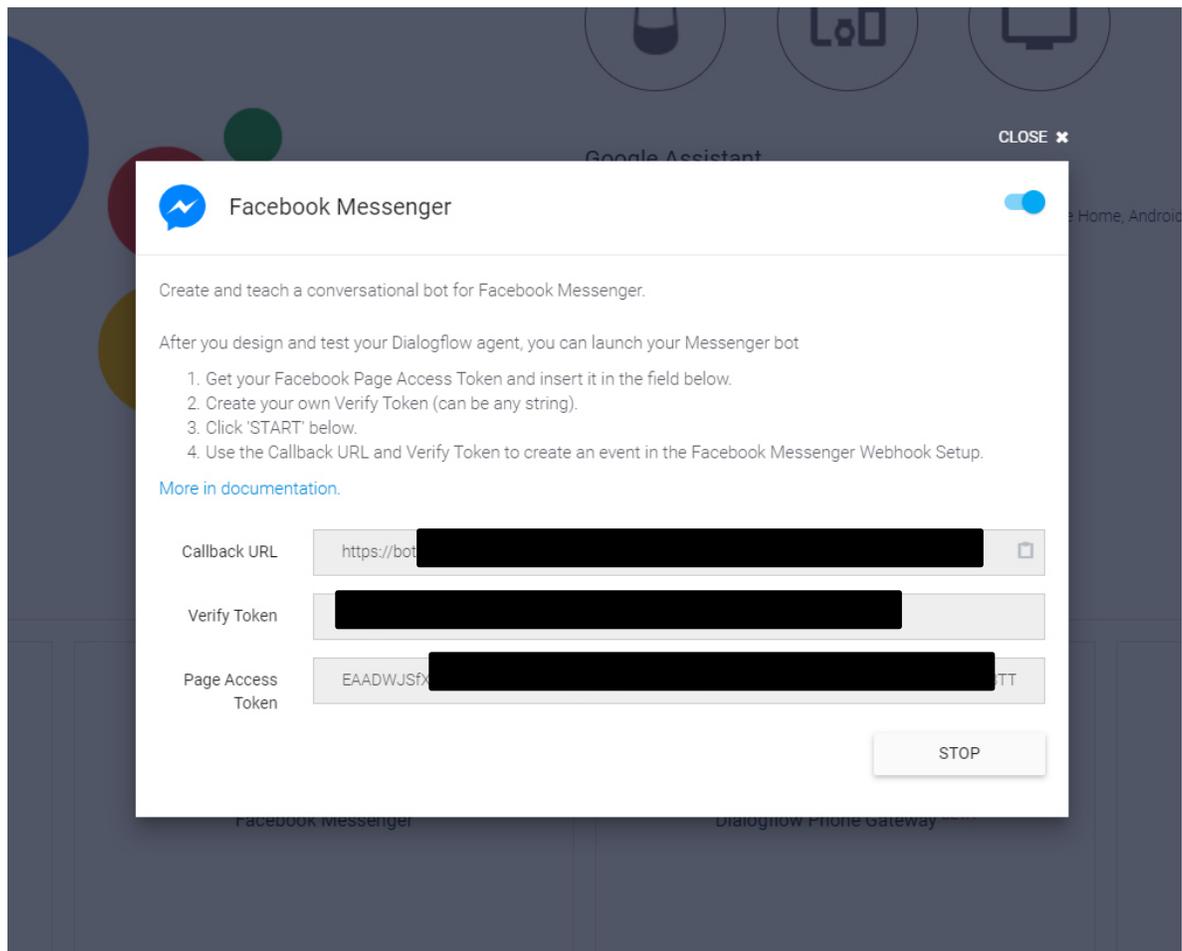


Figura 31. Integración con Facebook Messenger
Fuente: Tomado del proyecto de Dialogflow del asistente

5.1.2. Desarrollo del Webhook para Extensión de Funcionalidad del Asistente Virtual

Para el desarrollo del Webhook principal que incrementa la funcionalidad del Agente de Dialogflow se utilizó Node JS y una arquitectura serverless

desplegada en la plataforma Netlify. El webhook, véase Figura 32, se conecta con el agente y puede ser activado sólo en las intenciones que así lo requieran de esta forma se optimiza las llamadas que se hacen al webhook, tal y como se muestra en la Figura 33.

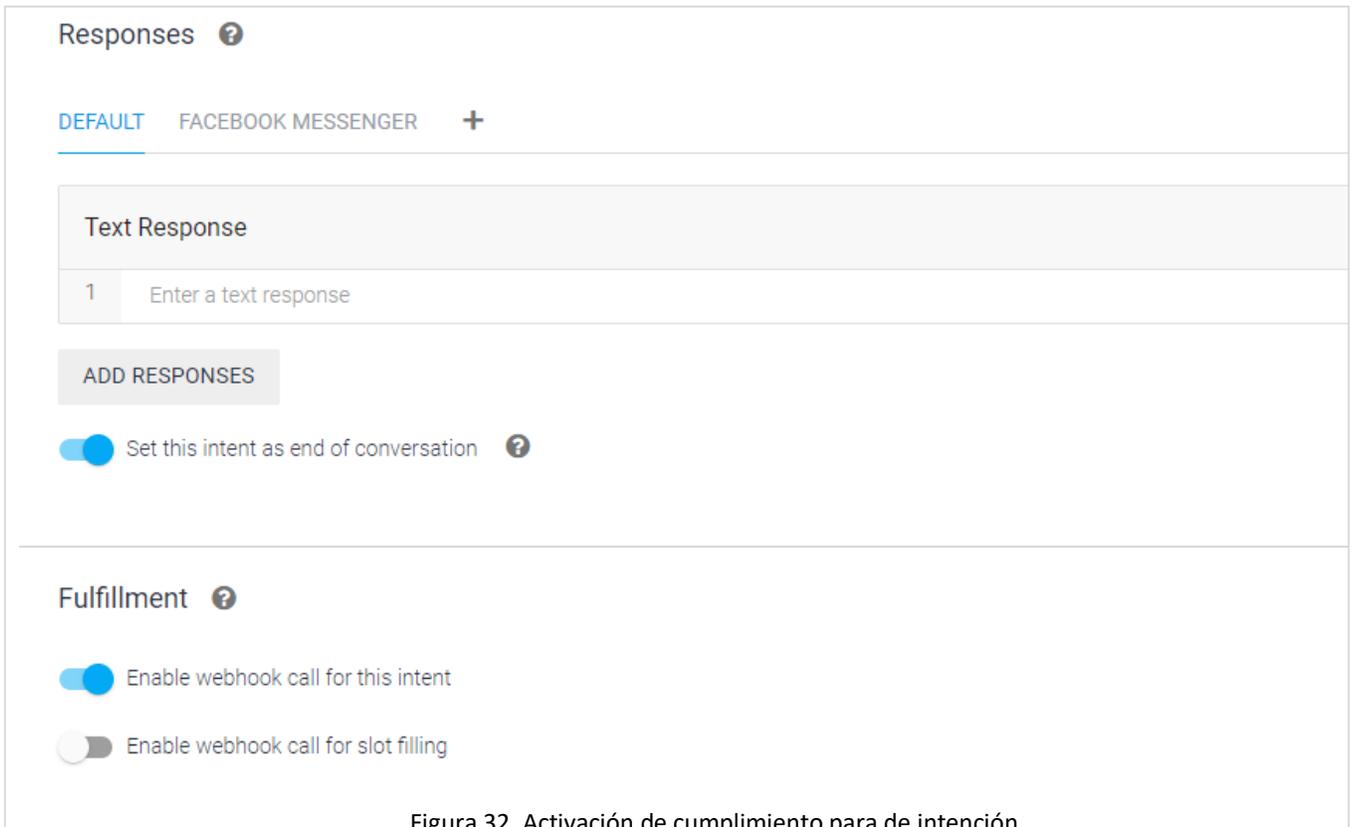


Figura 32. Activación de cumplimiento para de intención
Fuente: Tomado de proyecto en Dialogflow del asistente

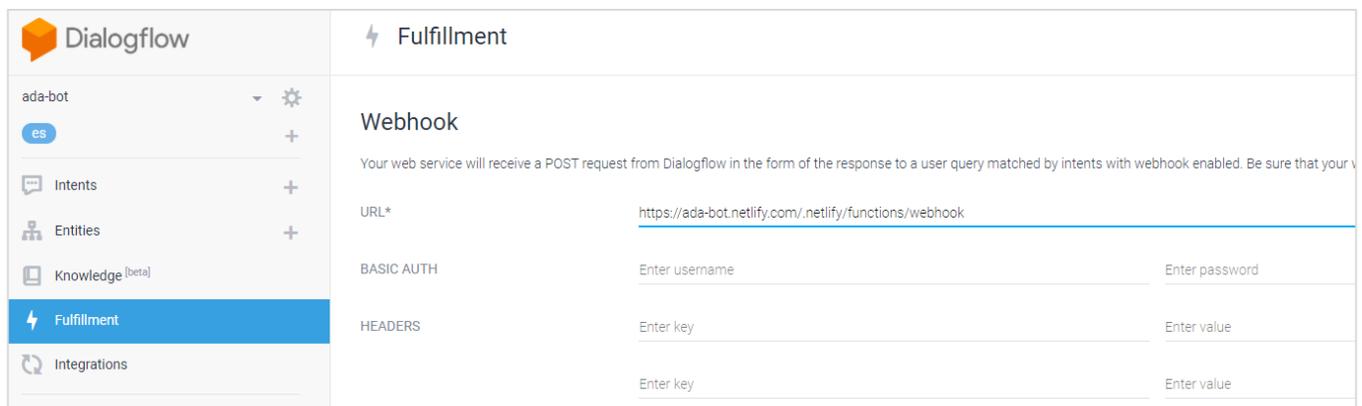


Figura 33. Integración de cumplimiento con servicio serverless
Fuente: Tomado del proyecto de Dialogflow del asistente

La aplicación es alojada en Github como repositorio para su manejo y control de versiones, pero además esto permite conectarse directamente con Netlify para un despliegue continuo, es decir que cada vez que se realice un cambio en el código y se suba al repositorio, este se sincroniza automáticamente con Netlify y publica una nueva versión de la aplicación, véase Figura 34.

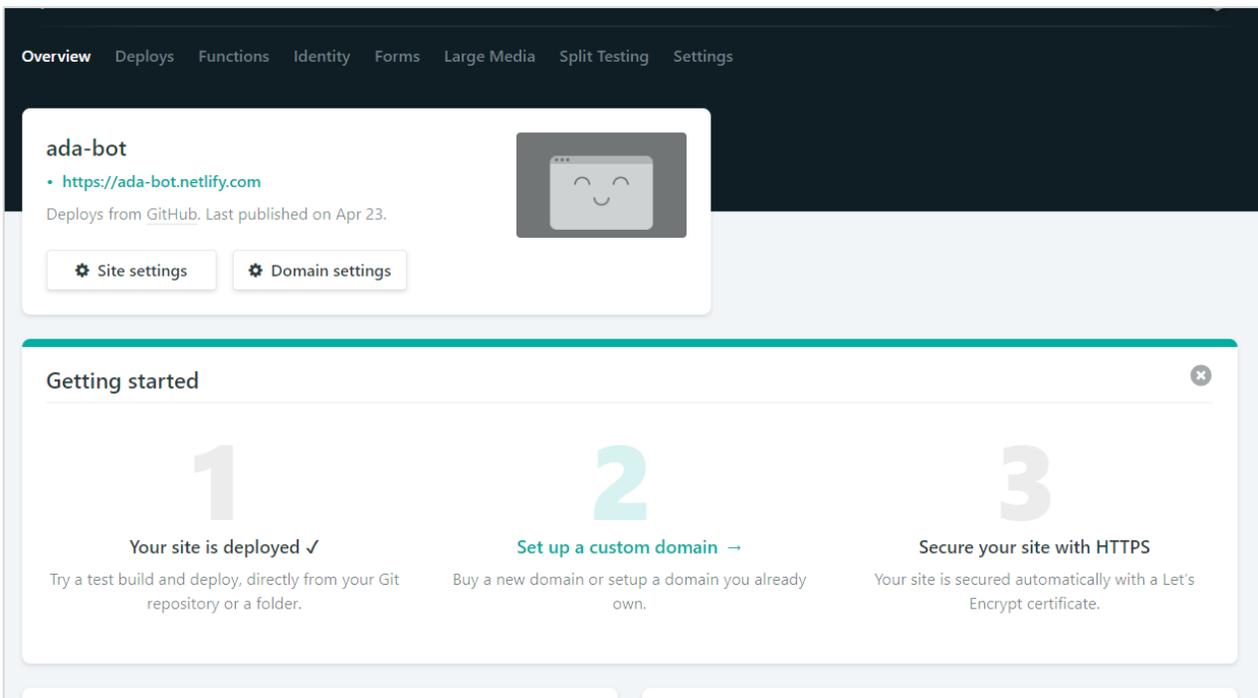


Figura 34. Aplicación serverless desplegada en Netlify
Fuente: Tomado del proyecto del asistente en Netlify

5.1.3. Desarrollo de funciones Serverless

Al igual que el webhook principal que está desarrollado con Node JS y está desplegado usando Netlify y Github, cada una de estas funciones realiza un trabajo único y son mapeadas dentro del webhook principal del sistema, en la Figura 35 se muestra la estructura del proyecto en Javascript.

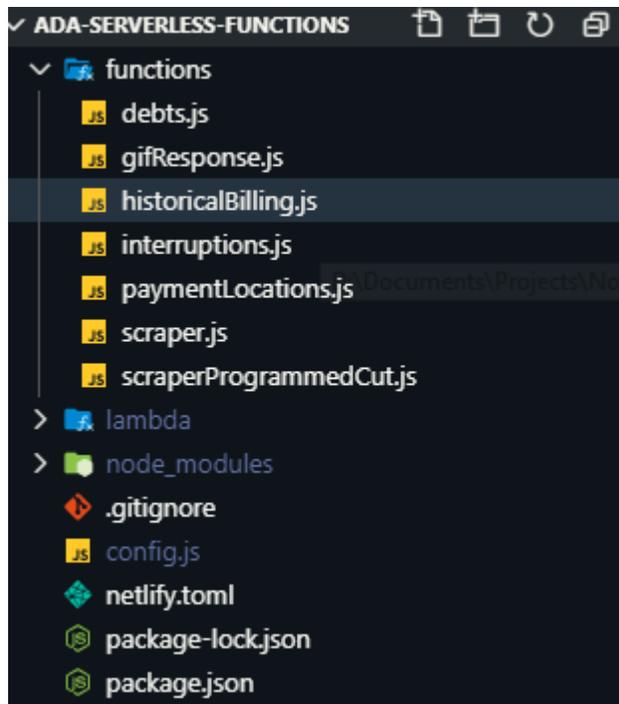


Figura 35. Listado de funciones serverless del proyecto
Fuente: Elaborado por autor

Estas funciones se encargan de consumir directamente los servicios del servicio REST API de Electrocentro realizando consultas (GET, POST) utilizando los números de servicio de los clientes de la empresa, véase la Figura 36, para luego procesar la data cruda recibida, debido a que esta es consumida por muchos otros servicios y no se estructura de la manera que la necesita el proyecto. Esta data debe ser formateada correctamente para poder ser luego enviada a la aplicación de Facebook Messenger y finalmente mostrada al usuario, véase la Figura 37.

```

POST http://servicios.distriluz.com.pe:51 200 OK 78 ms 2.3 KB A Year Ago
JSON Auth Query Header Docs Preview Header Cookie Timeline
1 {"ParametroControlTag":"idnoservicio|67248484"}
2
3
4
5
6
7
8
9
10
11
12
13
14
1 {
  "ListaInformacion": [
    {
      "Cabecera": "",
      "Configuracion":
        "colorTextoTitulo|#003591;colorTextoSubTitulo|#C0C0C0;ponerMarco|false;colorFondoMarco|#EDEF0;colorTextoCabecera|#E74C3C;colorTextoPie|#C0C0C0;EsNegritaCabecera|false;EsFondoMarcoTransparente|false",
      "ListaRegistro": [
        "clave|Periodo;valor|201903;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072*clave|Emisi3n;valor|29 Mar 2019;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072*clave|Importe Mes S;valor|6.20;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072*clave|Deuda Facturada S;valor|6.20;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072*clave|Vencimiento;valor|15 Abr 2019;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072*clave|Deuda Actual S;valor|6.20;colorLetraClave|#000000;colorLetraValor|#FFFFFF;colorFondoClave|#FA07A;colorFondoValor|#FF000*clave|Ultimo Pago S;valor|4.60;colorLetraClave|#000000;colorLetraValor|#FFFFFF;colorFondoClave|#FA07A;colorFondoValor|#FF000*clave|Fecha Pago;valor|- -;colorLetraClave|#000000;colorLetraValor|#FFFFFF;colorFondoClave|#FA07A;colorFondoValor|#FF000*clave|Fecha Corte;valor|16 Abr 2019;colorLetraClave|#000000;colorLetraValor|#FFFFFF;colorFondoClave|#FA07A;colorFondoValor|#FF000*clave|Recibos en Deuda;valor|1;colorLetraClave|#000000;colorLetraValor|#FFFFFF;colorFondoClave|#FA07A;colorFondoValor|#FF000*clave|Saldo Convenio;valor|-1;colorLetraClave|#000000;colorLetraValor|#FFFFFF;colorFondoClave|#FA07A;colorFondoValor|#FF000*clave|Recibo;valor|0078623065182;colorLetraClave|#000000;colorLetraValor|#FFFFFF;colorFondoClave|#FA07A;colorFondoValor|#FF000*clave|Consumo;valor|0.0000;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072*clave|Telefono;valor|;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072*clave|Correo;valor|;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072*clave|PeriodoDes;valor|2019-Mar;colorLetraClave|#000000;colorLetraValor|#000000;colorFondoClave|#FA07A;colorFondoValor|#FA072"
      ]
    }
  ],
  "Pie": "",
  "TipoPanel": 3,
  "Titulo": "ESTADO DE CUENTA"
}

