Capítulo 2: Análisis de la Aplicación Prototipo

2.1 Antecedentes

La comunicación es un proceso de interacción social entre sujetos que se basa principalmente en la transmisión de información de un ente a otro a través de símbolos, señales y sistemas de mensajes, es el principal medio para llevar a cabo la interacción entre dos individuos, ya sea través del lenguaje o por otros medios. Este proceso está formado principalmente por los siguientes elementos:

- Emisor: es el sujeto donde comienza el proceso, es el encargado de generar y transmitir el mensaje.
- Mensaje: es la información transmitida.
- Receptor: es el individuo que recibe el mensaje y lo interpreta.
- Canal: es el medio por el cual se transmite la información.
- Código: es un conjunto de reglas y símbolos, que el emisor y el receptor deben de conocer para generar e interpretar el mensaje.

Es importante destacar que dentro de este proceso los roles del emisor y el receptor, no deben ser estáticos si no mantener un intercambio constante.

La comunicación oral es una herramienta que permite a los seres humanos materializar sus pensamientos, expresar sus ideas, sentimientos y necesidades de forma natural. Es principalmente usada, para que los seres humanos se comuniquen entre sí. El ser humano tiene, por naturaleza la necesidad de comunicarse, la acción de relacionarse con los demás es tan importante que significa incluso, la supervivencia misma del hombre. Esta necesidad se presenta desde que nacemos y es una constante en las diferentes etapas de nuestra vida [Cantú et al, 1999].

Alan Hancock presenta la siguiente idea: “Comunicación es un concepto lo suficiente elástico para incluir formas de comunicación interpersonal, institucional y medios masivos de comunicación; y si es visto principalmente como un proceso, puede ser entendido en muchos términos…” [Hancock, 1981]. Basados en esta idea es posible

pensar en una variante distinta de la comunicación una en la cual uno de los elementos básicos (receptor o emisor) es desarrollado por una computadora. La computadora se ha usado dentro de este proceso de comunicación desde los inicios de Internet (1969) como el Canal, medio por donde se transmite la información. En la actualidad se pretende que la computadora tome un papel más activo dentro del proceso de comunicación realizando las actividades propias de receptor y emisor.

El uso de herramientas de reconocimiento de voz, nos permitirá usar este canal de comunicación y trasladarlo a un ambiente dónde la comunicación no solo se base entre personas, si no de una persona a una computadora y de una computadora a una persona, en la cual la computadora toma el rol de receptor, recibiendo la información que la persona le da por medio de una entrada de voz, esta información será procesada y la computadora responderá en base a ella.

La función principal de un reconocedor de voz o reconocedor del habla, como también es conocido, es convertir un mensaje hablado, capturado por un micrófono o un teléfono a texto. El autor Sadaoki Furui nos presenta la siguiente definición: "Reconocimiento de voz es el proceso de extraer automáticamente y determinar la información lingüística dada por un archivo de voz usando computadoras o circuitos electrónicos …" [Furui,1989].

Los sistemas de reconocimiento de voz se componen fundamentalmente de tres funciones importantes. Primero, las palabras se capturan y se traducen a una señal digital. Entonces un algoritmo de reconocimiento de voz traduce la señal a texto y lo compara con las palabras y las frases de un diccionario preestablecido. Finalmente, ofrece la alternativa más probable para la frase dictada (Figura 5) [Kate, 2006].

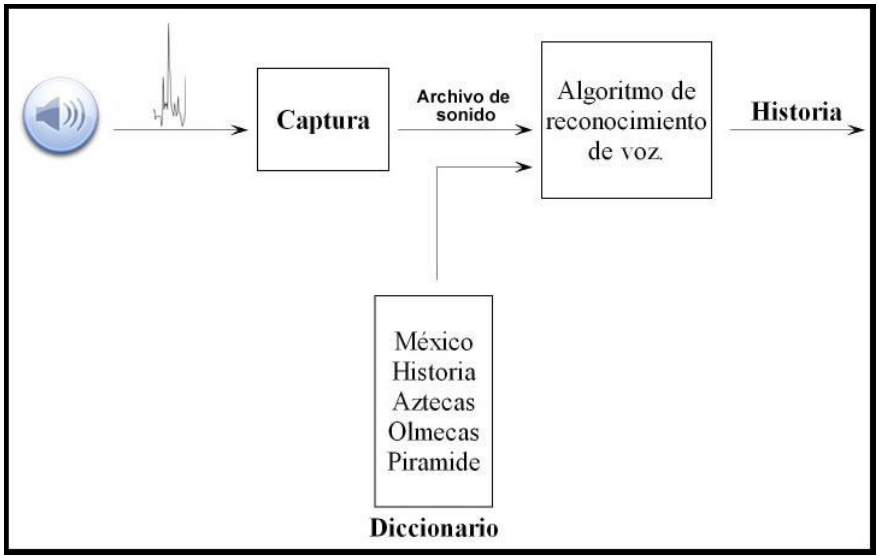

Figura 5. Sistema reconocedor de voz.

