

UNIVERSIDAD DE MONTEMORELOS  
FACULTA DE INGENIERÍA Y TECNOLOGÍA



JULIA: ASISTENTE DE NEGOCIOS BASADO  
EN LENGUAJE NATURAL

Investigación  
Presentada en cumplimiento parcial  
de los requisitos para el grado de Ingeniería en Tecnologías de la  
Información y comunicación.

por

Enmanuel Alejandro De Oleo

Mayo de 2014

## Dedicatoria

A mis padres y maestros que estuvieron apoyándome en el desarrollo de este proyecto.

## **Resumen**

Este proyecto presenta la documentación del desarrollo de un sistema computacional web llamada Julia, el cual opera como asistente de negocios, utilizando comandos de voz para la realización de sus funcionalidades elementales. En el proceso de desarrollo de Julia fue necesaria la integración de diferentes tecnologías web que harían posible el cumplimiento de los objetivos planteados. Julia es un sistema que funciona recibiendo y reproduciendo voz, utilizando el framework Ruby on Rails para funcionalidades de creación, lectura, eliminación y actualización de contenidos, y Speech API de google para poder reproducir voz y transcribir la voz del usuario a texto, para su posterior uso.

Julia actualmente es capaz de realizar solo tres funcionalidades, la primera es la creación de avisos generales, los cuales representan información pública para los miembros de los equipos de trabajo, la segunda es la creación de actividades personales, que hace referencia a la funcionalidad de un asistente personal, limitándose a las actividades individuales de cada usuario y por último las actividades grupales, las cuales presentan las tareas pendientes de los equipos de trabajo.

**Palabras clave:** Speech API, asistente de negocios, tecnologías web.

# TABLA DE CONTENIDOS

TABLA DE CONTENIDOS .....	iii
<b>CAPITULO I.....</b>	<b>1</b>
<b>INTRODUCCION.....</b>	<b>1</b>
<b>Problema.....</b>	<b>2</b>
<b>Justificación.....</b>	<b>3</b>
<b>Objetivos.....</b>	<b>3</b>
<b>Preguntas.....</b>	<b>4</b>
<b>Limitaciones.....</b>	<b>4</b>
<b>Delimitaciones.....</b>	<b>5</b>
<b>Definición de términos.....</b>	<b>6</b>
<b>Diseño de la Investigación.....</b>	<b>7</b>
<b>Antecedentes.....</b>	<b>8</b>
<b>CAPITULO II.....</b>	<b>17</b>
<b>APORTE.....</b>	<b>17</b>
<b>Google Speech continuos recognition.....</b>	<b>17</b>
<b>Ruby on Rails.....</b>	<b>20</b>
<b>Presupuesto.....</b>	<b>26</b>
<b>Resultados Obtenidos.....</b>	<b>27</b>
<b>CAPITULO III.....</b>	<b>29</b>
<b>CONCLUSIONES Y TRABAJOS FUTUROS.....</b>	<b>29</b>
<b>Conclusiones.....</b>	<b>29</b>
<b>Aportaciones Futuras.....</b>	<b>31</b>
<b>Apéndices.....</b>	<b>32</b>
<b>REFERENCIAS.....</b>	<b>34</b>

# CAPITULO I

## INTRODUCCION

Las nuevas tendencias tecnológicas nos indican que los servicios de VPA (Virtual Personal Assistant) son cada vez más importantes y necesarios, no solo para aplicaciones con fines tecnológicos sino también para fines cotidianos, prácticos y en especial en el mundo de los negocios (Elshafei).

Este tipo de software sirve como soporte para las organizaciones, especialmente en la automatización de tareas individuales y coordinación entre miembros de la organización. Es necesario que estos sistemas cuenten con posibilidades de calendarización de actividades, compartir información, monitoreo y seguimiento de tiempos de trabajo, además de filtros de información, entre otros (Okamoto, Sycara, & Scerri, 2009).

El principal objetivo de un software asistente como lo es Julia, debe ser la efectividad, eficiencia y facilidad de uso al realizar funcionalidades elementales del sistema que siempre deben estar enfocadas en la automatización de tareas generalmente de orden comunicativo. De esta manera mejorando las comunicaciones remotas entre los miembros de los equipos de trabajo y permitiéndoles avanzar más satisfactoriamente en sus actividades laborales. Para lograr esto, el principal punto de atención de un sistema de este tipo debe ser economizar y optimizar el tiempo, ya que es un recurso sumamente importante en cualquier negocio y brindar oportunidades de comunicación en trabajos colaborativos (Blair, 1992).

## **Problema**

Solo en el ámbito tecnológico 203 de las 500 empresas más importantes de Estados Unidos, están ubicándose en mercados emergentes para el desarrollo de sus productos, debido a los ahorros financieros que otros países ofrecen. Sin embargo existen muchos desafíos para los equipos de trabajo que están físicamente distanciados, todos esos desafíos hacen referencia a la falta de comunicación (Sengupta, Chandra, & Sinha, 2006), lo cual impide que se pueda avanzar más rápidamente en el desarrollo del producto. Y es que en los proyectos de ubicación compartida, la comunicación permite la fácil difusión del conocimiento del proyecto en desarrollo sea o no, tecnológico.

En la actualidad existen diferentes tipos de asistentes virtuales, aunque si es cierto que cada uno de ellos tiene sus particularidades y probablemente un enfoque que los diferencie, también es cierto que todos estos tienen un factor común, este factor es el aislamiento.

Ofrecer servicios de reconocimiento de voz, grabación y almacenamiento de mensajes, intercambio de información en respuesta a las peticiones del usuario, entre otras, es generalmente su principal objeto (Cooper et al., 2004).

En el mundo de los asistentes inteligentes, es notable la ausencia de softwares que sean desarrollados con la finalidad de satisfacer las necesidades operativas y comunicativas de equipos y grupos colaborativos, formados por miembros remotos. Ahora bien: ¿Será un asistente de negocios la solución a la falta de comunicación entre los miembros, en el desarrollo de proyectos colaborativos?

## **Justificación**

Con el crecimiento y expansión de las grandes empresas y organizaciones, las visiones de estas han cambiado, dando oportunidades a ideas más amplias y flexibles de trabajo en equipo. Entre estas, el trabajo remoto, que generalmente se refiere a un individuo, que es miembro de la empresa u organización y que aunque se encuentra físicamente distante, desempeña un papel crítico en el desarrollo de los proyectos colaborativos.

El desarrollo de un sistema que funcione como asistente y plataforma base de comunicación entre la empresa y estos empleados remotos, evita retrasos causados por la falta de comunicación y conocimiento de las actividades a realizar.

## **Objetivos**

En este proyecto de investigación tiene como meta principal el desarrollo de un sistema web que sea capaz de asistir virtualmente a todos los miembros de equipos de trabajo, evitando la falta de conocimiento de estos, como resultado de una falta de comunicación.

