



UNIVERSIDAD TECNOLÓGICA ISRAEL

MEMORIA DEL PROYECTO DE FIN DE CARRERA

INGENIERÍA EN SISTEMAS INFORMÁTICOS

**TEMA:
DISEÑO DE PROTOTIPO DE CHAT BOT DE FACEBOOK PARA
INSTITUCIONES EDUCATIVAS DE NIVEL SUPERIOR.**

**AUTOR:
Andrés Paul Dávila Bastidas**

**TUTOR:
MSc. Ing. Pablo Recalde**

**QUITO, ECUADOR
2020**

UNIVERSIDAD TECNOLÓGICA ISRAEL

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Titulación certifico:

Que el trabajo de titulación **“DISEÑO DE PROTOTIPO DE CHAT BOT DE FACEBOOK PARA INSTITUCIONES EDUCATIVAS DE NIVEL SUPERIOR”**, presentado por Andrés Paul Dávila Bastidas, estudiante de la Carrera Ingeniería en Sistemas Informáticos, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito, agosto de 2020

TUTOR

MSc. Ing. Pablo Recalde

TABLA DE CONTENIDOS

INTRODUCCIÓN.....	I
Antecedentes de la situación objeto de estudio	I
Planteamiento del problema	I
Justificación.....	II
Objetivos General.....	II
Objetivos específicos.....	II
Alcance.....	II
CAPÍTULO 1. PROPUESTA	1
1.1 Diagrama de procesos	1
1.1.1 Inicio de conversación	2
1.1.2 Flujo de conversación.....	3
1.1.3 Inicio de sesión y sincronización de alertas	4
1.2 Especificación de requerimientos	5
1.2.1 Ámbito del software	5
1.2.2 Funciones del producto.....	5
1.2.3 Características de los usuarios del sistema.....	8
1.2.4 Restricciones.....	9
1.2.5 Requisitos	9
1.2.6 Api Graph de Facebook.....	10
1.2.7 DialogFlow	13
1.2.8 Ngrok.....	15
CAPÍTULO 2. RESULTADOS	16
2.1 Diseño general.....	16
2.2 Esquema de la base de datos	17
2.3 Diagrama de la arquitectura del sistema	18
2.4 Diseño de interfaces	18

2.4.1 Ventana de chat de Facebook	19
2.4.2 Ventana de inicio de sesión web	19
2.4.3 Ventana administrativa	20
2.5 Estándares de programación utilizados	20
2.5.1 Arquitectura	20
2.5.2 Codificación	20
2.5.3 Base de datos	21
2.6 Pruebas	21
2.7 Implementación	24
2.8 Factibilidad técnica y económica	24
2.8.1 Factibilidad técnica:	24
2.8.2 Factibilidad económica:	25
2.8.3 Requerimientos Hardware y software	26
CONCLUSIONES	28
RECOMENDACIONES	29
REFERENCIAS BIBLIOGRÁFICAS	30
ANEXOS	31

ÍNDICE DE FIGURAS

Figura 1.1 Inicio de conversación.....	1
Figura 1.2 Flujo de conversación.....	3
Figura 1.3 Inicio de sesión y sincronización de alertas	4
Figura 1.4 Quick Replies	10
Figura 1.5 Plantillas de mensajes.....	11
Figura 1.6 Código Facebook Api.....	11
Figura 1.7 Entidades Api Graph	12
Figura 1.8 Código WebHook.....	12
Figura 1.9 Diagrama de Dialog Flow	13
Figura 1.10 Parametrización de palabras en DialogFlow	14
Figura 1.11 Definición de Intent DialogFlow.....	14
Figura 1.12 Definición de método en Java	15
Figura 1.13 Ejecución de Ngrok	15
Figura 2.1 Esquema de la base de datos	17
Figura 2.2 Diagrama de la arquitectura del sistema.....	18
Figura 2.3 Ventana de chat de Facebook	19
Figura 2.4 Ventana de inicio de sesión web	19
Figura 2.5 Administración web	20
Figura 2.6 Ejecución de Quickstarts	22
Figura 2.7 Tiempo de respuesta.....	22
Figura 2.8 Tiempo de respuesta (Ampliación)	23
Figura 2.9 Ejecución de Wildfly Project	23
Figura 2.10 Ejecución de Report Issue	24

ÍNDICE DE TABLAS

Tabla 1.1 Interacción con el chat bot.....	5
Tabla 1.2 Solicitud de información/conversación	6
Tabla 1.3 Sincronización de horario de estudiante con alertas de chat.	6
Tabla 1.4 Generación de Pop Up para acceso a web externa	7
Tabla 1.5 Generación entorno web para la subida de información	7
Tabla 1.6 Generar cambio y restablecimiento de contraseña.	8
Tabla 1.7 Tipos de usuarios	8
Tabla 2.1 Product Backlog.....	16
Tabla 2.2 Recursos técnicos para el desarrollo del proyecto	25
Tabla 2.3 Recursos humanos	25
Tabla 2.4 Recursos tecnológicos (hardware).....	26
Tabla 2.5 Recursos tecnológicos (software)	26

INTRODUCCIÓN

El presente proyecto se ha desarrollado con la finalidad de poner en práctica los conocimientos obtenidos durante el transcurso de la carrera de ingeniería en sistemas de la información, además de aplicar herramientas investigativas las cuales brindarán una visión necesaria para implementar tecnologías de comprensión del lenguaje natural por medio del desarrollo de un chat bot.

Antecedentes de la situación objeto de estudio

Analizando el mundo actual en el que se vive, sobre todo la inesperada situación la que azoto al mundo recientemente es decir el aislamiento, se considera primordial contar con medios de comunicación que difundan no solo noticias, sino también temas de interés personal, es por esto que a la par de la creación de las redes sociales ha surgido un nuevo auge de las comunicaciones, en el cuál cada persona se ha vuelto un transmisor de información el mismo que usa como medio transmisor sus propias redes sociales y a su vez las usa para realizar temas que en la antigüedad solían realizarse de manera presencial, este auge de las comunicaciones no solo representa un cambio en la forma de interactuar, sino que representa una gran oportunidad de emprendimiento, desarrollo y sobre todo de nuevos mercados, es por esto que la creación de un chat capaz de brindar información automáticamente suena la mejor opción ante este auge y despunte de las redes sociales.

Planteamiento del problema

En la actualidad son pocas las instituciones educativas que cuentan con un chat capaz de responder mensajes automáticamente, y ante la situación extrema de aislamiento que vive el mundo es primordial crear medios capaces de brindar información, pero esto a su vez representa un gasto ya que al contar con más accesos de comunicación por obvias razones debe ser mayor el personal dispuesto para dar respuesta a los requerimientos del

público, es por esto que se plantea la opción de implementar una aplicación capaz de solventar y brindar información a los cuestionamientos del público de manera desatendida, por este motivo se toma como referencia a una Universidad, la misma que cuenta actualmente con un chat de Facebook, sin embargo el mencionado chat no posee una interacción dinámica que brinde al usuario comodidad al usarlo, y no permite una interacción continua con el mismo.

Justificación

La visión con la que se inicia este proyecto, es la de brindar una herramienta capaz de solventar información y permitir interacción con el usuario de manera desatendida, con la creación de este chat bot se planea automatizar el proceso de consulta de estudiantes y público en general acerca de las opciones que brindan las Instituciones de Educación Superior, adicional a esto se prevé realizar un flujo de conversación continuo, el cual mantenga el interés del usuario y le permita realizar un número ilimitado de veces sus consultas.

Objetivos General

Diseñar un prototipo de chat bot de Facebook para Instituciones Educativas de Nivel Superior.

Objetivos específicos

- Desarrollar un chat bot adaptable a los requerimientos de la Institución Educativa de Nivel Superior.
- Usar la página de Facebook de la Institución Educativa de Nivel Superior para la implementación del chatbot.
- Implantar un servidor Wilfly demo de la solución en la nube.
- Generar sincronización de alertas de clases.

Alcance

Se implementará en una página de Facebook de la Institución Educativa de Nivel Superior. En ella se mostrarán datos acerca de la Institución, servicios brindados, carreras, datos y formas de contacto, se permitirá interacción con el público en general esto con el

fin de mostrar datos reales, actualizados y que permitan simular una interacción válida, esta página contará con un chat el cual estará programado para dar respuesta automática a las peticiones del usuario, además permitirá iniciar la sesión a los estudiantes y generar alertas previas a cada una de sus clases, se planea lograr una interacción dinámica e interesante.

CAPÍTULO 1. PROPUESTA

A continuación, se detallan los conceptos de la propuesta planteados para este trabajo, cada concepto fue analizado y desarrollado de manera en que se manifiesten todos los conocimientos aprendidos durante el transcurso de la carrera.

1.1 Diagrama de procesos

En esta sección se describen los procesos gráficamente mediante diagramas de flujo los cuales detallan las funciones de la aplicación:

Para iniciar conversación con el chat bot es necesario contar con una cuenta de Facebook, la misma por la cual se accederá a la página de la aplicación y por medio de la ventana de chat se realizarán las distintas conversaciones, actualmente el chat bot esta entrenado para reconocer palabras relacionadas con la universidad, las mismas que se presentan como botones en el menú principal y nos permiten tener una interacción más sencilla.

Al ser una ventana de chat nativa de Facebook cuenta con las ventajas de diseño y tamaños responsivos, adaptables a cualquier dispositivo (teléfono móvil, tablet, computadora).

Los diagramas de procesos de a continuación se realizaron con el objetivo de reducir al máximo los pasos para comunicarse con el chat bot, teniendo en cuenta conceptos de Customer Engagement y estrategias de marketing empleadas para controlar la cantidad de clickeos que debe dar el usuario para obtener una respuesta de la aplicación, obteniendo de esta manera una interacción más fluida con la aplicación.

1.1.1 Inicio de conversación

Para iniciar la conversación es necesario tener una cuenta de Facebook y completar el proceso de inicio de sesión.



Figura 1.1 Inicio de conversación

Autor: Andrés Dávila Bastidas

1.1.2 Flujo de conversación

En esta opción se debe digitar manualmente la opción que se quiere visualizar.

