

**Universidad Católica De Santa María**  
**Facultad De Ciencias E Ingenierías Físicas Y Formales**  
**Escuela Profesional De Ingeniería De Sistemas**



**DISEÑO DE ASISTENTE DE VOZ BASADO EN WEB PARA  
PERSONAS CON VISIÓN SUBNORMAL**

*Tesis presentada por los Bachilleres:*  
***Collado Salazar Katherin Zenaida***  
***Aparicio Maldonado Edmundo Aurelio***  
*para optar el Título Profesional*  
*de Ingeniero de Sistemas: Especialidad*  
*Ingeniería de Software*

Asesor: Ing. Fernández del Carpio Alvaro

**AREQUIPA – PERÚ**  
**2017**

UNIVERSIDAD CATOLICA DE SANTA MARIA  
URB. SAN JOSE S/N - UMACOLLO

FACULTAD DE CIENCIAS E INGENIERIAS FISICAS Y FORMALES  
ESCUELA PROFESIONAL DE INGENIERIA DE SISTEMAS  
DICTAMEN DE BORRADOR DE TESIS

VISTO

El Borrador de Tesis titulado:

"Diseño de Asistente de voz basado en web para  
personas con visión subnormal"

Presentado por (el) (la) (los) Bachilleres

- Katherin Zenaida Collado Salazar  
- Edmundo Aurelio Aparicio Maldonado

Nuestro dictamen es:

PROCEDENTE

OBSERVACIONES:

Arequipa, 16 de OCTUBRE de 2017

  
Ing. Alvaro Fernandez

  
Ing Carlo Corrales D.

## PRESENTACIÓN

Sr. director de la Escuela Profesional de Ingeniería de Sistemas

Sres. miembros del Jurado

De conformidad con las disposiciones de grados y Títulos de la escuela Profesional de Ingeniería de Sistemas, ponemos a vuestra consideración el presente trabajo de desarrollo tecnológico: **“DISEÑO DE ASISTENTE DE VOZ BASADO EN WEB PARA PERSONAS CON VISIÓN SUBNORMAL”**, el mismo que de ser aprobado nos permitirá optar por el Título Profesional de Ingeniero de Sistemas.

**COLLADO SALAZAR KATHERIN ZENAIDA**

**APARICIO MALDONADO EDMUNDO AURELIO**

## DEDICATORIA

### **Katherin Zenaida Collado Salazar**

Dedico este trabajo a mis padres, familia, amigos que me apoyaron para culminar este trabajo y lograr mis metas, este trabajo implico mucho esfuerzo además de ser una ayuda social, como que también dedico este trabajo a las personas con visión subnormal que me hicieron ver otra realidad en el mundo en el que vivimos y que debemos brindar ayuda.

### **Edmundo Aurelio Aparicio Maldonado**

Dedico este trabajo a mis padres, familia y personas más cercanas que me apoyaron durante todo este periodo a lograr mis metas, este trabajo es un esfuerzo grande que involucra a muchas personas cercanas a mí. También dedico este trabajo a esas personas que tiene una discapacidad de visión subnormal que me ayudaron a comprender mejor el mundo en el que vivimos y que no debemos dejarlo de lado.

## AGRADECIMIENTO

*A nuestro asesor de Tesis, Ing. Álvaro Fernández y a los directores de las instituciones CEBA Nuestra Señora del Pilar, CERCIA y CEPTRÓ Luz y Alegría, nuestro más amplio agradecimiento por habernos confiado este proyecto, por su paciencia, por su valiosa dirección y apoyo para seguir este camino y llegar a la culminación del trabajo de tesis.*

*A nuestras familias y a ustedes, nuestro mayor reconocimiento y gratitud.*



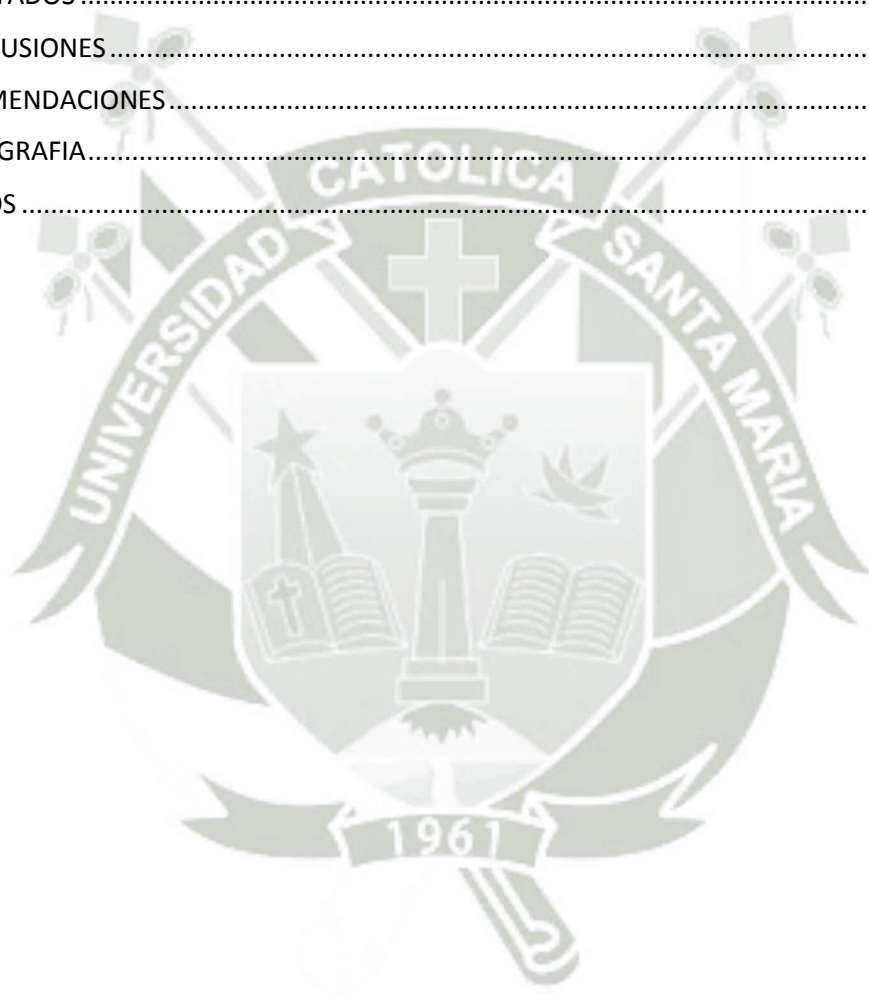
## ÍNDICE

RESUMEN .....	13
ABSTRACT .....	
INTRODUCCIÓN .....	
CAPITULO I: PLANEAMIENTO TEÓRICO.....	1
1.1. Título del proyecto .....	1
1.2. Descripción del problema .....	1
1.3. Delimitaciones y definición del problema.....	2
1.3.1. Delimitaciones.....	2
1.3.2. Definición del Problema.....	3
1.4. Formulación del Problema .....	3
1.4.1. Problema Principal .....	3
1.5. Objetivos de Investigación .....	3
1.5.1. Objetivo General .....	3
1.5.2. Objetivos Específicos.....	3
1.6. Viabilidad de la Investigación.....	4
1.6.1. Económica .....	4
1.6.2. Técnica .....	4
1.6.3. Operativa.....	5
1.7. Justificación e Importancia de la Investigación.....	5
1.7.1. Justificación .....	5
1.7.2. Importancia .....	5
1.7.3. Limitaciones de la investigación.....	5
1.8. Área, línea, tipo y nivel de investigación.....	5
1.8.1. Área de Investigación .....	5
1.8.2. Línea de Investigación .....	5
1.8.3. Tipo de Investigación .....	5
1.8.4. Nivel de Investigación .....	6
1.9. Método y diseño de Investigación .....	6
1.9.1. Método de Investigación .....	6
1.9.2. Forma de tratamiento de los datos .....	6
1.10. Cobertura del Estudio .....	6
1.10.1. Universo .....	6
1.10.2. Plan de Trabajo .....	7