Con la integración de la Speech API de google sobre el framework de desarrollo web Ruby on Rails se espera lograr los siguientes objetivos:

1. Recuperar información del usuario sobre cualquier actividad que desee calendarizar comandos de voz.
2. Implementar tecnologías web que permitan la transcripción de voz a texto de la información emitida por el usuario.
3. Desarrollar un sistema capaz de responder de forma audible a las acciones realizadas por el usuario.

4. Integrar conceptos y funcionalidades de un asistente personal y un asistente de negocios para trabajos colaborativos en un mismo sistema.

### **Preguntas**

1. ¿Puede un asistente de negocios funcionar adecuadamente como puente de comunicación en proyectos colaborativos?

2. ¿Cuán útil puede ser un sistema capaz de implementar funcionalidades colaborativas y personales en una misma plataforma?

3. ¿Puede el lenguaje natural mejorar la experiencia del usuario al interactuar con un sistema asistente de negocios en el trabajo colaborativo?

### **Limitaciones**

Julia, debe cumplir con las funcionalidades básicas de un asistente, como lo son el manejo de las gestiones de actividades grupales y personales, además de avisos. Uno de sus principales desafíos se encuentra en la implementación de lenguaje audible. Este tiene un gran número de limitaciones que deben ser consideradas hasta cierto punto como irremediables. Es de suma importancia comprender que las computadoras deben aprender y comprender nuestro lenguaje y no viceversa. La principal dificultad en el proceso de comprensión del usuario en su lenguaje natural es la inmensidad de la gramática. Lo cual por asunto de tiempo y complejidad debe limitarse a lo más elemental posible. A continuación mostraremos las principales limitaciones que se presentan al adentrarse en esta área de ingeniería:

- De acuerdo con Eva M. Rodríguez (1999), la polisémica es la posibilidad de que una palabra en una oración, tenga diversos significados, diversos valores. El carácter polisémico de un



lenguaje, tiende a incrementar la riqueza de su componente semántico, más aun, este hecho no hace la formalización difícil, sino imposible.

1. Imposibilidad de hacer un diccionario o corpus con el lenguaje completo.
2. Distintas formas de expresar lo mismo, como menciona Alejandro Peña Ayala (2006).
3. Complejos algoritmos y técnicas para la validación de combinaciones correctas con respecto a sintaxis y semántica.

### **Delimitaciones**

Para poder realizar este proyecto y cumplir fielmente con los objetivos propuestos, es necesario conocer cuál es el alcance del mismo. Entendiendo la complejidad del lenguaje natural se ha determinado que es conveniente tener las siguientes delimitaciones:

1. El sistema solo tendrá registrado un equipo de trabajo, lo que significa que todos los usuarios que se registren en el sistema, serán incluidos de forma automática en este equipo que lleva el nombre de Julia Demo Testing.
2. Julia será capaz de dar la bienvenida a los usuarios, cada vez que estos ingresen al sistema, además será capaz de recordar al usuario las actividades y/o avisos pendientes.
3. El usuario puede mencionar cualquier palabra existente en el diccionario castellano, sin embargo habrán palabras ya definidas que permitirán al sistema identificar cuál de las tres operaciones ofrecidas por Julia, quiere realizar. Es importante recordar que este sistema solo creará nuevos avisos, actividades personales y grupales, lo cual significa, que siempre que el usuario quiera realizar alguna de estas tres operaciones, debe hacerlo saber al sistema.

## Definición de términos

En el desarrollo de aplicaciones web que integran técnicas de procesamiento de lenguaje natural, debemos explicar algunos términos que nos ayudarán a comprender mejor los puntos a presentar posteriormente:

1. *Tecnologías Web (Cloud computing)*: Hace referencia a interfaz orientada a servicios de internet y explotación de técnicas de virtualización en la web (Talia, 2011).

2. *Software asistente*: Es un software que tiene ciertos atributos de inteligencia, autonomía, adaptabilidad, precepción o que puede actuar en sustitución del usuario. Según (Nair, 2013), la inteligencia de un asistente o agente virtual se refiere a la capacidad de realizar tareas o acciones mediante información relevante recopilada como parte de técnicas de resolución de problemas.

3. *Lenguaje Natural*: Es un término que muestra la forma de comunicación que el humano aprende de su ambiente y usa para comunicarse con los demás, para expresar sus ideas, emociones y necesidades, esperando cierta clase de respuesta por parte de su interlocutor

4. *Google Speech API*: Es una librería de JavaScript que permite el reconocimiento de voz, transcripción de voz a texto y de texto a voz, que funciona solamente en la versión 25 de google Chrome y versiones sucesoras.

5. *Proyectos Colaborativos*: convocan a los participantes y miembros del equipo a sumar esfuerzos, competencias y habilidades, mediante una serie de trabajos en colaboración e interacciones que les permiten alcanzar juntos un propósito en común. Los proyectos colaborativos constan de tres elementos decisivos: participación, interacción y síntesis (Ingram & Hathorn, 2004).

6. *Framework*: Es con frecuencia una estructura en capas que indica de qué forma deben ser construido y relacionado un sistema computacional o software. En muchas ocasiones estas estructuras son más bien un conjunto de funcionalidades de un sistema que se relacionan entre sí.

7. *Ingeniería Lingüística*: Se enfoca en el tratamiento computacional del lenguaje natural y como aplicarlo para dar soluciones a problemas de ingeniería.

### **Diseño de la Investigación**

Esta investigación consiste en el desarrollo de un proyecto tecnológico que hace hincapié en la integración de tecnologías web para el desarrollo de un sistema que permita a múltiples usuarios acceder y utilizar un software asistente que les permita compartir información y trabajar en sus proyectos colaborativos, utilizando comandos de voz para interactuar con Julia.

## **Antecedentes**

Alexander Gelbukh (2010), afirma que las computadoras fueron creadas para solucionar nuestros problemas y no para crearnos más problemas. La capacidad de las computadoras para interactuar con el usuario utilizando el lenguaje natural es de gran importancia para el presente y futuro de los medios computacionales ya que se considera que ésta debe ser nuestra ayudante natural, útil, eficiente y fácil de usar.

Para cumplir los requerimientos mencionados es necesario que las computadoras puedan tener un medio de comunicación más inteligente con el usuario, esto debido a que el hombre siempre ha asociado la inteligencia con el habla. La manera en como los humanos determinan la inteligencia de un objeto, animal o cosa, es por su capacidad parlante, capacidad de hablar pero también de entender lo que les dicen.