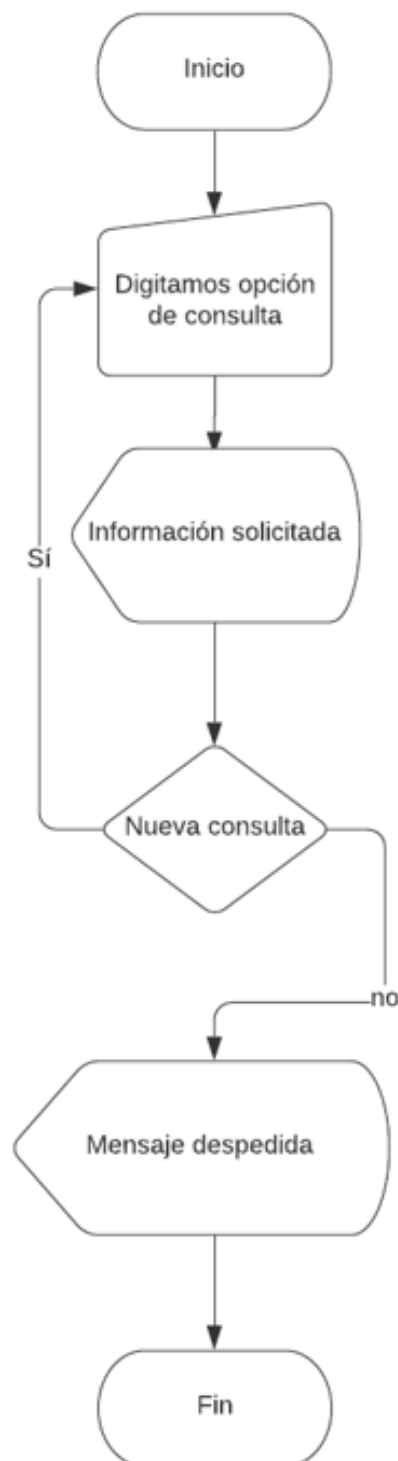


Figura 1.2 Flujo de conversación

Autor: Andrés Dávila Bastidas

1.1.3 Inicio de sesión y sincronización de alertas

Se puede iniciar sesión con las credenciales propias de la institución Universitaria para sincronizar nuestro horario y generar alertas para cada clase.

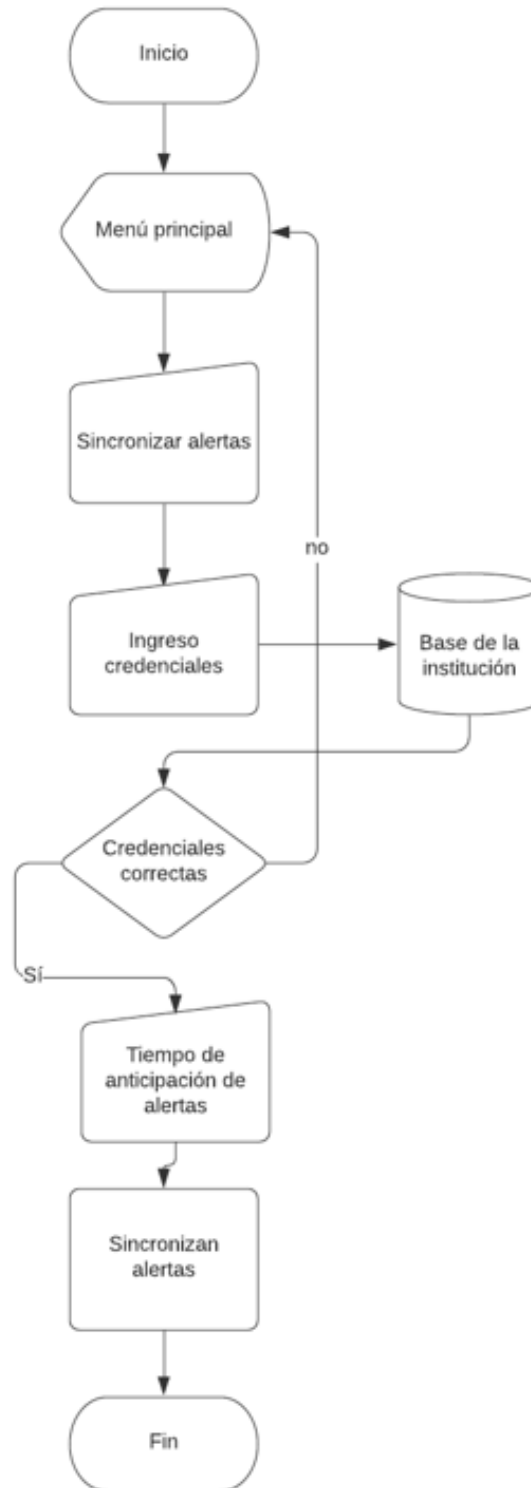


Figura 1.3 Inicio de sesión y sincronización de alertas
Autor: Andrés Dávila Bastidas

1.2 Especificación de requerimientos

En esta sección se tratará acerca de los requerimientos necesarios para lograr una interacción óptima con la aplicación.

1.2.1 Ámbito del software

Con el uso de este chat bot se planea tener una interacción dinámica en los procesos que requieren los estudiantes y el público en general, para de esta manera agilizar recursos y brindar una experiencia tecnológica a los usuarios además de esto permitirá a los estudiantes sincronizar el chat de Facebook con su horario, el cual generará alertas programadas como recordatorio de cada clase por medio del chat.

La aplicación no permitirá el registro de usuarios nuevos.

1.2.2 Funciones del producto

En esta sección se detallan una a una las funciones de la aplicación.

Se describen los usuarios y una breve descripción, analizando la función, el rendimiento, las restricciones, las interfaces y la fiabilidad del aplicativo. Se evalúan las funciones descritas en la declaración del ámbito.

HISTORIA DE USUARIO	
Número: 1	Usuario: Estudiante, usuario de Facebook
Nombre: Interacción con el chat bot	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 1
Descripción: Los usuarios deben estar registrados en Facebook para poder interactuar con el chat. <ul style="list-style-type: none">- El usuario debe tener cuenta válida en Facebook- Se restringe la interacción para usuarios logueados con Facebook	
Observación: Usuarios no registrados en Facebook pueden visitar el Fan Page, pero no pueden interactuar con el chat.	

Tabla 1.1 Interacción con el chat bot

HISTORIA DE USUARIO

Número: 2**Usuario:** Estudiante, usuario de Facebook**Nombre:** Solicitud de información/conversación**Prioridad en negocio:** Alta**Riesgo en Desarrollo:** Alta**Puntos estimados:** 1**Iteración asignada:** 1**Descripción:** Los usuarios pueden solicitar información al chat

- El usuario debe digitar/presionar una opción válida
- La información será obtenida directamente de la base de datos de la aplicación

Observación: En el caso de presionar o digitar una opción inválida el chat mostrará un mensaje de error y automáticamente redireccionará al usuario al menú principal de conversación.

Tabla 1.2 Solicitud de información/conversación

HISTORIA DE USUARIO

Número: 3**Usuario:** Estudiante activo**Nombre:** Sincronización de horario de estudiante con alertas de chat.**Prioridad en negocio:** Alta**Riesgo en Desarrollo:** Alta**Puntos estimados:** 1**Iteración asignada:** 1**Descripción:** Los estudiantes con semestre activo en la institución pueden loguearse y sincronizar su horario actual de clases para generar recordatorios para cada una de las clases.

- El usuario debe tener semestre activo.
- El estudiante debe estar matriculado en al menos una materia.
- Se debe indicar el tiempo de anticipación para el recordatorio.

Observación: Las alertas serán generadas como mensajes directos al usuario, por lo cual debe el usuario tener iniciada su sesión con Facebook para recibirlas.

Tabla 1.3 Sincronización de horario de estudiante con alertas de chat.

HISTORIA DE USUARIO

Número: 4**Usuario:** Estudiante, usuario de Facebook**Nombre:** Generación de Pop Up para acceso a web externa (web de la Universidad)**Prioridad en negocio:** Alta**Riesgo en Desarrollo:** Alta**Puntos estimados:** 1**Iteración asignada:** 1

Descripción: En caso de necesitar información adicional que no se encuentre en la base de datos de la aplicación, el chat redireccionara a la web de la Universidad automáticamente por medio de pop ups.

- El usuario interactúa con la página externa por medio del chat.
- La página web debe ser responsiva, para que pueda ser correctamente visualizada.

Observación: El chat únicamente levanta el pop up con la web externa, la interacción con la página ya es propia de dicha web, la página a la que se acceda debe ser responsiva para asegurar el correcto manejo de la misma.

Tabla 1.4 Generación de Pop Up para acceso a web externa

HISTORIA DE USUARIO

Número: 5**Usuario:** Administrador**Nombre:** Generación entorno web para la subida de información**Prioridad en negocio:** Alta**Riesgo en Desarrollo:** Alta**Puntos estimados:** 1**Iteración asignada:** 1

Descripción: La información mostrada por el chat será cargada en esta sección.

- Se permite crear nuevas opciones.
- La información será cargada desde la web y se mostrará por medio del chat.

Observación: El usuario administrador es único, permite botones para nuevas opciones de interacción, para que los cambios se vean representados se deben reiniciar las conversaciones inmediatamente después de la carga de las nuevas opciones.

Tabla 1.5 Generación entorno web para la subida de información

HISTORIA DE USUARIO

Número: 6**Usuario:** Administrador**Nombre:** Generar cambio y restablecimiento de contraseña.**Prioridad en negocio:** Alta**Riesgo en Desarrollo:** Alta**Puntos estimados:** 1**Iteración asignada:** 1**Descripción:** Se creará la opción para reestablecer la contraseña del administrador.

- Se enviará un código al correo para reestablecer la contraseña

Observación: El usuario administrador es único y es el encargado de subir los datos de la institución universitaria.

Tabla 1.6 Generar cambio y restablecimiento de contraseña.

