

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

***IMPLEMENTACIÓN DE UN CHATBOT CON BOTFRAMEWORK:
CASO DE ESTUDIO, SERVICIOS A CLIENTES DEL ÁREA DE
FIANZAS DE SEGUROS EQUINOCCIAL***

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO
INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

OMAR HUMBERTO ZARABIA ZUÑIGA

omar.zarabia@epn.edu.ec

DIRECTOR: MSc. MARITZOL TENEMAZA

maritzol.tenemaza@epn.edu.ec

QUITO, ABRIL 2018

DECLARACIÓN

Yo, Omar Humberto Zarabia Zuñiga, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Omar Humberto Zarabia Zuñiga

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Omar Humberto Zarabia Zuñiga, bajo mi supervisión.

MSc. Maritzol Tenemaza

Directora del Proyecto

AGRADECIMIENTOS

Agradezco primeramente a Dios, quien me da la fuerza de voluntad para alcanzar mis metas y objetivos.

A mi abuelita y a mi madre, quienes me han apoyado en todo momento y me han animado a seguir adelante.

A mi familia, que me apoya en todos mis proyectos.

A la Msc. Maritzol Tenemaza, por su ayuda en el desarrollo de este proyecto.

A Rubén Sánchez, de la empresa Seguros Equinoccial, por ayudarme a crecer profesionalmente y darme la oportunidad de realizar mi proyecto de grado en el Área que el lidera.

A mis compañeros y amigos, con quienes he compartido buenos momentos durante mi vida de estudiante.

Omar

DEDICATORIA

El presente proyecto está dedicado a mi abuelita y a mi madre, quienes son las dos mujeres que más quiero en este mundo. Gracias a ellas he podido seguir adelante para alcanzar mis metas.

Omar.

CONTENIDO

1. INTRODUCCIÓN	1
1.1 PLANTEAMIENTO DEL PROBLEMA	1
1.2 OBJETIVOS	1
1.2.1 OBJETIVO GENERAL	1
1.2.2 OBJETIVOS ESPECÍFICOS	1
1.3 CONCEPTOS DE IA RELACIONADOS A LOS CHATBOTS	2
1.3.1 SISTEMAS EXPERTOS	2
1.3.1.1 TIPOS DE SISTEMAS EXPERTOS	3
1.3.1.2 IMPORTANCIA DE LOS SISTEMAS EXPERTOS	3
1.3.2 MACHINE LEARNING	3
1.3.3 PROCESAMIENTO DE LENGUAJE NATURAL	4
1.3.3.1 ARQUITECTURA DE UN SISTEMA PLN	4
1.3.3.2 APLICACIONES PLN	5
1.4 CHATBOTS O AGENTES CONVERSACIONALES	6
1.4.1 RESEÑA HISTORICA DE LOS CHATBOTS	6
1.4.2 CARÁCTERISTICAS DE UN CHATBOT	8
1.4.3 ARQUITECTURA DE UN CHATBOT	9
1.4.4 CATEGORÍA DE CHATBOTS	10
1.4.4.1 CHATBOTS SEGÚN EL TIPO DE SERVICIO	11
1.4.4.2 CHATBOTS SEGÚN SU DISEÑO DE INTERFAZ	11
1.4.4.3 CHATBOTS SEGÚN LA TECNOLOGÍA UTILIZADA	11
1.4.5 APLICACIONES DE LOS CHATBOTS	11
1.4.6 CASOS DE ÉXITO	12
1.4.7 TECNOLOGÍA PARA EL DESARROLLO DE CHATBOTS	14
1.4.7.1 BOT FRAMEWORK	15
1.4.7.1.1 BOT BUILDER	15
1.4.7.1.2 CANALES QUE SE PUEDE UTILIZAR CON BOTFRAMEWORK	15
1.4.7.1.3 BOT CONNECTOR	16
1.4.7.1.4 LA SEGURIDAD CON BOT FRAMEWORK	17
1.4.7.1.5 TECNOLOGÍAS DE AUTENTICACIÓN	17
1.4.8 SERVICIOS COGNITIVOS DE MICROSOFT	18
1.4.8.1 QNAMAKER	18
1.4.8.2 LANGUAGE UNDERSTANDING INTELLIGENT SERVICE (LUIS)	19
2. METODOLOGÍA	21

2.1 METODOLOGÍA DE DESARROLLO.....	21
2.1.1 SCRUM	21
2.1.1 ROLES DEL PROYECTO.....	21
2.1.2 ARTEFACTOS	22
2.2 ARQUITECTURA DEL PROYECTO.....	22
2.3 HERRAMIENTAS DE DESARROLLO.....	24
2.4 DESARROLLO DEL SISTEMA.....	24
2.4.1 DEFINICIÓN DEL PROYECTO.....	24
2.4.1.1 REQUERIMIENTOS	24
2.4.1.2 HISTORIAS DE USUARIO ÉPICAS	25
2.4.1.3 PRODUCT BACKLOG	27
2.4.1.4 REALEASE PLANNING	28
2.4.2 EJECUCIÓN DE SPRINTS.....	28
2.4.2.1 SPRINT 0	28
2.4.2.1.1 SPRINT PLANNING	28
2.4.2.1.2 SPRINT REVIEW.....	30
2.4.2.1.3 SPRINT RETROSPECTIVE.....	32
2.4.2.2 SPRINT 1	32
2.4.2.2.1 SPRINT PLANNING	32
2.4.2.2.2 SPRINT REVIEW.....	43
2.4.2.2.3 SPRINT RETROSPECTIVE.....	45
2.4.2.3 SPRINT 2	46
2.4.2.3.1 SPRINT PLANNING	46
2.4.2.3.2 SPRINT REVIEW.....	50
2.4.2.3.3 SPRINT RETROSPECTIVE.....	51
2.4.2.4 SPRINT 3	52
2.4.2.4.1 SPRINT PLANNING	52
2.4.2.4.2 SPRINT REVIEW.....	55
2.4.2.4.3 SPRINT RETROSPECTIVE.....	56
2.4.2.5 SPRINT 4	57
2.4.2.5.1 SPRINT PLANNING	57
2.4.2.5.2 SPRINT REVIEW.....	60
2.4.2.5.3 SPRINT RETROSPECTIVE.....	61
2.4.3 APLICACIÓN DE CASOS DE PRUEBA	62
2.4.3.1 CASO DE PRUEBA - MÓDULO SOLICITUD DE PÓLIZAS NUEVAS	62

2.4.3.2 CASO DE PRUEBA - MÓDULO CONSULTA DE ESTADOS DE CUENTA Y PÓLIZAS POR VENCER	65
2.4.3.3 CASO DE PRUEBA – MÓDULO PREGUNTAS FRECUENTES.....	67
2.4.3.4 CASO DE PRUEBA – MÓDULO RENOVACIONES DE PÓLIZAS.....	68
3 RESULTADOS Y DISCUSIÓN.....	70
3.1 APLICACIÓN DEL CASO DE ESTUDIO	70
3.1.1 EJEMPLO DE USO DEL CHATBOT.....	70
3.1.1.1 Módulo de solicitudes de pólizas nuevas	71
3.1.1.2 Módulo de consultas.....	76
3.1.1.3 Módulo de renovaciones de pólizas.....	78
3.2 EVALUACION DE SATISFACCION Y EFICIENCIA DE LA APLICACIÓN	81
3.2.1 ESTRUCTURA DE LA ENCUESTA.....	81
3.2.2 ANÁLISIS DE RESULTADOS DE LA ENCUESTA	83
3.2.2.1 RESULTADO POR PREGUNTA.....	83
3.2.2.2 RESULTADO POR CATEGORÍA.....	84
3.2.2.3 RESULTADO GLOBAL DE LA ENCUESTA	85
4 CONCLUSIONES.....	87
5 REFERENCIAS.....	88
ANEXOS.....	92
ANEXO A.....	93
ANEXO B.....	97
ANEXO C.....	99

FIGURAS

Figura 1. Ilustración de componentes de un sistema experto basado en [2].	2
Figura 2. Ilustración de la arquitectura de un sistema de procesamiento de lenguaje natural, tomando como fuente [7].	5
Figura 3. Ilustración de algunos eventos en la historia de la IA, recurso fundamentado de [11] [12] [13] [14] [17].	8
Figura 4. Ilustración de la arquitectura básica de un chatbot, basado en [9].	9
Figura 5. Ilustración del funcionamiento básico de un chatbot, fundamentado en [9].	10
Figura 6. Ilustración de campos de aplicación de los chatbots, fundamentado en [9,17].	12
Figura 7. Ilustración de chatbot creado por Mastercard, recurso tomado de [22].	13
Figura 8. Ilustración chatbot BBC mundo, recurso tomado de [23].	13
Figura 9. Ilustración chatbot Billy Seguros, recurso tomado de [25].	14
Figura 10. Ilustración de la arquitectura del Bot Framework, recurso tomado de [21].	16
Figura 11. Ilustración de funcionamiento QnA Maker Api, recurso tomado de [30].	18
Figura 12. Ilustración funcionamiento de Language Understanding Intelligent Service (LUIS) recurso elaborado por el autor.	19
Figura 13. Arquitectura MVC del proyecto (Fuente: elaborado por el autor y fundamentado en [21,30,31]).	23
Figura 14. Arquitectura basada en intenciones del chatbot (Fuente: elaborado por el autor)	25
Figura 15. Prototipo presentación chatbot (Fuente: elaborado por el autor).	30
Figura 16. Burndown Chart del Sprint 0 (Fuente: elaborado por el autor).	32
Figura 17. Módulo de solicitud de pólizas nuevas - consulta de cliente, beneficiario - información general de póliza (Fuente: elaborado por el autor).	40
Figura 18. Módulo solicitud de póliza nueva – selección del tipo de póliza (Fuente: elaborado por el autor).	41
Figura 19. Módulo solicitud de póliza nueva – adjuntar documentos – generar solicitud (Fuente: elaborado por el autor).	42
Figura 20. Burndown Chart del Sprint 1 (Fuente: elaborado por el autor).	45
Figura 21. Módulo consultas -estados de cuenta – pólizas por vencer (Fuente: elaborado por el autor).	49
Figura 22. Burndown Chart del Sprint 1 (Fuente: elaborado por el autor).	51
Figura 23. Módulo de preguntas frecuentes (Fuente: elaborado por el autor).	54
Figura 24. Burndown Chart del Sprint 3 (Fuente: elaborado por el autor).	56
Figura 25. Módulo de renovaciones (Fuente: elaborado por el autor).	59
Figura 26. Burndown Chart del Sprint 4 (Fuente: elaborado por el autor).	61
Figura 27. Inicio de sesión para acceder a los módulos de Fianzas (Fuente: elaborado por el autor).	70
Figura 28. Interacción con el módulo de preguntas frecuentes (Fuente: elaborado por el autor).	71
Figura 29. Ingreso de Cliente y beneficiario de la póliza a ser solicitada (Fuente: elaborado por el autor).	72
Figura 30. Ingreso del monto, plazo y objeto contractual de la póliza a ser solicitada (Fuente: elaborado por el autor).	72
Figura 31. Ingreso de un tipo de póliza para la solicitud (Fuente: elaborado por el autor).	73
Figura 32. Diálogo para adjuntar documentos (Fuente: elaborado por el autor).	74

Figura 33. Interfaz web para adjuntar documentos (Fuente: elaborado por el autor). .	74
Figura 34. Diálogo para generar o cancelar la solicitud (Fuente: elaborado por el autor).	75
Figura 35. Solicitud generada desde el chatbot (Fuente: elaborado por el autor).....	75
Figura 36. Reporte del convenio de la solicitud generada (Fuente: elaborado por el autor).	76
Figura 37. Opciones de consulta desde el chatbot (Fuente: elaborado por el autor)...	76
Figura 38. Consulta estado de cuenta desde chatbot (Fuente: elaborado por el autor).	77
Figura 39. Reporte web de consulta de estado (Fuente: elaborado por el autor).	77
Figura 40. Consulta pólizas por vencer desde chatbot (Fuente: elaborado por el autor).	78
Figura 41. Reporte web de pólizas por vencer (Fuente: elaborado por el autor).	78
Figura 42. Lista de sucursales presentada desde el chatbot (Fuente: elaborado por el autor).	79
Figura 43. Lista de tipos de póliza y diálogo de ingreso de número de póliza (Fuente: elaborado por el autor).....	79
Figura 44. Información desplegada de póliza consultada y encontrada (Fuente: elaborado por el autor).....	80
Figura 45. Opciones para realizar la renovación desde el chatbot (Fuente: elaborado por el autor).	80
Figura 46. Mensaje de espera del chatbot (Fuente: elaborado por el autor).	81
Figura 47. Resumen de póliza renovada (Fuente: elaborado por el autor).	81
Figura 48. Sección de efectividad del chatbot (Fuente: elaborado por el autor).	82
Figura 49. Sección de satisfacción de usuario (Fuente: elaborado por el autor).	82
Figura 50. Resultado por pregunta (Fuente: elaborado por el autor).	83
Figura 51. Porcentaje de aceptación por pregunta (Fuente: elaborado por el autor)...	84
Figura 52. Resultado por categoría (Fuente: elaborado por el autor).	84
Figura 53. Porcentaje por categoría (Fuente: elaborado por el autor).	85
Figura 54. Resultado global de la encuesta (Fuente: elaborada por el autor).	85
Figura 55. Porcentaje global de los resultados de la encuesta (Fuente: elaborada por el autor).	86
Figura 56. Diagrama de flujo proceso de solicitud de póliza nueva parte A (Fuente: elaborado por el autor).....	93
Figura 57. Diagrama de flujo proceso de solicitud de póliza nueva parte B (Fuente: elaborado por el autor).....	94
Figura 58. Diagrama de flujo proceso de solicitud de póliza nueva parte C (Fuente: elaborado por el autor).....	95
Figura 59. Diagrama de flujo proceso de solicitud de póliza nueva parte D (Fuente: elaborado por el autor).....	96
Figura 60. Diagrama de flujo proceso de consulta de estados de cuenta (Fuente: elaborado por el autor).....	97
Figura 61. Diagrama de flujo proceso de consultas de polizas por vencer (Fuente: elaborado por el autor).....	98
Figura 62. Diagrama de flujo proceso de renovación de póliza (Fuente: elaborado por el autor).	99

TABLAS

Tabla 1. Cuadro de tecnologías de autenticación que usa Bot Framework, recurso tomado de [21].	17
Tabla 2. Roles del Proyecto (Fuente: elaborado por el autor).	22
Tabla 3. Historia épica HUE01 (Fuente: elaborado por el autor).	26
Tabla 4. Historia épica HUE02 (Fuente: elaborado por el autor).	26
Tabla 5. Historia épica HUE03 (Fuente: elaborado por el autor).	26
Tabla 6. Historia épica HUE04 (Fuente: elaborado por el autor).	26
Tabla 7. Product Backlog (Fuente: elaborado por el autor).	27
Tabla 8. Release Planning (Fuente: elaborado por el autor).	28
Tabla 9. Historia de Usuario 1 (Fuente: elaborado por el autor).	28
Tabla 10. Historia de Usuario 2 (Fuente: elaborado por el autor).	29
Tabla 11. Product backlog del Sprint 0 parte A (Fuente: elaborado por el autor).	29
Tabla 12. Product backlog del Sprint 0 parte B (Fuente: elaborado por el autor).	30
Tabla 13. Revisión de Criterios de Aceptación del Sprint 0 (Fuente: elaborado por el autor).	31
Tabla 14. Tablero Kanban del Sprint 0 (Fuente: elaborado por el autor).	31
Tabla 15. Historia de Usuario 3 (Fuente: elaborado por el autor).	33
Tabla 16. Historia de Usuario 4 (Fuente: elaborado por el autor).	33
Tabla 17. Historia de Usuario 5 (Fuente: elaborado por el autor).	34
Tabla 18. Historia de Usuario 6 (Fuente: elaborado por el autor).	34
Tabla 19. Historia de Usuario 7 (Fuente: elaborado por el autor).	35
Tabla 20. Historia de Usuario 8 (Fuente: elaborado por el autor).	35
Tabla 21. Historia de Usuario 9 (Fuente: elaborado por el autor).	35
Tabla 22. Product backlog del Sprint 1 parte A (Fuente: elaborado por el autor).	36
Tabla 23. Product backlog del Sprint 1 parte B (Fuente: elaborado por el autor).	37
Tabla 24. Product backlog del Sprint 1 parte C (Fuente: elaborado por el autor).	38
Tabla 25. Product backlog del Sprint 1 parte D (Fuente: elaborado por el autor).	39
Tabla 26. Revisión de Criterios de Aceptación del Sprint 1 parte A (Fuente: elaborado por el autor).	43
Tabla 27. Revisión de Criterios de Aceptación del Sprint 1 parte B (Fuente: elaborado por el autor).	44
Tabla 28. Tablero Kanban del Sprint 1 (Fuente: elaborado por el autor).	45
Tabla 29. Historia de Usuario 10 (Fuente: elaborado por el autor).	46
Tabla 30. Historia de Usuario 11 (Fuente: elaborado por el autor).	47
Tabla 31. Product backlog del Sprint 2 parte A (Fuente: elaborado por el autor).	47
Tabla 32. Product backlog del Sprint 2 parte B (Fuente: elaborado por el autor).	48
Tabla 33. Revisión de Criterios de Aceptación del Sprint 2 (Fuente: elaborado por el autor).	50
Tabla 34. Tablero Kanban del Sprint 2 (Fuente: elaborado por el autor).	51
Tabla 35. Historia de Usuario 12 (Fuente: elaborado por el autor).	52
Tabla 36. Historia de Usuario 13 (Fuente: elaborado por el autor).	52
Tabla 37. Product backlog del Sprint 3 (Fuente: elaborado por el autor).	53
Tabla 38. Revisión de Criterios de Aceptación del Sprint 3 (Fuente: elaborado por el autor).	55
Tabla 39. Tablero Kanban del Sprint 3 (Fuente: elaborado por el autor).	56
Tabla 40. Historia de Usuario 14 (Fuente: elaborado por el autor).	57
Tabla 41. Historia de Usuario 15 (Fuente: elaborado por el autor).	57
Tabla 42. Product backlog del Sprint 4 (Fuente: elaborado por el autor).	58

Tabla 43. Revisión de Criterios de Aceptación del Sprint 4 (Fuente: elaborado por el autor)	60
Tabla 44. Tablero Kanban del Sprint 4 (Fuente: elaborado por el autor).	61
Tabla 45. Casos de prueba Módulo solicitudes de pólizas nuevas parte A (Fuente: elaborado por el autor).....	62
Tabla 46. Casos de prueba Módulo solicitudes de pólizas nuevas parte B (Fuente: elaborado por el autor).....	63
Tabla 47. Casos de prueba Módulo solicitudes de pólizas nuevas parte C (Fuente: elaborado por el autor).....	64
Tabla 48. Casos de prueba Módulo solicitudes de pólizas nuevas parte D (Fuente: elaborado por el autor).....	65
Tabla 49. Casos de prueba Módulo de consulta de estados de cuenta y pólizas por vencer de clientes parte A (Fuente: elaborado por el autor).....	65
Tabla 50. Casos de prueba Módulo de consulta de estados de cuenta y pólizas por vencer de clientes parte B (Fuente: elaborado por el autor).....	66
Tabla 51. Casos de prueba Módulo de preguntas frecuentes por vencer de clientes (Fuente: elaborado por el autor).	67
Tabla 52. Casos de prueba Módulo de renovaciones de pólizas (Fuente: elaborado por el autor).	68
Tabla 53. Tablero Kanban finalizadas las pruebas funcionales (Fuente: elaborado por el autor).	69

Resumen

El presente documento se centra en el desarrollo de un chatbot a ser usado por los brókers y clientes del área de Fianzas de Seguros Equinoccial (caso de estudio) como medio de atención al cliente. El documento propone una arquitectura que integra los servicios web propios del caso de estudio, una base de conocimiento, los servicios cognitivos y un bot framework de Microsoft. Esta arquitectura permite al chatbot dar respuestas efectivas a través de un diálogo personalizado y enfocado en el procesamiento de lenguaje natural y machine learning. El diálogo es creado en función de los módulos de solicitud de pólizas nuevas, consultas, renovaciones y preguntas frecuentes. Para la interfaz del chatbot se designaron tres canales (página web de Fianzas de Seguros Equinoccial, Facebook Messenger y Skype como opcional). Adicionalmente, se plantea el uso de dos características de calidad para la evaluación del chatbot, estas son: la efectividad y la satisfacción del usuario.

Palabras clave: Chatbot, procesamiento de lenguaje natural, machine learning, Microsoft Bot Framework, Scrum.

Abstract

This document focuses on the development of a chatbot to be used by brokers and clients of Seguros Equinoccial bond area (case study) as a means for customer service. The document proposes an architecture that integrates the web services of the case study, a knowledge base, cognitive services and the Microsoft bot framework. This architecture allows the chatbot to give effective answers through a personalized dialogue focused on the processing of natural language and machine learning. The dialogue is created based on the modules of request of new policies, consultations, renewals and frequently asked questions. For the chatbot interface, three channels were designated (the webpage of Seguros Equinoccial bond area, Facebook Messenger and Skype as optional). Additionally, the use of two quality characteristics is proposed for the evaluation of the chatbot, these are: effectiveness and user satisfaction.

Keywords: Chatbot, natural language processing, machine learning, Microsoft Bot Framework, Scrum.

1. INTRODUCCIÓN

1.1 PLANTEAMIENTO DEL PROBLEMA

Actualmente los usuarios (brókers y clientes) de Seguros Equinoccial utilizan el sistema de fianzas publicado en la página web de la aseguradora. Los usuarios requieren soporte para el uso de la página web y esta carga operativa afecta directamente a la productividad del personal de fianzas.

Para dar respuesta a este problema, se plantea implementar un chatbot que permita al usuario tener una interacción personalizada, con tiempos de respuesta óptimos y con distintos canales de acceso.

1.2 OBJETIVOS

1.2.1 OBJETIVO GENERAL

- Implementar un chatbot con Bot Framework: caso de estudio, servicios a clientes del área de fianzas de seguros equinoccial.

1.2.2 OBJETIVOS ESPECÍFICOS

- Analizar los procesos empleados en los servicios del área de fianzas de Seguros Equinoccial.
- Diseñar un flujo que adapte los procesos del área de fianzas de Seguros Equinoccial a la arquitectura de un chatbot.
- Construir un prototipo chatbot con cuatro módulos que son: módulo de solicitudes de pólizas nuevas, módulo de consultas de estados de cuenta y pólizas por vencer, módulo renovaciones de pólizas y módulo de preguntas frecuentes para el área de fianzas de Seguros Equinoccial.
- Configurar un servicio de aprendizaje (machine learning) con Luis (Language Understanding Intelligent Service) que proporcione los modelos conceptuales necesarios para su integración con el prototipo chatbot.
- Realizar pruebas funcionales para detectar errores y posibles mejoras.
- Implementar y evaluar el chatbot en el área de fianzas de Seguros Equinoccial como caso de estudio.

1.3 CONCEPTOS DE IA RELACIONADOS A LOS CHATBOTS

En esta sección se da una breve descripción de la definición de algunas áreas de la Inteligencia Artificial (IA) que se relacionan al funcionamiento de un chatbot.

1.3.1 SISTEMAS EXPERTOS

Un SE puede definirse como un sistema informático (hardware y software), que simula a los expertos humanos en una área de especialización [2].

En la Figura 1, se ilustra un esquema de los principales componentes de un sistema experto.

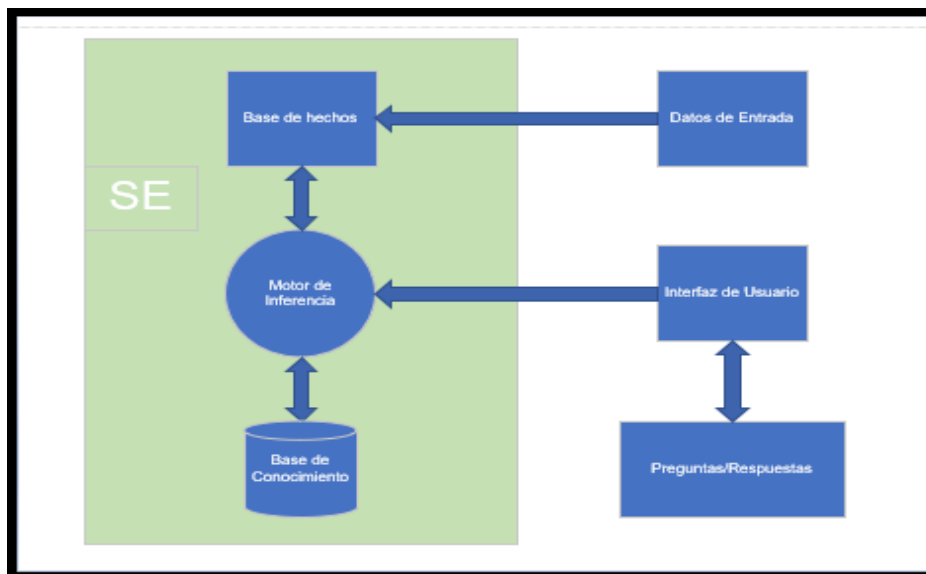


Figura 1. Ilustración de componentes de un sistema experto basado en [2].

En la ilustración se puede observar que un SE está conformado por un motor de inferencia, una base de hechos y una base de conocimiento, los cuales interactúan entre sí para obtener una solución a un problema de un dominio específico [2].

- Motor de Inferencia: es un programa que se encarga de construir una serie de razonamientos a partir de la información que se tiene en las bases de conocimientos y de hechos [2].
- Base de conocimiento: contiene una serie de datos en forma de reglas y hechos. Estos datos representan el conocimiento que el especialista o experto provee al sistema [3].

- Base de hechos: también conocida como memoria auxiliar o global, es la que almacena los modos de razonamiento que han sido utilizados en una situación concreta [2].

1.3.1.1 TIPOS DE SISTEMAS EXPERTOS

Existen dos tipos de SE: los deterministas o basados en reglas y los estocásticos o probabilísticos [3].

- SE basados en reglas: estos sistemas sacan sus conclusiones basándose en un conjunto de reglas utilizando un mecanismo de razonamiento lógico [3].
- SE probabilísticos: estos sistemas se basan en la probabilidad como una medida de incertidumbre para poder tratar situaciones en que la incertidumbre es lo común y no la excepción [3].

1.3.1.2 IMPORTANCIA DE LOS SISTEMAS EXPERTOS

La función principal de un SE es resolver problemas como lo haría un experto humano. Por esta razón, los sistemas expertos son útiles en distintos dominios, por ejemplo:

- Un SE puede ser utilizado para contestar preguntas y resolver problemas más rápido que un ser humano [3].
- Los SE pueden ser utilizados para realizar operaciones monótonas, aburridas e incómodas para los humanos [3].
- Con la ayuda de un SE, personal con poca experiencia puede resolver problemas que requieren el conocimiento de un experto [3].

1.3.2 MACHINE LEARNING

Es la construcción de algoritmos que pueden aprender y hacer predicciones en conjuntos de datos. Estos algoritmos operan mediante la construcción de un modelo, a partir de entradas de ejemplo, con el fin de realizar predicciones o elecciones basadas en datos; en lugar de seguir instrucciones fijas de un programa estático [4].

Machine Learning implica dos modelos de aprendizaje:

- Aprendizaje supervisado (Supervised machine learning): trabaja en un conjunto predefinido de ejemplos de entrenamiento, que luego facilitan su capacidad para llegar a una conclusión cuando recibe nuevos datos [4]. Algunos modelos

fundamentados en este tipo de aprendizaje son: redes neuronales artificiales, arboles de decisión, máquina de vectores, otros [5].

- Aprendizaje no supervisado (Unsupervised machine learning): trabaja con gran cantidad de datos, encontrando patrones y relaciones entre ellos para luego agruparlos [4]. Este tipo de aprendizaje se distingue del aprendizaje supervisado por el hecho de que no existe conocimiento a priori [5]. Algunos modelos fundamentados en este tipo de aprendizaje son: K-means, clustering, cadena de markov y análisis de componentes independientes [5].

1.3.3 PROCESAMIENTO DE LENGUAJE NATURAL

Procesamiento del lenguaje natural (PLN) se entiende como la habilidad de una máquina para procesar la información recibida en letras o sonidos del lenguaje [6]. El PLN consiste en la utilización del lenguaje natural para la comunicación del usuario con la computadora, debiendo ésta entender las oraciones que le sean proporcionadas [7].

1.3.3.1 ARQUITECTURA DE UN SISTEMA PLN

La arquitectura de un sistema PLN se sustenta en una definición del Lenguaje Natural (LN) por niveles [7]:

- Nivel Fonológico: trata de cómo las palabras se relacionan con los sonidos que las representan.
- Nivel Morfológico: trata de las palabras que se construyen a partir de unas unidades de significado más pequeño.
- Nivel Sintáctico: trata de como las palabras pueden unirse para formar oraciones.
- Nivel Semántico: trata del significado de las palabras, y cómo estos significados se unen para dar significado a una oración.
- Nivel Pragmático: trata de como las oraciones se usan en distintas situaciones, tomando distintos significados dependiendo de la situación.