Las lenguas son sistemas más o menos complejos, que asocian contenidos de pensamiento y significación a manifestarse simbólicamente tanto orales como escritas. Es en sí misma un conjunto de palabras. El estudio de las lenguas con fines de resolución de problemas, representan uno de los desafíos más significativos en la ingeniería lingüística la cual, de acuerdo a Julio Villena R. (2011), nació aproximadamente en las décadas de los 40 y 50 donde su campo de interés principal se centraba en:

1. Traducción automática
2. Recuperación de la información.

Es evidente que en estas épocas las limitaciones en la implementación de los modelos morfológicos y sintácticos eran considerablemente mayores a las actuales, además del poco interés en la comprensión exhaustiva del significado de los datos introducidos en lenguaje natural. El principal enfoque en estas fechas era:

1. Procesamiento de frases específicas.
2. Desarrollo de interfaces amigables y comprensibles para el usuario.

Entre alguno de los proyectos y sistemas desarrollados durante la década de los 60 y 70 se encuentran los siguientes:

### **Acceso a base de datos**

- **BASEBALL:** Un conocido sistema de estas épocas, categorizado como programa de preguntas y respuestas.

### **Resolución de problemas**

- **ELIZA:** Es un chatbot creado por Joseph W. en 1966 que representa el papel de un psicoanalista e imita sus respuestas típicas en una entrevista inicial.

No fue hasta la década de los 80 cuando comenzaron considerarse algunos puntos importantes en el procesamiento de lenguaje natural que de forma general eran desafiantes e inalcanzables en aquella época, según (Nadkarni, Ohno-Machado, & Chapman, 2011) estas complicaciones surgieron como resultado de tratar de obtener el significado semántico de los enunciados, intentando evaluar cada unidad de texto y categorizándola como pronombre, verbos y/o adjetivos. Fue hasta entonces cuando comenzaron a desarrollar sistemas para analizar y evaluar más profundamente los textos y oraciones en cuestión.

Es importante comprender que el lenguaje natural en estas épocas no se extendía al reconocimiento fonológico del usuario, de forma que el lenguaje natural se limitaba a interacciones textuales con el sistema. Y no es hasta unos años atrás en lo que se empieza a estudiar la voz como herramienta de introducción de datos para los sistemas.

Bellesi Facundo y Flavio Ortiz (2008), afirman que la señal de voz es una onda de presión acústica que se genera voluntariamente a partir de movimientos de la estructura anatómica del sistema fonador humano. Una aplicación virtual de asistencia debe ser capaz de recibir instrucciones expresadas en voz humana de múltiples usuarios remotos y el sistema debe responder según las expectativas del usuario (Cooper et al., 2004).

La primera operación que debe realizar un reconocedor es procesar la señal de voz de entrada al sistema, con objeto de extraer la información acústica relevante para la tarea que debemos realizar.

La señal de entrada puede venir acompañada por efectos perturbadores, los cuales deben ser eliminados, por este motivo se han generado tres líneas principales de trabajo:

1. Detección robusta de voz
2. Reducción de ruido
3. Cancelación de Ecos

El reconocimiento de voz por computadora es una tarea compleja, en la cual intervienen el reconocimiento de patrones y los sistemas biométricos. Por lo regular, la señal de voz se muestra en un rango entre los 8 y 16 KHz.

El micrófono es un transductor electroacústico. Su función es la de transformar la presión acústica ejercida sobre su capsula por las ondas sonoras en energía eléctrica. El audio es un fenómeno analógico.

Considerando la complejidad del estudio de la voz, los desarrolladores realizan un estudio acústico (la formación y propagación de sonidos) y fisiológico (cómo funcionan los órganos para producir sonido) de las expresiones pronunciadas, representadas como fonemas. Cada fonema representa un sonido, pero no siempre está ilustrado por una sola letra o grafía. El español posee fonemas que se identifican como vocales: a/e/i/o/u, y consonantes: /d/f/l/m...

Una vez se reconocen de forma fonológica los datos introducidos de forma audible por el usuario es necesario aplicar las siguientes técnicas:

1. Conocimiento de fonemas
2. Algoritmo de reconocimiento

El reconocimiento de fonemas permite pre-definir las palabras que están siendo pronunciadas por el usuario. Con frecuencia el análisis de los fonemas requiere conocimientos de los niveles superiores de análisis como el morfológico y sintáctico.

Una vez se define un algoritmo de reconocimiento entran los diccionarios, los cuales son una lista de palabras de una lengua, junto con diversas informaciones: morfología, definición, según Lourdes Araujo (2010). Los diccionarios son un recurso crítico en la mayoría de las tareas procesamiento de Lenguaje Natural. El diccionario representa el acervo de morfemas y derivados oficialmente aceptados para una lengua por lo que un SLN incluye para cada "termino" su significado, información para identificar su información estructural, referencia a la raíz o a sus derivados.

En el mismo contexto de la lingüística se introducen algunas áreas de estudio muy importantes como la morfología, sintaxis y semántica.

1. **Morfología:** se refiere a la revisión de la correcta estructura de la instrucción. Para ello estudia la formación de las palabras a partir de las unidades más básicas de significado denominadas morfemas, esto de acuerdo con José Manuel Soto Corzo (2001).

2. **Sintaxis:** estudia la forma en que se combinan las palabras para formar sintagmas y oraciones correctas, determinando el papel estructural de cada palabra y sintagma. Los sintagmas pueden estar constituidos por sujeto, predicado, objeto directo-indirecto, etc. La propuesta para este análisis es el uso de un diccionario de palabras válidas para el dominio del sistema. Es la etapa en el proceso de lenguaje natural en donde una oración lineal de texto alimentado por el usuario es convertida a una estructura jerárquica que corresponde a las unidades de significado de la oración de acuerdo a la naturaleza, componte y reglas de interrelación propias del lenguaje. Este análisis es especialmente útil para la recuperación de información por parte del sistema.

3. **Semántica:** su objetivo es alcanzar el conocimiento temático de los textos, el significado, por tanto de sus oraciones. La semántica contiene algunos elementos que son de suma importancia para el procesamiento de los textos:

1. Contexto: Es el ámbito social en donde se circunscribe el lenguaje que se emplea (raza, país, religión,...) y por lo tanto el significado particular que se asocia a sus elementos.
2. Significad: Es el concepto, idea, in, razón o conocimiento que está detrás del símbolo o expresión oral, que le proporciona personalidad específica dentro del contexto que usa, otorgándole vida y razón de existir.



3. Relación: De igual forma que los elementos del alfabeto al permanecer separados entre sí no tendrían utilidad alguna, tampoco lo sería para las palabras al estar aisladas unas de otras. Por tal motivo, es conveniente concebir al lenguaje como un sistema con elementos específicos e interrelacionados con el propósito de compartir conceptos.

Desde hace muchos años atrás se han hecho muchos esfuerzos en mejorar la experiencia de los usuarios mientras navegan en internet o simplemente están dando uso de un sistema definido. Es por este motivo que el 14 de junio de 2011 sale a la luz los primeros pasos para la búsqueda de información, utilizando la voz el cual sería la primera iniciativa en llevar la total comprensión del lenguaje natural a la web, esta implementada por Google (Adorf, 2013). Con un equipo de aproximadamente 32 desarrolladores, google pudo elaborar una API específicamente para el Web Speech, que permitiera a usuarios y desarrolladores de tercer nivel de Chrome poder explorar estas nuevas alternativas.

