

UNIVERSIDAD NACIONAL
Facultad de Ciencias Exactas y Naturales
ESCUELA DE INFORMÁTICA



**“Propuesta de un chatbot utilizando herramientas de software libre para mejorar el
proceso de comunicación”**

**Para optar al grado de Licenciado en Informática
con énfasis en Sistemas de Información**

Ing. Alexandra Aguilar Nájera
Ing. Daydalin Yariela Hernández Artola

Heredia, Costa Rica

Dedicatoria

En primer lugar, dedico este trabajo a Dios y agradezco que me haya permitido concluirlo y alcanzar esta meta por medio de perseverancia, sabiduría y esfuerzo.

A mí mamá y mi abuela por todo el sacrificio y esfuerzo que han hecho para que esto se lleve a cabo, ya que han estado en todo momento apoyándome en cada una de las metas que me he propuesto. Y a mi familia que siempre han sido parte importante de mi vida.

Y muchas gracias todas las personas involucradas que de alguna u otra manera nos han colaborado con sus consejos y recomendaciones, siempre impulsándonos a seguir adelante.

Ing. Yariela Hernández Artola

Dedicatoria

Dedico este proyecto, primeramente, a Dios por permitirme concluir esta etapa profesional y de aprendizaje. A mi familia, la cual me ha brindado su apoyo en todo momento y han formado parte de este proceso.

A mi esposo el cual ha estado a mi lado incondicionalmente y me ha brindado el soporte para seguir adelante. Por último, a mis compañeros y amigos los cuales me han brindado su ayuda y buenos deseos.

Ing. Alexandra Aguilar Nájera

Reconocimientos y Agradecimientos

Agradecemos a Dios y nuestras familias por su apoyo incondicional y a todo el personal administrativo y docente de la Universidad Nacional, ya que han contribuido a nuestra formación académica. Al profesor Felipe Ovarés tutor del proyecto por guiarnos en este proceso, por apoyarnos. Gracias por todos los aportes y observaciones que nos hicieron para que el proyecto se realizara con éxito.

Resumen del Proyecto

El proyecto de graduación “*Propuesta de un chatbot utilizando herramientas de software libre para mejorar el proceso de comunicación*”, consiste en el desarrollo de un sistema (chatbot). Un chatbot es un programa computacional que interactúa con personas utilizando lenguaje natural, el cual brinda respuestas a consultas realizadas por el usuario de forma automática, este permitirá a los estudiantes hacer preguntas relacionados a temas de información general de la escuela de informática de la Universidad Nacional y este le brindará respuesta. Este sistema se realizará utilizando herramientas de software libre.

Los estudiantes podrán acceder al chatbot por medio de una dirección web y realizar preguntas sobre procesos de la escuela, horas asistentes, matrícula, beca entre otros. El sistema responderá esas consultas, si el chatbot no tuviera la respuesta, mostrará un mensaje “No se encontró su respuesta”, sin embargo, cuenta con una opción donde el estudiante podrá escribir la pregunta y enviarla a los administradores.

Los chatbot cuentan con diferentes beneficios, una de ellas es la disponibilidad, ya que trabaja las 24 horas del día, los 7 días de la semana y puede atender la consulta de varias personas a la misma vez. También, los chatbots son fáciles de usar y acceder, así que cualquier persona puede utilizarlo, ya sea padre de familia, profesor o un estudiante. Dado lo anterior el chatbot será un sistema de apoyo para la escuela de informática, en especial en fechas de alta demanda de atención como lo son procesos de graduación y matrícula.

Además, el chatbot ayudará a mejorar el proceso de comunicación entre los estudiantes y la institución ya que será otro medio de comunicación que tendrá disponible el estudiantado. El sistema permitirá generar una relación más transparente entre ambas partes y a su vez ayudar al personal de atención de la escuela en su día a día.

Palabras Claves

Chatbot, Comunicación, Distancia de Leveshtein, Flask, Mongo DB, SCRUM, Software libre, Python.

Índice General

1.	Descripción de la investigación	10
1.1.	Antecedentes y problemática	10
1.2.	Justificación	13
1.3.	Descripción del proyecto	15
1.4.	Objetivos	16
1.4.1.	Objetivo general:	16
1.4.2.	Objetivos específicos:	16
1.4.3.	Relación objetivos, productos e indicadores de logro por producto	17
1.5.	Áreas de conocimiento	17
1.6.	Alcance del proyecto	18
1.7.	Beneficios esperados	19
1.8.	Resultados o productos esperados	20
1.9.	Riesgos potenciales del proyecto	21
2.	Marco teórico	22
2.1.	Marco referencial	22
2.1.1.	Misión de la Escuela de Informática	22
2.1.2.	Visión de la Escuela de Informática	22
2.1.3.	Objetivos estratégicos	22
2.1.4.	Fortalezas	23
2.2.	Marco conceptual	24
3.	Marco metodológico	27
3.1.	Product backlog o pila de producto	28
3.2.	Sprint planning	28
3.3.	Sprint	28
3.4.	Sprint backlog	29
3.5.	Daily meeting	29
3.6.	Sprint review	29
3.7.	Sprint retrospective	29
3.8.	Roles	29
4.	Metodología de desarrollo	31

4.1.	Resumen ejecutivo de costos	32
4.2.	Análisis de herramientas	34
4.2.1.	Back-End	34
4.2.1.1.	Python	34
4.2.1.2.	Node	35
4.2.1.3.	Java	35
4.2.1.4.	PHP	36
4.2.2.	Front-End	37
4.2.2.1.	JavaScript	37
4.2.2.2.	Angular	37
4.2.2.3.	HTML	38
4.2.3.	Gestores de Bases de Datos	39
4.2.3.1.	MySQL	39
4.2.3.2.	MongoDB	40
4.2.3.3.	Postgres	41
4.3.	Modelado	43
4.4.	Desarrollo	44
4.4.1.	Diseño y arquitectura	44
4.4.1.1.	MongoDB	46
4.4.1.2.	Python	46
4.4.1.3.	Flask	47
4.4.1.4.	Fuzzywuzzy	47
4.4.1.5.	JQuery	48
4.4.1.6.	Bootstrap	48
4.4.2.	Desarrollo del proyecto	48
4.4.3.	Implementación	52
4.4.3.1.	Chatbot	53
4.4.3.1.1.	Pantalla principal del chatbot	53
4.4.3.1.2.	Agregar nuevas preguntas	54
4.4.3.1.3.	Ayuda	54
4.4.3.2.	Mantenimiento del chatbot	56
4.4.3.2.1.	Ingreso al sistema	56

4.4.3.2.2.	Olvidaste contraseña	57
4.4.3.2.3.	Agregar preguntas	57
4.4.3.2.4.	Agregar usuarios	58
4.4.3.2.5.	Mantenimiento de preguntas	59
4.4.3.2.6.	Actualizar preguntas	60
4.4.3.2.7.	Mantenimiento de usuarios	61
4.4.3.2.8.	Actualizar usuarios	62
4.5.	Evaluación de la herramienta	64
4.5.1.	Análisis de resultados	65
4.5.1.1.	Estudiantes	65
4.5.1.2.	Personal administrativo	69
5.	Análisis retrospectivo	70
5.1.	Objetivo general:	70
5.2.	Objetivos específicos:	71
6.	Conclusiones y recomendaciones	72
6.1.	Conclusiones	72
6.2.	Recomendaciones	73
7.	Referencias bibliográficas	74
8.	Anexo	79
9.	Apéndices	79
a.	Manual de usuario de chatbot y mantenimiento	79
b.	Manual técnico	79
c.	Formulario “Chatbot”	79
d.	Formulario “Chat Proyecto Piloto”	79
e.	Plan de pruebas	79
10.	Glosario	80

Índice de Tablas

Tabla 1. <i>Objetivos y Producto</i>	17
Tabla 2. <i>Los Riesgos, Probabilidad y acciones</i>	21
Tabla 3. <i>Detalle del tiempo invertido</i>	33

Índice de Figuras

Figura 1: Proceso de SCRUM	28
Figura 2: Diagrama de clases	45
Figura 3: Diagrama de Arquitectura	45
Figura 4: Definición entre la distancia de dos cadenas de caracteres	49
Figura 5: Formula para calcular el radio de similitud	50
Figura 6: Método de fuzz_token_set_ratio	50
Figura 7: Pantalla principal del chatbot	53
Figura 8: Agregar nuevas preguntas	54
Figura 9: Apartado de ayuda	55
Figura 10: Ingreso al mantenimiento de chatbot	56
Figura 11: Olvidaste contraseña	57
Figura 12: Agregar preguntas	58
Figura 13: Agregar usuario	59
Figura 14: Mantenimiento de preguntas	60
Figura 15: Actualizar preguntas	61
Figura 16: Mantenimiento de usuarios	62
Figura 17: Actualizar usuario	63
Figura 18: Pantalla principal del mantenimiento del chatbot	64
Figura 19: Nivel académico de los estudiantes encuestados	66
Figura 20: Cantidad de preguntas realizadas a los estudiantes	67
Figura 21: Cantidad de preguntas respondidas por el chatbot	67
Figura 22: Cantidad de preguntas contenidas en el chatbot	68

1. Descripción de la investigación

1.1. Antecedentes y problemática

Hoy en día la Universidad Nacional es una de las universidades más grandes del país, la cual tiene una gran población estudiantil. Según artículos reportados por la institución en el 2015 la universidad contaba con más de 19189 estudiantes en donde 3994 de ellos eran de nuevo ingreso. (Universidad Nacional Costa Rica [UNA], 2016). La población estudiantil de la UNA se divide en estudiantes de diplomado, bachillerato, licenciatura, maestría y doctorado. En el 2015 según publicaciones de la universidad se graduaron más de 3000 estudiantes en donde 597 eran de diplomado y profesorados, 1520 de bachillerato, 681 de licenciatura y 282 de masters-doctorados (UNA, 2016).

Parte de la población estudiantil descrita anteriormente pertenece a la Escuela de Informática, la cual en la actualidad tiene aproximadamente 1505 estudiantes en el Campus Benjamín Núñez. Esta cantidad de estudiantes está distribuida en los distintos grados académicos que ofrece la institución desde diplomado hasta maestría.

Los funcionarios de la Universidad Nacional y en este caso el personal de la Escuela de Informática cumple como parte de sus tareas diarias la atención a los estudiantes, la cual incluye dar respuesta a dudas y consultas generadas por el estudiantado, las cuales aumentan en ciertas épocas del año por los procesos institucionales como por ejemplo matrícula y graduación. Las preguntas realizadas en estas épocas suelen ser repetitivas, relacionadas a temas de información general.

En estos procesos, los estudiantes suelen realizar preguntas con relación a la matrícula y graduación que corresponde: ¿dónde encuentro las guías de horario?, ¿cuáles son las fechas de matrícula?, ¿cómo puedo des matricular un curso?, ¿cómo hacer un sobrepaso?, ¿cuál es el periodo de retiro de cursos?, ¿cuál es el proceso para graduarse?, ¿cuál es la fecha de graduación? Por otra parte, existen otras preguntas que, aunque no pertenecen a estos dos procesos, el estudiantado suele realizar, como, por ejemplo: ¿cuál es el horario de las busetas del Campus Benjamín Núñez al campus Omar Dengo?, ¿cuál es el proceso de ampliación o apertura de cursos?, ¿cuál es el calendario de pago de créditos? Asimismo, se puede mencionar

otras áreas de consulta como lo son becas, información de la institución, pago de créditos, proceso de certificaciones, entre otros.

La Universidad Nacional cuenta con cuatro canales de comunicación: página web, correos electrónicos, redes sociales y recepción de llamada telefónicas (UNA, 2018); todos los medios antes mencionados cuentan con diferentes limitaciones. En el caso de la Escuela de Informática los medios de comunicación directa con los estudiantes son las redes sociales, email y llamadas telefónicas, ya que el sitio web es una herramienta de información general donde se especifican planes de estudio, teléfonos, misión y visión. Sin embargo, esta no cuenta con información de procesos institucionales, con área de preguntas frecuentes ni noticias estudiantiles que les permita a los estudiantes estar informados.

Otra limitante de los medios de comunicación actuales son los horarios de atención por parte de los departamentos de la institución, ya que la mayoría de ellos cuenta con un horario de oficina de 7:30am u 8:00 am a 4:30pm o 5:00 pm de lunes a viernes, donde el horario no es continuo. Un ejemplo de esta situación es el departamento de Registro y Financiero, donde según el horario de atención que se muestra en su página web es de lunes a viernes de 8:00am a 11:00 am y de 1:00 a 4:30 pm, en donde se cierra de 11:30am a 1:00 pm, por hora de almuerzo todos los días, siendo el miércoles el único día de jornada continua (UNA, 2015).

Asimismo, las Escuelas también tienen un problema similar al de los departamentos con respecto a la hora de atención, por ejemplo, la Escuela de Matemática utiliza un horario parecido al de los departamentos el cual es de lunes a viernes de 8:00 am - 12:00 md y de 1:00 pm a 5:00pm. Por otro lado, la Escuela de Informática trabaja con un horario extendido que va de 7:30 am a 7:30pm. El problema que presenta este horario son las consultas que pueden ser realizadas en la jornada de la tarde noche, cuya respuesta la tenga el personal que trabaja en la jornada diurna o la persona que atiende la consulta en ese momento no conozca la respuesta. En este caso el estudiante debe de llamar el día siguiente por su respuesta, dado que hasta el otro día el personal que trabaja de noche puede consultar al que trabaja en jornada diurna.

Todos los medios de comunicación empleados por la Universidad están limitados a la disponibilidad de una persona, la cual solo brinda el servicio de atención en su horario laboral

de 8:00 am a 5:00 pm de lunes a viernes, como se mencionó anteriormente, este hecho hace que el tiempo de respuesta de consultas realizadas se vea afectado. En un mundo donde el ritmo de las personas es acelerado, el tiempo de respuesta se torna importante para toda organización debido que directamente está relacionado con la satisfacción del cliente, es este caso los estudiantes. Y no solo es importante por el hecho de brindar un buen servicio sino también que es determinante para el estudiantado, dado lo anterior al no recibir una pronta respuesta los estudiantes se ven en la obligación de utilizar un medio más rápido como las llamadas telefónicas.

Dado los puntos mencionados anteriormente, la problemática presente es la forma en cómo se da el proceso de comunicación entre estudiantes y la Escuela de Informática, así como el acceso a la información, el cual está limitado en tiempo, accesibilidad y disponibilidad, donde no se encuentra respuesta hasta de preguntas básicas y generales. Como parte de una solución para estos problemas, este proyecto propone el desarrollo de una herramienta tecnológica llamada “Chatbot”. Un chatbot es un programa computacional que interactúa con personas utilizando lenguaje natural, el cual brinda respuestas a consultas realizadas por el usuario de forma automática. En este caso un Chatbot ayudaría a facilitar la comunicación e intercambio de información de forma rápida y sencilla, sin limitaciones de tiempo y disponibilidad entre la Escuela de Informática de la Universidad Nacional y sus estudiantes.

1.2. Justificación

Este proyecto brindará una herramienta (chatbot) que facilite el proceso de comunicación en la Escuela de Informática de la Universidad Nacional, donde será un complemento al método tradicional de atención de los estudiantes, el cual permitirá acceder de manera sencilla a cualquier información que necesite el estudiantado sobre los diferentes procesos universitarios. La herramienta ayudará a la Escuela a brindar una atención personalizada con la calidez y eficiencia que los alumnos esperan.