```

Figura 36. Consulta al Servicio REST API de Electrocentro
Fuente: Elaborado por autor

POST http://localhost:9000/historicalBilling Send 200 OK 78 ms 1489 B 6 Months Ago

JSON Auth Query Header 1 Docs

```

1 {
2   "IdNroServicio": 67248484
3 }

```

Preview Header 4 Cookie Timeline

```

1 {
2   "historicalBilling": [
3     {
4       "bill": "Periodo Mar-19:Recibo Nro. 0078623065182",
5       "issuance": "Fecha facturación: Mar-19",
6       "expiration": "Fecha vencimiento: 5 Abr",
7       "currentDebt": "Monto del mes: :6.20",
8       "expiredDebt": "Monto del vencida: 6.20"
9     },
10    {
11     "bill": "Periodo Abr-19:Recibo Nro. 0078623205199",
12     "issuance": "Fecha facturación: Abr-19",
13     "expiration": "Fecha vencimiento: 5 May",
14     "currentDebt": "Monto del mes: :9.00",
15     "expiredDebt": "Monto del vencida: 6.20"
16    },
17    {
18     "bill": "Periodo May-19:Recibo Nro. 0078623345155",
19     "issuance": "Fecha facturación: May-19",
20     "expiration": "Fecha vencimiento: 4 Jun",
21     "currentDebt": "Monto del mes: :6.10",
22     "expiredDebt": "Monto del vencida: 9.00"
23    },
24    {
25     "bill": "Periodo Jun-19:Recibo Nro. 0078623486251",
26     "issuance": "Fecha facturación: Jun-19",
27     "expiration": "Fecha vencimiento: 5 Jul",
28     "currentDebt": "Monto del mes: :6.30",
29     "expiredDebt": "Monto del vencida: 15.10"
30    },
31    {
32     "bill": "Periodo Jul-19:Recibo Nro. 0078623627250",
33     "issuance": "Fecha facturación: Jul-19",
34     "expiration": "Fecha vencimiento: 5 Ago",
35     "currentDebt": "Monto del mes: :6.20",
36     "expiredDebt": "Monto del vencida: 0.00"
37    },
38    {
39     "bill": "Periodo Ago-19:Recibo Nro. 0078623768671",
40     "issuance": "Fecha facturación: Ago-19",
41     "expiration": "Fecha vencimiento: 4 Set",
42     "currentDebt": "Monto del mes: :10.70",
43     "expiredDebt": "Monto del vencida: 0.00"

```

Figura 37. Consulta a Servicio del Asistente Virtual con Data Formateada

Fuente: Elaborado por autor

5.2. PRUEBAS DEL PRODUCTO TECNOLÓGICO

5.2.1. Pruebas de componentes

Las pruebas de componentes fueron realizadas por el área de TI de la Empresa Electrocentro Huancayo, y fueron enfocados en los siguientes aspectos:

La comprobación de la intención de bienvenida se da mediante el ingreso de mensajes de saludo por parte del usuario y a continuación se muestra el mensaje de bienvenida preparado, véase la Tabla 34.

Tabla 34. Pruebas de Componentes 1

Objetivos de la Prueba	Comprobar intención de bienvenida - saludo
Técnicas	Realizar envío de mensajes relacionados a un saludo
Código Involucrado	Funcionalidad manejada directamente por el Agente desarrollado en Dialogflow
Caso de Prueba	Formato de casos de pruebas
	Tipo de prueba: Unitaria (Caja Blanca) Objetivo: Validar que la pueda captar la intención con mensajes relacionados
	Caso No. 1
	Descripción: Mensajes variados con intención de saludo
	Entradas: Hola, Hola cómo estas, Hola qué tal, Hey
	
Resultado	El asistente detecto la intención con los mensajes enviados
Observaciones	Si la intención no fue entendida es redirigida a la intención de no comprensión

Fuente: Elaborado por el autor

El listado de opciones disponibles es mostrador al ingresar algún mensaje que refleje esta intención, además es accesible a través del botón del menú rápido, véase la Tabla 35.

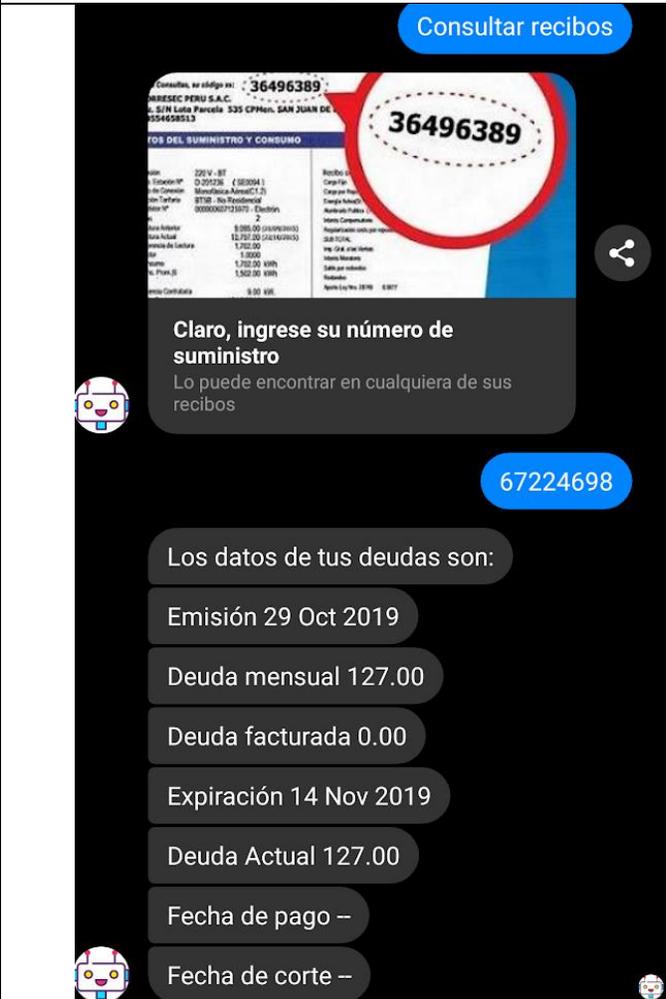
Tabla 35. Pruebas de Componentes 2

Objetivos de la Prueba	Comprobar intención de opciones
Técnicas	Realizar envío de mensajes relacionados a la solicitud de opciones
Código Involucrado	Funcionalidad manejada directamente por el Agente desarrollado en Dialogflow, incluyendo imágenes y detalles
Caso de Prueba	Formato de casos de pruebas
	Tipo de prueba: Unitaria (Caja Blanca) Objetivo: Validar que la pueda captar la intención con mensajes relacionados
	Caso No. 1
	Descripción: Mensajes variados con intención de listar opciones
	Entradas: Opciones, Dame las opciones, me puedes decir las opciones del sistema, qué opciones tienes
	
Resultado	El asistente detecto la intención con los mensajes enviados
Observaciones	

Fuente: Elaborado por el autor

En la Tabla 36, se aprecia que al ingresar la intención relacionada con la obtención del recibo del mes y posteriormente ingresar el número de recibo del cliente se obtienen los datos de su recibo.

Tabla 36. Pruebas de Componentes 3

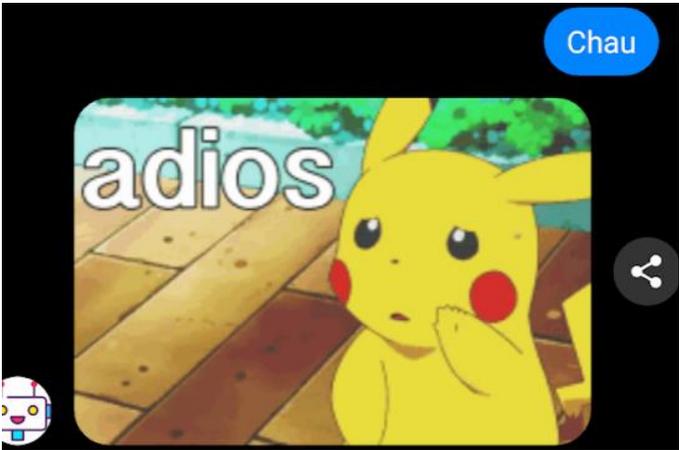
Objetivos de la Prueba	Comprobar intención de recibo del mes
Técnicas	Realizar envío de mensajes relacionados a la obtención del recibo del mes
Código Involucrado	Funcionalidad desarrollada en Node JS véase Anexo
Caso de Prueba	<p>Formato de casos de pruebas</p> <p>Tipo de prueba: Unitaria (Caja Blanca) Objetivo: Validar que la pueda captar la intención con mensajes relacionados</p> <p>Caso No. 1</p> <p>Descripción: Mensajes variados con intención de obtención del recibo del mes</p> <p>Entradas: Dame mi recibo, cuánto debo, dime mi deuda</p> 

Resultado	El asistente detecto la intención con los mensajes enviados
Observaciones	

Fuente: Elaborado por el autor

Al realizar las pruebas para comprobar la intención de despedida se ingresan mensajes que reflejen esta intención, a continuación, se obtiene una imagen animada a través de Tenor GIF la cual es mostrada al usuario, véase la Tabla 37.

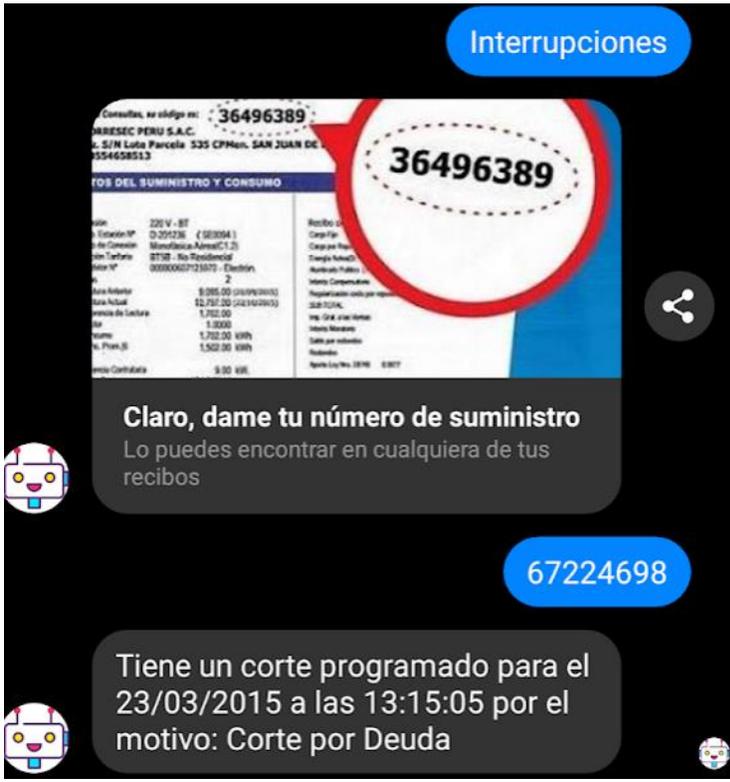
Tabla 37. Pruebas de Componentes 4

Objetivos de la Prueba	Comprobar intención de despedida
Técnicas	Realizar envío de mensajes relacionados a una despedida
Código Involucrado	Funcionalidad desarrollada en Node JS y conectividad con el API de Tenor GIF, véase anexo
Caso de Prueba	Formato de casos de pruebas
	Tipo de prueba: Unitaria (Caja Blanca) Objetivo: Validar que la pueda captar la intención con mensajes relacionados
	Caso No. 1
	Descripción: Mensajes variados con intención de despedida
	Entradas: Chau, Nos vemos, Bye, Adiós
	
Resultado	El asistente detecto la intención con los mensajes enviados
Observaciones	

Fuente: Elaborado por el autor

Al realizar las pruebas para comprobar la interrupción de servicio se envían mensajes que reflejen esta intención, además del número de servicio, luego el asistente muestra los resultados en pantalla, véase la Tabla 38.

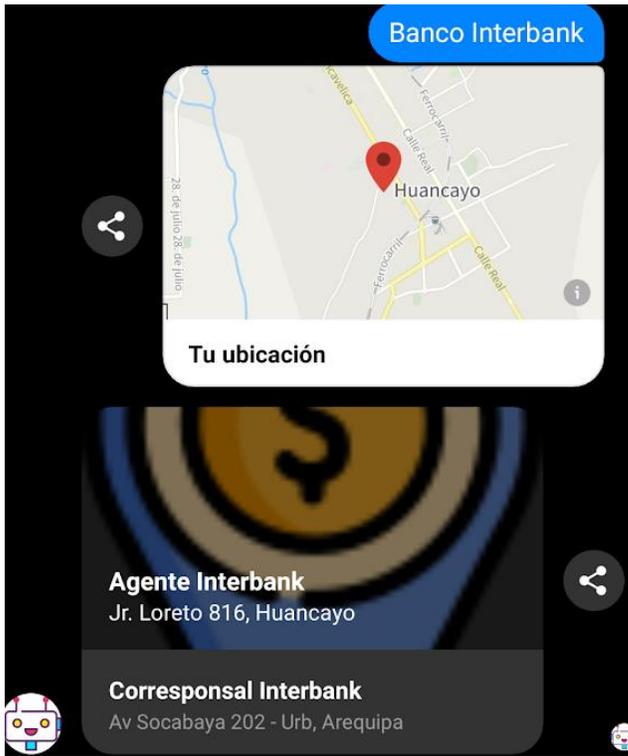
Tabla 38. Pruebas de Componentes 5

Objetivos de la Prueba	Comprobar intención de interrupciones de servicio
Técnicas	Realizar envío de mensajes relacionados a la interrupción de servicio
Código Involucrado	Funcionalidad desarrollada en Node JS, véase anexo
Caso de Prueba	<p>Formato de casos de pruebas</p> <p>Tipo de prueba: Unitaria (Caja Blanca) Objetivo: Validar que la pueda captar la intención con mensajes relacionados</p> <p>Caso No. 1</p> <p>Descripción: Mensajes variados con intención de consulta de interrupciones</p> <p>Entradas: interrupciones, tendré interrupciones, se cortará mi servicio</p> 
Resultado	El asistente detecto la intención con los mensajes enviados
Observaciones	

Fuente: Elaborado por el autor

En la Tabla 39 se prueba la intención para conocer lugares cercanos de pago al que se le envían mensajes que tengan esta intención y por consiguiente se muestra como resultado un listado con lugares de pago.