1.2.3 Características de los usuarios del sistema

A continuación, se describe a los usuarios que pueden tener interacción con el chat bot, se detallan sus actividades y su área de acceso.

La aplicación puede interactuar con usuarios nativos de Facebook y con usuarios registrados en la universidad, los mismos que pueden acceder a funciones especiales, tales como sincronización materias con alertas.

En la siguiente tabla se muestra el tipo de usuario que puede acceder e interactuar con el chat bot.

Nombre de Usuario	Tipo de Usuario	Área Funcional	Actividad
Administrador	Administrador del sistema	Administración	Encargado de cargar la información desde la web, para que el chat bot la muestre.
Estudiante	Usuario registrado en la Universidad	Chatbot	Visualización de opciones. Logueo y sincronización de horario.,
Usuario de Facebook	Usuario registrado en Facebook	Chatbot	Visualización de opciones.

Tabla 1.7 Tipos de usuarios

1.2.4 Restricciones

- Se debe implementar un factor de seguridad de envío de correos en cuanto el estudiante ingrese con las credenciales propias de la institución.
- El chat mostrará únicamente información de servicios y temas de la institución.
- No se permitirán la realización de trámites propios de la universidad desde el chat.

1.2.5 Requisitos

Para la correcta implementación del aplicativo se debe cumplir con los siguientes requisitos:

Funcionales.

A continuación, se detallan los requerimientos funcionales:

RF01: Generar botones de acción para las opciones de la Universidad.

RF02: Se debe permitir tanto presionar el botón como digitar el texto para acceder a las opciones.

RF03: Se debe implementar un ambiente Web para gestionar las opciones del chat, ingreso y edición de opciones

RF04: Se debe implementar un factor de seguridad de notificación mediante correo en cuanto el estudiante inicie sesión.

RF05: El chat mostrará un mensaje de error en cuanto se digite una opción no válida.

RF06: El chat debe permitir un flujo continuo, es decir que al finalizar el flujo se debe regresar al menú principal

RF07: Se debe solicitar el tiempo previo para el envío de notificaciones mediante el chat.

No funcionales.

A continuación, se detallan los requerimientos no funcionales:

RNF01: Se debe validar el ingreso de opciones en el chat, esto con el fin de evitar SQL Injection.

RNF02: El usuario debe estar registrado en Facebook.

RNF03: Se debe enviar una alerta al correo del usuario que inicia sesión para sincronizar sus horarios

.

1.2.6 Api Graph de Facebook

La API Graph es la principal herramienta, la misma que permite que la aplicación pueda realizar tareas de lectura y escritura usando la interfaz nativa de Facebook. Entender el funcionamiento de la API Graph es una de las tareas fundamentales que se debe conocer, ya que la aplicación interactúa con ella de cierta forma por medio del API.

En el presente sistema el Api de Facebook es la primera interacción con la que el usuario se encontrará, ya que por medio del chat nativo de Facebook el usuario envía peticiones en forma de texto para que el Api de Facebook las procese.

Para una mas adecuada interacción con el aplicativo se hizo uso de herramientas propias del Api, las cuales son descritas a continuación:

Botones Quick Replies

Estos botones actúan como accesos directos al texto, y permiten al usuario realizar una solicitud solamente con clicar a una de las opciones como se muestra a continuación:

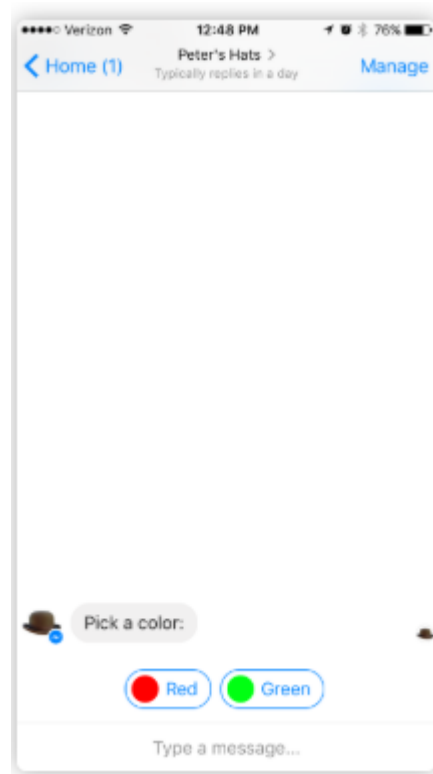


Figura 1.4 Quick Replies
Autor: Facebook API GRAPH, (2020).

Message Templates

Este tipo de plantilla facilita la comunicación y sobre todo la presentación de un producto, en la aplicación se usarán este tipo de botones para dar un look and feel más atrayente para el usuario y de esta forma otorgar opciones personalizadas para cada uno de los menús presentados.

A continuación, se observan dos plantillas distintas que pueden ser aplicadas:

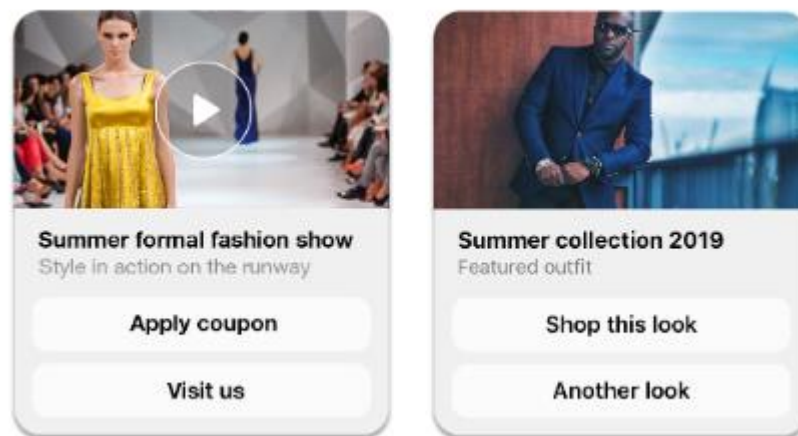


Figura 1.5 Plantillas de mensajes
 Autor: Facebook API GRAPH, (2020).

Para el uso de este Api Graph de Facebook se debe implementar el Api en nuestro proyecto como se observa a continuación:

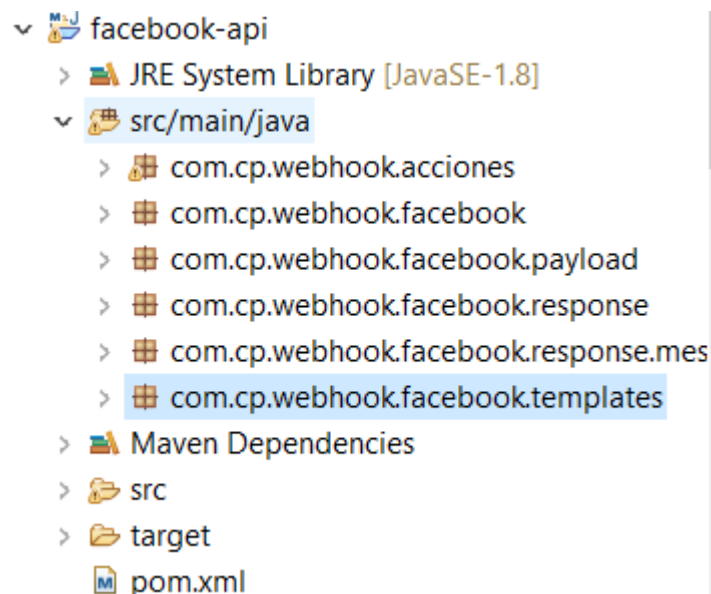


Figura 1.6 Código Facebook Api
 Autor: Andrés Dávila Bastidas

Es indispensable implementar todas las entidades necesarias para el envío y recepción de peticiones por medio del Api Graph como se muestra a continuación.

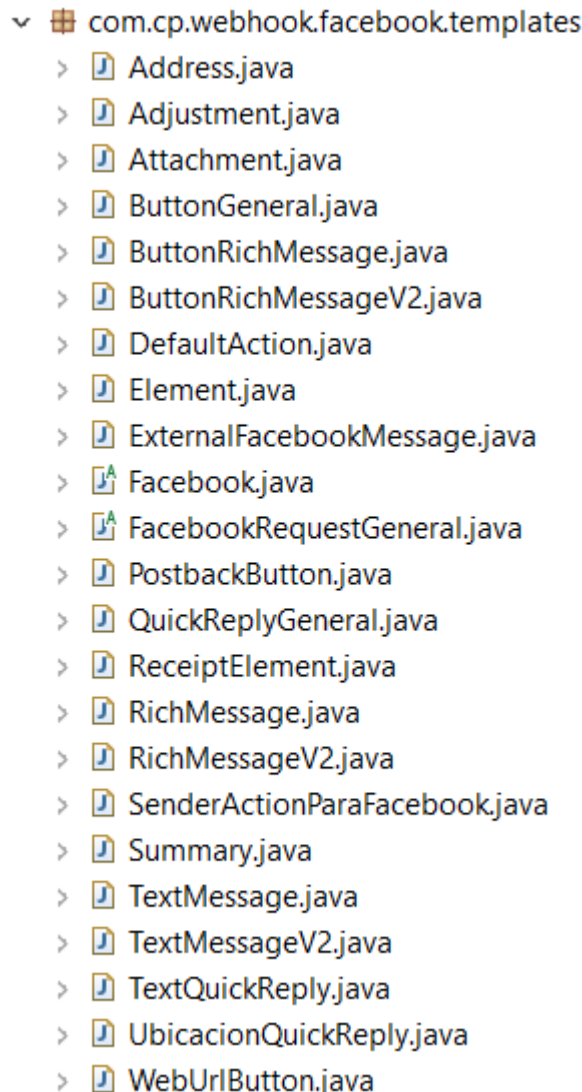


Figura 1.7 Entidades Api Graph
Autor: Andrés Dávila Bastidas

Adicional a esto se debe realizar la comunicación con el WebHook por medio de una palabra clave la misma que es definida en el Api de Facebook:

```
final static Logger Log = LoggerFactory.getLogger(ConsultarFacebook.class);

private final static Gson GSON = new GsonBuilder().disableHtmlEscaping().create();

private static final String facebookUri = "https://graph.facebook.com/v4.0/";
private static final String claveToken = "probandoWebhook";
private static final String paramFieldsUsuario = "&fields=first_name,last_name,profile_pic,birthday";
private static final String paramFieldsUbicacionUsuario = "&fields=location";
private static final String messagesUri = "me/messages";
private static final String attachmentsUri = "me/message_attachments";
private static final String whitelistUri = "/me/messenger_profile";

private static String getParamAccesToken(String facebookToken) {
    return "?access_token=" + facebookToken;
}
```