Los chatbots son herramientas que ofrecen muchas ventajas a las organizaciones como lo es el ahorro de tiempo y optimización de procesos, respondiendo de forma inmediata las consultas realizadas, ahorrando de esta manera tiempo al usuario y dinero a la empresa. También un chatbot permite recopilar información importante a las compañías que los utilizan. Dado lo anterior, esta herramienta podrá ayudar tanto a funcionarios como a estudiantes con la respuesta de consultas, permitiendo la utilización del tiempo en otras responsabilidades y a su vez evitar consultas repetitivas por parte de los estudiantes, facilitando la labor para ambas partes. Además, dada sus características se podrá evacuar dudas de la población estudiantil en general, ya sean postulantes, estudiantes de primer ingreso, regulares, egresados y funcionarios, todo en una sola herramienta.

Una característica importante de esta tecnología es su fácil adaptabilidad al negocio, un chatbot permite abarcar distintos temas de diferentes áreas a la vez, generando así un recurso integral donde el estudiantado podría obtener la información que necesite por un solo medio; además respondería preguntas de diferente índole como becas, registro, admisión, matrícula, graduación. También los estudiantes ya no tendrían la limitante de apegarse a un horario de oficina o esperar días u horas por respuestas, debido que el chatbot trabaja 24 horas al día los 7 días a la semana y los 365 días del año brindando respuestas inmediatas y automáticas.

Además, el chatbot al ser una herramienta de comunicación por medio de mensajería, forma parte de los medios de comunicación que las personas prefieren utilizar en la actualidad para interactuar con las organizaciones según la investigación realizada en el 2016 en Estados Unidos, 1 de cada 4 clientes prefieren usar tecnología de chat para interactuar con las empresas,

también 4 de cada 10 personas encuestadas están dispuestas a utilizar conversaciones automáticas con un chatbot. Otro dato complementario proporcionado por 24/7 Customer, Inc (2016), es que las personas jóvenes con edades entre 18 a 29 años prefieren utilizar aplicaciones de mensajería para realizar sus consultas. Asimismo, Oracle en una infografía publicada sobre chatbots menciona que el 65% de los clientes prefieren usar aplicaciones de mensajería (Oracle Corporation, 2016).

La tecnología de chatbot tiene un gran potencial, sin embargo, en este momento la Universidad no ha tomado provecho de este recurso que permitiría a los estudiantes recibir una experiencia personalizada por medio de un modelo adaptable a sus vidas a través de las tecnologías y con la disponibilidad 24/7 de comunicación con la Escuela.

Finalmente, el desarrollo del chatbot es una oportunidad de crecimiento para las estudiantes que realizarán este proyecto, ya que este recurso tecnológico es innovador a nivel educativo en el país, debido a que es un tema que no se ha desarrollado, lo cual representa un desafío y genera una curva alta de aprendizaje para el equipo. También dado que se utilizarán herramientas de software libre para la construcción del bot se generará el desarrollo de conocimiento enriquecedor tanto para las estudiantes como para la Escuela, donde se pueden presentar hallazgos importantes y lecciones aprendidas que pueden ser tomados como referencia para futuros proyectos de estudiantes y de la institución.

1.3. Descripción del proyecto

La Universidad Nacional de Costa Rica (UNA) es una universidad pública costarricense con gran prestigio a nivel nacional e internacional además de ser reconocida como una universidad dedicada a la investigación. La Universidad Nacional de Costa Rica fue creada en 1973 y la sede principal encuentra ubicada en la ciudad de Heredia.

Actualmente cuenta con 5 sedes universitarias: Central (Heredia), Interuniversitaria (Alajuela), Región Chorotega (Liberia y Nicoya), Región Brunca (Pérez Zeledón y Coto) y Región Huetar Norte y Caribe (Sarapiquí). Cada una de estas sedes brindan la oportunidad de estudio a la población de cada uno de los lugares.

En la actualidad la Universidad Nacional cuenta con cuatro canales de comunicación los cuales son página web, correos electrónicos, redes sociales y recepción de llamada telefónicas, sin embargo, el proceso de comunicación entre la Universidad y el estudiantado tiene varias limitaciones.

Dado lo anterior se propone el desarrollo de un chatbot, el cual es una herramienta tecnológica que la Escuela de Informática podrá utilizar como complemento del proceso de comunicación entre la universidad y el estudiantado. La herramienta permitirá a los estudiantes realizar consultas de temas frecuentes de procesos universitarios como lo es matrícula y admisión.

El chatbot permitirá superar algunas de las limitantes existentes en este momento como lo es tiempo, horarios, disponibilidad, acceso rápido a la información de forma casi inmediata utilizando un medio amigable para el usuario. Por medio de ella se atenderán dudas de los estudiantes de temas frecuentes de una forma personalizada y con la calidez que los estudiantes esperan y merecen.

Para el desarrollo del Chatbot se utilizará herramientas de software libre, donde posibles candidatas son MongoDB y Python. Esta última cuenta con varias bibliotecas que pueden ser utilizadas para el desarrollo de algoritmos que ayuden a la construcción del chatbot las cuales se pueden utilizar en machine learning. El machine learning es una disciplina de la inteligencia

Artificial que crea sistemas que aprenden automáticamente, es decir, identifica patrones complejos en millones de datos, todo lo anterior realizado por un algoritmo.

Además, se evaluará su uso en la Escuela de Informática de la Universidad de Nacional que permitirá apoyar tanto a estudiantes como al personal administrativo de la institución.

1.4. Objetivos

1.4.1. Objetivo general:

Facilitar y mejorar la comunicación entre los estudiantes y el personal administrativo de la Escuela de Informática de la Universidad Nacional por medio de un chatbot utilizando herramientas de software libre.

1.4.2. Objetivos específicos:

1. Investigar las herramientas tecnológicas de software libre utilizadas para el diseño de un chatbot.
2. Desarrollar un chatbot basado en herramientas de software libre en la Escuela de Informática de la Universidad Nacional
3. Evaluar el uso del chatbot como medio para mejorar el proceso de comunicación entre los estudiantes y funcionarios de la Escuela de Informática de la Universidad Nacional mediante un proyecto piloto.

1.4.3. Relación objetivos, productos e indicadores de logro por producto

Tabla 1

OBJETIVOS Y PRODUCTO

Tabla de objetivo	Producto
Investigar las herramientas tecnológicas de software libre utilizadas para el diseño de un chatbot en una organización.	Una investigación de las herramientas de software libre utilizadas en chatbot
Desarrollar un chatbot basado en herramientas de software libre en la Escuela de Informática de la Universidad Nacional	Un chatbot
Evaluar el uso del chatbot mediante un proyecto piloto con los estudiantes y funcionarios de la Escuela de Informática de la Universidad Nacional	Chatbot evaluado en la Escuela de Informática

Fuente. Elaboración propia

1.5. Áreas de conocimiento

Dentro de las áreas de conocimiento del campo de las Tecnologías de información y comunicación se pueden mencionar las siguientes:

- a. **Bases de datos:** Son parte fundamental del desarrollo del proyecto, debido a que se requiere almacenar información para ser consultada y se requiere que esté actualizada. Se necesita guardar las preguntas y respuestas acerca de matrícula, constancias, requisitos de trámites, graduación, becas, entre otros.
- b. **Sistemas de información:** Es importante conocer los diferentes sistemas de información y tecnología que utilizan actualmente las empresas. Con el fin de

desarrollar un chatbot acorde a las necesidades de la Escuela de Informática, el cual sea de ayuda para los estudiantes.

- c. **Administración de proyectos:** La administración de proyectos es lo principal para que cualquier proyecto tenga éxito en un tiempo determinado. Para este proyecto se implementará SCRUM como metodología ágil con el objetivo de entregar iteraciones con un producto funcional.
- d. **Lenguajes de programación:** Conocer los diferentes lenguajes de programación para desarrollar un chatbot es muy importante para elegir el óptimo y conocer cuál se adapta más a las necesidades del proyecto.
- e. **Trámites institucionales:** Se debe conocer a detalle sobre todos los trámites institucionales de la Universidad Nacional, con el fin de brindar la información de forma fácil, rápida y actualizada a los estudiantes sobre matrícula, constancias, requisitos de trámites, graduación, becas, entre otros.

1.6. Alcance del proyecto

El proyecto presenta una propuesta de chatbot con el objetivo de ser una herramienta de apoyo que ayude en el proceso de comunicación de la Escuela de informática y sus estudiantes. Dado lo anterior se deberán obtener y desarrollar conocimientos en los diferentes procesos de la escuela de temas de interés para la comunidad estudiantil como lo es matrícula, graduación, becas entre otros.

La propuesta del chatbot será elaborada para la Escuela de Informática, la cual es una de las más grandes de la Universidad Nacional. Esta escuela actualmente cuenta con 1505 estudiantes en el Campus Benjamín Núñez, esta cantidad de estudiantes está distribuida en los distintos grados académicos que ofrece la institución desde diplomado hasta maestría. Por lo cual el desarrollo del chatbot tendrá como función principal facilitar la comunicación entre la Escuela de Informática y el estudiantado al brindar respuesta a dudas sobre los procesos anteriormente mencionados.

El chatbot estará compuesto de dos áreas, la parte administrativa de la herramienta y la parte dirigida al público. El módulo administrativo contará con una sección para la asignación de roles, el objetivo de este módulo es brindar seguridad y actualización de los datos. El administrador del sistema tendrá todos los permisos y será el encargado de mantener la información actualizada que brindará el chatbot, además podrá agregar, editar y eliminar la información.

La información contenida en la base de datos del chatbot deberá ser agregada por el administrador del sistema utilizando el módulo de mantenimiento, sin embargo, el chatbot será entregado con datos de pruebas de varios de los temas que son consultados por la población estudiantil. La actualización de la información dependerá del administrador de la herramienta.

Asimismo, contará con la sección principal que es donde el estudiante o usuario podrá realizar una pregunta y se le facilitará la respuesta. Si por algún motivo el chatbot no cuenta con la respuesta, el usuario tendrá la posibilidad de enviar la pregunta como sugerencia al administrador del sistema.

Finalizado el desarrollo del chatbot se realizará una prueba piloto de la herramienta con estudiantes de grados académicos de diplomado, bachillerato y licenciatura de la Escuela de Informática de la Universidad Nacional para evaluar el funcionamiento de este.

1.7. Beneficios esperados

El proyecto brindará a los estudiantes una herramienta que les facilite información sobre temas de matrícula, constancias, requisitos de trámites, graduación, becas, entre otros, de forma actualizada. Por lo que permitirá a los estudiantes ahorrar tiempo y obtener una guía óptima en la forma de cómo deben realizar los trámites y procesos.

Otro beneficio que permitir el chatbot será de apoyo a la labor del personal administrativo, ya que al existir esta herramienta los estudiantes no requerirán llamar a la universidad, sino que podrán resolver sus consultas por medio del bot y de esa forma el personal administrativo pueda realizar otras labores.

La Universidad Nacional se ve beneficiada ya que esta herramienta estará a disposición de la Escuela de Informática en caso de que decidan implementarla en la página web, para uso de toda la población estudiantil.

El desarrollo del sistema será por medio de herramientas de software libre, lo cual permitirá a las estudiantes desarrolladoras del proyecto tener una curva de aprendizaje en temas variados relacionados con el desarrollo del bot. Por otra parte, se obtendrán hallazgos y conclusiones importantes sobre el trabajo con chatbots, las cuales pueden ser tomadas como referencia para futuros proyectos de estudiantes y de la institución.

1.8. Resultados o productos esperados

- a. Investigación de las herramientas de software libre.
- b. Desarrollo de un chatbot.

1.9. Riesgos potenciales del proyecto

Tabla 2

LOS RIESGOS, PROBABILIDAD Y ACCIONES

N°	Tipo Riesgo	Descripción	Probabilidad	Impacto	Magnitud	Acciones
1	Tiempo	Mala estimación del tiempo en el desarrollo de las tareas	3	2	6	Dedicar tiempo extra a la realización de tareas, realizar una reorganización de las prioridades de las tareas
2	Alcance	Mala estimación del alcance del proyecto	3	2	6	Realizar un análisis más detallado del proyecto, los puntos principales para realizar una priorización de tareas y delimitar bien el alcance.
3	Tecnología	Problemas de conocimiento en el desarrollo del proyecto dadas las herramientas que se van a utilizar	2	2	4	Realizar investigaciones exhaustivas para completar las tareas de acuerdo con las herramientas.
4	Riesgos externos	Desastres naturales, fallos de las herramientas tecnológicas	1	3	3	Utilizar herramientas de almacenamiento en la nube y backups para evitar la pérdida del trabajo realizado
5	Participación de los estudiantes del proyecto	Enfermedad o situación externa que afecte el trabajo de los estudiantes del proyecto	2	3	6	Finalizar todas las tareas que se puedan antes de la fecha de entrega. Trabajar horas extra por parte de los estudiantes.

Fuente. Elaboración propia

2. Marco teórico

2.1. Marco referencial

En este apartado se detalla los temas relacionados con la Escuela de Informática, institución para la cual se desarrollará el proyecto. Por lo tanto, su lectura nos permitirá comprender mejor acerca de la historia, misión y visión de la institución y el objetivo de este proyecto.

La Escuela de Informática de la Universidad Nacional tiene como objetivo principal la formación de profesionales conscientes de las necesidades del país. La innovación, la excelencia, el humanismo, trabajo en equipo, con espíritu positivo y de mucha colaboración son algunas de las características de los estudiantes de la Escuela de Informática.

2.1.1. Misión de la Escuela de Informática

Formar profesionales altamente competitivos en tecnologías de la información y comunicación, e informática educativa, y contribuir a la capacitación de la población en el uso de la tecnología de la información para propiciar el desarrollo humano de la sociedad costarricense. (USTDS-Escuela de Informática-UNA, 2018)

2.1.2. Visión de la Escuela de Informática

Liderar como Escuela, el desarrollo de las tecnologías de la información y de la comunicación en Costa Rica mediante su creación, promoción y uso en las distintas áreas del conocimiento, como un motor para el desarrollo humano que contribuya a lograr una sociedad justa, equitativa y solidaria. (USTDS-Escuela de Informática-UNA, 2018)

2.1.3. Objetivos estratégicos

- La formación de profesionales responsables y comprometidos con el desarrollo tecnológico y humano.
- El fortalecimiento la cultura tecnológica de la sociedad en sus distintas actividades.
- La investigación para aumentar la aplicación de la tecnología de la información y comunicación en la sociedad.

- Actualización continua de su personal para mejorar la calidad de la docencia y dar respuesta a las necesidades de la sociedad costarricense. (USTDS-Escuela de Informática-UNA, 2018)

2.1.4. Fortalezas

La Escuela de Informática cuenta con algunas fortalezas que se describirán a continuación:

- El 100% de los académicos en propiedad cuentan con una experiencia profesional-laboral mayor o igual a 10 años
- Las áreas de formación y especialización de los docentes guardan estrecha relación con el plan de estudios.
- La Escuela de Informática cuenta con un plan de estudios que contempla las últimas recomendaciones a nivel nacional de instituciones como Pro-Software y CONARE. Esto la ubica en el mercado costarricense con un plan de estudios actualizado con las nuevas tendencias en el área de la Tecnología de la Información.
- Los laboratorios y el equipo tecnológico son adecuados y suficiente para ejecutar eficiente y eficazmente el plan de estudios.
- La Escuela de Informática ha propiciado en los últimos cuatro años la investigación asignando carga académica a los profesores que formulan y presentan proyectos.
- Existe evaluación del proceso enseñanza-aprendizaje de manera periódica y sistemática. (USTDS-Escuela de Informática-UNA, 2018)

2.2. Marco conceptual

Uno de los avances tecnológicos en el área de la informática son los Chatbots, Abu Shawar y Atwell en su artículo, “Chatbots: ¿Son realmente útiles?”, definen el concepto como “los chatbot son programas computacionales que interactúan con usuarios utilizando lenguaje natural” (Shawar, 2007, p.29). Esta herramienta es un programa utilizado para participar en conversaciones con personas, el cual está programado para anticipar respuestas a preguntas realizadas por un usuario, utilizando técnicas de Inteligencia Artificial a la hora de responder. Para efectos de este proyecto se define chatbot como un programa el cual permite las respuestas de consultas realizadas por un usuario utilizando lenguaje natural. Entiéndase lenguaje natural como el lenguaje utilizado por los humanos para comunicarse entre sí. (RAE, 2017)

Los Chatbots tienen sus inicios en los años 60s, uno de los primeros es ELIZA, diseñado en 1966 por Joseph Weizenbaum, así como ALICE que fue Premio Loebner del año 2000 (Polatidis, 2014). En la actualidad los chatbots han sido utilizados principalmente en el área de mercadeo, ventas y de atención al cliente en grandes empresas, esto dado por su adaptación a cualquier área de negocio.