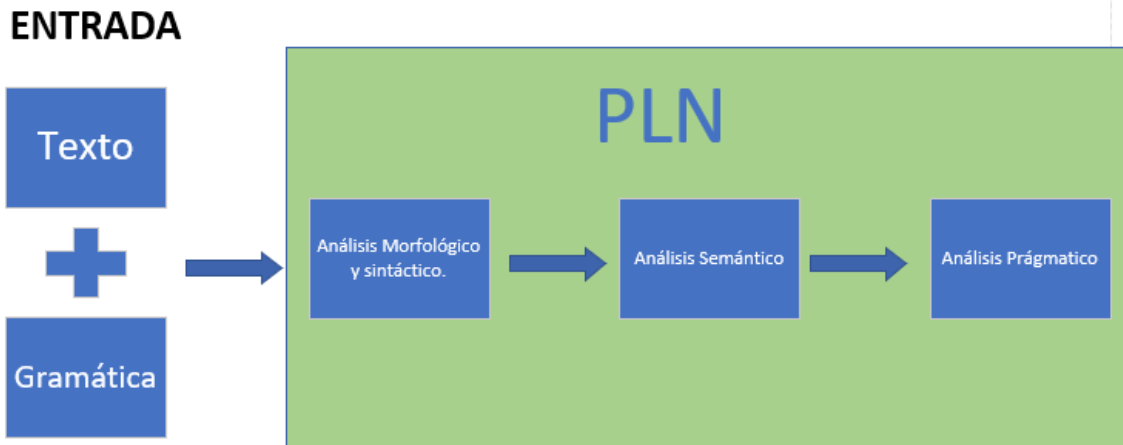


Figura 2. Ilustración de la arquitectura de un sistema de procesamiento de lenguaje natural, tomando como fuente [7].

Como se puede observar en la Figura 2, la entrada de información a un sistema PLN es una expresión en LN que el usuario quiere comunicar al sistema.

En el siguiente paso, el sistema utiliza un analizador lexicográfico y sintáctico. El primero, denominado *scanner*, se encarga de identificar los componentes léxicos definidos a priori. El segundo, denominado *parser*, se encarga de verificar si se cumple un orden gramatical entre los elementos identificados por el *scanner* [7].

A continuación, se realiza un análisis semántico para saber cuál es el significado de cada oración y asignar dicho significado a una expresión lógica (true o false) [7].

Por último, se hace un análisis pragmático de la información, es decir, el sistema toma todas las oraciones ya analizadas y, dependiendo de la situación, obtiene una expresión resultante, que es la que se envía al usuario [7].

1.3.3.2 APLICACIONES PLN

Un sistema de PLN puede aplicarse en distintos campos ya que puede combinarse con otros sistemas de IA para construir aplicaciones potentes a nivel cognitivo, algunas de las aplicaciones son [7]:

- Tutores inteligentes.
- Agentes Conversacionales (Chatbots).
- Respuestas automáticas.
- Análisis de sentimientos.
- Reconocimiento de Voz.

1.4 CHATBOTS O AGENTES CONVERSACIONALES

Un chatbot es un software que imita una conversación inteligente con un usuario, mediante el uso del LN [8]. Un chatbot integra técnicas de lingüística computacional para interpretar y responder declaraciones realizadas por los usuarios a través de Internet, canales de mensajería, emails, foros, etc. [9]. Además, permite crear un hilo conversacional con el usuario, dando respuestas en forma de mensajes cuyo origen está en una base de conocimiento [10].

Los chatbots son un nuevo tipo de interfaz de usuario que tienen la capacidad de comunicarse mediante el diálogo con un ser humano, dejando de lado la utilización de comandos, ventanas, iconos, etc. [11].

Estas características definen a un chatbot como un SE, que usa razonamiento basado en casos y simula un diálogo inteligente con el usuario [11].

1.4.1 RESEÑA HISTORICA DE LOS CHATBOTS

En 1950, Alan Turing, uno de los precursores de la informática moderna, propuso una forma de responder a la interrogante: "¿pueden pensar las máquinas?". Su método, que actualmente se lo conoce comúnmente como Test de Turing, consistía en un juego en el que participan dos personas y una máquina. Una de las personas actúa como juez y debe discernir quién de los restantes es la persona. La otra persona y la máquina no se encuentran en la misma habitación y la única forma de comunicarse entre el juez y los dos restantes es mediante preguntas escritas, las cuales son respondidas tanto por la persona como por la máquina. Turing sostuvo que si la máquina lograba engañar al jurado por un tiempo definido, se puede considerar que dicha máquina piensa [12].

El primer chatbot que se aproximó a los resultados deseados en el test de Turing fue ELIZA, que fue diseñado en 1966 por Joseph Weizenbaum. Este chatbot logró responder a interrogantes hechas a través de una consola de texto y confundir a una persona al punto de no saber si en verdad estaba hablando con una máquina. Pero estos resultados solo se lograron en las primeras líneas de diálogo y con frases determinadas [11].

En 1990, Hugh Loebner acordó, con el Centro de Estudios del Comportamiento de Cambridge, diseñar un certamen designado para implementar el Test de Turing [11]. Este evento impulsó el desarrollo de nuevas tecnologías y chatbots cada vez más sofisticados. Una de las tecnologías que surgió para el desarrollo de chatbots más

complejos fue AIML, que fue implementado en un chatbot conocido como ALICE y que salió vencedor del certamen en tres oportunidades [11].

A partir de la última década, se han alcanzado grandes hitos en el desarrollo de sistemas con IA.

En el 2014, un chatbot llamado Eugene Goostman logra ganar el test de Turing por primera vez en la historia. Este bot se hacía pasar por un chico ucraniano de 13 años y consiguió convencer al 33% de los jueces de que era un ser humano [14]. Este suceso también puso en tela de duda de que la prueba de Turing no sea suficiente para definir la inteligencia de un programa. Referente a este tema, Evan Ackerman menciona:

“El problema con la prueba de Turing es que realmente no demuestra si un programa de IA es capaz de pensar, más bien indica si un programa de IA puede engañar a un ser humano. Y los seres humanos somos realmente tontos. Caemos en toda clase de trampas que un programa bien hecho puede utilizar para convencernos de que estamos hablando con una persona capaz de pensar”[15].

Basándose en estas palabras, Manuel Alfonsea, profesor honorario de la Universidad Autónoma de Madrid menciona: “no basta con que un ordenador sea capaz de pasar la prueba de Turing para que podamos considerarlo inteligente como un ser humano. Hacen falta dos cosas más: que el ordenador comprenda lo que escribe, y que sea consciente de la situación” [14].

Aunque a la IA todavía le falta mucho por recorrer, los desarrollos de nuevas tecnologías y avances en este campo no paran.

En 2016, Microsoft indicó que han creado un sistema de reconocimiento de voz humana que puede reconocer una conversación, como lo realizan las personas, logrando vencer la barrera del reconocimiento de voz[16]. En el mismo año, Mark Zuckerberg anunció la disponibilidad de Messenger Platform, que permite a las organizaciones crear chatbots que proporcionan algún tipo de servicio a sus usuarios. Zuckerberg mencionó que los bots son una forma más inmediata de comunicación entre las personas y las organizaciones [17].

En la actualidad, las grandes empresas de tecnología Google DeepMind, Microsoft Cortana e IBM Watson anuncian innovaciones en reconocimiento de voz, traducción de texto a voz, aprendizaje automático, visión por computadora, y una serie de otras tecnologías que, en conjunto, pretenden dar una interacción más naturalista e inteligente entre usuario y máquina [17].

Con esta breve reseña histórica se puede constatar que los científicos han estado experimentando con IA por casi 70 años y la construcción de bots por más de 50 años. [17] (Figura 3).

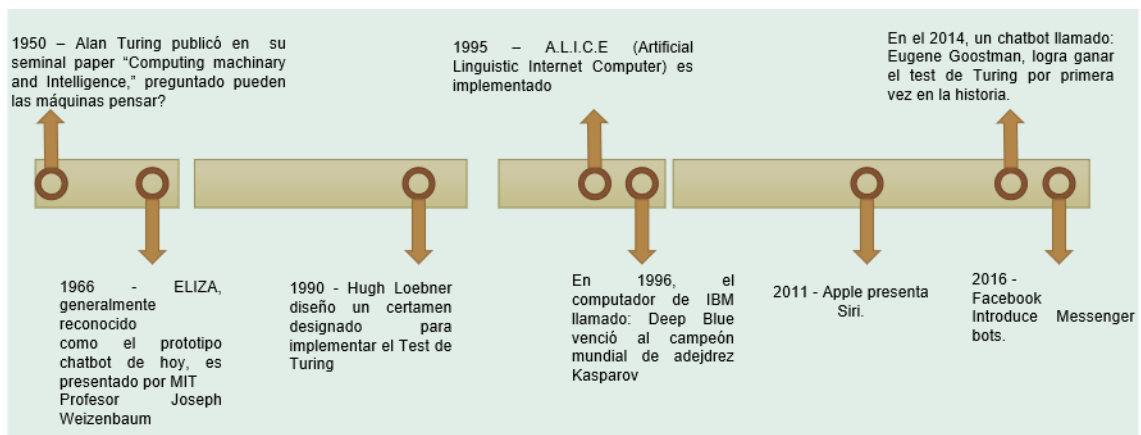


Figura 3. Ilustración de algunos eventos en la historia de la IA, recurso fundamentado de [11] [12] [13] [14] [17].

1.4.2 CARÁCTERÍSTICAS DE UN CHATBOT

Un chatbot posee características que le permiten desempeñar múltiples funciones, entre las más relevantes están las siguientes:

- **Adaptabilidad:** esta característica se relaciona con la capacidad de aprendizaje que puede tener un chatbot, cambiando su comportamiento en función de lo aprendido [9].
- **Expresividad:** esta característica aprovecha los elementos que poseen las aplicaciones de mensajería u otros tipos de canales para poder utilizar gráficos, videos y sonidos con la finalidad de que el diálogo sea más expresivo [18].
- **Racionalidad:** el chatbot siempre intenta dar la respuesta más apropiada a partir de los datos que percibe del entorno [9].
- **Proactividad:** un chatbot es capaz de exhibir un comportamiento dirigido a un objetivo, al tomar la iniciativa en una conversación [1].
- **Personalizable:** un agente conversacional puede tener la cantidad de características que su programador desee [9]. Un chatbot puede mostrar emociones, interpretar sentimientos, intenciones, o tener un comportamiento no verbal. Estas características y otras más pueden ser añadidas dependiendo de las necesidades que el programador quiera cubrir con un chatbot [9].

1.4.3 ARQUITECTURA DE UN CHATBOT

La arquitectura básica de un chatbot se divide en tres componentes, como se observa en la Figura 4.

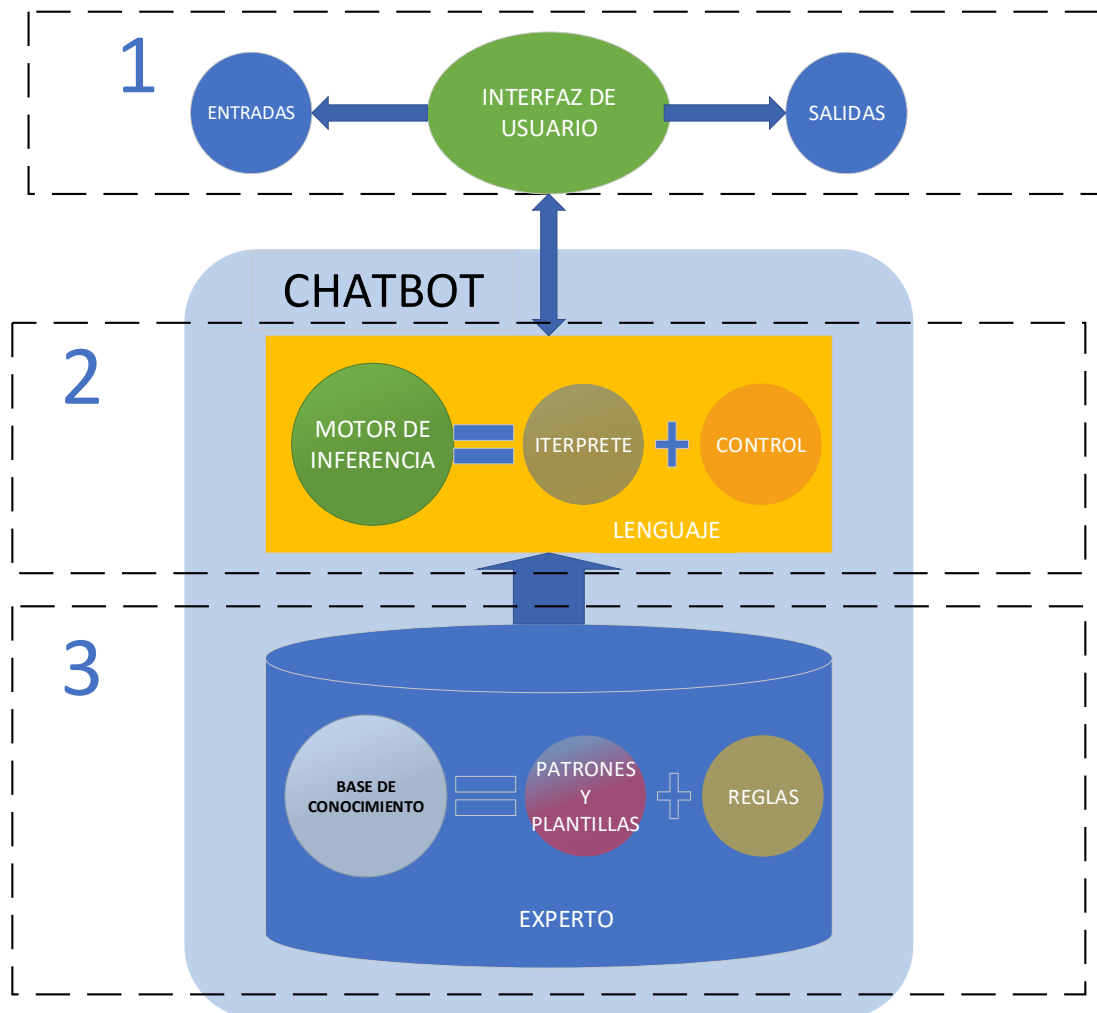


Figura 4. Ilustración de la arquitectura básica de un chatbot, basado en [9].

- 1) Interfaz de Usuario: es el medio por el cual el usuario envía las entradas de información hacia el chatbot.
- 2) Motor de inferencia: analiza la información, establece los objetivos y obtiene la respuesta de acuerdo con la base de conocimiento, enviándola de vuelta a la interfaz de usuario [9]. Un motor de inferencia utiliza dos elementos: los datos (hechos o evidencias) y el conocimiento (conjunto de reglas almacenado en la base de conocimiento) para obtener nuevas conclusiones o hechos [20].
- 3) La Base de conocimiento: contiene todo el conocimiento del experto humano, en función de plantillas, patrones y reglas [9].

En la Figura 5, se puede observar más detalladamente el funcionamiento básico de un chatbot.

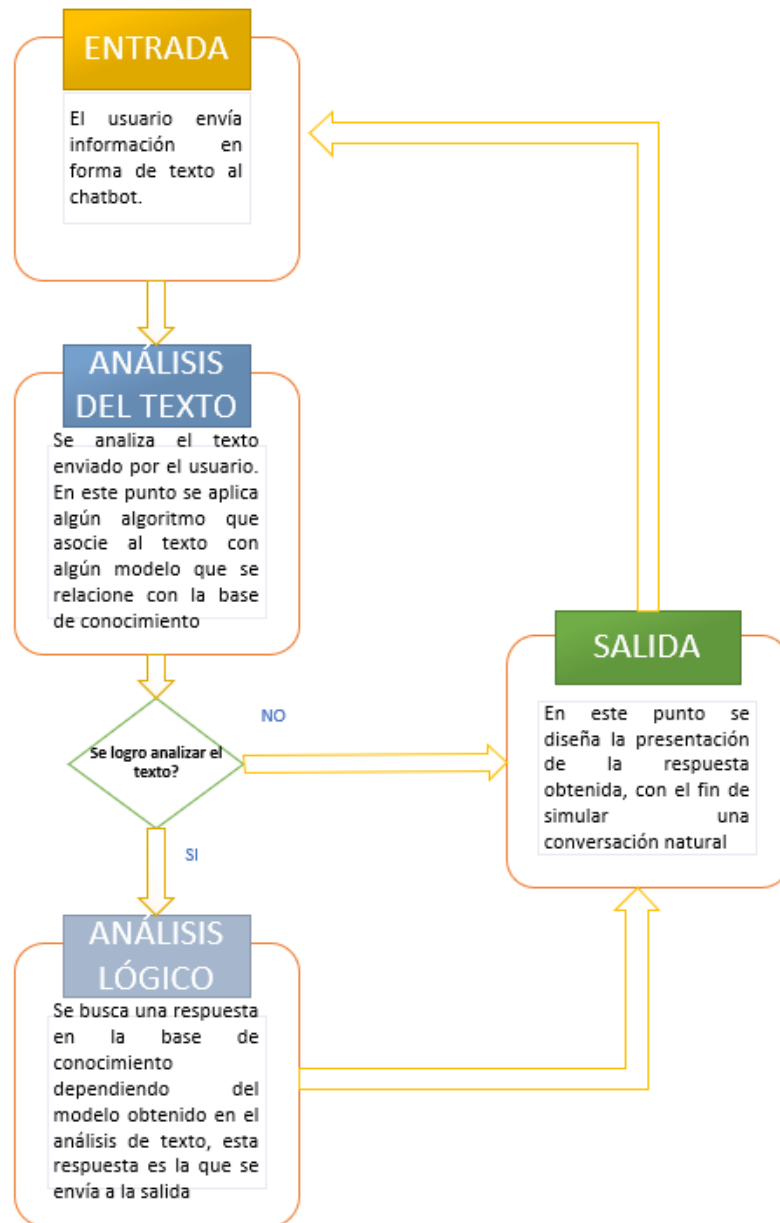


Figura 5. Ilustración del funcionamiento básico de un chatbot, fundamentado en [9].

1.4.4 CATEGORÍA DE CHATBOTS

Se categorizan los chatbots según el servicio que provee, diseño de interfaz y tecnología que usa [21]:

1.4.4.1 CHATBOTS SEGÚN EL TIPO DE SERVICIO

- Chatbots Operativos o Empresariales: son bots que actúan como ayudantes o facilitadores de servicios de una organización. Estos bots son usados para agilizar los procesos y mejorar los tiempos de respuesta.
- Chatbots Informativos: son bots que cumplen funciones específicas de poca complejidad, por ejemplo, un bot que utiliza un sistema de pregunta-respuesta QA para responder a preguntas frecuentes. Estos bots también son conocidos como frequent asked question FAQ bot [21].
- Chatbots e-commerce: conocidos como asistentes virtuales comerciales, son bots cuyo propósito es ayudar a los usuarios con la compra de algún producto [19].

1.4.4.2 CHATBOTS SEGÚN SU DISEÑO DE INTERFAZ

- Chatbots con interfaz solo texto: son bots que interactúan con los usuarios solo a través de entradas y salidas de texto [21].
- Chatbots con interfaz combinada entre texto, botones e imágenes: son bots que interactúan con los usuarios mediante diálogos enriquecidos complejos que contienen texto, imágenes y botones de acción. Esta característica permite a los usuarios tener una interacción más guiada con el bot [21].

1.4.4.3 CHATBOTS SEGÚN LA TECNOLOGÍA UTILIZADA

- Chatbots Simple: son bots sencillos cuyo funcionamiento se basa en una coincidencia de patrones básicos con una respuesta [21].
- Chatbots Complejos: estos bots son más sofisticados ya que utilizan técnicas de IA con un complejo seguimiento del estado conversacional y la integración a servicios empresariales existentes [21].

1.4.5 APLICACIONES DE LOS CHATBOTS

La ventaja de poder integrarse en distintos canales de mensajería, además de su disponibilidad y su fácil usabilidad, hacen que los chatbots puedan incorporarse en cualquier sector [19]. En la Figura 6 se pueden observar algunos campos en los que puede ser utilizado un chatbot. Estos son solo una muestra de la diversidad de campos en los que puede ser utilizado un chatbot.



Figura 6. Ilustración de campos de aplicación de los chatbots, fundamentado en [9,17].

1.4.6 CASOS DE ÉXITO

Actualmente las empresas están cada vez más interesadas en la implementación de chatbots. En el artículo “THE CONVERSATIONAL BUSINESS” de Susan Eltinger, se mencionan algunos casos de éxito de empresas que incursionaron en la aplicación de agentes conversacionales en sus negocios. Uno de los casos mencionados es el de Mastercard, que creó su propio chatbot, “MasterCard KAI”, cuya función es facilitar la compra en Facebook Messenger con Subway FreshDirect y The Cheesecake Factory [17]. Ver Figura 7.

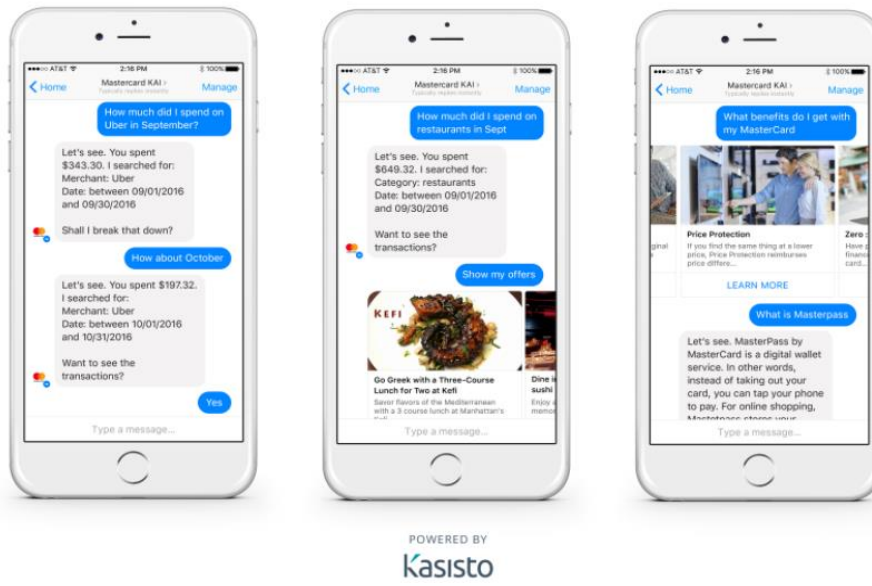


Figura 7. Ilustración de chatbot creado por Mastercard, recurso tomado de [22].

Otro caso de éxito mencionado en el artículo “THE CONVERSATIONAL BUSINESS” es el de BBC Earth. El chatbot desarrollado por la BBC Earth envía un video personalizado basado en el contenido de BBC Earth después de que el usuario responde algunas preguntas y da su consentimiento [17]. Actualmente BBC Mundo tiene su propio chatbot en Facebook, el cual proporciona información de noticias a sus usuarios, ver Figura 8.



Figura 8. Ilustración del chatbot de BBC Mundo, recurso tomado de [23].

En el caso de seguros existe, un chatbot llamado Billy Seguros, creado por Chatbot Chocolate, que es una agencia de diseño, desarrollo y entrenamiento de chatbots para empresas [24]. Ver Figura 9.

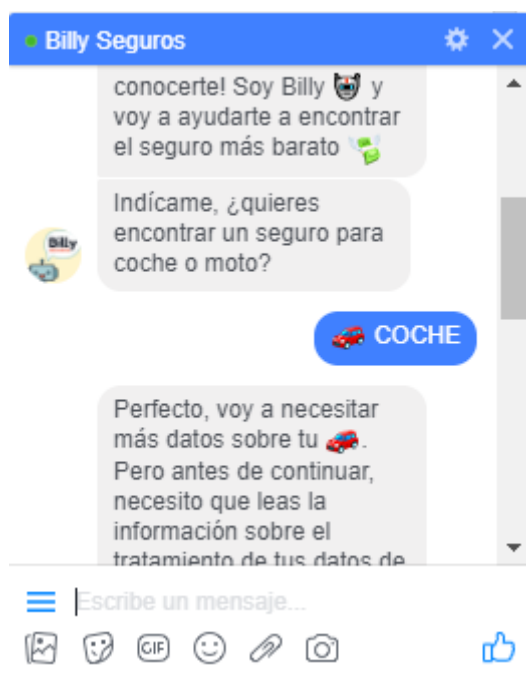


Figura 9. Ilustración chatbot Billy Seguros, recurso tomado de [25].

Múltiples empresas están incursionando en el mundo de los chatbot, incluso hay empresas que se dedican solo al desarrollo de estos. Los agentes conversacionales están en auge y pueden representar una nueva era en la interacción entre humanos y máquinas.

Tomando como ejemplo estos casos de éxito, el chatbot desarrollado en este proyecto muestra funcionalidades similares, pero con características propias basadas en el modelo de negocio del caso de estudio. Este chatbot funciona con dos servicios cognitivos que le permiten simular una conversación natural. Además, interactúa con servicios propios del caso de estudio, lo cual le permite dar un servicio personalizado y transaccional en algunos de sus módulos.

1.4.7 TECNOLOGÍA PARA EL DESARROLLO DE CHATBOTS

En la actualidad, las grandes empresas de software ven a las interfaces conversacionales como una evolución del uso del software a través de internet, ofreciendo soluciones que proporcionen una infraestructura que facilite el desarrollo de aplicaciones basada en IA [26].

Existen varias herramientas para la creación de chatbots, las mayorías de estas cobran valor dependiendo del tipo de uso que se le dé al chatbot. Entre las herramientas más conocidas están:

- Microsoft Bot Framework: framework de Microsoft para construir y administrar bots en distintos canales con la posibilidad de integrar varios servicios cognitivos de azure [21].
- IBM Watson Conversation: framework de IBM para crear y desplegar chatbots en varios canales, dispositivos móviles e incluso robots. Este framework tiene un kit de desarrollo basado en los servicios cognitivos de Watson e IBM cloud [27].
- Wit.ai: Servicio web disponible para que los desarrolladores puedan crear aplicaciones y dispositivos con los que se pueda hablar o enviar mensajes de texto para la plataforma Facebook [28].
- Amazon Lex: servicio de Amazon que permite crear chatbots conversacionales. Pone las tecnologías de aprendizaje profundo de Amazon Alexa en manos de cualquier desarrollador [29].

Como se puede ver, existen varias herramientas que se pueden utilizar para crear un chatbot. En este proyecto para la creación del chatbot se utilizó Bot Framework de Microsoft. La razón de utilizar este Framework es la posibilidad de utilizar la infraestructura proporcionada por el caso de estudio.

1.4.7.1 BOT FRAMEWORK

Bot Framework proporciona un entorno integrado y diseñado especialmente para el desarrollo de bots, permitiendo construir, conectar, probar, implementar y administrar bots inteligentes con soporte para .Net y Node.js mediante el SDK de Bot Builder [21].

1.4.7.1.1 BOT BUILDER

Bot Builder proporciona SDK's, bibliotecas, muestras y herramientas con la finalidad de construir y depurar bots tanto en C# como en Node.js. Para poder examinar el funcionamiento de los bots, BotBuilder proporciona: (1) un emulador (Bot Framework Emulator) para hacer pruebas de desarrollo y (2) el Channel Inspector para evaluar la experiencia de usuario de los bots en diferentes canales [21].

1.4.7.1.2 CANALES QUE SE PUEDEN UTILIZAR CON BOTFRAMEWORK

Bot Framework proporciona la posibilidad de usar distintos canales como GroupMe, Facebook Messenger, Kik, Skype, Slack, Microsoft Teams, Telegram, texto / SMS,

Twilio, Cortana y Skype, para que los usuarios puedan comunicarse con los bots, aumentando la cantidad de interacciones que puede darse entre los usuarios y bots [21].

1.4.7.1.3 BOT CONNECTOR

Bot Conector permite conectar a un bot con uno o más canales a través de un API REST, permitiendo que el esquema de mensajes sea el mismo sin importar qué canal se esté utilizando [21]. Ver Figura 10.

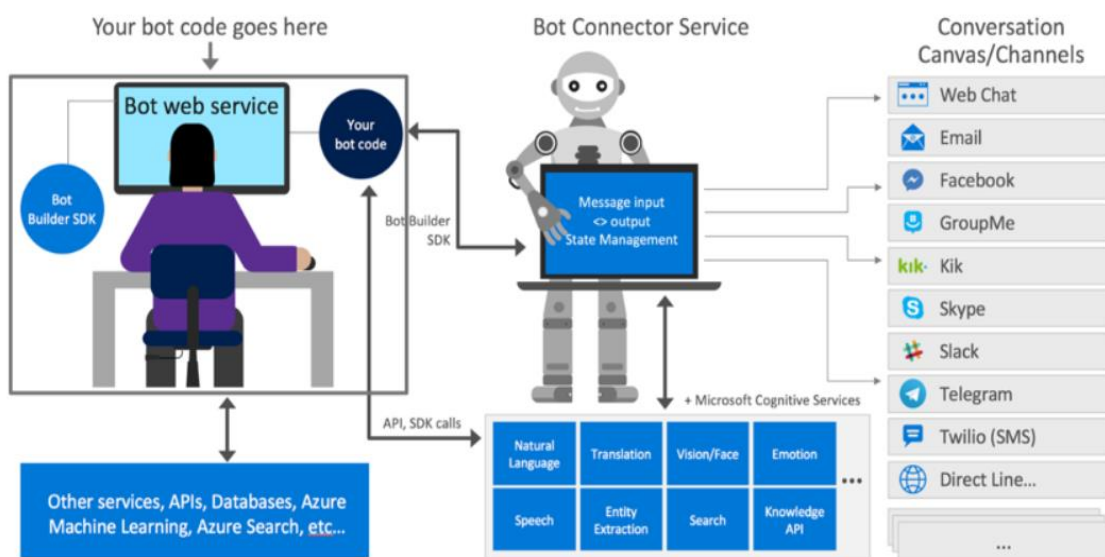


Figura 10. Ilustración de la arquitectura del Bot Framework, recurso tomado de [21].