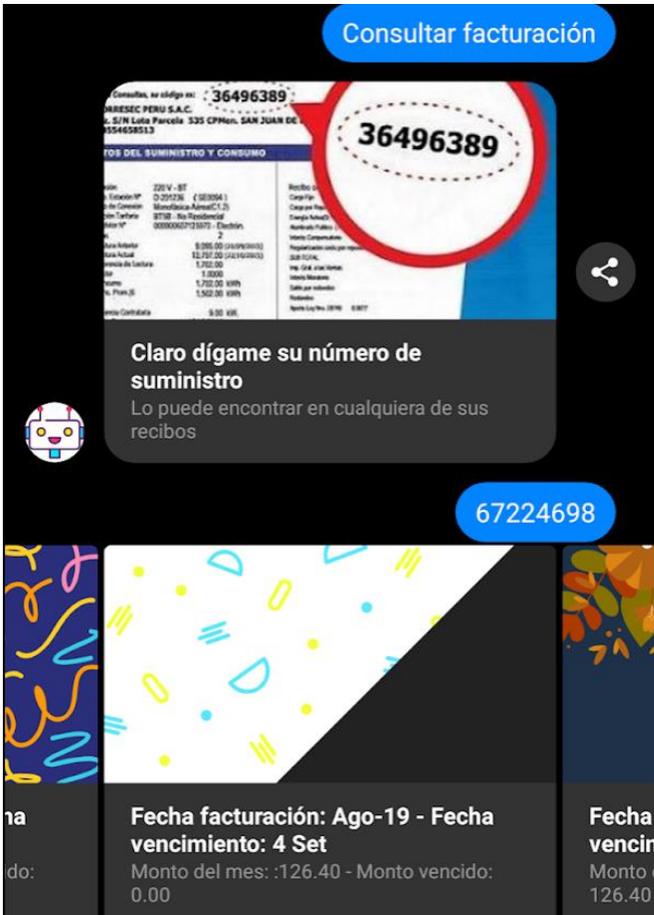
Tabla 39. Pruebas de Componentes 6

Objetivos de la Prueba	Comprobar intención de ubicación de lugares cercanos de pago
Técnicas	Realizar envío de mensajes relacionados a la ubicación de lugares de pago cercano
Código Involucrado	Funcionalidad desarrollada en Node JS, véase anexo
Caso de Prueba	<p>Formato de casos de pruebas</p> <p>Tipo de prueba: Unitaria (Caja Blanca) Objetivo: Validar que la pueda captar la intención con mensajes relacionados</p> <p>Caso No. 1</p> <p>Descripción: Mensajes variados con intención de ubicación de lugares de pago cercano</p> <p>Entradas: Dónde pago mi recibo, en qué lugares pago mi recibo</p> 
Resultado	El asistente detecto la intención con los mensajes enviados
Observaciones	

Fuente: Elaborado por el autor

Para comprobar la facturación de los últimos 4 meses el usuario ingresa su número de servicio y recibe mediante cartas sus facturaciones recientes, véase la Tabla 40.

Tabla 40. Pruebas de Componentes 7

Objetivos de la Prueba	Mostrar la facturación de los últimos 4 meses
Técnicas	Realizar envío de mensajes relacionados a la facturación de los últimos meses
Código Involucrado	Funcionalidad desarrollada en Node JS, véase anexo
Caso de Prueba	Formato de casos de pruebas
	Tipo de prueba: Unitaria (Caja Blanca) Objetivo: Validar que la pueda captar la intención con mensajes relacionados
	Caso No. 1
	Descripción: Mensajes variados con intención de facturación de los últimos meses
	Entradas: Dame mi facturación de los últimos meses, dime mi facturación, facturación
	
Resultado	El asistente detecto la intención con los mensajes enviados
Observaciones	

Fuente: Elaborado por el autor

5.2.2. Pruebas de aceptación

Las Pruebas de Aceptación fueron realizadas por parte del equipo de TI de Electrocentro a cargo de la Ingeniera Karina Laura León y el Ingeniero José Lucas Cornelio, al finalizar la realización de estas pruebas se firmó una carta de aceptación del proyecto que se encuentra detallada en el **Anexo 2** del documento. Además se realizó un testing por parte de los usuarios del asistente virtual a través de la plataforma de Facebook Messenger y los resultados se pueden apreciar en el ítem de **Resultados**.

5.3. RESULTADOS

El desarrollo del Asistente Virtual busca mejorar la atención que se brinda a sus clientes, por ende, luego de la finalización del proyecto y de su puesta en producción se realizaron encuestas de satisfacción dirigidas a los usuarios del Asistente Virtual ADA a través de Google Forms, para la siguiente encuesta se tomó como **población el total de usuarios que es 117 usuarios** que usan el Asistente Virtual por medio de Facebook Messenger, de los cuales escogeremos nuestra muestra mediante la fórmula de muestra de población finita:

$$n = \frac{N * Z^2 * p * q}{(N - 1) E^2 + Z^2 * p * q}$$

Se obtendrá la cantidad de usuarios a encuestar:

$$n = \frac{117 * (0.9)^2 * 0.5 * 0.5}{(117 - 1) * (0.05)^2 + (0.9)^2 * 0.5 * 0.5}$$

Luego obtenemos, $n = 48.11$ redondeado obtenemos que nuestra **muestra es de 49 usuarios**, de los cuales se obtuvieron los siguientes resultados a través de las encuestas:

6.3.1. DATOS GENERALES DE LOS ENCUESTADOS

- **Género de los Encuestados**

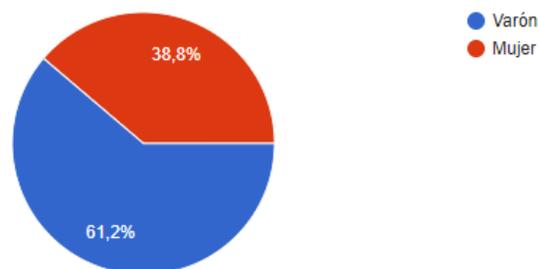


Figura 38. Género del participante
Fuente: Tomado de la encuesta de satisfacción realizada a través de Google Forms

De acuerdo a la encuesta realizada a usuarios del Asistente, se puede apreciar que un 61.2% son hombres y un 38.8% son mujeres, véase la Figura 38.

- **Edad de los Encuestados**

Como se puede apreciar en la Figura 39, la edad promedio de las personas que usan el Asistente Virtual están entre 23 y 63 años, además los usuarios que más utilizan la aplicación se encuentran entre los 29 y 34 años de edad.

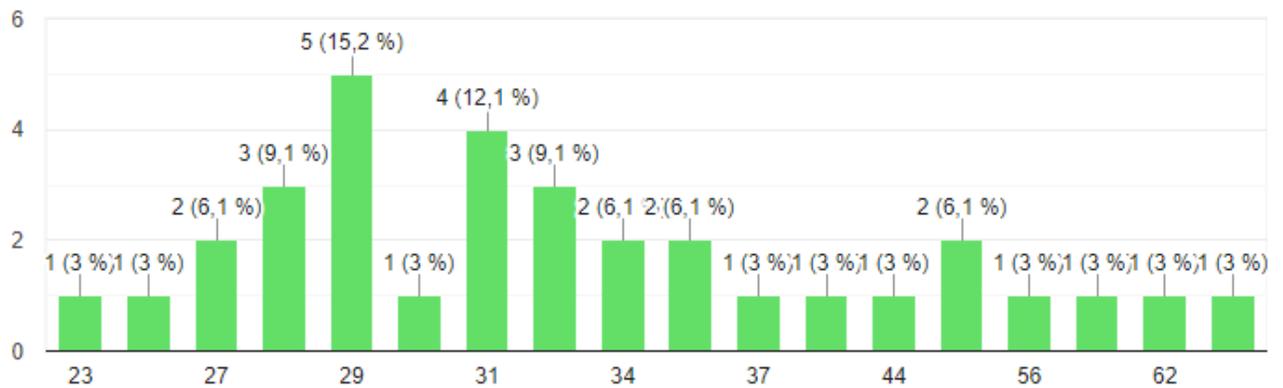


Figura 39. Edad promedio del participante
 Fuente: Tomado de la encuesta de satisfacción realizada a través de Google Forms

- **Lugar de Procedencia**

También se puede apreciar en la Figura 40 que mediante la encuesta que un 46.9% de los usuarios son de Huancayo, 34.7% son de El Tambo y un 18.4% son de Chilca.

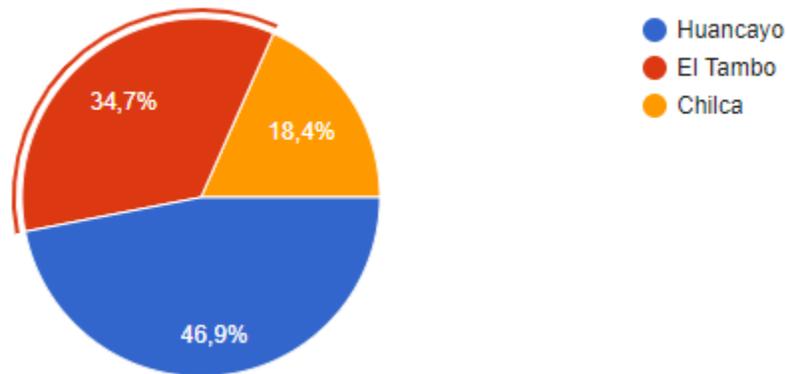


Figura 40. Lugar de procedencia
Fuente: Tomado de la encuesta de satisfacción realizada a través de Google Forms

- **Frecuencia de Uso del Asistente Virtual**

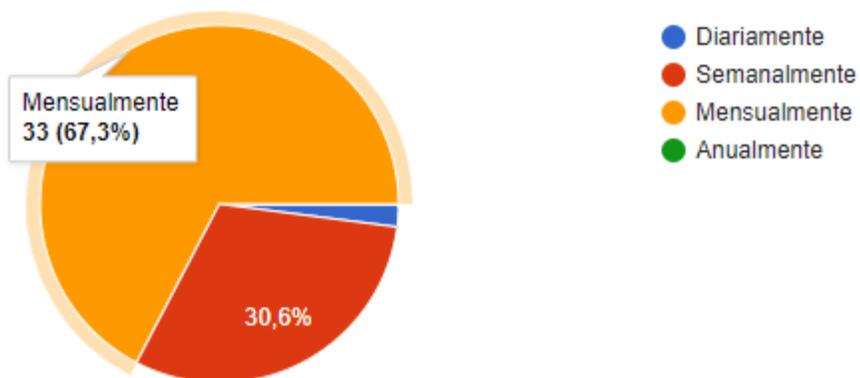


Figura 41. Frecuencia de uso del asistente
Fuente: Tomado de la pregunta ¿Con qué frecuencia utiliza al Asistente Virtual ADA de Electrocentro? – Encuesta realizada a través de Google Forms

Otra observación obtenida de la encuesta es que un 67.3% de los encuestados utilizan la aplicación mensualmente tal y como se muestra en la Figura 41.

- **Funcionalidades más Usadas**

La encuesta refleja que las funcionalidades más usadas del Asistente con un 93.9% de consultar sus recibos, un 51% para la consulta de lugares cercanos de pago y un 46.9% para la consulta de facturación de los últimos meses, véase la Figura 42.

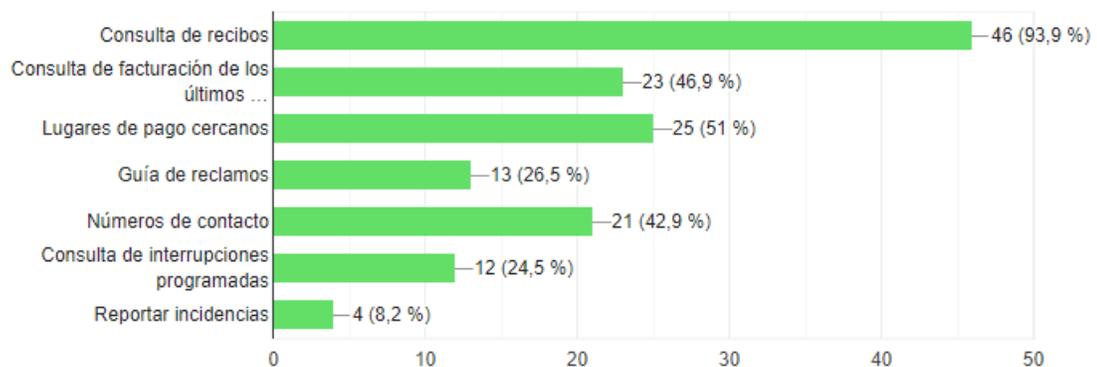


Figura 42. Funcionalidades más usadas

Fuente: Tomado de la pregunta ¿Con qué fin utiliza el Asistente Virtual Ada de Electrocentro? – Encuesta realizada a través de Google Forms

6.3.2. RESULTADOS OBTENIDOS PRUEBAS DE ACEPTACIÓN

A continuación, se presentan los resultados obtenidos a través de las encuestas clasificados por cada uno de los objetivos planteados en la tesis

6.3.2.1. Implementación del Servicio que realice consultas al API de Electrocentro

En el Anexo 2 como parte del Acta de Aceptación del proyecto por parte de la empresa Electrocentro y la Ing. Karina Laura León, Jefa del Área de Tecnologías de la Información, se tiene como COMPLETADO

todos los requerimientos solicitados por Electrocentro, entre ellos se tiene el desarrollo de funciones serverless, las cuales son las que realizan consultas al API de la empresa para así poder los datos de cada uno de los clientes. Estas funciones son completamente funcionales son desplegadas en un servicio de Netlify que permite su alojamiento y funcionan las 24 horas del día, a la par con los servicios de Electrocentro.

6.3.2.2. Implementación de Consulta de Último Recibo

Se puede apreciar en la Figura 43 que un 65.3% de los encuestados respondió con un “muy de acuerdo” acerca de la utilidad que significa obtener la información que el asistente virtual les brinda acerca de sus recibos emitidos por la empresa cada mes a través de la plataforma de Facebook Messenger.

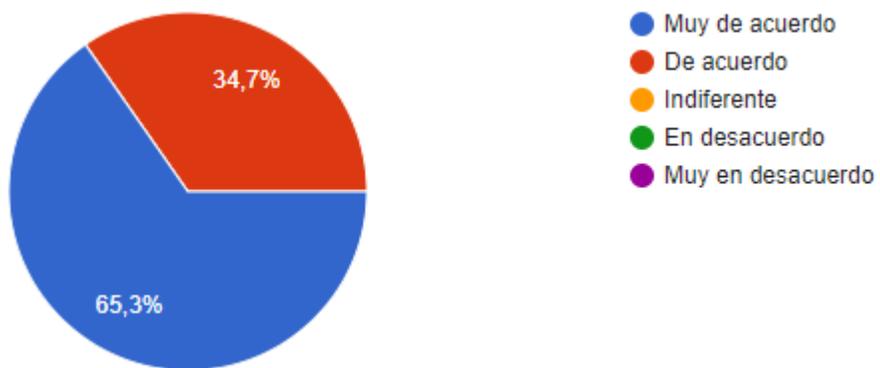


Figura 43. Información brindada acerca de los recibos de los clientes
Fuente: Tomado de pregunta ¿Considera que el asistente ADA le ayuda conocer los detalles de su recibo? – Encuesta realizada a través de Google Forms

6.3.2.3. Implementación de Consulta de Facturación de Últimos 4 Meses

De la Figura 44 se puede apreciar que un 66.7% de los usuarios está “de acuerdo” con la información que el asistente brinda acerca de la facturación de su servicio de los últimos 4 meses y cabe resaltar que un 31.3% se encuentra “muy de acuerdo” con la utilidad que esta funcionalidad brinda, lo cual nos da a entender que el servicio brindado por el asistente es muy valorado por los clientes.

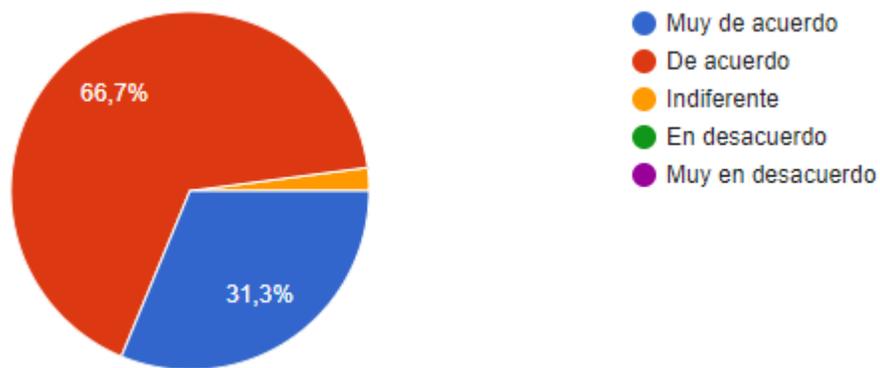


Figura 44. Información brindada de la Facturación de los últimos 4 meses
Fuente: Tomado de pregunta ¿Considera que el asistente ADA le ayuda a conocer su facturación de los últimos meses? – Encuesta realizada a través de Google Forms

6.3.2.4. Implementación de Búsqueda de Lugares Cercanos

Otro dato importante de la encuesta según la Figura 45 es que un 53.1% de los usuarios está de “muy acuerdo” con que el asistente les ayuda a conocer lugares de pago cercanos a su ubicación actual además de un 44.9% que está “de acuerdo” con esta información, cabe resaltar que de acuerdo a esta gráfica gran mayoría de los encuestados valoran muy positivamente esta funcionalidad.

