



UNIVERSIDAD NACIONAL AUTÓNOMA DE  
MÉXICO  
FES ACATLÁN

SISTEMAS INTELIGENTES

---

# Chatbot

---

David Felipe CASTILLO VELÁZQUEZ  
Kenny Yahir MÉNDEZ RAMÍREZ

12 de febrero de 2022

# Índice

<b>1. Requerimientos</b>	<b>2</b>
1.1. Introducción . . . . .	2
1.2. Información técnica y requerimientos . . . . .	2
<b>2. Casos de uso</b>	<b>3</b>
<b>3. Introducción</b>	<b>5</b>
<b>4. Análisis</b>	<b>5</b>
4.1. Diagrama Entidad-Relación . . . . .	6
4.2. Software . . . . .	7
<b>5. Diseño</b>	<b>7</b>
5.1. Estructura <b>Grafo</b> . . . . .	8
5.2. Búsqueda sobre el <b>Grafo</b> . . . . .	9

# 1. Requerimientos

## 1.1. Introducción

**Alcance** En este apartado se presentarán los requisitos o las razones por las cuáles se dió origen al proyecto.

Se presentarán los siguientes puntos:

- Definición del problema
- Requerimientos técnicos

**Propósito** Nuestros clientes de la Empresa necesitan el software adecuado para poder llevar una administración de los empleados, donde ellos puedan consultar información respecto a su cargo o acerca de los proyectos en los que se estén trabajando. Uno de los objetivos es que este software pueda ser accesado mediante lenguaje natural, con ésto nos referimos a que no se deba tener conocimiento previo de una serie de comandos o palabras específicas. En resumen el software buscado o adecuado para situación es una especie de “Chatter bot”.

## 1.2. Información técnica y requerimientos

**Definición del Sistema** Esta especie de Chatter bot al que denominamos con el nombre de “Chat bot”, será accesado mediante alguna pantalla y un teclado, y se entablará comunicación mediante una consola o terminal, por lo que solo se hará uso de texto en el idioma Español latinoamericano.

A continuación definimos las entradas para nuestro “Chat Bot”.

1. Saludos
2. Consultas referentes a la empresa

### Requerimientos técnicos del Sistema

- **Funcionamiento** El Chat bot deberá responder a cualquier pregunta reelevate a la empresa, por lo tanto casi cualquier forma en la cual se formule ésta, deberá ser procesada. Podrá contestar a las formas habituales de saludo o pláticas sencillas.
- **Manejo de errores** Estos pueden darse si la pregunta hecha no es específica o sale del contexto de trabajo. En estos casos se le pedirá al usuario que vuelva a formular la pregunta, entonces la persona deberá tratar de rehacer su pregunta utilizando otra forma similar, es decir, puede hacer uso de sinónimos y/o artículos, según el idioma usado.

## 2. Casos de uso

Presentaremos los diferentes casos de uso existentes:

<b>Caso de uso:</b> Autenticándose	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) El empleado saluda de forma cotidiana	
2) Vicky responde con un saludo y pregunta por el nombre del empleado	
3) El empleado ingresa su nombre ya sea completo o parcial	
4) Vicky corrobora que exista en la base de datos y pregunta por la contraseña	Si el usuario existe en la base continuará el proceso si no se preguntará nuevamente
5) El empleado ingresa su contraseña	
6) Si Vicky confirma que están correctos los datos le da acceso al empleado	Si la contraseña es incorrecta se le pedirá al usuario ingresarla nuevamente
7) Se repite desde el paso 3) hasta que pueda ser logueado el usuario	

<b>Caso de uso:</b> Consultando	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) Vicky pregunta al empleado que acción desea realizar	
2) El empleado pregunta por información acerca de un proyecto o de los demás empleados de la empresa	
3) Vicky recibe la petición, la procesa y verifica que se pueda devolver la respuesta deseada y responde al empleado	Si no entiende la solicitud se pide que formule de nuevo la pregunta
4) Se repite desde el paso 1) hasta que el empleado se despida	

<b>Caso de uso:</b> Preguntando por los diferentes roles	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) El empleado pregunta ya sea por los programadores, tester, clientes, etc.	
2) Vicky recibe la petición, la procesa y verifica que el puesto exista	Si no entiende la solicitud se pide que formule de nuevo la pregunta

<b>Caso de uso:</b> Preguntando por el puesto de una persona	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) El empleado pregunta que puesto ocupa una persona.	
2) Vicky recibe el nombre de la persona y verifica que exista en la base	Si no entiende la solicitud se pide que formule de nuevo la pregunta
3) Vicky da una respuesta	

<b>Caso de uso:</b> Solicitando tareas	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) El empleado pregunta por las tareas disponibles	
2) Vicky lista las tareas pendientes	
3) El empleado pide una tarea	
4) Vicky pregunta si ya terminó su tarea	
5) El empleado contesta si o no	
6) Si la respuesta es si, Vicky asigna la tarea	No asigna la tarea

<b>Caso de uso:</b> Preguntando por el jefe de un empleado	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) El empleado pregunta por el jefe de alguien	
2) Vicky procesa la solicitud validando que exista dicha persona	
3) Vicky responde	Si no existe le pide al usuario verificar el nombre