Como se observa en la Figura 10, la arquitectura del Bot Framework se divide en tres bloques. En el primer bloque están los Canales de conversación, en los cuales intervienen las entradas y salidas del diálogo entre el bot y el usuario [21].

En el segundo bloque se encuentra el Bot Connector. El servicio se encarga del envío y recepción de mensajes entre el usuario y el bot. El bot conector también puede conectarse con servicios cognitivos de IA y recopilar información sobre el estado del servicio [21].

La comunicación entre Bot web service, el Bot conector y el canal que usa el usuario se da a través de REST, por lo cual toda la información que se intercambia entre el usuario y el bot es convertido a Json. Este funcionamiento permite que la comunicación se dé sin importar que canal se esté utilizando [21].

En el tercer bloque se encuentra el código del bot, que es básicamente la funcionalidad y la lógica que el desarrollador le da al bot. En este punto, el código del bot se conecta

con el Bot Builder SDK, que es la librería que se encarga de gestionar la conversación. Además, el código del bot puede conectarse con otros servicios, lo mismo propios de una empresa o servicios que estén en Azure [21].

1.4.7.1.4 LA SEGURIDAD CON BOT FRAMEWORK

Para garantizar que Bot Connector acceda desde distintos canales al servicio del chatbot, se utiliza una configuración basada en el protocolo https a través de un canal seguro (SSL/TLS) y autenticación. Esta configuración se realiza al momento de registrar un chatbot en la plataforma Bot Framework. Bot Connector especifica un token de acceso en la cabecera de cada solicitud. Por lo cual, la comunicación es permitida únicamente si se tiene el token de acceso [21].

1.4.7.1.5 TECNOLOGÍAS DE AUTENTICACIÓN

Se usan cuatro tecnologías de autenticación para establecer la conexión entre un bot y el Bot Connector [21]. Ver Tabla 1.

TECNOLOGÍA	DESCRIPCIÓN
SSL / TLS	SSL / TLS se usa para todas las conexiones de servicio a servicio. Los certificados se usan para establecer la identidad de todos los servicios HTTPS.
Oauth 2.0	El inicio de sesión de Oauth 2.0 en el servicio de inicio de sesión de Cuenta de Microsoft (MSA) / AAD v2 se utiliza para generar un token seguro que un bot puede usar para enviar mensajes.
Token web JSON (JWT)	Los tokens web JSON se utilizan para codificar tokens que se envían desde y hacia el bot.
Metadatos OpenID	El servicio Bot Connector publica una lista de tokens válidos que utiliza para firmar sus propios tokens JWT a los metadatos OpenID en un punto final estático bien conocido.

Tabla 1. Cuadro de tecnologías de autenticación que usa Bot Framework, recurso tomado de [21].

1.4.8 SERVICIOS COGNITIVOS DE MICROSOFT

Microsoft Cognitive Services (anteriormente Project Oxford) es un conjunto de APIs, SDK y servicios que ayudan a los desarrolladores a crear aplicaciones más inteligentes, atractivas y reconocibles [21].

Entre el conjunto de APIs que Microsoft ofrece en sus Servicios Cognitivos están: Computer Vision, Face Api, QnaMaker, Language Understanding (LUIS) que son muy útiles para el desarrollo de chatbots. En el presente proyecto se utilizan las siguientes APIs: QnaMaker y LUIS [21].

1.4.8.1 QNAMAKER

Microsoft QnA Maker es una API REST gratuita y fácil de usar, y un servicio basado en la web, que entrena a una IA para responder a las preguntas de los usuarios de una manera más natural y conversacional [30]. Es un servicio basado en un sistema de QA con la capacidad de crear bases de conocimiento de una manera simple, ya que permite extraer pares de preguntas y respuestas de entradas como páginas web y documentos fuera de línea (.docx, .doc, .pdf, .xlsx y .tsv) que contengan la estructura y semántica de QnA [30].

La capacidad de crear una base de conocimiento proveniente de varias fuentes proporciona una mayor probabilidad de que un chatbot tenga la respuesta a alguna pregunta enviada por el usuario, como se puede observar en la Figura 11.

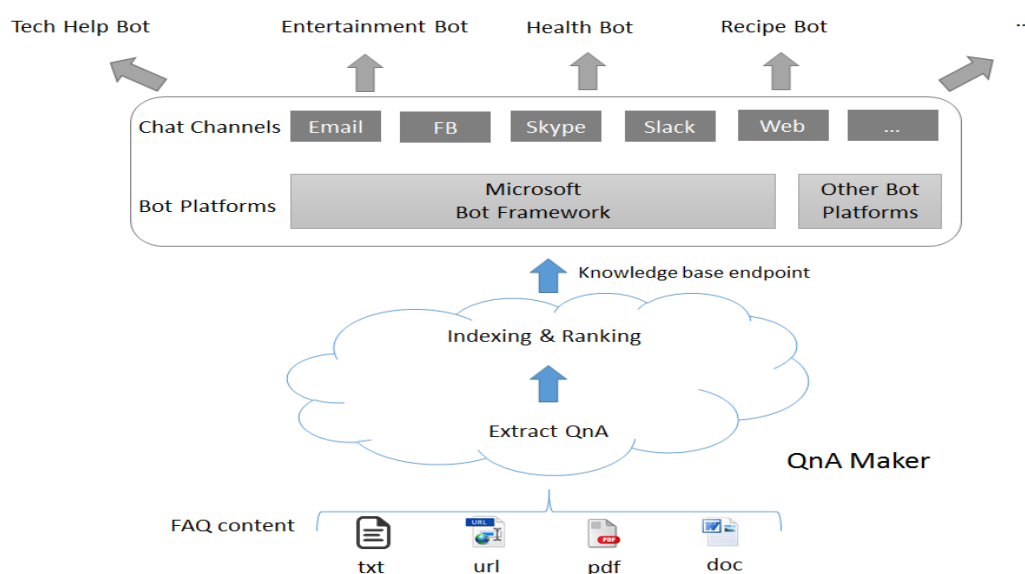


Figura 11. Ilustración de funcionamiento QnA Maker Api, recurso tomado de [30].

1.4.8.2 LANGUAGE UNDERSTANDING INTELLIGENT SERVICE (LUIS)

LUIS es un servicio cognitivo basado en Machine Learning para crear una comprensión lingüística natural en aplicaciones, bots y dispositivos IoT (Internet de las cosas). LUIS utiliza el aprendizaje automático para permitir a los desarrolladores crear aplicaciones que puedan recibir las entradas de los usuarios en un lenguaje natural y extraer su significado [31].

La configuración de este servicio se fundamenta en la definición de intenciones que pertenecen a un dominio específico. Por ejemplo, si el dominio se enfoca al negocio de seguros, una intención sería “Realizar una solicitud de Póliza” o “Consultar pólizas por vencer”. Si el dominio se enfoca a un servicio de vuelos, una intención sería “Reservar vuelo” o “Consultar vuelos disponibles”. Todas las intenciones definidas tienen sus declaraciones o expresiones que básicamente son oraciones que se relacionan con la intención definida. Por ejemplo, en el caso de seguros, si se quiere realizar una solicitud de póliza, una expresión sería “Deseo realizar una solicitud de póliza” [31].

Al crear una expresión, se pueden incluir Entidades que definen una información específica y que ayudan a que las expresiones empleadas sean más precisas. LUIS tiene incorporado entidades pre-construidas como número, edad, porcentaje, temperatura y fecha, que ayudan a crear expresiones más detalladas. LUIS también da la posibilidad de crear nuevas entidades, lo cual permite tener expresiones con entidades personalizadas según el criterio del desarrollador [31].

Tomando como dominio a una empresa de seguros, un ejemplo de expresión con una entidad sería: “Quiero consultar la póliza número 1234”, en la cual el valor 1234 representa la entidad número. Este dato puede ser utilizado por una aplicación para realizar algún tipo de operación.

Un ejemplo de creación de intenciones en LUIS se puede observar en la Figura 12.

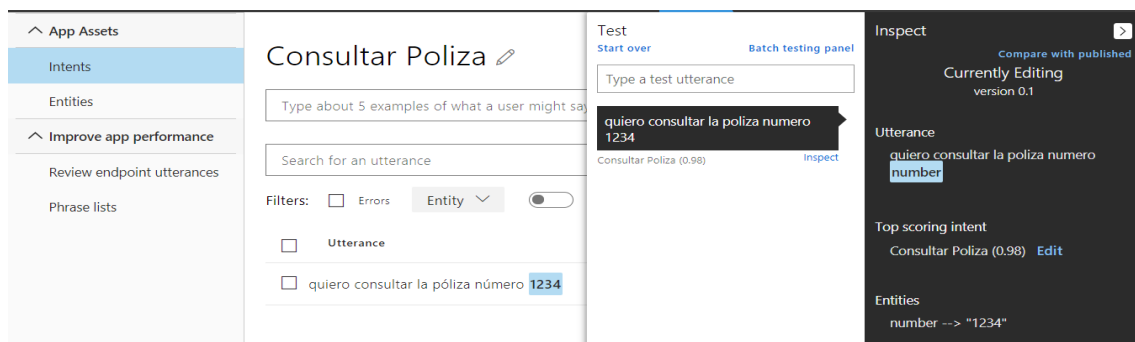


Figura 12. Ilustración funcionamiento de Language Understanding Intelligent Service (LUIS) recurso elaborado por el autor.

Continuando con el documento, en la Sección II se detalla la metodología empleada en el desarrollo e implementación del chatbot. En la Sección III, se muestra un análisis de resultados y discusiones obtenidas en el desarrollo del proyecto y de los cuales se fundamentan las conclusiones presentadas en la Sección IV. En la Sección V, se presentan las referencias bibliográficas y en la Sección VI, los anexos necesarios.

2. METODOLOGÍA

2.1 METODOLOGÍA DE DESARROLLO

En el desarrollo de la aplicación (chatbot) es conveniente que los módulos sean funcionales y presentados en el menor tiempo posible con la finalidad de ser evaluados y tener una retroalimentación constante durante el desarrollo de este proyecto [32]. Además, se necesita que el desarrollo sea interactivo e incremental para que las soluciones y requerimientos evolucionen con el tiempo [32]. Tomando en cuenta las características de desarrollo mencionadas, se utilizará el framework de desarrollo SCRUM.

2.1.1 SCRUM

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente en equipo y obtener el mejor resultado posible de un proyecto [32].

Las razones para usar un enfoque de desarrollo interactivo e incremental usando SCUM son [32]:

- Modularidad, esta característica se basa en incrementar o modificar funcionalidades a partir de una funcionalidad ya creada.
- Gestión regular de las expectativas del cliente y basada en resultados tangibles, esta característica se fundamenta en la entrega de módulos completos en tiempos mínimos permitiendo la mejora continua.
- Flexibilidad y adaptación, esta característica se basa en poder atender a los requerimientos cambiantes del cliente durante el desarrollo del proyecto.

2.1.1 ROLES DEL PROYECTO

Los roles principales que se definen en el proyecto son [32]:

- Equipo de desarrollo, grupo de personas que de manera conjunta desarrollan el producto del proyecto.
- Dueño del Producto (Product Owner), es el responsable del Product Backlog y del éxito del proyecto.
- Scrum Máster, es el responsable de guiar al equipo para que sigan los principios ágiles, reglas y procesos de Scrum.

PERSONA	ROL
Msc. Maritzol Tenemaza	Scrum Master
Ruben Sanchez	Product Owner
Omar Zarabia	Equipo de Desarrollo

Tabla 2. Roles del Proyecto (Fuente: elaborado por el autor).

2.1.2 ARTEFACTOS

En Scrum se tiene la siguiente documentación como artefactos [32]:

- Pila de productos (Product Backlog), es una lista de requisitos priorizada. Representa la visión y expectativas del cliente respecto a los objetivos y entregas del producto o proyecto.
- Pila de sprint (Sprint Backlog), es una lista de tareas o ítems que el equipo de desarrollo elabora para cumplir los objetivos de cada Sprint.
- Gráfico de Trabajo Pendiente (Burndown Chart), es un gráfico de trabajo pendiente a lo largo del tiempo. Muestra la velocidad a la que se están completando los objetivos/requisitos.

2.2 ARQUITECTURA DEL PROYECTO

La arquitectura utilizada en este proyecto es Modelo Vista Controlador (MVC). Esta arquitectura permite crear sistemas robustos con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos. Su fundamento es la separación del código en tres capas diferentes [33].

- **Modelo**, “Es la capa donde se trabaja con los datos, por tanto, contendrá mecanismos para acceder a la información” [33].
- **Vista**, “Contienen el código de la aplicación que va a producir la visualización de las interfaces de usuario” [33].
- **Controlador**, “Es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de la aplicación” [33].

En la Figura 13, se puede observar la interacción de las capas de la arquitectura MVC del proyecto.

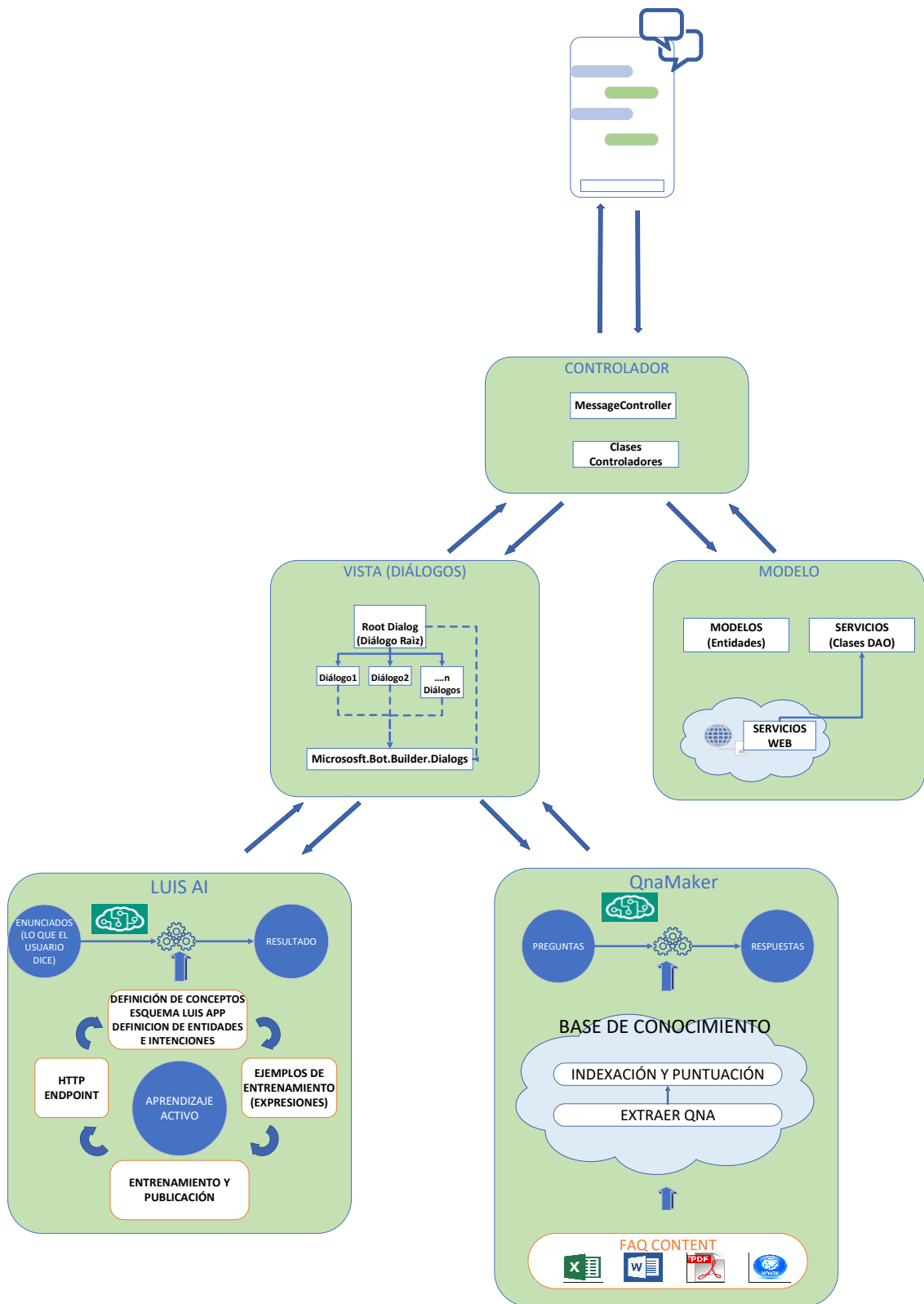


Figura 13. Arquitectura MVC del proyecto (Fuente: elaborado por el autor y fundamentado en [21,30,31]).

En una aplicación tradicional, la interfaz de usuario (UI) es una serie de pantallas. Una sola aplicación o sitio web puede usar una o más pantallas según sea necesario para intercambiar información con el usuario. Al igual que las aplicaciones y los sitios web, los bots tienen una UI la cual está formada por cuadros de diálogo en lugar de pantallas. Los cuadros de diálogo permiten al desarrollador del bot separar lógicamente varias áreas de la funcionalidad del bot y guiar el flujo conversacional [31].

2.3 HERRAMIENTAS DE DESARROLLO

El proyecto será desarrollado con el uso de herramientas Microsoft para poder utilizar la infraestructura que posee el caso de estudio.

Entre las herramientas utilizadas están el Framework de desarrollo de chatbots (Bot Framework) y los servicios cognitivos Microsoft QnaMaker y LUIS cuyas definiciones se encuentran en la Sección I.

Para la creación de los servicios web que van a ser consumidos por el chatbot, se utilizan dos lenguajes de programación: C# y visual basic. Además, utilizan dos gestores de bases de datos: Sql Server y Sybase.

Finalmente, se necesita de un servidor web (INTERNET INFORMATION SERVICE), el cual sirve para hospedar la aplicación con la respectiva configuración que le permita funcionar correctamente.

2.4 DESARROLLO DEL SISTEMA

2.4.1 DEFINICIÓN DEL PROYECTO

2.4.1.1 REQUERIMIENTOS

El sistema (chatbot) requiere de los siguientes módulos:

- Módulo para solicitudes de pólizas nuevas.
- Módulo de consultas de estados de cuenta y pólizas por vencer de los clientes.
- Módulo de preguntas frecuentes sobre fianzas y seguros.
- Módulo para renovaciones de pólizas.

El chatbot debe estar disponible en Facebook, embebido en la página web del caso de estudio y en Skype (opcional). Además, se requiere que la interacción con el usuario sea guiada para facilitar el proceso de cada módulo.

La interacción con el chatbot debe ser basada en intenciones. En la Figura 14, se muestra la arquitectura basada en intenciones de este chatbot, en la cual se observa la interacción con los servicios cognitivos que hacen posible cumplir este requerimiento.

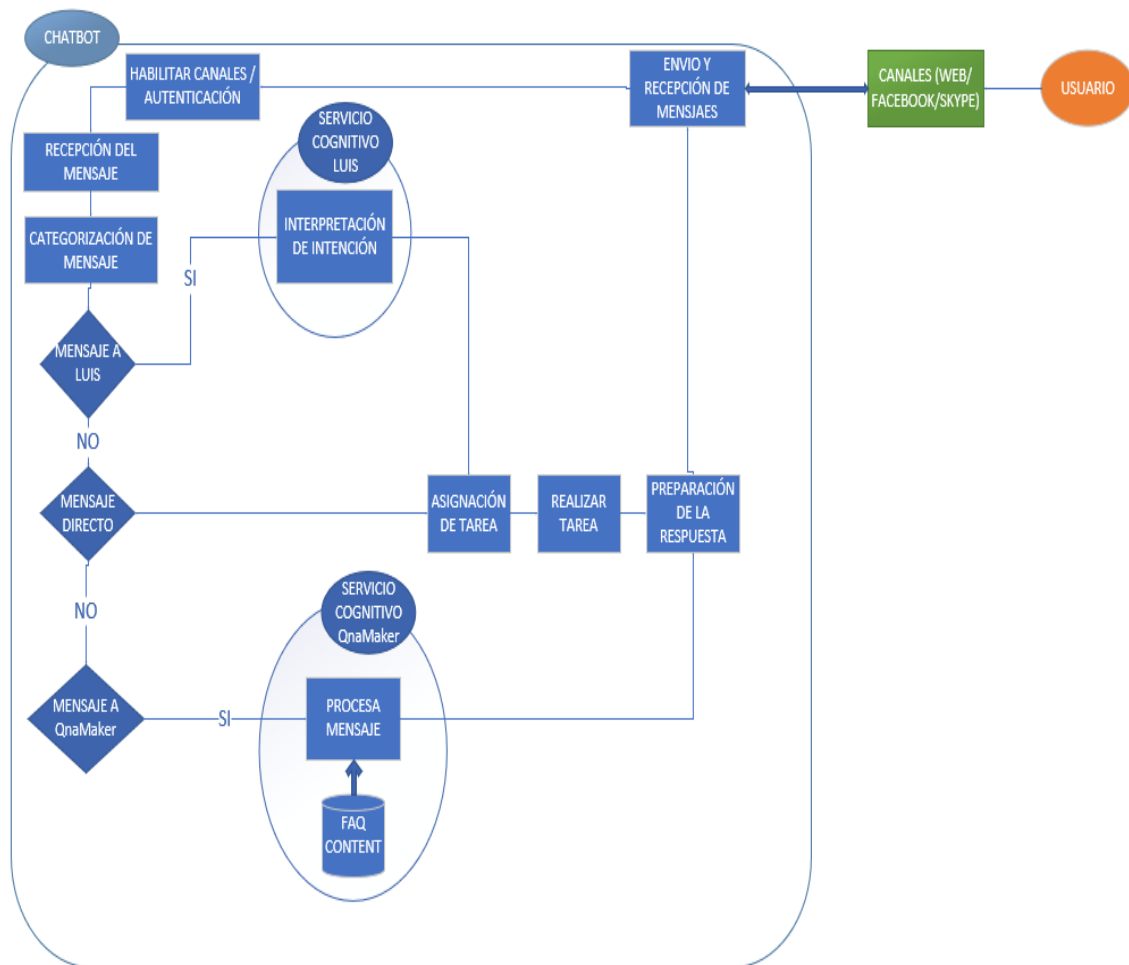


Figura 14. Arquitectura basada en intenciones del chatbot (Fuente: elaborado por el autor)

2.4.1.2 HISTORIAS DE USUARIO ÉPICAS

De acuerdo con los requerimientos planteados, se definen las siguientes historias de usuario épicas:

HISTORIA DE USUARIO ÉPICA	HUE01
Título: Realizar la solicitud de una póliza nueva	
Descripción: Como usuario quiero poder realizar una solicitud de póliza, ingresando toda la información requerida para que se efectúe la solicitud.	

Tabla 3. Historia épica HUE01 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO ÉPICA	HUE02
Título: Consultar Estados de cuenta y pólizas por vencer	
Descripción: Como usuario quiero poder buscar y visualizar los estados de cuenta pólizas por vencer de mis clientes.	

Tabla 4. Historia épica HUE02 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO ÉPICA	HUE03
Título: Consultar información con respecto a Fianzas y Seguros	
Descripción: Como usuario quiero poder preguntar al chatbot sobre temas correspondientes a fianzas o seguros y obtener una respuesta de parte de este.	

Tabla 5. Historia épica HUE03 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO ÉPICA	HUE04
Título: Realizar la renovación de una póliza	
Descripción: Como usuario quiero poder realizar una renovación de póliza, ingresando toda la información requerida para que se efectúe esta operación.	

Tabla 6. Historia épica HUE04 (Fuente: elaborado por el autor).

2.4.1.3 PRODUCT BACKLOG

Las historias de usuario épicas descritas se subdividirán en historias de usuario específicas, como se puede observar en la siguiente tabla.

PRODUCT BACKLOG				
EU ÉPICA	CÓDIGO HU	HISTORIA DE USUARIO	PRIORIDAD	DURACIÓN
HUE01	HU01	Consultar Datos de clientes	alta	1 día
	HU02	Consultar Datos de beneficiarios de la póliza	alta	1 día
	HU03	Llenar información de la póliza	alta	1 día
	HU04	Escoger tipos de póliza	alta	2 días
	HU05	Adjuntar Documentos requeridos	alta	3 días
	HU06	Procesar solicitud de póliza	alta	5 días
	HU07	Visualizar Detalle de Póliza Generada	media	1 día
HUE02	HU08	Consultar Datos de clientes	alta	1 día
	HU09	Consultar y visualizar estados de cuenta del cliente	alta	3 días
	HU10	Consultar y visualizar pólizas por vencer de los clientes	alta	3 días
HUE03	HU11	Tener contenido referente a Fianzas y Seguros	media	2 días
	HU12	Preguntas y respuestas de Fianzas y Seguros	media	1 día
HUE04	HU13	Consultar y visualizar la póliza a renovar	alta	2 días
	HU14	Renovar póliza	alta	10 días

Tabla 7. Product Backlog (Fuente: elaborado por el autor).

Cabe mencionar que las dos primeras historias de usuario HU01 y HU02 corresponden a la configuración del servidor y base de datos. Estas historias de usuario pertenecen al Sprint 0, y no se encuentran en el Product Backlog.

2.4.1.4 REALEASE PLANNING

Una vez definido el Product Backlog, se procede a elaborar el Release Planning donde se asignan los Sprints con sus respectivas actividades, como se puede observar en la siguiente tabla.

RELEASE PLANNING				
SPRINT 0	SPRINT 1	SPRINT 2	SPRINT 3	SPRINT 4
HU01	HU03	HU10	HU13	HU15
HU02	HU04	HU11	HU14	HU16
	HU05	HU12		
	HU06			
	HU07			
	HU08			
	HU09			

Tabla 8. Release Planning (Fuente: elaborado por el autor).

2.4.2 EJECUCIÓN DE SPRINTS

2.4.2.1 SPRINT 0

2.4.2.1.1 SPRINT PLANNING

2.4.2.1.1.1 OBJETIVO

Configurar el servidor IIS y preparación de tablas y Scripts en bases de datos (SQL, Sybase).

2.4.2.1.1.2 HISTORIAS DE USUARIO DEL SPRINT

HISTORIA DE USUARIO		HU01
Título: Configurar IIS para poner en funcionamiento el chatbot		
Descripción: Como usuario, quiero un servidor web que permita al chatbot funcionar de manera optima		
Prioridad: Alta	Esfuerzo: 2 días	
Criterio de Aceptación:		
1. Framework 4.6 instalado. 2. Configuración de un pool de conexiones para el chatbot 3. Proyecto chatbot en funcionamiento, con salida al exterior con protocolo https		

Tabla 9. Historia de Usuario 1 (Fuente: elaborado por el autor).

Observación: No se menciona la instalación y configuración de IIS, ya que este ya se encontraba debidamente instalado y configurado, al ser un recurso propio del caso de estudio.

HISTORIA DE USUARIO		HU01
Título: Creación de modelo de bases de datos		
Descripción: Como usuario, quiero contar un modelo de base de datos que le permita al chatbot gestionar toda la información requerida en cada módulo de este.		
Prioridad: Alta	Esfuerzo: 1 día	
Criterio de Aceptación:		
<ol style="list-style-type: none"> 1. Esquema y Scripts de las tablas a utilizarse en las bases de datos. 2. Librerías de conexión Sql y Sybase probados. 		

Tabla 10. Historia de Usuario 2 (Fuente: elaborado por el autor).

Observación: No se menciona la instalación y configuración de los DBMS SQL Server y Sybase, ya que estos se encontraban debidamente instalados y configurados, al ser recursos propios del caso de estudio.

2.4.2.1.1.3 SPRINT BACKLOG

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ACTIVIDADES
HU01	Configurar IIS para poner en funcionamiento chatbot.	Framework 4.6 instalado	Instalar Framework 4.6, validando que no se produzca conflictos con otras aplicaciones
		Configuración de un pool de conexiones para el chatbot.	Configurar un pool específico para el chat en donde se asigne los recursos necesarios para su correcto funcionamiento
		Proyecto chatbot en funcionamiento, con salida al exterior con protocolo https	Poner al proyecto chatbot como aplicación web, configurado con protocolo https y con el nombre del dominio.

Tabla 11. Product backlog del Sprint 0 parte A (Fuente: elaborado por el autor).

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ACTIVIDADES
HU02	Creación de modelo de bases de datos	Esquema y Scripts de las tablas a utilizarse en las bases de datos.	Crear los esquemas de las tablas necesarias para el chatbot.
			Generar los Scripts correspondientes
		Librerías de conexión SQL y Sybase probados.	Instalar las librerías de SQL, Sybase y probar su conexión en el ambiente de desarrollo de los servicios para el chatbot

Tabla 12. Product backlog del Sprint 0 parte B (Fuente: elaborado por el autor).