CAPITULO 2: MARCO TEORICO .....	13
2.1. Estado del Arte.....	13
2.2. Ingeniera de Software.....	17
2.3. Aplicaciones Web.....	19
2.4. Lenguaje Unificado de Modelado .....	22
2.4.1. Objetivos del lenguaje UML.....	22
2.4.2. Semántica y notación .....	23
2.4.3. Diagramas que ofrece UML.....	24
2.5. Modelamiento de Clases.....	24
2.6. Casos de Uso .....	25
2.7. Diagrama de interacción .....	28
2.8. Sintetizador de Voz .....	30
2.9. Diagrama de un sistema Text-to-Speech (TTS).....	31
2.10. Reconocimiento de Voz .....	32
2.11. API Google.....	34
2.12. Lenguajes Web y Comparación.....	35
2.13. Lenguaje HTML.....	38
2.14. Lenguaje PHP.....	40
2.15. Lenguaje JavaScript.....	41
2.16. Diseño web responsive .....	42
2.17. JSON .....	43
2.18. Metodología AGILE SCRUM.....	43
CAPITULO 3: IDENTIFICACION Y ESPECIFICACIÓN DE REQUERIMIENTOS.....	45
3.1. Análisis de situación actual .....	45
3.2. Definición de requerimientos .....	46
3.2.1. Identificar actores .....	46
3.3. Catalogación de requerimientos.....	47
3.3.1. Requerimientos funcionales .....	47
3.3.2. Requerimientos no funcionales .....	48
3.4. Análisis de factibilidad para implementación web .....	50
3.5. Análisis de casos de uso .....	51
3.5.1. Elaboración de Caso de Uso Ingreso de valor de Búsqueda a petición del Usuario .....	51
3.5.2. Elaboración de Caso de Uso Confirmar Búsqueda a petición del Usuario.....	52

3.5.3. Elaboración de Caso de Uso Elección de respuesta a detallar a petición del Usuario .....	53
3.5.4. Elaboración de Caso de Uso recepción auditiva de la búsqueda detallada a petición del Usuario .....	54
<b>CAPITULO 4: DISEÑO E IMPLEMENTACION DE SOLUCION WEB.....</b>	<b>56</b>
4.1. Arquitectura de sistema.....	56
4.1.1. Arquitectura lógica.....	56
4.1.2. Arquitectura física.....	59
4.2. Secuencia de Solución Web .....	59
4.3. Actividades de Solución Web.....	60
4.4. Métodos principales.....	62
4.4.1. Método Start.....	62
4.4.2. Método Stop .....	62
4.4.3. Método Abort .....	62
4.5. Eventos Principales .....	62
4.5.1. Evento audiostart.....	63
4.5.2. Evento soundstart.....	63
4.5.3. Evento speechstart.....	63
4.5.4. Evento speechstop.....	63
4.5.5. Evento soundend .....	63
4.5.6. Evento audioend .....	63
4.5.7. Evento error .....	64
4.5.8. Evento Start.....	64
4.5.9. Evento End .....	64
4.6. Frases reservadas.....	64
4.6.1. "Hola" .....	64
4.6.2. "Sí" .....	64
4.6.3. "No".....	64
4.6.4. "Detener":.....	65
4.6.5. "Repetir": .....	65
4.6.6. "Nueva Búsqueda":.....	65
4.7. Diseño de interfaces.....	65
4.7.1. Diseño de interfaz pantalla inicial.....	66
4.7.2. Diseño de interfaz resultado búsqueda .....	67
4.7.3. Diseño de interfaz resultado específico.....	68

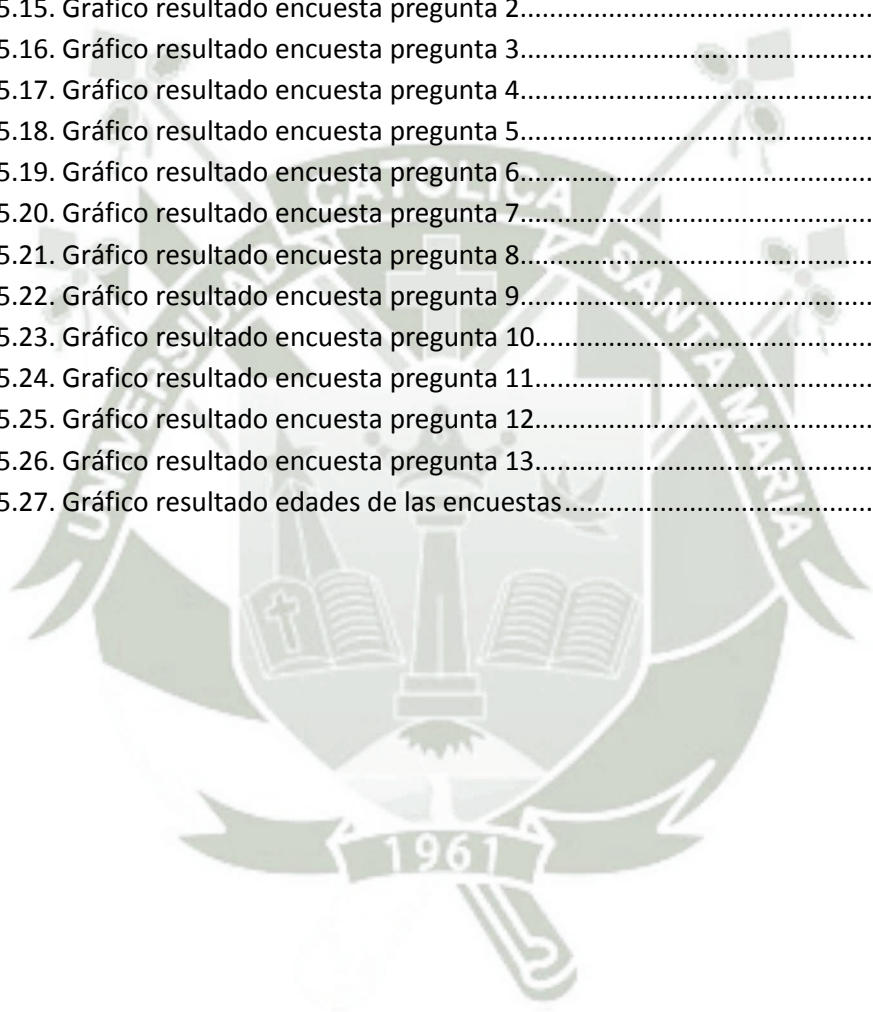
4.8. Desarrollo de solución web.....	69
4.8.1. FrontEnd.....	69
4.8.2. BackEnd.....	70
4.8.3. Librerías JS.....	72
CAPITULO 5: VALIDACIÓN DE SOLUCION WEB .....	79
5.1. Escenarios previos a puesta en producción .....	79
5.2. Publicación de solución web .....	82
5.3. Escenario posterior a puesta en producción .....	82
RESULTADOS .....	87
CONCLUSIONES .....	108
RECOMENDACIONES.....	110
BIBLIOGRAFIA.....	111
ANEXOS .....	114



## ÍNDICE DE FIGURAS

Figura 2.1. Representación de Clase UML.....	25
Figura 2.2. Representación de Actor UML .....	26
Figura 2.3. Representación de Caso de Uso UML .....	26
Figura 2.4. Representación de Asociación UML.....	27
Figura 2.5. Representación de Dependencia o instancia UML .....	27
Figura 2.6. Representación de generalización UML.....	27
Figura 2.7. Representación de Objeto/Actor UML.....	29
Figura 2.8. Representación de mensaje entre objetos UML.....	29
Figura 2.9. Representación de envió mensaje al mismo objeto UML.....	30
Figura 2.10. Representación de API TTS.....	32
Figura 2.7. Representación de Metodología Agile Scrum.....	44
Figura 3.1. Representación de la tecnología usada en el sistema web .....	51
Figura 3.2. Caso de uso búsqueda de petición del usuario.....	52
Figura 3.3. Caso de uso búsqueda de petición del usuario.....	53
Figura 3.4. Caso de uso búsqueda de petición del usuario.....	54
Figura 3.5. Caso de uso búsqueda de petición del usuario.....	55
Figura 4.1. Patrón de Diseño por capas .....	56
Figura 4.2. Patrón detallado de Diseño por capas .....	58
Figura 4.3. Arquitectura Física de la Solución Web.....	59
Figura 4.4. Diagrama de secuencia de la Solución Web.....	60
Figura 4.5. Diagrama de actividades de la Solución Web .....	61
Figura 4.6. Pantalla principal de la Solución Web .....	66
Figura 4.7. Pantalla de resultado de búsqueda de la Solución Web.....	67
Figura 4.8. Pantalla de interfaz de resultado específico de la Solución Web .....	68
Figura 4.9. Pantalla de interfaz inicial .....	69
Figura 4.10. Pantalla de interfaz de resultado búsqueda .....	70
Figura 4.11. Pantalla de interfaz de resultado específico .....	70
Figura 4.12. Fragmento de código definición de variables URL búsqueda .....	71
Figura 4.13. Fragmento de código ejecución de búsqueda y recuperación de resultado.....	71
Figura 4.14. Fragmento de código recorrer JSON para dar un formato a los resultados.....	72
Figura 4.15. Fragmento JS, función que inicia el evento de escuchar. ....	73
Figura 4.16. Fragmento JS, función que se encarga de mostrar el contenido.....	73
Figura 4.17. Fragmento JS, función que se encarga de inicializar la síntesis de voz.....	74
Figura 4.18. Fragmento JS, función que se encarga de enviar el contenido a buscar. ....	74
Figura 4.19. Fragmento JS, variables y función que define el tipo de lenguaje.....	75
Figura 4.20. Fragmento JS, variables que inicializan el reconocimiento de voz. ....	76
Figura 4.21. Fragmento JS, función de captura de error al reconocer la voz. ....	77
Figura 4.22. Fragmento JS, inicializa y confirma texto final de reconocimiento de voz. ....	78
Figura 5.1. Capacitación CEBE – Nuestra Señora del Pilar .....	81
Figura 5.2. Capacitación CERCIA – Centro de Rehabilitación para ciegos adultos.....	81
Figura 5.3. Capacitación CEPTRONuestra Señora de los Ángeles .....	82