En el campo empresarial, las grandes empresas como American Express, CNN, NBA, Disney, UBER, entre otras, utilizan esta tecnología en áreas como mercadeo, experiencia de usuario y ventas (Nava, 2016). Por otra parte, según una investigación realizada por Oracle en el 2016, el 36% de las compañías encuestadas en su estudio utilizaban en ese momento chatbots para interactuar con los clientes y el 44% tenían planeado utilizarla para el 2020 (Oracle Corporation, 2016).

En este punto Costa Rica no se ha quedado atrás y ha iniciado a incursionar en esta área. En los últimos años algunas empresas han iniciado a crear bots, estos son de relevancia en el país por lo cual los periódicos La República y La Nación han publicado noticias con respecto al tema de chatbots como Edna quien es un bot asistente o Layla que es una bot utilizada en temas bancarios (Vargas, 2018).

En el área educativa algunas compañías y universidades han visto las ventajas de utilizar esta herramienta, por ende, la han implementado como, por ejemplo, la empresa AIVO y Unit4 ofrecen plataformas de bots para la educación superior (Unit4, 2016; AIVO, 2018). También en universidades como Leeds Beckett de Inglaterra y la Universidad Tecnológica de Georgia han desarrollaron sus propios chatbots como solución a diferentes necesidades (Cardwee, 2016). A nivel nacional también se puede mencionar EconBot, un bot creado por el Departamento de Economía Aplicada de la UNED en su plan piloto del curso “Economía: Fundamentos Microeconómicos” con el objetivo de brindar un apoyo extra a los estudiantes en el curso (Tamayo, 2018).

En la actualidad existen varias herramientas para la elaboración de chatbots y algunas de estas son de software libre. En esta parte, software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. Como parte de las primeras herramientas de software libre para la creación de chatbot, se pueden encontrar lenguajes como AIML por sus siglas en inglés (Artificial Intelligence Markup Language) el cual fue utilizado para la creación de ALICE y Chatscript el cual es un lenguaje diseñado para manipular lenguaje natural, también utilizado para la creación de chatbots. (Arsovski et al. 2017).

Por otra parte, hoy en día existen otras herramientas para crear un chatbot, en internet se pueden encontrar más de 30 plataformas las cuales tienen la capacidad de integrarse con herramientas de mensajería como telegram o facebook messenger, por ejemplo: Chatfuel, Wit.ai y Botsify, Flow xo y Beep Bood. Algunas de estas plataformas son libres, otras tienen opciones libres con métodos de pago y otras hay que pagar por ellas.

Asimismo, se encuentran otras herramientas de programación que se pueden utilizar para el desarrollo de chatbots como por ejemplo Python, MongoDB, Postgres, PHP, Node, entre otros. Algunas de estas cuentan con características especiales para la creación de bots como, por ejemplo, Python tiene una biblioteca llamada ChatterBot la cual utiliza algoritmos de machine learning que ayuda a producir las respuestas automáticas en un chatbot.

Como se mencionó anteriormente también se pueden encontrar diferentes herramientas de software privativo para el desarrollo del chatbots como por ejemplo Microsoft bot

framework, el cual es un framework desarrollado por Microsoft y permite crear bots de una forma muy completa para Facebook Messenger y Skype. Además, está IBM que ofrece su producto llamado Watson Conversation la cual es un API cognitiva para construir bots. Esta funciona con Slack y telegram, además puede ser integrado a Facebook Messenger con Stamply.

El uso de los chatbots trae consigo algunas ventajas como lo son la disponibilidad, dado que es una herramienta que está a disposición 24 horas al día los 365 días de año, permitiendo así que los operadores reales se pueden dedicar a otras tareas. Otra característica es su precisión, ya que sus respuestas son concretas y van al grano en cuanto obtiene una pregunta o respuesta por parte del usuario.

Como parte de las ventajas se puede mencionar el ahorro de tiempo debido a la eficiencia y facilidad de interacción a la hora de brindar la información. Los chatbots son capaces de hacerle frente a un gran volumen de solicitudes o peticiones, brindando respuesta a todas las preguntas recibidas. Por último, un chatbot cuenta con la opción de análisis, este puede recoger diferentes tipos de datos, como, por ejemplo, las palabras más usadas por los usuarios, para que posteriormente se puedan ejecutar otras acciones de marketing o inteligencia artificial; ya que de esta forma acumulan el conocimiento que reciben volviéndose más inteligentes y flexibles después de cada conversación con el cliente. (Benítez, s.f)

De igual forma el uso de los chatbots tiene algunas desventajas, de las cuales se pueden mencionar la falta de información, debido que algunas veces no cuentan con la respuesta a todas las preguntas realizadas por el usuario. También se puede dar la pérdida de información por la cual deben tener copias de seguridad y no exponerse a perder todos los datos que ya se tienen almacenados.

Otra desventaja son los fallos técnicos, un chatbot puede fallar al recibir muchas peticiones o quedarse bloqueado perdiéndose la información, así mismo existe la posibilidad de que no sea aceptado por los usuarios, a muchas personas no les puede gustar interactuar con los chatbots, prefiriendo así interactuar con una persona real. (Benítez, s.f)

3. Marco metodológico

Este apartado corresponde al resultado de la investigación sobre la metodología de desarrollo que se implementará en el proyecto, esto con el fin de determinar el alcance e impacto de SCRUM la solución propuesta.

SCRUM es una metodología ágil y flexible que puede ser aplicada de manera regular, mediante un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Esta metodología de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades.

Algunos beneficios que conlleva esta metodología (SCRUM, 2020):

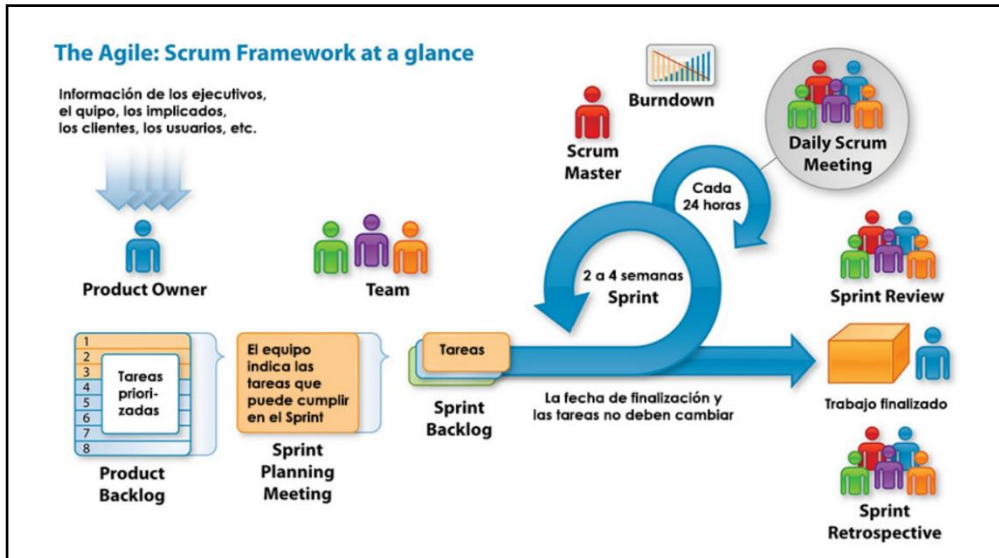
- a. Mayor productividad.
- b. Cumplimiento de expectativas.
- c. Flexibilidad a cambios.
- d. Productos de mejor calidad.
- e. Reducción del tiempo de comercialización.
- f. Mejor satisfacción de los interesados.
- g. Mejor dinámica de equipo.
- h. Reducción de riesgos.

A continuación, se hará una descripción más detallada del proceso a utilizar.

El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada Sprint, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. (SCRUM, 2020)

Figura 1

PROCESO DE SCRUM



Fuente: Scrum (2015)

3.1. Product backlog o pila de producto

Conjunto de requisitos denominados historias descritos en un lenguaje no técnico y priorizados por valor de negocio considerando su beneficio y costo. (SCRUM, 2020)

3.2. Sprint planning

Reunión durante la cual el dueño del producto presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint. (SCRUM, 2020)

3.3. Sprint

Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo. (SCRUM, 2020)

3.4. Sprint backlog

Lista de las tareas necesarias para llevar a cabo las historias del sprint. (SCRUM, 2020)

3.5. Daily meeting

Es una breve reunión de comunicación (no más de 15 minutos) en la que cada miembro del equipo cubre de manera rápida y transparente el progreso, cada miembro comenta que hizo el día anterior, que hará hoy y si hay inconvenientes. (SCRUM, 2020)

3.6. Sprint review

Es momento de demostrar al equipo presente el trabajo completado durante el sprint. El propietario del producto verifica el trabajo con criterios de aceptación predefinidos y acepta o rechaza el trabajo. (SCRUM, 2020)

3.7. Sprint retrospective

Es la reunión final del equipo en el Sprint para determinar qué salió bien, qué no salió bien y cómo el equipo puede mejorar en el próximo Sprint. Es una oportunidad importante para que el equipo se concentre en su desempeño general e identifique estrategias para la mejora continua de sus procesos. (SCRUM, 2020)

3.8. Roles

El equipo Scrum está formado por los siguientes roles (SCRUM, 2020):

- a. El dueño del producto: es la parte interesada clave del proyecto, generalmente un cliente interno o externo, o un portavoz del cliente.
- b. El Scrum Máster: es el líder de servicio al propietario del producto, equipo de desarrollo y organización. Garantiza que el equipo se adhiera a la teoría, las prácticas y las reglas de Scrum. Scrum Master protege al equipo haciendo todo lo posible para ayudar al

equipo a desempeñarse al más alto nivel. Esto puede incluir eliminar impedimentos, facilitar reuniones y ayudar al propietario del producto a preparar el trabajo atrasado.

- c. El equipo de desarrollo: es un grupo autoorganizado, multifuncional, con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.

4. Metodología de desarrollo

Como parte inicial, se investiga la existencia de chatbots en el área educativa y el uso de este en el país. A nivel internacional existen universidades que utilizan esta tecnología, por ejemplo, la universidad de Georgia, universidad Empresarial Siglo 21 de Argentina y la Universidad Leed Beckett de Inglaterra, sin embargo, a nivel nacional no se encontró alguna institución que hiciera uso de la herramienta.

Luego de analizar el funcionamiento de los chatbots, se procede con la recolección de la información que iba a utilizar el sistema, para lo anterior se realizan varias actividades. Como por ejemplo el trabajo de campo realizado en una etapa anterior del proyecto, en donde logramos conocer algunas de las preguntas que realizaban los estudiantes en fechas de matrícula. Además, se contactó a la Asociación de estudiantes de escuela y al Departamento de Registro de la universidad, los cuales nos facilitaron las preguntas frecuentes que suelen hacerles los estudiantes. Asimismo, para complementar esta información se realiza una entrevista al personal encargado de la recepción de la escuela. Además de preguntar a los estudiantes sobre las consultas frecuentes que realizaban a la escuela

Como segunda parte se realiza una investigación la cuál actúa como guía para el diseño y creación del chatbot; es por ello que se investigará sobre las diferentes plataformas que existen en el mercado y herramientas actuales para la elaboración de bots. Se hará una selección de al menos 10 de las herramientas de software libre y se analizarán las características con el fin de elegir la adecuada para desarrollar el proyecto. Además, dichas herramientas están dentro de la categoría de software libre

Seguidamente, se inicia la etapa de planeación y desarrollo del chatbot. En la fase de planeación se realizará el diseño, arquitectura, se creará los requerimientos y criterio de aceptación de cada requerimiento. Todos los elementos mencionados anteriormente serán fundamentales para el proceso de desarrollo de la herramienta.

La metodología de trabajo que se utilizará para el desarrollo del proyecto será SCRUM, la cual se encuentra en la categoría de metodologías ágiles. Dado lo anterior, se espera realizar el desarrollado del proyecto con una serie de sprint, donde al finalizar cada sprint se tendrá un

pequeño producto funcional, el cual formará parte del proyecto principal. Para cada entregable que se genere, al terminar cada sprint, se realizarán varias pruebas que incrementará la calidad del producto, como, por ejemplo: pruebas funcionales y pruebas de usabilidad. Estas pruebas ayudarán a que el trabajo se realice de la mejor manera.

Dentro de la fase de desarrollo parte de las actividades a realizar son: creación de la base de datos que será utilizada, la construcción del canal de comunicación entre la base de datos y el chatbot, el desarrollo del chatbot y la documentación correspondiente. Finalizado el desarrollo del sistema se procederá a realizar un conjunto de pruebas tanto funcionales como de usabilidad para corroborar el debido funcionamiento del sistema de forma integral. Para esto, se desarrollará un plan de pruebas que se adjuntará en los documentos y se realizarán por medio de un plan piloto en el cual se pondrá a prueba con estudiantes, los cuales evaluarán el sistema. Los estudiantes serán tanto de bachillerato como de licenciatura.

4.1. Resumen ejecutivo de costos

Para el desarrollo del proyecto se utilizan herramientas de software libre, donde se aprovecha la característica de ser gratuita y así disminuir los costos y dependencias del proyecto. Al ser el chatbot una propuesta para la Escuela de Informática, se crea de forma independiente al sistema de la Universidad, contemplando el caso de si la escuela quisiera implementar, esta utilizaría los recursos actuales y no necesitaría de costos adicionales.