Figura 1.8 Código WebHook
Autor: Andrés Dávila Bastidas

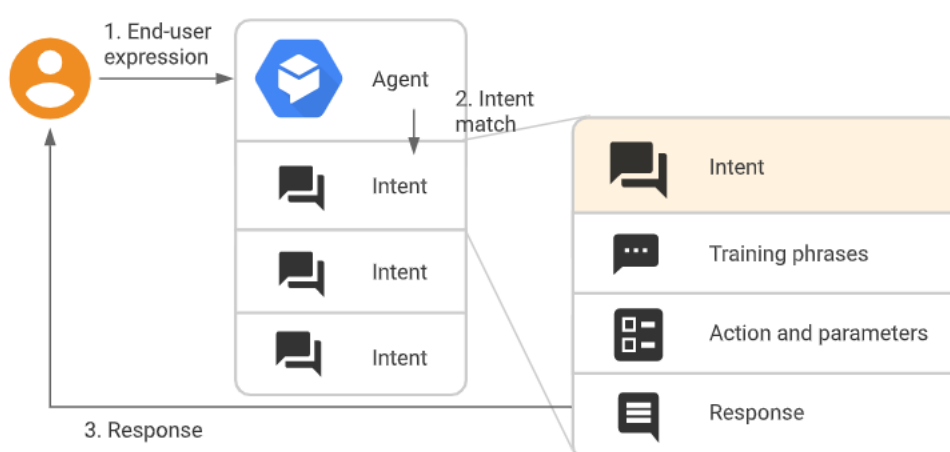
1.2.7 DialogFlow

DialogFlow es una herramienta de Google la misma que permite la creación de chatbots capaces de entender el lenguaje natural que el mismo Google pone a disposición.

Dialog Flow actúa como intermediario entre el Api de Facebook y el servidor Wildfly de aplicaciones, el momento en que el usuario digita o selecciona una opción del chat de Facebook de inmediato entra DialogFlow en acción, este verifica el texto previamente configurado por nuestra parte y lo transforma en un Intent, el mismo que es enviado al servidor de aplicaciones y dispara un evento, cada texto dispara eventos distintos los mismos que devuelven distintas respuestas validada para cada una de las interacciones, DialogFlow es capaz de aprender respuestas por defecto, las mismas que se dispararán cuando la conexión con el servidor se pierda, además DialogFlow cuenta con un módulo de reconocimiento de voz el cual podría ser implementado en futuras versiones del aplicativo. Se programo Dialog Flow para aceptar las distintas peticiones enfocadas a los servicios de la Institución Universitaria.

DialogFlow permite la conexión por medio de Webhooks con Facebook Messenger, Whatsapp, Instagram, y la gran mayoría de redes sociales de mayor audiencia y renombre en la actualidad, otra de las ventajas y por la cual fue seleccionada para el proyecto es porque es de libre uso y no tiene restricciones de implementación.

En el diagrama siguiente se muestra como trabaja el intérprete de DialogFlow:



*Figura 1.9 Diagrama de Dialog Flow
Autor: Google Cloud.*

En el siguiente gráfico se puede observar la configuración de cada una de las palabras en el DialogFlow:

Training phrases ? Search training phrases 🔍 ^

” Add user expression

” Me parece muy interesante. Por favor detalles

” Quiero más información sobre la empresa 🗑

” Quiero chatear con alguien

” Quiero más información

” como funciona

” ayudame

” como funciona?

” ayuda

” ayúdame

” reiniciemos

1 OF 2 →

Figura 1.10 Parametrización de palabras en DialogFlow
Autor: Andrés Dávila Bastidas

Las expresiones del gráfico anterior desencadenan en un Intent, el mismo que es enviado a WildFly en forma de un método, y este método realiza una acción de respuesta hacia el usuario.

Action and parameters ^

mostrarServicios

REQUIRED ?	PARAMETER NAME ?	ENTITY ?	VALUE	IS LIST ?
<input type="checkbox"/>	estadoConversacion	@estadoConversacion	SestadoConversacion	<input type="checkbox"/>
<input type="checkbox"/>	Enter name	Enter entity	Enter value	<input type="checkbox"/>

+ New parameter

Figura 1.11 Definición de Intent DialogFlow
Autor: Andrés Dávila Bastidas

En el siguiente gráfico se muestra la definición del método que es enviado desde DialogFlow:

```

-- ----,
case AccionesApiAi.MOSTRAR_SERVICOS:
    resultadoJson = GSON.toJson(conversar.mostrarServicios());
    break;
case AccionesApiAi.SALUDAR:
    resultadoJson = GSON.toJson(conversar.saludar(usuario));
    break;
case AccionesApiAi.CONVERSACION_DUMMY:
    resultadoJson = GSON.toJson(conversar.conversarDummyV2(resultAi, usuario));
    break;
-- ----,

```

Figura 1.12 Definición de método en Java
Autor: Andrés Dávila Bastidas

1.2.8 Ngrok

Esta herramienta permite crear un túnel público y accesible a través de un dominio que nos asigna la misma aplicación, la que permite acceder a un servidor local, fue usada para permitir la conexión entre el Api Graph de Facebook, el servidor de aplicaciones WildFly y Dialog Flow, ya que por políticas de seguridad el Api únicamente permite conexiones Https, las mismas que deben contar con un certificado SSL (ver anexo 1).

```

C:\Users\andres.davila\Desktop\ngrok.exe - ngrok.exe http 8080
ngrok by @inconshreveable (Ctrl+C to quit)

Session Status      online
Session Expires    7 hours, 59 minutes
Version             2.3.35
Region              United States (us)
Web Interface       http://127.0.0.1:4040
Forwarding           http://87c8ed97d8ac.ngrok.io -> http://localhost:8080
                   https://87c8ed97d8ac.ngrok.io -> http://localhost:8080

Connections
  ttl  opn  rt1  rt5  p50  p90
   0    0    0.00 0.00 0.00 0.00

```

Figura 1.13 Ejecución de Ngrok
Autor: Andrés Dávila Bastidas

CAPÍTULO 2. RESULTADOS

En este capítulo se describen los resultados obtenidos durante el desarrollo del presente proyecto.

2.1 Diseño general

La metodología aplicada en este proyecto es SCRUM por lo tanto a continuación se procede a detallar las historias de usuario:

<u>Product Backlog</u>			
Historia	Prioridad	Descripción	Estimación de esfuerzo
1	Muy alta	Diseño del flujo de conversación	100
2	Muy alta	Diseño de base de datos	100
3	Muy alta	Creación de la base de datos	70
4	Muy alta	Creación de fan page de Facebook	30
5	Muy alta	Creación de modelos de entidades	50
6	Alta	Implementación del API de Facebook	70
7	Alta	Creación y configuración de agente en Dialog Flow	70
8	Alta	Levantamiento de servidor (Wildfly)	80
9	Alta	Integración de Dialog Flow con la aplicación	50
10	Media	Entrenamiento de Dialog Flow	40
11	Alta	Creación de flujo de conversación	100
12	Alta	Diseño de interfaz de chat	40
13	Baja	Carga de contenido	80
14	Baja	Diseño de interfaz web	40
15	Muy Alta	Pruebas unitarias	100

Tabla 2.1 Product Backlog

2.2 Esquema de la base de datos

En esta sección se detalla la realización de la base de datos, fue diseñada con un enfoque estándar para la definición de las opciones de cada Institución Educativa de Nivel Superior, está diseñado para que sea adaptable para cada requerimiento y permite el almacenamiento de cada usuario que interactúa con el chat por medio de su identificador único de Facebook el mismo que brinda la opción de acceder a datos personales del usuario, permitiendo así una comunicación y respuestas personalizadas para cada caso