Existen diferentes clasificaciones para los sistemas de reconocimiento de voz:

- Los sistemas independientes del locutor pueden reconocer voz de cualquier persona.
- Los sistemas dependientes del locutor deben ser entrenados para cada usuario individual, pero típicamente tienen más altas tasas de exactitud.
- Los sistemas adaptables al locutor, un enfoque híbrido, inicia con plantillas independientes del locutor y las adapta a usuarios específicos sobre el tiempo sin entrenamiento explícito.
- Los sistemas de reconocimiento de voz continua pueden reconocer palabras habladas en un ritmo natural mientras que los sistemas de palabras aisladas requieren de una pausa deliberada entre cada palabra. El primero es el más deseable y natural para el usuario pero la voz continua es más difícil de procesar por la dificultad en detectar los límites de cada palabra [Grasso et al, 1999].

Los sistemas de reconocimiento de voz tienen sus inicios en 1952, año en el que fue desarrollada la primera aplicación sobre una computadora analógica para reconocer los dígitos del 0 al 9, dependiente de la persona que habla. Con una exactitud de 98%. Más tarde en esa misma década, un sistema con atributos similares fue desarrollado, este sistema reconoció consonantes y vocales. Las limitaciones tecnológicas en cuanto

a arquitecturas computacionales freno cualquier tipo de desarrollo de sistemas comerciales de reconocimiento de voz [Grasso et al, 1999].

Es importante tomar en cuenta las limitaciones tecnológicas con las que enfrentan los sistemas de reconocimiento de voz, pero estas no son las únicas. El autor Chris Rowden muestra cinco tipos de dificultades a las que se enfrentan los reconocedores de voz [Rowden, 1992]:

Primero, las señales del habla tienden a ser continuas. El habla es un patrón de sonidos en continuo cambio sin una señal o marca que especifique en qué punto termina un sonido y donde se ha llegado al comienzo de otro sin un límite obvio entre una palabra y otra. En particular no existen pausas regulares entre palabras en una expresión [Rowden, 1992].

Segundo, las señales del habla son extremadamente cambiantes. La voz de una persona es muy diferente comparada con otra, cada individuo posee diferentes acentos y dialectos o simplemente debido a una desigualdad física en el sistema vocal. Dentro de estas diferencias entran los distintos lenguajes que existen en todo el mundo, pues obviamente las palabras se pronuncian de forma diferente. Incluso en el español existen variaciones geográficas en la pronunciación. Un ejemplo de esto es el español usado en España comparado con el de México. Incluso la voz de un mismo individuo puede variar en diferentes condiciones, por ejemplo mientras susurra o grita, mientras sufre un resfriado. De hecho, es virtualmente imposible para una persona decir la misma palabra o frase exactamente de la misma manera en dos ocasiones diferentes [Rowden, 1992].

Tercero, el habla es ambigua. Las palabras usualmente tienen varios significados. Por ejemplo, la palabra sal define varias cosas, en gastronomía se refiere a un condimento y en química al producto típico de una reacción entre una base y un ácido. Además, existen palabras con pronunciaciones muy parecidas, es decir, suenan muy similares al ser pronunciadas [Rowden, 1992].

Cuarto, las señales del habla se encuentran frecuentemente contaminadas. Usualmente las señales ocurren en un ambiente donde existe algún grado de eco o

donde hay ruidos acústicos que compiten con la señal, los cuales incluyen otras voces, y en algunos casos estas señales de interferencia y ruidos pueden ser más fuertes que la señal de nuestro interés. También, la señal tal vez ha pasado por un canal de comunicación, como son una línea telefónica o un radio, los cuales pueden agregar más complicaciones como distorsión y retrasos [Rowden, 1992].

Finalmente, el habla es muy compleja. Lenguaje y habla están íntimamente relacionados y el habla es solo un pequeño componente de un complicado sistema de símbolos y señales para la comunicación de pensamientos e ideas entre personas. El dialogo humano a humano está repleto con comportamientos los cuales están diseñados para llevar a cabo una comunicación efectiva [Rowden, 1992].

Otra de las limitaciones que es muy importante destacar es el tamaño del vocabulario, debido a que entre más aumenta este el reconocedor pierde precisión en sus resultados. Antes de la década de los ochentas los sistemas de reconocimiento de voz mostraban problemas con vocabularios limitados a 200 palabras. Estos primeros reconocedores funcionaban para tareas simples como control de calidad, clasificación postal, y servicios bancarios, solo funcionaban para un número muy limitado de aplicaciones. Sin embargo para disfrutar de los beneficios de una comunicación hombre-máquina son necesario vocabularios más extensos. Se estima que los humanos pueden reconocer entre 50,000 y 100,000 palabras y están continuamente olvidando y actualizando su léxico interno. Los problemas a los que nos enfrentamos al usar vocabularios muy extensos son los siguientes:

- La factibilidad y costo de incrementar y mantener una gran base de datos que contenga plantillas de palabras es prohibitivo.
- Problema con nuevas palabras. El sistema debe ser fácilmente modificable y extensible para incorporar o eliminar nuevas palabras o vocabularios sin necesidad de readaptar el sistema.
- Aprendizaje y adaptación. Los sistemas de reconocimiento de voz en aproximación al rendimiento humano deben de ser capaces de aprender nuevos conocimientos útiles, mejorar en futuros reconocimientos así como también adaptarse a las circunstancias de una tarea, hablante, ambientes con ruido etc.