En este proyecto de graduación se genera inversión de tiempo para concluirlo, realizamos un estimado de los costos a nivel del mercado laboral de nuestro país el precio con respecto a la documentación se basa en el salario de un bachillerato según el Ministerio de Trabajo y el de desarrollo del aplicativo según el salario de un ingeniero en sistemas con experiencia con las que cuenta actualmente las estudiantes aproximadamente de 3 años, a continuación, un estimado del tiempo y costo invertido en cada etapa del proyecto:

Tabla 3*DETALLE DEL TIEMPO INVERTIDO*

Actividad	Horas	Precio por hora	Precio total
Anteproyecto	60	€3500	€210,000
Objetivo específico 1, “Investigar las herramientas de software libre”	20	€8997	€179,940
Desarrollo de la comunicación base de datos con el servidor	33	€8997	€296,901
Desarrollo de vistas del sistema	40	€8997	€359,880
Creación de preguntas	20	€8997	€179,940
Preguntas agregadas luego de la implementación del formulario	6	€8997	€53,982
Desarrollo de la aplicación	264	€8997	€2,374,944
Análisis de resultados	20	€3500	€70,000
Preguntas luego del plan piloto	14	€8997	€125,958
Módulo de ayuda y categoría “Maestría”	6	€8997	€53,976
Elaboración de manuales	20	€3500	€70,000
Documentación del proyecto	40	€3500	€140,000
Total	543		€4,115,521

Fuente. Elaboración propia

4.2. Análisis de herramientas

A continuación, se presenta el análisis de software libre realizado, el cual incluye lenguajes de programación, motores de base de datos, frameworks y otros aplicativos, lo anterior con el objetivo de identificar las herramientas a utilizar en el desarrollo del chatbot:

4.2.1. Back-End

4.2.1.1. Python

Python es un lenguaje de programación de propósito general, el cual es orientado a objetos y puede ser utilizado para el desarrollo web. Este lenguaje cuenta con varias características de las cuales se puede mencionar. (Python, 2020)

a. Gratuito:

Python no dispone licencia para programar.

b. Propósito general:

Puede ser utilizado para crear todo tipo de programas.

c. Multiplataforma:

Hay versiones disponibles de Python para diferentes sistemas informáticos, es compatible con cualquier sistema siempre y cuando exista un intérprete programado para el.

d. Interactivo:

Python dispone de un intérprete por la línea de comandos el cual cada sentencia ejecutada produce un resultado visible, ayudando a entender mejor el código.

e. Orientado a Objetos:

Soporta la programación orientada a objetos.

f. Funciones y librerías:

Dispone de muchas funciones incorporadas para el manejo de datos, así como librerías que pueden ser incorporadas al programa permitiendo funciones específicas.

4.2.1.2. Node

Node.JS es un entorno de JavaScript el cual se utiliza del lado del servidor utilizando un modelo asíncronico y dirigido a eventos. Node cuenta con las siguientes características (Moreno, 2017):

- a. Es un entorno de ejecución multiplataforma de código abierto para desarrollar aplicaciones web.
- b. Puede utilizar en varios sistemas como Windows, Linux y Mac OS X.
- c. Soporta protocolos TCP, DNS y HTTP.
- d. Gestiona todas las operaciones utilizando programación asíncronica.
- e. Facilita la creación de aplicaciones web escalables de una forma sencilla y con gran escalabilidad.

4.2.1.3. Java

Java es un lenguaje de programación multiplataforma con propósito general, el cual puede ser obtenido de forma gratuita por los usuarios. Java cuenta con las siguientes características (Joyanes y Fernández, 2004):

- a. Lenguaje sencillo:

Java reduce errores comunes de programación realizados con lenguajes como C y C++ al no contener características de estos como por ejemplo aritmética de punteros y la no existencia de referencias.
- b. Orientado a objetos:

Java permite la programación orientada a objetos.

c. Asegurado:

Al estar confeccionado dentro de un ambiente delimitado de una máquina virtual, acredita su seguridad de factores de peligro como virus.

d. Robusto:

Java un servicio de memoria sólida, así como un recolector de basura.

e. Arquitectura neutral:

Este lenguaje de programación es imparcial en arquitectura porque no existe característica dependiente de la ejecución.

f. Portátil:

Java es imparcial en arquitectura dado que no existe característica dependiente de la ejecución. No pretende ninguna ejecución.

g. Multi-Hilo:

Dada esta característica Java permite muchas actividades simultáneas en un programa.

h. Dinámico:

Java utiliza una lengua dinámica, la cual aprueba la carga dinámica de géneros. Igualmente acepta funciones de sus dialectos nativos, es decir, de C y C ++.

4.2.1.4. PHP

PHP es un lenguaje de programación de código abierto utilizado en el desarrollo web de páginas dinámicas. Algunas características son las siguientes (Arias, 2013):

a. Lenguaje Libre.

b. Multiplataforma: Puede ser utilizado en Linux, Windows, Unix, Mac OS, entre otros.

c. Extensa documentación disponible.

- d. PHP tiene varios módulos o extensiones que permiten ampliar la funcionalidad del lenguaje.
- e. Es un lenguaje que se utiliza para programar la lógica interna de muchas aplicaciones web, es lo que se le conoce como backend de la aplicación web.

4.2.2. Front-End

4.2.2.1. JavaScript

JavaScript (JS) es un lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, más conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador, tales como node.js, Apache CouchDB y Adobe Acrobat. Es un lenguaje script multi-paradigma, basado en prototipos, dinámico, soporta estilos de programación funcional, orientada a objetos e imperativa. (Eguíluz, 2008)

Las características de JavaScript que hace de este lenguaje uno de los más populares en la actualidad es:

- a. Es liviano.
- b. Multiplataforma, ya que se puede utilizar en Windows, Linux o Mac o en el navegador de tu preferencia.
- c. Es imperativo y estructurado, mediante un conjunto de instrucciones indica al computador qué tarea debe realizar.
- d. Prototipado, debido a que usa prototipos en vez de clases para el uso de herencia.
- e. Orientado a objetos y eventos.
- f. Es interpretado, no se compila para poder ejecutarse.

4.2.2.2. Angular

Angular es un framework de código abierto desarrollado por Google para facilitar la creación y programación de aplicaciones web de una sola página, las webs SPA (Single Page Application).

Angular separa completamente el frontend y el backend en la aplicación, evita escribir código repetitivo y mantiene todo más ordenado gracias a su patrón MVC (Modelo-Vista-Controlador) asegurando los desarrollos con rapidez, a la vez que posibilita modificaciones y actualizaciones. (Angular, 2020)

Incluye una gran cantidad de características esenciales:

- a. **Nativo:** Cree aplicaciones móviles nativas con estrategias de Cordova, Ionic o NativeScript.
- b. **Universal:** Sirva la primera vista de su aplicación en Node.js, .NET, PHP.
- c. **Plantillas:** Cree rápidamente vistas de IU con una sintaxis de plantilla simple y potente.
- d. **Accesibilidad:** Cree aplicaciones accesibles con componentes habilitados para ARIA, guías para desarrolladores e infraestructura de prueba integrada.
- e. **Código de GENERACION:** Angular convierte sus plantillas en código altamente optimizado para las máquinas virtuales JavaScript de hoy, brindándole todos los beneficios del código escrito a mano con la productividad de un marco.
- f. **Aplicaciones web progresivas:** Utilice las capacidades modernas de la plataforma web para ofrecer experiencias similares a las aplicaciones. Instalación de alto rendimiento, fuera de línea y cero pasos.

4.2.2.3. HTML

HTML es el lenguaje con el que se define el contenido de las páginas web. Básicamente se trata de un conjunto de etiquetas que sirven para definir el texto y otros elementos que compondrán una página web, como imágenes, listas, vídeos, etc. (Van, 2012)

Algunas características son las siguientes:

- a. **Técnica empleada:** Es la técnica empleada que logra que el usuario se mueva por la web, solo debes hacer clic en un texto específico denominado como hipervínculos que te conducirá a la siguiente página.
- b. **Función:** HTML se fundamenta por una serie de códigos breves escritos en un registro de texto por el escritor del sitio web. Estos códigos son las etiquetas que habitualmente observamos al añadir una nueva entrada en nuestra página web.

- c. Función de las etiquetas en el HTML: Las etiquetas cumplen una excelente función en una publicación basada en HTML, a través de ellas hacen separar el texto habitual del código HTML.

4.2.3. Gestores de Bases de Datos

4.2.3.1. MySQL

MySQL es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basada en código abierto. MySQL es un gestor de bases de datos que permite la interacción con los lenguajes de programación más utilizados como PHP o Java. La mayor parte de su código se encuentra escrito en lenguaje C/C++.

Características de MySQL

MySQL presenta algunas ventajas que lo hacen muy interesante para los desarrolladores. La más evidente es que trabaja con bases de datos relacionales, es decir, utiliza tablas múltiples que se interconectan entre sí para almacenar la información y organizarla correctamente. (Milani, 2006)

Algunas características a continuación:

- a. Arquitectura Cliente y Servidor: MySQL basa su funcionamiento en un modelo cliente y servidor. Es decir, clientes y servidores se comunican entre sí de manera diferenciada para un mejor rendimiento.
- b. Compatibilidad con SQL: SQL es un lenguaje generalizado dentro de la industria. Al ser un estándar MySQL ofrece plena compatibilidad.
- c. Vistas: Desde la versión 5.0 de MySQL se ofrece compatibilidad para poder configurar vistas personalizadas del mismo modo que podemos hacerlo en otras bases de datos SQL.
- d. Procedimientos almacenados: MySQL posee la característica de no procesar las tablas directamente, sino que a través de procedimientos almacenados es posible incrementar la eficacia de nuestra implementación.

- e. Desencadenantes: MySQL permite además poder automatizar ciertas tareas dentro de nuestra base de datos. En el momento que se produce un evento otro es lanzado para actualizar registros u optimizar su funcionalidad.
- f. Transacciones: Una transacción representa la actuación de diversas operaciones en la base de datos como un dispositivo. El sistema de base de registros avala que todos los procedimientos se establezcan correctamente o ninguna de ellas.

4.2.3.2. MongoDB

MongoDB es una base de datos orientada a documentos. Esto quiere decir que, en lugar de guardar los datos en registros, guarda los datos en documentos. Estos documentos son almacenados en BSON, que es una representación binaria de JSON. Una de las diferencias más importantes con respecto a las bases de datos relacionales, es que no es necesario seguir un esquema (MongoDB, Inc., 2020).

Estas son sus principales características (Castillo et al, 2017):

- a. De propósito general, casi tan rápida como las bases de datos NoSQL de tipo clave valor, y con casi todas las funcionalidades de las bases de datos relacionales
- b. Alta disponibilidad
- c. Escalabilidad: desde un servidor aislado a arquitecturas distribuidas de grandes clusters (Permite particionar, de manera transparente para el usuario, nuestra base de datos en tantas como shards tengamos. Esto aumenta el rendimiento del procesado de los datos al ser cada una de ellas más pequeña que la original)
- d. Aggregation Framework– Procesamiento batch de datos para cálculos agrupados usando operaciones nativas de MongoDB
- e. Auto balanceado de carga a través de los distintos shards. El balanceador decide cuándo migrar los datos, y a qué Shard, para que estén uniformemente distribuidos entre todos los servidores del cluster. Cada shard aloja los datos correspondientes a un rango de la clave escogida para particionar nuestra colección.
- f. Replicación nativa: sincronización de datos entre servidores
- g. Seguridad: autenticación, autorización, etc

- h. Gestión avanzada de usuarios
- i. Automatic failover: elección automática de un nuevo primario cuando este se ha caído
- j. Es capaz de actualizarse sin dejar de dar servicio
- k. No tiene los cuellos de botella que se producen en las bases de datos relacionales (RDBMS) al procesar grandes cantidades de información
- l. Utiliza objetos JSON para guardar y transmitir la información. JSON es el estándar web hoy en día, por lo que supone una gran ventaja que web y base de datos hablen el mismo lenguaje.
- m. Permite realizar consultas y cálculos espaciales, tanto en 2D como en 3D.
- n. Permite utilizar Map-Reduce para el procesado de la información a través de funciones JavaScript que se ejecutan en los servidores
- o. Puede almacenar y ejecutar funciones JavaScript en el servidor
- p. Tiene una potente herramienta web que nos permite monitorizar todo lo que pasa en nuestras bases de datos y en nuestras máquinas, MongoDB Management Service (MMS).

4.2.3.3. Postgres

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto que utiliza y amplía el lenguaje SQL combinado con muchas características que almacenan y escalan de forma segura las cargas de trabajo de datos más complicadas. Los orígenes de PostgreSQL se remontan a 1986 como parte del proyecto POSTGRES en la Universidad de California en Berkeley y tiene más de 30 años de desarrollo activo en la plataforma central. (PostgreSQL, 2020)

Algunas características son las siguientes:

- a. Tipos de datos
 - Primitivas: entero, numérico, cadena, booleano
 - Estructurado: fecha / hora, matriz, rango, UUID
 - Documento: JSON / JSONB, XML, valor-clave (Hstore)
 - Geometría: Punto, Línea, Círculo, Polígono
 - Personalizaciones: Compuestas, Tipos Personalizados.

b. Integridad de los datos

- ÚNICO, NO NULO
- Llaves primarias
- Llaves extranjeras
- Restricciones de exclusión
- Cerraduras explícitas, cerraduras consultivas

c. Concurrencia, rendimiento

- Indexación: B-tree, Multicolumn, Expresiones, Parcial
- Indexación avanzada: GiST, SP-Gist, KNN Gist, GIN, BRIN, índices de cobertura, filtros Bloom
- Planificador / optimizador de consultas sofisticado, análisis de solo índice, estadísticas de varias columnas
- Transacciones, transacciones anidadas (a través de puntos guardados)
- Control de concurrencia multi-versión (MVCC)
- Paralelización de consultas de lectura y creación de índices de árbol B
- Particionamiento de tablas
- Todos los niveles de aislamiento de transacciones definidos en el estándar
- SQL, incluido Serializable.
- Recopilación de expresiones Just-in-time (JIT)

d. Confiabilidad, Recuperación de Desastres

- Registro de escritura anticipada (WAL)
- Replicación: asíncrona, síncrona, lógica.
- Recuperación de punto en el tiempo (PITR), recursos activos
- Espacios de tabla

e. Seguridad

- Autenticación: GSSAPI, SSPI, LDAP, SCRAM-SHA-256, certificado y más
- Sistema robusto de control de acceso
- Seguridad de columnas y filas

f. Extensibilidad

- Funciones y procedimientos almacenados.
- Lenguajes de procedimiento: PL / PGSQL, Perl, Python (y muchos más)
- Contenedores de datos externos: conéctese a otras bases de datos o flujos con una interfaz SQL estándar
- Muchas extensiones que proporcionan funcionalidad adicional, incluyendo PostGIS

g. Internacionalización, búsqueda de texto

- Soporte para conjuntos de caracteres internacionales, por ejemplo, a través de colaciones de UCI
- Búsqueda de texto completo

4.3. Modelado

La primera etapa del proyecto fue la recolección de los datos y poder definir los temas de interés de los estudiantes que se incluirían en la herramienta, se realizaron varias actividades. Primeramente, se solicitó información al Departamento de Registro de la Universidad, el cual nos proporcionó las preguntas frecuentes de su sitio web, así como de sus redes sociales (Facebook).

Como parte de este esfuerzo se realizaron entrevistas informales tanto a los estudiantes como al personal de recepción de la Escuela de Informática. También se realizó un trabajo de campo el segundo día de matrícula del segundo semestre del 2018, con el objetivo de conocer las preguntas que realizan los estudiantes en fechas importantes.

La segunda etapa fue la investigación de las herramientas de software con el objetivo de conocer y utilizar las indicadas en el proyecto. Asimismo, se analizó algunas herramientas de chatbot que existen en la actualidad, estudiando su comportamiento y características.

Por último, para el diseño del sistema se crearon requerimientos y se definió la arquitectura con la cual se iba a desarrollar el chatbot y su sistema de mantenimiento. El desarrollo del chatbot se realizó bajo la metodología de SCRUM.

4.4. Desarrollo

Para el desarrollo del chatbot se tomó en cuenta diferentes aspectos, uno de ellos fue la utilización de herramientas de software libre, como parte de la investigación se encontraron plataformas gratuitas para la construcción de chatbots como Facebook, Botsify o Telegram. Sin embargo, estos no cuentan con las características de software libre. Por esta razón se decidió desarrollar la herramienta desde cero, lo cual brinda ventajas a nivel económico y de disponibilidad, ya que la herramienta no estará ligada a plataformas de terceros o licenciamientos futuros. También podrá ser modificada en cualquier momento por la escuela y podrá ser tomada de referencia por otras personas sin violar ninguna regla.