En su lanzamiento y hasta la actualidad solo Google Chrome, en sus versiones 25 o en adelante, soporta el Web Speech API. Y aunque las primeras intenciones eran solo para realizar búsquedas de información utilizando la voz, actualmente puede ser utilizado para el desarrollo de aplicaciones web como en el presente proyecto de investigación. La API de Web Speech es en sí mismo muchos bloques de códigos de JavaScript que permiten activar y desactivar el micrófono de la computadora en el cual se está ejecutando. Una vez que el usuario comienza a hablar, por medio de JavaScript la computadora de este, se comunica con un servidor de google.

La Web Speech API es capaz de convertir tanto voz a texto, como de texto a voz, en ambos casos esta API tiene a la disponibilidad del usuario más de 15 lenguajes diferentes para poder

realizar una transcripción confiable u efectiva, y para poder generar voz utilizando el lenguaje escogido con anterioridad.

El centro de atención de Web Speech API es lo que se denomina TTS que se refiere a *Texto To Speech*, el cual promete simplificar y mejorar la precisión de un número de tareas cotidianas en el entorno laboral. Sin embargo, hasta que los sistemas TTS se hagan más intuitivos y fáciles de usar, no llegaran a su máxima capacidad. Actualmente los servicios de TTS no están diseñados para funcionar sobre cualquier aplicación y lo ideal sería que la voz emitida suene suave y continua, sin pausa ni inflexiones, intentando simular el habla natural de los seres humanos. Sin embargo hasta el momento no se han alcanzado estos objetivos ya que la voz suena más como una concatenación de palabras y el entrecortado es notable (Holm & Pearson, 1998).

Como ya se ha presentado la Speech API de google solo se encarga de convertir la voz a texto y viceversa. Sin embargo esta API no realiza ninguna funcionalidad con el texto obtenido, y es por este motivo que los desarrolladores se han dado la tarea de a través de frameworks y otras tecnologías, comenzar a evaluar los textos introducidos y realizar operaciones con ellos.

Ruby on Rails (RoR) es un framework que trae consigo un conjunto de librerías, automatismos y convenciones destinados a resolver los problemas más comunes a la hora de desarrollar una aplicación web, para que el programador pueda centrarse solamente el aspectos únicos y diferenciales de su proyecto en lugar de en los problemas recurrentes.

Fue desarrollado por David Heinemeier Meier Hasson en Julio de 2004 como open source (Bächle & Kirchberg, 2007). Su principal objetivo es brindar facilidad de uso, mejorar la productividad, fomentar la reutilización de código fuente, brindar un ambiente de desarrollo flexible y extenso, permitiendo al usuario realizar pruebas fácilmente.

Ruby on Rails tiene una arquitectura muy característica de funcionamiento. Al pasar el tiempo muchos desarrolladores han adoptado la estructura MVC, la cual significa modelo – vista – controlador. Se ha determinado que esta estructura permite a las aplicaciones ser codificadas más organizada y limpiamente.

Aunque Ruby on Rails es un framework relativamente joven, la estructura (MVC) que utiliza, fue ideado por Trygve Reenskaug en el año 1979, con el fin de ayudar en el desarrollo de aplicaciones. En su diseño propuesto Trygve divide el desarrollo en los tres componentes ya presentados, modelo-vista-controlador.

Donde el modelo es más que simples datos, también se encarga de ingresar y consultar la base datos en busca de información relevante a las peticiones que el controlador está ejecutando. El modelo se encarga de aplicar todas las reglas de acceso y utilización de la información establecidos en la base de datos. Es en el modelo donde se realizan las relaciones entre una tabla de información con otra y generación de extracción de información la base de datos.

La vista es responsable de la generación de una interface visual para el usuario, que normalmente presenta los datos extraídos del modelo. Es importante recordar que la vista no puede acceder ni introducir datos en el sistema por sí sola, siempre que necesite acceder a información de la base de datos, enviara una petición en la cual especificará que información quiere recibir. De igual forma si el usuario introduce datos, la vida no puede validarlos y mucho menos autenticarlos, es en este momento que el controlador toma el escenario. Los controladores toman la información en los inputs o campos de información y/o cualquier otro evento del navegador y hace la interacción con el modelo, como se muestra en la Figura 1.

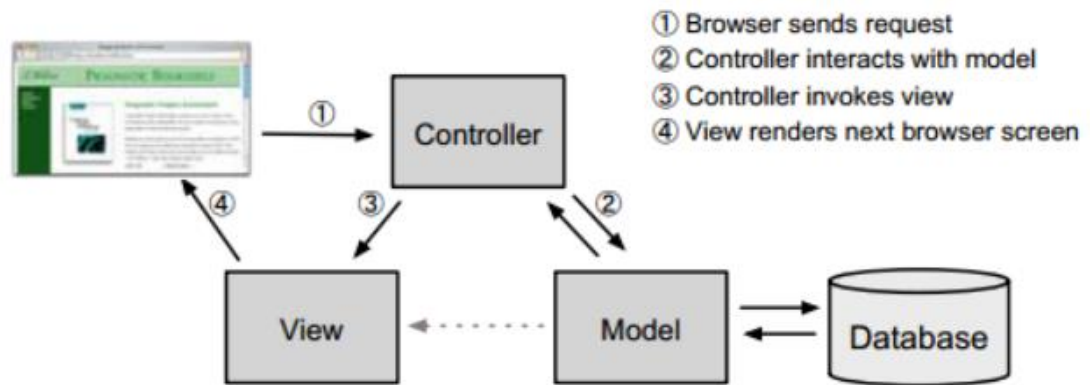


Figura 1. Estructura MVC de Ruby on Rails (Guide, 2006)

Otro aspecto importante de Ruby on Rails es un framework que corre sobre el lenguaje de programación Ruby, y este es orientado a objetos y es considerado de muy alto nivel por su similitud de términos respecto al idioma inglés, lo cual permite que sea exageradamente descriptivo y legible (Guide, 2006). Una de las filosofías más comunes de Ruby es DRY que significa Don't Repeat Yourself, es una forma de evitar la duplicación innecesaria de códigos, insistiendo en que cada bloque de código debe ser expresado solo una vez en un solo lugar.

Determinar el estado de los eventos y encuentros de negocios, guardar informaciones relevantes a estos, constituyen uno de los principios básicos de los asistentes personales, en especial en el mundo de los negocios, en especial si se pueden integrar la emisión de comandos por voz (Cooper, McElroy, Rolandi, Sanders, & Ulmer, 2002).

## **CAPITULO II**

### **APORTE**

La evolución de las tecnologías web en los últimos 5 años, ha sido enorme debido al aumento del tráfico en la red, provocado por la constante necesidad de información de los usuarios. Como resultado de este aumento, los desarrolladores web se han esforzado en el desarrollo de módulos tecnológicos que provoquen más satisfacción al usuario mientras navega por la red. Y la compañía tecnología más grande del mundo, es decir Google, ha desarrollado softwares y complementos para su navegador Chrome que permitan a terceros desarrolladores usar su creatividad en el desarrollo de aplicaciones web.