2.4.2.1.1.4 PROTOTIPO

El prototipo diseñado para este sprint corresponde a la interfaz de presentación del chatbot, como se puede observar en la Figura 15.

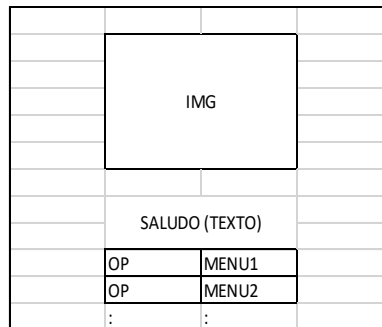


Figura 15. Prototipo presentación chatbot (Fuente: elaborado por el autor).

2.4.2.1.2 SPRINT REVIEW

Se cumplió con el objetivo del sprint, que consistía en la preparación de los servicios necesarios para el correcto funcionamiento del chatbot, como se puede observar en la siguiente tabla.

CÓDIGO	HISTRIA DE USUARIO	CRITERIO DE ACEPTACIÓN	CUMPLIDO
HU01	Configurar IIS para poner en funcionamiento chatbot	Framework 4.6 instalado	SI
		Configuración de un pool de conexiones para el chatbot	SI
		Proyecto chatbot en funcionamiento, con salida al exterior con protocolo https	SI
HU02	Creación de modelo de bases de datos	Esquema y Scripts de las tablas a utilizarse en las bases de datos.	SI
		Librerías de conexión SQL y Sybase probados.	SI

Tabla 13. Revisión de Criterios de Aceptación del Sprint 0 (Fuente: elaborado por el autor).

La siguiente tabla nos muestra el estado del tablero Kanban durante la ejecución del Sprint 0.

PENDIENTE		EN DESARROLLO		EN PRUEBAS		FINALIZADO	
EU10	EU11	EU03	EU04			EU01	EU02
EU12	EU13	EU05	EU06				
EU14	EU15	EU07	EU08				
		EU09					

Tabla 14. Tablero Kanban del Sprint 0 (Fuente: elaborado por el autor).

2.4.2.1.3 SPRINT RETROSPECTIVE

La ventaja de tener preinstalado y configurado el IIS y los DBMS Sql Server y Sybases, facilitaron el cumplimiento de las actividades en tiempos óptimos para el Sprint 0, como se puede observar en la Figura 16.

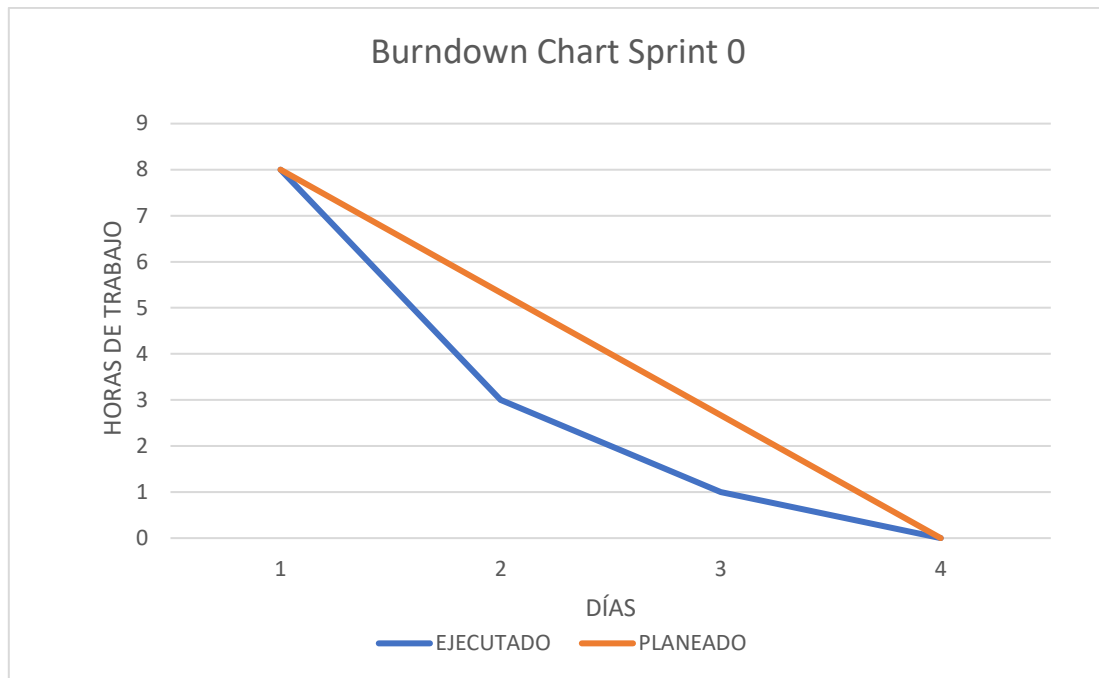


Figura 16. Burndown Chart del Sprint 0 (Fuente: elaborado por el autor).

2.4.2.2 SPRINT 1

2.4.2.2.1 SPRINT PLANNING

2.4.2.2.1.1 OBJETIVO

Construir un módulo que permita realizar solicitudes de pólizas nuevas a través del chatbot y que esta se vea reflejada en el sistema web del caso de estudio.

2.4.2.2.1.2 HISTORIAS DE USUARIO DEL SPRINT 1

HISTORIA DE USUARIO		HU03
Título: Consultar Datos de clientes		
Descripción: Como usuario quiero poder buscar por nombre, código o identificación a mis clientes. Además, quisiera poder visualizar la información del asegurado consultado para verificar si es el correcto.		
Prioridad: Alta	Esfuerzo: 2 días	
Criterio de Aceptación:		
<ol style="list-style-type: none"> 1. Búsqueda por nombre, código o identificación. 2. El chatbot debe distinguir el formato de entrada del usuario si es un nombre buscará por nombre, si es un código buscará por código y si es una identificación buscará por identificación. 3. En búsqueda por nombre, el chatbot debe mostrar una lista de nombres encontrados por coincidencia. 4. Visualizar resumen de datos básicos del cliente encontrado. 		

Tabla 15. Historia de Usuario 3 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU04
Título: Consultar Datos de beneficiarios de la póliza		
Descripción: Como usuario quiero poder buscar por nombre, código o identificación el beneficiario de la póliza a solicitar. Además, quiero poder visualizar la información del beneficiario consultado para verificar si es el correcto.		
Prioridad: Alta	Esfuerzo: 1 día	
Criterio de Aceptación:		
<ol style="list-style-type: none"> 1. Búsqueda por nombre, código o identificación. 2. El chatbot debe distinguir el formato de entrada del usuario si es un nombre buscará por nombre, si es un código buscará por código y si es una identificación buscará por identificación. 3. En búsqueda por nombre, mostrar lista de nombres encontrados por coincidencia. 4. Visualizar resumen de datos básicos del beneficiario encontrado. 		

Tabla 16. Historia de Usuario 4 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU05
Título: Llenar información de la póliza		
Descripción: Como usuario, quiero poder ingresar toda la información correspondiente al monto, plaza, objeto de contrato de la póliza. Además, quiero poder visualizar un resumen de la información ingresada.		
Prioridad: Alta	Esfuerzo: 1 día	
Criterio de Aceptación:		
1. El Chatbot tiene que facilitar las opciones para ingresar el monto, plazo y objeto contractual de la solicitud de póliza.		
2. Visualizar resumen de información ingresada.		

Tabla 17. Historia de Usuario 5 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU06
Título: Escoger tipos de póliza		
Descripción: Como usuario, quiero poder escoger los tipos de pólizas que requiero para la solicitud de póliza. Además, quiero poder asignar el porcentaje de prima y fechas de vigencia correspondiente a cada póliza elegida.		
Prioridad: Alta	Esfuerzo: 2 días	
Criterio de Aceptación:		
1. El chatbot deber mostrar la lista de tipos de pólizas para escoger.		
2. El chatbot debe facilitar el ingreso de porcentaje de prima y fechas de vigencias por cada tipo de póliza seleccionada.		

Tabla 18. Historia de Usuario 6 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU07
Título: Adjuntar Documentos requeridos		
Descripción: Como usuario, quiero poder adjuntar varios documentos al momento de hacer la solicitud de una póliza y que estos se vean reflejados en el sistema web.		
Prioridad: Alta	Esfuerzo: 3 días	
Criterio de Aceptación:		
<ol style="list-style-type: none"> 1. El chatbot deber permitir adjuntar documentos. 2. El chatbot debe guardar el documento adjuntado en el directorio designado para documentos de solicitudes de pólizas. 		

Tabla 19. Historia de Usuario 7 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU08
Título: Procesar solicitud de póliza		
Descripción: Como usuario, quiero que el chatbot procese toda la información ingresada y genere la solicitud.		
Prioridad: Alta	Esfuerzo: 5 días	
Criterio de Aceptación:		
<ol style="list-style-type: none"> 1. El chatbot debe generar la solicitud de póliza nueva, si se cumple con todas las condiciones. 2. El chatbot debe mostrar el número de solicitud generado. 		

Tabla 20. Historia de Usuario 8 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU09
Título: Visualizar Detalle de Póliza Generada		
Descripción: Como usuario, quiero poder visualizar un reporte de la póliza generada.		
Prioridad: Alta	Esfuerzo: 1 día	
Criterio de Aceptación:		
<ol style="list-style-type: none"> 1. El chatbot debe facilitar un reporte detallado de la solicitud de póliza generada. 		

Tabla 21. Historia de Usuario 9 (Fuente: elaborado por el autor).

2.4.2.2.1.3 SPRINT BACKLOG

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ACTIVIDADES
HU03	Consultar Datos de clientes	Búsqueda por nombre, código o identificación.	Crear métodos, servicios de consulta de clientes por código, nombre o identificación.
		El chatbot debe distinguir el formato de entrada del usuario. Si es un nombre, buscará por nombre, si es un código buscará por código y si es una identificación, buscará por identificación.	Crear la dinámica correspondiente a la identificación del tipo de entrada por parte del usuario, haciendo uso de servicios cognitivos y de código.
		En búsqueda por nombre, el chatbot debe mostrar una lista de nombres encontrados por coincidencia.	Mostrar la lista de clientes por coincidencia de nombre, como un Prompt Dialog (componente propio de Bot Framework).
		Visualizar resumen de datos básicos del cliente encontrado.	Crear métodos, servicios para mostrar la información del cliente consultado.

Tabla 22. Product backlog del Sprint 1 parte A (Fuente: elaborado por el autor).

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ACTIVIDADES
HU04	Consultar Datos de beneficiarios de la póliza	Búsqueda por nombre, código o identificación.	Crear métodos, servicios de consulta de clientes por código, nombre o identificación.
		El chatbot debe distinguir el formato de entrada del usuario. Si es un nombre, buscará por nombre, si es un código, buscará por código y si es una identificación, buscará por identificación.	Crear la dinámica correspondiente a la identificación del tipo de entrada por parte del usuario, haciendo uso de servicios cognitivos y de código.
		En búsqueda por nombre, mostrar lista de nombres encontrados por coincidencia.	Mostrar la lista de beneficiarios por coincidencia de nombre, como un Prompt Dialog (componente propio de Bot Framework).
		Visualizar resumen de datos básicos del beneficiario encontrado.	Crear métodos, servicios para mostrar la información del beneficiario consultado.

Tabla 23. Product backlog del Sprint 1 parte B (Fuente: elaborado por el autor).

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ACTIVIDADES
HU05	Llenar información de la póliza	Chatbot tiene que facilitar las opciones para ingresar el monto, plazo y objeto contractual de la solicitud de póliza.	Crear el flujo de diálogos para que el usuario ingrese la información correspondiente al monto, plazo y objeto contractual de la solicitud.
		Visualizar resumen de información ingresada.	Crear método para visualizar un resumen de la información ingresada.
HU06	Escoger tipos de póliza	El chatbot deber mostrar la lista de tipos de pólizas para escoger.	Mostrar la lista de tipos de póliza como Prompt Dialog (Componente propio de Bot Framework).
		El chatbot debe facilitar el ingreso de porcentaje de prima y fechas de vigencias por cada tipo de póliza seleccionada.	Crear el flujo de diálogos para que el usuario pueda ingresar la información correspondiente a cada tipo de póliza.

Tabla 24. Product backlog del Sprint 1 parte C (Fuente: elaborado por el autor).

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ACTIVIDADES
HU07	Adjuntar documentos requeridos	El chatbot deber permitir adjuntar documentos.	Crear el flujo de diálogo para que el usuario pueda adjuntar un archivo en el chatbot.
		El chatbot debe guardar el documento adjuntado en el directorio designado para documentos de solicitudes de pólizas.	Crear métodos, servicios para guardar el archivo adjuntado en el directorio de documentos para solicitud de pólizas nuevas.
HU08	Procesar solicitud de póliza	El chatbot debe generar la solicitud de póliza nueva, si se cumple con todas las condiciones.	Crear métodos, servicios para generar la solicitud de póliza nueva, con las validaciones correspondientes.
		El chatbot debe mostrar el número de solicitud generado.	Crear el método, servicio que retorna el número de solicitud generado.
HU09	Visualizar detalle de póliza generada	El chabot debe facilitar un reporte detallado de la solicitud de póliza generada.	Crear método, servicio para visualizar el detalle de la solicitud generada.

Tabla 25. Product backlog del Sprint 1 parte D (Fuente: elaborado por el autor).

Observación: Para poder realizar el módulo de solicitud de pólizas nuevas y realizar cada una de las actividades correspondientes al Sprint1, se diseñó el diagrama de Flujo ilustrado en el Anexo A.

2.4.2.2.1.4 PROTOTIPO

El prototipo diseñado para este sprint corresponde al módulo de solicitud de póliza nueva. Como se puede observar en las Figuras 17,18 y 19, se mostrarán partes del módulo de solicitud de pólizas nuevas.

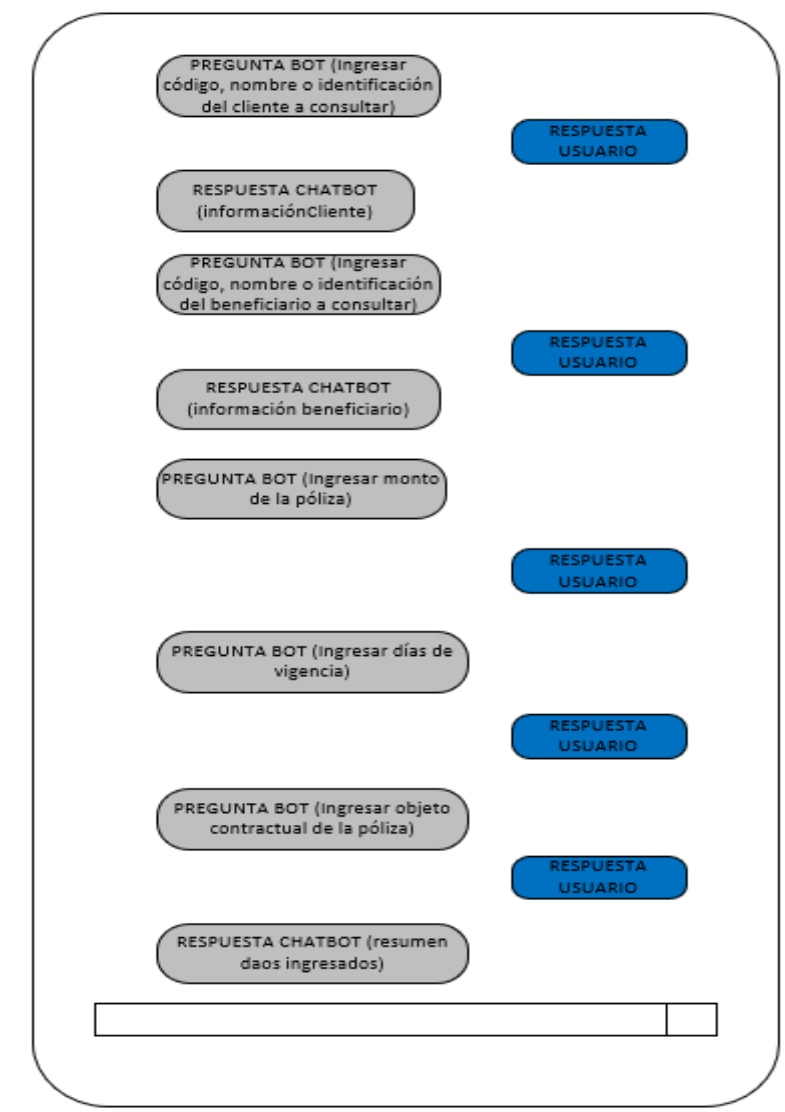


Figura 17. Módulo de solicitud de pólizas nuevas - consulta de cliente, beneficiario - información general de póliza (Fuente: elaborado por el autor).

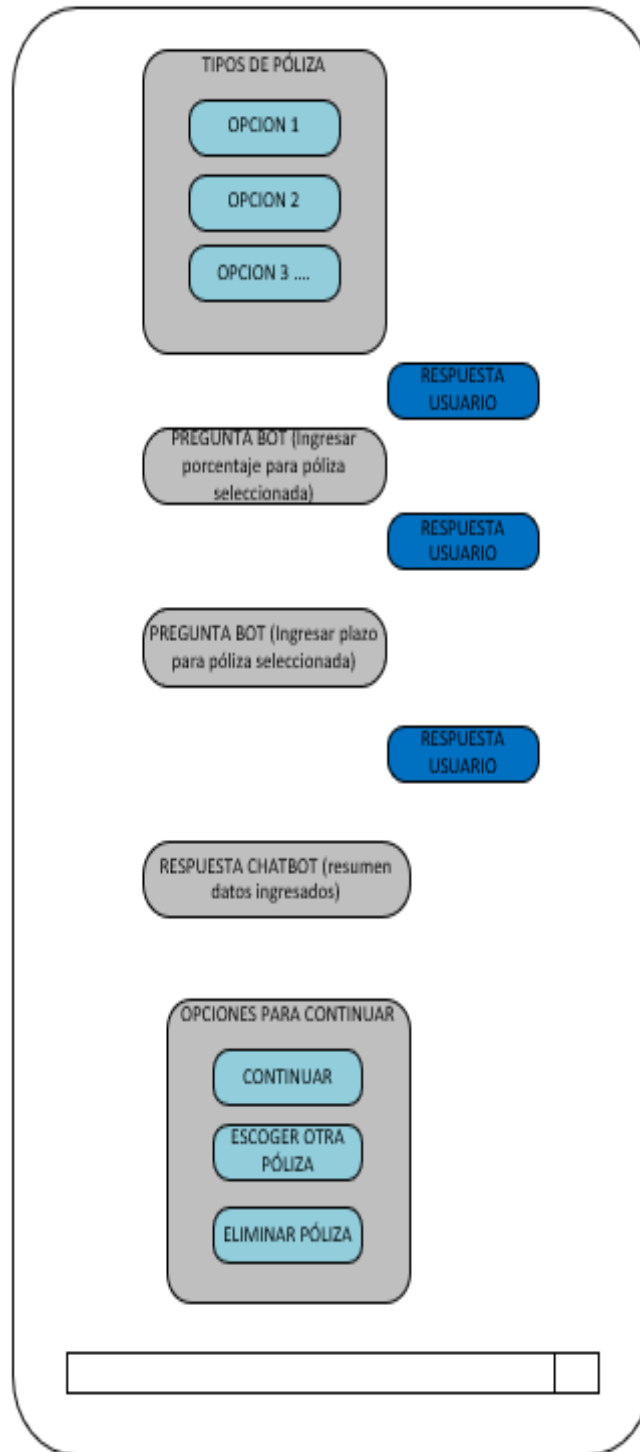


Figura 18. Módulo solicitud de póliza nueva – selección del tipo de póliza (Fuente: elaborado por el autor).

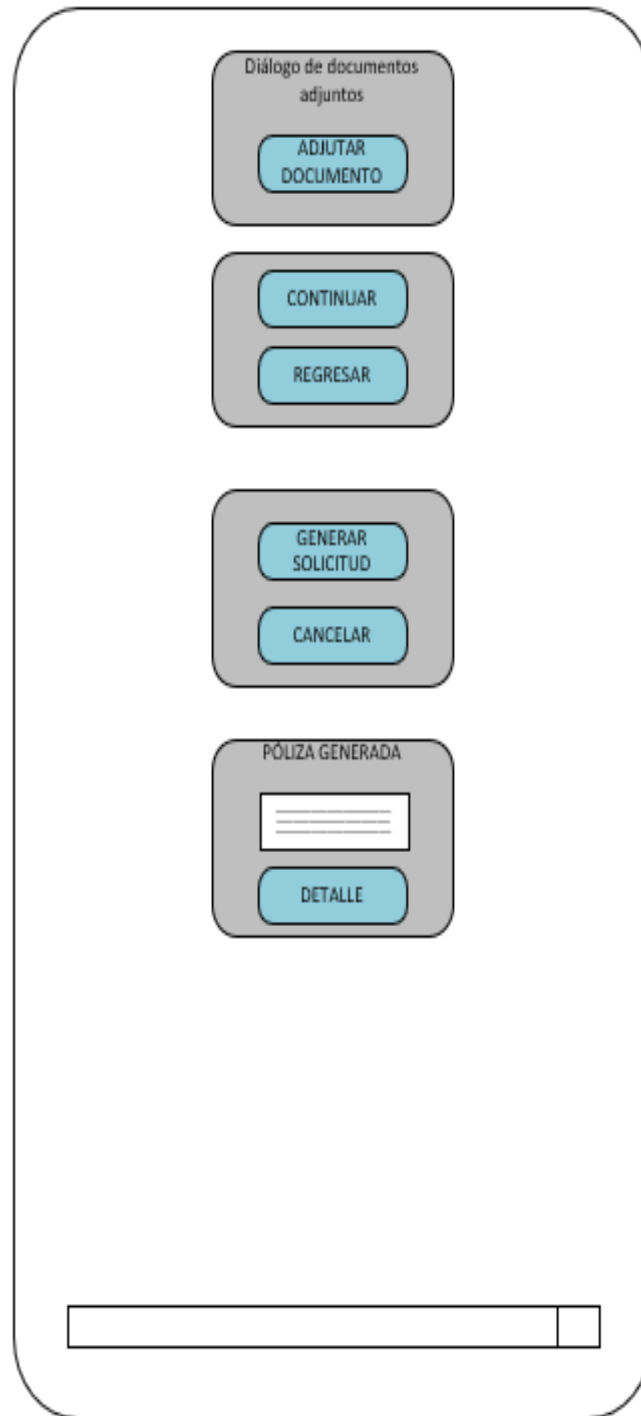


Figura 19. Módulo solicitud de póliza nueva – adjuntar documentos – generar solicitud
(Fuente: elaborado por el autor).

2.4.2.2.2 SPRINT REVIEW

Se cumplió con el objetivo del sprint que consistía en la creación del módulo de solicitud de pólizas nuevas, como se puede observar en la siguiente tabla.

CÓDIGO	HISTRIA DE USUARIO	CRITERIO DE ACEPTACIÓN	CUMPLIDO
HU03	Consultar datos de clientes	Búsqueda por nombre, código o identificación.	SI
		El chatbot debe distinguir el formato de entrada del usuario. Si es un nombre, buscará por nombre; si es un código, buscará por código; y si es una identificación, buscará por identificación.	SI
		En búsqueda por nombre, el chatbot debe mostrar una lista de nombres encontrados por coincidencia.	SI
		Visualizar resumen de datos básicos del cliente encontrado.	SI
HU04	Consultar datos de beneficiarios de la póliza	Búsqueda por nombre, código o identificación.	SI
		El chatbot debe distinguir el formato de entrada del usuario. Si es un nombre, buscará por nombre, si es un código, buscará por código y si es una identificación, buscará por identificación.	SI
		En búsqueda por nombre, mostrar lista de nombres encontrados por coincidencia.	SI
		Visualizar resumen de datos básicos del beneficiario encontrado.	SI

Tabla 26. Revisión de Criterios de Aceptación del Sprint 1 parte A (Fuente: elaborado por el autor).

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	CUMPLIDO
HU05	Llenar información de la póliza	El chatbot tiene que facilitar las opciones para ingresar el monto, plazo y objeto contractual de la solicitud de póliza.	SI
		Visualizar resumen de información ingresada.	SI
HU06	Escoger tipos de póliza	El chatbot deber mostrar la lista de tipos de pólizas para escoger.	SI
		El chatbot debe facilitar el ingreso de porcentaje de prima y fechas de vigencias por cada tipo de póliza seleccionada.	SI
HU07	Adjuntar documentos requeridos	El chatbot deber permitir adjuntar documentos.	SI
		El chatbot debe guardar el documento adjuntado en el directorio designado para documentos de solicitudes de pólizas.	SI
HU08	Procesar solicitud de póliza	El chatbot debe generar la solicitud de póliza nueva, si se cumple con todas las condiciones.	SI
		El chatbot debe mostrar el número de solicitud generado.	SI
HU09	Visualizar detalle de póliza generada	El chatbot debe facilitar un reporte detallado de la solicitud de póliza generada.	SI

Tabla 27. Revisión de Criterios de Aceptación del Sprint 1 parte B (Fuente: elaborado por el autor).

La siguiente tabla nos muestra el estado del tablero Kanban durante la ejecución del Sprint 1.

PENDIENTE		EN DESARROLLO		EN PRUEBAS		FINALIZADO	
EU10	EU11	EU03	EU04			EU01	EU02
EU12	EU13	EU05	EU06				
EU14	EU15	EU07	EU08				
		EU09					

Tabla 28. Tablero Kanban del Sprint 1 (Fuente: elaborado por el autor).

2.4.2.2.3 SPRINT RETROSPECTIVE

En el desarrollo de este Sprint, se contó con la ventaja de tener algunos servicios reutilizables que fueron útiles para cumplir con los criterios de aceptación de este sprint, con tiempos óptimos, como se puede observar en la Figura 20.

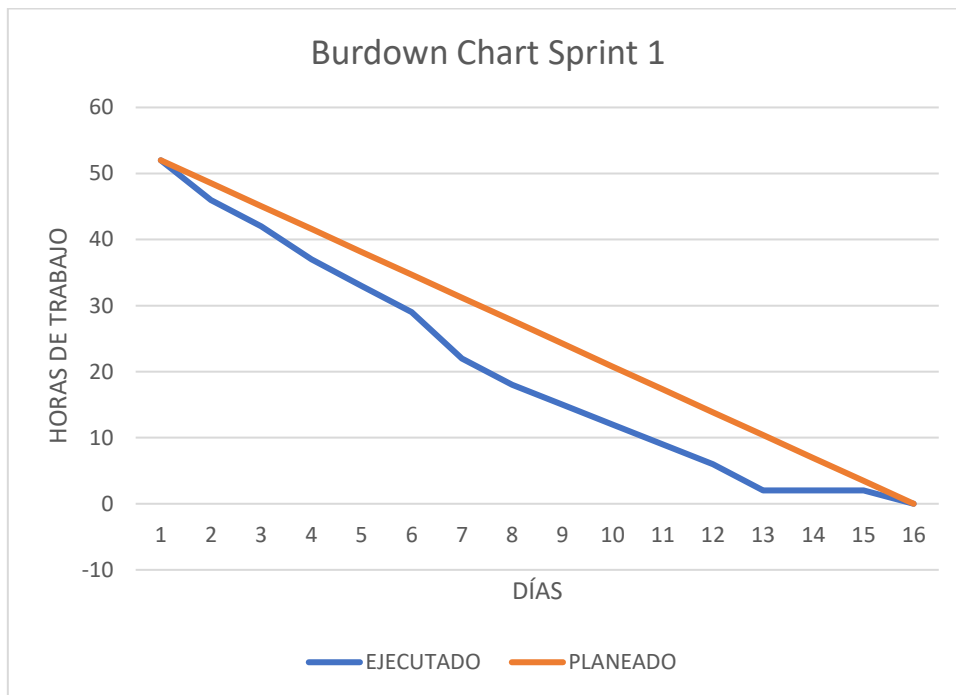


Figura 20. Burndown Chart del Sprint 1 (Fuente: elaborado por el autor).

2.4.2.3 SPRINT 2

2.4.2.3.1 SPRINT PLANNING

2.4.2.3.1.1 OBJETIVO

Construir un módulo que permita realizar consultas de estados de cuentas y pólizas por vencer de los clientes.

2.4.2.3.1.2 HISTORIAS DE USUARIO DEL SPRINT 2

HISTORIA DE USUARIO		HU10
Título: Consultar estados de cuenta		
Descripción: Como usuario, quiero poder consultar los estados de cuenta de mis clientes. La búsqueda de mis clientes debe ser por nombre, código o identificación del cliente. Además, quiero poder visualizar una tabla del estado de cuenta solicitado.		
Prioridad: Alta	Esfuerzo: 3 días	
Criterio de Aceptación:		
<ol style="list-style-type: none">1. Búsqueda de los clientes por nombre, código o identificación.2. El chatbot debe distinguir el formato de entrada del usuario. Si es un nombre, buscará por nombre, si es un código, buscará por código, y si es una identificación, buscará por identificación.3. En búsqueda por nombre, el chatbot debe mostrar una lista de nombres encontrados por coincidencia.4. Visualizar nombre del cliente junto con un link de la tabla que muestra los resultados de la consulta.		