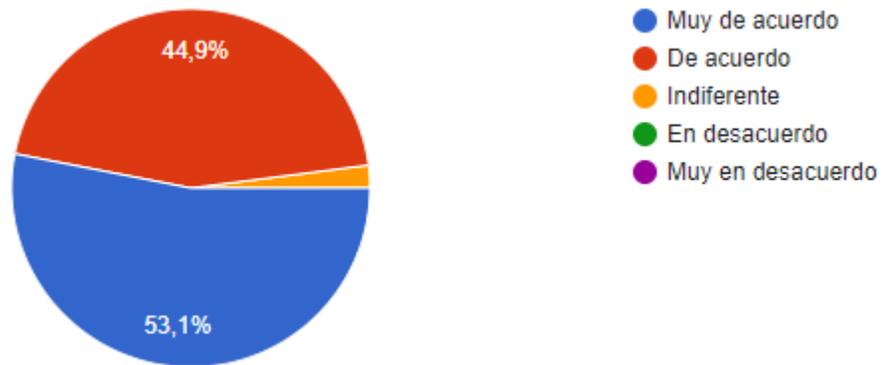


Figura 45. Información brindada acerca de los lugares de pago cercanos
Fuente: Tomado de pregunta ¿Considera que el asistente ADA le ayuda a encontrar lugares de pago cercanos? – Encuesta realizada a través de Google Forms

6.3.2.5. Implementación de Consulta de Pasos Para Reclamos y Números de Contacto de la Empresa

Además, en la Figura 46, se observa que el 77.6% de los usuarios del asistente virtual se encuentra “de acuerdo” con la utilidad y la información que se les brinda a través de Messenger acerca de los pasos que se deben de tener en cuenta para realizar sus reclamos.

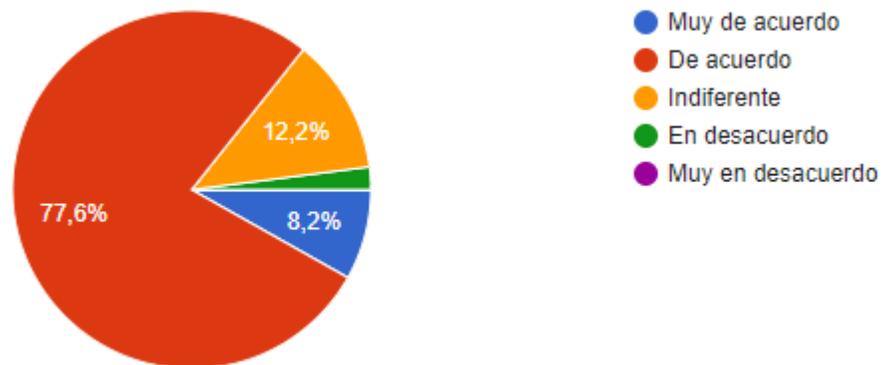


Figura 46. Información de pasos para los reclamos

Fuente: Tomado de la pregunta ¿Considera que el asistente ADA le ayuda a conocer los pasos para realizar sus reclamos? – Encuesta realizada a través de Google Forms

Otro dato importante de la encuesta es que un 75.5% de los usuarios está “de acuerdo” con la utilidad que el asistente le brinda acerca de los números de contacto de la empresa, mientras que un 20.4% está “muy de acuerdo” con esta información, véase la Figura 47.

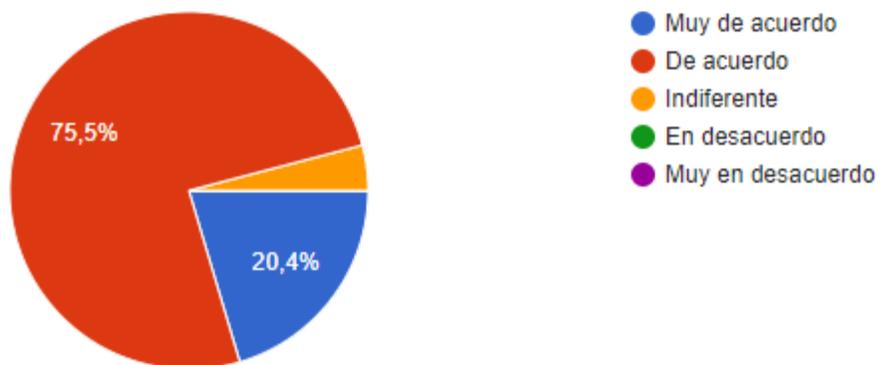


Figura 47. Información de números de contacto

Fuente: Tomado de la pregunta ¿Considera que el asistente ADA le ayuda a conocer los números de contacto de Electrocentro? – Encuesta realizada a través de Google Forms

6.3.2.6. Implementación de Consulta de Interrupción de Servicio

De la encuesta, en la Figura 48, se puede apreciar que un 85.4% de los usuarios está “de acuerdo” con la utilidad que brinda el asistente acerca de las interrupciones de servicio programadas por la empresa, mientras que un 14.6% se encuentra “muy de acuerdo” con esta información, no se puede apreciar que los usuarios estén “en desacuerdo” con esta funcionalidad.

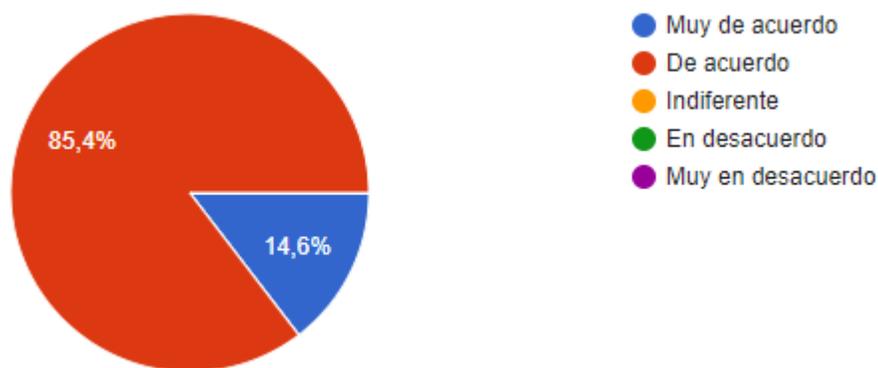


Figura 48. Interrupciones de servicio

Fuente: Tomado de la pregunta ¿Considera que el asistente ADA le ayuda a conocer las interrupciones de servicio programadas?

6.3.2.7. Implementación de Reporte de Incidencias

Además, un 89.8% de los usuarios encuestados se encuentra “de acuerdo” con el servicio de reporte de incidencias que funciona dentro del asistente virtual, véase la Figura 49.

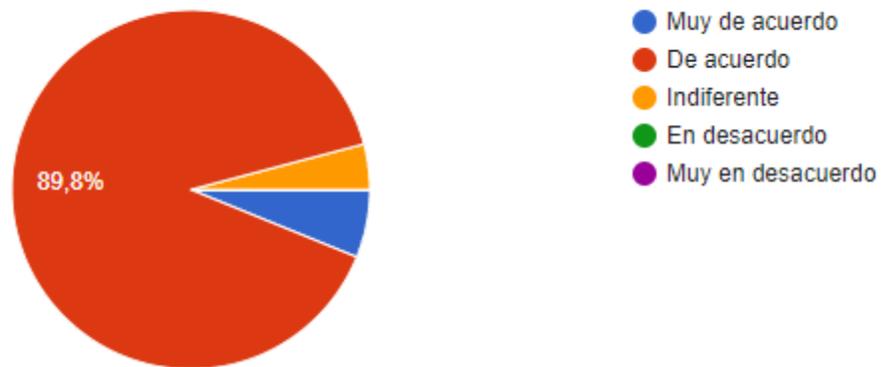


Figura 49. Reportes de incidencias

Fuente: Tomado de la pregunta ¿Considera que el asistente ADA le ayuda a reportar incidencias con su servicio? – Encuesta realizada a través de Google Forms

Por otro lado, en la Figura 50 de la encuesta se puede observar el canal o medio por el que los usuarios llegaron a conocer el Asistente Virtual de ello se obtiene que un 71.4% de los usuarios llegaron a través de la misma página de Facebook del Asistente, un 36.7% de los usuarios llegaron a través de la recomendación de amigos y familiares y un 16.3% mediante banners informativos.

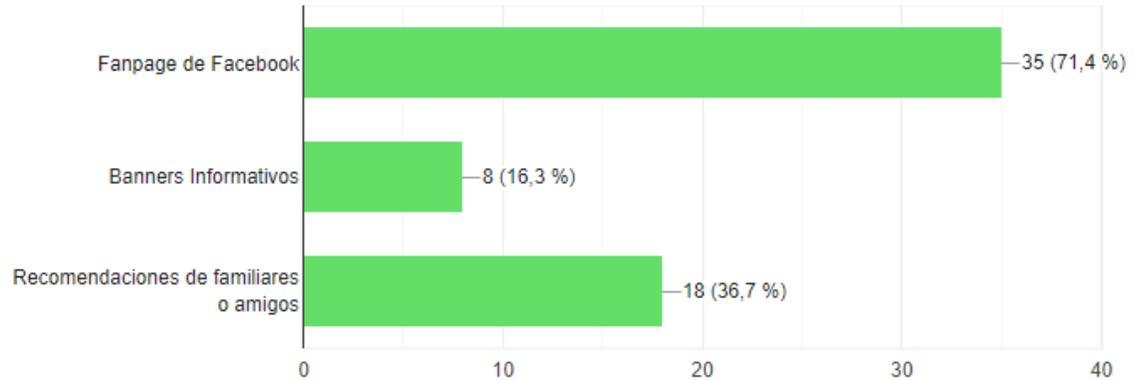


Figura 50. Canal de referencia

Fuente: Tomado de la pregunta ¿Cómo descubriste el Asistente Virtual ADA de Electrocentro? – Encuesta realizada a través de Google Forms

6.3.3. RESULTADOS OBTENIDOS POR PRUEBAS DE FIABILIDAD

A continuación, se presentan los resultados obtenidos a través de las Pruebas de Fiabilidad realizadas con ArtilleryJS la cual es una librería moderna y fácil de usar para este propósito, está desarrollada en NodeJS, permite hacer pruebas de carga simulando varias solicitudes simultaneas al servidor. En este proyecto se realizaron múltiples pruebas clasificadas por cada uno de los objetivos planteados detallados a continuación.

6.3.3.1. Implementación del Servicio que realice consultas al API de Electrocentro

Se observa en la Figura 51, los tiempos de respuesta obtenidos luego de la prueba de carga, de ello se obtiene como tiempo mínimo 430.2 milisegundos, como tiempo máximo de respuesta unos 15566.1 milisegundos y una mediana de 454.4 milisegundos.

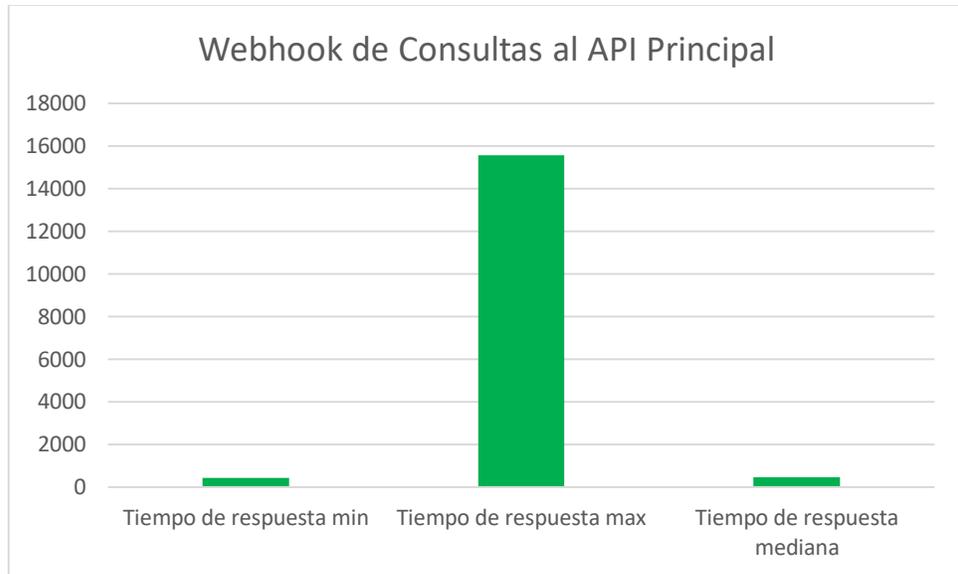


Figura 51. Test de fiabilidad webhook de consultas al API Principal
Fuente: Elaborado por el autor

Además, al realizar las pruebas de fiabilidad sobre este endpoint se puede ver que de 1200 llamadas 800 tuvieron una respuesta positiva y por otro lado se obtuvo unos 400 errores de estas llamadas, es decir el 66.7% fueron atendidas correctamente, véase la Tabla 41.

Tabla 41. Prueba de fiabilidad webhook de consultas al API Principal

Todos los usuarios virtuales finalizados
Informe resumido @ 23:36:11(-0500) 2020-04-14
Escenarios lanzados: 1200
Escenarios completados: 800
Solicitudes completadas: 800
Respuesta media/seg: 16
Tiempo de respuesta (mseg):
mínimo: 430.2 máximo: 15566.1 mediana: 454.4 latencia p95: 15451.5 latencia p99: 15466.6
Cantidad de escenarios:

0: 1200 (100%)
Códigos:
200: 800
Errores:
ETIMEDOUT: 400

Fuente: Elaborado por el autor

6.3.3.2. Implementación de Consulta de Último Recibo

En la prueba de fiabilidad de la funcionalidad de consulta del último recibo del cliente se obtiene como tiempos de respuesta: mínimo de 667.8, máximo de 10728 y mediana de 723.8 milisegundos, véase la Figura 52.

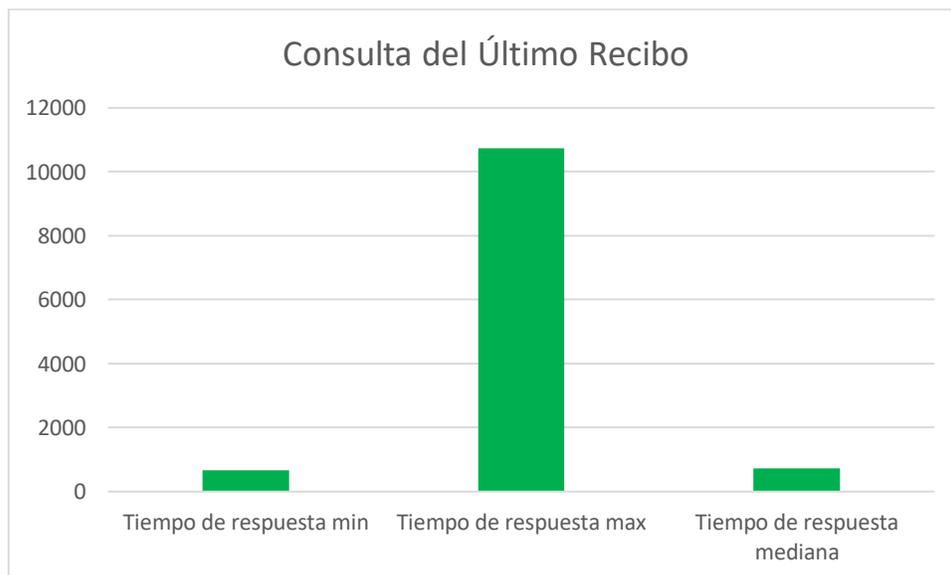


Figura 52. Test de Fiabilidad Api Consulta del último recibo

Fuente: Elaborado por el autor

Al realizar la prueba de fiabilidad se puede apreciar que de las 300 llamadas al endpoint 298 son aceptadas como satisfactorias, es decir el 99.3%, mientras que sólo 2 fueron erróneas, véase la Tabla 42.

Tabla 42. Test de fiabilidad consulta del último recibo

Todos los usuarios virtuales finalizados
Informe resumido @ 00:48:03(-0500) 2020-04-15
Escenarios lanzados: 300
Escenarios completados: 300
Solicitudes completadas: 300
Respuesta media/seg: 9.56
Tiempo de respuesta (mseg):
mínimo: 667.8 máximo: 10728.3 mediana: 723.8 latencia p95: 1280 latencia p99: 3817.4
Cantidad de escenarios:
0: 300 (100%)
Códigos:
200: 298
502: 2

Fuente: Elaboración propia

6.3.3.3. Implementación de Consulta de Facturación de Últimos 4 Meses

En la Figura 53, se observa que de la carga aplicada al servidor se tiene una mínima de 655, máxima de 4835.9 y mediana de 768.1 milisegundos.