#### **Google Speech continuos recognition**

Es importante conocer la diferencia que existe entre Speech recognition y Speech continuos recognition, aunque ambas son parte de la misma API, su principal diferencia no está en su configuración. Google Speech recognition fue creado para realizar búsquedas en un campo de texto, de forma que se desactiva más rápidamente, ya que no está esperando recibir textos largos. Mientras que Speech continuos recognition fue desarrollada para que el usuario pudiera dictar párrafos, interactuar libremente con algún sistema, etc. En términos simples se pudiera afirmar que Speech recognition fue desarrollada para textos cortos mientras que la Speech continuos recognition para textos largos.

El reconocimiento de las operaciones que el usuario quiere realizar, es la funcionalidad más importante de Julia, es por este motivo que se ha implementado la versión Speech continuous recognition.

De forma predeterminada el micrófono, no está activo, lo que significa que el usuario tendrá que avisar al programa, cuando quiera emitir un comando utilizando su voz. Para lograrlo debe hacer clic en la imagen de amplificador (Figura 2). Para hacer posible que el navegador escuche al usuario, activar la Speech incorporado de Chrome.

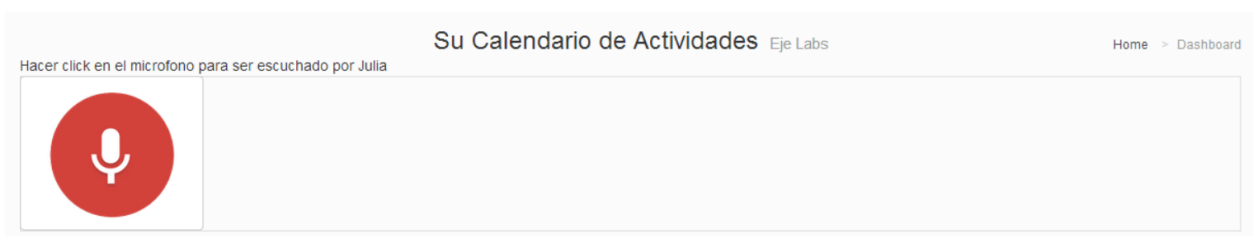


Figura 2. Interfaz de reconocimiento de voz, para transcripción a texto.

El proceso de traducción de voz a texto puede dividirse en otras tres partes importantes:

1. Implementación de JavaScript

- a. Como ya se conoce JavaScript es un lenguaje de programación que se ejecuta generalmente del lado del cliente en una conexión cliente – servidor, y en este caso se encarga de hacer una solicitud al Speech de Chrome para que este se conecte con el micrófono de la computadora en la cual se está ejecutando.

## 2. Permisos para utilización de hardware

- a. La implementación de JavaScript permite que google Chrome pueda pedir permisos al sistema operativo para poder utilizar algunos componentes del hardware del computador del cliente, en este caso el micrófono integrado en el computador (Figura 3). Dependiendo el sistema operativo de la computadora cliente, se puede configurar google Chrome para que una vez aceptada la solicitud de permiso, quede guardada para siempre, y ya no sea necesario aceptar cada vez que se requiera.

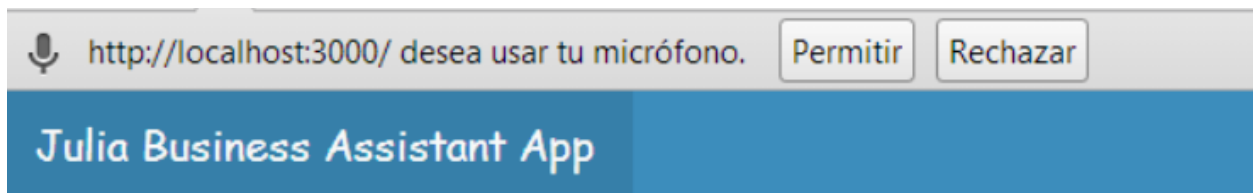
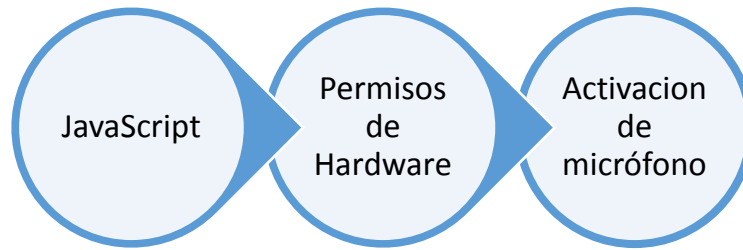


Figura 3. Interfaz de permiso de acceso a micrófono de Google Chrome

## 3. Activación de micrófono integrado para recuperación de información

- a. Como resultado de los dos pasos anteriores, la web puede detectar la voz del usuario, y posteriormente transcribirlo en un campo de texto (Figura 3).



*Figura 4.* Diagrama de procesos de reconocimiento de voz.

### **Ruby on Rails**

Julia debe ser capaz de funcionar como plataforma de operaciones organizacionales de cualquier empresa, desempeñando un papel crítico en la comunicación e interacción entre los miembros que la conforman los equipos de trabajo. Siendo así un asistente interactivo, comunicativo y útil, para la organización que lo implemente.

Julia tiene una interfaz bastante amigable, la cual tiene como objetivo principal informar al usuario las capacidades del sistema antes de ingresar al mismo, ofreciéndole la oportunidad de registrarse como nuevo usuario en caso de que sea su primera visita. El sistema cuenta con un formulario de registro y también uno de entrada, una vez que el usuario sea miembro de algún equipo de trabajo.

En el cumplimiento de objetivos de interactividad y comunicación, debe ser utilizada una tecnología web que permita el intercambio de información entre los usuarios. Esta tecnología es Ruby on Rails, ya que nos permite de forma segura y ordenada, administrar las principales tareas del usuario, las cuales se pueden clasificar en tres diferentes categorías:



## 1. Actividades individuales

- a. En cualquier empresa, el empleador tiene responsabilidades que debe cumplir, Julia fue capaz de funcionar como agenda, para estos usuarios, recordándoles sus actividades y compromisos individuales. Utilizando comandos de voz, el usuario puede comenzar a agendar una actividad personal. Las actividades personales, solo son presentadas en la pantalla del usuario creador de la actividad, de forma que se garantice su privacidad y exclusividad (figura 5).
- b. El orden en cómo se presentan las actividades personales, está relacionado con la fecha en cuando se deben llevar a cabo, lo que significa que Julia por defecto las acomoda según la actividad que está más próxima a cumplirse.



Figura 5 Interfaz de muestra de actividades personales calendarizadas y activas

c. Cada actividad personal tiene ciertos campos que deben ser completados antes de poder registrarse en la base de datos del sistema. Comprender que información debe ir en cada campo es importante para lograr el funcionamiento óptimo de Julia (Figura 6).