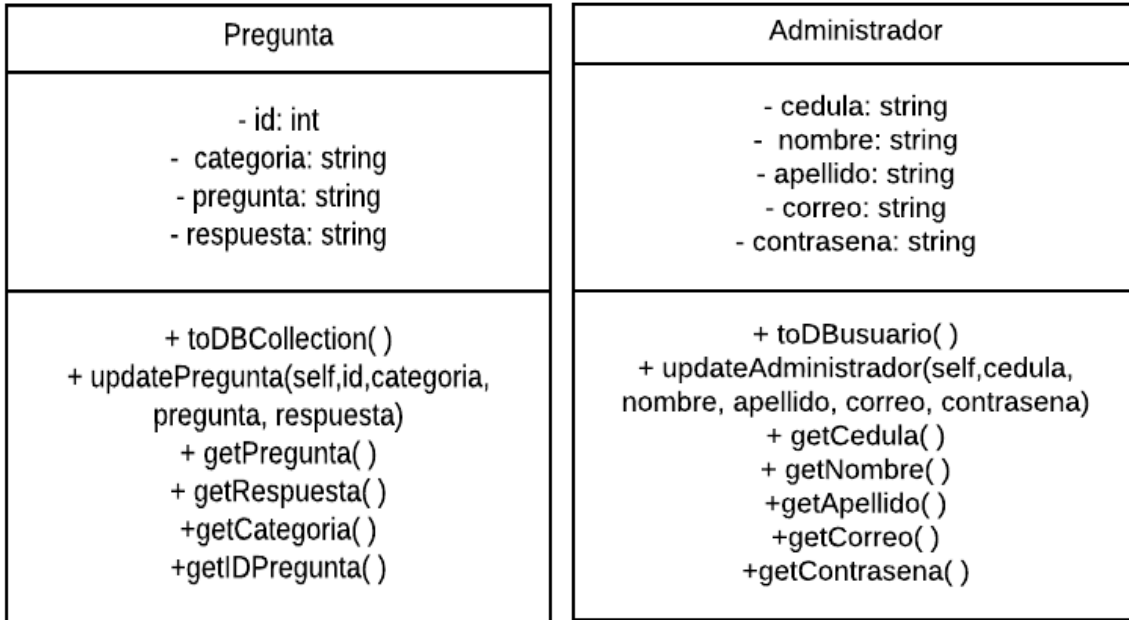
Como resultado de esta fase, se obtiene la propuesta de sistema de chatbot para la Escuela de Informática y el mantenimiento de este. El chatbot permite y facilita el proceso de comunicación entre la Escuela de Informática de la Universidad Nacional y los estudiantes. El chatbot será un complemento al método tradicional de atención de los estudiantes, el cual permitirá acceder de manera sencilla a cualquier información que necesiten sobre los diferentes procesos universitarios. La herramienta ayudará a la Escuela a brindar una atención personalizada a los estudiantes.

4.4.1. Diseño y arquitectura

El sistema de chatbot a nivel de diseño cuenta con dos clases, la clase Pregunta, la cual utiliza el chatbot para trabajar y la clase Administrador, la cual es utilizada por el sistema en el módulo de mantenimiento. A continuación, se presenta el diagrama de cada una de las clases.

Figura 2

DIAGRAMA DE CLASES

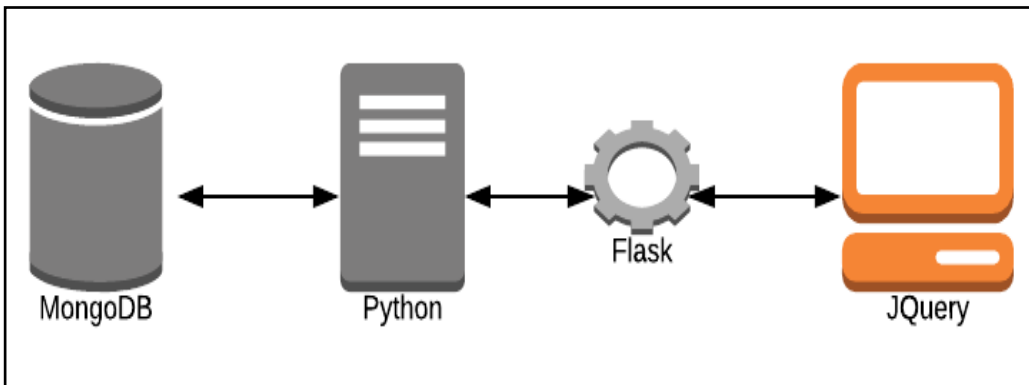


Fuente: Elaboración propia

A nivel de arquitectura se utilizaron las siguientes herramientas: MongoDB, Python, Flask y JQuery. Como se puede ver en la siguiente imagen:

Figura 3

DIAGRAMA DE ARQUITECTURA



Fuente. Elaboración propia

A continuación, se detalla y justifica las herramientas principales utilizadas en las diferentes partes del sistema:

4.4.1.1. MongoDB

Se utiliza MongoDB como base de datos, aprovechando las siguientes características:

a. Código abierto: MongoDB cumple con una de las características fundamentales del proyecto, la cual es código abierto, permitiendo una gran ventaja al no tener que pagar licencias para poder usarse. Además, MongoDB ofrece su servicio de Database as a Services, MongoDB_Atlas el cual permitió construir la Base de Datos en la nube para el desarrollo del plan piloto del chatbot, utilizando su versión gratuita.

b. Flexibilidad: Las clases utilizadas en el proyecto no tienen una relación directa, por lo tanto, se aprovecha la característica NoSQL de mongo para crear los esquemas, tomando ventaja de este modelo en el momento de almacenar cada uno de los atributos de las clases, en especial el atributo respuesta, ya que no se conoce de antemano que tan grande puede ser la respuesta de una pregunta. También se evita de esta forma futuros errores en la base de datos por sobrepasar la capacidad de almacenamiento de alguna variable. Para la versión 4 utilizada en el proyecto el límite de capacidad es de un documento BSON (Formato de serialización Binaria) de 16 megabytes

c. Escalabilidad: Si pensamos a futuro Mongo permite escalar, permitiendo la creación de nodos y compartir entre estos la carga del sistema.

4.4.1.2. Python

Se trabaja con Python versión 3.7 en la parte del servidor, dado que es un lenguaje de programación simplificado y de fácil uso, en donde sobresalieron aspectos como:

a. Flexibilidad: Python es un lenguaje de programación muy flexible el cual permite trabajar temas como machine learning, chatbots, Inteligencia Artificial, Análisis de Datos entre otros.

b. Bibliotecas y marcos de trabajo: Python posee muchas bibliotecas de trabajo lo cual permite facilidad en el momento de crear las funcionalidades que el sistema necesita. En el proyecto se utilizan varias de ellas por ejemplo: fuzzywuzzy, unicodedata,

email.mime.multipart, json y pymongo. También permite trabajar con frameworks de gran utilidad, en este caso se utiliza Flask el cual se detallará más adelante.

c. Programación orientada a objetos: Se aprovecha esta característica para construir las clases que se utilizaron en el proyecto, clase Pregunta y la clase Administrador.

4.4.1.3. Flask

Flask es un micro marco de trabajo, es utilizado en el proyecto para la comunicación entre el front end y back end, dado que está escrito en Python, se utiliza de la siguiente forma en la construcción del chatbot:

a. Esquema de rutas: Se aprovecha el buen manejo de rutas que posee flask para realizar la comunicación entre ambos ambientes (front end y back end) en el momento de mostrar cada pantalla de los diferentes módulos, tanto del chatbot como el módulo de mantenimiento.

b. Seguridad: Flask se usa dentro del trabajo para la configuración de la seguridad del sistema de mantenimiento del chatbot, utilizando sesiones para evitar que personas no autorizadas ingresen al sistema.

c. Depurador: Como tercer punto se utiliza el depurador de Flask en el momento de hacer pruebas el cual fue de mucha utilidad y sencillo de utilizar. Este depurador permitió encontrar rápidamente los errores y poder solucionarlos.

4.4.1.4. Fuzzywuzzy

Para la creación del algoritmo del chatbot el cual cumpliría con la función de responder las consultas realizadas por los estudiantes, se iba a utilizar la biblioteca ChatterBot de Python. Sin embargo, luego de implementarla, el algoritmo no brindaba las respuestas correctamente y el margen de error era muy grande, por otra parte, no se logró encontrar mucha información o ejemplos sobre la librería. Dado lo anterior se procede a analizar otras opciones. Una de estas opciones fue Fuzzywuzzy, esta herramienta tiene una función que calcula la relación de similitud de distancia estándar de Levenshtein entre dos secuencias de caracteres, la cual permite al chatbot comparar la similitud entre la pregunta que el usuario ingresó con las de la

base de datos, luego de encontrar la pregunta con mayor similitud, el sistema retorna la respuesta.

4.4.1.5. JQuery

Se utiliza esta herramienta como parte del proceso de comunicación entre front end y back end del sistema. JQuery permite hacer la solicitud de la información y poder desplegarla en cada una de las ventanas de forma fácil y ágil.

4.4.1.6. Bootstrap

Se utiliza Bootstrap en el proyecto tomando en cuenta sus características para el diseño de front end, en donde se aprovecha:

a. Facilidad: Es fácil de usar y permite crear sitios web con un estilo amigable para el usuario final.

b. Diseño Responsivo: Se aprovecha esta característica de Bootstrap considerando que el chatbot puede ser usado en varios dispositivos electrónicos, como computadoras, celulares o tablets, gracias al diseño responsivo el chatbot se va a adaptar a cualquiera de los objetos mencionados, así mismo no habrá problemas entre sistemas operativos ya que la dirección del chatbot podrá ser ingresada desde cualquier explorador.

c. Adaptabilidad: Se aprovecha esta característica ya que el sistema utiliza HTML, JavaScript y CSS, en el momento de creación de las diferentes vistas del sistema.

4.4.2. Desarrollo del proyecto

Como parte del primer objetivo específico se hizo una investigación para conocer las posibles herramientas de software libre que podían ser usadas en el proyecto. Al finalizar la investigación se determina utilizar las herramientas de Python, MongoDB, HTML, JavaScript y Bootstrap.

El desarrollo del chatbot se dividió en cuatro partes, la primera fue la creación de la base de datos (BD) en MongoDB, para esto se utiliza la biblioteca de Python pymongo para poder hacer a la conexión entre ambas partes. La BD cuenta con dos esquemas, uno para las preguntas y el otro para los administradores del sistema. Así mismo se crearon dos clases en el servidor, Pregunta y Administrador con sus respectivas variables.

Como segunda parte del desarrollo, se intentó crear la lógica del chatbot utilizando la biblioteca de Python llamada Chatterbot, sin embargo, el sistema mostraba muy poco asertividad en las respuestas, además no se encontró suficiente información sobre la librería. Dado lo anterior se decidió buscar otras opciones.

Luego de investigar un poco más sobre los chatbots se optó por utilizar la biblioteca FuzzyWuzzy de Python y crear un algoritmo basado en coincidencia entre caracteres. FuzzyWuzzy utiliza la distancia de Levenshtein para medir que tan separadas están dos secuencias de palabras. A continuación, se presenta la definición formal de distancia entre dos cadenas de caracteres a y b:

Figura 4

DEFINICIÓN ENTRE LA DISTANCIA DE DOS CADENAS DE CARACTERES

$$\text{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \text{lev}_{a,b}(i-1,j) + 1 \\ \text{lev}_{a,b}(i,j-1) + 1 \\ \text{lev}_{a,b}(i-1,j-1) + 1_{(a \neq b)} \end{cases} & \text{otherwise.} \end{cases}$$

Fuente: (Carrera, 2019)

También es posible calcular el radio de similitud con base a la distancia de Levenshtein como se muestra a continuación:

Figura 5

FÓRMULA PARA CALCULAR EL RADIO DE SIMILITUD

$$\frac{(|a| + |b|) - lev_{a,b}(i, j)}{|a| + |b|}$$

Fuente (Carrera, 2019)

En el proyecto se utilizó el método `fuzz.token_set_ratio()` el cual tokeniza la cadena, la ordena alfabéticamente y luego las une, para aplicar el método `fuzz.ratio()` y obtener el porcentaje de similitud. También se utilizó el método `process.extractOne()` el cual compara una cadena de caracteres con cada uno de los objetos de un vector de caracteres y retorna la cadena con mayor similitud. A continuación, se presenta el código del método utilizado en el proyecto:

Figura 6

MÉTODO DE FUZZ.TOKEN_SET_RATIO

```
def consulta(pregunta):
    pMejores=[]
    pOpciones=[]
    for pre in my_data:
        Token_Set_Ratio = fuzz.token_set_ratio(eliminaTildes(pregunta), eliminaTildes(pre.getPregunta()))
        if Token_Set_Ratio > 84:
            print(Token_Set_Ratio, pre.getPregunta())
            pMejores.append(pre)
            pOpciones.append(eliminaTildes(pre.getPregunta()))

    if len(pMejores) != 0:
        highest = process.extractOne(eliminaTildes(pregunta), pOpciones)
        print(highest[0], highest[1])

        for p in pMejores:
            if highest[0] == eliminaTildes(p.getPregunta()):
                return p.getRespuesta()

    return None
```

Fuente Aguilar, A y Hernández, D. (2020)

El método consulta recibe como parámetro la pregunta que realiza el usuario, esta pregunta se compara con la información del chatbot utilizando `fuzz.token_set_ratio()`, si el resultado del ratio al hacer la comparación es mayor a 85% la pregunta es guardada en una lista temporal. Luego de obtener las preguntas con mayor coincidencia de similitud, la lista temporal es comparada una segunda vez con la consulta del usuario utilizando el método `process.extractOne()` para brindar la respuesta que se mostrará en pantalla. Si ninguna pregunta almacenada en el chatbot cumplió con el porcentaje de similitud, el método retornará `None` y el sistema le indicara al usuario que no se encontró respuesta. También se utiliza la biblioteca `unicodedata` para eliminar las tildes entre consultas y aumentar de esta manera el porcentaje de similitud en las sentencias.

Parte de las funcionalidades de la herramienta es poder recuperar la contraseña, esto para los administradores de la aplicación. También, el chatbot permite al estudiante poder enviar un mensaje a los administradores con la pregunta que el sistema no encontró. Para poder realizar estas acciones se utiliza la biblioteca de Python `MIMEMultipart`.

En la tercera parte del desarrollo se crean las vistas, se utilizan las herramientas de HTML, CSS y Bootstrap, estas fueron utilizadas para diseñar cada una de las pantallas que se utilizaron en el chatbot y el mantenimiento de este. Por último, en la cuarta etapa se trabaja con la biblioteca de Python llamada Flask la cual permite llamar cada uno de los archivos `.html` según la pantalla, esto se logra por medio de uno de sus métodos `render_template()`. Las acciones de las pantallas se realizan utilizando JQuery y JavaScript.

Finalmente, el proceso de pruebas no se pudo realizar de forma presencial debido a la crisis del COVID-19 que está enfrentando nuestro país. Dada esta limitante se procede a publicar el chatbot, la herramienta utilizada es Heroku, el cual es un servidor gratuito. También se utiliza MongoDB Atlas como gestor de base de datos para el sistema almacenado en Heroku. Mongo Atlas es una versión de Mongo en la nube y gratuita.

4.4.3. Implementación

El chatbot se implementó en un servidor para facilitar el proceso de pruebas de aceptación por parte de los estudiantes. El mantenimiento del chatbot es solo para uso de usuarios autorizados por la Escuela de Informática por lo que se requiere que el usuario esté registrado en el sistema.