Tabla 29. Historia de Usuario 10 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU11
Título: Consultar pólizas por vencer		
Descripción: Como usuario, quiero poder consultar las pólizas por vencer de mis clientes, desde una fecha inicial hasta una fecha final. Además, quiero poder visualizar esta información en una tabla.		
Prioridad: Alta	Esfuerzo: 3 días	
Criterio de Aceptación:		
<p>1. El chatbot debe facilitar el ingreso de un rango de fechas para realizar la consulta.</p> <p>2. Debe visualizarse los datos básicos del usuario junto con un link de la tabla que muestra los resultados de la consulta.</p>		

Tabla 30. Historia de Usuario 11 (Fuente: elaborado por el autor).

2.4.2.3.1.3 SPRINT BACKLOG

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	CUMPLIDO
HU10	Consultar estados de cuenta	Búsqueda de los clientes por nombre, código o identificación.	SI
		El chatbot debe distinguir el formato de entrada del usuario. Si es un nombre, buscará por nombre, si es un código, buscará por código, y si es una identificación, buscará por identificación.	SI
		En búsqueda por nombre, el chatbot debe mostrar una lista de nombres encontrados por coincidencia.	SI
		Visualizar nombre del cliente junto con un link de la tabla que muestra los resultados de la consulta.	SI

Tabla 31. Product backlog del Sprint 2 parte A (Fuente: elaborado por el autor).

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	CUMPLIDO
HU11	Consultar pólizas por vencer	El chatbot debe facilitar el ingreso de un rango de fechas para realizar la consulta.	SI
		Debe visualizarse los datos básicos del usuario junto con un link de la tabla que muestra los resultados de la consulta.	SI

Tabla 32. Product backlog del Sprint 2 parte B (Fuente: elaborado por el autor).

Observación: Para poder realizar el módulo de consultas y realizar cada una de las actividades correspondientes al Sprint 2, se diseñó el diagrama de Flujo ilustrado en el Anexo B.

2.4.2.3.1.4 PROTOTIPO

El prototipo diseñado para este sprint corresponde al módulo de consultas de estados de cuenta y pólizas por vencer, se mostrará una parte del módulo de consultas (Figura 21).

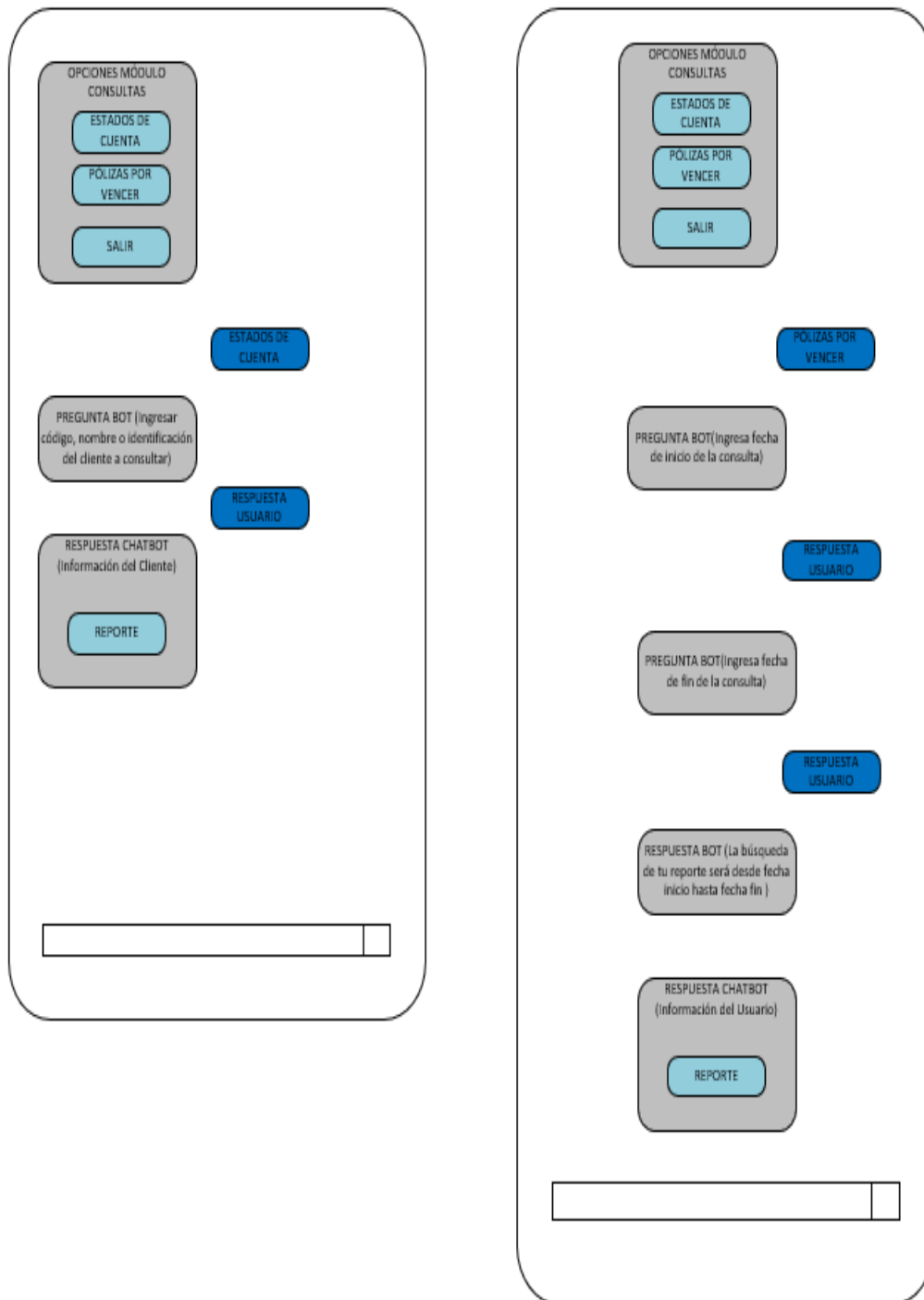


Figura 21. Módulo consultas -estados de cuenta – pólizas por vencer (Fuente: elaborado por el autor).

2.4.2.3.2 SPRINT REVIEW

Se cumplió con el objetivo del sprint que consistía en la creación del módulo de consulta de estados de cuenta y pólizas por vencer de clientes, como se puede observar en la siguiente tabla.

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	CUMPLIDO
HU10	Consultar estados de cuenta	Búsqueda de los clientes por nombre, código o identificación.	SI
		El chatbot debe distinguir el formato de entrada del usuario. Si es un nombre, buscará por nombre, si es un código, buscará por código, y si es una identificación, buscará por identificación.	SI
		En búsqueda por nombre, el chatbot debe mostrar una lista de nombres encontrados por coincidencia.	SI
		Visualizar nombre del cliente junto con un link de la tabla que muestra los resultados de la consulta.	SI
HU11	Consultar pólizas por vencer	El chatbot debe facilitar el ingreso de un rango de fechas para realizar la consulta.	SI
		Debe visualizarse los datos básicos del usuario junto con un link de la tabla que muestra los resultados de la consulta.	SI

Tabla 33. Revisión de Criterios de Aceptación del Sprint 2 (Fuente: elaborado por el autor).

La siguiente tabla nos muestra el estado del tablero Kanban durante la ejecución del Sprint 2.

PENDIENTE		EN DESARROLLO		EN PRUEBAS		FINALIZADO	
EU12	EU13	EU10	EU11	EU05	EU06	EU01	EU02
EU14	EU15			EU07	EU08	EU03	EU04
				EU09			

Tabla 34. Tablero Kanban del Sprint 2 (Fuente: elaborado por el autor).

2.4.2.3.3 SPRINT RETROSPECTIVE

En el desarrollo de este Sprint, ciertas funcionalidades se realizaron con facilidad y rapidez, gracias a la lógica de funcionamiento realizada en el módulo de solicitud de pólizas nuevas, la cual fue reutilizada y adaptada al módulo correspondiente de este sprint. Como podemos observar en la Figura 22, los tiempos están en el rango esperado, pese a que hubo retrasos, se pudo completar el Sprint en el tiempo estimado.

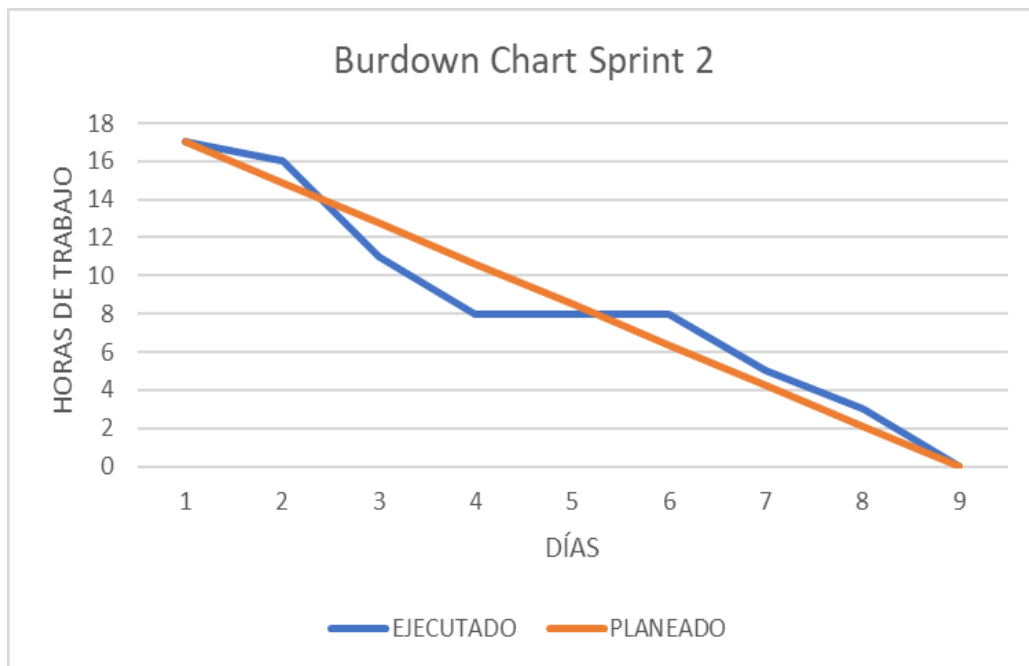


Figura 22. Burndown Chart del Sprint 1 (Fuente: elaborado por el autor).

2.4.2.4 SPRINT 3

2.4.2.4.1 SPRINT PLANNING

2.4.2.4.1.1 OBJETIVO

Construir un módulo que permita realizar preguntas frecuentes con respecto a fianzas y seguros.

2.4.2.4.1.2 HISTORIAS DE USUARIO DEL SPRINT 3

HISTORIA DE USUARIO		HU12
Título: Tener contenido referente a Fianzas y Seguros.		
Descripción: Como usuario, quiero constatar que la información que el chatbot me proporcione sea referente a fianzas y seguros.		
Prioridad: Alta	Esfuerzo: 2 días	
Criterio de Aceptación:		
1. La información que tiene el chatbot, tiene que ser administrable y sintetizada. 2. La información que se almacena en el chatbot, debe ser incremental.		

Tabla 35. Historia de Usuario 12 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU13
Título: Consultar información con respecto a Fianzas y Seguros		
Descripción: Como usuario, quiero poder preguntar al chatbot sobre temas correspondientes a fianzas o seguros y obtener una respuesta de parte de este.		
Prioridad: Alta	Esfuerzo: 1 día	
Criterio de Aceptación:		
1. El chatbot debe facilitar el ingreso de las preguntas que desee el usuario. 2. El chatbot debe contestar en función de la base de conocimiento que tenga almacenado.		

Tabla 36. Historia de Usuario 13 (Fuente: elaborado por el autor).

2.4.2.4.1.3 SPRINT BACKLOG

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ACTIVIDADES
HU12	Tener contenido referente a Fianzas y Seguros	La información que tiene el chatbot, tiene que ser administrable y sintetizada.	Configurar el servicio QnaMaker para que pueda ser usado por el chatbot de forma eficiente.
		La información que se almacena en el chatbot, debe ser incremental.	Alimentar al servicio cognitivo QnaMaker con la información que se considere pertinente de forma continua. Esta actividad continua en el tiempo
HU13	Consultar información con respecto a Fianzas y Seguros	El chatbot debe facilitar el ingreso de las preguntas que desee el usuario.	Crear el flujo de diálogos para que el usuario ingrese la pregunta que desee.
		El chatbot debe contestar en función de la base de conocimiento que tenga almacenado.	Crear método que permitan al chatbot contestar con el conocimiento almacenado en su base de conocimiento.

Tabla 37. Product backlog del Sprint 3 (Fuente: elaborado por el autor).

2.4.2.4.1.4 PROTOTIPO

El prototipo diseñado para este sprint corresponde al módulo de preguntas frecuentes, como se puede observar en la Figura 23.

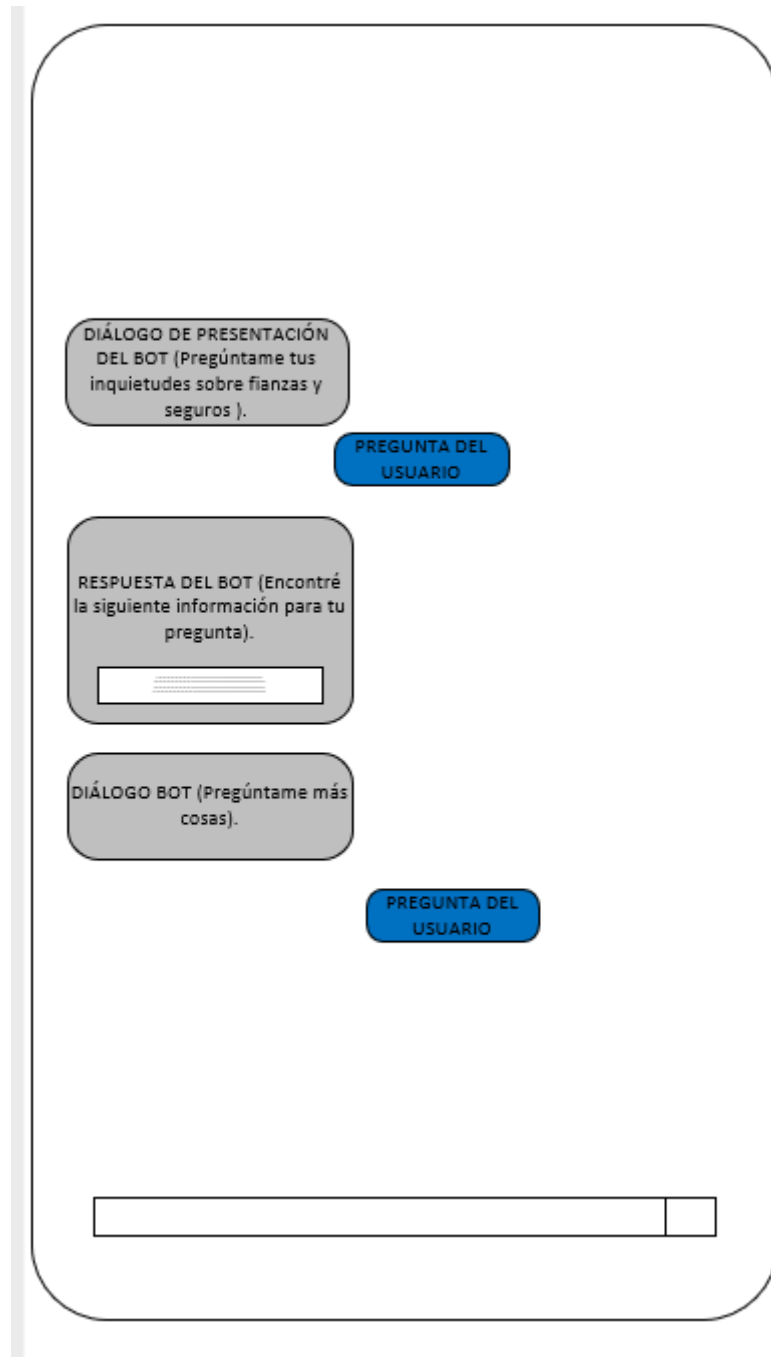


Figura 23. Módulo de preguntas frecuentes (Fuente: elaborado por el autor).

2.4.2.4.2 SPRINT REVIEW

Se cumplió con el objetivo del sprint, que consistía en la creación del módulo de preguntas frecuentes, como se puede observar en la siguiente tabla.

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	CUMPLIDO
HU12	Asegurar y administrar contenido referente a Fianzas y Seguros	La información que tiene el chatbot, tiene que ser administrable y sintetizada.	SI
		La información que se almacena en el chatbot, debe ser incremental.	SI
HU13	Consultar información con respecto a Fianzas y Seguros	El chatbot debe facilitar el ingreso de las preguntas que desee el usuario.	SI
		El chatbot debe contestar en función de la base de conocimiento que tenga almacenado.	SI

Tabla 38. Revisión de Criterios de Aceptación del Sprint 3 (Fuente: elaborado por el autor).

La siguiente tabla nos muestra el estado del tablero Kanban durante la ejecución del Sprint 3.

PENDIENTE		EN DESARROLLO		EN PRUEBAS		FINALIZADO	
EU14	EU15	EU12	EU13	EU10	EU11	EU01	EU02
						EU03	EU04
						EU05	EU06
						EU07	EU08
						EU09	

Tabla 39. Tablero Kanban del Sprint 3 (Fuente: elaborado por el autor).

2.4.2.4.3 SPRINT RETROSPECTIVE

La construcción del módulo de preguntas frecuentes se basa en un mínimo de conocimiento para el chatbot, por lo cual los tiempos destinados a esta actividad son estimados en función de esta condición. Por tanto, el tiempo que se tome en incrementar el conocimiento del chatbot no será tomado en cuenta, ya que esta actividad sería creciente en el tiempo. Tomando en cuenta esta condición, se puede observar en la Figura 24 que los tiempos de desarrollo para este Sprint son óptimos.

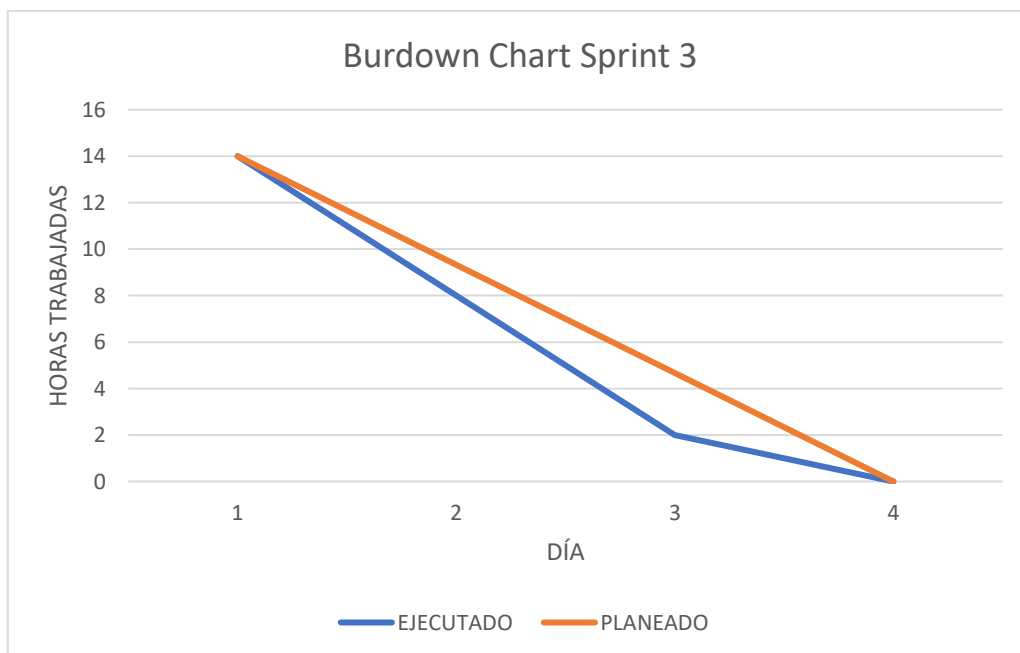


Figura 24. Burndown Chart del Sprint 3 (Fuente: elaborado por el autor).

2.4.2.5 SPRINT 4

2.4.2.5.1 SPRINT PLANNING

2.4.2.5.1.1 OBJETIVO

Construir un módulo que permita realizar renovaciones de pólizas por vencer.

2.4.2.5.1.2 HISTORIAS DE USUARIO DEL SPRINT 4

HISTORIA DE USUARIO		HU14
Título: Consultar póliza a renovar		
Descripción: Como usuario, me gustaría poder consultar la póliza que quiero renovar filtrada por sucursal, tipo de póliza y número de póliza. Además, quiero visualizar la información de la póliza consultada.		
Prioridad: Alta	Esfuerzo: 2días	
Criterio de Aceptación:		
1. Para realizar la consulta de la póliza a renovar, el chatbot debe facilitar la lista de sucursales y tipos de pólizas para que el usuario pueda seleccionarlas. Además de facilitar el ingreso del número de póliza.		
2. Se debe visualizar la información de la póliza seleccionada.		

Tabla 40. Historia de Usuario 14 (Fuente: elaborado por el autor).

HISTORIA DE USUARIO		HU15
Título: Renovar Póliza		
Descripción: Como usuario, quiero poder renovar una póliza, directamente o asignando un número de días para su vigencia. Además, quiero poder visualizar un resumen de la póliza renovada.		
Prioridad: Alta	Esfuerzo: 10 días	
Criterio de Aceptación:		
1. El chatbot debe preguntar si se quiere hacer una renovación directamente o cambiando los días de vigencia.		
2. Se debe visualizar la información de la póliza renovada.		

Tabla 41. Historia de Usuario 15 (Fuente: elaborado por el autor).

2.4.2.5.1.3 SPRINT BACKLOG

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ACTIVIDADES
HU14	Consultar póliza a renovar	Para realizar la consulta de la póliza a renovar, el chatbot debe facilitar la lista de sucursales y tipos de pólizas para que el usuario pueda seleccionarlas. Además de facilitar el ingreso del número de póliza.	Crear un flujo de diálogos para que el usuario pueda ingresar la información de la póliza a consultar
			Crear métodos, servicios para traer lista de sucursales y tipos de pólizas.
			Listar las sucursales y tipos de pólizas, con un Prompt Dialog (Componente propio de Bot Framework).
		Se debe visualizar la información de la póliza seleccionada.	Crear método, servicio para traer la información de la póliza consultada.
HU15	Renovar Póliza	El chatbot debe preguntar si se quiere hacer una renovación directamente o cambiando los días de vigencia.	Crear el flujo de diálogos para que el usuario pueda seleccionar si quiere renovar directamente o cambiando los días de vigencia.
		Se debe visualizar la información de la póliza renovada.	Crear métodos, servicios para realizar la renovación de la póliza consultada y posteriormente mostrar el detalle de esta.

Tabla 42. Product backlog del Sprint 4 (Fuente: elaborado por el autor).

Observación: Para poder realizar el módulo de renovaciones y realizar cada una de las actividades correspondientes al Sprint 4, se diseñó el diagrama de flujo ilustrado en el Anexo C.

2.4.2.5.1.4 PROTOTIPO

El prototipo diseñado para este sprint corresponde al módulo de renovaciones, como se puede observar en la Figura 25, se mostrará ciertas partes del módulo de renovaciones.

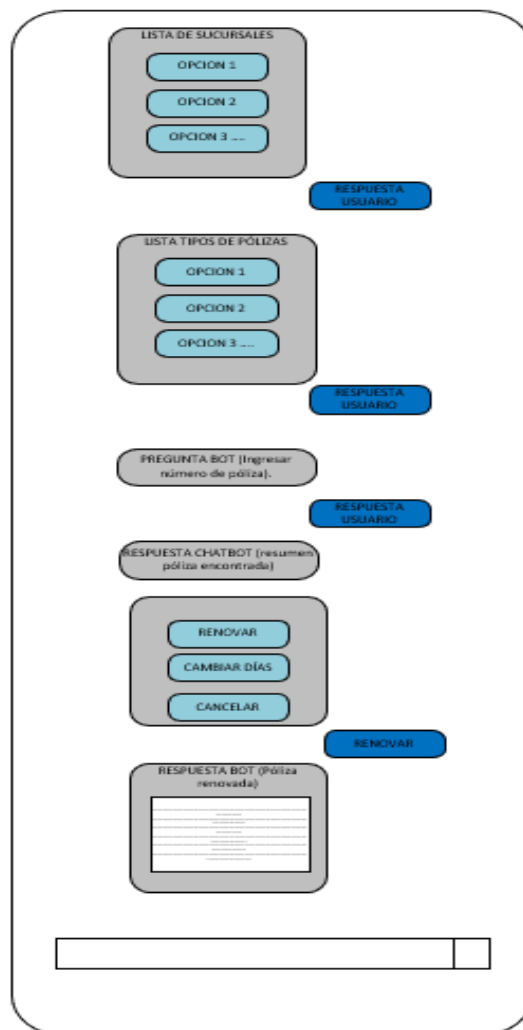


Figura 25. Módulo de renovaciones (Fuente: elaborado por el autor).

2.4.2.5.2 SPRINT REVIEW

Se cumplió con el objetivo del sprint que consistía en la creación del módulo de renovaciones, como se puede observar en la siguiente tabla.

CÓDIGO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	CUMPLIDO
HU14	Consultar póliza a renovar	Para realizar la consulta de la póliza a renovar, el chatbot debe facilitar la lista de sucursales y tipos de pólizas para que el usuario pueda seleccionarlas. Además de facilitar el ingreso del número de póliza.	SI
		Se debe visualizar la información de la póliza seleccionada.	SI
HU13	Renovar Póliza	El chatbot debe preguntar si se quiere hacer una renovación directamente o cambiado los días de vigencia.	SI
		Se debe visualizar la información de la póliza renovada.	SI

Tabla 43. Revisión de Criterios de Aceptación del Sprint 4 (Fuente: elaborado por el autor)

La siguiente tabla nos muestra el estado del tablero Kanban durante la ejecución del Sprint 4.

PENDIENTE		EN DESARROLLO		EN PRUEBAS		FINALIZADO	
		EU14	EU15	EU12	EU13	EU01	EU02
						EU03	EU04
						EU05	EU06
						EU07	EU08
						EU09	EU10
						EU11	

Tabla 44. Tablero Kanban del Sprint 4 (Fuente: elaborado por el autor).

2.4.2.5.3 SPRINT RETROSPECTIVE

La construcción del módulo de renovaciones se agilizo al reutilizar la lógica del módulo de solicitud de pólizas en ciertas funcionalidades, a pesar de que este módulo es más complejo a nivel de servicios que de interfaz, se logró cumplir con los tiempos estimados, como se puede observar en la Figura 26.

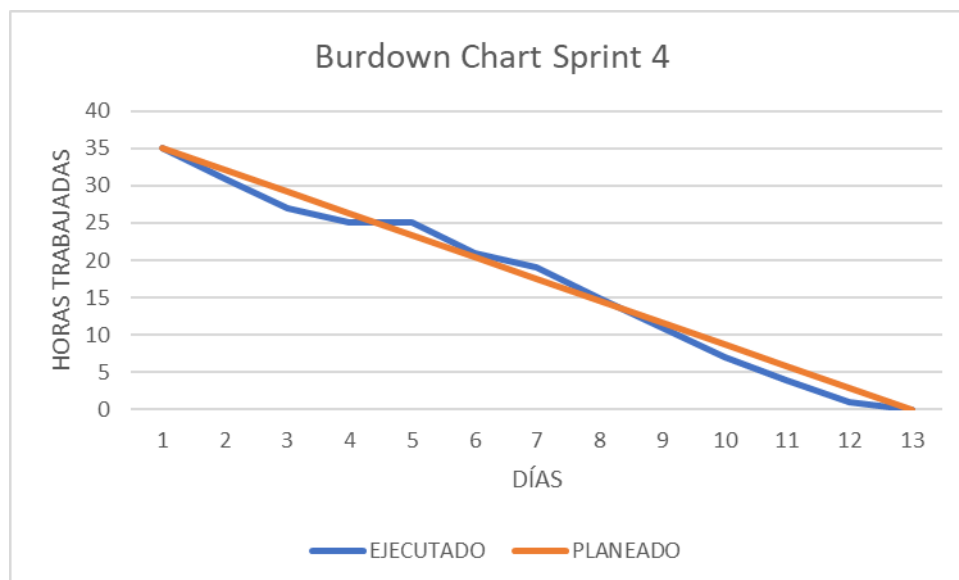


Figura 26. Burndown Chart del Sprint 4 (Fuente: elaborado por el autor).

Una vez finalizada la construcción de los módulos del chatbot, se procede a la ejecución de las pruebas funcionales para poder validar la funcionalidad establecida en los requerimientos.

2.4.3 APLICACIÓN DE CASOS DE PRUEBA

2.4.3.1 CASO DE PRUEBA - MÓDULO SOLICITUD DE PÓLIZAS NUEVAS

Para el módulo de solicitud de pólizas nuevas se definieron 21 casos de prueba. Como se puede observar en la siguiente tabla, los resultados obtenidos son los esperados.