- Costo, velocidad y capacidad de almacenaje computacional. Búsquedas usando fuerza bruta llegan a ser cada vez más costosas tanto en hardware como en velocidad con el aumento del vocabulario.
- Ambigüedad. Grandes vocabularios contienen muchos subconjuntos de palabras acústicamente ambiguos [Waibel, 1988]

Estas limitantes no han sido un impedimento para que se desarrollen aplicaciones comerciales enfocadas al reconocimiento de voz. Las cuales debido a las dificultades que aun existen la gran mayoría están limitadas a aplicaciones telefónicas, aplicaciones de respuesta automatizada, aplicaciones para controlar la computadora u otros dispositivos y aplicaciones que realizan tareas de dictado. [Joch, 2002]. A continuación un ejemplo de estas aplicaciones:

Dragon Naturally Speaking Preferred 9 Español. Esta es la última versión liberada por la compañía NUANCE. Las características de esta aplicación es que no necesita capacitación previa, permite realizar correcciones en los dictados, ofrece la máxima precisión de reconocimiento de voz, así como controlar tu ordenar y navegar por la Web, TTS, entre otras. El precio de esta versión es de $199 euros. Los requerimientos mínimos son: procesador a 1 GHz., 512 MB. de RAM, 1 GB. de disco duro, Windows 2000, XP o Vista, y un sistema de audio (tarjeta de sonido más micrófono) [NUANCE, 2007].

IBM ViaVoice Pro USB Edition. Esta aplicación desarrollada por IBM, habilita las funciones de dictado, capaz de realizar correcciones, TTS, permite navegar por la Web usando la versión ViaVoice Web Millenium, cuenta con comandos de voz y control de la computadora, después de que el usuario realice un previo entrenamiento. Este reconocedor está disponible para la lengua española. Cuenta con un vocabulario de 300,000 palabras y vocabularios con temas especializados como computación o finanzas. El precio de esta versión es de $190 dólares. Via Voice corre en sistemas operativos como son Windows y MAC.

FreeSpeech 2000. Está disponible en 14 idiomas, entre ellos tres versiones diferentes de español (para España, Centroamérica y Suramérica). Necesita de 15

minutos de entrenamiento previo. El vocabulario está basado en un léxico de referencia de 450,000 palabras desarrollado en cooperación con Oxford University Press. Posibilidad de trabajar con más de un usuario. Permite tareas de dictado, control de las aplicaciones del ordenador así como navegar por la Web. Su precio está alrededor de los $160 euros [Madrid, 2007].

Por último, tenemos Voice XPress Professional 5 desarrollado por Lernout & Hauspie. Esta versión cuenta con funciones de dictado y control de la computadora por medio de comandos de voz intuitivos para el usuario, TTS. Requiere entrenamiento previo. No permite navegar a través de la Web. Su precio aproximado es de $150 dólares. Estas son las cuatro aplicaciones más populares en la actualidad.

En cuanto a software para invidentes se cuenta con la aplicación JAWS que es un lector de pantalla, usa un sintetizador de voz para leer en voz alta la información contenida en la pantalla, puede ser de una aplicación o de la Web, es capaz de trabajar en varios idiomas incluyendo el español, permite navegar en internet su precio es de $1095 dólares [Freedom Scientific, 2007].

ZoomText Magnifier/Reader 9.1 es una aplicación que permite ampliar cualquier aplicación dentro de la pantalla y al mismo tiempo reproduce en voz alta el texto que se encuentran en la aplicación. Disponible en ingles y su precio es de $595 dólares [Ai Squared, 2006].

Home Page Reader 3.0 es un navegador de internet que permite a las personas invidentes o con visión reducida utilizar el Internet sin dificultad. Lee la información que aparece en el monitor, facilitando la lectura de pantallas completas, párrafos, oraciones, palabras y letras. Tienes un precio de $ 469 dólares disponible en español [IBM, 2007].

En esta tesis se investigó cómo implementar una herramienta que permita hacer búsquedas en la Web utilizando la voz como entrada. Se analizó la viabilidad de este proyecto, las aplicaciones y tecnologías existentes, los requerimientos que son necesarios para realizarse y por último se desarrolló una aplicación prototipo que a su

vez pueda ligarse a un módulo (desarrollado en otra tesis) que es capaz de leer los resultados de la búsqueda. Esto con el propósito de dar más libertad a usuarios que tienen ocupados manos y ojos en otra tarea o en ambientes donde el uso del teclado resulta incomodo o poco práctico. Así como ayudar a usuarios con diferentes discapacidades como invidentes o con visión reducida y con problemas motrices. Además esta aplicación debe de ser gratuita y basada en español mexicano

2.2Tendencias Actuales

Actualmente los motores de búsqueda están en una continua evolución. Tal es el caso de Google, el cual desde abril del 2006 recibió la patente número 7027987 para una interfaz de voz para los motores de búsqueda. Esto ha dado como resultado el servicio GOOG-411. Básicamente es un número gratuito (01-800-4664-411) el cual permite a los usuarios comunicarse con Google desde cualquier teléfono. Una vez realizada la llamada una voz preguntará en que ciudad está lo que está buscando, el tipo de negocio que está buscando para después enumerar los resultados, basta con que se pronuncie el numero que le interesa para que nos transfiera la llamada al negocio seleccionado o solamente para que nos dé información como dirección exacta y número de teléfono del local. Este servicio es gratuito, solo está disponible para los Estados Unidos y en ingles [Google, 2007].