Componente práctico

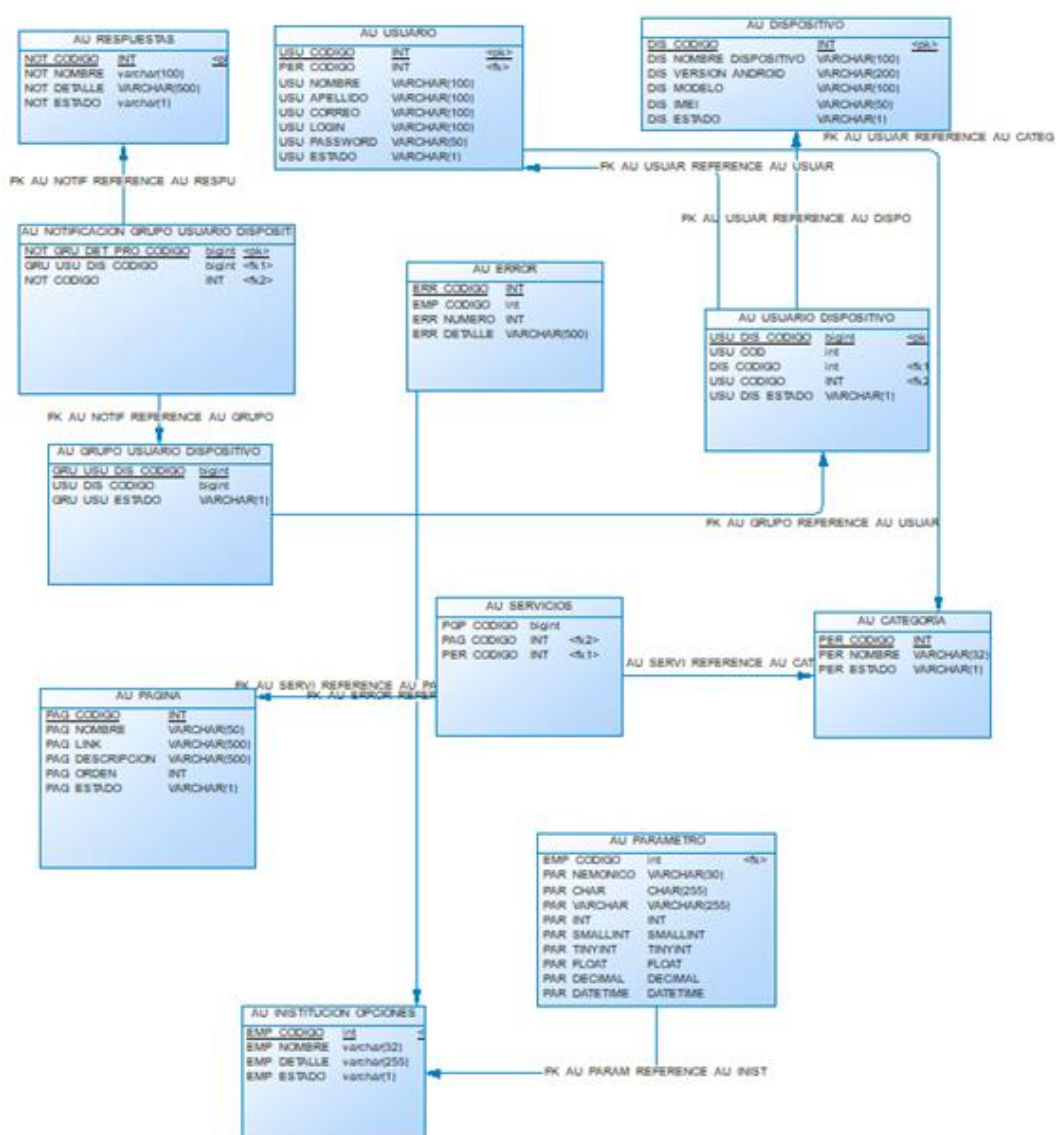


Figura 2.2 Esquema de la base de datos
Autor: Andrés Dávila Bastidas

2.3 Diagrama de la arquitectura del sistema

En la siguiente figura se muestra la relación que se establece entre cada uno de los componentes del sistema.

El cuadro resaltado en azul es la parte esencial del sistema, ya que es ahí donde se configura el proceso el texto enviado desde Messenger, el mismo que es interpretado y transformado en un Intent, el mismo que es transmitido por Webhook como petición al servidor de aplicaciones WildFly, el que se encarga de traducir la el Intent, realizar una acción y devolver una respuesta oportuna, acorde al contexto de la conversación (ver anexo 2).

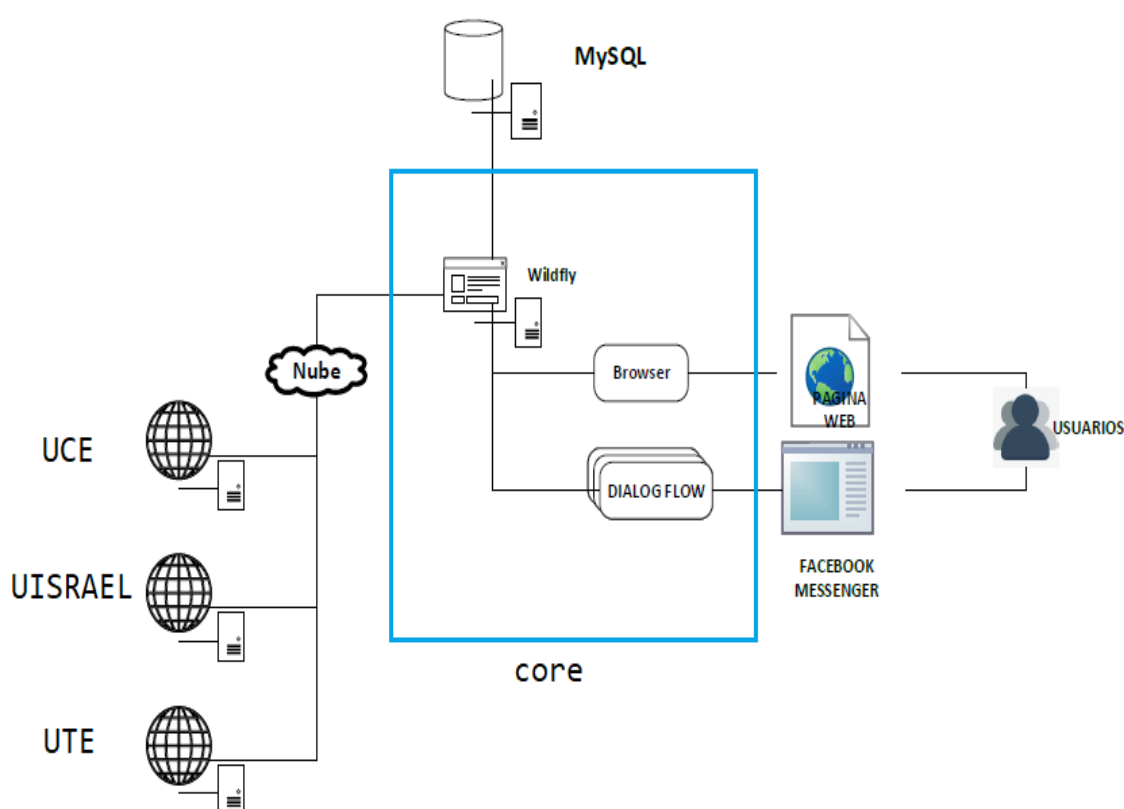


Figura.2.3 Diagrama de la arquitectura del sistema
Autor: Andrés Dávila Bastidas

2.4 Diseño de interfaces

Al tratarse de un chat que trabaja bajo la plataforma de Facebook son limitadas las opciones de edición de interfaz que se tiene, sin embargo, haciendo uso de las diferentes herramientas que nos brinda el API de Facebook se ha establecido un modelo estético y visualmente agradable.

Las interfaces son responsivas por lo que, sin importar el dispositivo, se adecuan automáticamente a cualquier tamaño.

2.4.1 Ventana de chat de Facebook

Ventana de chat de Facebook por defecto.

En esta ventana el usuario puede interactuar con la aplicación por medio del chat nativo de Facebook (ver anexo 3).

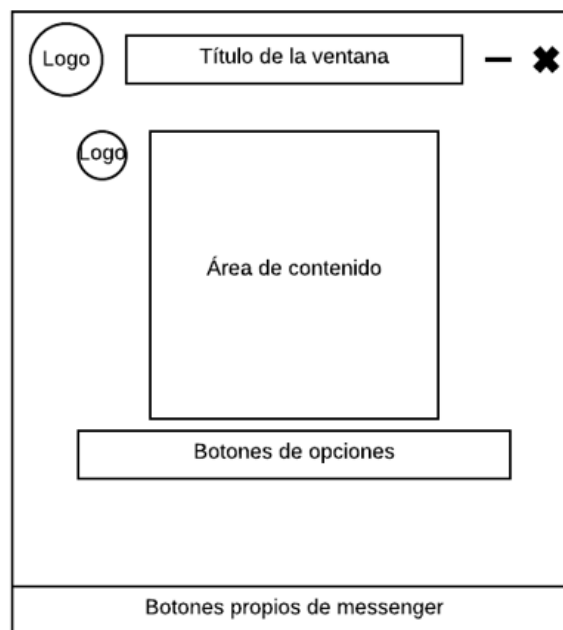


Figura 2.4 Ventana de chat de Facebook
Autor: Andrés Dávila Bastidas

2.4.2 Ventana de inicio de sesión web

Ventana mostrada para iniciar sesión mediante la web.

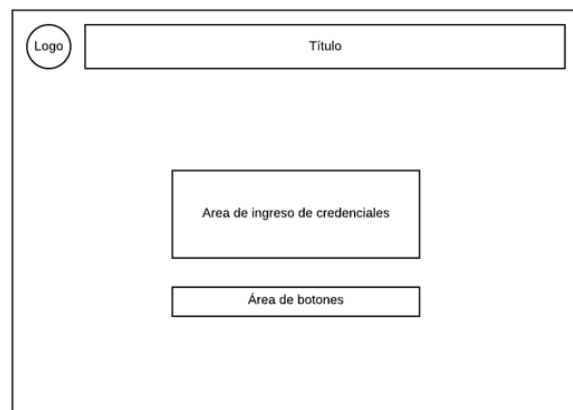
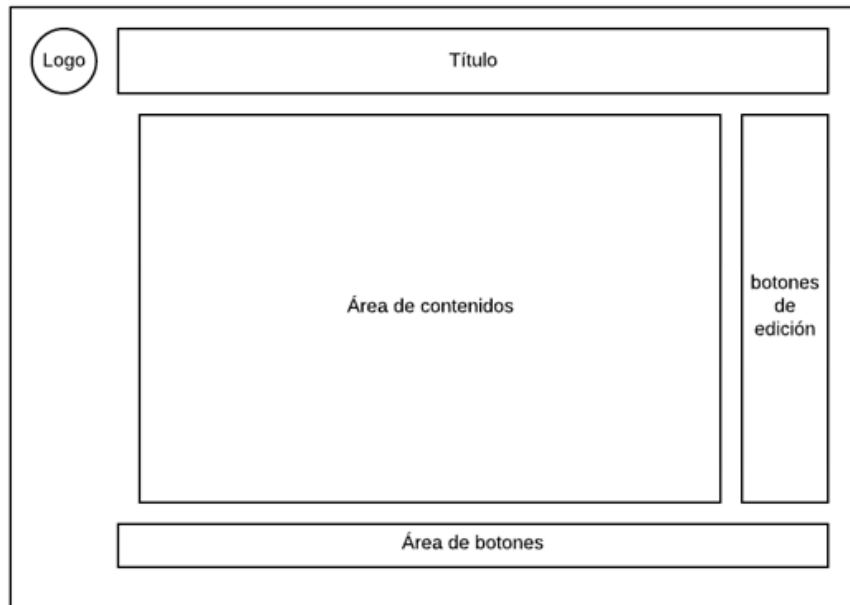


Figura 2.4 Ventana de inicio de sesión web
Autor: Andrés Dávila Bastidas

2.4.3 Ventana administrativa

En esta ventana se puede cargar nuevos servicios y recursos para ser mostrados en el chat.