Figura 5.4. Tiempo de carga de la aplicación .....	87
Figura 5.5. Elementos utilizan correctamente atributos HTML.....	88
Figura 5.6. Mejores prácticas en uso de atributos HTML .....	89
Figura 5.7. Uso correcto de nombres en atributos HTML.....	89
Figura 5.8. Estructuración correcta y lenguaje especificado por pagina .....	90
Figura 5.9. Uso correcto de etiquetas portabilidad en dispositivos .....	90
Figura 5.10. Tiempo de carga de la aplicación .....	91
Figura 5.11. Trafico de la aplicación en el periodo 10 Set – 10 Oct.....	91
Figura 5.12. APHAD ejecutándose en Windows .....	92
Figura 5.13. APHAD ejecutándose en iOS Sierra.....	93
Figura 5.15. Grafico resultado encuesta pregunta 2.....	95
Figura 5.16. Gráfico resultado encuesta pregunta 3.....	96
Figura 5.17. Gráfico resultado encuesta pregunta 4.....	97
Figura 5.18. Gráfico resultado encuesta pregunta 5.....	98
Figura 5.19. Gráfico resultado encuesta pregunta 6.....	99
Figura 5.20. Gráfico resultado encuesta pregunta 7.....	100
Figura 5.21. Gráfico resultado encuesta pregunta 8.....	101
Figura 5.22. Gráfico resultado encuesta pregunta 9.....	102
Figura 5.23. Gráfico resultado encuesta pregunta 10.....	103
Figura 5.24. Grafico resultado encuesta pregunta 11.....	104
Figura 5.25. Gráfico resultado encuesta pregunta 12.....	105
Figura 5.26. Gráfico resultado encuesta pregunta 13.....	106
Figura 5.27. Gráfico resultado edades de las encuestas.....	107



## ÍNDICE DE TABLAS

Tabla 1.1: Análisis de Costos APHAD.....	4
Tabla 1.2: Plan de trabajo para el proyecto de tesis.....	7
Tabla 2.1: Comparación de Asistentes.....	16
Tabla 2.2. Comparación de lenguajes web .....	35
Tabla 3.1. Privilegios de actores.....	46
Tabla 5.1. Encuestas aplicadas.....	80
Tabla 5.2. Numero de Sesiones de entrenamiento realizadas .....	83
Tabla Resultado Encuesta Pregunta 1.....	94
Tabla Resultado Encuesta Pregunta 2.....	95
Tabla Resultado Encuesta Pregunta 3.....	96
Tabla Resultado Encuesta Pregunta 4.....	97
Tabla Resultado Encuesta Pregunta 5.....	98
Tabla Resultado Encuesta Pregunta 6.....	99
Tabla Resultado Encuesta Pregunta 7.....	100
Tabla Resultado Encuesta Pregunta 8.....	101
Tabla Resultado Encuesta Pregunta 9.....	102
Tabla Resultado Encuesta Pregunta 10.....	103
Tabla Resultado Encuesta Pregunta 11.....	104
Tabla Resultado Encuesta Pregunta 12.....	105
Tabla Resultado Encuesta Pregunta 13.....	106

## RESUMEN

El presente trabajo consiste en el diseño e implementación de una aplicación informática utilizando tecnología Web, con el objetivo de incluir en el uso de tecnologías a comunidades con habilidades diferentes (como personas con visión subnormal, y personas con problemas motores) para permitirles acceder a información publicada en el amplio mundo de Internet y sobre lo cual puedan realizar consultas a través de comandos de voz.

Para llevar a cabo este proyecto, se realizó un análisis de requerimientos y funcionalidades deseadas según las necesidades de las personas con habilidades diferentes. La aplicación se desarrolló utilizando la metodología AGILE SCRUM.

Teniendo en cuenta que el usuario final no necesita contar con un equipo de alta generación para hacer uso del Asistente para Personas con Habilidades Diferentes (APHAD), además que el uso del asistente no sea complicado para el usuario final, se ha propuesto implementar una página web en la plataforma Google Chrome, ya que este navegador cuenta con una participación en el mercado de 50.95% (Mundo, 2016) y permite que el usuario final no realice instalaciones externas.

Este trabajo consta de tres etapas:

- a) En la primera etapa se realizó el análisis de los requerimientos para la construcción del asistente de voz basado en la Web. Así mismo, se realizó el diseño.
- b) En la segunda etapa se desarrolló la integración del sintetizador de voz, el reconocimiento de voz, búsqueda de resultados en Google y el reconocimiento de palabras claves reservadas para el uso del asistente de voz basado en la Web.
- c) En la tercera etapa se elaboró la implementación del asistente de voz basado en la Web realizando un muestreo con diferentes usuarios con habilidades diferentes, así como también con personas de la tercera edad.

Palabras claves: Ingeniería de Software, plataforma Web, Asistente Web, Voz sintetizada, reconocimiento de voz, búsqueda por voz, personas con visión subnormal.

## ABSTRACT

The present work consists in the design and implementation of a computer application using Web technology, in order to include in the use of technologies to communities with different abilities (such as people with low vision and people with motor problems) to enable them to access information published in the wide world of Internet and on which they can make queries through voice commands.

In order to carry out this project, an analysis of requirements and desired functionalities was carried out according to the needs of the people with different abilities. The application was developed using the AGILE SCRUM methodology.

Taking into account that the end user does not need a high-end computer to use the APHAD, the use of the wizard is not complicated for them, so it has been proposed to implement a website on the Google Chrome platform, since this browser has a market share of 50.95% (World, 2016) and this platform allows end user not to perform external facilities.

This work consists of three stages:

- a) In the first stage, the analysis of the requirements for the construction of the Web-based voice assistant was performed. Also, the design was done.

b) In the second stage, the integration of voice synthesizer, speech recognition, search results in Google and the recognition of keywords reserved for the use of the Web-based voice assistant was developed.

c) In the third stage, the implementation of the Web-based voice assistant was carried out by sampling different users with different skills, as well as with seniors.

Keywords: Software Engineering, Web platform, Web Assistant, Synthesized voice, voice recognition, voice search, people with low vision.



## INTRODUCCIÓN

Con el transcurso del tiempo, la tecnología ha ido avanzando a pasos agigantados haciendo su uso más fácil, pero en nuestro país y el mundo hay personas que sufren de discapacidad, como por ejemplo las personas que no pueden ver, escuchar, hablar, etc. Existen empresas tecnológicas que no han tomado en cuenta incluir a estas personas en el mundo tecnológico. Por consiguiente, hemos tomado la decisión de crear un asistente de búsqueda por voz, que permitirá que las personas con dificultades de visión, motoras puedan introducirse en el mundo tecnológico orientado en la búsqueda de información por medio de la web, creando así una ayuda social para estas personas, poniendo a su alcance los nuevos avances. En el contexto mundial actual, la gran fuente de información proviene de Internet, lo cual nos permite realizar búsquedas de diferentes hechos, acontecimientos, fuentes de información globalizada, permitiendo enriquecer nuestros conocimientos. La metodología aplicada para el presente trabajo fue de lógica deductiva la cual nos permite desarrollar principios nuevos a partir de otros principios ya conocidos y así poder resolver el problema expuesto. Con ayuda de este Asistente para Personas con Habilidades Diferentes (APHAD) estas personas que sufren de alguna discapacidad visual o motora pueden informarse utilizando la voz y recibiendo la información por medio del audio, todo esto integrado al motor de búsqueda de Google, teniendo un amplio campo de búsqueda. Actualmente en la ciudad de Arequipa estas personas ya cuentan con acceso al asistente y así mismo pueden hacer uso del asistente sin ninguna dificultad, en las sesiones

realizadas a las instituciones de personas ciegas los resultados son exitosos  
obteniendo una grande acogida por parte de los usuarios finales.



## CAPITULO I: PLANEAMIENTO TEÓRICO

### 1.1. Título del proyecto

Diseño de asistente de voz basado en web para personas con visión subnormal.

### 1.2. Descripción del problema

Con el transcurso del tiempo la tecnología ha ido avanzando a pasos agigantados haciendo su uso más fácil, pero en nuestro país y el mundo hay personas que sufren de discapacidad, como por ejemplo las personas que no pueden ver, escuchar, hablar, etc. Existen empresas tecnológicas que no han tomado en cuenta incluir a estas personas en el mundo tecnológico orientado en la búsqueda de información por medio de la web. Debido a ello se decidió crear un asistente de búsqueda por voz basada en web que permitirá a las personas con discapacidad visual o motoras introducirse en el mundo tecnológico, ya que en el contexto mundial actual la gran fuente de información proviene de Internet.