Otra compañía que sigue esta misma tendencia es Microsoft pues recientemente adquirió la compañía Tellme, una empresa estadounidense, sus servicios permiten que los clientes usen la voz para buscar en la Web información sobre negocios en su ciudad, direcciones, cotizaciones de acciones, información del tiempo e incluso noticias. Con esto Microsoft conjuntamente con la compañía telefónica Sprint, desarrollan un servicio de búsquedas multimodal. De esta forma los clientes de Sprint serán capaces de realizar búsquedas usando la voz. Para ello, Microsoft ha integrado tecnología de reconocimiento de voz que adquirió con la compra de Tellme. Es decir, los usuarios podrán introducir la palabra o palabras clave para su búsqueda mediante teclado o voz [ZDNet News, 2007].

Estas no son las únicas compañías pues Yahoo está haciendo algo parecido con su servicio OneSearch. De la misma manera que NUANCE con Nuance Mobile. Lo anterior nos muestra que existe una clara tendencia en la actualidad a realizar las consultas por medio de la voz. Pareciera ser el nuevo camino que están tomando la mayoría de los motores de búsqueda, el problema con estos es que solo funcionan en ingles.

2.3 Análisis de los componentes del sistema

Para la construcción de esta aplicación prototipo se analizarán los elementos disponibles con los que se cuentan. Los elementos básicos que se usarán son los siguientes: reconocedor de voz, sintetizador de voz (TTS), vocabulario y una herramienta que nos permita realizar consultas en la Web. A continuación se muestra la arquitectura, sin detalles, de cómo se pretende que la aplicación funcione (Figura 6).

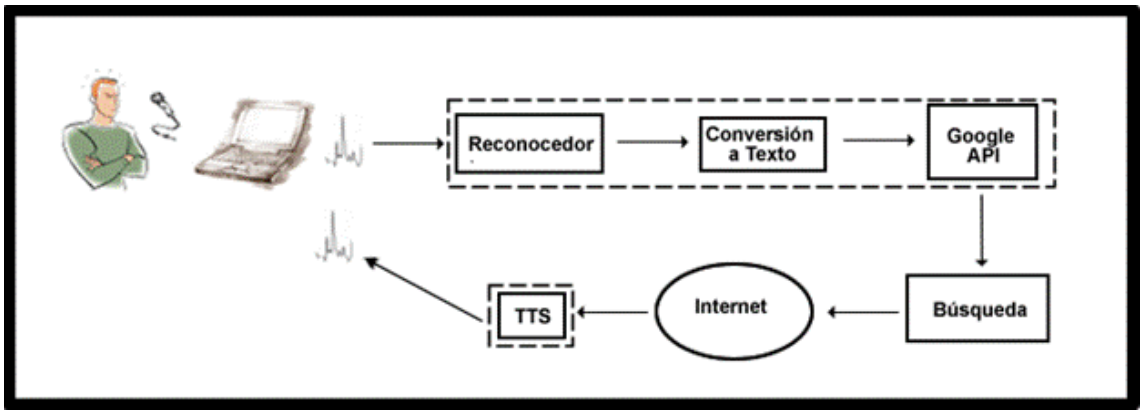

Figura 6. Arquitectura simple de la aplicación (Los rectángulos punteados indican los módulos que requieren ser programados).

La aplicación utiliza alguno de los reconocedores de voz disponibles. Este se encarga de convertir la consulta realizada por medio de voz a texto el cual es enviado al API de Google para que realice la búsqueda en la Web y regresen los resultados al usuario por medio de voz usando un sintetizador de voz.

Esta aplicación fue desarrollada en Java usando Java Development Kit (JDK) 5.0, esto debido principalmente a que los trabajos previos existentes en la universidad

están desarrollados en este lenguaje, Java cuenta con una gran cantidad de APIs (Application Programming Interface), entre ellas el API de Google que es de vital importancia para el funcionamiento de la aplicación, hace posible el uso de librerías que permiten la interacción con el reconocedor y el sintetizador de voz. Además de sus características principales como el hecho de ser un lenguaje orientado a objetos, independiente de la arquitectura, permite el uso de la maquina virtual (JVM) entre otras.

2.4 Herramientas para la construcción del sistema

La herramienta base de esta aplicación es el reconocedor de voz, es el encargado de permitir una comunicación natural entre las personas y la computadora, por medio de la voz. Existen varias opciones las cuales serán mostradas a continuación.