CÓDIGO	DESCRIPCIÓN DE CASO	PREREQUISITOS	RESULTADOS	
			ESPERADO	OBTENIDO
CPAS01	Consultar cliente por código	Sesión iniciada por usuario habilitado (BROKER), cliente a buscar pertenece a cartera de clientes del broker	Información encontrada y desplegada	Información encontrada y desplegada
CPAS02	Consultar cliente por identificación	Sesión iniciada por usuario habilitado (BROKER), cliente a buscar pertenece a cartera de clientes del broker	Información encontrada y desplegada	Información encontrada y desplegada
CPAS03	Consultar cliente por nombre	Sesión iniciada por usuario habilitado (BROKER), cliente a buscar pertenece a cartera de clientes del broker	Lista de nombres por coincidencia desplegada	Lista de nombres por coincidencia desplegada
CPAS04	Cliente no pertenece al BROKER	Cliente consultado	No se despliega la información del cliente y se envía un mensaje "Este cliente no pertenece al Broker"	No se despliega la información del cliente y se envía un mensaje "Este cliente no pertenece al Broker"

Tabla 45. Casos de prueba Módulo solicitudes de pólizas nuevas parte A (Fuente: elaborado por el autor).

CÓDIGO	DESCRIPCIÓN DE CASO	PREREQUISITOS	RESULTADOS	
			ESPERADO	OBTENIDO
CPAS05	Consultar Beneficiario por código	Cliente encontrado y desplegado	Información encontrada y desplegada	Información encontrada y desplegada
CPAS06	Consultar Beneficiario por identificación	Cliente encontrado y desplegado	Información encontrada y desplegada	Información encontrada y desplegada
CPAS07	Consultar Beneficiario por nombre	Cliente encontrado y desplegado	Lista de nombres por coincidencia desplegada	Lista de nombres por coincidencia desplegada
CPAS08	Ingresar monto, plazo y objeto contractual	Cliente y Beneficiario encontrado y desplegado	Resumen de datos ingresados desplegado	Resumen de datos ingresados desplegado
CPAS09	Visualizar tipos de pólizas a escoger	Cliente y Beneficiario encontrado, monto, plazo y objeto contractual registrados	Lista de tipos de Pólizas para seleccionar desplegado	Lista de tipos de Pólizas para seleccionar desplegado
CPAS10	Opciones para prima del tipo de póliza seleccionado	Tipo de póliza seleccionada	lista de opciones para prima de la póliza seleccionada desplegada	lista de opciones para prima de la póliza seleccionada desplegada
CPAS11	Conservar prima por defecto del tipo de póliza seleccionado	Tipo de póliza seleccionada	Información con prima del tipo de póliza, desplegado	Información con prima del tipo de póliza desplegada
CPAS12	Cambiar prima del tipo de póliza seleccionado	Tipo de póliza seleccionada	Información de la nueva prima ingresado por el usuario desplegado	Información de la nueva prima ingresado por el usuario desplegado
CPAS13	Opciones para plazo del tipo de póliza seleccionado	Tipo de póliza seleccionada, prima registrada	lista de opciones para plazo de la póliza seleccionada desplegada	lista de opciones para plazo de la póliza seleccionada desplegada
CPAS14	Conservar plazo por defecto del tipo de póliza seleccionado	Tipo de póliza seleccionada, prima registrada	Información con el plazo del tipo de póliza, desplegado	Información con el plazo del tipo de póliza, desplegado

Tabla 46. Casos de prueba Módulo solicitudes de pólizas nuevas parte B (Fuente: elaborado por el autor).

CÓDIGO	DESCRIPCIÓN DE CASO	PREREQUISITOS	RESULTADOS	
			ESPERADO	OBTENIDO
CPAS15	Cambiar plazo del tipo de póliza seleccionado	Tipo de póliza seleccionada, prima registrada	Información del nuevo plazo ingresado por el usuario desplegado	Información del nuevo plazo ingresado por el usuario desplegado
CPAS16	Ingresar fecha de inicio y fin del tipo de póliza seleccionado	Tipo de póliza seleccionada, prima registrada, plazo registrado	Información con la fecha de Inicio y fin de vigencia del tipo de póliza seleccionado	Información con la fecha de Inicio y fin de vigencia del tipo de póliza seleccionado
CAPS17	Validar combinación correcta de nuevo tipo de póliza seleccionado con los tipos de pólizas registrados.	Al menos un tipo de póliza registrado	Proceso de selección de póliza continua sin interrupción	Proceso de selección de póliza continua sin interrupción
CAPS18	Validar combinación incorrecta de nuevo tipo de póliza seleccionado con los tipos de pólizas registrados.	Al menos un tipo de póliza registrado	Se envía mensaje de información " Este tipo de póliza no puede combinarse con los tipos de pólizas registrados"	Se envía mensaje de información " Este tipo de póliza no puede combinarse con los tipos de pólizas registrados"
CAPS19	Adjuntar documentación	Cliente y Beneficiario encontrado, monto, plazo y objeto contractual registrados, al menos un tipo de póliza registrado	Diálogo para adjuntar documento desplegado	Diálogo para adjuntar documento desplegado

Tabla 47. Casos de prueba Módulo solicitudes de pólizas nuevas parte C (Fuente: elaborado por el autor).

CÓDIGO	DESCRIPCIÓN DE CASO	PREREQUISITOS	RESULTADOS	
			ESPERADO	OBTENIDO
CAPS20	Validar Documento Adjunto	Cliente y Beneficiario encontrado, monto, plazo y objeto contractual registrados, al menos un tipo de póliza registrado, documento enviado	Se envía un mensaje "Se guardo su documento en nuestro gestor documental"	se envía un mensaje "Se guardo su documento en nuestro gestor documental"
caps21	Generación de solicitud de póliza nueva	Cliente y Beneficiario encontrado, monto, plazo y objeto contractual registrados, al menos un tipo de póliza registrada, al menos un documento adjuntado	Se envía un mensaje con el número de solicitud generado y un link al detalle de la solicitud	Se envía un mensaje con el número de solicitud generado y un link al detalle de la solicitud

Tabla 48. Casos de prueba Módulo solicitudes de pólizas nuevas parte D (Fuente: elaborado por el autor).

2.4.3.2 CASO DE PRUEBA - MÓDULO CONSULTA DE ESTADOS DE CUENTA Y PÓLIZAS POR VENCER

Para el módulo de consultas de estados de cuenta y pólizas por vencer se definieron 6 casos de prueba. Como se puede observar en la siguiente tabla, los resultados obtenidos son los esperados.

CÓDIGO	DESCRIPCIÓN DE CASO	PREREQUISITOS	RESULTADOS	
			ESPERADO	OBTENIDO
CPAC01	Consultar cliente por código	Sesión iniciada por usuario habilitado (BROKER), cliente a buscar pertenece a cartera de clientes del broker	Información encontrada y desplegada	Información encontrada y desplegada
CPAC02	Consultar cliente por identificación	Sesión iniciada por usuario habilitado (BROKER), cliente a buscar pertenece a cartera de clientes del broker	Información encontrada y desplegada	Información encontrada y desplegada

Tabla 49. Casos de prueba Módulo de consulta de estados de cuenta y pólizas por vencer de clientes parte A (Fuente: elaborado por el autor).

CÓDIGO	DESCRIPCIÓN DE CASO	PREREQUISITOS	RESULTADOS	
			ESPERADO	OBTENIDO
CPAC03	Consultar cliente por nombre	Sesión iniciada por usuario habilitado (BROKER), cliente a buscar pertenece a cartera de clientes del broker	Lista de nombres por coincidencia desplegada	Lista de nombres por coincidencia desplegada
CPAC04	Cliente no pertenece al BROKER	Cliente consultado	No se despliega la información del cliente y se envía un mensaje "Este cliente no pertenece al Broker"	No se despliega la información del cliente y se envía un mensaje "Este cliente no pertenece al Broker"
CPAC05	Mostrar link a tabla con información de los estados de cuenta del cliente consultado.	Cliente encontrado y desplegado	Información básica del cliente desplegada junto con un link que redirecciona a página web que contiene una tabla con la información encontrada.	Información básica del cliente desplegada junto con un link que redirecciona a página web que contiene una tabla con la información encontrada.
CPAC06	Consultar y Mostrar link a tabla con información de las pólizas por vencer de los clientes del Broker.	Sesión iniciada por usuario habilitado (BROKER)	Información básica del broker desplegada junto con un link que redirecciona a página web que contiene una tabla con la información encontrada.	Información básica del broker desplegada junto con un link que redirecciona a página web que contiene una tabla con la información encontrada.

Tabla 50. Casos de prueba Módulo de consulta de estados de cuenta y pólizas por vencer de clientes parte B (Fuente: elaborado por el autor).

2.4.3.3 CASO DE PRUEBA – MÓDULO PREGUNTAS FRECUENTES

Para el módulo de preguntas frecuentes se definieron dos casos de prueba. Como se puede observar en la siguiente tabla, los resultados obtenidos son los esperados.

CÓDIGO	DESCRIPCIÓN DE CASO	PREREQUISITOS	RESULTADOS	
			ESPERADO	OBTENIDO
CPAP01	Consultar temas de Fianzas y Seguros	El chatbot solicita una pregunta de parte del Usuario. El usuario escribe y envía la pregunta solicitada por el chatbot.	Mensaje Informativo con la respuesta que el chatbot encontró.	Mensaje Informativo con la respuesta que el chatbot encontró.
CPAP02	Salir del módulo de preguntas frecuentes	Usuario escribe su intención de salir.	El chatbot interpreta la intención de salir del usuario y cierra el módulo de preguntas frecuentes	El chatbot interpreta la intención de salir del usuario y cierra el módulo de preguntas frecuentes

Tabla 51. Casos de prueba Módulo de preguntas frecuentes por vencer de clientes
(Fuente: elaborado por el autor).

2.4.3.4 CASO DE PRUEBA – MÓDULO RENOVACIONES DE PÓLIZAS

Para el módulo de preguntas frecuentes se definieron tres casos de prueba. Como se puede observar en la siguiente tabla, los resultados obtenidos son los esperados.

CÓDIGO	DESCRIPCIÓN DE CASO	PREREQUISITOS	RESULTADOS	
			ESPERADO	OBTENIDO
CPAR01	Consultar póliza a renovar	Sesión iniciada por usuario habilitado (BROKER)	Información encontrada y desplegada	Información encontrada y desplegada
CPAR02	Proporcionar opción de cambio de días de vigencia para renovar póliza	Sesión iniciada por usuario habilitado (BROKER), póliza encontrada.	El chatbot debe mostrar una opción para que el usuario pueda cambiar los días de vigencia de la póliza a renovar si este lo desea	El chatbot debe mostrar una opción para que el usuario pueda cambiar los días de vigencia de la póliza a renovar si este lo desea
CPAR3	Visualizar información de póliza renovada	Sesión iniciada por usuario habilitado (BROKER), renovación procesada.	EL chatbot debe mostrar la información de la póliza renovada.	EL chatbot debe mostrar la información de la póliza renovada.

Tabla 52. Casos de prueba Módulo de renovaciones de pólizas (Fuente: elaborado por el autor).

Una vez finalizada la ejecución de las pruebas funcionales y con la aceptación de los usuarios, la aplicación ha sido terminada. La siguiente tabla muestra el tablero Kanban con las historias de usuario finalizadas.

PENDIENTE		EN DESARROLLO		EN PRUEBAS		FINALIZADO	
						EU01	EU02
						EU03	EU04
						EU05	EU06
						EU07	EU08
						EU09	EU10
						EU11	EU12
						EU13	EU14
						EU15	

Tabla 53. Tablero Kanban finalizadas las pruebas funcionales (Fuente: elaborado por el autor).

3 RESULTADOS Y DISCUSIÓN

3.1 APLICACIÓN DEL CASO DE ESTUDIO

3.1.1 EJEMPLO DE USO DEL CHATBOT

El chatbot consta de dos opciones principales: la primera opción es específica para el área de Fianzas, en el cual se despliegan los módulos correspondientes a solicitudes, consultas y renovaciones. Este conjunto de módulos solo puede ser accedido por personas que tengan un usuario y clave registrados en la base de datos del caso de estudio, y que tengan un perfil de bróker o cliente, en caso contrario, no podrán acceder a estos módulos, como se observa en la Figura 27.

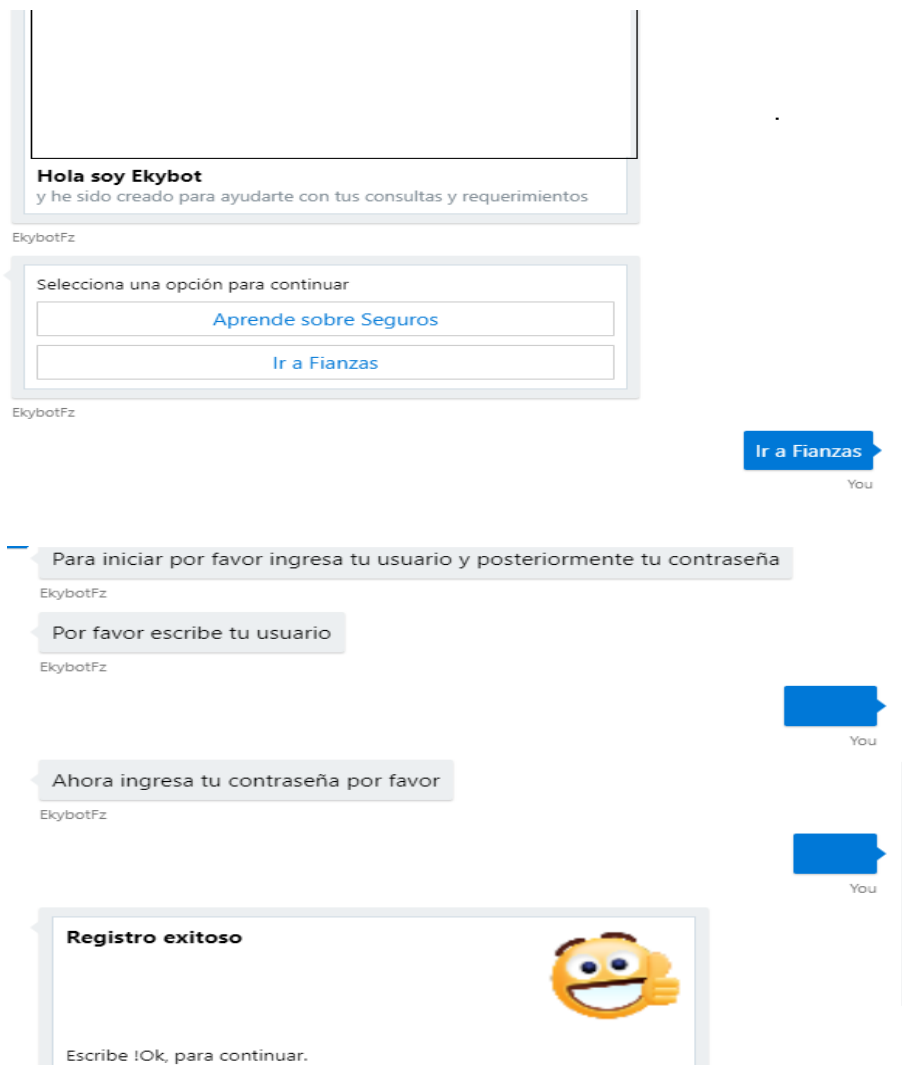


Figura 27. Inicio de sesión para acceder a los módulos de Fianzas (Fuente: elaborado por el autor).

La segunda opción corresponde al módulo de preguntas frecuentes. Para acceder a este módulo, no es necesario que la persona inicie una sesión, como se puede observar en la Figura 28.

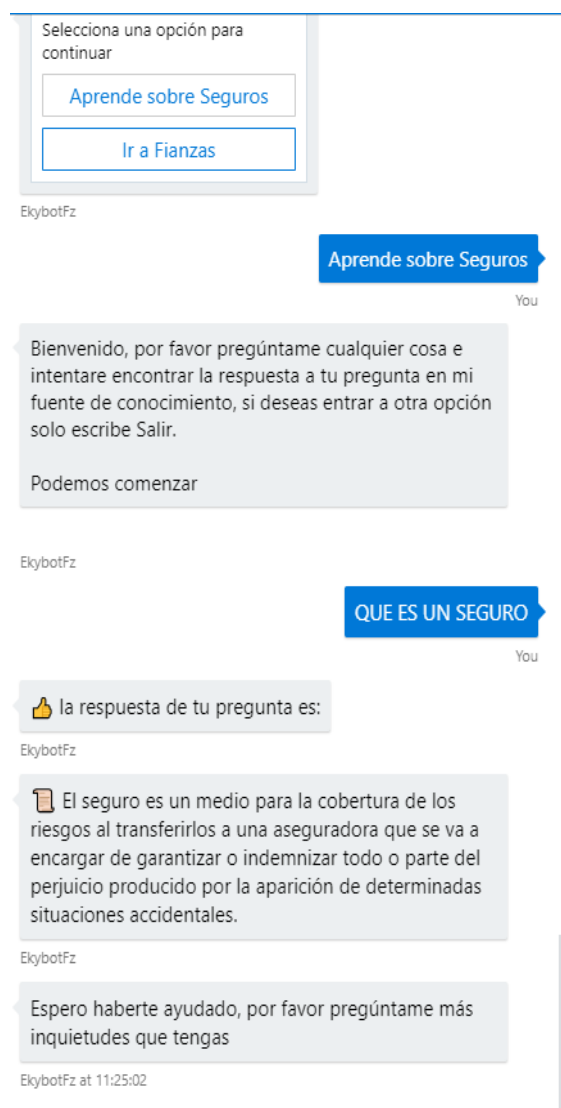


Figura 28. Interacción con el módulo de preguntas frecuentes (Fuente: elaborado por el autor).

3.1.1.1 Módulo de solicitudes de pólizas nuevas

Al iniciar sesión, el usuario tiene cuatro opciones, una de las cuales es la de solicitar póliza. Si el usuario escoge esta opción, el chatbot empieza a entablar un diálogo con el usuario, en el cual le pedirá ingresar la información necesaria para que sea posible la generación de la solicitud.

En las Figuras (29-36) se puede observar paso a paso el diálogo que se realiza entre el usuario y el chatbot.

En la Figura 29, se observa cómo el chatbot solicita al usuario que ingrese cierta información, esta información es utilizada por el chatbot para poder encontrar el cliente y el beneficiario que el usuario desea asignar a la póliza.

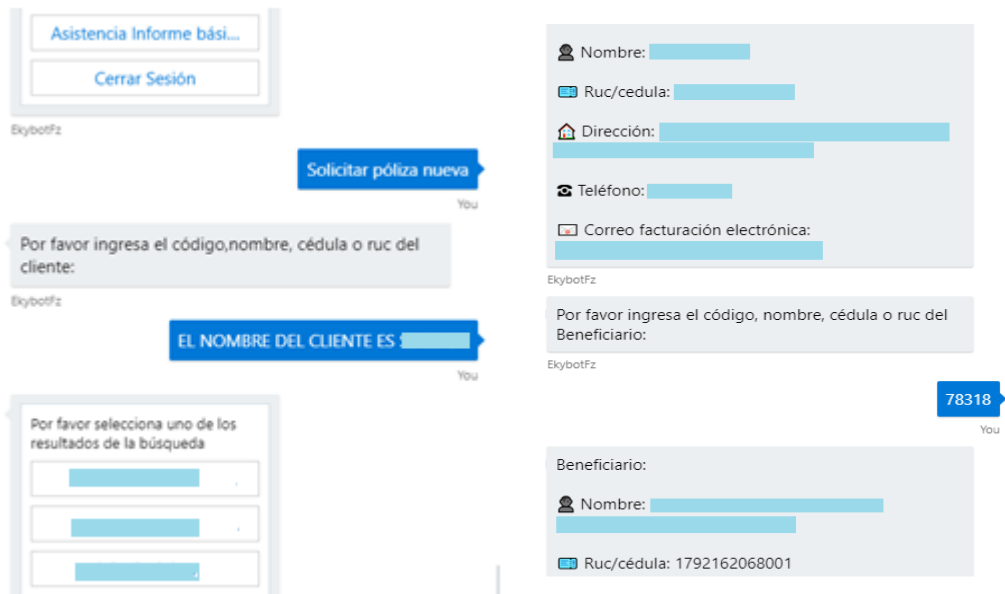


Figura 29. Ingreso de Cliente y beneficiario de la póliza a ser solicitada (Fuente: elaborado por el autor).

La Figura 30 muestra como el chatbot solicita al usuario que ingrese información del monto, plazo y objeto contractual de la póliza y posteriormente se muestra un resumen de los datos ingresados.

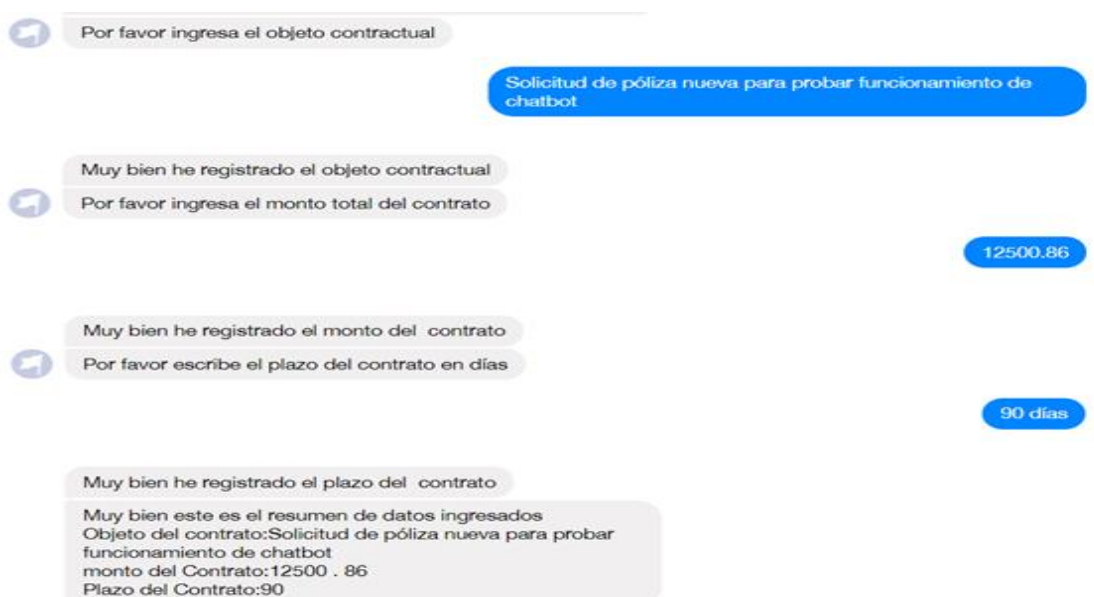


Figura 30. Ingreso del monto, plazo y objeto contractual de la póliza a ser solicitada (Fuente: elaborado por el autor).

La Figura 31 muestra cómo se agrega un tipo de póliza en la solicitud, en este caso el chatbot ofrece algunas opciones que el usuario puede escoger durante el diálogo.

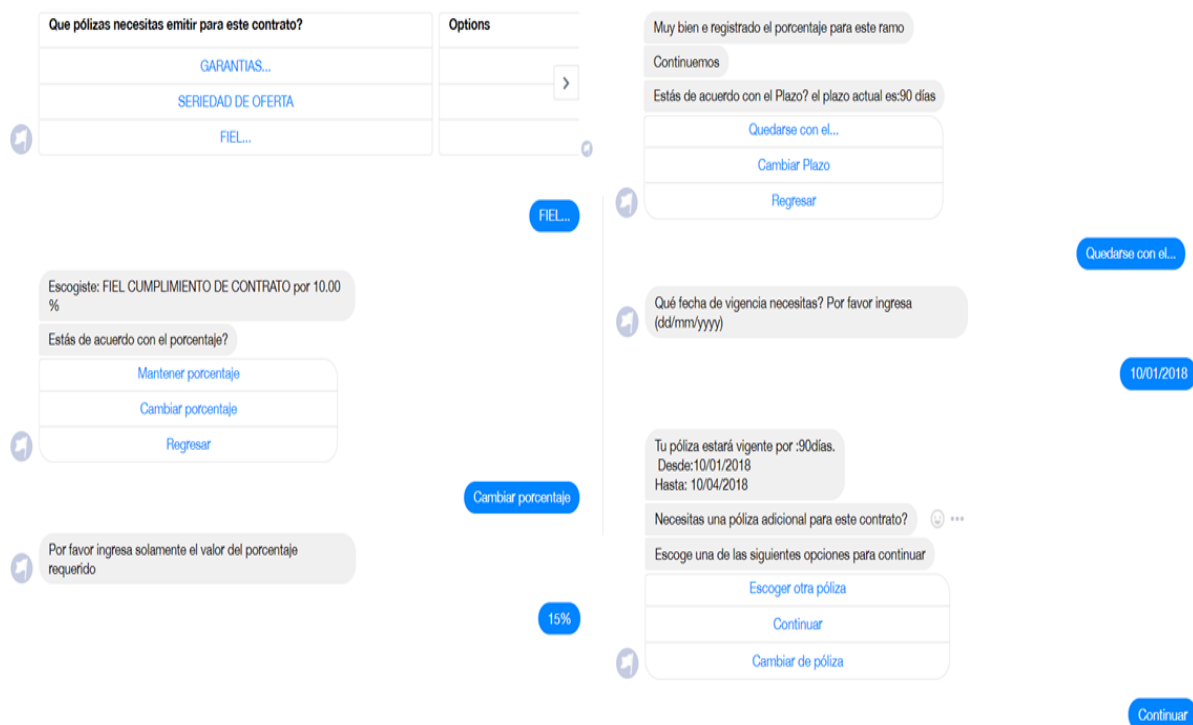


Figura 31. Ingreso de un tipo de póliza para la solicitud (Fuente: elaborado por el autor).

En la Figura 32, se observa cómo el chatbot proporciona un diálogo que contiene un botón, el cual abre una interfaz web, que sirve para adjuntar los documentos que el usuario crea necesarios para la solicitud.

Observación: Se puede adjuntar documentos a través del chatbot, pero por requerimiento del caso de estudio se optó por la opción del link.



Figura 32. Diálogo para adjuntar documentos (Fuente: elaborado por el autor).

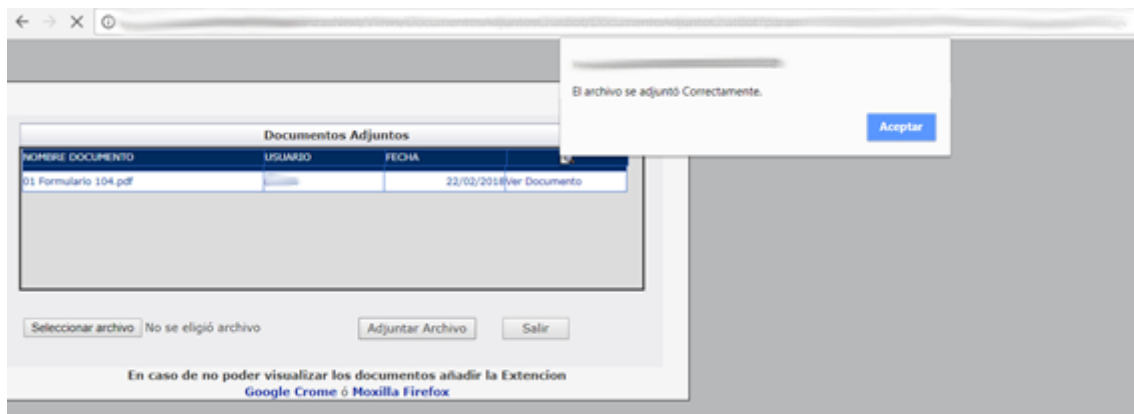


Figura 33. Interfaz web para adjuntar documentos (Fuente: elaborado por el autor).

Para poder continuar con el proceso de la solicitud, el chatbot valida si se adjuntó al menos un documento.

Una vez adjuntado los documentos, el chatbot muestra un diálogo, el cual tiene dos opciones, la primera para generar la solicitud y la segunda para cancelar el proceso (ver Figura 34).

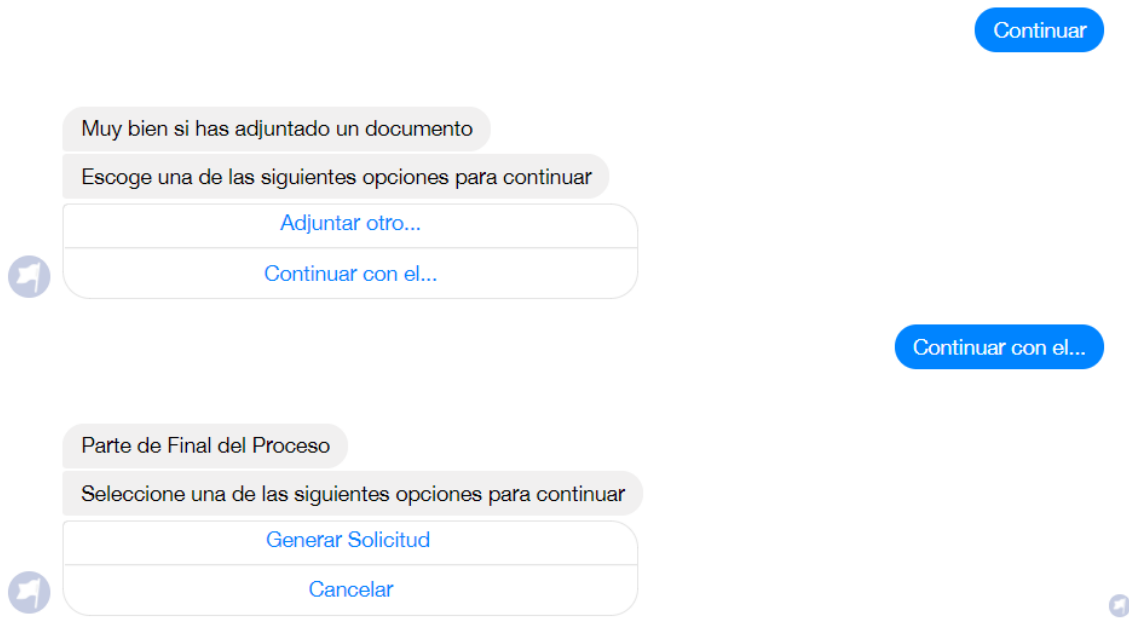


Figura 34. Diálogo para generar o cancelar la solicitud (Fuente: elaborado por el autor).