La Internet nos permite realizar búsquedas de diferentes hechos, acontecimientos y fuentes de información, enriqueciendo nuestros conocimientos. Con ayuda de este Asistente para Personas con Habilidades Diferentes (APHAD) se logrará que estas personas no tengan dificultad en su uso, presentando una interfaz amigable para el usuario final que interactuará por medio de la voz y audio. En la actualidad contamos con aplicaciones denominadas asistentes personales, que en su mayoría están destinados para dispositivos móviles, que a través de la voz interactúa con el dispositivo móvil sin necesidad de utilizarlo

físicamente. También es de conocimiento que las empresas propietarias de estas aplicaciones están implementando dichos asistentes para dispositivos de escritorio como pcs, y laptops. Estas aplicaciones realizan búsquedas de información y muestran por pantalla los resultados, pero la limitante es que no lee los resultados mostrados, y una vez realizada la búsqueda termina la iteración con el usuario, es por eso que nosotros proponemos el Asistente para Personas con Habilidades Diferentes (APHAD) incorporando mayor transparencia y fluidez de la interacción con el usuario.

### **1.3. Delimitaciones y definición del problema**

#### **1.3.1. Delimitaciones**

##### **a. Delimitación espacial**

El presente proyecto se desarrolla en la ciudad de Arequipa para ser distribuida gratuitamente a la comunidad de habla hispana por medio de Google Chrome.

##### **b. Delimitación temporal**

El trabajo tendría como inicio junio del 2016 y culminaría en agosto del 2017

##### **c. Delimitación Social**

Está orientada a mejorar la calidad de vida y promover la inclusión social en el aspecto tecnológico orientado en la búsqueda de información por medio de la web, para personas con visión subnormal y problemas motores

##### **d. Delimitación conceptual**

Diseño e implementación de una solución web.

### **1.3.2. Definición del Problema**

Actualmente las aplicaciones de voz no están orientadas para personas con visión subnormal y basada en tecnología Web, las cuales faciliten la interacción del usuario con la aplicación al realizar búsquedas de información (noticias, definiciones, conceptos, etc.) en internet.

### **1.4. Formulación del Problema**

#### **1.4.1. Problema Principal**

No existe un asistente de voz basado en web que ayude a un usuario con visión subnormal, a informarse de los últimos acontecimientos o informarse de algún tema en específico de una manera fácil y rápida.

### **1.5. Objetivos de Investigación**

#### **1.5.1. Objetivo General**

Diseñar e implementar un asistente de voz basado en Web con el uso de Tecnologías de Información y Comunicación que nos permita incluir al mundo de la tecnología orientado en la búsqueda de información por medio de la web a personas con visión subnormal.

#### **1.5.2. Objetivos Específicos**

- a. Establecer una aplicación web con integración de sintetizador de voz, reconocimiento de voz y búsquedas en el motor de Google.
- b. Diseñar una interfaz intuitiva y sencilla para el asistente web que sea fácil de entender para el usuario final.
- c. Desarrollar e implementar el asistente de voz basado en web, creando una buena comunicación con las personas con visión subnormal.

- d. Validar que el asistente de voz basado en web funcione correctamente.
- e. Lograr que las personas con visión subnormal se integren en el mundo tecnológico orientado en la búsqueda de información por medio de la web.

## 1.6. Viabilidad de la Investigación

### 1.6.1. Económica

Los recursos económicos necesarios para solventar el presente trabajo de investigación, diseño e implementación son asumidos por los tesisistas se detalla el análisis de costos de este proyecto en la Tabla 1.1.

Tabla 1.1: Análisis de Costos APHAD

Costos Proyecto APHAD							
N°	Item	Cant.	Unidad	Precio	1 año	2 año (Proyección)	
<b>Recurso Humano</b>							
1	Desarrollador 1	2920	Horas	S/ 1.25	S/ 3,650.00	S/ 3,650.00	
2	Desarrollador 2	2920	Horas	S/ 1.25	S/ 3,650.00	S/ 3,650.00	
<b>Recurso Tecnológico</b>							
3	Laptop	2	Unidad	S/ 2,000.00	S/ 4,000.00	S/ -	
4	Hosting	12	Meses	S/ 25.00	S/ 300.00	S/ 300.00	
5	Google Query	1000	Unidad	S/ 0.02	S/ 16.25	S/ 16.25	
<b>Recursos Educativos</b>							
6	Especialización lenguaje Web	9	Meses	S/ 90.00	S/ 810.00	S/ -	
<b>Servicios</b>							
7	Luz	2920	Horas	S/ 0.52	S/ 1,518.40	S/ 1,518.40	
8	Internet	2920	Horas	S/ 0.06	S/ 172.28	S/ 172.28	
<b>Total</b>					S/ 14,116.93	S/ 9,306.93	

### 1.6.2. Técnica

Se utilizarán 2 equipos informáticos para el desarrollo del asistente web.

En una etapa inicial se mantendrá un servicio local. Además, se

utilizarán micrófonos y audífonos. Para esta etapa de las pruebas se utilizará un servidor público para exponer el asistente web en internet.

### **1.6.3. Operativa**

Se hizo uso de Bibliotecas Virtuales (tesis, papers), conferencias, estudios en Instituciones (Institutos), Internet e infraestructura propia, programación, para llevar a cabo este asistente.

## **1.7. Justificación e Importancia de la Investigación**

### **1.7.1. Justificación**

Es importante incluir a las personas con visión subnormal para que puedan informarse sin necesidad de ayuda de otras personas, y a incluirlas al mundo tecnológico orientado en la búsqueda de información por medio de la web, que cada día va avanzando más.

### **1.7.2. Importancia**

Permitir que las personas con habilidades diferentes puedan obtener información sobre los acontecimientos del mundo actual o conceptos que estén buscando.

### **1.7.3. Limitaciones de la investigación**

Este aporte está orientado a personas con visión subnormal y con problemas motores los cuales pueden oír y hablar correctamente para poder tener una buena interacción con la aplicación.

## **1.8. Área, línea, tipo y nivel de investigación**

### **1.8.1. Área de Investigación**

Ingeniería de Software.

### **1.8.2. Línea de Investigación**

Ingeniería en Tecnología Web.

### **1.8.3. Tipo de Investigación**

Aplicada

#### **1.8.4. Nivel de Investigación**

Experimental ya que se quiere ayudar a las personas con visión subnormal y no se tienen casos de estudios enfocados en este ámbito.

### **1.9. Método y diseño de Investigación**

#### **1.9.1. Método de Investigación**

Investigación adaptativa basada en metodología lógica deductiva que constan en encontrar principios desconocidos de los principios ya conocidos.

#### **1.9.2. Forma de tratamiento de los datos**

Para el tratamiento de datos recopilada de Internet y proporcionada por el usuario se utilizarán:

##### **a) Sintetizador de Voz**

Es un API (Application Programming Interface) desarrollado en JavaScript la cual contiene un diccionario fonético de las palabras en español para que puedan ser reproducidas al momento de que se le muestre los resultados al usuario.

##### **b) Reconocedor de Voz**

Es un API (Application Programming Interface) proporcionada por Google para desarrolladores la cual contiene un diccionario fonético de diferentes lenguajes el cual ayuda a realizar comparaciones de las palabras dictadas por el usuario al computador.

### **1.10. Cobertura del Estudio**

#### **1.10.1. Universo**

Personas con visión subnormal y con problemas motores de habla hispana que tengan acceso a Internet.

### 1.10.2. Plan de Trabajo

Cronograma de Plan de trabajo con actividades y tareas a realizar durante el desarrollo de este proyecto en la Tabla 1.2.