El **CSLU Toolkit** fue desarrollado por el CSLU (Center for Spoken Language Understanding). Desde la década de los noventas en el CSLU han estado trabajando en el desarrollo de nuevas herramientas, dando como resultado el CSLU Toolkit, una plataforma para la investigación y desarrollo de sistemas de lenguaje hablado. El Toolkit integra un conjunto de tecnologías que incluyen reconocimiento y síntesis de voz, así como animación facial. La arquitectura del Toolkit esta basa en tres elementos principales: un conjunto de librerías, un Shell interactivo de programación y un ambiente de desarrollo rápido de aplicaciones (RAD). Es importante mencionar que esta herramienta es gratuita, además de que el reconocedor y el sintetizador de voz pueden trabajar en español especialmente en español mexicano [CSLU, 2007].

Microsoft Speech API 5.3: SAPI como también es conocida, es una interfaz de reconocimiento del habla y de síntesis de voz para la programación de aplicaciones basadas en Win32 (Intel Win32s, Windows NT, Windows 95/98, MIPS Windows NT, DEC Alpha Windows NT, Power PC Windows NT, Windows XP). La SAPI proporciona una interfaz de alto nivel entre una aplicación y motores del habla. SAPI implementa todos los detalles de bajo nivel necesitados para controlar y manejar operaciones en tiempo real de varios motores. Los dos tipos básicos de motores SAPI son sistemas “text-to-speech” (TTS) y reconocedores de habla. Los sistemas TTS sintetizan texto y archivos en audio hablado usando síntesis de voz. Reconocedores del

habla convierten un audio con habla humana en texto legible y archivos [Microsoft, 2007].

SAPI es gratuita pero sólo permite usar el sintetizador de voz con los idiomas ingles de Estados Unidos y chino simplificado. El reconocedor de voz está disponible para trabajar con los mismos idiomas que el sintetizador además del japonés.

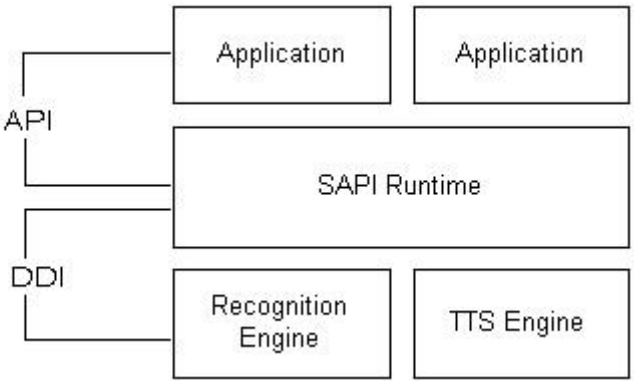

Figura 7. Visión general de la capas de la SAPI 5.3 Tomada de [Microsoft, 2007].

Java Speech API define un estándar, fácil de usar, relacionado con las tecnologías del habla. Dos núcleos son soportados a través de este API: reconocimiento de voz y síntesis de voz. El reconocimiento provee a las computadoras la habilidad de escuchar el lenguaje hablado y determinar que se ha dicho. En otras palabras, procesa la entrada de audio para convertirla a texto. El sintetizador de voz permite invertir el proceso genera audio a partir del texto, el cual puede ser generado por una aplicación, un applet o un usuario. Es importante destacar que Java Speech API no tiene un reconocedor y un sintetizador de voz propios, pero nos permite mediante el uso de este API manejar algunos de ellos, como son: FreeTTS, IBM's "Speech for Java", IBM's "Speech for Java" on Linux, The Cloud Garden, Lernout & Hauspie's TTS for Java Speech API, Conversa Web 3.0, Festival, Elan Speech Cube [Sun, 2007].

Después de analizar las herramientas antes mencionadas se tomó la decisión de usar el reconocedor y el sintetizador de voz ofrecidos por CSLU. Debido a que esta herramienta es gratuita y ofrece la capacidad de trabajar en español mexicano, este último módulo fue desarrollado dentro de la Universidad de las Américas-Puebla por el

Laboratorio de Tecnologías de Voz Tlatoa, además de permitir la interacción con Java por medio de unas librerías que serán mencionadas posteriormente.

Jacl 1.4.0 es un intérprete de Tcl (Tool Command Language) escrito totalmente en Java. Incluye características que facilitan la comunicación entre Java y Tcl. Jacl es usado típicamente para incorporar funcionalidad scripting dentro de una aplicación de Java existente. Además existe un API de Tcl que hace más fácil llamar código en Tcl desde Java. [Tcl/Java, 2006]. Dentro de la aplicación las librerías de Jacl permitirían la comunicación entre Tcl y Java.

Tcl (Tool Command Language) es un lenguaje de programación interpretado y multiplataforma. Fue creado por John K. Ousterhout y su equipo de la Universidad de California. Tcl es un lenguaje de comandos, cuyo intérprete recibe el nombre de tclsh el cual trabaja sobre una terminal. Es distribuido de forma gratuita, aunque su uso sea para aplicaciones comerciales, a través de Internet. [Universidad de Oviedo, 1998]. La importancia del Tcl radica en que por medio de scripts realizados en Tcl, es como se obtendrán las funciones de reconocimiento y síntesis de voz del CSLU Toolkit, y estos serán llamados por la aplicación desarrollada en Java para que esta obtenga, procese y presente los resultados.