El sistema de chatbot se divide en dos partes:

- a. Chatbot
- b. Módulo de mantenimiento

El área del Chatbot cuenta con tres vistas, la primera es la pantalla principal donde el estudiante hará las preguntas, la segunda ventana permite al estudiante sugerir nuevas temas o preguntas para agregar al chatbot, la tercera vista hace referencia al módulo de ayuda la cual sirve de guía al usuario.

El Módulo de mantenimiento del chatbot cuenta con cinco partes, la primera es el inicio de sesión como se visualiza, la segunda es el área de Agregar Preguntas, la tercera Agregar Usuarios, la cuarta es el Mantenimientos de Preguntas y la quinta es el Mantenimiento de Usuarios. Cada uno de los módulos se detallará en el siguiente apartado.

4.4.3.1. Chatbot

A continuación, se explicará cada una de las partes del área del Chatbot.

4.4.3.1.1. Pantalla principal del chatbot

En la pantalla principal, el chatbot cuenta con un campo de texto donde se podrá ingresar la pregunta, al dar clic al botón de enviar que está al lado, la pregunta se desplegará en la pantalla y el chatbot le facilitará la respuesta. Como se visualiza en la imagen en la figura 8. Además, cuenta con dos botones en la parte superior, uno para ingresar nuevas preguntas y otro que servirá de ayuda a los usuarios de cómo utilizar la herramienta.

Figura 7

PANTALLA PRINCIPAL DEL CHATBOT



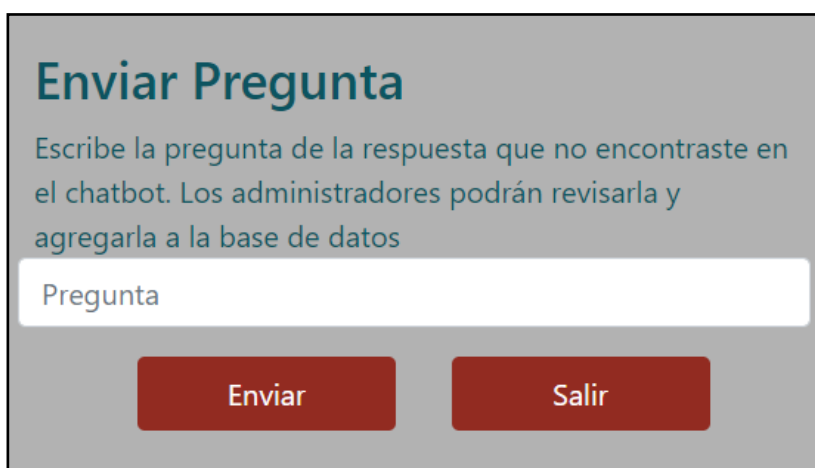
Fuente Aguilar, A y Hernández, D. (2020)

4.4.3.1.2. Agregar nuevas preguntas

En este apartado se le permite al usuario sugerir nuevas preguntas o temas que le gustaría que el administrador incluyera en el chatbot, en caso en que el chatbot no tenga respuesta. Esta sección cuenta con un campo de texto que permite ingresar la pregunta y dos botones “Enviar”, “Salir”. Al hacer clic al botón “Enviar” se le enviará la pregunta al administrador y con el botón “Salir” podrá regresar a la pantalla principal del chatbot.

Figura 8

AGREGAR NUEVAS PREGUNTAS



Enviar Pregunta

Escribe la pregunta de la respuesta que no encontraste en el chatbot. Los administradores podrán revisarla y agregarla a la base de datos

Pregunta

Enviar Salir

Fuente Aguilar, A y Hernández, D. (2020)

4.4.3.1.3. Ayuda

Este apartado servirá de guía al usuario para que pueda usar el Chatbot. Esta sección cuenta con una explicación del uso de la herramienta y una imagen del chat en funcionamiento. Contiene con un botón “Salir” que al seleccionarlo podrá regresar a la pantalla principal del chatbot.

Figura 9

APARTADO DE AYUDA

Ayuda

1. Escriba su pregunta en el campo de texto, al dar click en el botón enviar, el Chatbot mostrará la respuesta a su consulta, si el sistema no cuenta con la respuesta, mostrará el mensaje "No se encontró respuesta para tu pregunta"
2. En la esquina superior derecha se encuentra un botón, el cual le permitirá ingresar la pregunta de la respuesta que no encontró, esto con el objetivo de que el administrador la revise y pueda ser ingresada a la base de datos



The screenshot displays a chatbot interface titled "Chatbot UNA". In the top right corner, there is a help icon (question mark) and a red button with a white plus sign. The chat history shows the following messages:

- Bot: "Hola! Escribe tu consulta"
- User: "Cuál es el teléfono de la Escuela de Informática"
- Bot: "El número de teléfono es 2562 6363"
- User: "Cuáles son las fechas de matrícula"
- Bot: "Ingresar a www.registro.una.ac.cr, Sección "Calendario"."

At the bottom of the chat window is a text input field with the placeholder "Escriba el mensaje..." and a red send button with a white paper plane icon. Below the chat window is a large red button labeled "Salir".

Fuente Aguilar, A y Hernández, D. (2020)

4.4.3.2. Mantenimiento del chatbot

A continuación, se presentan los diferentes módulos del área de Mantenimiento del Chatbot:

4.4.3.2.1. Ingreso al sistema

Esta pantalla contiene dos campos de texto, el campo “Usuario” donde se ingresa la cédula del administrador y campo “Contraseña” en el cual se ingresa la clave. El botón “Entrar” le permitirá ingresar al sistema. Además, cuenta con un enlace de “Olvidaste la contraseña”, al dar clic en esta opción, el sistema muestra otra pantalla, en caso de que el usuario haya olvidado su contraseña y requiera recuperarla.

Figura 10

INGRESO AL MANTENIMIENTO DE CHATBOT



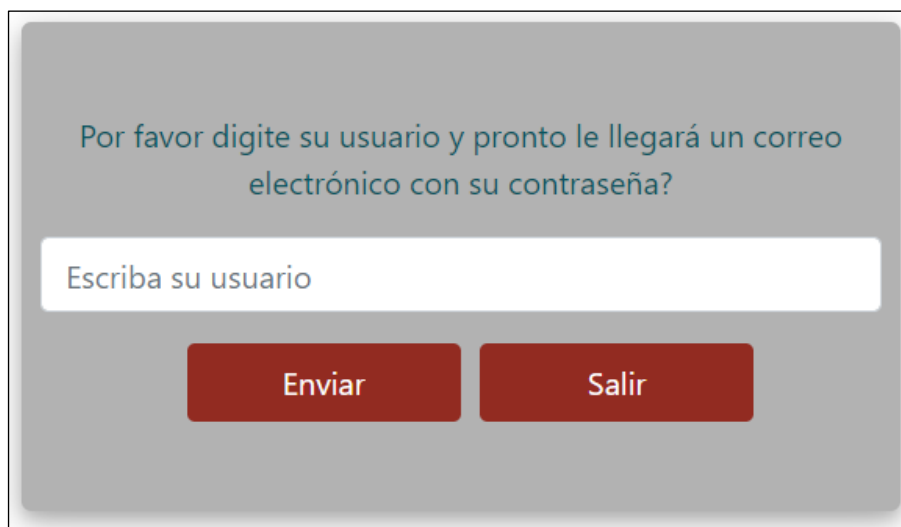
Fuente Aguilar, A y Hernández, D. (2020)

4.4.3.2.2. Olvidaste contraseña

Si el usuario olvidó su contraseña, puede ingresar a esta sección, el sistema le ayudará a recuperarla, ya que por medio de éste se enviará un correo al administrador con la contraseña. Este apartado contiene un campo de texto que permite ingresar el usuario, dos botones “Enviar” y “Salir”. Al darle clic al botón “Enviar” se le enviará un correo al administrador con la contraseña y con el botón “Salir” podrá regresar a la pantalla de inicio de sesión.

Figura 11

OLVIDASTE CONTRASEÑA



Por favor digite su usuario y pronto le llegará un correo electrónico con su contraseña?

Escriba su usuario

Enviar Salir

Fuente Aguilar, A y Hernández, D. (2020)

4.4.3.2.3. Agregar preguntas

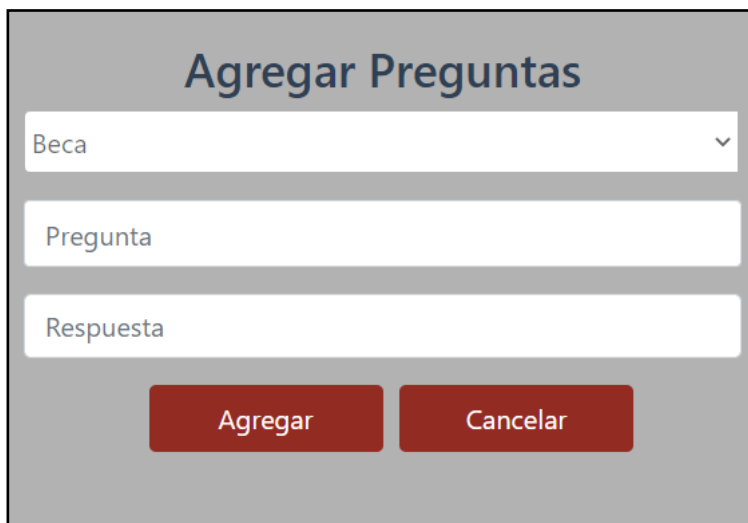
La sección de “Agregar Preguntas” le permite al administrador ingresar nuevas preguntas al chatbot. El administrador debe agregar los datos que se solicitan. Entre los cuales se encuentran:

- Categoría: Lista desplegable con las categorías de las preguntas (Beca, Cursos, General, Graduación, Maestría, Matrícula).
- Pregunta: Campo de texto para ingresar la pregunta.
- Respuesta: Campo de texto, donde el usuario ingresa la respuesta a la pregunta ingresada en el campo anterior.

Además de dos botones “Agregar” y “Cancelar”, al dar clic al “Agregar” se agrega la pregunta a la base de datos. Al darle clic al botón “Cancelar” se anula el ingreso y lo direcciona a la página de “Mantenimiento de Preguntas”.

Figura 12

AGREGAR PREGUNTAS



El formulario 'Agregar Preguntas' tiene un fondo gris. En la parte superior, el título 'Agregar Preguntas' está en un recuadro gris con el texto en azul. Debajo del título, hay tres campos de entrada de texto blancos con bordes grises: el primero contiene 'Beca' y tiene un ícono de flecha hacia abajo a la derecha; el segundo contiene 'Pregunta'; el tercero contiene 'Respuesta'. En la parte inferior del formulario, hay dos botones rectangulares de color rojo oscuro con el texto 'Agregar' y 'Cancelar' en blanco.

Fuente Aguilar, A y Hernández, D. (2020)

4.4.3.2.4. Agregar usuarios

El módulo de Agregar Usuario es para que el administrador del chatbot ingrese posibles nuevos usuarios. El administrador debe agregar los datos que se solicitan. Entre los cuales se encuentran:

- a. Cédula: Campo de texto para ingresar la cédula del administrador del chatbot
- b. Nombre: Campo de texto alfabético para ingresar el nombre del administrador del chatbot
- c. Apellido: Campo de texto alfabético para ingresar los apellidos del administrador del chatbot
- d. Correo: Campo de texto para ingresar el correo del administrador.
- e. Contraseña: Campo de texto para ingresar la contraseña con la que podrá ingresar al sistema.
- f. Confirmación Contraseña: Campo de texto para confirmar la contraseña con la que podrá ingresar al sistema.

Además de dos botones “Agregar” y “Cancelar”, al dar clic al botón “Agregar” se agrega el registro a la base de datos. Al darle clic al botón “Cancelar” se anula el ingreso y lo direcciona a la página de “Mantenimiento de Preguntas”.

Figura 13

AGREGAR USUARIO



El formulario "Agregar Usuario" está diseñado con un fondo gris claro. En la parte superior, el título "Agregar Usuario" está en un color azul oscuro. A continuación, se encuentran seis campos de entrada de texto, cada uno con un ícono rojo a la izquierda: un ícono de persona para "Cédula", "Nombre" y "Apellido"; un ícono de correo para "Correo"; un ícono de llave para "Contraseña"; y un ícono de llave para "Confirmación Contraseña". Los campos de texto son blancos con una línea inferior gris. En la parte inferior del formulario, hay dos botones rectangulares de color rojo oscuro con texto blanco: "Agregar" a la izquierda y "Cancelar" a la derecha.

Fuente Aguilar, A y Hernández, D. (2020)

4.4.3.2.5. Mantenimiento de preguntas

Este apartado es para que el administrador visualice la lista de preguntas con las que cuenta el chatbot, esta vista cuenta con los siguientes campos: categoría, pregunta, respuesta y dos botones en la cual se podrá actualizar o eliminar preguntas. Además, contiene una lista desplegable con las categorías para realizar un filtro de búsqueda de las preguntas y un campo de texto en el que se podrá ingresar palabras claves para filtrar la información al darle clic al botón buscar.

Se visualiza un botón Modificar por cada pregunta al darle clic lo direcciona a la pantalla de “Actualizar Pregunta”. De igual forma el botón eliminar por cada pregunta al seleccionarlo visualiza el mensaje de si desea eliminar el registro, este mensaje cuenta con dos botones “Aceptar” elimina el registro y muestra la tabla de nuevo, sin dicho registro y “Cancelar” no se elimina el registro.

Figura 14

MANTENIMIENTO DE PREGUNTAS

Mantenimiento de Preguntas				
Beca				
Escriba el criterio de búsqueda...				
Categoría	Pregunta	Respuesta	Modificar	Eliminar
Beca	Cuál es mi categoría de beca socioeconómica	Los resultados de la solicitud de beca se publican en el área de “Noticias de Vida Estudiantil”: https://www.una.ac.cr/index.php/informacion-para/estudiantes		
Beca	Cómo renuevo la beca	Visite http://www.vidaestudiantil.una.ac.cr/Beca/tipos		
Beca	Cada cuánto se debe renovar la beca	https://www.vidaestudiantil.una.ac.cr/Beca/periodos		
Beca	Cuando debo solicitar la beca	https://www.vidaestudiantil.una.ac.cr/Beca/periodos		
Beca	Puedo obtener pedir varias beca	Se pueden combinar Beca por condición socioeconómica y por desempeño		

Fuente. Aguilar, A y Hernández, D. (2020)

4.4.3.2.6. Actualizar preguntas

Este apartado es para que el administrador del chatbot actualice preguntas. El administrador debe actualizar (si se requiere) los siguientes datos, los datos se muestran ya precargados con la información de la tabla:

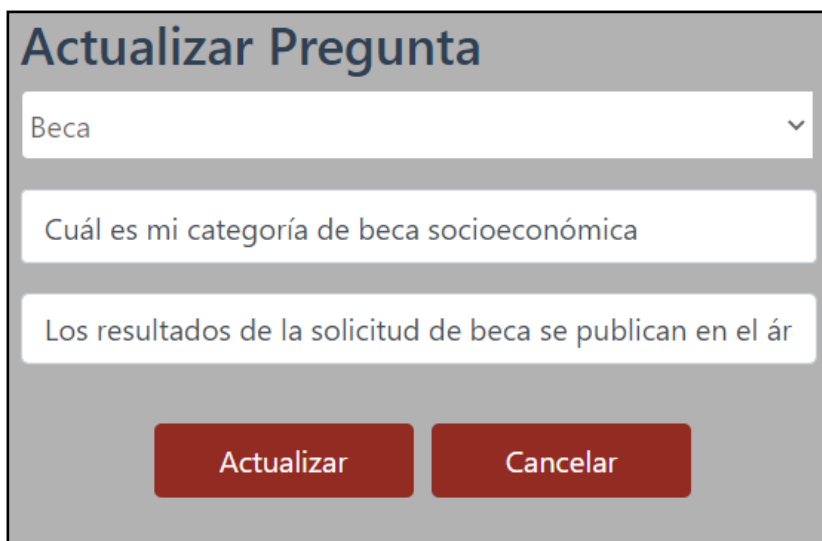
- a. Categoría: Lista desplegable con las categorías de las preguntas (Beca, Cursos, General, Graduación, Maestría, Matrícula).
- b. Pregunta: Campo de texto para ingresar la pregunta.