Tabla 1.2: Plan de trabajo para el proyecto de tesis

N°	Actividad	Duración	Inicio	Fin	Acción
1	Investigación del tema	40 días	2 de mayo 2016	24 de junio 2016	Culminada
2	Definición del problema	3 días	24 de junio 2016	28 de junio 2016	Culminada
3	Toma de requerimiento funcionales	5 días	29 de junio 2016	5 de julio 2016	Culminada
4	Toma de requerimientos no funcionales	3 días	6 de julio 2016	8 de julio 2016	Culminada
5	Listar requerimientos	7 días	11 de julio 2016	19 de julio 2016	Culminada
6	Elaborar diagramas	2 días	20 de julio 2016	21 de julio 2016	Culminada
7	Elaborar casos de uso	2 días	22 de julio 2016	25 de julio 2016	Culminada
8	Elaborar diagrama de secuencia	2 días	26 de julio 2016	27 de julio 2016	Culminada
9	Elaborar diagrama de actividades	2 días	28 de julio 2016	29 de julio 2016	Culminada
10	Diseño de interfaz	15 días	1 de agosto	19 de agosto	Culminada

			2016	2016	
<b>11</b>	Búsqueda de librerías JS	20 días	22 de agosto 2016	16 de setiembre 2016	Culminada
<b>12</b>	Implementación de reconocimiento de voz	5 días	19 de setiembre 2016	23 de setiembre 2016	Culminada
<b>13</b>	Implementación de sintetizador de voz	10 días	26 de setiembre 2016	7 de octubre 2016	Culminada
<b>14</b>	Implementación de búsqueda por voz	10 días	10 de octubre 2016	21 de octubre 2016	Culminada
<b>15</b>	Integración de búsqueda con motores de Google	15 días	24 de octubre 2016	11 de noviembre 2016	Culminada
<b>16</b>	Integración entre reconocimiento de voz y sintetizador de voz	20 días	14 de noviembre 2016	9 de diciembre 2016	Culminada
<b>17</b>	Integración entre reconocimiento de voz, sintetizador de voz y motor de búsqueda	20 días	12 de diciembre 2016	6 de enero 2017	Culminada
<b>18</b>	Diseño de asistente APHAD	2 días	9 de enero 2017	10 de enero 2017	Culminada

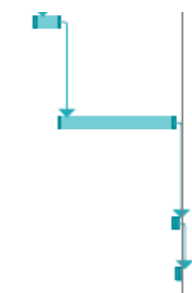
<b>19</b>	Recuperación y tratamiento de resultados de búsqueda	20 días	11 de enero 2017	7 de febrero 2017	Culminada
<b>20</b>	Demostración de requerimientos	1 días	8 de febrero 2017	8 de febrero 2017	Culminada
<b>21</b>	Retrospectiva de la demostración de requerimientos	1 días	9 de febrero 2017	9 de febrero 2017	Culminada
<b>22</b>	Iteración de demostración y retrospectiva	20 días	10 de febrero 2017	9 de marzo 2017	Culminada
<b>23</b>	Desarrollo del informe de tesis	60 días	10 de marzo 2017	1 de junio 2017	Culminada
<b>24</b>	Revisar formato APA	3 días	2 de junio 2017	6 de junio 2017	Culminada
<b>25</b>	Anexos y documentación del Asistente APHAD	5 días	6 de junio 2017	10 de junio 2017	Culminada
<b>26</b>	Testeo del asistente APHAD	2 días	7 de junio 2017	8 de junio 2017	Culminada
<b>27</b>	Publicación de asistente APHAD a Internet	1 días	8 de junio 2017	8 de junio 2017	Culminada
<b>28</b>	Elaboración de Encuesta	3 días	8 de junio 2017	10 de junio 2017	Culminada
<b>29</b>	Prueba de Aplicación con personas adultos mayores	5 días	12 de junio 2017	16 de junio 2017	Culminada
<b>30</b>	Prueba de Asistente	10 días	21 de junio	21 de julio	Culminada

	APHAD con personas con visión subnormal		2017	2017	
<b>31</b>	Presentación de borrador de tesis	1 día	1 de agosto 2017	1 de agosto 2017	Culminada
<b>32</b>	Corrección de observaciones	1 día	4 de agosto 2017	4 de agosto 2017	Culminada





29		Prueba de Aplicación con Personas adultas mayores	5 días	jue 22/06/17	mié 28/06/17	28
30		Prueba de Asistente APHAD con personas con visión subnormal	23 días	jue 29/06/17	lun 31/07/17	29
31		Presentación de Borrador de Tesis	1 día	mar 1/08/17	mar 1/08/17	30
32		Corrección de Observaciones	1 día	mié 2/08/17	mié 2/08/17	31



Proyecto: Proyecto1  
 Fecha: jue 16/11/17

Tarea		Resumen inactivo		Tareas externas	
División		Tarea manual		Hito externo	
Hito		solo duración		Fecha límite	
Resumen		Informe de resumen manual		Progreso	
Resumen del proyecto		Resumen manual		Progreso manual	
Tarea inactiva		solo el comienzo			
Hito inactivo		solo fin			

Proyecto: Proyecto1  
 Fecha: jue 16/11/17

Tarea		Resumen inactivo		Tareas externas	
División		Tarea manual		Hito externo	
Hito		solo duración		Fecha límite	
Resumen		Informe de resumen manual		Progreso	
Resumen del proyecto		Resumen manual		Progreso manual	
Tarea inactiva		solo el comienzo			
Hito inactivo		solo fin			



## CAPITULO 2: MARCO TEORICO

### 2.1. Estado del Arte

A través de estos años, la tecnología ha ido avanzando en diferentes tareas. Los asistentes personales es uno de estos avances tecnológicos que nos facilita el trabajo en grandes o pequeñas tareas que se realizan cotidianamente, sin embargo, se dejó de lado a las personas con discapacidades (físicas, motoras) las cuales puedan aprovechar este tipo de tecnología. Es por esto que nace este proyecto Asistente Para Personas con Habilidades Diferentes (APHAD). Se realizó este asistente basado en tecnología web ya que las persona con visión subnormal tienen un fácil acceso a ordenadores y portátiles a diferencia de un smartphone que es complicado para estas personas aprender las funcionalidades y que cuentan con asistentes nativos como Siri, Cortana, Sherpa y Google.

Existe en el mercado una plataforma de web y voz, lo cual creó una herramienta que genera diálogos llamada Application Generation Platform, (Plataforma de generación de aplicaciones AGP) que es un conjunto integrado de Asistente de descripción de la aplicación (ADA). Esta herramienta define las características de la aplicación, tales como, idiomas, bibliotecas, modalidades, los cuales se incorporan en esta investigación. (D'Haro, 2005).

Se considera para esta investigación el uso de una API de sintetizador de voz ("La síntesis de voz es la generación de voz artificial a partir de ver un

texto escrito” (Flores, 2001)) que permite procesar los textos en audio. En la actualidad existe una herramienta de software que permite hacer llamadas de telefonía móvil mediante el uso de síntesis de voz. La concatenación de difonos específicamente se usa para generar una voz artificial en una computadora de un texto de entrada para reproducirla en un dispositivo móvil. (Fuentes, 2012). El sintetizador produce una voz sintética a partir de la transformación de una entrada de texto. En otras palabras, se encuentra la representación sonora de un texto dado. La síntesis vocal es la creación de una voz sintetizada, donde lo que se quiere es que una máquina pueda ser capaz de expresarse, emitiendo sonidos que se puedan entender como las palabras u oraciones. Esto también lleva a la comprensión total del proceso del habla, implicando la interacción de disciplinas en las que está incluida la fisiología del órgano humano, así como la lingüística. (Andrés, 2011). Así existiendo diversos sintetizadores como Acapela group, el API Google en JavaScript y vozMe, etc.

Grupo Acapela es un proveedor de soluciones de voz que crea sintetizadores que leen, informan explican presentan, educan, etc con el fin de brindar este servicio. Este servicio está desarrollado en html5 (Acapela, 2016), pero no optamos este servicio ya que el costo es muy elevado.

VozMe es un sintetizador de voz que está publicado en la web que genera la síntesis de voz por medio de un mp3, contiene dos opciones de voces que es la voz femenina y la voz masculina, este sintetizador está

desarrollado en PHP, la debilidad de este sintetizador es que la voz es muy robotizada y rápida que no se deja entender con claridad es por ello que no optamos por esta opción.

El API de Google basado en JavaScript es un servicio que establece la transformación del texto a voz, se optó por este servicio, ya que es gratuito y se puede controlar la voz, la velocidad cuando el sintetizador empiece a convertir de texto a voz y además te da la libertad de modificar el código y realizar mejoras.

Para poder llevar el reconocimiento de voz se debe realizar un proceso que reconozca la entrada de voz del usuario, con el objetivo de extraer la información acústica relevante para el procesamiento al texto (Felipe, 2008), existiendo actualmente diversos reconocedores de voz como VozMe, Acapela y la API de Google.

Vozme es un reconocedor de voz que está publicado en la web, que captura audio por medio de un micrófono y posteriormente realiza el proceso a texto, su debilidad de Vozme es que se le dificulta reconocer la voz en español creando conflicto al decir una oración.

API de google es un servicio que nos permite realizar una transcripción del audio que se recibe por medio de un micrófono en nuestra aplicación y realiza la transcripción de audio a texto.