- i. Título: Debe introducirse un nombre o frase sirva de referencia para la actividad que será calendarizada.
- ii. Foto / cover: Una imagen que categorice el tipo de actividad a realizar.
- iii. Descripción: Información detallada de la actividad a realizar.
- iv. Fecha: Este campo responde a cuando se realizara la actividad.
- v. Hora: Define en que momento del día se llevara a cabo la actividad.
- vi. Duración: Cuanto se estima, debe tardar la actividad desde su principio hasta el final.
- vii. Lugar: responde a la ubicación en la cual se realizara la actividad.



Figura 6 Diagrama estructural de representación de datos de una actividad individual

## 2. Actividades grupales

a. Juntas administrativas, reuniones de personal, avisos generales, todos estos eventos requieren de más de un miembro de la empresa, Julia es un asistente grupal, lo cual significa que puede manejar las actividades que involucran a diferentes entidades dentro de una misma empresa (Figura 7).

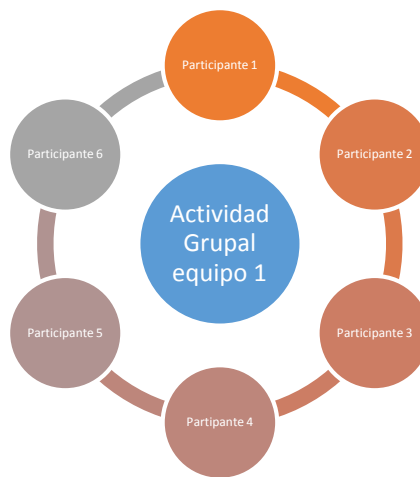


Figura 7 Diagrama estructural de comunicación entre miembros de los equipos

b. Al igual que las actividades personales o individuales, las actividades grupales son ordenadas según su proximidad cronológica.

c. Su principal particularidad, es que esta actividad aparecerá en las pantallas de todos los usuarios que formen parte del grupo, permitiendo así el intercambio de información entre sus miembros (Figura 8).

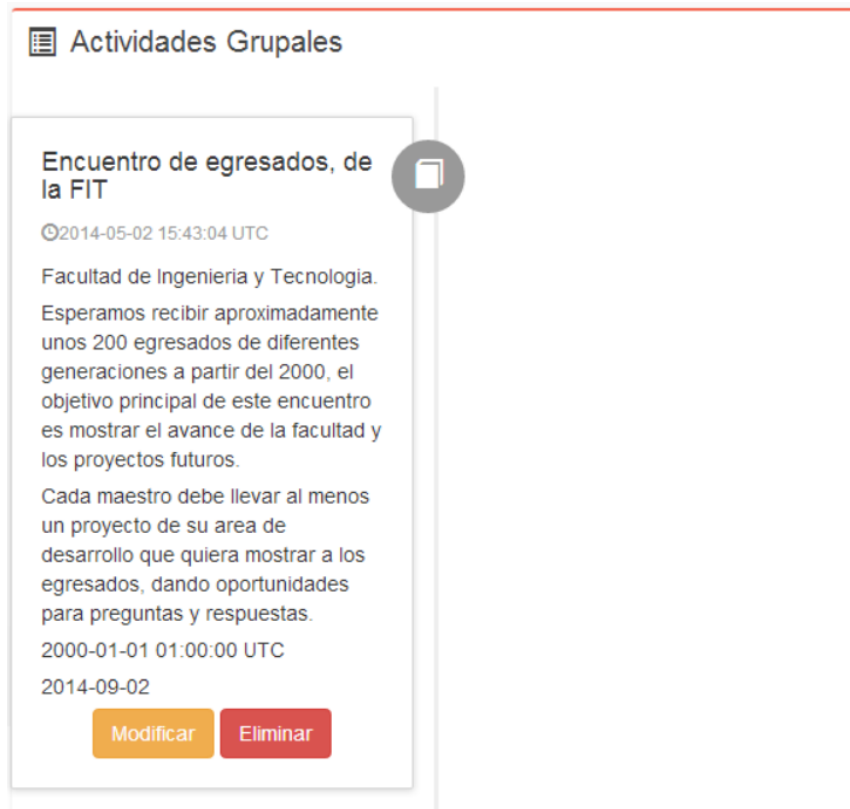


Figura 8 Interfaz de actividades grupales calendarizadas y activas

- d. Todas las actividades grupales, pueden ser modificadas o eliminadas. Además de que cuenta con una estructura visual de línea del tiempo.
- e. La estructura de la actividad grupal es muy parecida a la personal, solo que esta tiene un prerrequisito, el cual presenta que esperan los demás miembros del equipo tener listo, antes de realizar la actividad (Figura 9).



Figura 9 Diagrama de representación de datos de una actividad grupal

### 3. Avisos Generales

- a. Este es el modelo más básico de todos. Y su objetivo es informar sobre cualquier evento, actividad o anuncio que quiera ser divulgado en el equipo de trabajo (Figura 10).
- b. Como los anuncios son generalmente breves, están maquetados en la parte superior, de forma que los usuarios puedan verlos a primera instancia.



Figura 10. Interfaz de avisos dados de alta en el sistema

- c. Los anuncios no pueden ser modificados, solo eliminados.
- d. Los avisos tienen una estructura mucho más simple que las dos anteriores, y puede ser representado de según los campos de texto que la componen (Figura 11).



Figura 11 Diagrama de representación de datos de módulo de avisos.

### **Presupuesto**

Las implicaciones económicas para el desarrollo e implementación de este proyecto son realmente mínimas, debido a que las tecnologías y softwares utilizados son de licencia libre, lo cual permite utilizarlas sin tener que realizar ningún pago. Sin embargo el futuro alojamiento de este sistema en la web si tiene un costo aproximado de \$936.00 mx al año. Ruby on Rails es un framework que no es soportado por muchos servidores web en la actualidad, es por este motivo que puede salir costoso su alojamiento en la red. . Pero por el momento está siendo desarrollado localmente.

## **Resultados Obtenidos**

La finalización de este proyecto, implica el desarrollo completado de un sistema capaz de convertir de voz a texto las peticiones y enunciados de los usuarios, y convertir de texto a voz las respuestas generadas por el sistema. Es definitivo que este tipo de proyectos pueden ser desarrollados con muchas más funcionalidades que las presentadas. Sin embargo en el cumplimiento de los objetivos presentados solo fueron necesarios obtener estos resultados:

1. Sistema Web: una aplicación que funciona sobre la estructura de la red computacional, compartiendo las ventajas de accesibilidad, y la interacción cliente-servidor, que le permite suplir a múltiples usuarios al mismo tiempo.

2. Interfaz (Visual): el aspecto visual de los sistemas, es muy importante en su adaptación a los modelos mentales del usuario, y a pesar de que Julia no fue enfocada a trabajar solo con la voz de usuario, se ha desarrollado una interfaz que muestre los elementos necesarios.