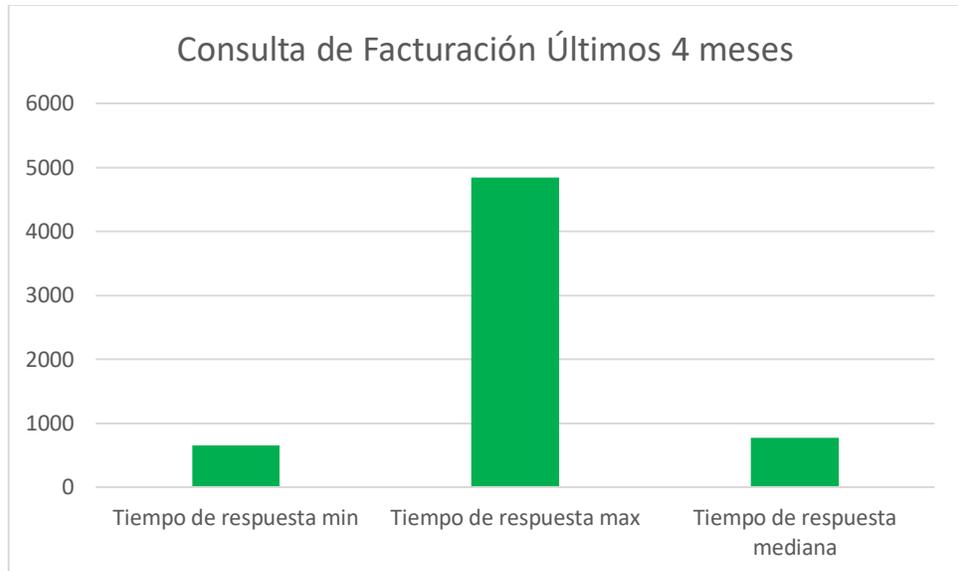


Figura 53. Prueba de fiabilidad api consulta facturación últimos 4 meses
Fuente: Elaborado por el autor

En la Tabla 43, se tiene como resultado de las pruebas de fiabilidad que de las 300 peticiones realizadas el 100% fueron satisfactorias.

Tabla 43. Prueba de fiabilidad api consulta facturación de los últimos 4 meses

Todos los usuarios virtuales finalizados
Informe resumido @ 00:52:07(-0500) 2020-04-15
Escenarios lanzados: 300
Escenarios completados: 300
Solicitudes completadas: 300
Respuesta media/seg: 9.56
Tiempo de respuesta (mseg):
min: 655
max: 4835.9
mediana: 768.1
latencia p95: 1626.4
latencia p99: 3691.9
Cantidad de escenarios:
0: 300 (100%)
Códigos:
200: 300

Fuente: Elaborado por el autor

6.3.3.4. Implementación de Búsqueda de Lugares Cercanos

De la prueba de fiabilidad realizada se obtiene como tiempo mínimo de respuesta por parte del endpoint 546.8 milisegundos, como tiempo máximo 5433.5 milisegundos y una mediana de 1012.8 milisegundos, véase la Figura 54.

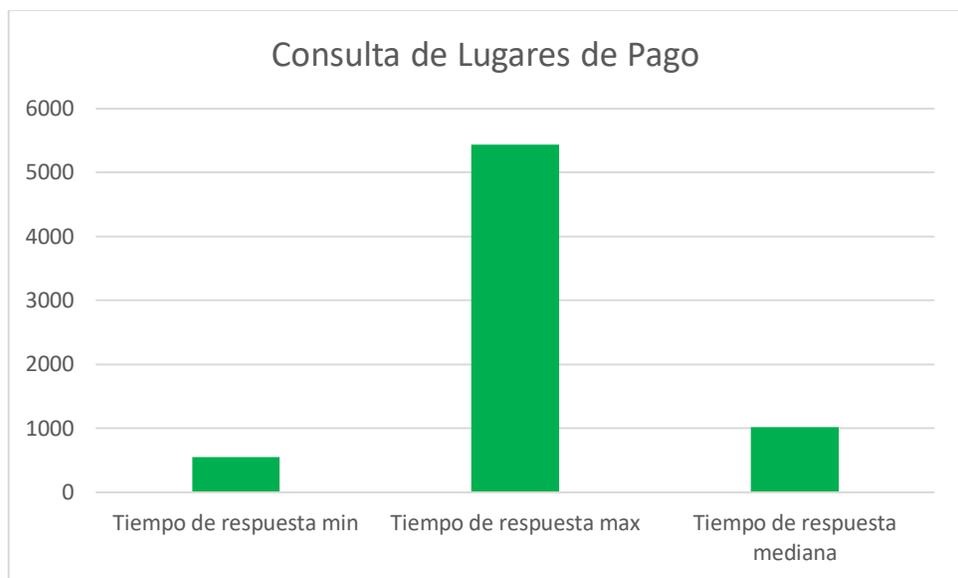


Figura 54. Prueba de fiabilidad api consulta lugares de pago cercanos
Fuente: Elaborado por el autor

Además, como datos adicionales se obtiene de las 300 solicitudes realizadas al endpoint el 100% fueron satisfactorias, asegurando así su fiabilidad, véase la Tabla 44.

Tabla 44. Prueba de fiabilidad api consulta lugares de pago cercanos

Todos los usuarios virtuales finalizados
Informe resumido @ 00:21:55(-0500) 2020-04-15
Escenarios lanzados: 300
Escenarios completados: 300
Solicitudes completadas: 300
Respuesta media/seg: 9.72
Tiempo de respuesta (mseg):
mínimo: 546.8 máximo: 5433.5 mediana: 1012.8 Latencia p95: 3514.5 Latencia p99: 4150
Cantidad de escenarios:
0: 300 (100%)
Códigos:
200: 300

Fuente: Elaborado por el autor

6.3.3.5. Implementación de Consulta de Pasos Para Reclamos y Números de Contacto de la Empresa

En la Figura 55 se detalla la medición de los tiempos de respuesta obtenidos al realizar la carga al servidor, de ello se observa que el tiempo mínimo de respuesta es de 822.5 milisegundos, el tiempo máximo es de 2852.2 milisegundos y la mediana es de 1320.3 milisegundos.

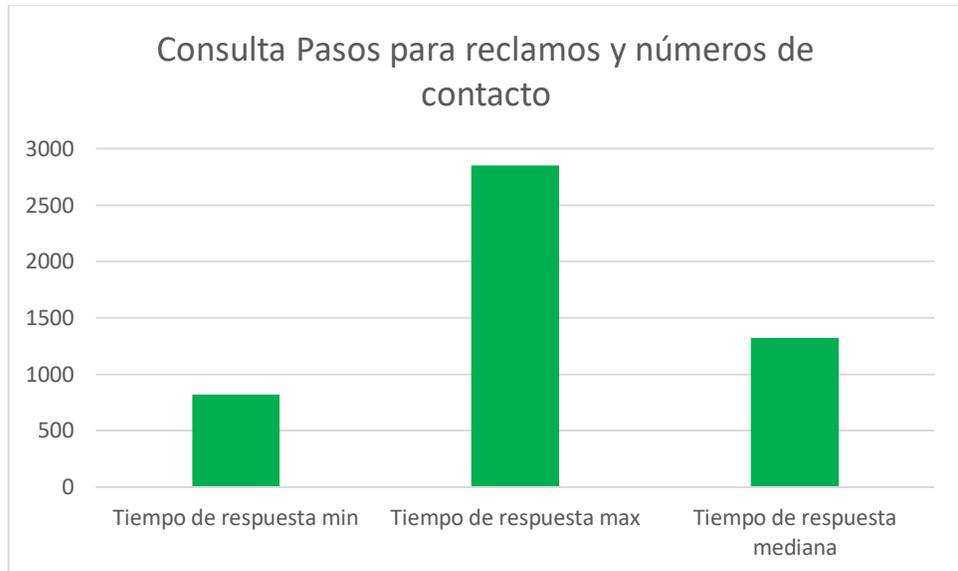


Figura 55. Prueba de fiabilidad api consulta pasos para reclamos y números de contacto

Fuente: Elaborado por el autor

Además, al observar la Tabla 45, que muestra los resultados completos de la prueba de fiabilidad se aprecia que de las 300 solicitudes enviadas al servidor el 100% fueron satisfactorias.

Tabla 45. Prueba de fiabilidad api consulta pasos para reclamos y números de contacto

Todos los usuarios virtuales finalizados
Informe resumido @ 01:15:54(-0500) 2020-04-15
Escenarios lanzados: 300
Escenarios completados: 300
Solicitudes completadas: 300
Respuesta media/seg: 9.56
Tiempo de respuesta (mseg):
mínimo: 822.5
máximo: 2852.2
mediana: 1320.3
Latencia p95: 1758.4
Latencia p99: 2210.6
Cantidad de escenarios:
0: 300 (100%)
Códigos:
200: 300

Fuente: Elaborado por el autor

6.3.3.6. Implementación de Consulta de Interrupción de Servicio

Al realizar las pruebas de fiabilidad para la consulta de interrupción de servicio, se observa en la medición del tiempo de respuesta del servidor lo siguiente: el tiempo de respuesta mínimo es de 670.2 milisegundos, el tiempo de respuesta máxima es 3911.1 milisegundos y la mediana es 782.7 milisegundos, véase la Figura 56.

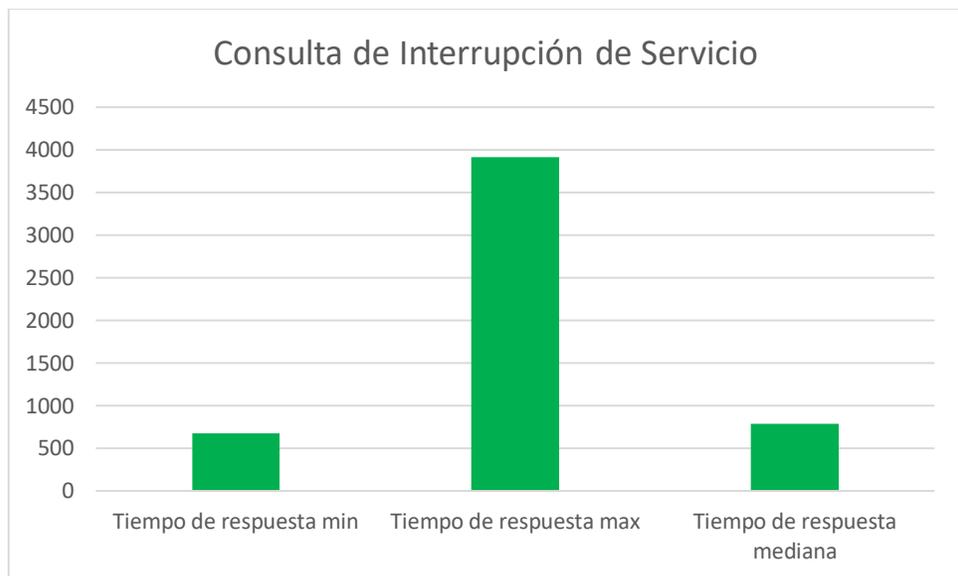


Figura 56. Prueba de fiabilidad api consulta interrupción de servicio
Fuente: Elaborado por el autor

También de las pruebas realizadas se obtiene que de las 300 llamadas realizadas al servidor el 100% tuvo una respuesta satisfactoria lo cual refleja la fiabilidad del servicio, véase la Tabla 46.

Tabla 46. Prueba de fiabilidad api consulta interrupción de servicio

Todos los usuarios virtuales finalizados
Informe resumido @ 00:54:55(-0500) 2020-04-15
Escenarios lanzados: 300
Escenarios completados: 300
Solicitudes completadas: 300
Respuesta media/seg: 9.4
Tiempo de respuesta (mseg):
mínimo: 670.2 máximo: 3911.1 mediana: 782.7 Latencia p95: 1470.8 Latencia p99: 2920.4
Cantidad de escenarios:
0: 300 (100%)
Códigos:
200: 300

Fuente: Elaborado por el autor

6.3.3.7. Implementación de Reporte de Incidencias

En la Figura 57, se observa del resultado obtenido de las pruebas de fiabilidad que el tiempo de respuesta mínima del servidor es 676.5 milisegundos, el tiempo máximo de respuesta es 7462.7 milisegundos y la mediana es 779.8 milisegundos.

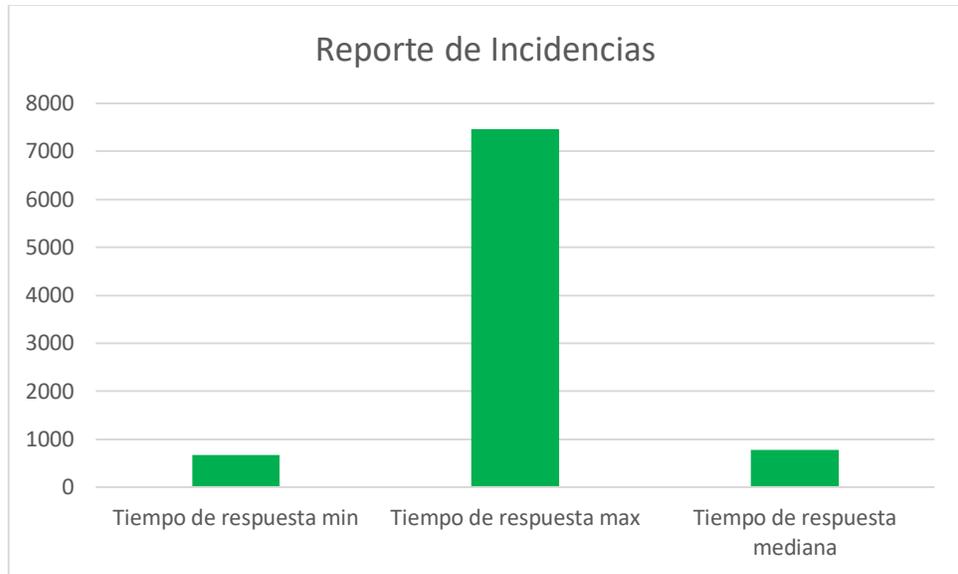


Figura 57. Prueba de fiabilidad api reporte de incidencias
Fuente: Elaboración Propia

Además, se aprecia que de las 75 solicitudes enviadas para la persistencia de datos el 100% fue satisfactoria, es decir no se perdieron datos que deberían ser guardados al reportar incidencias, véase la Tabla 47.

Tabla 47. Prueba de fiabilidad api reporte de incidencias

Todos los usuarios virtuales finalizados
Informe resumido @ 11:37:11(-0500) 2020-04-15
Escenarios lanzados: 75
Escenarios completados: 75
Solicitudes completadas: 75
Respuesta media/seg: 4.75
Tiempo de respuesta (mseg):

mínimo: 676.5
máximo: 7462.7
mediana: 779.8
Latencia p95: 6289.6
Latencia p99: 7306.6
Cantidad de escenarios:
0: 75 (100%)
Códigos:
200: 75

Fuente: Elaboración Propia

6.3.4. RESULTADOS FINALES

De la encuesta en la Figura 58 se obtiene que un 77.6% de los usuarios están “muy de acuerdo” con la utilidad del Asistente Virtual, además un 22.4% está “de acuerdo” con su utilidad, de tal manera, se considera que el desarrollo del Asistente Virtual como canal extra para la Atención al Cliente por parte de la empresa fue exitosa

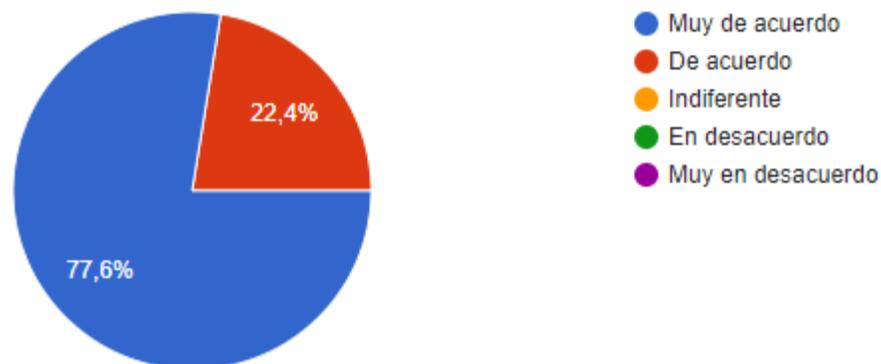


Figura 58. Valoración de la utilidad del asistente

Fuente: Tomado de la pregunta ¿Considera que el Asistente Ada le ayuda a conocer los detalles de su recibo? – Encuesta realizada a través de Google Forms

Además, en la Figura 59 se observa la comparativa de las pruebas de fiabilidad sobre cada uno de los objetivos planteados y sus tiempos de respuesta, todos ellos detallados en los puntos del módulo anterior.