Google es el motor de búsqueda más grande y más utilizado actualmente, pues según la compañía Google responde a más de 200 millones de consultas al día. Además como se mostro en el cápitulo anterior el 85% por ciento de los internautas mexicanos usan a Google como motor de búsqueda. Es por esto que se llego a la decisión de usar este motor para el desarrollo de la aplicación, el cual será encargado de recibir la consulta, realizar la búsqueda y entregar los resultados. Todo esto se llevará a cabo gracias a la API de Google la cual nos permite la interacción entre Google y la aplicación en Java.

Linter-Vox [López et al, 2006; Elizalde, 2006] un aplicación creada por Alejandro López R. y rediseñada por Gustavo Elizalde ambos del departamento de Computación, Electrónica y Mecatrónica (CEM) de Universidad de las Américas, Puebla. Esta aplicación permita a personas invidentes navegar en la Web. El usuario

ingresa su consulta mediante el teclado y Linter-Vox realiza la búsqueda en Google obtiene el resultado y los lee en voz alta, después de que el usuario escribe el numero de liga deseado la aplicación Linter-Vox lee toda la información contenida dentro de la pagina y las ligas existentes en la nueva página. De esta manera el usuario puede navegar en la Web a través de Linter-Vox. Dentro de la aplicación propuesta en esta tesis, Linter-Vox será la encargada de interactuar con el servicio de Google para realizar las búsquedas y leer los resultados, en base a la consulta dictada por el usuario. Linter-Vox fue programada en Java y usa el API de Google.

2.5 Arquitectura de la aplicación

Después de conocer todos los elementos que conforman esta aplicación y saber porque se van a usar. Se mostrará la arquitectura de la aplicación con todos los elementos funcionando. La arquitectura queda de la siguiente manera ver figura 8.

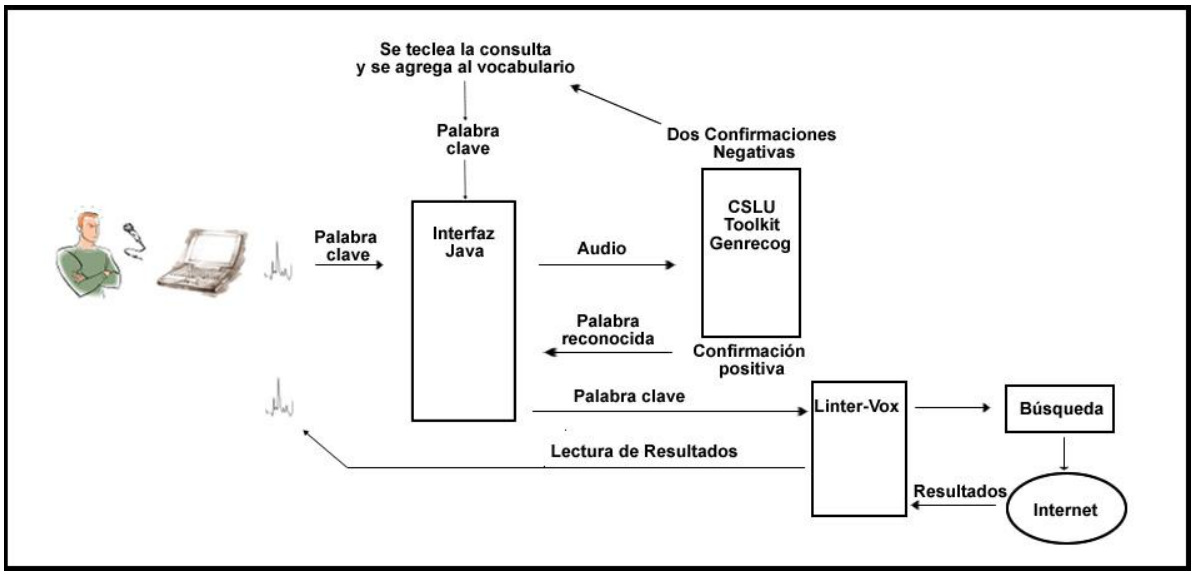

Figura 8. Arquitectura de la aplicación.

La entrada del usuario es recibida, por la aplicación que llamaremos Conquiro-Vox, creando un archivo WAV. Interfaz Java se encargara de llamar el CSLU Toolkit el cual recibirá el archivo WAV que se creó. El CSLU Toolkit procesa el audio y compara esta información contra las palabras que se encuentran dentro del vocabulario, en base a ellas toma un resultado, de esta forma la consulta es convertida a texto. En este

momento se lanza una confirmación al usuario, preguntándole si la palabra reconocida es la que el dicto, si la respuesta es positiva, la palabra se manda a Linter Vox. En caso que la respuesta a la confirmación sea negativa el reconocedor da otro resultado, con la palabra más parecida, el cual nuevamente necesita ser confirmado si la respuesta vuelve a ser negativa la aplicación abre una opción de agregar palabras la cual le pide al usuario que ingrese su consulta mediante el teclado dicha consulta se agrega al vocabulario y finalmente se manda a Linter-Vox. Una vez que Linter-Vox obtiene la consulta en texto lo manda al API Google como palabra clave, este realiza la búsqueda en la Web y los resultados de la búsqueda son regresados a Linter-Vox la cual se encarga leerlos en voz alta y permite navegar al usuario entre ellos. Esta es la función principal de la aplicación, pero además de esta permite realizar otras operaciones, las cuales serán descritas a detalle en el siguiente capítulo.