3. Actividades personales: incluso en los trabajos colaborativos, existen algunas responsabilidades individuales que deben ser tratadas de forma privada e independiente del equipo de trabajo, y es por este motivo que se desarrolló este módulo individual.

4. Actividades Grupales: la plataforma de comunicación de actividades grupales es la principal funcionalidad de Julia ya que permite la interacción entre múltiples usuarios, para compartir información.

5. Avisos generales: informan de mensajes que quieren ser comunicados a los miembros de un equipo de manera informal.

Con la finalidad de probar la fidelidad de conversión de voz a texto de Julia, y su utilidad en la calendarización de cualquiera de las tres funcionalidades, se escribió un texto base, el cual fue leído por 6 participantes seleccionados sin ningún criterio específico. Todo leyeron el texto y

en la sección de anexos se presentan los resultados obtenidos al realizar este análisis (Tabla 1). Y además se presenta una gráfica de género y resultados.



## CAPITULO III

### CONCLUSIONES Y TRABAJOS FUTUROS

#### Conclusiones

Al finalizar con este proyecto, se puede determinar que el procesamiento de lenguaje natural es un mundo muy amplio en el cual la complejidad es un término dominante en cada aspecto de desarrollo. Y en definitiva este proyecto representa solo un comienzo de una nueva forma de procesamiento de lenguaje natural utilizando tecnologías web.

El desarrollo de Julia fue un desafío que requiere un máximo esfuerzo, realmente para lograr desarrollar este sistema fue necesario el aprendizaje de varias tecnologías:

1. Ruby.
2. Ruby on Rails.
3. Google Speech.

El aprendizaje de estas tecnologías para su implementación e integración en el sistema, tomó aproximadamente el 70% de la totalidad del tiempo tomado para el desarrollo del proyecto.

Es importante utilizar las API's originales de google para voz y texto, ya que existen códigos utilizados y desarrollados por terceros que no funcionan con consistencia, y que su implementación, además de ser muy compleja, también crea conflictos con otros posibles scripts que se estén ejecutando, lo cual provoca el colapso de la aplicación en su totalidad. Es recomendable aprender las tecnologías antes de comenzar a implementarlas, esto no significa que sea necesario manejar las tecnologías aplicadas en un 100% de conocimiento, pero si es importante

saber lo necesario como para llevar a cabo el proyecto, porque hay problemas con soluciones muy simples, solo si se conoce la herramienta en uso.

Uno de los factores que se deben tomar en cuenta para proyectos futuros es la conexión a internet, el sistema en cierta forma depende de google para la traducción de voz a texto y viceversa, lo que significa que mientras más rápido sea la conexión a internet también será el tiempo de duración de transcripción y de generación de sonido.

Todos los algoritmos que se están aplicando a Julia, pueden y deben ser más robustos, de forma que puedan realizar análisis más complejos en el procesamiento de lenguaje natural, que puedan reconocer palabras utilizando un corpus o diccionario rico en palabras, lo cual aumentara la inteligencia de la aplicación.

Los sistemas de información tienen un papel importante en las empresas, ya que permiten exclusividad, seguridad y funcionalidad. Una de las principales preocupaciones en las organizaciones de gran tamaño es la falta de comunicación entre sus miembros, lo que con frecuencia crea malos entendidos y puede debilitar la efectividad de la organización. Una plataforma en la cual los miembros de la organización puedan interactuar sin ningún tipo de distracciones, utilizando las tecnologías de información y comunicación, puede ser de mucha utilidad el cumplimiento de sus objetivos.

Sin embargo, la utilidad de un asistente se mide en la cantidad de ventajas que ofrece, y Julia presenta las siguientes ventajas al usuario:

1. Creación y administración de Grupos de colaboración.
2. Intercambio de información entre miembros.
3. Calendarización de Actividades grupales e individuales.
4. Divulgación de anuncios dentro de la organización.

Es definitivo que los seres humanos siempre preferirán sistemas que se adapten a su forma de comunicación, y no viceversa. Uno de los objetivos principales de Julia fue poder realizar operaciones simplemente utilizando comandos de voz, lo cual permitiera que el usuario se comunique más inteligentemente con el sistema, de esta forma mejorando su experiencia de uso. Sin embargo también es importante considerar que el posicionamiento de los elementos en la pantalla del usuario debe acomodarse a su modelo mental, de forma que facilite el entendimiento del usuario en cuanto al funcionamiento del sistema.

### **Aportaciones Futuras**

Existen todavía muchas áreas más que fortalecer en el desarrollo de un agente colaborativo, una de ellas hace es un sistema de mensajería interno, que permita a los miembros de un equipo comunicarse de forma privada y exclusiva, sin necesidad de utilizar ninguna plataforma alterna y externa del alcance de la empresa u organización.

Otro aspecto que pudiera ser integrado es la videoconferencia, en la cual múltiples miembros pueden comunicarse en vivo, tanto por voz como por video. Entre las alternativas y complementos de Ruby on Rails existen algunas que pueden hacer esta implementación más simple de lo acostumbrado.

Otro asunto que debería seguir siendo trabajado es el nivel de interacción de lenguaje natural de Julia, ya que los algoritmos son muy simples aun, deberían desarrollarse más, para de esta manera mejorar la inteligencia de Julia y permitir mejor comunicación entre los miembros de los equipos de trabajo.

## Apéndices

### Avisos

---

1. Crear nuevo aviso general.
2. Con un título de ‘El día 18 de mayo no habrá clases’, con una descripción de ‘Por este medio les informamos que durante el día 18 de mayo, no realizaremos ninguna actividad laboral, por una decisión administrativa’.

### Actividad Personal

---

1. Calendarizar nueva actividad personal.
2. Con un título de ‘Reunión de vicerrectoría Académica’, en el lugar de ‘Universidad de Montemorelos, Nuevo León’, con una descripción de ‘En esta reunión se tomarán las decisiones que definirán el rumbo de las actividades académicas del siguiente semestre’.

### Actividad Grupal

---

1. Calendarizar nueva actividad grupal.
2. Con un título de ‘Encuentro en el departamento de sistemas de la UM’, en el lugar de ‘Sala de juntas del departamento de sistemas de la Universidad de Montemorelos’, con una descripción de ‘Discutiremos temas de implementación de sistemas, desarrollo de tecnologías y congresos.’ Y con un prerrequisito de ‘llevar los documentos de proyectos de innovación’.