Adicionalmente detallamos en la tabla 2.1 un comparativo con los actuales asistentes existentes en el Mercado

Tabla 2.1: Comparación de Asistentes

	Siri	Cortana	Sherpa	Google	APHAD
<b>Escucha Voz</b>	Escucha al usuario hablar, convirtiendo la voz a texto.	Escucha al usuario hablar, convirtiendo la voz a texto.	Escucha al usuario hablar, convirtiendo la voz a texto.	Escucha al usuario hablar, convirtiendo la voz a texto.	Escucha al usuario hablar, convirtiendo la voz a texto.
<b>Procesa Información</b>	Convierte la voz a texto para hacer la búsqueda.	Convierte la voz a texto para hacer la búsqueda.	Convierte la voz a texto para hacer la búsqueda.	Convierte la voz a texto para hacer la búsqueda.	Convierte la voz a texto para hacer la búsqueda.
<b>Búsqueda en Google</b>	Busca el texto en el motor de búsqueda de Google.	Busca el texto en el motor de búsqueda de Google.	Busca el texto en el motor de búsqueda de Google.	Busca el texto en el motor de búsqueda de Google.	Busca el texto en el motor de búsqueda de Google.
<b>Palabras Claves</b>	Contiene palabras claves para necesidades del móvil como	Contiene palabras claves para necesidades del móvil como	No contiene palabras claves.	Contiene palabras claves para necesidades del móvil como	Contiene palabras claves para la iteración con el asistente.

	llamadas, mensajes, etc.	llamadas, mensajes, etc.		llamadas, mensajes, etc.	
<b>Responde</b>	Responde en texto la petición.	Responde en texto la petición.	Responde en texto la petición.	Responde con voz cuando son preguntas concretas, pero cuando la petición no es concreta responde en texto mostrando los links de manera visual.	Responde con voz las respuestas de acuerdo con la búsqueda.

Fuente: Propia

## 2.2. Ingeniera de Software

El contexto actual y la rápida evolución tecnológica ha puesto en acción nuevos servicios y esquemas de trabajos para organizaciones la cual permite una mejora continua en la calidad de sus servicios para los clientes y estas condiciones genera una demanda permanente a la Ingeniería de Software. “El desarrollo de software es considerado tanto como un arte y una ciencia” (Lawrence P., 2002). Como ciencia nos permite medir, evaluar y estimar el proceso de desarrollo de manera continua y repetitiva. Como

arte la ingeniería de software requiere de resultados que representen ingenio y originalidad para presentar diseños que satisfagan al cliente.

“El desarrollo de software se ha convertido en una actividad tan especializada, que ha originado un desplazamiento de la actividad hacia compañías cuya actividad fundamental es el desarrollo de software, permitiendo que las empresas de producción o servicio se dediquen a las actividades orientadas hacia sus objetivos organizacionales. Este fenómeno creó una nueva actividad económica alrededor de la industria del software que hoy se considera promisoria para los países latinoamericanos. Sin embargo, los estudios actuales coinciden en afirmar que las empresas de software latinoamericanas están en desventaja frente a los niveles de competitividad exigidos a nivel internacional “ (Mayer, Bunge, 2004) , (Valdes, 2004)

El proceso, el producto, el método, las personas y la tecnología son los elementos básicos involucrados en el desarrollo de software, los cuales puede ser interrelacionados de diversas maneras. Pressman, por ejemplo, establece una relación de jerarquía de dichos elementos al cual denomina tecnología multicapa, que parte del enfoque de calidad sobre el cual se soporta el proceso, el método y las herramientas (Pressman, 2006). Un adecuado proceso de formación en la enseñanza de la ingeniería de software debe considerar de manera natural las interrelaciones que se

sucedan entre dichos elementos que, de acuerdo a Paulish, representan los “factores controlables en la mejora de la calidad del software y el desempeño organizacional” (Anaya, 2006).

### 2.3. Aplicaciones Web

Los primeros pasos de los sitios web contenían un conjunto de archivos de hipertexto que vinculados representaban la información agregando texto y gráficas limitadas. Al pasar el tiempo y con el avance de esta tecnología, el aumento del uso de HTML por medio de herramientas de desarrollo (XML, Java) permitió a los especialistas de la web una mejor capacidad de cómputo con información más elaborada. Este fue el inicio de los sistemas y aplicaciones basados en la web denominadas formas colectiva como webapps. Actualmente, las webapps son herramientas sofisticadas de cómputo que proporcionan funciones aisladas al usuario final y también se han integrado con bases de datos y aplicaciones de negocios.

Las webapps son una de varias categorías distintas de software. Powell indica que las aplicaciones y sistemas basados en web “involucran una mezcla entre las publicaciones impresas y el desarrollo de software, entre la mercadotecnia y la computación, entre las comunicaciones internas y las relaciones exteriores, y entre el arte y la tecnología” (Pressman, 2010). En su mayoría de webapps contienen los siguientes atributos:

- **Uso intensivo de redes.** Una webapp se encuentra en una red y esta tiene que atender las necesidades de una gran comunidad de usuarios. La red permite acceso y comunicación mundial por medio del internet.
- **Concurrencia.** A la webapp pueden acceder un gran número de usuarios en un mismo momento en muchos casos, los patrones de uso entre los usuarios finales son aleatorios y varían entre sí.
- **Carga impredecible.** El número de usuarios que ingresan a una webapp cambia constantemente de magnitud de un día a otro.
- **Rendimiento.** Si el cliente de la webapp espera demasiado para ingresar, el procesamiento por parte del servidor, el despliegue del lado del cliente, el usuario quizá decida irse a otra webapp con mejores tiempos de respuesta.
- **Disponibilidad.** Es razonable esperar una disponibilidad de 100% pero no es frecuente que los usuarios de webapps populares demanden acceso de 24/7 los 365 días del año.
- **Orientadas a los datos.** Como principal función de muchas webapp es el uso de hipermedios que se deben presentar al usuario final contenido en forma de texto, gráficas, media, audio y video. Las webapps se utilizan para acceder a información que existe en bases de datos que no están en parte integral del ambiente basado en web (por ejemplo, e-commerce o aplicaciones financieras).

- **Contenido sensible.** La estética del contenido constituye un rasgo importante de la calidad y naturaleza de una webapp.
- **Evolución continúa.** La diferencia del software de aplicación convencional a las aplicaciones webapp es que evoluciona en el transcurso de una serie de etapas planeadas y cronológicamente separadas, las aplicaciones web avanzan de forma continua. No es raro que el contenido de ciertas webapp se actualicen en cada momento o que su contenido se recalculen en cada solicitud.
- **Inmediatez.** Hay necesidad de que el software llegue con rapidez al mercado una característica en muchos despliegues de aplicación, es muy frecuente que las webapps contengan plazos de pocos días o semanas para llegar al mercado.
- **Seguridad.** Ya que las webapps se encuentran con el acceso a una red interconectada, es difícil limitar la población de usuarios finales que acceden a la aplicación publicada. Para así proteger el contenido sensible y tener modos seguros de transmisión de los datos, deben también implementarse medidas estrictas de seguridad por medio de la infraestructura de apoyo de una webapp y dentro de una aplicación.
- **Estética.** El atractivo de una webapp es muy importante y por eso se debe tener cuidado en su apariencia y percepción. En el momento que se ha diseñado una aplicación para ser comercializada o vendida como productos o ideas, la estética tiene un valor alto.

## 2.4. Lenguaje Unificado de Modelado

El lenguaje unificado de modelado (UML) se utiliza para el modelado de sistemas de software este lenguaje es el más conocido y utilizado en la actualidad; UML está respaldado por el Object Management Group (OMG). UML es un lenguaje que permite visualizar, especificar, construir y documentar un sistema. UML contiene un estándar que permite describir un modelo del sistema que incluye aspectos conceptuales como procesos, funciones del sistema, aspectos concretos, expresiones de lenguajes de programación, bases de datos esquematizadas.

Es necesario poner en énfasis que el lenguaje UML es un lenguaje de modelado para describir métodos y procesos de un sistema además de documentar y construir.

### 2.4.1. Objetivos del lenguaje UML

Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones (Enrique, 2002):

- **Visualizar:** Permite al usuario ver lo que se quiere lograr con el sistema, dando facilidad en la comunicación y una representación gráfica del problema propuesto y su solución al problema.
- **Especificar:** Se puede construir modelos teniendo una estructura precisa, teniendo en cuenta las decisiones de análisis, diseño e implementación.

- **Construir:** UML no es un lenguaje de programación, pero nos permite el desarrollo de plantillas que se pueden aplicar a diferentes lenguajes de programación.
- **Documentar:** La documentación se puede realizar para cubrir la arquitectura y los detalles de un sistema. Se puede expresar requisitos y pruebas, así mismo el modelado de las actividades y gestión de las versiones del proyecto.