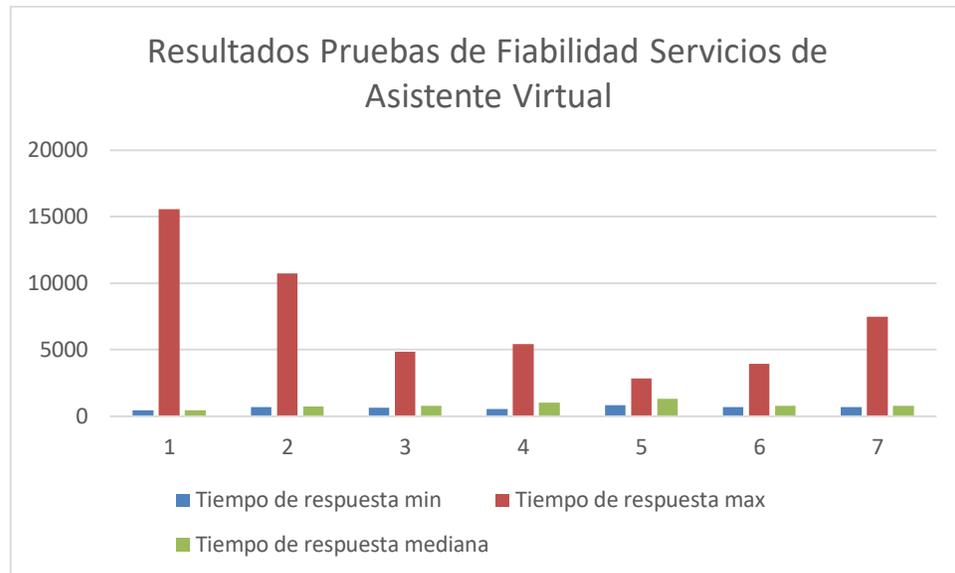


Figura 59. Resultados Pruebas de Fiabilidad Servicios del Asistente virtual
Fuente: Elaborado por el autor

1. Servicio que realiza consultas REST al API de Electrocentro
2. Consulta de último recibo en el asistente virtual
3. Consulta la facturación de los últimos 4 meses en el asistente virtual
4. Búsqueda de lugares cercanos de pago en el asistente virtual
5. Consulta información sobre los pasos para realizar reclamos y los números de contacto
6. Consulta de interrupciones del servicio eléctrico
7. Reporte de incidencias del servicio eléctrico

Finalmente, de todas las solicitudes efectuadas al servidor en cada una de las pruebas de fiabilidad realizadas se obtiene que 1873 solicitudes fueron exitosas es decir el 82.3% del total, sólo 2 solicitudes resultaron en error y

400 no se completaron al obtener un timeout como respuesta, tan solo el 17.6% del total de solicitudes, véase la Figura 60.

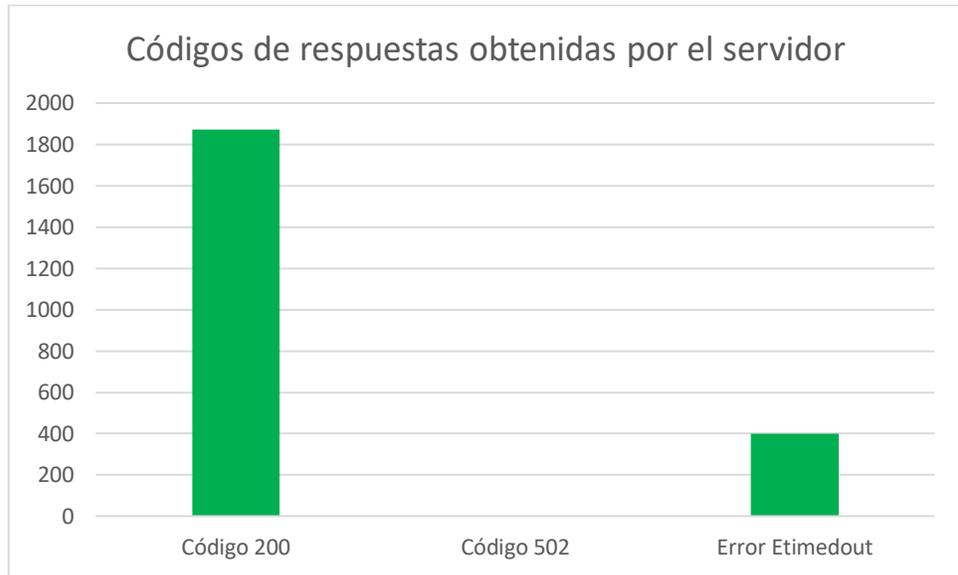


Figura 60. Resultados pruebas de fiabilidad códigos de respuesta obtenidos
Fuente: Elaborado por el autor

CONCLUSIONES

- a) Mediante el uso de una arquitectura serverless se implementó un servicio basado en funciones que realiza consulta al API desarrollada por Electrocentro y obtiene los datos de los clientes basados en su número de suministro, este servicio se encuentra desarrollado con un lenguaje muy potente como lo es Node JS que trabaja muy bien sobre una arquitectura serverless y desplegado sobre la plataforma Netlify a la que se le realizó pruebas de fiabilidad donde se aprecia que el 66.7% de las peticiones realizadas fueron procesadas correctamente con un tiempo de respuesta mínimo de 430.2 milisegundos y un máximo de 15566.1 milisegundos, además de proveer un despliegue continuo que permitió tener cualquier cambio en la aplicaciones prácticamente inmediatamente.

- b) Se desarrolló una intención que permite detectar cuando un cliente desea revisar los datos de su último recibo además de una función serverless asociada a esta intención que se encarga de realizar consultas al API dado el número de suministro otorgado por el cliente, así mismo se dio un formato a los datos recibidos debido a que se encuentran de forma muy desorganizada y se tiene que tener el formato correcto al momento de ser enviados al cliente a través de Messenger, de las pruebas de fiabilidad se observó que el 99.3% de las solicitudes enviadas al servidor fueron respondidas correctamente con un tiempo de respuesta mínimo de 667.8 milisegundos y un máximo de 10728.3 milisegundos, por otro lado según la encuesta realizada es la funcionalidad más usada, además 65.3% de los usuarios respondió “muy de acuerdo” cuando se les consultó acerca de su utilidad.

- c) Se implementó del mismo modo una intención específica para la consulta de la facturación de los últimos 4 meses del cliente dentro del agente de Dialogflow que luego de recibir el número de suministro del cliente realiza la consulta a su respectiva función para así realizar el request al API de la empresa y obtener los

datos requeridos que también tienen que ser formateados debido a estar muy desorganizados, por consiguiente al efectuar las pruebas de fiabilidad se observó que el 100% de las solicitudes efectuadas fueron respondidas satisfactoriamente con un tiempo de respuesta mínimo de 655 milisegundos y un máximo de 4835.9 milisegundos, además se utilizó el API de Messenger que ofrece widgets que hacen que la presentación de estos datos sea más agradable. El 66.7% de usuarios calificaron con “muy de acuerdo” cuando se les consultó acerca de su utilidad.

- d) Se implementó en su totalidad el servicio para la consulta de lugares cercanos de pago con la ayuda del servicio de localización implementado dentro la aplicación de Facebook Messenger la cual se encarga de obtener la ubicación en tiempo real del cliente y que luego es enviada al servicio desarrollado y de esta manera realiza la respectiva consulta al API de Google Places que permite encontrar lugares determinados basados en las coordenadas enviadas en el request al hacer pruebas de fiabilidad a este servicio se observó que el 100% de las solicitudes enviadas fueron respondidas correctamente con un tiempo de respuesta mínimo de 546.8 milisegundos y un máximo de 5433.5 milisegundos, además la encuesta refleja que el 53.1% de los usuarios calificó con “muy de acuerdo” cuando se les consultó acerca de la utilidad de esta funcionalidad.

- e) Se desarrolló un servicio que realiza webscrapping sobre la página de Electrocentro que es la que contiene los pasos para la realización de reclamos además de los números de contacto de la empresa y que si en algún momento se realizase algún cambio se puedan obtener los datos actualizados en lugar de escribirlos directamente en código ya que de esa forma resultaría inconveniente debido a que se tendría que liberar una nueva versión cada vez que se realice algún cambio en los números de contacto o pasos para los reclamos, además se practicó pruebas de fiabilidad sobre este servicio para medir su capacidad de carga y se obtuvo que el 100% de las solicitudes enviadas fueron procesadas satisfactoriamente se observó también que el tiempo de respuesta mínimo fue de 822.5 milisegundos y el máximo

de 2852.2 milisegundos. Por otro lado, el 77.6% de los usuarios encuestados calificó como “muy de acuerdo” cuando se les consultó acerca de la utilidad de esta de esta parte funcional de la aplicación.

- f) Se desarrolló un servicio serverless que realiza consultas al API de la empresa para poder obtener las interrupciones del servicio eléctrico luego de enviar el número de suministro del cliente, se da un formato al resultado obtenido para que puede ser utilizado por la aplicación de Facebook Messenger y de esta manera ser mostrado al usuario, a esta funcionalidad se le realizó pruebas de fiabilidad obteniéndose que el 100% de las consultas fueron respondidas satisfactoriamente con un tiempo de respuesta mínimo de 670.2 y un máximo de 3911.1 milisegundos, además la funcionalidad resultó muy útil debido que por diseño también muestra si habrá corte del servicio por falta de pago, por otro lado el 85.4% de usuarios calificó con “muy de acuerdo” acerca de la utilidad de esta funcionalidad.

- g) Se desarrolló una intención que detecte cuando el cliente desea reportar una incidencia del servicio que Electrocentro provee además de 3 intenciones consecutivas ligadas a esta, las cuales permiten recolectar los datos del cliente tales como su nombre, DNI, ubicación y descripción del problema los cuales son enviados a un servicio serverless que se encarga de procesar estos datos y almacenarlos en una base de datos no relacional provista por Firebase la cual puede ser consultada posteriormente por el personal de la empresa para su posterior solución, brindando así una mejor atención a sus clientes, a esta funcionalidad se le realizó pruebas de fiabilidad y se obtuvo que el 100% de las solicitudes fueron respondidas correctamente además se observó que el tiempo de respuesta mínimo fue de 676.5 milisegundos y máximo de 7462.7 milisegundos, además cuando se encuestó a los usuarios por esta funcionalidad un 89.8% respondió con “muy de acuerdo” acerca de su utilidad.

TRABAJOS FUTUROS

- a) En cuanto el número de usuarios crezca y se sobrepase el límite del plan gratuito de Netlify y Dialogflow la empresa tendrá que moverse al plan pagado que mejor le convenga que generalmente no incrementa demasiado sus costos ya que estos varían de acuerdo a la cantidad de consultas hechas y no un pago fijo como se tendría si la aplicación estuviera alojada en otro proveedor de cloud, además la forma en que se desarrolló permite agregar funcionalidad fácilmente, por otro lado se espera mejorar la fiabilidad del servidor evitando respuestas de fuera de tiempo que fueron de un 32.3%.

- b) Se recomienda mejorar la forma en la que se encuentra implementada el API que responde las consultas del último recibo del cliente de Electrocentro que seguramente está desarrollada para alimentar otros servicios al mismo tiempo lo que hace que en la forma en la que se encuentra implementada en la actualidad dificulta el desarrollo del Asistente Virtual.

- c) Se sugiere mejorar la cantidad de meses de facturación mostrados al cliente, la cual se encuentra limitada actualmente por el propio API de Facebook Messenger en el que sólo se puede mostrar hasta 4 cartas al mismo tiempo lo que impide mostrar más meses, en futuras versiones se pueden implementar soluciones que no se limiten por este tipo de inconvenientes.

- d) Se recomienda mejorar la búsqueda de lugares cercanos agregando más opciones de entidades bancarias autorizadas y además cambiar algunas partes de la implementación debido a que Facebook ha decidido cambiar la forma en la que se trabaja con la ubicación en tiempo real de los usuarios.

- e) Se sugiere implementar un endpoint específico dentro del API de Electrocentro que al ser consultado retorne los pasos para la realización de reclamos además de los

números de contacto de la empresa, para permitir que el Asistente logre responder rápidamente.

- f) Se recomienda implementar un servicio que anuncie a todos los usuarios del Asistente Virtual cuando se vaya a realizar una interrupción de servicio programada el cuál sería más practico que el actualmente implementado, el cuál es limitado por la plataforma de Messenger, que ha limitado el envío masivo de mensajes por parte de cualquier asistente virtual.

- g) Se sugiere mejorar el cliente web que permite visualizar las incidencias reportadas por los usuarios debido a que inicialmente se tiene un desarrollo básico, esta mejora permitirá a la empresa tener data estadística sobre cuáles son los problemas recurrentes más reportados por los clientes, además de su ubicación y de esta manera mejorar su servicio.

REFERENCIAS BIBLIOGRÁFICAS

1. MIKKONEN, Tommi. Building a Serverless Messenger Chatbot. [en línea]. 2012. Vol. 7059, p. 75-86. DOI 10.1007/978-3-642-27997-3. Disponible en:
<http://link.springer.com/10.1007/978-3-642-27997-3>
2. Banco de Crédito lanza “Arturito BCP”, chat cognitivo para Facebook Messenger | Empresas | Gestión. [en línea]. [Fecha de Consulta: 14 octubre 2019]. Disponible en:
<https://gestion.pe/economia/empresas/banco-credito-lanza-arturito-bcp-chat-cognitivo-facebook-messenger-149468-noticia/>
3. Revisa tus notas y más con CONTIbot, nuestro asistente virtual. [en línea]. [Fecha de Consulta: 14 octubre 2019]. Disponible en:
<https://estudiantes.ucontinental.edu.pe/comunicados/revisa-tus-notas-y-mas-con-contibot-nuestro-asistente-virtual-pruebalo-ya/>
4. Sunat espera reducir en 20% las colas en ventanillas con nuevo chatbot | Economía | Gestión. [en línea]. [Fecha de Consulta: 14 octubre 2019]. Disponible en:
<https://gestion.pe/economia/sunat-espera-reducir-20-colas-ventanillas-nuevo-chatbot-234705-noticia/>
5. Papa John’s Pizza - Publicaciones. [en línea]. [Fecha de Consulta: 14 octubre 2019]. Disponible en:
<https://www.facebook.com/papajohnsperu/photos/a.286099768123031/2006490459417278/?type=1&theater>
6. INSTITUCIONAL, Marco. PROYECTO PLAN ESTRATÉGICO INSTITUCIONAL DE ELECTROCENTRO 2017 – 2021 Huancayo, junio 2019.
7. Electrocentro - Memoria 2018 - Iluminamos la vida de más de 4 millones de peruanos. [sin fecha].
8. DALTON, Jeffrey, AJAYI, Victor y MAIN, Richard. Vote Goat: Conversational Movie Recommendation. [SIGIR2018] The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018. P. 1285-1288.
9. VASCONCELOS, Marisa, CANDELLO, Heloisa, PINHANEZ, Claudio y DOS SANTOS, Thiago. Bottester. . 2018. P. 1-4. DOI 10.1145/3160504.3160584.
10. SANGROYA, Amit, SAINI, Pratik y ANANTARAM, C. Chatbot as an Intermediary between a Customer and the Customer Care Ecosystem. 2018. P. 128-133. DOI 10.1145/3167020.3167040.
11. ALVAREZ CAMPOS, Luz María y MALCA DÍAZ, Brayan Oscar. Diseño de un sistema Web de búsqueda inteligente conversacional para ubicación de empresas y servicios. [en línea].