<b>Caso de uso:</b> Preguntando por los subordinados de alguien	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) El empleado pregunta por el subordinado inmediato de alguien	
2) Vicky procesa la solicitud validando que exista dicha persona	
3) Vicky lista los subordinados en el caso que sea más de uno	No hay subordinados al cargo de esa persona

<b>Caso de uso:</b> Preguntando por la clave del wifi	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) El empleado pregunta por la clave del wifi	
2) Vicky recupera la contraseña del wifi y se la muestra al empleado	

<b>Caso de uso:</b> Preguntando por el avance del proyecto	
<b>Actor:</b> Algún empleado	
Curso normal	Alternativas
1) El empleado pregunta por el avance del proyecto o el estado de las tareas	
2) Vicky muestra al usuario una tabla con las tareas y su estado	

### 3. Introducción

Como todos sabemos, en toda organización es necesario llevar un orden en las actividades realizadas a diario, como lo son la gestión de proyectos, o el seguimiento de qué empleados existen en ella. Algunas veces estas tareas se vuelven tediosas y se llegan a cometer errores de forma frecuente, perdiendo tiempo y dinero.

Por este tipo de situaciones se llegan a conclusiones como, tener un sistema de información que nos ayude en estas tareas. Pero no solo cualquier tipo de sistema de información, si no alguno que sea intuitivo, sin que el usuario que lo use tenga que saber muchos aspectos de él. Así no se invertirá tiempo en una capacitación muy compleja para poder usar dicho sistema.

### 4. Análisis

En esta sección describimos parte de la solución general tomada para el problema planteado.

De acuerdo a los requisitos solicitados se resumen los siguientes aspectos:

- Se requiere de una interfaz amigable, es decir, no tener que usar un lenguaje específico
- Conocer el avance del proyecto en curso
- Obtener la lista de empleados
- Obtener la lista de los jefes
- Asignar tickets (tareas faltantes)

- Preguntar por la clave del Internet
- Puestos o el rol de los empleados

Con lo anterior hemos diseñado un modelo que utilice la base de datos que tiene la empresa, de modo que va a interactuar con “Vicky”, pero así mismo Vicky va tener una base de datos, la cual servirá para reconocer las diferentes palabras utilizadas por el usuario. Lo ilustramos con la siguiente imagen:

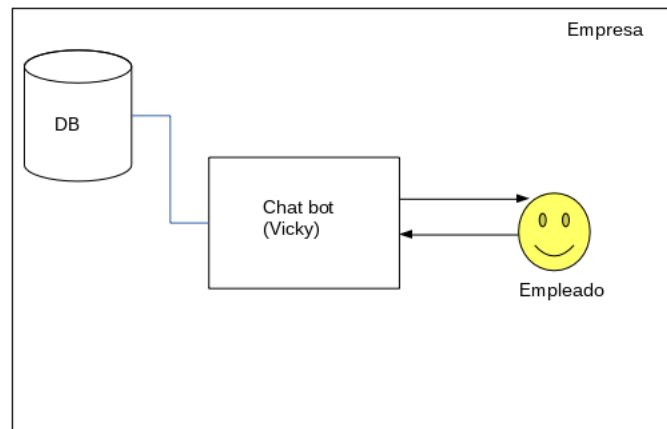


Figura 1: Usuario - Chat bot

#### 4.1. Diagrama Entidad-Relación

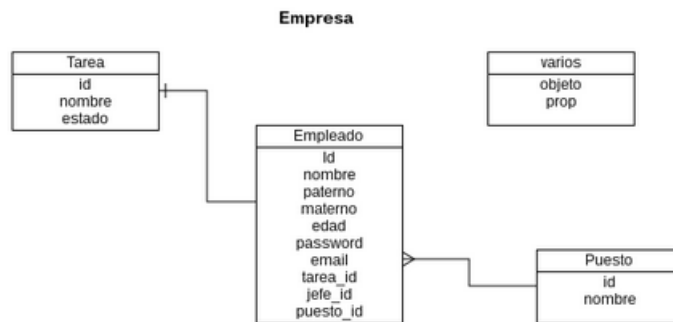


Figura 2: Diagrama Entidad - Relación de la empresa

Como podemos ver en la Figura 2 tenemos una relación uno a muchos entre la entidad Empleado y Puesto, ya que un puesto lo pueden ocupar varios empleados. Pero para las tareas cambia, ya que una tarea solo puede ser asignada

a un solo empleado. Ahora la entidad varios utilizada es para cubrir necesidades como el almacenamiento de claves como la del Wifi.