Cuando el usuario decide generar la solicitud, el chatbot procesa toda la información ingresada durante todo el diálogo, si se cumple con todas las condiciones requeridas para la generación de la solicitud, el chatbot muestra un diálogo con el número de solicitud generada y un link con el reporte del convenio de la solicitud, caso contrario muestra la información con la razón de por qué no se generó la solicitud (ver Figura 35).

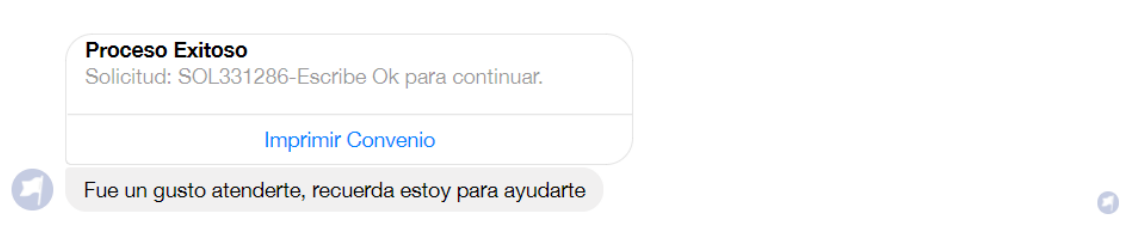


Figura 35. Solicitud generada desde el chatbot (Fuente: elaborado por el autor).



Figura 36. Reporte del convenio de la solicitud generada (Fuente: elaborado por el autor).

Finalmente, el usuario puede constatar si la solicitud fue generada correctamente, ingresando a la página web de Fianzas del caso de estudio.

3.1.1.2 Módulo de consultas

Otra opción que el chatbot ofrece al usuario es el módulo de consultas. Si el usuario escoge esta opción, el chatbot empieza a entablar un diálogo con el usuario, en el cual le pedirá ingresar la información necesaria para que se pueda realizar la consulta solicitada.

Cuando el usuario selecciona la opción Mis Consultas, el chatbot muestra las opciones de consulta que se puede realizar, como se muestra en la Figura 37.

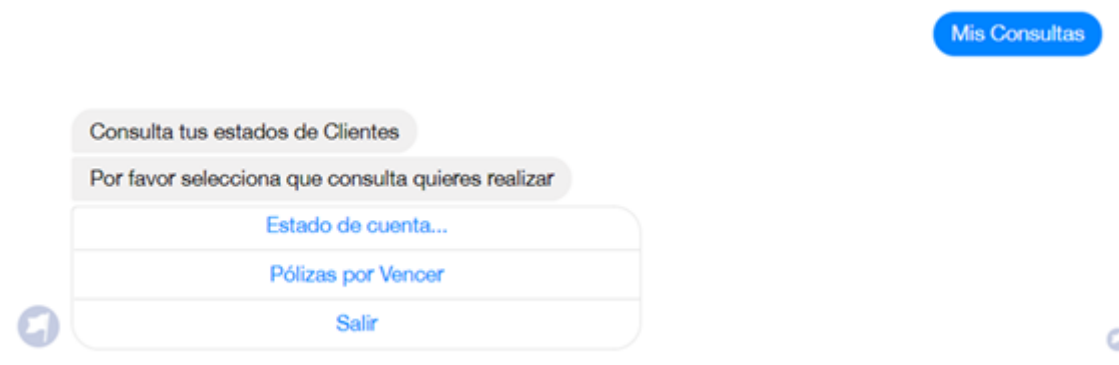


Figura 37. Opciones de consulta desde el chatbot (Fuente: elaborado por el autor).

Para la primera opción (Estado de cuenta) el usuario tiene que ingresar el nombre, código o identificación del cliente que desea consultar, tal como lo hace en el módulo de

solicitud de póliza nueva. Una vez encontrado el cliente, se muestra su información y el respectivo link al reporte web, como se observa en la figura 38 y 39.



Figura 38. Consulta estado de cuenta desde chatbot (Fuente: elaborado por el autor).

Fecha Emisión	Cliente	Póliza	Anexo	Ramo	Beneficiario	Monto Asegurado	Prima Neta	Deuda Total	Número Factura
16/01/2018	[redacted]	60672	130602	BUEN USO DE ANTICIPO	UNION CEMENTERA NACIONAL UCEN S.A.	\$[redacted]	\$[redacted]	\$[redacted]	2000011 544291
16/01/2018	[redacted]	66740	0	BUEN USO DE ANTICIPO	UNION CEMENTERA NACIONAL UCEN S.A.	\$[redacted]	\$[redacted]	\$[redacted]	2000700 544340
17/01/2018	[redacted]	84094	230600	FIEL CUMPLIMIENTO DE CONTRATO	EMPRESA ELECTRICA QUITO S.A. E.E.Q.	\$[redacted]	\$[redacted]	\$[redacted]	2000000 544395
17/01/2018	[redacted]	61514	130606	BUEN USO DE ANTICIPO	EMPRESA ELECTRICA QUITO S.A. E.E.Q.	\$[redacted]	\$[redacted]	\$[redacted]	2000000 544391
17/01/2018	[redacted]	64011	130900	BUEN USO DE	CORPORACION ELECTRICA DEL ECUADOR	\$[redacted]	\$[redacted]	\$[redacted]	2000000 544392

Figura 39. Reporte web de consulta de estado (Fuente: elaborado por el autor).

Para la segunda opción (Pólizas por vencer), el usuario tiene que ingresar la fecha de inicio y fin de la consulta, por lo cual el chatbot muestra el diálogo correspondiente para el ingreso de esta información. Una vez ingresada esta información el chatbot muestra la información del usuario junto con el link al reporte web, como se observa en la figura 40 y 41.

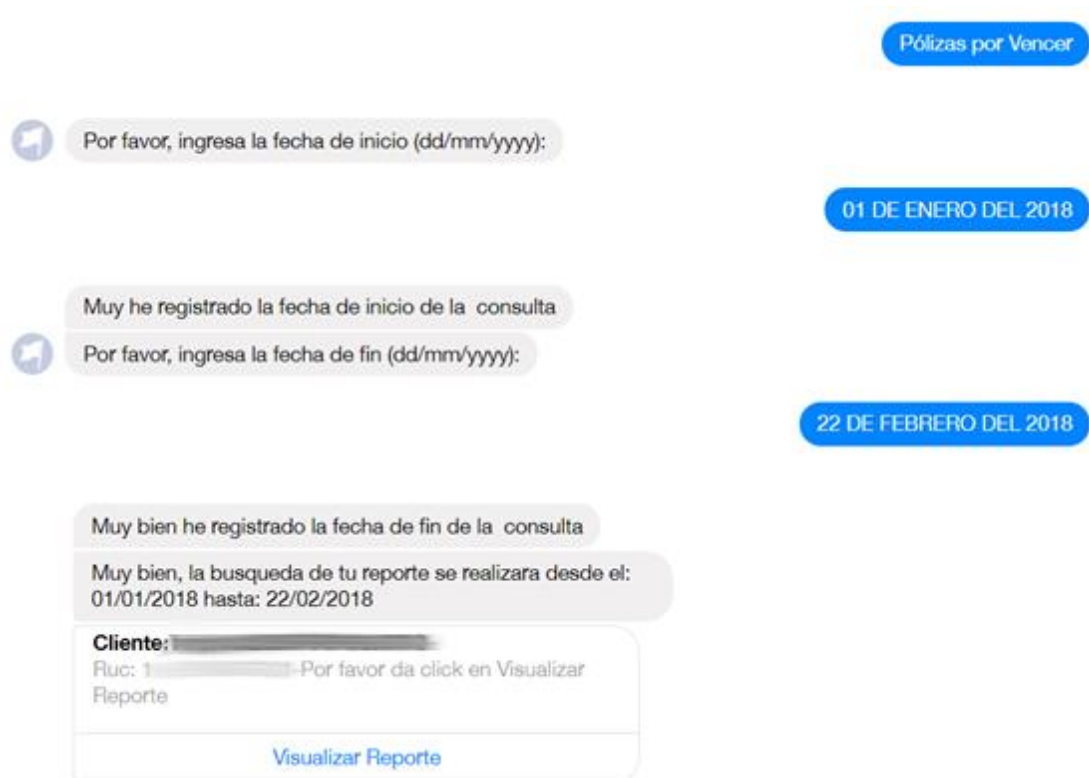


Figura 40. Consulta pólizas por vencer desde chatbot (Fuente: elaborado por el autor).

Reporte de Polizas por Vencer

CODIGO CUENTE	CUENTE	POLIZA	RAMO	SUMA ASEGURADA	BENEFICIARIO	FECHA VENCIMIENTO	OBJETO	SUCURSAL
154	[Redacted]	15631	GARANTIA ADUANERA	150000	[Redacted]	31/01/2018	GARANTIA ADUANERA PARA ASEGURAR EL PAGO DE LAS FORMALIDADES DETERMINADAS POR LA	QUITO

Exportar Excel

Figura 41. Reporte web de pólizas por vencer (Fuente: elaborado por el autor).

3.1.1.3 Módulo de renovaciones de pólizas

Otra opción que el chatbot ofrece al usuario es el módulo de renovaciones, si el usuario escoge esta opción, el chatbot entabla diálogo con el usuario, en el cual le pedirá

ingresar la información necesaria para que se pueda realizar la renovación de una póliza.

Cuando el usuario selecciona la opción Renovaciones, el chatbot muestra una lista con las sucursales que puede escoger para realizar la consulta de la póliza a renovar (ver Figura 42).



Figura 42. Lista de sucursales presentada desde el chatbot (Fuente: elaborado por el autor).

Una vez seleccionada la sucursal, el chatbot muestra una lista con los tipos de pólizas de Fianzas, además muestra un diálogo para que el usuario ingrese el número de póliza a consultar (ver Figura 43).



Figura 43. Lista de tipos de póliza y diálogo de ingreso de número de póliza (Fuente: elaborado por el autor).

Después de ingresada la información solicitada por el chatbot, este procesa la información y despliega la información de la póliza consultada, como se observa en la Figura 44.



Figura 44. Información desplegada de póliza consultada y encontrada (Fuente: elaborado por el autor).

Luego de mostrar la información de la póliza consultada, el chatbot muestra las opciones correspondientes al módulo de renovaciones, las cuales son: renovar cancelar, ingresar días de vigencia (ver Figura 45).



Figura 45. Opciones para realizar la renovación desde el chatbot (Fuente: elaborado por el autor).

Cuando el usuario selecciona renovar, el chatbot realiza todo el proceso que implica realizar la renovación de una póliza. Por la cantidad de información que este proceso maneja, el chatbot muestra un mensaje informando al usuario que esta operación va a tardar un tiempo (ver Figura 46).



Figura 46. Mensaje de espera del chatbot (Fuente: elaborado por el autor).

Una vez realizada la renovación, el chatbot muestra un resumen de la póliza renovada, como se puede observar en la Figura 47.

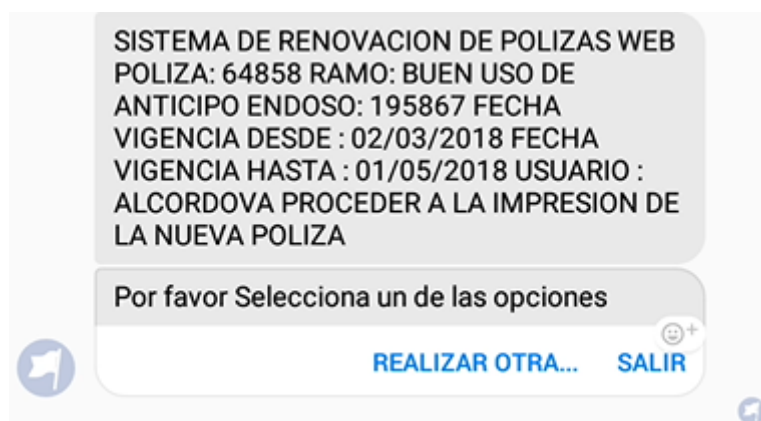


Figura 47. Resumen de póliza renovada (Fuente: elaborado por el autor).

3.2 EVALUACION DE SATISFACCION Y EFICIENCIA DE LA APLICACIÓN

La aplicación será evaluada a través de una encuesta para obtener una estadística en función de la satisfacción y eficiencia. La encuesta será aplicada a un grupo de usuarios (Brokers, clientes) del caso de estudio, los cuales usaron la aplicación.

3.2.1 ESTRUCTURA DE LA ENCUESTA

La encuesta estará dividida en dos secciones, la primera para evaluar la efectividad del chatbot y la segunda, para evaluar la satisfacción de usuario. La respuesta de cada pregunta será dimensionada en un rango del 1 al 5, donde 1 es el valor mínimo y 5 el valor máximo.

Para facilitar la recolección de datos de la encuesta, se utilizó un formulario online como se puede observar en la Figura 48 y 49.

SECCIÓN DE EFECTIVIDAD

Esta Sección se basa en tabular la eficiencia del chatbot

El chatbot proporciona el servicio que el usuario solicita?

1

2

3

4

5

El tiempo de respuesta del chatbot es optimo?

1

2

3

4

5

La información presentada por el chatbot es fiable?

1

2

3

4

5

Figura 48. Sección de efectividad del chatbot (Fuente: elaborado por el autor).

SECCIÓN DE SATISFACCIÓN DE USUARIO

Esta Sección se basa en tabular la satisfacción del usuario con respecto al uso del chatbot

El chatbot presenta un dialogo amigable e intuitivo?

1

2

3

4

5

Que tan efectiva es la comunicación en lenguaje natural con el chatbot?

1

2

3

4

5

El chatbot cumple con las expectativas de uso?

1

2

3

4

5

Recomendaría el uso del chatbot a otros usuarios?

1

2

3

4

5

Figura 49. Sección de satisfacción de usuario (Fuente: elaborado por el autor).

3.2.2 ANÁLISIS DE RESULTADOS DE LA ENCUESTA

Los resultados serán evaluados de la siguiente manera: resultado por pregunta, resultado por categoría, y resultado global de la encuesta.

3.2.2.1 RESULTADO POR PREGUNTA

La Figura 50, muestra los resultados obtenidos en cada pregunta de la encuesta aplicada a los usuarios del chatbot.

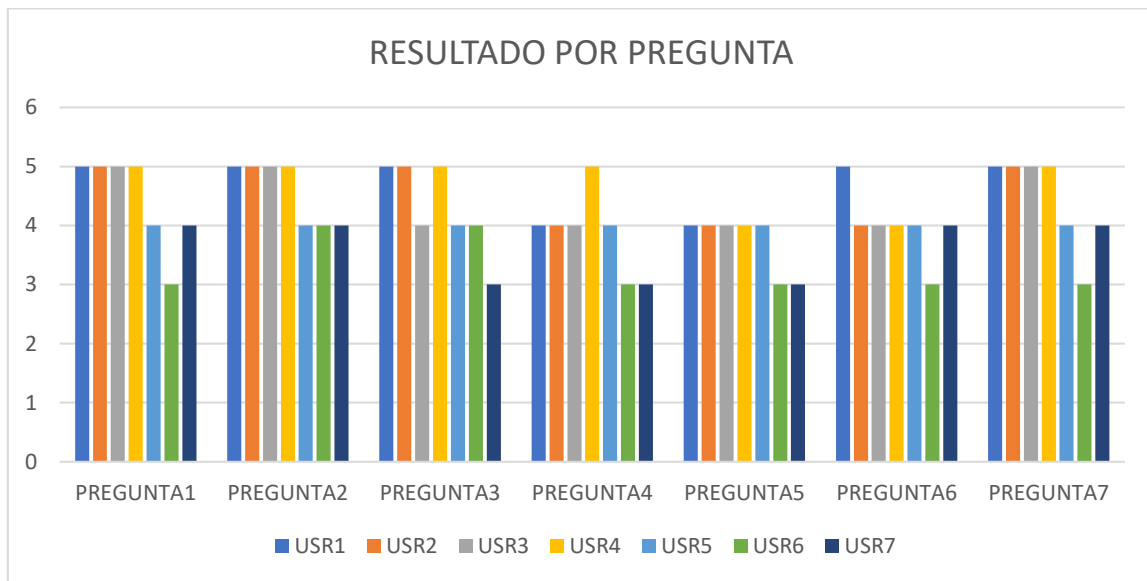


Figura 50. Resultado por pregunta (Fuente: elaborado por el autor).

El porcentaje que tiene cada pregunta se observa en la Figura 51. El mayor porcentaje de aceptación la tiene la pregunta 2 que corresponde al tiempo de respuesta del chatbot con un 91.42% y el menor porcentaje de aceptación la tiene la pregunta 5 que corresponde a qué tan efectiva es la comunicación en lenguaje natural con el chatbot, con un 74.28%.

Nota: El tiempo de respuesta es mediante una comparación entre el tiempo de respuesta de la página web de fianzas del caso de estudio y el chatbot, con respecto a un mismo servicio. Los resultados de esta comparación quedan al criterio del usuario (Broker, Cliente) quien utiliza las dos aplicaciones.

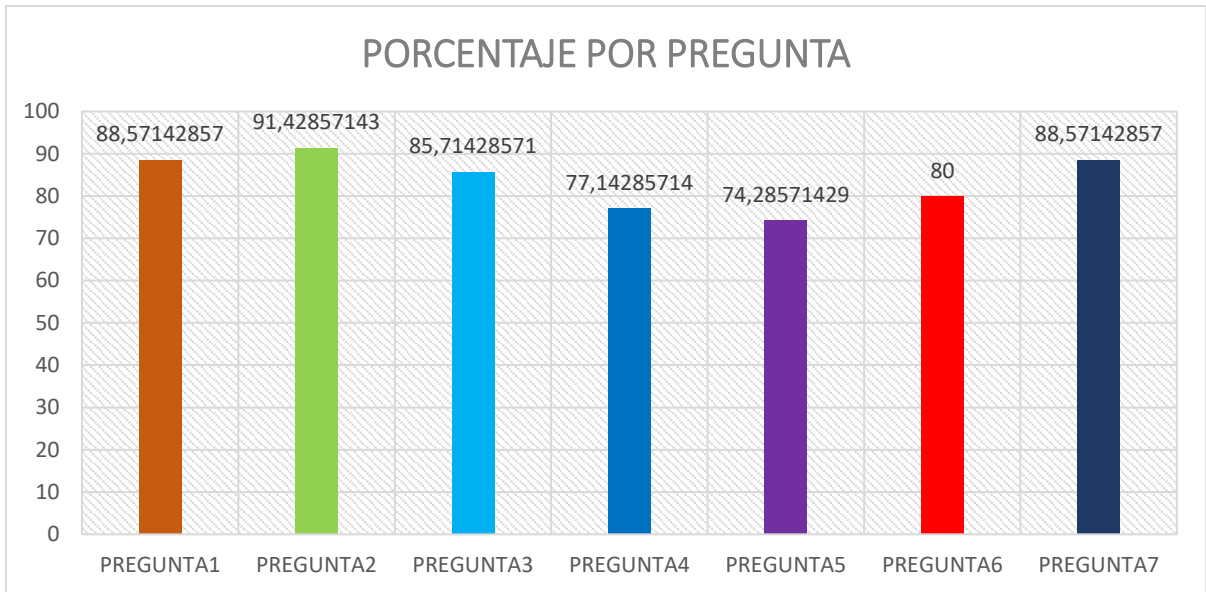


Figura 51. Porcentaje de aceptación por pregunta (Fuente: elaborado por el autor).

3.2.2.2 RESULTADO POR CATEGORÍA

En la Figura 52 se observa el resultado de cada categoría (Satisfacción, Efectividad) de la encuesta.

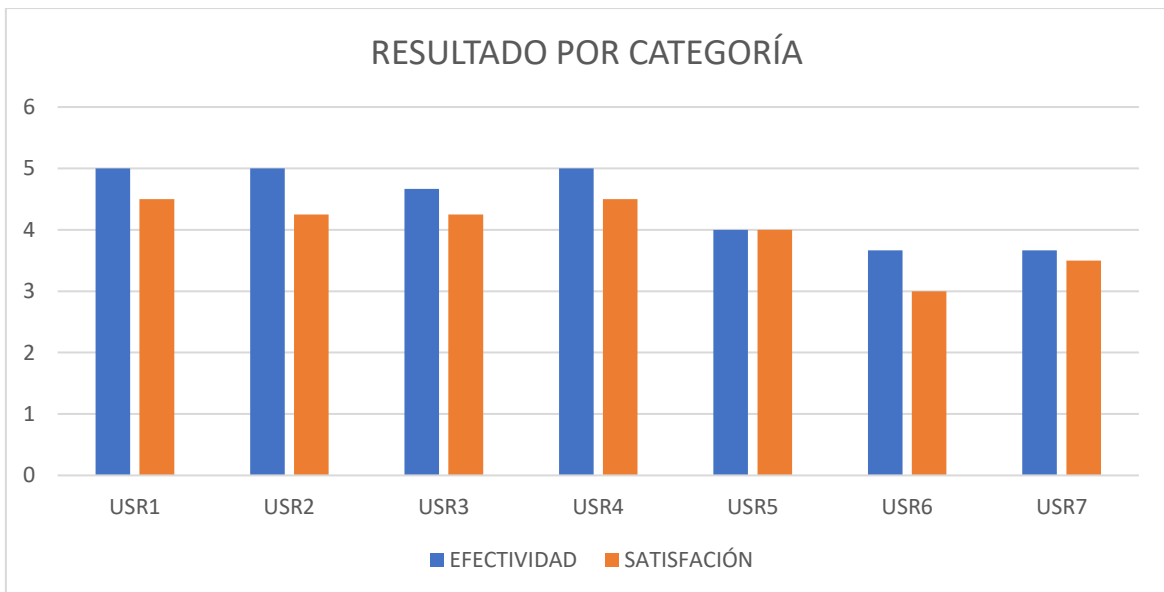


Figura 52. Resultado por categoría (Fuente: elaborado por el autor).

El porcentaje que tiene cada categoría puede verse en la Figura 53. La categoría mejor valorada es la efectividad del chatbot con 88.57%, por lo cual la categoría con menor valoración es la satisfacción con un 80%.

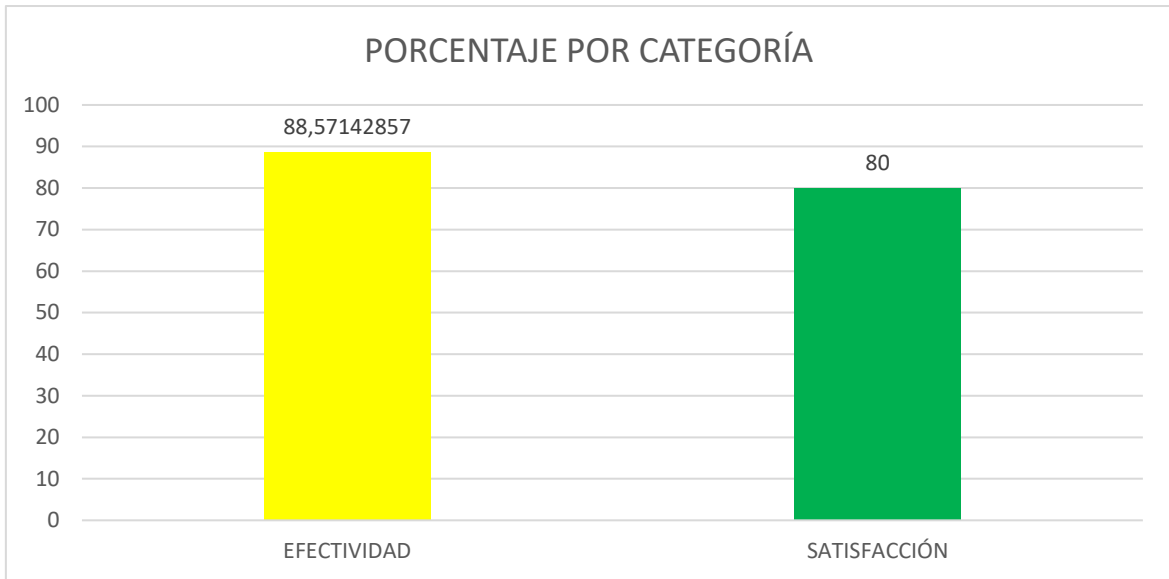


Figura 53. Porcentaje por categoría (Fuente: elaborado por el autor).

3.2.2.3 RESULTADO GLOBAL DE LA ENCUESTA

En la Figura 54 se muestra el resultado global de la encuesta. En esta gráfica se observa el desempeño general del chatbot según los encuestados.

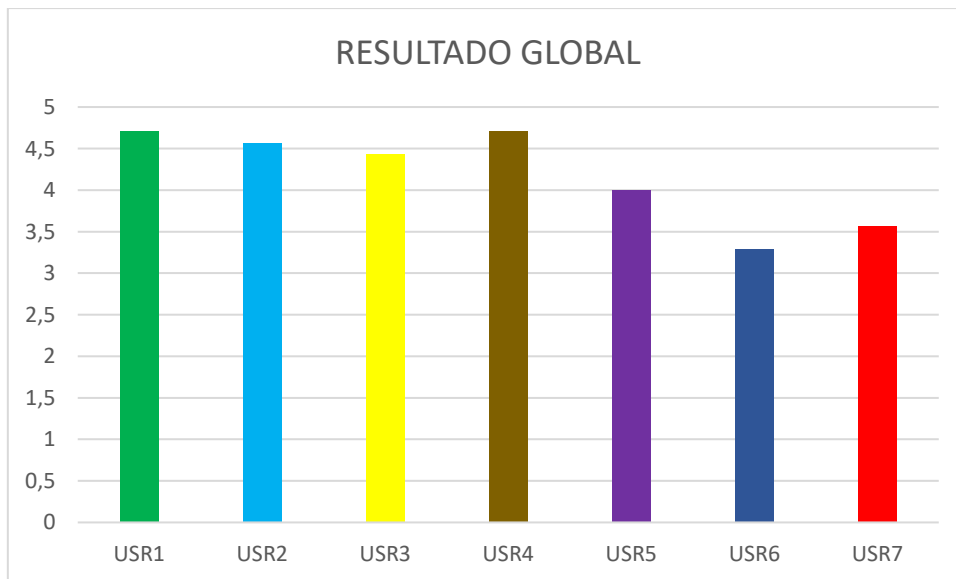


Figura 54. Resultado global de la encuesta (Fuente: elaborada por el autor).

El porcentaje de desempeño global que se obtuvo al final de la encuesta es del 83.67% (ver Figura 55).

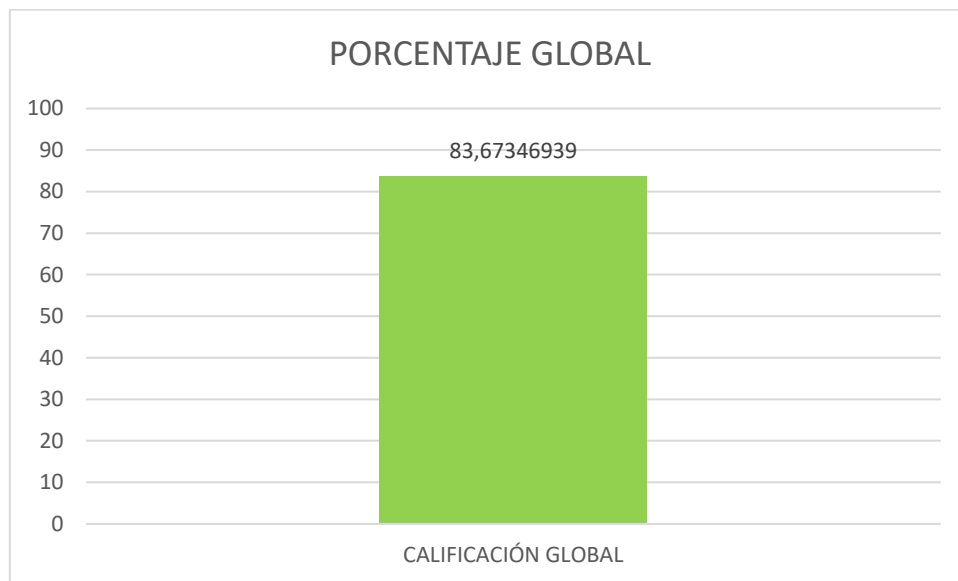


Figura 55. Porcentaje global de los resultados de la encuesta (Fuente: elaborada por el autor).