2015. Disponible en: [http://dspace.unitru.edu.pe/bitstream/handle/UNITRU/9642/ALVAREZ CAMPOS%2C Luz - MALCA DIAZ%2C Bryan.pdf?sequence=1&isAllowed=y](http://dspace.unitru.edu.pe/bitstream/handle/UNITRU/9642/ALVAREZ%20CAMPOS%20Luz%20-%20MALCA%20DIAZ%20Bryan.pdf?sequence=1&isAllowed=y)

12. CONDORI, William. Desarrollo De Un Asistente Virtual Utilizando Facebook Messenger Para La Mejora Del Servicio De Atención Al Cliente En La Universidad Privada De Tacna En El 2017. [en línea]. 2017. P. 1-292. Disponible en: <http://www.upt.edu.pe/upt/web/home/contenido/100000000/65519409>

13. PETERS, Florian. Design and implementation of a chatbot in the context of customer support. [en línea]. 2018. Disponible en: <http://matheo.uliege.be>

14. ÅBERG, Jakob. Chatbots as A Mean to Motivate Behavior Change How to Inspire Pro-Environmental Attitude with Chatbot Interfaces. [en línea]. 2017. P. 60. Disponible en: <http://www.diva-portal.org/smash/get/diva2:1106358/FULLTEXT01.pdf>

15. GURUPRASAD, R y MARIMUTHU, P. Factor's Deciding Scientists' Choice for Publishing Papers in E-Journals: A Research Study of Aerospace Scientists and Engineers of Bangalore [en línea]. International Journals of Multidimensional Research (IJMR), 2015. [Fecha de Consulta: 3 abril 2019]. Disponible en: https://www.academia.edu/38585527/AN_OVERVIEW_OF_NATURAL_LANGUAGE_PROCESSING_AND_BIG_DATA_ANALYTICS

16. FERRARI, Alessio. Natural language requirements processing. In: Proceedings of the 40th International Conference on Software Engineering Companion Proceedings - ICSE '18 [en línea]. New York, New York, USA: ACM Press, 2018. p. 536-537. [Fecha de Consulta: 3 abril 2019]. ISBN 9781450356633. Disponible en: <http://dl.acm.org/citation.cfm?doid=3183440.3183467>

17. KANSO, Ali y YOUSSEF, Alaa. Serverless. In: Proceedings of the 2nd International Workshop on Serverless Computing - WoSC '17 [en línea]. New York, New York, USA: ACM Press, 2017. p. 6-10. [Fecha de Consulta: 3 abril 2019]. ISBN 9781450354349. Disponible en: <http://dl.acm.org/citation.cfm?doid=3154847.3154854>

18. ADZIC, Gojko y CHATLEY, Robert. Serverless computing: economic and architectural impact. In: Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2017 [en línea]. New York, New York, USA: ACM Press, 2017. p. 884-889. [Fecha de Consulta: 3 abril 2019]. ISBN 9781450351058. Disponible en: <http://dl.acm.org/citation.cfm?doid=3106237.3117767>

19. Comparación de integración continua, entrega continua e implementación continua. [en línea]. [Fecha de Consulta: 7 mayo 2019]. Disponible en: <https://es.atlassian.com/continuous-delivery/principles/continuous-integration-vs-delivery-vs-deployment>

20. What is continuous deployment? - Definition from WhatIs.com. [en línea]. [Fecha de Consulta: 9 mayo 2019]. Disponible en:
<https://searchitoperations.techtarget.com/definition/continuous-deployment>
21. What is REST? | Codecademy. [en línea]. [Fecha de Consulta: 4 abril 2019]. Disponible en: <https://www.codecademy.com/articles/what-is-rest>
22. What's a Webhook? | SendGrid. [en línea]. [Fecha de Consulta: 2 abril 2019]. Disponible en: <https://sendgrid.com/blog/whats-webhook/>
23. What is a Webhook? - Twilio. [en línea]. [Fecha de Consulta: 3 abril 2019]. Disponible en: <https://www.twilio.com/docs/glossary/what-is-a-webhook>
24. BAUDART, Guillaume, HIRZEL, Martin, MANDEL, Louis, SHINNAR, Avraham y SIMÉON, Jérôme. Reactive chatbot programming. In: Proceedings of the 5th ACM SIGPLAN International Workshop on Reactive and Event-Based Languages and Systems - REBLS 2018 [en línea]. New York, New York, USA: ACM Press, 2018. p. 21-30. [Fecha de Consulta: 4 abril 2019]. ISBN 9781450360708. Disponible en:
<http://dl.acm.org/citation.cfm?doid=3281278.3281282>
25. BOZZON, Alessandro. Enterprise crowd computing for human aided chatbots. In: Proceedings of the 1st International Workshop on Software Engineering for Cognitive Services - SE4COG '18 [en línea]. New York, New York, USA: ACM Press, 2018. p. 29-30. [Fecha de Consulta: 4 abril 2019]. ISBN 9781450357401. Disponible en:
<http://dl.acm.org/citation.cfm?doid=3195555.3195566>
26. ¿Qué es machine learning y cómo mejorará la IA de su negocio? | SAP. [en línea]. [Fecha de Consulta: 30 enero 2020]. Disponible en:
<https://www.sap.com/latinamerica/insights/what-is-machine-learning.html>
27. SIEBRA, Clairton, CORREIA, Walter, PENHA, Marcelo, MACEDO, Jefte, QUINTINO, Jonysberg, ANJOS, Marcelo, FLORENTIN, Fabiana, DA SILVA, Fabio Q.B. y SANTOS, Andre L.M. Virtual assistants for mobile interaction: A review from the accessibility perspective. In: ACM International Conference Proceeding Series. Association for Computing Machinery, 4 diciembre 2018. p. 568-571. ISBN 9781450361880.
28. RAFAILIDIS, Dimitrios y MANOLOPOULOS, Yannis. Can Virtual Assistants Produce Recommendations? In: Association for Computing Machinery (ACM), 2019. p. 1-6.
29. FØLSTAD, Asbjørn, KVALE, Knut y HAUGSTVEIT, Ida Maria. Customer support as a source of usability insight: Why users call support after visiting self-service websites. In: Proceedings of the NordiCHI 2014: The 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational. Association for Computing Machinery, Inc, 26 octubre 2014. p. 167-170. ISBN 1595930361.

30. Proyecto Plan Estratégico Institucional Electrocentro 2017-2021. [en línea]. 2019. [Fecha de Consulta: 15 octubre 2019]. Disponible en:
https://www.distriluz.com.pe/transp/ftp/elcto/transp2/PlanEstrategico_elcto.pdf
31. GUÍA, La y SCRUM, Definitiva De. La Guía de Scrum. . 2016.
32. Entities overview Dialogflow. [en línea]. [Fecha de Consulta: 5 abril 2019]. Disponible en:
<https://dialogflow.com/docs/entities>
33. Descripción general de los agentes Documentación de Dialogflow. [en línea]. [Fecha de Consulta: 6 noviembre 2019]. Disponible en:
<https://cloud.google.com/dialogflow/docs/agents-overview?hl=es-419>
34. Intents overview Dialogflow. [en línea]. [Fecha de Consulta: 5 abril 2019]. Disponible en:
<https://dialogflow.com/docs/intents>
35. Contextos de entrada y salida Dialogflow. [en línea]. [Fecha de Consulta: 6 noviembre 2019]. Disponible en: <https://cloud.google.com/dialogflow/docs/contexts-input-output?hl=es-419>
36. Events overview Dialogflow. [en línea]. [Fecha de Consulta: 5 abril 2019]. Disponible en:
<https://dialogflow.com/docs/events>
37. Fulfillment overview Dialogflow. [en línea]. [Fecha de Consulta: 5 abril 2019]. Disponible en: <https://dialogflow.com/docs/fulfillment>
38. Welcome to Netlify | Netlify Docs. [en línea]. [Fecha de Consulta: 6 noviembre 2019]. Disponible en: <https://docs.netlify.com/>
39. HTTP request methods - HTTP | MDN. [en línea]. [Fecha de Consulta: 6 noviembre 2019]. Disponible en: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods>
40. JavaScript | MDN. [en línea]. [Fecha de Consulta: 6 noviembre 2019]. Disponible en: <https://developer.mozilla.org/es/docs/Web/JavaScript>
41. Introduction to Node.js. [en línea]. [Fecha de Consulta: 29 julio 2019]. Disponible en: <https://nodejs.dev/>
42. Express Web Framework (Node.js/JavaScript) - Aprende sobre desarrollo web | MDN. [en línea]. [Fecha de Consulta: 5 abril 2019]. Disponible en:
https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs
43. dialogflow/dialogflow-fulfillment-nodejs: Dialogflow agent fulfillment library supporting v1 and v2, 8 platforms, and text, card, image, suggestion, custom responses. [en línea]. [Fecha de Consulta: 29 julio 2019]. Disponible en: <https://github.com/dialogflow/dialogflow-fulfillment-nodejs>

44. Introducción — Vue.js. [en línea]. [Fecha de Consulta: 29 julio 2019]. Disponible en: <https://es-vuejs.github.io/vuejs.org/v2/guide/>

45. Cloud Firestore. [en línea]. [Fecha de Consulta: 29 julio 2019]. Disponible en: <https://firebase.google.com/docs/firestore?hl=es-419>

46. Bases de datos no relacionales AWS. [en línea]. [Fecha de Consulta: 6 noviembre 2019]. Disponible en: <https://aws.amazon.com/es/nosql/>

ANEXOS

Karina Laura León

Jefa Área de TI

Electrocentro - Huancayo

klaural@distriluz.com.pe

Huancayo, 26 de Abril del 2019

Edwing Cristian Artica LLacta

DNI: 45963283

kjetlag@gmail.com

Estimado Sr. *Edwing Cristian Artica Llacta*:

Quiero agradecerle por la donación del proyecto de Asistente Virtual para la atención a los clientes de Electrocentro Huancayo.

Proyecto que optimizará la información y el tiempo que invierten los usuarios para realizar estas consultas, gracias a que todas estas se solucionarán en el tiempo más breve utilizando esta nueva tecnología de Procesamiento de Lenguaje Natural que fue adaptada para este proyecto y será provista a través de la plataforma de Facebook Messenger y de esa forma tener un servicio óptimo y oportuno.

Algunos de los beneficios que se les ofrecerá a los clientes de Electrocentro con el proyecto donado son tener mayor información sobre el estado de sus servicios, así como brindar ayuda para realizar consultas, tales como conocer el estado de sus deudas, conocer los pasos para realizar quejas, números de contacto, ubicar lugares de pago, entre otros.

Un saludo cordial.



ING. KARINA LAURA LEÓN
Jefa Área Tecnología de la Información
DISTRILUZ

Anexo 1. Carta de Agradecimiento

ACTA DE ACEPTACIÓN DEL PROYECTO

NOMBRE DEL PROYECTO	SIGLAS DEL PROYECTO
ASISTENTE VIRTUAL PARA LA ATENCIÓN A LOS CLIENTES ELECTROCENTRO	-
NOMBRE DEL CLIENTE O SPONSOR	
Ing. Karina Laura León	

DECLARACIÓN DE LA ACEPTACIÓN FORMAL	
<p>Por la presente se deja constancia de que el Proyecto IMPLEMENTACIÓN DE UN ASISTENTE VIRTUAL PARA LA ATENCIÓN AL CLIENTE EN ELECTROCENTRO S.A. Huancayo, ha sido aceptado y aprobado por el Sponsor del Proyecto, Ing. Karina Laura León, por lo que concluye que el proyecto ha sido culminado exitosamente.</p>	
LISTADO DE REQUERIMIENTOS	ESTADO
Desarrollo de Agente NLP en Dialogflow	<i>COMPLETADO</i>
Creación de Página de Facebook para Agente Conversacional	<i>COMPLETADO</i>
Creación de Aplicación de Facebook Messenger Developers	<i>COMPLETADO</i>
Desarrollo de webhook para ampliación de funcionalidad del Asistente Virtual	<i>COMPLETADO</i>
Desarrollo de función serverless para la conexión con el API de deudas de Electrocentro	<i>COMPLETADO</i>
Desarrollo de función serverless para la conexión con el API de Google para la obtención de lugares cercanos de pago	<i>COMPLETADO</i>
Desarrollo de función serverless para la mostrar la guía de reclamos de Electrocentro	<i>COMPLETADO</i>
Desarrollo de función serverless para la conexión con el API de interrupciones de Electrocentro	<i>COMPLETADO</i>



Desarrollo de función serverless para conexión con API de Tenor GIF para la respuesta más amena a usuarios	COMPLETADO
Desarrollo de función serverless para la recepción de reclamos	COMPLETADO
Desarrollo de Intenciones y Entidades para Agente Conversacional	COMPLETADO
El proyecto comprendía de la entrega de los siguientes entregables: <ul style="list-style-type: none"> • Agente de Dialogflow • Página de Facebook • Aplicación de Facebook Developers • Aplicación Webhook en Netlify • Aplicación de Funciones Serverless en Netlify 	
OBSERVACIONES ADICIONALES	
El proyecto ha sido desarrollado dentro de los tiempos planificados, siendo la fecha de término en la planificación 15 de Diciembre del 2018 y la 26 de Abril del 2019.	
ACEPTADO POR	
NOMBRE DEL CLIENTE, SPONSOR U OTRO FUNCIONARIO	FECHA
Ing. Karina Laura León	26 de Abril del 2019
DISTRIBUIDO Y ACEPTADO	
NOMBRE DEL STAKEHOLDER	FECHA
Ing. Karina Laura León	26 de Abril del 2019
Ing. José Antonio Lucas Cornelio	26 de Abril del 2019


ING. KARINA LAURA LEÓN
 Jefe Área Tecnología de la Información
 ESEI-COINTEC S.A.

SOLICITUD PARA EL DESARROLLO DEL PROYECTO DE CHATBOT PARA ATENCIÓN AL CLIENTE

Huancayo, 23 de noviembre del 2018

Ingeniero Romeo Rojas Bravo

Gerente Regional – Electrocentro S.A.



Yo, Edwing Cristian Artica LLacta, bachiller en Ingeniería de Sistemas e Informática, ante usted me presento y le solicito que con el fin de obtener el grado de Ingeniero me conceda autorización para ejecutar el proyecto del desarrollo de una aplicación Chatbot (Asistente Virtual) de Facebook Messenger que permitirá la atención a los clientes, brindándoles a través de esta plataforma información acerca de sus deudas, trámites, procesos, entre otros. Adjunto los detalles del proyecto en un plan de trabajo.

En espera de su respuesta a la solicitud presentada,

Atentamente.

Artica LLacta Edwing Cristian

45963283

PLAN DE TRABAJO

ANTECEDENTES

A lo largo de los últimos años empresas Google, Microsoft y Facebook han enfocado grandes esfuerzos en desarrollar plataformas de Interfaz de usuario conversacional que permiten con una gran facilidad la creación de chatbots y bots que se comunican a través de **mensajes de texto y voz**. Estas plataformas de **Inteligencia Artificial** utilizan Machine Learning y Natural Language Processing (NLP) para su funcionamiento y entrenamiento.

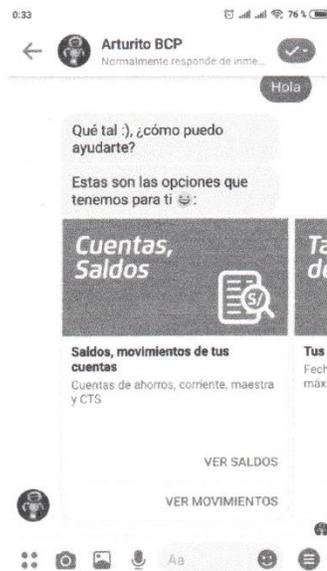
En nuestro país hay casos exitosos del uso de Chatbots e Inteligencia Artificial entre los que tenemos:

Arturito

Desarrollado por IBM para el Banco de Crédito del Perú que funciona sobre la plataforma de Facebook Messenger y la tecnología de Procesamiento de Lenguaje IBM Watson, debido a esta implementación el bot puede realizar funciones como:

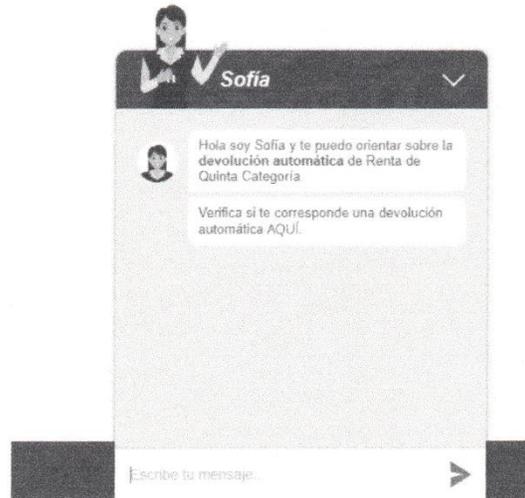
- Consultas a cuentas bancarias
- Movimientos
- Consultas de tarjetas de crédito
- Tipo de cambio
- Localización de un Agente BCP cercano
- Gestionar notificaciones

Actualmente cuenta con alrededor de 500 mil usuarios y en el futuro se planea agregar funcionalidades como manejar cuentas favoritas, pago de servicios y activar tarjetas de crédito.



Sofia

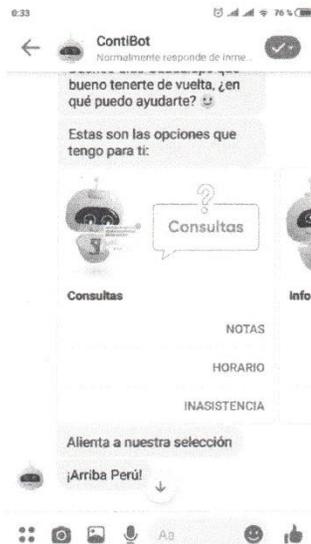
Desarrollado por IBM para la SUNAT para su plataforma web y así como en el caso del BCP hace uso de la tecnología IBM Watson, al funcionar las 24 horas del día a reducido alrededor de un 20% en las consultas tradicionales, es decir vía presencial y por teléfono, aunque por ahora el asistente ayuda a personas naturales en temas relacionados con la quinta y cuarta categoría.