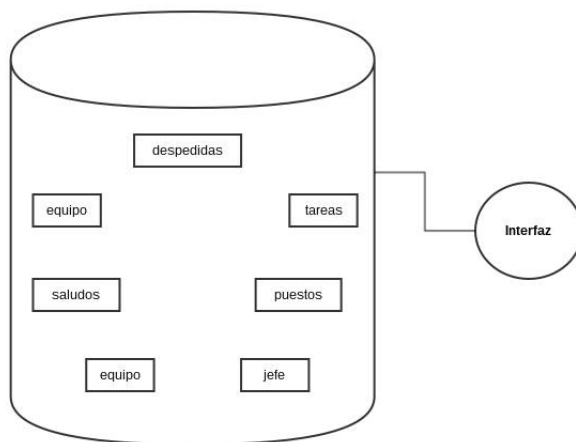


Figura 3: Interfaz y su base de datos

En la imagen de arriba apreciamos la estructura que tendrá nuestro Chat bot, que como vemos, usará una base de datos en la cuál almacenará las diferentes palabras que podrá reconocer, el funcionamiento de éste se detallará en la siguiente sección.

## 4.2. Software

Debido a que solo se requiere de una interfaz sin gráficos, podemos hacer una implementación usando lo siguiente:

- Lenguaje de Programación Java version 1.7.0\_79 OpenJDK Runtime Environment (IcedTea 2.5.5) (Arch Linux build 7.u79\_2.5.5-1-x86\_64)
- DBMS MySQL (10.0.18-MariaDB) Para la base de datos se creó un usuario llamado “admin” usando el password “guapa”.

## 5. Diseño

Vicky está compuesta por cuatro clases: **Vicky**, **ChatBotController**, **VickyHelper** y **DBVicky**. Su interacción se describe en la siguiente imagen.

A continuación describiremos la funcionalidad de cada una de las clases.

- **Vicky**: En esta clase se encuentra la función principal (**main**). Ésta se encarga de hacer llamadas a **VickyHelper** para poder reconocer la cadena que se ingresa y a **ChatBotController**, una vez que se reconoce la cadena, para obtener una respuesta.



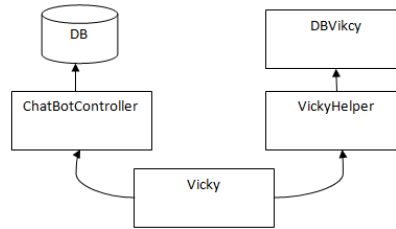


Figura 4: Diseño de las cuatro clases

- **ChatBotController**: Esta clase está en comunicación con la base de datos de la “empresa”. Se encarga de obtener la información de la base de datos y darle formato para que Vicky la muestre.
- **VickyHelper**: Aquí hacemos una búsqueda sobre la estructura **Grafo** definida en **DBVicky** para poder reconocer la cadena que se ingresa. Ahondaremos más adelante sobre cómo funciona esta clase y cómo está definida la estructura **Grafo**.
- **DBVicky**: Esta clase guarda la información que utiliza Vicky para reconocer las cadenas que se ingresan. Aquí se encuentran la estructura que define el grafo de búsqueda y un diccionario de sinónimos, este último para poder reconocer mayor rango de palabras.

### 5.1. Estructura Grafo

Para poder reconocer las cadenas que recibe Vicky decidimos definir un grafo con palabras como vértices y un arco (arista dirigida) de  $u$  a  $v$  si la palabra  $u$  debe ir antes de la palabra  $v$ . En la siguiente figura mostramos un ejemplo de lo anterior.

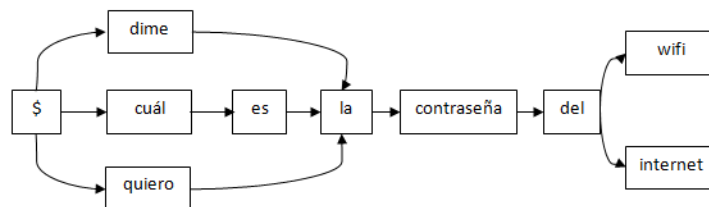


Figura 5: Ejemplo de grafo que reconoce si preguntan por la contraseña del wifi

El grafo se puede ver como un autómata de estados finito, en donde el estado inicial se encuentra en el vértice “\$” y los vértices terminales son aquellos que

no tienen arcos de salida. Cada terminal está asociado a una tarea que puede realizar Vicky.

Esta estructura nos permite, en ocasiones, reducir la memoria a la hora de definir las frases que puede reconocer Vicky, pues como vemos se dan los casos en que hay empalmes entre las frases. Por otro lado nos permite ir avanzando palabra por palabra y por lo tanto podemos obtener una distancia de edición entre la palabra ingresada y la palabra esperada, de tal forma que le podemos dar una holgura de error al usuario, por ejemplo, si esperamos la palabra “contraseña”, el usuario puede escribir “contrasña”, “contrasena” o “contraseñaa” y de todos modos se tomará como válida.

## 5.2. Búsqueda sobre el Grafo

Una vez que se ingresa una cadena, se realiza una búsqueda en profundidad sobre la estructura **Grafo** definida en **DBVicky**. La búsqueda se hace avanzando simultáneamente sobre la cadena recibida, de tal forma que si la siguiente posición en la cadena concuerda con alguno de los vecinos del vértice actual, entonces avanzamos sobre el grafo. Si en algún momento llegamos a un vértice terminal, se hace una llamada al método en **ChatBotController** que controla la tarea asociada con dicho vértice terminal.