#### 2.4.2. Semántica y notación

UML tiene como una meta principal avanzar en el estado de la integración proporcionando herramientas de interoperabilidad que ayudara en el modelado visual de objetos. Para lograr un intercambio de modelos de información exitoso entre herramientas que se va a utilizar, se requirió definir una semántica y una notación. (Cornejo, 2016)

La parte gráfica es la notación la cual se ve en los modelos y representa la sintaxis del lenguaje de modelado. Se puede tomar como ejemplo, la definición de la notación de un diagrama de clases como se representan los elementos y conceptos. una clase, una asociación y una multiplicidad. UML debe cubrir con la semántica y con la notación para que un proveedor diga que cumple con lo requerido. (Enrique, 2002)

El lenguaje tiene múltiples herramientas que te ayudan a lograr una especificación determinante del modelo, pero en este caso se realizara de forma simplificada sobre:

- Modelamiento de Clases
- Casos de Uso
- Diagrama de Interacción

### 2.4.3. Diagramas que ofrece UML

### 2.5. Modelamiento de Clases

El diagrama de clases nos permite visualizar las relaciones entre clases que engloban el sistema, estas pueden ser asociativas, de herencia, de uso y de acontecimiento.

El diagrama de clases está formado por los siguientes elementos:

**Clase:** Atributos, métodos y visibilidad.

**Relaciones:** Herencia, composición, agregación, asociación y uso.

Elementos

Clase

Encapsula la información de un objeto y se conoce como una unidad básica (Figura 2.1 un objeto es una instancia de una clase). Con la cual nos permite modelar el entorno en estudio (un carro, un edificio, una cuenta bancaria, etc.).

En UML, rectángulo que posee tres divisiones es la representación gráfica de una clase:

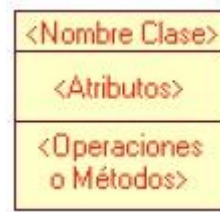


Figura 2.1. Representación de Clase UML

Fuente: (Cornejo, 2016).

- **Superior:** Denominación de la clase
- **Intermedio:** Contiene las variables de instancia que caracterizan a la clase (atributos: private, protected o public).
- **Inferior:** Se especifican los métodos u operaciones, los que interactúan con su entorno (tomar en cuenta la visibilidad: private, protected o public). (Cornejo, 2016)

## 2.6. Casos de Uso

Para representar como un cliente (Figura 2.2 Actor) opera con el sistema en desarrollo se utiliza un diagrama de caso de uso, además se puede agregar forma, tipo y orden en como los elementos interactúan (casos de uso).

Los elementos son:

- Actor.
- Casos de Uso.
- Relaciones de Uso, Herencia y Comunicación.

## Elementos

### Actor:

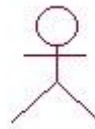


Figura 2.2. Representación de Actor UML

Fuente: (Cornejo, 2016).

Un actor es un rol que un usuario juega con respecto al sistema. Se puede indicar que un Actor no siempre representa a una persona en particular, también puede ser una labor que realiza frente al sistema.

### Caso de Uso:

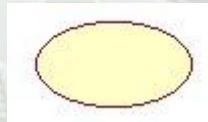


Figura 2.3. Representación de Caso de Uso UML

Fuente: (Rumbaugh, 2007).

Un caso de uso es una tarea específica (Figura 2.3) que es realizada tras una orden de algún objeto externo, por una petición de un actor o bien de la invocación de otro caso de uso.

## Relaciones:

### Asociación



Figura 2.4. Representación de Asociación UML

Fuente: (Rumbaugh, 2007).

Esta relación se considera básica, que indica la invocación de un actor o a un caso de uso hacia otra operación (caso de uso). La relación se representa con una flecha simple como se muestra en la figura 2.4.

### Dependencia o Instanciación



Figura 2.5. Representación de Dependencia o instancia UML

Fuente: (Cornejo, 2016).

Esta forma es muy particular en relación entre clases, la cual una clase depende de otra, se crea. La relación se representa con una flecha punteada como se muestra en la figura 2.5.

### Generalización



Figura 2.6. Representación de generalización UML

Fuente: (Fowler, 2000).

Esta relación es la más utilizada, la cual cumple una doble función dependiendo del estereotipo, la cual puede ser de uso (<<uses>>) o de herencia (<<extends>>) como se muestra en la figura 2.6.

Está orientada solo para casos de uso

- **extends:** Se utiliza cuando un caso de uso es similar a otras características.
- **uses:** Se utiliza cuando se tiene un grupo de características que se parecen en más de un caso de uso y no se desea continuar con la descripción copiada de la característica.

Cabe mencionar que se tiene un mismo paradigma en diseño y modelamiento de clases, en donde se puede usar o heredar. (Cornejo, 2016)

## 2.7. Diagrama de interacción

El diagrama de interacción denota la forma en como un Cliente/Actor u Objetos/Clases pueden comunicarse entre sí, en llamar a un evento. Esto también implica seguir la secuencia de llamadas, de la cual se obtienen las responsabilidades.

Este diagrama puede ser representado en dos partes, desde los diferentes diagramas como el Diagrama Estático de Clases o el de Casos de Uso.

Los componentes son:

- Un Objeto o Actor.

- Mensaje de un objeto a otro objeto.
- Mensaje de un objeto a sí mismo.

## Elementos

### Objeto/Actor:



Figura 2.7. Representación de Objeto/Actor UML

Fuente: (Scott, 2003).

El rectángulo denota una instancia de un Objeto, y la línea punteada denota las llamadas a métodos/tareas del objeto como se muestra en la figura 2.7.

### Mensaje a Otro Objeto:

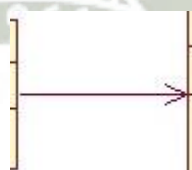


Figura 2.8. Representación de mensaje entre objetos UML

Fuente: (Scott, 2003).

Es representada por una flecha entre un objeto y otro, la representación de la llamada de un método (operación) de un objeto en particular como se muestra en a figura 2.8.

### **Mensaje al Mismo Objeto:**

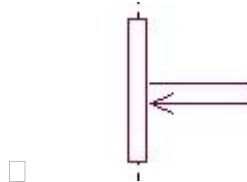


Figura 2.9. Representación de envió mensaje al mismo objeto UML

Fuente: (Cornejo, 2016).

Pueden realizarse llamadas a métodos de objetos externos, también es posible ver llamadas a métodos/tareas desde un mismo objeto en estudio.

## **2.8. Sintetizador de Voz**

Un sintetizador de voz es un dispositivo que se encarga de la producción artificial del habla humana. Un sintetizador de voz puede convertir el texto a voz.

Los sintetizadores de voz son considerados periféricos de salida que generalmente suelen incluir un microprocesador, memoria ROM con programas pre instalados y datos, un conversor Digital/Analógica, un amplificador de audiofrecuencia y altavoz. En su mayoría los dispositivos sintetizadores de voz tienen memorizados digitalmente cada uno de los fonemas/palabras que son capaces de emitir. En el sintetizador los datos

que recibe procedente del ordenador corresponden a la identificación de los fonemas o palabras a emitir. Una vez que se analiza el dato ingresado, se activa una rutina encargada de generar el sonido correspondiente. Para lograr la más completa claridad en el habla, lo adecuado es almacenar pronunciaciones de palabras completas. Los sintetizadores de habla son muy útiles para ser usados por aquellas personas con discapacidades; por ejemplo, se utiliza un sintetizador de habla en programas de asistencia como los lectores de pantalla.

### **2.9. Diagrama de un sistema Text-to-Speech (TTS)**

Actualmente varias compañías ofrecen APIs TTS a los consumidores/usuarios para acelerar el desarrollo de nuevas aplicaciones usando la tecnología TTS. Dichas compañías que ofrecen APIs TTS incluyen a AT&T, IVONA, Neospeech, Readspeaker y YAKiToMe!. Su principal mercado está enfocado para el desarrollo de aplicaciones para móviles, el sistema operativo Android es el que ha ofrecido una API TTS por un largo tiempo. Seguidamente con iOS7, Apple ha comenzado a ofrecer de igual manera una API TTS representada en la figura 2.10.

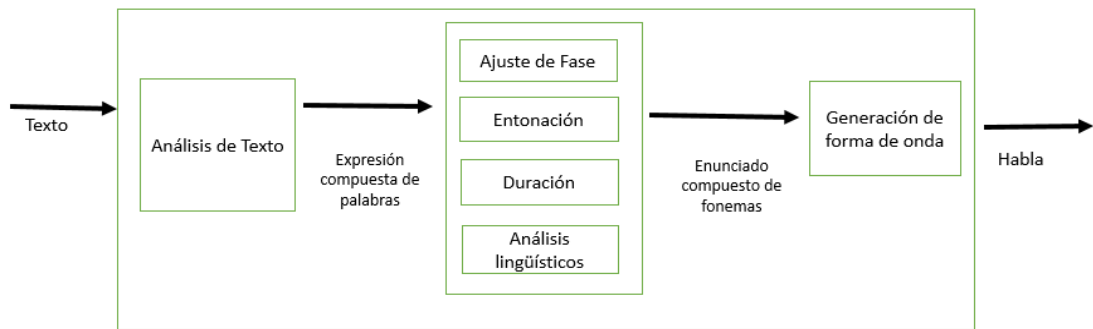


Figura 2.10. Representación de API TTS

Fuente: (Florian Hinterleitner, 2015)

Vozme se trata de un servicio online de forma gratuito que nos permite subir o escribir archivos en formato de texto y convertir a archivos de audio. Posteriormente se puede descargar en formato \*.mp3 para su edición. Su uso puede ser muy útil a la hora de establecer enunciados de problemas o definiciones de contenidos además Vozme permite utilizar su API para agregarlos en diferentes páginas web para que puedas enviar texto y la API lea el texto proporcionado.