2.6 Diagrama de casos de uso.

El siguiente diagrama muestra como los diferentes usuarios pueden interactuar con el sistema en los diferentes escenarios que puede llegar a presentarle Conquiro-Vox durante su funcionamiento (Figura 9).

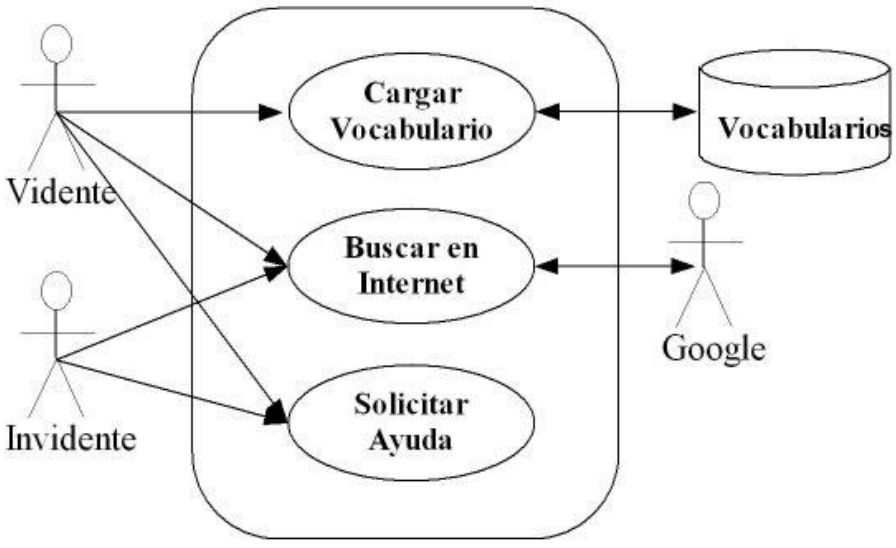

Figura 9. Diagrama de casos de uso general.

Sin importar el si el usuario es vidente o invidente, la interfaz permite que ambos interactúen con todos los componentes y funciones de la aplicación. Sin embargo, los usuarios invidentes no podrán realizar la acción de cargar un nuevo vocabulario, debido a que para realizar esta función es necesario que el usuario haga uso del "ratón" para seleccionar el vocabulario que más se adapte a sus necesidades, entre los diferentes vocabularios que la aplicación contiene. Lo cual es una acción difícil de realizar para una persona invidente.

Es importante mencionar que la aplicación opera dos formas distintas. La primera es la más común, la interfaz permite ser manipulada mediante el uso del "ratón" y el teclado para controlar todos los menús y botones que aparecen a lo largo de la ejecución del sistema, esto para los usuarios videntes o con visión parcial o reducida. La segunda permite a los usuarios invidentes, controlar la aplicación haciendo uso del teclado presionando la tecla "Alt" y la primera letra con la que comienza el nombre del botón, simule que fue cliqueado. Por ejemplo, si el usuario quiere salir de la aplicación tiene que presionar el botón salir presionando las teclas "Alt" y "s" al mismo tiempo.

A continuación se detallan los diferentes casos de uso con los que puede interactuar el usuario.

Caso de uso: Cargar Vocabulario.
Actores: Vidente.
Propósito: Cargar en la aplicación un nuevo vocabulario.
Descripción: La aplicación permite al usuario agregar o crear un nuevo vocabulario lo cual posibilita que el reconocedor pueda realizar su función basándose en diferentes contextos.
Referencias cruzadas:
Flujo normal de los eventos.

Acción del actor	**Respuesta del Sistema.**
1. Presiona el menú de "Cargar archivo".	2. Muestra una ventana con un selector de ficheros que contiene los vocabularios disponibles.

3. Selecciona la nueva lista de palabras.	4. Muestra una ventana la cual pregunta al usuario si quiere crear un vocabulario nuevo o quiere agregarlo al existente.
5. Selecciona “Vocabulario nuevo”.	6. Guarda una lista de palabras basada en el vocabulario existente y lo elimina.
	7. Procesa la nueva lista de palabras y crea un nuevo vocabulario o lo agrega al existente.
	8. Muestra una ventana de información avisando que el vocabulario se cargo correctamente.

Flujo alternativo de los eventos.

Línea 3. El actor decide no cargar ningún vocabulario. Termina este caso de uso.
Línea 5. El actor selecciona “Agregar vocabulario”. El sistema continúa en la línea 7.
Línea 7. El vocabulario contiene algún error de formato. El sistema muestra una ventana de error avisando que el vocabulario contiene números, acentos o signos no permitidos. Termina este caso de uso.