- c. Respuesta: Campo de texto, donde el usuario ingresa la respuesta a la pregunta ingresada en el campo anterior.

Además de dos botones “Actualizar” y “Cancelar”, al hacer clic al “Actualizar” se actualiza la base de datos y a la lista de preguntas que responde el chatbot. Y al darle clic al botón “Cancelar” se anula la actualización y lo direcciona a la página de “Mantenimiento de Preguntas”.

Figura 15

ACTUALIZAR PREGUNTAS



Fuente. Aguilar, A y Hernández, D. (2020)





4.4.3.2.7. Mantenimiento de usuarios

Este apartado es para que el administrador del chatbot visualice la lista de usuarios con los que cuenta el sistema, esta ventana cuenta con los siguientes campos: cedula, nombre, apellido, correo y dos botones en los que podrá actualizar o eliminar administradores.

Se visualiza un botón Modificar por cada usuario al darle clic lo direcciona a la pantalla de “Actualizar Usuario”. De igual forma el botón eliminar por cada usuario al seleccionarlo visualiza el mensaje si desea el eliminar el registro, esta pantalla cuenta con dos botones “Aceptar” elimina el registro y muestra la tabla de nuevo, sin dicho registro y “Cancelar” no se elimina el registro.

Figura 16

MANTENIMIENTO DE USUARIOS

Mantenimiento de Usuarios					
Cedula	Nombre	Apellido	Correo	Modificar	Eliminar
111111111	admin	administrador	chatbotUNA@gmail.com		
604200756	Yariela	Hernández Artola	dayyari18@hotmail.com		

Fuente Aguilar, A y Hernández, D. (2020)

4.4.3.2.8. Actualizar usuarios

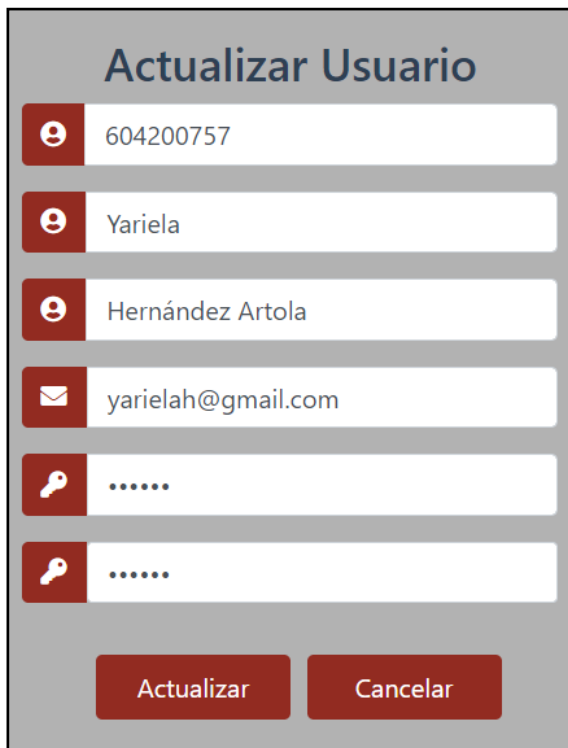
Este apartado es para que el administrador del chatbot actualice usuarios. El sistema muestra ya precargados la información en la pantalla con los campos:

- a. Cédula: Campo de texto para ingresar la cédula del administrador del chatbot
- b. Nombre: Campo de texto alfabético para ingresar el nombre del administrador del chatbot
- c. Apellido: Campo de texto alfabético para ingresar los apellidos del administrador del chatbot
- d. Correo: Campo de texto para ingresar el correo del administrador.
- e. Contraseña: Campo de texto para ingresar la contraseña con la que podrá ingresar al sistema.
- f. Confirmación Contraseña: Campo de texto para confirmar la contraseña con la que podrá ingresar al sistema.

Además de dos botones “Actualizar” y “Cancelar”, al dar clic al “Actualizar” se actualiza a la base de datos y a la lista de usuarios. Al darle clic al botón “Cancelar” se anula la actualización y lo direcciona a la página de “Mantenimiento de Usuarios”.

Figura 17

ACTUALIZAR USUARIO



The image shows a web form titled "Actualizar Usuario" with a light gray background. It contains six input fields, each with a red icon on the left: a person icon for the first three fields, an envelope icon for the fourth, and a key icon for the last two. The first field contains the number "604200757", the second "Yariela", and the third "Hernández Artola". The fourth field contains the email "yarielah@gmail.com". The fifth and sixth fields contain six dots, indicating a password field. At the bottom of the form are two red buttons: "Actualizar" on the left and "Cancelar" on the right.

Fuente Aguilar, A y Hernández, D. (2020)

A continuación, se muestra la vista principal del Módulo de mantenimiento del chatbot con el menú de los módulos mencionados anteriormente.

Figura 18

PANTALLA PRINCIPAL DEL MANTENIMIENTO DE CHATBOT



Fuente Aguilar, A y Hernández, D. (2020)

4.5. Evaluación de la herramienta

Para la evaluación de la herramienta se realizó un plan piloto, en el cual se incluyó a estudiantes de la Escuela de Informática de la Universidad Nacional, tanto de grado de Bachillerato como de Licenciatura. En este plan también se toma en cuenta la participación del personal de secretariado de la escuela.

El plan piloto consta de dos tipos de evaluación, una pertenece a la herramienta del Chatbot y la otra a su sistema de mantenimiento. Para esto se realizó dos procesos, uno para los estudiantes y el otro para el personal de la Escuela de Informática quienes serán los administradores del sistema.

Con respecto a las pruebas del Chatbot, se creó un formulario en el cual los estudiantes debían de realizar algunas preguntas a la herramienta y registrar los resultados en la solicitud. En este formulario se le pidió al estudiante información sobre el nivel académico ya sea Diplomado, Bachillerato o Licenciatura. También se le solicitó información sobre las preguntas

que realizó, las respuestas que le brindó la herramienta, su opinión sobre la herramienta como medio de comunicación entre la escuela y los estudiantes; y por último una sección de comentarios. Este formulario fue aplicado a 70 estudiantes.

Para la segunda parte del sistema, se realizó una demostración al personal de secretariado de la escuela, donde se hace una presentación del Chatbot, el sistema de mantenimiento y sus funcionalidades. Esta sesión fue realizada a tres miembros de la escuela.

4.5.1. Análisis de resultados

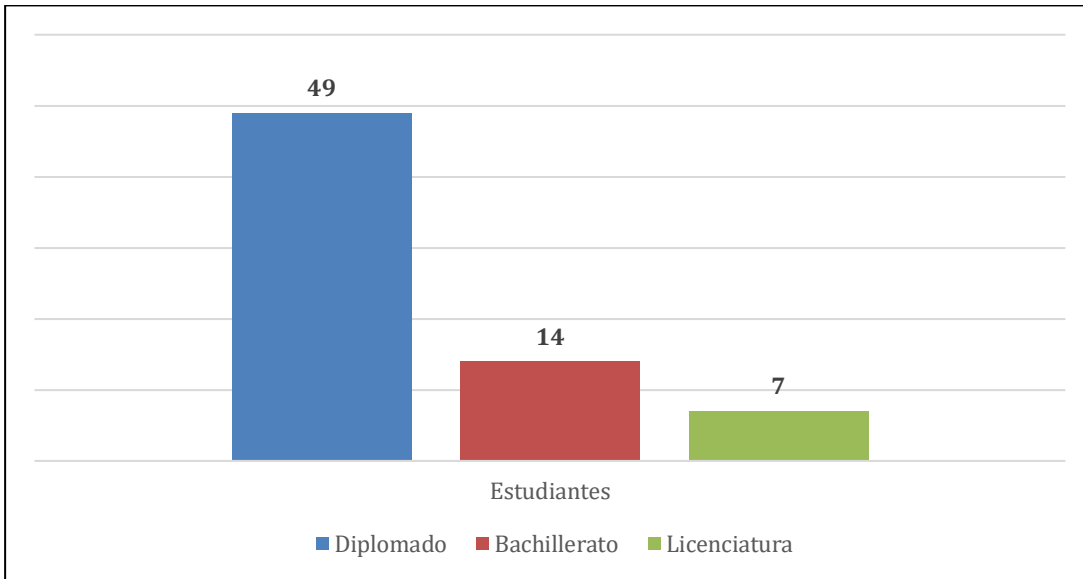
Como se menciona en el apartado de Evaluación de la herramienta, el plan piloto consta de dos partes, una demostración con el personal de la escuela de informática y un formulario aplicado a los estudiantes. A continuación, se presenta los resultados de ambas partes del plan piloto.

4.5.1.1. Estudiantes

En la etapa de pruebas se obtuvieron 70 respuestas, de las cuales 49 fueron de estudiantes de Diplomado, 14 fueron de estudiantes de Bachillerato y 7 fueron de estudiantes de Licenciatura.

Figura 19

NIVEL ACADÉMICO DE LOS ESTUDIANTES ENCUESTADOS

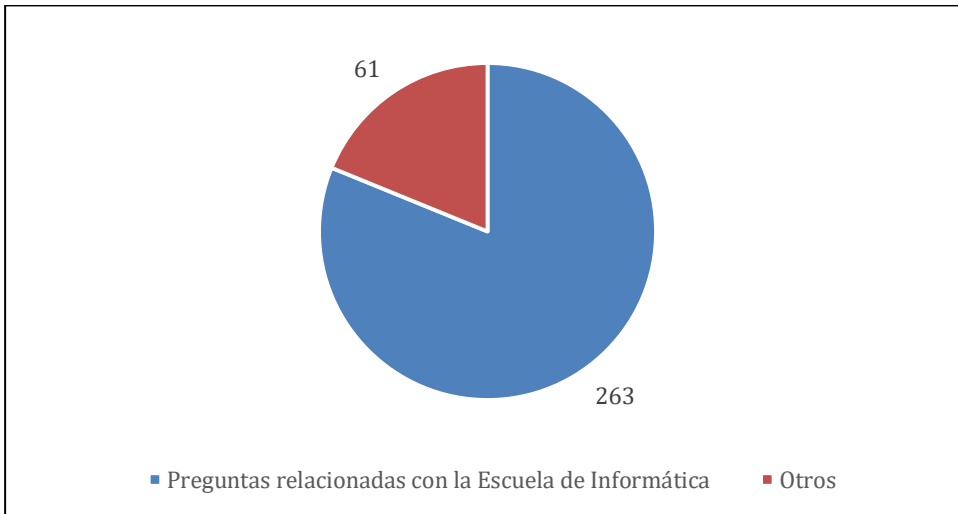


Fuente Elaboración propia

Las 70 respuestas recibidas representan 324 preguntas realizadas al chatbot, de las cuales 263 fueron relacionadas a temas de la escuela y 61 de ellas fueron sobre temas de otra índole.

Figura 20

CANTIDAD DE PREGUNTAS REALIZADAS POR LOS ESTUDIANTES

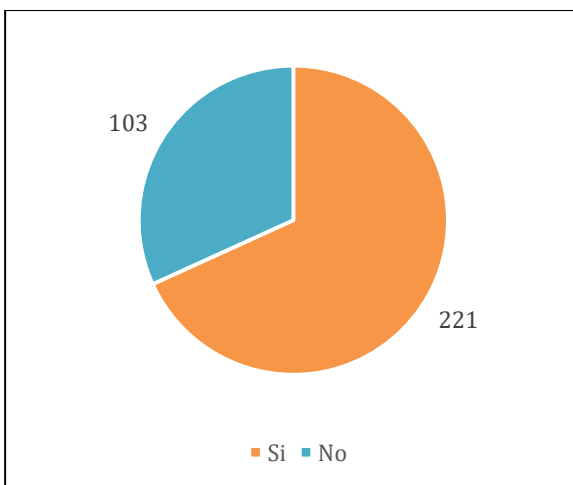


Fuente Elaboración propia

Dadas las preguntas realizadas al chatbot en el plan piloto, la herramienta respondió 221 preguntas y no mostró respuesta para 103 de ellas. Algunas de las preguntas realizadas por los estudiantes estaban mal redactadas, este factor afectó las respuestas del chatbot.

Figura 21

CANTIDAD DE PREGUNTAS RESPONDIDAS POR EL CHATBOT

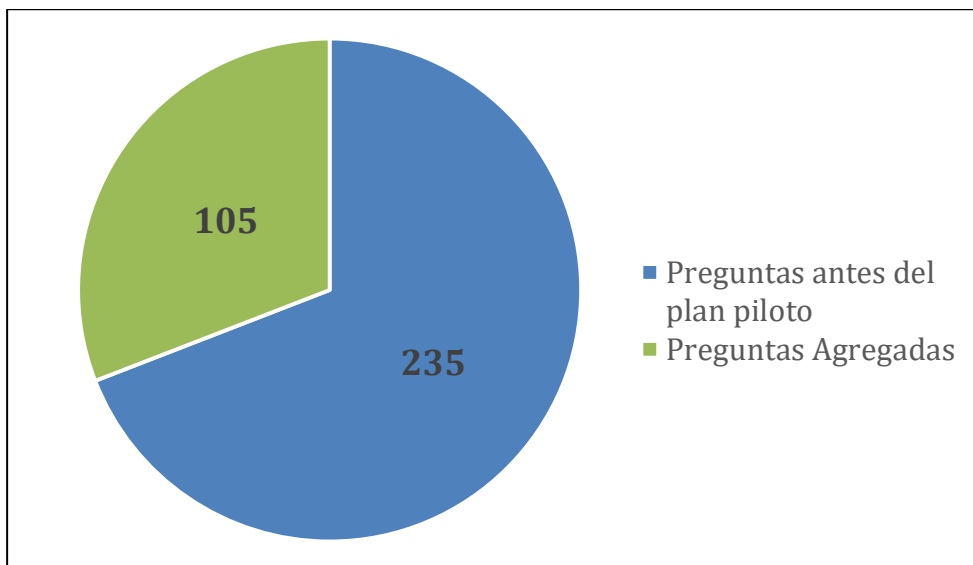


Fuente Elaboración propia

Al inicio del plan piloto la herramienta contaba con 235 preguntas, al finalizar esta etapa se agregan las consultas que los estudiantes hicieron, pero que la herramienta no tenía registradas, las cuales representan 105 nuevos registros, para un total de 340 preguntas.

Figura 22

CANTIDAD TOTAL DE PREGUNTAS CONTENIDAS EN EL CHATBOT



Fuente Elaboración propia

Como parte del proceso de evaluación del Chatbot se les preguntó a los estudiantes si consideraban que la herramienta ayudaría a mejorar el proceso de comunicación entre ellos y los funcionarios de la Escuela de Informática, para lo cual el 100% de las respuestas fueron afirmativas.