*Figura 2.5 Administración web
Autor: Andrés Dávila Bastidas*

2.5 Estándares de programación utilizados

Aquí se describen los distintos estándares empleados en este trabajo:

2.5.1 Arquitectura

Como estándar de arquitectura se implementó el modelo MVC - Modelo Vista Controlador, este modelo fue designado, ya que es uno de los más maduros y a lo largo del tiempo ha demostrado su fiabilidad y validez, se implementó con el fin de mantener las distintas capas separadas,

2.5.2 Codificación

Como estándar en el código se utilizó el estilo de escritura Lower Camel Case, esto con el fin de mantener una buena práctica de desarrollo.

Ejemplo:

`componentePracticoIsrael`

Datos Generales:

- Los nombres no deben poseer tildes.

2.5.3 Base de datos

Para la base de datos se manejó las siguientes opciones:

Nombre de base:

base_<nombreAplicacion o nombreEmpresa>

Menor o igual a 15 caracteres.

Ejemplo: base_componente_practico

Tablas:

Menor a igual a 15 caracteres.

La primera letra se escribe en mayúsculas y el resto en minúsculas.

Campos:

Todas las tablas tienen una sola clave primaria.

Cuando las claves no pueden ser autonómicas (en caso de requerir replicación) se hará uso de claves de tipo carácter.

Datos generales:

- Los nombres de las tablas representando a su contenido.
- Los nombres de las tablas en singular.
- Sin ningún espacio en blanco, este espacio fue reemplazado con un “_”.

2.6 Pruebas

En esta sección se detallan las pruebas realizadas a la aplicación, por ser una aplicación que trabaja dentro de la plataforma de Facebook es muy confiable y su ejecución está garantizada siempre y cuando Facebook esté funcionando correctamente, para la comunicación entre Facebook, DialogFlow y nuestro servidor de aplicaciones es necesario contar con un certificado de seguridad (SSL), al tener costo y no estar este considerado dentro del proyecto se utilizó una herramienta que permite acceder a nuestro servidor local por medio de una url pública generada dinámicamente, el inconveniente con el uso de esta herramienta es que cuenta con un número limitado de peticiones, al sobrepasarlas este se bloquea durante unos minutos, a continuación se detalla el gráfico de peticiones permitidas a esta url:

La ejecución de Quickstarts permite máximo hasta 16 usuarios, esta limitación se produce debido a que ngrok se bloquea después de un cierto número de peticiones, debido a que es una herramienta de software libre y tiene limitantes en su funcionamiento.

Etiqueta	# Muestr...	Media	Min	Máx	Desv. Es...	% Error	Rendimiento	Kb/sec	Sent ...	Media de Bytes
https://github.com/wildfly/quickstart	16	4945	2081	7131	1312,70	0,00%	27,2/min	862,74	5,03	1016020,7
https://github.com/wildfly/quickstart/commit/4b57744576d6b043b53bde0cf85624ba74082/rollup?direction=e	16	460	384	487	23,79	0,00%	29,7/min	1,62	0,37	3357,4
https://github.com/wildfly/quickstart/file-list/master	16	575	384	1257	246,07	0,00%	29,7/min	92,51	0,34	191350,4
https://github.com/wildfly/quickstart/security/overall-count	16	104	97	155	13,31	0,00%	30,6/min	0,73	0,35	1472,8
https://github.com/wildfly/quickstart/tree/master/app-client?_pjax=%23js-repo-pjax-container	16	480	335	973	204,06	0,00%	30,4/min	26,14	0,58	52792,3
https://github.com/wildfly/quickstart/commit/80c0e9b009e0148a47969b326b77428a1be11ee/rollup?direction=e	16	143	98	396	98,02	0,00%	31,3/min	1,69	0,39	3309,4
https://github.com/wildfly/quickstart/file-list/master/app-client	16	138	99	390	81,82	0,00%	31,5/min	6,14	0,36	11999,4
https://github.com/wildfly/quickstart/tree/master/app-client/client-simple?_pjax=%23js-repo-pjax-container	16	257	182	413	77,57	0,00%	31,5/min	17,12	0,58	33429,4
https://github.com/wildfly/quickstart/tree/master/app-client/client-simple/src/main?_pjax=%23js-repo-pjax-container	16	517	333	916	211,72	0,00%	31,3/min	17,82	0,61	34959,1
https://github.com/wildfly/quickstart/commit/7806c1747c9292956b14d762393adce0b0e253/rollup?direction=e	16	295	99	803	210,93	0,00%	31,5/min	0,79	0,39	1545,9
https://github.com/wildfly/quickstart/file-list/master/app-client/client-simple/src/main	16	150	98	398	98,22	0,00%	31,6/min	3,69	0,38	7182,4
https://github.com/wildfly/quickstart/refs/master/app-client/client-simple/src/main/source_action=disambiguate&source...	16	196	101	502	134,01	0,00%	31,7/min	7,39	0,40	14329,4
https://github.com/wildfly/quickstart/refs/master?_pjax=%23js-repo-pjax-container	16	167	102	380	89,60	0,00%	31,8/min	10,46	0,40	20216,4
https://github.com/wildfly/quickstart/file-list/master?_pjax=%23js-repo-pjax-container	16	520	292	1223	336,73	0,00%	31,7/min	69,90	0,38	135428,4
Quickstarts	16	8894	6625	11092	1449,73	0,00%	23,4/min	887,16	8,50	2327393,6
Total	340	1185	97	11092	2425,11	0,00%	5,9/sec	1774,32	17,01	310319,2

Figura 2.6 Ejecución de Quickstarts
Autor: Andrés Dávila Bastidas

En la siguiente gráfica se muestra el tiempo de respuesta de cada uno de los accesos, como se puede observar cada uno de los procesos responden de manera oportuna, la ejecución que más tarda en responder corresponde al servidor WildFly y está comprendida en 6ms, esto se produce por la que la petición no es directa:

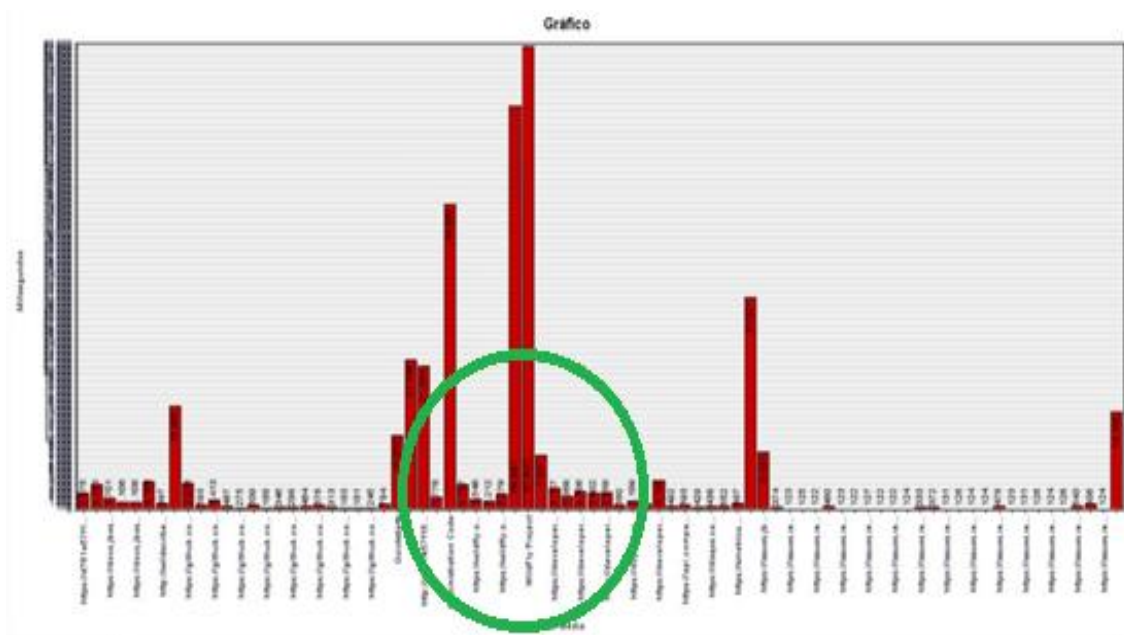


Figura 2.7 Tiempo de respuesta
Autor: Andrés Dávila Bastidas

En la siguiente gráfica se puede visualizar una ampliación de la etiqueta correspondiente al tiempo de respuesta en este caso representado a 6ms correspondiente a Wildfly.