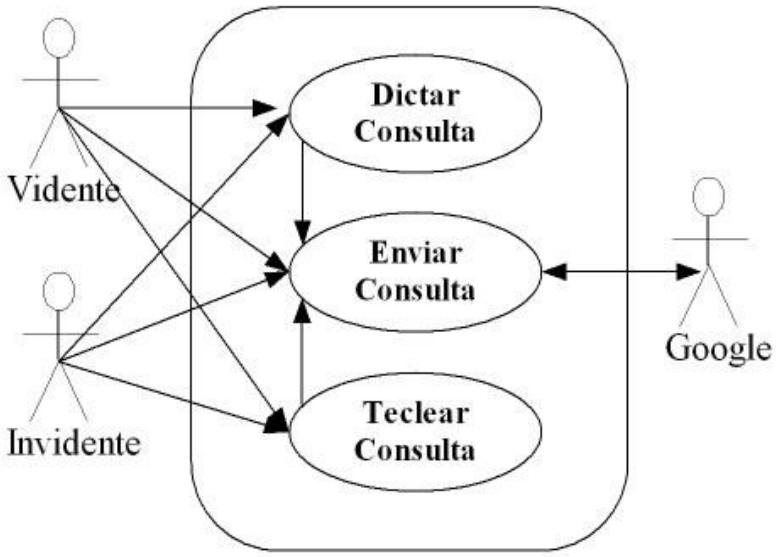

Figura 10. Diagrama caso de uso Buscar en Internet.

Caso de uso: Dictar Consulta.

Actores: Vidente e Invidente.

Propósito: Grabar la consulta dictada por el usuario.

Descripción: El usuario dicta la consulta deseada y esta es grabada en un archivo de audio.

Referencias cruzadas:

Flujo normal de los eventos.

Acción del actor	**Respuesta del Sistema.**
1. Presiona el botón de "Arranque".	2. Reproduce un sonido específico que indica que es momento de que el usuario dicta su consulta.
3. Dicta la consulta y enseguida presiona el botón de Alto.	4. Muestra una ventana que pregunta si se desea guardar esa grabación o se quiere reiniciar el proceso.
5. Presiona el botón "Guardar".	6. Se crea el archivo Res1.wav que contiene el audio con la consulta.

Flujo alternativo de los eventos.

Línea 5. Actor presiona el botón "Reiniciar". El sistema repite el proceso de dictado se regresa a la línea 2.

Caso de uso: Teclear Consulta.

Actores: Vidente e Invidente.

Propósito: Permitir que el usuario teclee su consulta.

Descripción: Una vez que el reconocedor ha fallado se muestra esta opción para que el usuario teclee su consulta.

Referencias cruzadas:

Flujo normal de los eventos.

Acción del actor	**Respuesta del Sistema.**
	1. Muestra ventana para que el usuario

	teclee su consulta.
2. Teclea consulta y presiona el botón "Aceptar".	3. Manda consulta a Linter-Vox

Flujo alternativo de los eventos.

Línea 2. No teclea consulta o presiona el botón "Cancelar". El sistema manda una ventana de error informando al usuario que debe de teclear su consulta, regresa a la línea 1.

Caso de uso: Buscar en Internet.

Actores: Vidente e Invidente.

Propósito: Realizar una búsqueda en la Web.

Descripción: La consulta del usuario es procesada por el reconocedor, el cual la convierte a texto y se la envía a Linter-Vox para que haga la búsqueda en la Web, a través de Google, y lea los resultados.

Referencias cruzadas:

Flujo normal de los eventos.

Acción del actor	**Respuesta del Sistema.**
1. Este caso comienza cuando el actor presiona el botón de Inicio.	2. Muestra una ventana con instrucciones de cómo el usuario debe dictar la consulta.
Entra en el caso de uso Dictar Consulta.	
	3. El reconocedor procesa la grabación obtiene los resultados.
	4. Muestra una ventana de confirmación preguntando si la palabra dictada fue la obtenida por el reconocedor.
5. Presiona el botón "Si".	6. Manda la palabra reconocida a Linter-Vox

Flujo alternativo de los eventos.

Línea 5. El usuario presiona el botón “No”. Sistema regresa a la línea 4 pero esta vez muestra otra palabra. Si en la línea 5 el usuario vuelve a presionar el botón “No” la aplicación entra en el caso de uso Teclear Consulta.

Caso de uso: Ayuda del sistema.

Actores: Vidente e Invidente.

Propósito: Mostar la ayuda del sistema al usuario.

Descripción: La aplicación lee en voz alta la ayuda del sistema.

Referencias cruzadas:

Flujo normal de los eventos.

Acción del actor	**Respuesta del Sistema.**
1. Presiona el menú de “Ayuda del sistema".	2. Lee en voz alta la ayuda del sistema.

En este capítulo se mostraron los antecedentes referentes al reconocimiento de voz así como las nuevas tendencias que están siguiendo los motores de búsqueda en cuanto al uso de esta herramienta. Otro punto importante fue el análisis que se hizo de las herramientas disponibles para la construcción de esta aplicación lo cual dio como resultado la elección del reconocedor y sintetizador del CSLU Toolkit. Se definió la arquitectura detallada de esta aplicación, la cual muestra su funcionamiento. Para terminar con los diagramas de casos de uso, los cuales muestran la forma en que el usuario interactúa con la aplicación durante el uso de sus diferentes funciones. Dentro del siguiente capítulo se describe el software de manera técnica y detallada. Se muestra el diseño y construcción de la aplicación así como los diagramas de clases y de secuencias.