Una de las funciones que permite el chatbot al usuario, es sugerir preguntas al administrador del sistema, estas llegarán a un correo electrónico determinado previamente dentro de la programación de la herramienta. En esta etapa de pruebas se recibió 80 correos electrónicos con sugerencias por parte de los estudiantes.

4.5.1.2. Personal administrativo

Para el análisis del módulo de mantenimiento, se realizó una demostración con los miembros del personal de secretariado de la escuela, en la cual se presenta el chatbot, así como las funcionalidades del área de mantenimiento. Luego de realizar algunas pruebas se procede a abrir un espacio para preguntas, respuestas y recomendaciones.

Como parte de los resultados se menciona la creación de una nueva categoría llamada Maestría, la cual permitirá ingresar preguntas sobre este tema. También son mencionadas por parte de las funcionarias características del sistema como fácil usabilidad, disponibilidad y accesibilidad del chatbot como medio de comunicación para los estudiantes, en especial en días de alta demanda de atención al público como lo son las fechas de matrícula.

5. Análisis retrospectivo

Se realizó el desarrollo de una propuesta de Chatbot para la Escuela de Informática, está con el objetivo de brindar una opción de comunicación entre la escuela y sus estudiantes en relación con temas de información general. Para la creación del sistema se utilizaron herramientas de Software libre el cual permitió mantener bajo los costos del proyecto.

Luego de terminar la fase de desarrollo se inició las pruebas del Chatbot por medio de un plan piloto. En el plan piloto participaron 70 estudiantes de la escuela de Informática, así como personal administrativo. En esta etapa se captura los resultados de las preguntas realizadas a la herramienta y retroalimentación de los estudiantes como por ejemplo la inclusión de un área de ayuda en el Chatbot.

Además, en este apartado se analizará el cumplimiento de los objetivos propuestos para esta investigación:

5.1. Objetivo general:

Facilitar y mejorar la comunicación entre los estudiantes y el personal administrativo de la Escuela de Informática de la Universidad Nacional por medio de un chatbot utilizando herramientas de software libre.

La finalidad del proyecto era brindar una opción que facilitara el proceso de comunicación entre los estudiantes y el personal de la Escuela de Informática de la Universidad Nacional mediante una herramienta “chatbot”. Este sistema les permite a los usuarios obtener una respuesta rápida a la consulta que realicen, como consecuencia se da el ahorro en tiempo tanto para los estudiantes como para el personal administrativo ya que tendrá menos llamadas y consultas por parte de los estudiantes, por lo que favorecería a la Escuela de Informática el uso de la herramienta.

5.2. Objetivos específicos:

1. *Investigar las herramientas tecnológicas de software libre utilizadas para el diseño de un chatbot.*

Este objetivo es completado al realizar la investigación de las herramientas de software libre existentes para el diseño de chatbots o que cuenten con funcionalidades similares para el desarrollo e investigación de este proyecto, para con ello seleccionar la óptima y así implementarla en el proyecto.

2. *Desarrollar un chatbot basado en herramientas de software libre en la Escuela de Informática de la Universidad Nacional*

Este objetivo se completa con el desarrollo de una herramienta “chatbot” que le brinde una respuesta a la consulta que realice el usuario, además de un área de mantenimiento que ayudará a mantener la información actualizada, este objetivo está basado en la información recolectada en la investigación realizada en el primer objetivo específico. Para el desarrollo de este proyecto se utilizó la metodología de SCRUM es una metodología ágil y flexible que puede ser aplicada de manera regular, mediante un conjunto de buenas prácticas para trabajar colaborativamente y obtener el mejor resultado posible de un proyecto.

3. *Evaluar el uso del chatbot como medio para mejorar el proceso de comunicación entre los estudiantes y funcionarios de la Escuela de Informática de la Universidad Nacional mediante un proyecto piloto.*

Este objetivo se cumple con la evaluación del chatbot realizada por los estudiantes y el personal administrativo de la universidad, ellos valoraron la herramienta y cómo esta ayudaría a mejorar el proceso de comunicación entre ambas partes. Para esto se establecen distintos tipos de pruebas, no solo se persigue evaluar la funcionalidad de la herramienta, sino también la calidad que se ha obtenido por medio del seguimiento realizado, las pruebas fueron realizadas en dos extractos primero a los estudiantes con pruebas de funcionalidad y usabilidad y en un segundo extracto con el personal administrativo de la Escuela de Informática de la Universidad.

6. Conclusiones y recomendaciones

6.1. Conclusiones

La estrategia utilizada en el proyecto fue efectiva, el iniciar investigando las herramientas de software libre ayudó a adquirir conocimientos relacionados al tema de chatbots, el cual se pudo aplicar durante el proyecto. Asimismo, se genera una curva alta de aprendizaje para las estudiantes al utilizar estas herramientas, dado que fueron temas nuevos los cuales tuvieron que aprender, donde se puede mencionar Python y MongoDB las cuales su conocimiento es importante en el mercado laboral.

Al utilizar SCRUM como metodología para la etapa de desarrollo, permite obtener productos funcionales en cada sprint. Además de asegurar la funcionalidad de cada entregable, logrando de esta forma alcanzar el objetivo de una manera ágil.

A partir de la retroalimentación obtenida en el plan piloto, se agregaron dos mejoras a la herramienta. La primera recomendación fue recibida por parte del personal administrativo de la escuela, la cual fue agregar la categoría de maestría al chatbot. La segunda fue dada por los estudiantes la cual fue agregar un módulo de ayuda en la pantalla del chatbot el cual brindará instrucciones a los estudiantes para que puedan utilizar la herramienta.

Los resultados del plan piloto permiten conocer la opinión de los estudiantes sobre la implementación de un chatbot en la escuela y cómo esta herramienta ayudaría en el proceso de comunicación. Además, dado la crisis de COVID-19 que está enfrentando el país, un chatbot podría ser un recurso de gran ayuda para los estudiantes y para la escuela, brindando apoyo a la modalidad remota de estudio con la que está trabajando la Universidad Nacional.

Por otra parte, al finalizar la etapa de pruebas, se obtiene 340 preguntas que podrán servir de referencia a la escuela, ya sea para la implementación de la herramienta o como ayuda para mejorar la comunicación con la población estudiantil.

6.2. Recomendaciones

A continuación, se presenta una lista de recomendaciones de la herramienta:

- a. Se recomienda leer los manuales antes de hacer uso de la herramienta para poder sacar el mayor provecho al chatbot.
- b. La herramienta cuenta con varias preguntas cuya respuesta hace referencia a fechas o datos institucionales, los cuales cambian cada semestre. Por lo tanto, se recomienda revisar este tipo de preguntas al menos una vez por semestre con el objetivo de mantener la información actualizada.
- c. El chatbot es una herramienta web diseñada para ser utilizada en navegadores de computadoras, sin embargo, aprovechando las características de responsivo de la herramienta de Bootstrap puede ser utilizado en navegadores de celulares. No obstante, podría no ser práctico en el uso del módulo de mantenimiento en el momento de mostrar la información.
- d. Se recomienda continuar actualizando y agregando más preguntas al chatbot con el objetivo de ir enriqueciendo la base de datos.
- e. Algunas de las preguntas realizadas por los estudiantes no correspondían a información de la escuela, sin embargo, los administradores podrían agregar esas otras preguntas al sistema y así ampliar el área de conocimiento que podría soportar la herramienta.
- f. La herramienta puede ser utilizada en otras sedes de la Universidad Nacional.

7. Referencias bibliográficas

Aivo. (2018). *Inteligencia Artificial para la atención al alumno: un asistente virtual en la universidad*. Consultado el 01 de octubre del 2018 <https://aivo.co/blog/inteligencia-artificial-evolucionaria-educacion-con-agentbot/>

Angular. (2010). *Features and Benefits*. (version 10.0.3) [Software] Google <https://angular.io/features>

Arias, M. (2013). *Introducción a php*. CreateSpace Independent Publishing Platform.

Arsovski, Sasa., Muniru, Idris., y David Cheok, Adrian. (2017). *Analysis of the Chatbot open source languages aiml and chatscript*.

Benítez, L (s.f). *Ventajas y desventajas del uso de chatbots para una empresa*. Consultado el 1 de octubre del 2020 <https://www.novicell.es/es/blog/chatbots-empresa//>

Cardwee S. (2017). *Becky the chatbot offers her first university place*. Leeds Beckett University. <http://www.leedsbeckett.ac.uk/news/0817-becky-chatbot-clearing/>

Carrera, F. (2019). *Fuzzy String Matching in Python*. DataCamp. <https://www.datacamp.com/community/tutorials/fuzzy-string-python>

Castillo, N., Ricardo, J., Milton, P., y Jácome, D. (2017). Base de datos nosql: mongodb vs. Cassandra en operaciones crud (create, read, update, delete). Revista publicando. https://revistapublicando.org/revista/index.php/crv/article/view/398/pdf_342

Departamento de Registro UNA. (2015). *Horario de atención*. <https://www.registro.una.ac.cr/index.php/241-horario-de-atencion>

Eguíluz, J. (2008). *Introducción a JavaScript*. Uniwebsidad. <http://www.librosweb.es/javascript>

El mundo. (2016). *Jill Watson, la profesora que en realidad era un robot*.

<http://www.elmundo.es/tecnologia/2016/05/17/573aca4f268e3ee75c8b457a.html>

Escobar, T. (2017). *30 Herramientas para crear chatbots*. Webempresa elegantthemes Divi 3.0. <http://mediollena.com/30-herramientas-crear-chatbots/>

Joyanes, L. y Fernández, M. (2014). *Java 2 Manual de Programación*. Osborne MCGraw-Hill. Academia. https://www.academia.edu/4444866/Java_2_Manual_de_Programacion

Layla. (2018). *Layla A smart Agent Platform* (versión 1.0) [Software] IMPESA <https://www.layla.ai/>

Milani, A. (2006). *MySQL - Guia do Programador*. Novatec

MongoDB. Inc (2020). *MongoDB documentation*. <https://docs.mongodb.com/>

Moreno, M. (2017). *¿Qué es Node.js y para qué sirve?* Universia.
<https://noticias.universia.es/ciencia-tecnologia/noticia/2017/07/07/1154054/nodejs-sirve.html>

Nava, I. (2016). *5 Grandes marcas que usan bots como herramienta de marketing.* Merca2.0.
<https://www.merca20.com/5-grandes-marcas-usan-bots-herramienta-marketing/>

Oracle Corporation. (2016). *Chatbots.* <http://www.oracle.com/us/technologies/mobile/chatbot-infographic-3672253.pdf>

Oracle Corporation. (2016). *CX Research Virtual Experiences.*
https://www.oracle.com/webfolder/s/delivery_production/docs/FY16h1/doc35/CXResearchVirtualExperiences.pdf

Oficinas de Relaciones Públicas UNA. (2016). *Población estudiantil.*
<https://www.una.ac.cr/index.php/component/content/article/173-ambito-estudiantil/273-poblacion-estudiantil>

Oficinas de Relaciones Públicas UNA. (2016). *Población estudiantil Becada/Graduada.*
<https://www.una.ac.cr/index.php/component/content/article/173-ambito-estudiantil/274-pobacion-becada-graduada>

Python. (2020). *What is Python.* <https://www.python.org/doc/essays/blurb/>

Polatidis, Nikolaos. (2014). *"Chatbot for Admissions."* [Tesis de Maestría, Universidad de Birmingham]. SIDUNA

PostgreSQL. (2020). *About.* <https://www.postgresql.org/about/>

Real Academia Española. (2020). *Definición de Comunicación.*
<https://dle.rae.es/?w=comunicaci%C3%B3n>

Real Academia Española. (2017). *Definición de Lengua.* <http://dle.rae.es/?id=N77BOII>

SCRUM. (2015). *Introducción.* Scrum.
<https://queondara.readthedocs.io/en/latest/Scrum.html#fuentes>

SCRUM. (2020). *What is Scrum?* <https://www.scrum.org/resources/what-is-scrum>

Shawar, B. A., y Atwell, E. (2007, January). *Chatbots: are they really useful?* In Ldv forum Vol. 22(No. 1), pp. 29-49.

Sitio web Escuela de Informática Universidad Nacional (2018). *Acerca de la escuela.*
<https://www.escinf.una.ac.cr/index.php/quienes-somos/acerca-de-la-escuela>

Stallman, R. (2004). *Software libre para una sociedad libre.*
<http://libros.metabiblioteca.org/handle/001/144>

Tamayo P. (2018). *EconBot: un chatbot de apoyo a la enseñanza y aprendizaje en línea de Economía*. Planeta Chatbot <https://planetachatbot.com/econbot-construyendo-chatbot-de-economia-3c1ee6bc1bb9>

Toxtli. C., Monroy-Hernández. A., y Cranshaw, J. (2018). *Understanding Chatbot-mediated Task Management*. Cornell University. <https://arxiv.org/pdf/1802.03109.pdf>

Unit4. (2016). *Unit4 presenta su primer chatbot para el estudiante de Educación Superior*. Unit 4. <https://www.unit4.com/news>.

Universidad Nacional. (2018). *Contáctenos*. <https://www.una.ac.cr/>

Valencia A. (2015). *Investigación de Mercado: Supermercado*. [Proyecto de grado] https://repository.icesi.edu.co/biblioteca_digital/bitstream/10906/78527/1/TG00947.pdf

Van, L. (2012). *HTML5: los fundamentos del lenguaje*. Ediciones ENI.

24/7 Customer Inc. (2016). *A Retailer's Guide to Chatbots, Live Chat, and Messaging*. <https://info.247.ai/rs/074-HBW-141/images/RetailersGuidetoChatbots.pdf>

8. Anexo

Los documentos que incluye el anexo podrá encontrarlos en el material digital incluido en CD donde se entregará del trabajo:

- Carta de aceptación del director de la escuela
- Carta de aceptación de anteproyecto
- Carta de aceptación del tutor

9. Apéndices

a. Manual de usuario de chatbot y mantenimiento

El Manual de Usuario del Chatbot y Mantenimiento lo encontrarán adjunto en el CD del material.

b. Manual técnico

El Manual Técnico lo encontrarán adjunto en el CD del material.

c. Formulario “Chatbot”

El formulario se encuentra adjunto en el CD del material, el formulario utilizado para recolectar algunas de las preguntas que se encuentran en el chatbot.

d. Formulario “Chat Proyecto Piloto”

El formulario se encuentra adjunto en el CD del material, el formulario utilizado para recolectar las respuestas de los estudiantes en el proyecto piloto.

e. Plan de pruebas

El documento del plan de pruebas se encuentra adjunto en el CD del material.

10. Glosario

Chatbot: Programas computacionales que interactúan con personas utilizando lenguaje natural.

Comunicación: Transmisión de señales mediante un código común al emisor y al receptor (RAE, 2020).

Distancia de Levenshtein: Es una métrica la cual se utiliza para medir qué tan separadas están dos secuencias de palabras (Carrera, 2019).

Flask: Es un pequeño marco de trabajo para python el cual está basado en Werkzeug (biblioteca de Python que describe como se comunica un servidor web con una aplicación web) y utiliza Jinja para plantillas (Python, 2020)..

MongoDB: Es una base de datos de documentos NoSQL, escalable y flexible. (MongoDB, Inc, 2020).

SCRUM: Es un marco de trabajo dentro del cual las personas pueden abordar problemas complejos, mientras que productiva y creativamente entregan productos de alto valor. (SCRUM, 2020).

Software Libre: Es aquel que una vez obtenido puede ser usado, copiado, estudiado, modificado y redistribuido libremente.

Python: Es un lenguaje de programación el cual cuenta con las características de ser orientado a objetos, de alto nivel, dinámico. (Python, 2020)