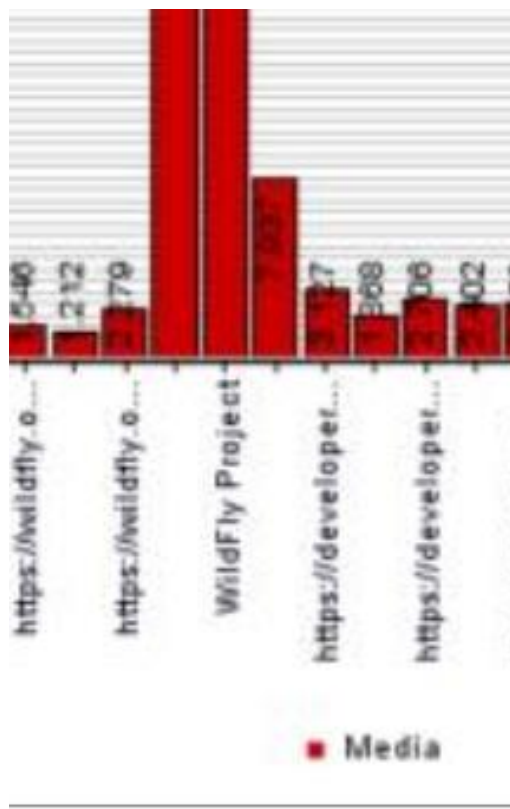


Figura 2.8 Tiempo de respuesta (Ampliación)
Autor: Andrés Dávila Bastidas

En ejecución de WildFly Project el máximo es de 90 usuarios, que es el dato más importante para el funcionamiento de la aplicación, ya que este servidor es el encargado de responder a cada una de las peticiones ingresadas desde el chat de Facebook y traducidas por DialogFlow a Intents:

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent Kb/sec	Media de Bytes
http://wildfly.org/	90	3310	2431	7460	870.63	0,00%	1,1/sec	1299,37	6,06	1241689,7
https://dpm.dende...	90	767	640	3866	411,99	0,00%	1,1/sec	2,14	0,53	1974,2
https://wildfly.org/n...	90	1941	1462	5029	663,95	0,00%	1,1/sec	37,04	7,53	35084,9
https://wildfly.org/n...	90	1896	1288	4109	537,55	0,00%	1,1/sec	24,06	7,56	22856,9
https://wildfly.org/d...	90	2804	1892	6215	694,58	0,00%	1,1/sec	126,23	18,34	122504,6
Wildfly Project	90	10720	8634	21906	1746,68	0,00%	54,0/min	1251,72	26,93	1424110,5
Total	540	3573	640	21906	9422,38	0,00%	5,4/sec	2503,44	93,86	474703,5

Figura 2.9 Ejecución de Wildfly Project
Autor: Andrés Dávila Bastidas

En la prueba de Report Issue implementada para un usuario el direccionamiento se produce con éxito y el usuario es redirigido a todos los servicios de manera correcta.

https://issues.jboss.org/browse/WFLY	1	7821	7821	7821	0,00	0,00%	7,7/min	1038,60	1,61	8317813,0
https://issues.redhat.com/rest/webResources/1.0/resources	1	370	370	370	0,00	100,00%	2,7/sec	1,33	5,50	504,0
https://issues.redhat.com/rest/menu/latest/appswitcher?_id=1596676...	1	125	125	125	0,00	0,00%	8,0/sec	6,21	3,68	795,0
https://issues.redhat.com/rest/scriptrunner/1.0/message?_id=159667...	1	124	124	124	0,00	0,00%	8,1/sec	3,58	3,38	455,0
https://issues.redhat.com/rest/orderbycomponent/latest/orderbyOpt...	1	126	126	126	0,00	100,00%	7,9/sec	3,91	4,50	504,0
https://issues.redhat.com/rest/secure/Action/default.jspa?issue...	1	460	460	460	0,00	0,00%	2,2/sec	99,79	1,18	47006,0
https://issues.redhat.com/rest/analyses/1.0/publish/bulk	1	125	125	125	0,00	100,00%	8,0/sec	3,94	10,62	504,0
https://issues.redhat.com/rest/scriptrunner/1.0/message?_id=159667...	1	124	124	124	0,00	0,00%	8,1/sec	3,58	3,38	455,0
https://issues.redhat.com/rest/api/latest/issue/WFLY-13633/remotel...	1	129	129	129	0,00	0,00%	7,8/sec	3,75	3,60	495,0
https://issues.redhat.com/rest/gliffy/1.0/diagrams/thumbnailList?iss...	1	125	125	125	0,00	100,00%	8,0/sec	5,67	3,55	726,0
https://issues.redhat.com/rest/projects/1.0/project/WFLY/latest/visited	1	123	123	123	0,00	100,00%	8,1/sec	5,76	4,44	726,0
https://issues.redhat.com/rest/scriptrunner/behaviours/latest/validat...	1	126	126	126	0,00	0,00%	7,9/sec	3,67	3,63	474,0
https://issues.redhat.com/rest/workflowDesigner/latest/workflows?n...	1	542	542	542	0,00	0,00%	1,8/sec	16,12	1,03	8945,0
https://issues.redhat.com/rest/workflowDesigner/latest/workflows?n...	1	848	848	848	0,00	0,00%	1,2/sec	10,30	0,66	8945,0
https://issues.redhat.com/rest/workflowDesigner/latest/workflows?n...	1	123	123	123	0,00	100,00%	8,1/sec	5,07	4,72	638,0
https://issues.redhat.com/rest/secure/Action/default.jspa?issue...	1	215	215	215	0,00	0,00%	4,7/sec	241,85	2,53	53246,0
https://issues.redhat.com/rest/scriptrunner/1.0/message?_id=159667...	1	125	125	125	0,00	0,00%	8,0/sec	3,55	3,35	455,0
https://issues.redhat.com/rest/api/latest/issue/WFLY-13632/remotel...	1	127	127	127	0,00	0,00%	7,9/sec	3,81	3,65	495,0
https://issues.redhat.com/rest/scriptrunner/behaviours/latest/validat...	1	127	127	127	0,00	0,00%	7,9/sec	3,64	3,60	474,0
https://issues.redhat.com/rest/gliffy/1.0/diagrams/thumbnailList?iss...	1	124	124	124	0,00	100,00%	8,1/sec	5,72	3,58	726,0
https://issues.redhat.com/rest/gliffy/1.0/diagrams/thumbnailList?iss...	1	122	122	122	0,00	100,00%	8,2/sec	5,81	3,64	726,0
https://issues.redhat.com/rest/secure/Action/default.jspa?issue...	1	240	240	240	0,00	0,00%	4,2/sec	191,58	2,27	47083,0
https://issues.redhat.com/rest/scriptrunner/1.0/message?_id=159667...	1	371	371	371	0,00	0,00%	2,7/sec	1,20	1,13	455,0
https://issues.redhat.com/rest/api/latest/issue/WFLY-12835/remotel...	1	133	133	133	0,00	0,00%	7,5/sec	3,63	3,49	495,0
https://issues.redhat.com/rest/scriptrunner/behaviours/latest/validat...	1	127	127	127	0,00	0,00%	7,9/sec	3,64	3,60	474,0
Report an issue	1	13002	13002	13002	0,00	100,00%	1,7/min	234,28	0,75	8493614,0
Total	26	1000	122	13002	2815,36	34,62%	44,1/min	468,57	1,50	653354,9

Figura 2.10 Ejecución de Report Issue
Autor: Andrés Dávila Bastidas

2.7 Implementación

Para la implementación se debe tomar en cuenta que todas las herramientas utilizadas en el presente proyecto son de libre acceso por lo que sus costos de implementación se ven significativamente reducidos.

2.8 Factibilidad técnica y económica

A continuación, se detallan los requerimientos necesarios tanto para la implementación técnica como para la parte económica.

2.8.1 Factibilidad técnica:

Debido a tratarse de un sistema relativamente pequeño, sus costos son bajos y la necesidad de equipos o servidores de grandes características es innecesaria, motivo por el cual el computador del estudiante servirá para alojar tanto la parte del servidor como la del interprete Dialog Flow.

En cuanto a conocimientos, se debe tener varios profesionales con amplios conocimientos, tanto en implementación de servidores (infraestructura) como en desarrollo de las distintas herramientas aplicadas (DialogFlow, Apis, Java, MySQL), por ser un proyecto individual estas vacantes son cubiertas solamente por una persona, así que no se consideran costos de capacitación, ya que se asume que el profesional conoce perfectamente las herramientas que se van a utilizar, los requerimientos se detallan a continuación:

Recursos Técnicos para el desarrollo del proyecto			
Nº	Tipo de recurso	Nombre del recurso	Descripción
1	Humano	Experto en el área	Diseño de base de datos
			Implementación de base de datos
			Conocimiento en Java
			Conocimiento de API's
			Conocimiento de DialogFlow
			Desarrollo web
2	Hardware	PC	Laptop
			Lenovo Legion Y720
			Intel Core I7 7th Gen.
			2.80GHz
			16GB
			Disco sólido de 256GB
3	Software	MySQL	V5.7
		Java SE Development Kit	V8
		DialogFlow	
		WildFly	V11
		Ngrok	

Tabla 2.2 Recursos Técnicos para el desarrollo del proyecto

2.8.2 Factibilidad económica:

A continuación, se describe los costos del recurso necesario para el desarrollo de nuestro Sistema de Información:

Recursos humanos:

La tabla de a continuación representa el costo del material humano, considerando la duración del proyecto 3 meses:

Recursos humanos			
Nº	Cargo	Costo mensual	Costo total (3 meses)
1	Ing. Sistemas (Líder de proyecto)	\$ 1,200.00	\$ 3,600.00
2	Analista/Diseñador	\$ 500.00	\$ 1,500.00
3	Ingeniero de software	\$ 700.00	\$ 2,100.00
4	Programador	\$ 700.00	\$ 2,100.00
		\$ 3,100.00	\$ 9,300.00

Tabla 2.3 Recursos humanos

Recursos tecnológicos:

La tabla de a continuación representa el costo de los recursos tecnológicos, al ser un prototipo se implementó en la propia computadora del estudiante, por lo que se considera un alquiler proporcional a las horas trabajadas, y al usar software libre para toda la implementación el costo de licencias y programas se lo considera con valor 0:

Recursos Tecnológicos			
	Hardware		
N°	Detalle	Costo/hora	Total
1	70 horas uso computador	\$ 0.80	\$ 256.00
		\$ 0.80	\$ 256.00

Tabla 2.4 Recursos tecnológicos (hardware)

Recursos Tecnológicos			
	Software		
N°	Detalle	Costo/hora	Total
1	DialogFlow	\$ -	\$ -
2	Java	\$ -	\$ -
3	Wildfly	\$ -	\$ -
		\$ -	\$ -

Tabla 2.5 Recursos tecnológicos (software)

2.8.3 Requerimientos Hardware y software

A continuación, se detallan los requerimientos para necesarios tanto para cliente como para el servidor.

Software para servidor

- MYSQL 5.7
- Servidor de aplicaciones WILDFLY versión 11
- Cuenta en Google activa (Dialog Flow)
- Cuenta en Facebook for developers
- Java EE v8
- Ngrok
- Fan Page de Facebook para interactuar directamente con nuestra aplicación.