Figura 12. Texto base para realización de Pruebas

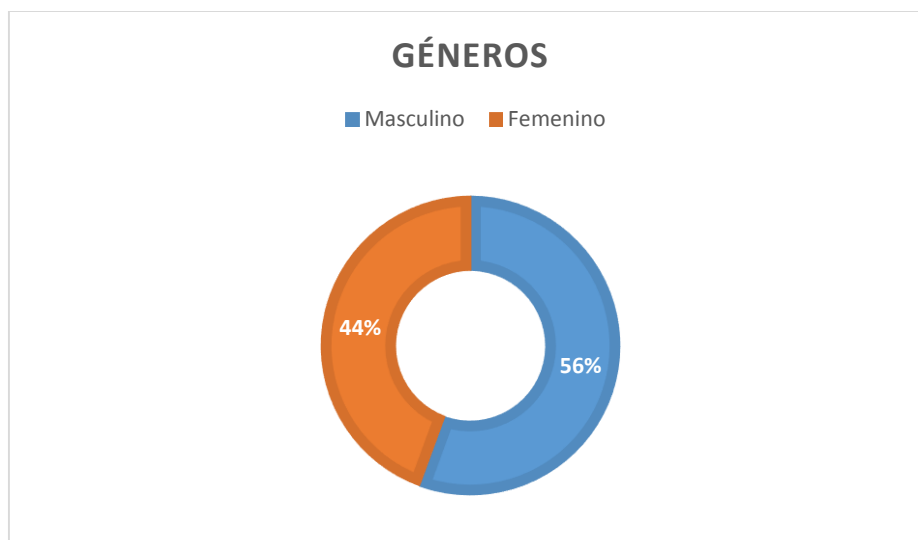


Figura 13 Gráfica de géneros de individuos participantes de prueba de lectura.

Tabla 1.

Nombre	Voz bienvenida	Voz de Espera de comando	Fidelidad					
			Primer texto	Segundo texto	Tercer texto	Cuarto texto	Quinto texto	Sexto texto
Participante 1			$\frac{3}{4} = 75\%$	$\frac{25}{37} = 67.5\%$	$\frac{3}{4} = 75\%$	$\frac{37}{39} = 94.8\%$	$\frac{4}{4} = 100\%$	$\frac{54}{54} = 100\%$
Participante 2			$\frac{3}{4} = 75\%$	$\frac{35}{37} = 94.5\%$	$\frac{3}{4} = 75\%$	$\frac{27}{39} = 69.2\%$	$\frac{4}{4} = 100\%$	$\frac{54}{54} = 100\%$
Participante 3			$\frac{4}{4} = 100\%$	$\frac{37}{37} = 100\%$	$\frac{4}{4} = 100\%$	$\frac{37}{39} = 94.8\%$	$\frac{4}{4} = 100\%$	$\frac{51}{54} = 94.4\%$
Participante 4			$\frac{1}{4} = 25\%$	$\frac{33}{37} = 89.1\%$	$\frac{4}{4} = 100\%$	$\frac{30}{39} = 81.0\%$	$\frac{4}{4} = 100\%$	$\frac{49}{54} = 90\%$
Participante 5			$\frac{4}{4} = 100\%$	$\frac{37}{37} = 100\%$	$\frac{4}{4} = 100\%$	$\frac{39}{39} = 100\%$	$\frac{4}{4} = 100\%$	$\frac{52}{54} = 96.2\%$
Participante 6			$\frac{3}{4} = 75\%$	$\frac{30}{37} = 81.5\%$	$\frac{4}{4} = 100\%$	$\frac{30}{39} = 76\%$	$\frac{4}{4} = 100\%$	$\frac{51}{54} = 94.4\%$
<b>Totales</b>	<b>100%</b>	<b>100%</b>	<b>75%</b>	<b>88.76%</b>	<b>91.6 %</b>	<b>85.9%</b>	<b>100%</b>	<b>95.8%</b>
<b>Nivel de reconcomiendo general: 89.51</b>								

Nombre: se refiere al nombre con el que el usuario a prueba se registró en el sistema.

Voz bienvenida: Julia pronuncia el nombre del usuario dándole la bienvenida al sistema.

Voz de espera de comando: Julia pronuncia “Julia está escuchando”, lo que significa que esta lista para escuchar el comando que el usuario va a introducir.

Fidelidad: porcentaje de aciertos en la traducción de textos.

## REFERENCIAS

- Julio Villena Roman, Raquel M. y otros (2011). *Comunicación en Redes de Comunicaciones*. Universidad Carlos III de Madrid
- Alexander Gelbukh (2010). *Procesamiento del Lenguaje Natural y sus Aplicaciones*. Año II, Vol 1. Komputer Sapiens.
- Alejandro Peña Ayala (2006). *Lenguaje Natural: Descripción de las Etapas para su Tratamiento*. Primera Edición. Instituto Politécnico Nacional. México.
- Eva M. Méndez Rodríguez y José A. Moreiro González (1999). *Lenguaje natural e Indización automatizada*. Universidad Carlos III de Madrid (España).
- Bellesi Facundo y Flavio Ortiz (2008). *Reconocimiento de Voz para Aplicación de Domótica*. Universidad Tecnológica Nacional.
- Lourdes Araujo (2010). *Procesamiento de Lenguaje Natural*.
- José Manuel Soto Corzo, David Díaz Portillo, José Antonio Cruz Zamora (2001). *Sistema de Consultas en Lenguaje Natural para Bases de Datos*. Instituto Tecnológico de Apizaco.
- Adorf, J. (2013). Web Speech API.
- Bächle, M., & Kirchberg, P. (2007). Ruby on Rails. *IEEE Software*, 24(6), 105-108.
- Blair, G. M. (1992). Personal time management for busy managers. *Engineering Management Journal*, 2(1), 33-38.
- Cooper, R. S., McElroy, J. F., Rolandi, W., Sanders, D., & Ulmer, R. M. (2002). Personal virtual assistant with semantic tagging: Google Patents.
- Cooper, R. S., McElroy, J. F., Rolandi, W., Sanders, D., Ulmer, R. M., & Peebles, E. (2004). Personal virtual assistant: Google Patents.
- Elshafei, M. Virtual Personal Assistant (VPA) for Mobile Users.
- Guide, A. P. (2006). Agile web development with Rails.
- Holm, F., & Pearson, S. (1998). User interface controller for text-to-speech synthesizer: Google Patents.
- Ingram, A. L., & Hathorn, L. G. (2004). Methods for analyzing collaboration in online. *Online collaborative learning: Theory and practice*, 215-241.

- Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5), 544-551.
- Nair, P. R. (2013). E-Supply Chain Management using Software Agents. *CSI Communications*, 37(4), 13-16.
- Okamoto, S., Sycara, K., & Scerri, P. (2009). Personal assistants for human organizations. *Organizations in Multi-Agent Systems*.
- Sengupta, B., Chandra, S., & Sinha, V. (2006). *A research agenda for distributed software development*. Paper presented at the Proceedings of the 28th international conference on Software engineering.
- Talia, D. (2011). *Cloud Computing and Software Agents: Towards Cloud Intelligent Services*. Paper presented at the WOA.
- Jose Luis Orpeza Rodriguez (2006). *Algoritmos y métodos para el Reconocimiento de Voz en Español Mediante Silabas*. *Computación y sistemas* Vol. 9 Num. 3, pp. 270-286. Centro de Investigacion en Computación-IPN, México D.F.