4 CONCLUSIONES

La innovación con chatbots representa una nueva forma en que las empresas brindan atención al cliente. Un chatbot transforma la interacción entre el usuario y un computador, ejecutando una serie de tareas a través de una conversación, dejando de lado a las interfaces tradicionales.

Para implementar un chatbot institucional es necesario analizar los procesos de las organizaciones, para la comprensión del trabajo y expresiones de las personas. Con este paso, se ha cumplido con el objetivo general propuesto de una forma integral y con resultados optimistas para la organización.

Con los servicios C3gnitivos QnaMaker y LUIS se logró simular una conversaci3n m3s natural entre el chatbot y el usuario, adem3s de tener una idea m3s clara del funcionamiento de los servicios especializados en el campo de la IA.

La implementaci3n del chatbot en el caso de estudio, mejora los siguientes aspectos:

- La carga operativa del personal de fianzas de Seguros Equinoccial disminuye, por el hecho de que los brokers y clientes hacen sus solicitudes y renovaciones de forma guiada y directa a trav3s del chatbot, sin tener la necesidad de comunicarse con un ejecutivo de fianzas para que lo ayude.
- Los clientes de Seguros Equinoccial tienen un nuevo medio para acceder a los servicios de fianzas.
- Los tiempos de operaci3n en renovaciones de p3lizas disminuye.
- Hay m3s inter3s en la implementaci3n de chatbots para otras 3reas del caso de estudio.

5 REFERENCIAS

- [1] S. Fitriane and L. Rothkrantz, "My_Eliza, A multimodal communication system," *Delft University of Technology*, 2002.
- [2] R. P. Díez, A. G. Gómez, and N. de Abajo Martínez, *Introducción a la inteligencia artificial: sistemas expertos, redes neuronales artificiales y computación evolutiva*. Universidad de Oviedo, 2001.
- [3] E. Castillo, J. M. Gutiérrez, and A. S. Hadi, "Sistemas expertos y modelos de redes probabilísticas," *Academia de Ingeniería*, 1997.
- [4] A. Simon, M. Singh Deo, V. Selvam, and R. Babu, *An Overview of Machine Learning and its Applications*. 2016, pp. 22-24.
- [5] D. A. Zamora Ávila, "Métodos Machine Learning aplicados para estimar la concentración de los contaminantes de la DQO y de los SST en hidrosistemas de saneamiento urbano a partir de espectrometría UV-Visible," 2013.
- [6] A. Gelbukh, "Procesamiento de lenguaje natural," *S. d. Virtual, Ed*, 2010.
- [7] A. C. Vásquez, J. P. Quispe, and A. M. Huayna, "Procesamiento de lenguaje natural," *Revista de investigación de Sistemas e Informática*, vol. 6, no. 2, pp. 45-54, 2009.
- [8] S. A. Abdul-Kader and J. Woods, "Survey on chatbot design techniques in speech conversation systems," *International Journal of Advanced Computer Science and Applications*, vol. 6, no. 7, pp. 72-80, 2015.
- [9] J. C. Cobos Torres, "Integración de un Chatbot como habilidad de un robot social con gestor de diálogos," ESPAÑA/Facultad de Ingeniería de Sistemas y Automática-Universidad Carlos III de Madrid/2013, 2013.
- [10] A. Augello, G. Pilato, A. Machi, and S. Gaglio, "An approach to enhance chatbot semantic power and maintainability: experiences within the FRASI project," in *Semantic Computing (ICSC), 2012 IEEE Sixth International Conference on*, 2012, pp. 186-193: IEEE.

- [11] J. M. Rodríguez, H. Merlino, and E. Fernández, "Comportamiento Adaptable de Chatbots Dependiente del Contexto," *Revista Latinoamericana de Ingeniería de Software*, vol. 2, no. 2, pp. 115-136, 2014.
- [12] C. Machinery, "Computing machinery and intelligence-AM Turing," *Mind*, vol. 59, no. 236, p. 433, 1950.
- [13] P. Espeso. (2016, 01/07/2018). *Deep blue el ordenador con una sola misión ganar al humano*. . Available: <https://www.xataka.com/otros/deep-blue-el-ordenador-con-una-sola-mision-ganar-al-humano>
- [14] M. Alfonseca, "¿ Basta la prueba de Turing para definir la "inteligencia artificial"?", *Scientia et Fides*, vol. 2, no. 2, pp. 129-134, 2014.
- [15] E. Ackerman, "Can Winograd Schemas Replace Turing Test for Defining Human-Level AI?," *IEEE Spectrum: Technology, Engineering, and Science News*, 2014.
- [16] W. Xiong *et al.*, "Achieving human parity in conversational speech recognition," *arXiv preprint arXiv:1610.05256*, 2016.
- [17] S. Etlinger, "THE CONVERSATIONAL BUSINESS," 2017.
- [18] J. Pereira, H. Medina, and Ó. Díaz, "Uso de Chatbots en la Docencia Universitaria."
- [19] planetachatbot. (2017, 05 de Noviembre). *Las diez características que todo Chatbot decente debe tener*. Available: <https://planetachatbot.com>
- [20] J. M. Gutiérrez, "Sistemas Expertos Basados en Reglas," *Recurso personal. Dpto. de Matemática Aplicada. Universidad de Cantabria*, pp. 1-12, 2008.
- [21] Microsoft. (2018, 11/18/2017). *Bot Framework*. Available: <https://docs.microsoft.com/en-us/bot-framework/>
- [22] MasterCard. (2018). *Welcome to the Engagement Bureau*. Available: <https://newsroom.mastercard.com/press-releases/mastercard-makes-commerce-more-conversational-with-launch-of-chatbots-for-banks-and-merchants/>
- [23] B. mundo. (2018, 01/10/2018). *BBC mundo*. Available: <https://www.facebook.com/bbcmundo/>
- [24] C. CHOCOLATE. (2018, 5 de Enero). *CHATBOT CHOCOLATE*. Available: <http://chatbotchocolate.com/>
- [25] C. Chocolate. (2018, 5 de Enero). *Billy Seguros*. Available: <https://billyseguros.com>
- [26] futurizable. (2017, 10 de Diciembre). *Estado del arte en el desarrollo de chatbots a nivel mundial*. Available: <https://futurizable.com/chatbot>

- [27] IBM. (2018, 11/20/2017). *Watson*. Available:
<https://www.ibm.com/watson/services/conversation/>
- [28] Venturebeat. (2018, 11 de Noviembre). *Natural Language for Developers*. Available: <https://wit.ai/>
- [29] Amazon. (2018, 22 de Enero). *Chatbots and AWS*. Available:
<https://aws.amazon.com/es/chatbots-in-call-centers/>
- [30] Microsoft. (2018, 25 de Noviembre). *QnaMaker*. Available:
<https://qnamaker.ai/Documentation>
- [31] Microsoft. (2018, 02 de Octubre). *Language Understanding (LUIS)*. Available:
<https://docs.microsoft.com/en-us/azure/cognitive-services/LUIS/Home>
- [32] P. Agiles. (2018, 5 de Noviembre). *SCRUM*. Available:
<https://proyectosagiles.org/>
- [33] M. A. Alvarez, «DesarrolloWeb.com,» 02 01 2014. [En línea]. Available:
<https://desarrolloweb.com/articulos/que-es-mvc.html>. [Último acceso: 06 03 2018].

ANEXOS

ANEXO A

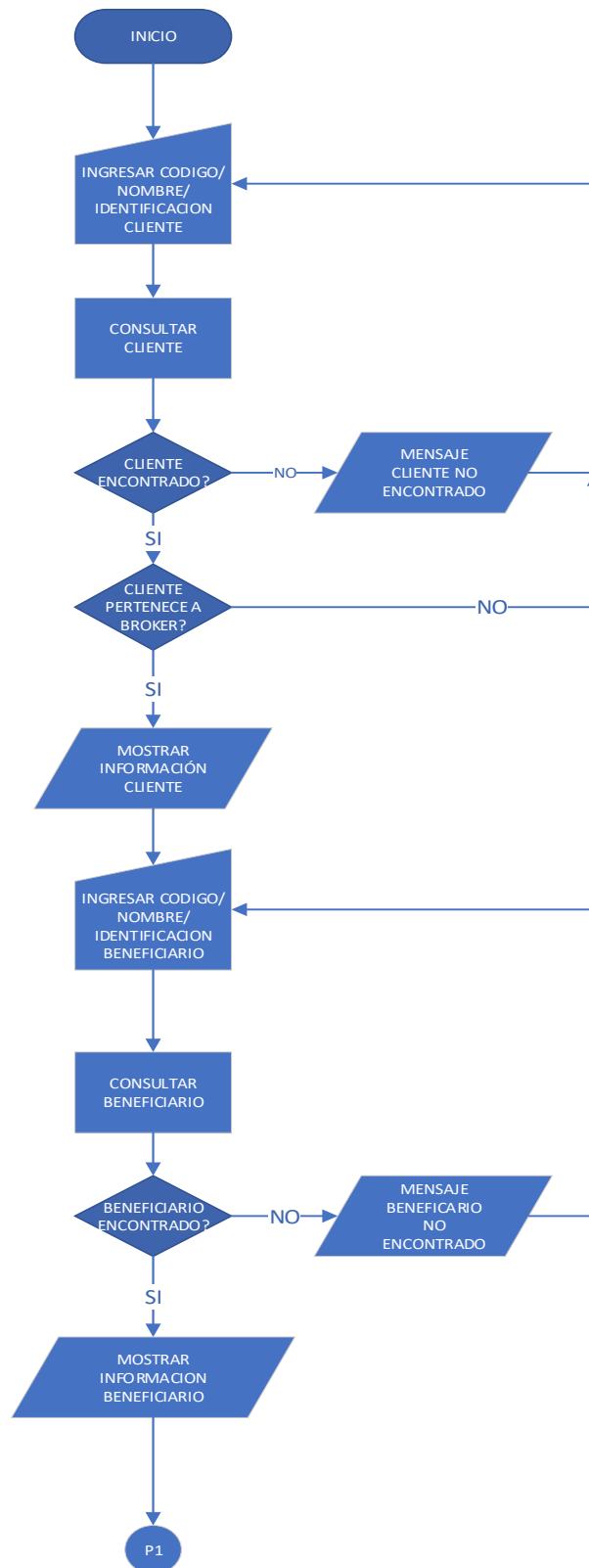


Figura 56. Diagrama de flujo proceso de solicitud de póliza nueva parte A (Fuente: elaborado por el autor).

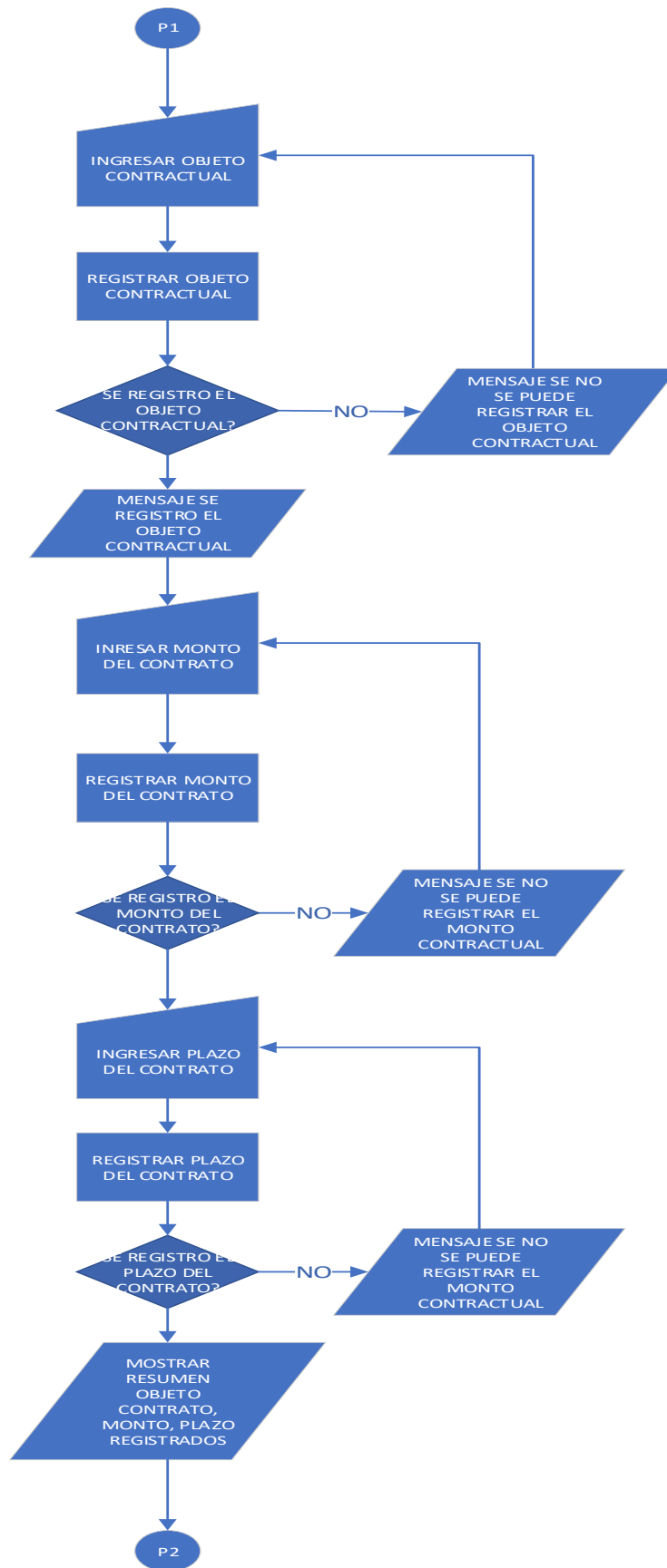


Figura 57. Diagrama de flujo proceso de solicitud de póliza nueva parte B (Fuente: elaborado por el autor).

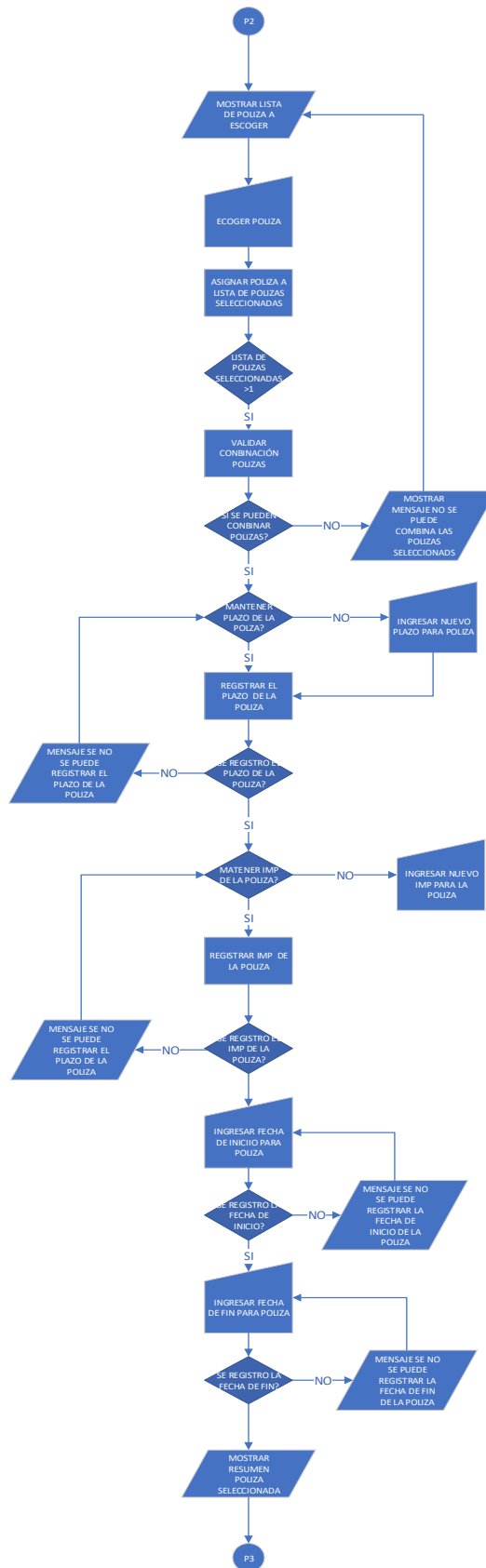


Figura 58. Diagrama de flujo proceso de solicitud de póliza nueva parte C (Fuente: elaborado por el autor).

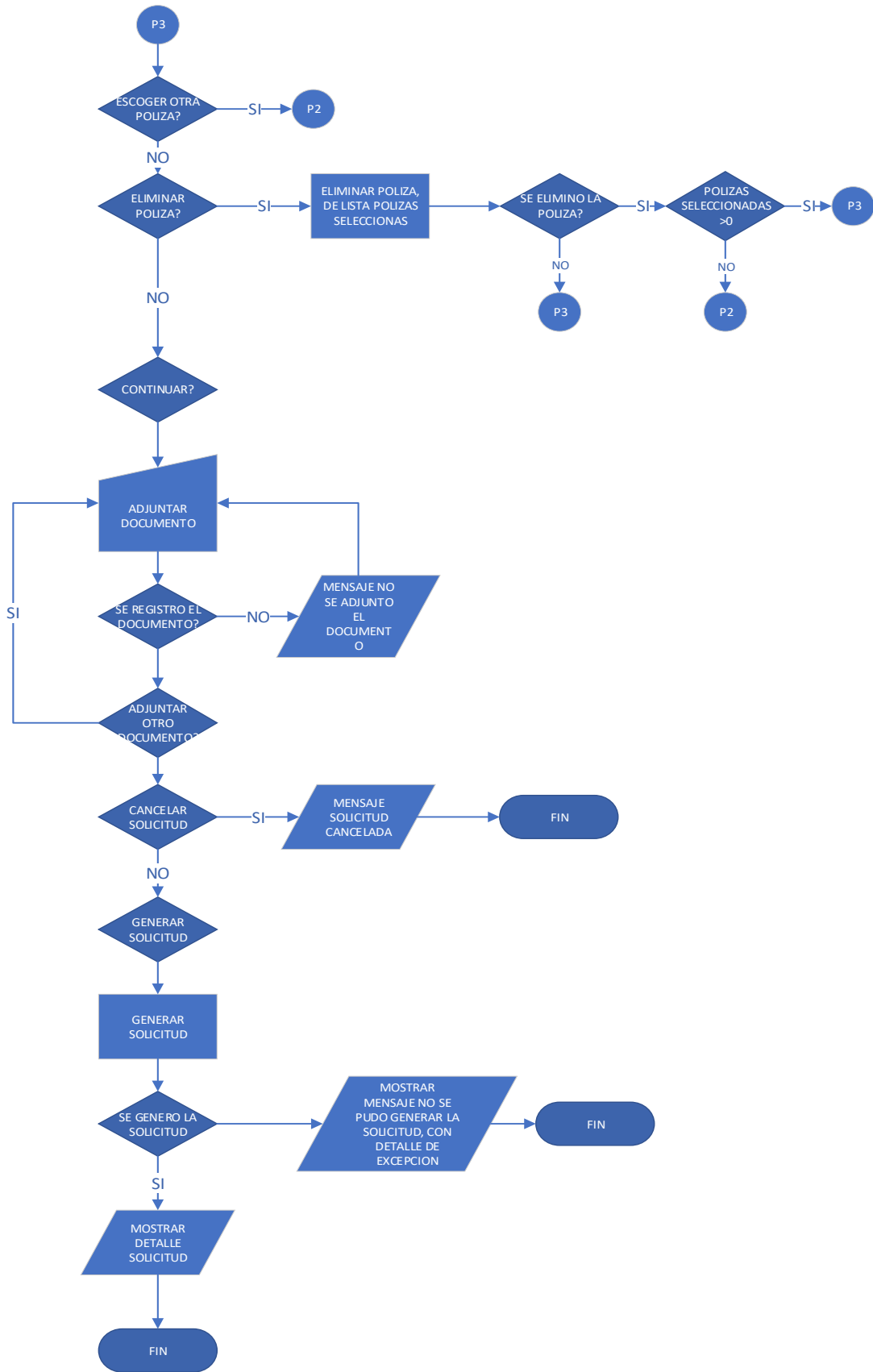


Figura 59. Diagrama de flujo proceso de solicitud de póliza nueva parte D (Fuente: elaborado por el autor).

ANEXO B

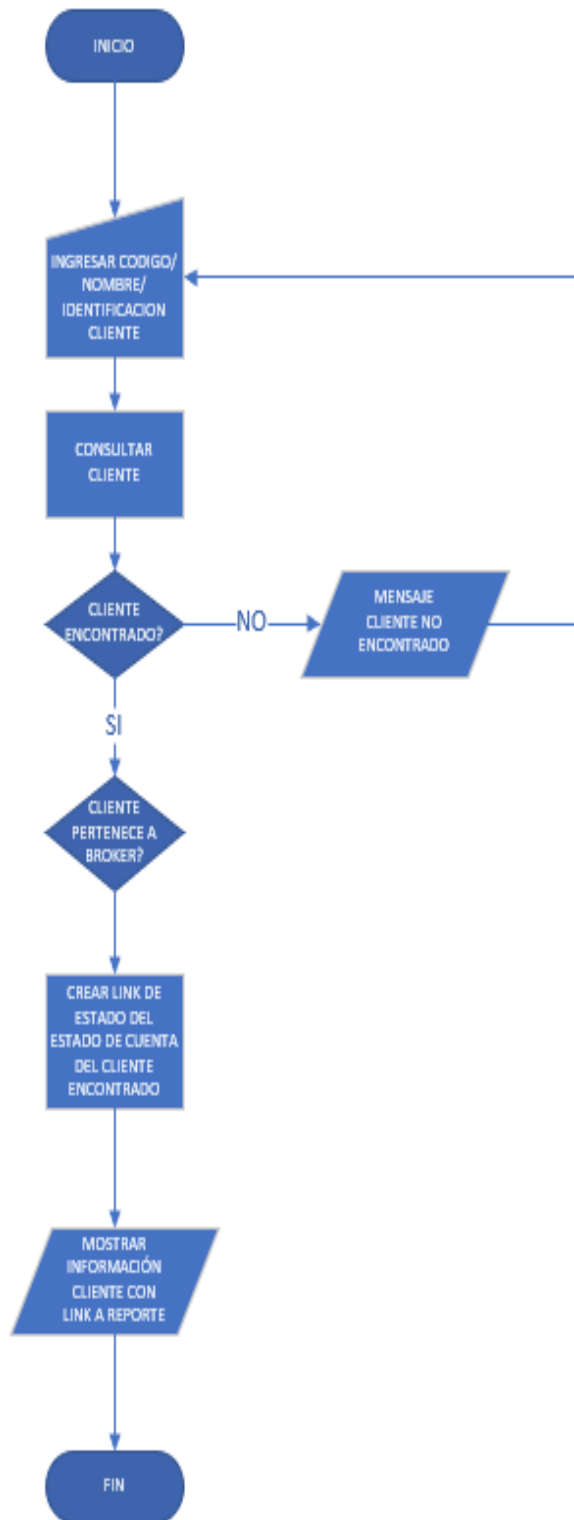


Figura 60. Diagrama de flujo proceso de consulta de estados de cuenta (Fuente: elaborado por el autor).

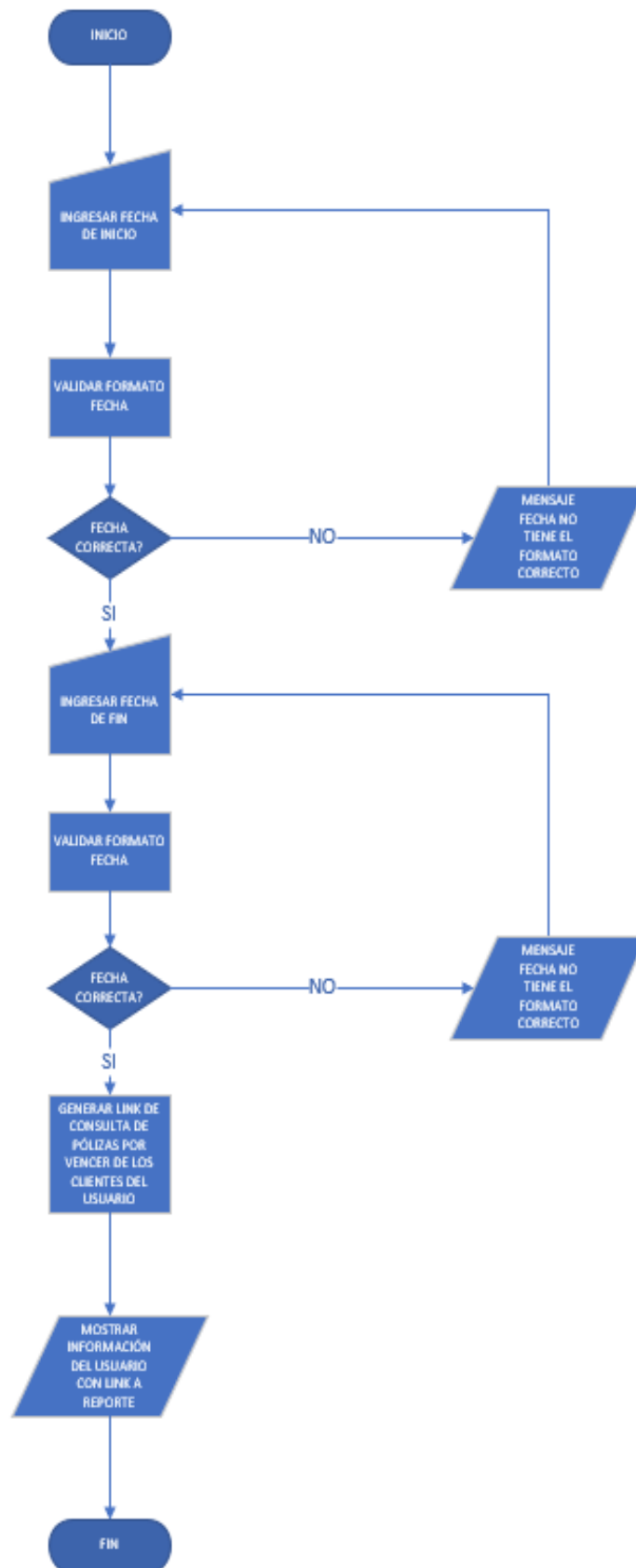


Figura 61. Diagrama de flujo proceso de consultas de polizas por vencer (Fuente: elaborado por el autor).

ANEXO C

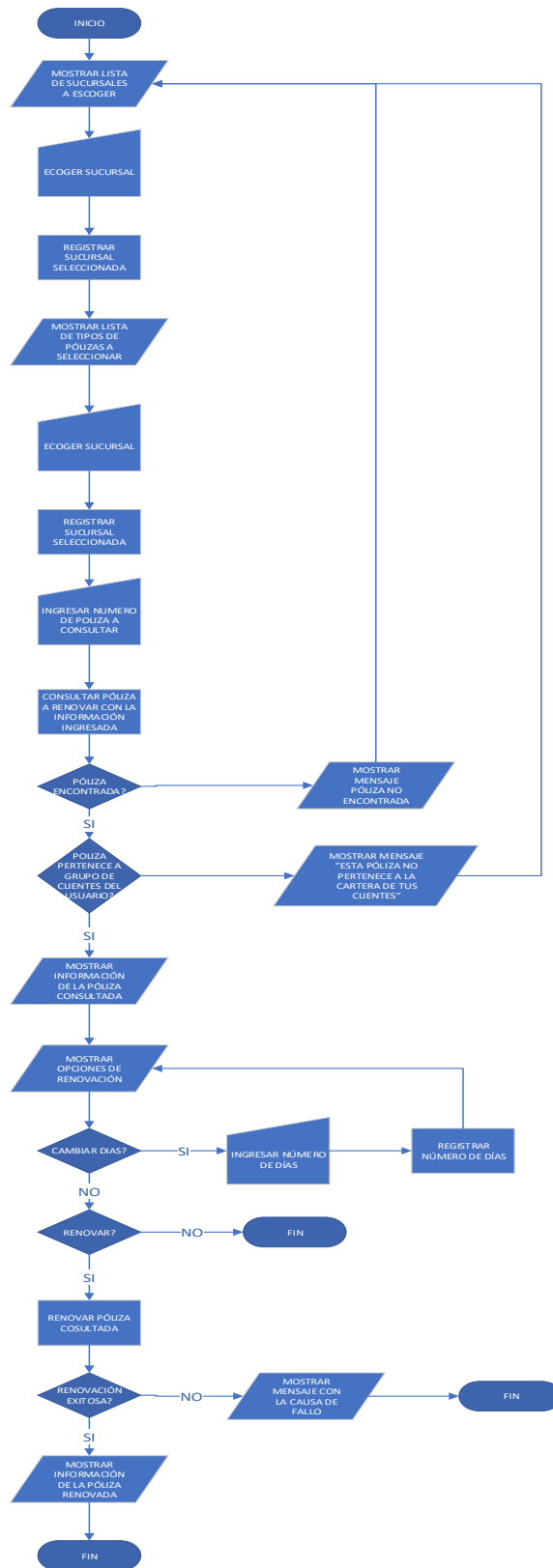


Figura 62. Diagrama de flujo proceso de renovación de póliza (Fuente: elaborado por el autor).