Software para usuario

- Navegador Web.
- Cuenta en Facebook o Messenger.
- Acceso a Internet

Hardware para servidor

- Procesador Intel Core I5
- Tarjeta de Red
- Acceso a Internet de por lo menos 5Mbps
- Memoria RAM de al menos 6GB
- Al menos 50 GB de espacio libre en disco

Hardware para usuario

- Dispositivo con acceso a Internet

CONCLUSIONES

En el desarrollo del presente proyecto se han alcanzado los objetivos inicialmente planteados:

- La implementación del chat bot incidió positivamente en la automatización en el proceso de consulta de requerimientos de la Institución Educativa de nivel Superior, brindando una experiencia agradable, tecnológica y atrayente.
- La construcción del sistema propuesto acorde con las necesidades de la Institución Educativa de nivel Superior agilitó el proceso de implementación y personalizó las opciones de las distintas pantallas.
- Debido a la correcta esquematización de la arquitectura se produjo una implementación y comunicación correcta entre las distintas herramientas de esta aplicación.
- Con la implementación adecuada de las herramientas, se logró generar alertas para que los estudiantes puedan sincronizar sus horarios con la aplicación.

RECOMENDACIONES

Una vez concluido este proyecto se considera interesante tomar en cuenta las siguientes recomendaciones:

- Se debe definir claramente la metodología de trabajo que se va a usar para el desarrollo del proyecto de esta manera se logra tener una visión clara de las actividades necesarias para concluir el proyecto.
- Es importante brindar el tiempo necesario al análisis de la solución, de esta manera se logra considerar todos los aspectos que involucrará.
- Una práctica que se debe realizar es revisar la documentación de cada una de las herramientas, así como sus versiones actuales y requerimientos necesarios para su ejecución y compatibilidad.
- El presente aplicativo debe tener el mantenimiento y el seguimiento necesario debido a que muchos de sus componentes son de tecnología de punta y constantemente se ven sujetos a cambios.

REFERENCIAS BIBLIOGRÁFICAS

Sznajdleder, P. (2020). *El gran libro de Java a Fondo 4ª Ed*, Buenos Aires, Argentina, Alfaomega.

Red Hat, Inc. (2018). *Red Hat JBoss Enterprise Application Platform 7.0*. Estados Unidos de América.

Van Heylen, H. (2020). *Build a chatbot with DialogFlow, nodeJs and Slack*. Francia, Autoeditado.

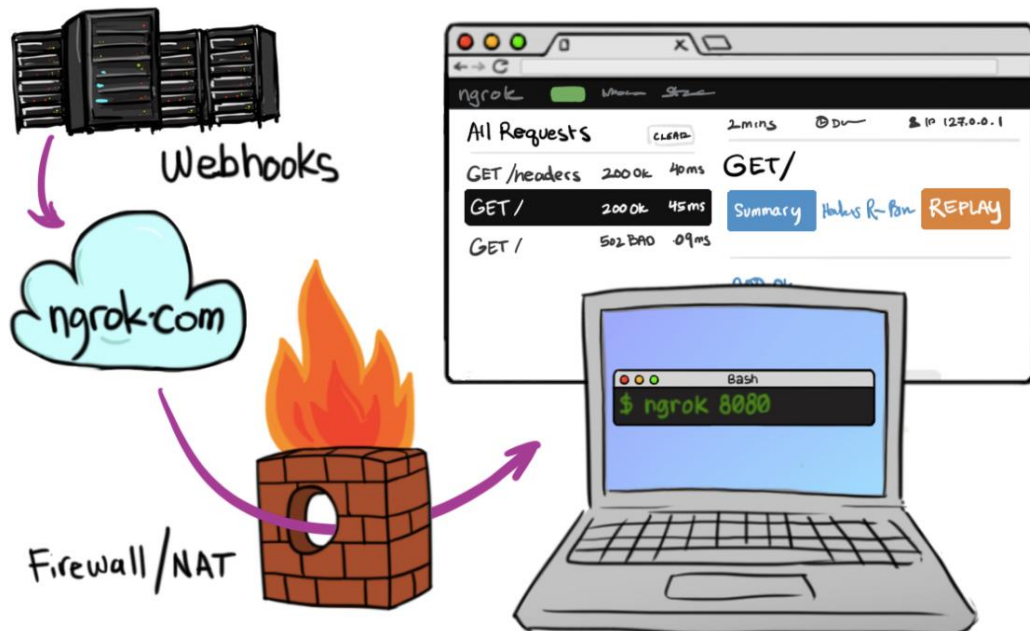
Google Cloud. (2020). Documentación de Dialogflow. Estados Unidos de América.

Facebook for developers. (2020). *API Graph V8.0*. Estados Unidos de América.

ANEXOS

Anexo 1 Funcionamiento de Ngrok

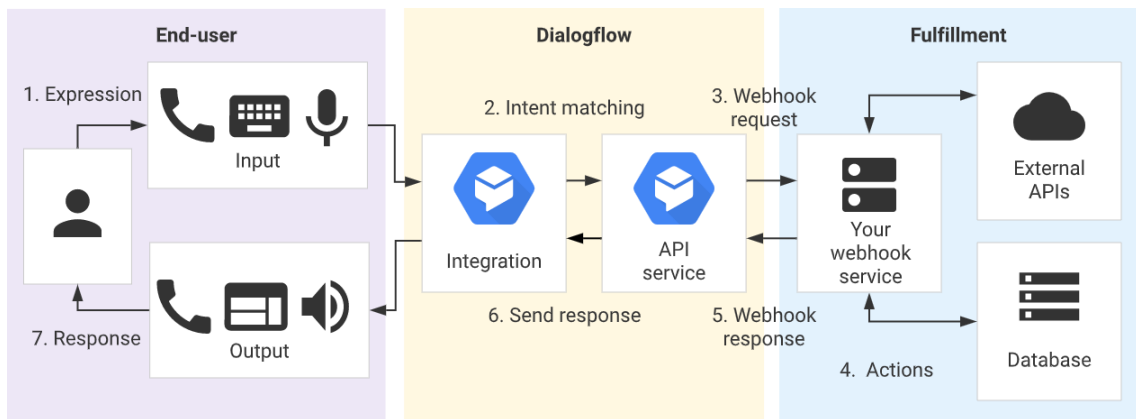
En el siguiente gráfico se muestra como ngrok funciona como intermediario entre las peticiones realizadas y simula un entorno seguro para que el firewall permita el paso de la comunicación.



Anexo 1 Diagrama de funcionamiento de ngrok
Autor: Snifer Lab's

Anexo 2 Funcionamiento de DialogFlow

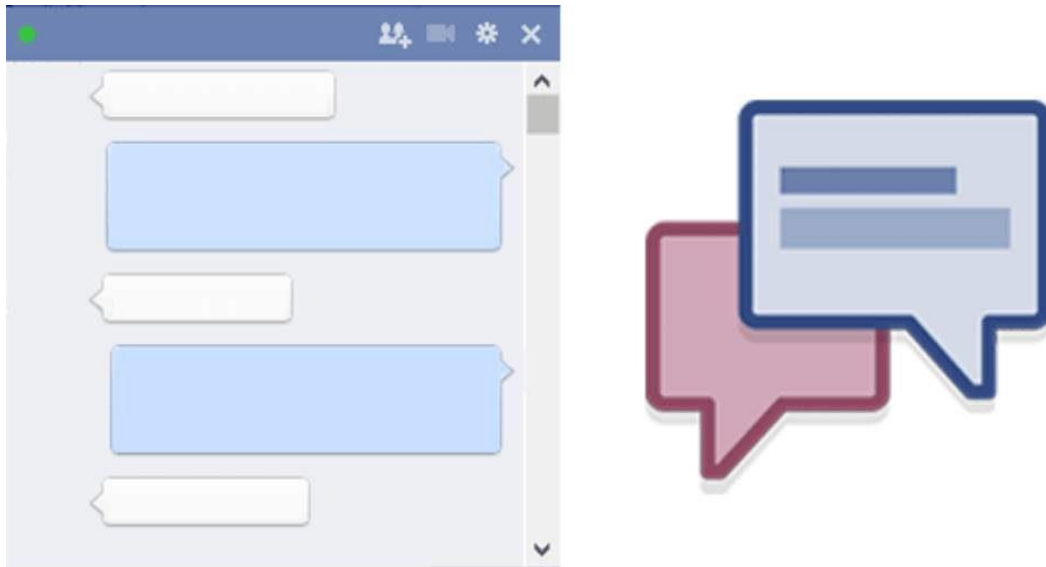
El siguiente gráfico muestra como se produce la comunicación desde el momento en el que el usuario realiza una petición la misma que es procesada, obtiene una respuesta y es devuelta al usuario.



Anexo 1 Diagrama de Dialog Flow
Autor: Google Cloud

Anexo 3 Ventana de chat de Facebook Messenger

Este gráfico representa la ventana del chat de Facebook, se puede observar las secciones para el envío y recepción de mensajes, además de los botones propios de acción de la ventana.



Anexo 3 Ventana de chat de Facebook Messenger
Autor: Facebook for Developers

Andres Dávila B

INFORME DE ORIGINALIDAD

7 %

INDICE DE SIMILITUD

7 %

FUENTES DE
INTERNET

0 %

PUBLICACIONES

%

TRABAJOS DEL
ESTUDIANTE

Revisado y aprobado para defensa
M.Sc. Ing. Pablo Recalde

FUENTES PRIMARIAS

1

repositorio.unamba.edu.pe

Fuente de Internet

2 %

2

thefide.blogspot.com

Fuente de Internet

1 %

3

repositorio.espe.edu.ec

Fuente de Internet

1 %

4

www.scribd.com

Fuente de Internet

<1 %

5

documents.mx

Fuente de Internet

<1 %

6

docs.microsoft.com

Fuente de Internet

<1 %

7

www.bce.fin.ec

Fuente de Internet

<1 %

8

informatica.uv.es

Fuente de Internet

<1 %

9

www.gepower.com

Fuente de Internet

<1 %