Contibot

Este chatbot fue desarrollado para la Universidad Continental para la atención a todos los alumnos de Pre Grado, funciona sobre Facebook Messenger y para el Procesamiento de lenguaje utiliza la tecnología desarrollada por Microsoft: LUIS.AI, esta implementación permite a los alumnos:

- Conocer sus notas
- Horarios de clases
- Porcentaje de Inasistencias
- Información sobre deudas
- Información sobre trámites presenciales y online
- Información sobre los Campus



Todos estos proyectos tienen además la suficiente inteligencia para llevar una conversación fluida, dando así la impresión a los usuarios de estar hablando con una persona real.

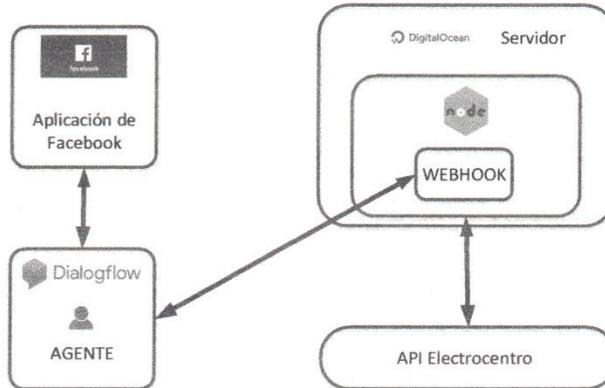
OBJETIVOS

El desarrollo de un sistema conversacional (chatbot) permitiría a los usuarios de Electrocentro realizar:

- Consultas de sus deudas
- Información de Trámites
- Pagos por servicios nuevos
- Ubicar centros de pago

Todo esto se realizaría a través de la plataforma Facebook Messenger y Dialogflow de Google orientado a la atención al cliente, permitiéndoles realizar las consultas antes mencionadas, además al ser Facebook una de las plataformas más usadas permitiría un gran alcance de usuarios por permitiría tener la data de los usuarios actualizada, así como conocer cuáles son las molestias que tienen y por supuesto brindarles una mejor atención.

ARQUITECTURA



La arquitectura mostrada permitirá a los usuarios comunicarse con el asistente a través de Facebook Messenger que captará las intenciones de los usuarios mediante la plataforma de Procesamiento de lenguaje Dialogflow desarrollada por Google, para las consultas más complejas se conectará mediante un webhook con el api de Electrocentro para obtener la data los respectivos usuarios.

COSTOS

Los principales beneficios del uso de las plataformas de Google es el costo prácticamente nulo que estas tienen ya que inicialmente son 0, aunque cabe la posibilidad que se incrementen si es que se superan los 180 usuarios por minuto.

Plataforma	Costo Mensual	Costo Anual
Dialogflow Standard	\$ 0	\$ 0
Firebase Cloud Functions	\$ 0	\$ 0
Digital Ocean Servidor (Despliegue API) Opcional	\$ 10	\$ 120

CRONOGRAMA

Actividad	Asignado a	Tiempo	Fecha Inicio	Fecha Fin
Desarrollo API (Data de los usuario)	Electrocentro	2 Meses	Por determinar	
Desarrollo de Agente Chatbot	Cristian Artica	1 Mes	26-11-2018	27-12-2018
Desarrollo Webhook	Cristian Artica	1 Mes	26-11-2018	27-12-2018

Electrocentro

ADA

ASISTENTE VIRTUAL DE ELECTROCENTRO

Encuesta de Satisfacción Ada Bot de Electrocentro

Nombres y Apellidos

Tu respuesta _____

Género

Varón

Mujer

Edad

Tu respuesta _____

Distrito en el que vive

Huancayo

El Tambo

Chilca

Otros: _____

Con qué frecuencia utiliza al Asistente Virtual ADA de Electrocentro

Diariamente

Semanalmente

Mensualmente

Anualmente

Anexo 9. Encuesta de satisfacción Ada Bot de Electrocentro (1)

¿Con qué fin utiliza el Asistente Virtual ADA de Electrocentro?

- Consulta de recibos
- Consulta de facturación de los últimos meses
- Lugares de pago cercanos
- Guía de reclamos
- Números de contacto
- Consulta de interrupciones programadas
- Reportar incidencias

¿Es realmente útil el Asistente ADA para usted?

- Muy de acuerdo
- De acuerdo
- Indiferente
- En desacuerdo
- Muy en desacuerdo

¿Considera que el asistente ADA le ayuda a conocer los detalles de su recibo?

- Muy de acuerdo
- De acuerdo
- Indiferente
- En desacuerdo
- Muy en desacuerdo

Anexo 10. Encuesta de satisfacción Ada Bot de Electrocentro (2)

¿Considera que el asistente ADA le ayuda a encontrar de lugares de pago cercanos?

- Muy de acuerdo
- De acuerdo
- Indiferente
- En desacuerdo
- Muy en desacuerdo

¿Considera que el asistente ADA le ayuda a conocer su facturación de los últimos meses?

- Muy de acuerdo
- De acuerdo
- Indiferente
- En desacuerdo
- Muy en desacuerdo

¿Considera que el asistente ADA le ayuda a conocer los pasos para realizar reclamos?

- Muy de acuerdo
- De acuerdo
- Indiferente
- En desacuerdo
- Muy en desacuerdo

Anexo 11. Encuesta de satisfacción Ada Bot de Electrocentro (3)

¿Considera que el asistente ADA le ayuda a conocer los números de contacto de Electrocentro?

- Muy de acuerdo
- De acuerdo
- Indiferente
- En desacuerdo
- Muy en desacuerdo

¿Considera que el asistente ADA le ayuda a conocer las interrupciones de servicio programadas?

- Muy de acuerdo
- De acuerdo
- Indiferente
- En desacuerdo
- Muy en desacuerdo

¿Considera que el asistente ADA le ayuda a reportar incidencias con su servicio?

- Muy de acuerdo
- De acuerdo
- Indiferente
- En desacuerdo
- Muy en desacuerdo

Anexo 12. Encuesta de satisfacción da Bot de Electrocentro (4)

¿Considera que el asistente ADA le ayuda a reportar incidencias con su servicio?

Muy de acuerdo

De acuerdo

Indiferente

En desacuerdo

Muy en desacuerdo

¿Cómo descubriste el Asistente Virtual ADA de Electrocentro?

Fanpage de Facebook

Banners Informativos

Recomendaciones de familiares o amigos

Otros: _____

¿Tiene alguna sugerencia para el Asistente Virtual ADA de Electrocentro?

Tu respuesta _____

ENVIAR

Nunca envíes contraseñas a través de Formularios de Google.

Anexo 13. Encuesta de satisfacción Ada Bot de Electrocentro (5)

```

"use strict";
const { WebhookClient } = require("dialogflow-fulfillment");
const { Text, Image, Payload } = require("dialogflow-fulfillment");
const express = require("express");
const serverless = require("serverless-http");
const app = express();
const bodyParser = require("body-parser");
const axios = require("axios");

const router = express.Router();

function WebhookProcessing(request, response) {
  const agent = new WebhookClient({ request: request, response: response });
  console.info(`agent set`);
  console.log("Dialogflow Request headers: " + JSON.stringify(request.headers));
  console.log("Dialogflow Request body: " + JSON.stringify(request.body));
  //Aquí se detallan las funciones que son mapeadas con sus respectivas
  //intenciones

  let intentMap = new Map();
  intentMap.set("payment.locations - bank - coordinates", getUserLocation);
  intentMap.set("user.goodbye", userGoodbye);
  intentMap.set("user.thanks", userThanks);
  intentMap.set("user.invoice - number", debt);
  intentMap.set("user.debts - number", debt);
  intentMap.set("user.interruptions - number", interruptions);
  intentMap.set("billing.amount - service", historicalBilling);
  agent.handleRequest(intentMap);
}

router.post("/", (req, res) => {
  WebhookProcessing(req, res);
});

app.use(bodyParser.json());
app.use(bodyParser.urlencoded({ extended: true }));
app.use("/.netlify/functions/webhook", router); // path must route to lambda

module.exports = app;
module.exports.handler = serverless(app);

```

Anexo 144. Código del webhook principal del Asistente Virtual

```

const axios = require("axios");
// const { SERVICES_MAIN_MOBILE } = require('../config').config;

exports.handler = function(event, context, callback) {
  let { IdNroServicio } = JSON.parse(event.body);

  console.log(
    "process env",
    `${process.env.SERVICES_MAIN_MOBILE}/consultaresumenultimafacturacion`
  );
  console.log("nro serv", IdNroServicio);
  let response = {};
  axios
    .post(
      `${process.env.SERVICES_MAIN_MOBILE}/consultaresumenultimafacturacion`,
      {
        ParametroControlTag: "idnroservicio|" + IdNroServicio
      }
    )
    .then(res => {
      console.log("res", res);
      if (res.data.ListaInformacion[0].ListaRegistro[0] === "") {
        response.serviceCut = "No tiene deudas registradas 😊";
        callback(null, {
          statusCode: 200,
          body: JSON.stringify(response)
        });
      } else {
        const result =
res.data.ListaInformacion[0].ListaRegistro[0].split(";");

        response.issuance = result[6].substr(6, result[6].length);
        console.log("Importe del Mes", result[11].substr(6,
result[11].length));
        response.monthlyDebt = result[11].substr(6, result[11].length);

        response.billedDebt = result[16].substr(6, result[16].length);

        response.expiration = result[21].substr(6, result[21].length);

        response.currentDebt = result[26].substr(6, result[26].length);

        response.paymentDate = result[36].substr(6, result[36].length);

```

```
response.cutDate = result[41].substr(6, result[41].length);

callback(null, {
  statusCode: 200,
  body: JSON.stringify(response)
});
}
})
.catch(err => {
  console.log(err);
});
};
```

Anexo 155. Código de consulta de última facturación

```

const axios = require('axios');
// const { TENOR_GIF_URL, TENOR_GIF_KEY, TENOR_ANON_ID } =
require('../config').config;

exports.handler = function(event, context, callback) {
  const { query } = event.queryStringParameters;
  const randomNumber = Math.floor(Math.random() * 10);

  console.log(`${process.env.TENOR_GIF_URL}?q=${query}&limit=10&key=${process.env.TENOR_GIF_KEY}&anon_id=${process.env.TENOR_ANON_ID}`)

  axios.get(`${process.env.TENOR_GIF_URL}?q=${query}&limit=10&key=${process.env.TENOR_GIF_KEY}&anon_id=${process.env.TENOR_ANON_ID}`)
    .then(res => {
      callback(null, {
        statusCode: 200,
        body: JSON.stringify({gif:
res.data.results[randomNumber].media[0].tinygif.url})
      });
    })
    .catch(err => {
      console.log(err);
    });
}

```

Anexo 166. Código de consulta de API de Tenor GIF

```

const axios = require("axios");
// const { SERVICES_MAIN_URL } = require('../config').config;

exports.handler = function(event, context, callback) {
  const { IdNroServicio } = JSON.parse(event.body);
  console.log(
    "process env",
    `${process.env.SERVICES_MAIN_URL}/ListarInterrupciones`
  );
  console.log("nro serv", IdNroServicio);
  let response = {};
  axios
    .post(`${process.env.SERVICES_MAIN_URL}/ListarInterrupciones`, {
      Parametros: { IdNroServicio: IdNroServicio }
    })
    .then(res => {
      if (res.data.Datos.length === 0) {
        response.serviceCut = "No tiene interrupciones programadas 😊";
        callback(null, {
          statusCode: 200,
          body: JSON.stringify(response)
        });
      } else {
        response.serviceCut = `Tiene un corte programado para el ${
          res.data.Datos[0].FechaInicio
        } a las ${res.data.Datos[0].HoraInicio} por el motivo: ${
          res.data.Datos[0].TipoRegistro
        }`;
        callback(null, {
          statusCode: 200,
          body: JSON.stringify(response)
        });
      }
    })
    .catch(err => {
      console.log(err);
    });
};

```

Anexo 177. Código de consulta de interrupciones programadas

```

const axios = require('axios');
// const { GOOGLE_PLACES_API_KEY, GOOGLE_PLACES_API_URL, LOCATION_IMAGE
} = require('../config').config;

exports.handler = function(event, context, callback) {
  const { latitude, longitude, bank } = event.queryStringParameters;

  axios.get(`${process.env.GOOGLE_PLACES_API_URL}?location=${latitude},${
longitude}&radius=400&type=banco&keyword=${bank}&key=${process.env.GOOG
LE_PLACES_API_KEY}`)
    .then(res => {
      const places = res.data.results.slice(0, 4).map((place, i) => {
        if (i === 0) {
          return {title: place.name, subtitle: place.vicinity,
image_url: process.env.LOCATION_IMAGE, buttons: []}
        }
        return {title: place.name, subtitle: place.vicinity, image_url:
'', buttons: []}
      });
      callback(null, {
        statusCode: 200,
        body: JSON.stringify(places)
      });
    })
    .catch(err => {
      console.log(err);
    });
}

```

Anexo 188. Código de consulta de API de Google Places

```

const cheerio = require('cheerio');
const request = require('request');
// const { SCRAPER_URL } = require('../config').config;

exports.handler = function(event, context, callback) {
  const { nroservicio, intention } = JSON.parse(event.body);
  request.post({url: process.env.SCRAPER_URL, form: { nroservicio }},
    function(err, httpResponse, body) {
      if (!err && httpResponse.statusCode == 200) {
        const $ = cheerio.load(body);
        const debts = $('<col-sm-4>').eq(1);
        const response = {};
        if (intention === 'recibo') {
          response.title = debts.find('h4').first().text();
          response.subtitle = debts.find('h5').first().text();
          response.description = debts.find('panel-
body').first().children('h6').map(function(i, el) {
            if ($(this).text() !== '') {
              return $(this).text()
            }
          }).get();
          console.log('response:', response)
        } else if (intention === 'deuda') {
          response.title = debts.find('h4').last().text();
          response.subtitle =
debts.find('h5').last().text().replace('Dueda', 'Deuda');
          response.description = debts.find('panel-
body').last().children('h6').map(function(i, el) {
            if ($(this).text() !== '') {
              return $(this).text()
            }
          }).get();
          console.log('response:', response)
        }

        callback(null, {
          statusCode: 200,
          body: JSON.stringify(response)
        });
      }
    })
}

```

Anexo 199. Código de webscrapping a Página de Electrocentro

```
config:
  target: "https://ada-bot.netlify.com/.netlify/functions"
  phases:
    - duration: 60
      arrivalRate: 20
  scenarios:
    - flow:
      - post:
          url: "/webhook"
```

Anexo 20. Prueba de Fiabilidad Consulta Webhook Principal API de Electrocentro

```
config:
  target: "https://ada-serverless-functions.netlify.com/.netlify/functions"
  phases:
    - duration: 30
      arrivalRate: 10
  payload:
    path: "service.csv"
    fields:
      - "number"
  scenarios:
    - flow:
      - post:
          url: "/debts"
          json:
            IdNroServicio: "{{number}}"
```

Anexo 21. Prueba de Fiabilidad Consulta Último Recibo

```
config:
  target: "https://ada-serverless-functions.netlify.com/.netlify/functions"
  phases:
    - duration: 30
      arrivalRate: 10
  payload:
    path: "service.csv"
    fields:
      - "number"
  scenarios:
    - flow:
      - post:
          url: "/historicalBilling"
          json:
            IdNroServicio: "{{number}}"
```

Anexo 22. Prueba de Fiabilidad Consulta Facturación Últimos 4 meses

```

config:
  target: "https://ada-serverless-functions.netlify.com/.netlify/functions"
  phases:
    - duration: 30
      arrivalRate: 10
scenarios:
  - flow:
    - get:
      url: "/paymentLocations?latitude=-12.076557&longitudo=-75.213813&bank=interbank"

```

Anexo 23. Prueba de Fiabilidad Consulta de Lugares Cercanos de Pago

```

config:
  target: "https://ada-serverless-functions.netlify.com/.netlify/functions"
  phases:
    - duration: 30
      arrivalRate: 10
  payload:
    path: "service.csv"
    fields:
      - "number"
scenarios:
  - flow:
    - post:
      url: "/interruptions"
      json:
        IdNroServicio: "{{number}}"

```

Anexo 24. Prueba de Fiabilidad Consulta de Interrupciones de Servicio Eléctrico

```

config:
  target: "https://us-central1-ada-bot-1b3d7.cloudfunctions.net"
  phases:
    - duration: 15
      arrivalRate: 5
  payload:
    path: "service.csv"
    fields:
      - "fullname"
      - "dni"
      - "location"
      - "description"
scenarios:
  - flow:

```

```
- post:  
  url: "/storeIncidents"  
  json:  
    fullname: "{{fullname}}"  
    dni: "{{dni}}"  
    location: "{{location}}"  
    description: "{{description}}"
```

Anexo 25. Prueba de Fiabilidad Reporte de Incidencias de Servicio Eléctrico