## 2.10. Reconocimiento de Voz

El reconocimiento automático del habla (RAH) es una disciplina de la inteligencia artificial (IA) que tiene como principal objetivo permitir y asegurar la comunicación hablada entre computadoras y humanos. El principal problema en un sistema de este tipo es hacer aportar un conjunto de información que provenga de diversas fuentes de conocimiento (acústica, fonológica, léxica, semántica, etc.), en presencia de ambigüedades según el lenguaje, errores inevitables e incertidumbres para

obtener una interpretación correcta del mensaje recepcionado por el ordenador.

El sistema de reconocimiento de voz es la herramienta computacional que procesa la señal de voz emitida por la persona y reconoce la información recepcionada, convirtiéndola en texto. Intervienen diversas disciplinas para su desarrollo, como la fisiología, la lingüística, el procesamiento de señales, la inteligencia artificial y la ciencia de la computación.

Los sistemas de reconocimiento de voz pueden clasificarse según los siguientes criterios (Tordera Yllescas, 2011)

- **Entrenabilidad:** Identifica si un sistema necesita previa preparación o entrenamiento previo antes del uso.
- **Dependencia del hablante:** Identifica si el sistema necesita entrenamiento para cada usuario que lo utilice o es autónomo.
- **Continuidad:** Identifica si el sistema puede reconocer voz continua como un dictado o se debe hacer pausas.
- **Robustez:** Identifica si el sistema está diseñado para el uso de ruidos de fondo o puede funcionar correctamente en condiciones ruidosas.
- **Tamaño del dominio:** Identifica si el sistema cuenta con el reconocimiento de un lenguaje de dominio reducido.

La API Web Speech tiene como objetivo permitir a los desarrolladores web proporcionar, en un navegador web, características de salida de voz a voz que típicamente no están disponibles cuando se usa software de reconocimiento de voz estándar o de lector de pantalla. La propia API es agnóstica de la implementación subyacente de reconocimiento y síntesis de voz y puede soportar tanto el reconocimiento basado en servidor como el basado en el cliente y la síntesis. La API está diseñada para permitir tanto la entrada de voz breve (un disparo) como la entrada de voz continua. Los resultados de reconocimiento de voz se proporcionan a la página web como una lista de hipótesis, junto con otra información relevante para cada hipótesis.

Esta especificación es un subconjunto de la API se define en la Speech Incubadora Específicamente, este subconjunto excluye el protocolo de transporte subyacente, las adiciones propuestas al marcado HTML y define un subconjunto simplificado de la API de JavaScript. (Glen Shires,Hans Wennborg, 2012)

### **2.11. API Google**

API de google es un servicio que nos permite modificar y mejorar su código para personalizarlo en un aplicación nueva que se desarrolle, la API nos permite realizar una transcripción del audio que se recibe por medio de un micrófono en nuestra aplicación y se aplica modelos de redes neuronales de un amplio alcancé, enfocados en el procesamiento del lenguaje natural

PNL, se debe tener en cuenta que al utilizar redes neuronales de procesamiento de lenguaje natural la API estará en constante aprendizaje convirtiéndose cada vez más preciso en las transcripción de audio a texto.

Se realizará la publicación del asistente en Internet para que puedan ingresar libremente las personas con visión subnormal y público en general, además Google Chrome nos permite crear accesos directos lo cual ayuda para que las personas con visión subnormal puedan ingresar de manera fácil.

## 2.12. Lenguajes Web y Comparación

Tabla 2.2. Comparación de lenguajes web

Nro.	Lenguaje	Fortalezas	Debilidades
1	PHP	<p>Orientado al desarrollo de aplicaciones web dinámicas</p> <p>Es considerado un lenguaje fácil de aprender, ya que en su desarrollo se simplificaron distintas especificaciones, como es el caso de la definición de las</p>	<p>Como es un lenguaje que se interpreta en ejecución, para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado.</p> <p>La ofuscación es una técnica que puede dificultar la lectura del código, pero no necesariamente impide que el código sea</p>

		<p>variables primitivas.</p> <p>El código fuente escrito en PHP es invisible al navegador web y al cliente, ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador.</p> <p>Permite aplicar técnicas de programación orientada a objetos</p> <p>No requiere definición de tipos de variables, aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución</p>	<p>examinado.</p> <p>un script en PHP suele funcionar considerablemente más lento que su equivalente en un lenguaje de bajo nivel, sin embargo, este inconveniente se puede minimizar con técnicas de caché tanto en archivos como en memoria</p> <p>dificulta a los diferentes IDEs ofrecer asistencias para el tipificado del código</p>
	ASP.NET	<p>Flexibilidad en opciones de idioma, le permite aprovechar habilidades actuales de</p>	<p>Es un código propietario de Microsoft no abierto.</p> <p>No es fácil de leer e interpretar, se necesita</p>

		<p>lenguaje de programación</p> <p>Gran herramienta de soporte</p> <p>ASP.NET no sólo funciona sobre el servidor de Microsoft IIS, también lo hace sobre Apache</p> <p>programación dinámica se use el modelo code-behind, o de respaldo, que coloca el código en un archivo separado</p>	<p>escribir más código para hacer lo mismo que con otros lenguajes como el PHP.</p> <p>Muchos hostings y alojamientos web no lo soportan por su alto coste. ASP.NET necesita tener instalado IIS con el Framework .Net.</p> <p>El consumo de recursos de ASP.NET es importante por lo que se requieren servidores de mayor capacidad.</p>
	<p>JSP</p>	<p>El código está bien estructurado y resulta fácil de leer si se conoce bien el lenguaje.</p> <p>La parte de JSP dinámico se escribe en Java, permitiendo una integración total con</p>	<p>Es un lenguaje bastante complejo y que requiere bastante tiempo de aprendizaje.</p> <p>El alojamiento web requiere tener instalado un servidor Tomcat.</p>

		módulos Java y la utilización de un motor de páginas basado en servlets de Java.	
	Ruby	<p>Es un lenguaje sencillo y fácil de leer.</p> <p>Soportado por la mayoría de las plataformas web.</p> <p>Se trata de un software libre u open source.</p> <p>Integra comandos de manejo de bases de datos</p>	<p>Su uso no está muy extendido.</p> <p>En crecimiento</p>

### 2.13. Lenguaje HTML

HTML está compuesto para estructurar documentos como párrafos listas títulos, etc. Pero no significa que HTML va a modificar la apariencia del diseño del documento, sino que se encarga de ofrecer las herramientas necesarias para el formato del documento, según para la capacidad del servidor web y la capacidad del navegador. HTML tiene dos ventajas que son la compatibilidad y la facilidad que ayuda a que su aprendizaje no sea

difícil sino fácil debido a que el número de etiquetas que ofrece HTML son pocas.

La sintaxis de HTML consiste en atributos elementos y etiquetas que hacen la página, tanto como otras cosas que conforman al documento son tablas, imágenes y gráficos, etc., Esto se plasma en comandos como <etiqueta> elemento afectado </etiqueta> como se puede observar la primera etiqueta es la que activa el orden y es la primera que se va situar al principio y la etiqueta con el signo / es la que desactiva para indicar el final de esa etiqueta, pero esto no significa que todas las etiquetas tienen un principio y final.

HTML permite realizar las siguientes operaciones:

- Agregar fondos a un documento y también poder cambiar el color del texto.
- Nos permite cambiar el tamaño, color y tipo de fuente.
- Poder definir enlaces también son llamados como hipervínculos o links, para relacionar otra página web u otros tipos de fuentes de información
- Hacer líneas horizontales para separar texto, poner definir párrafos y centrar los textos.

- Cambia el aspecto al texto de acuerdo al contenido como dirección de correo, contraseñas, etc y poder resaltar el texto.
- Usar listas y viñetas también se pueden utilizar imágenes para las listas y las viñetas.
- Incluir imágenes en formato GIF, PNG o JPEG, a estas imágenes se les puede personalizar en su tamaño o colocando bordes, cuadros, etc.
- Inserción de videos y ficheros multimedia.
- Definir tablas.
- Uso de formularios. Estos nos permiten que el usuario pueda mandar información como por ejemplo la respuesta de comentarios, llenar datos o encuestas, etc. (Vértice, 2009)

#### **2.14. Lenguaje PHP**

PHP es un lenguaje interpretado del lado del servicio que se denomina código abierto (open source) PHP se caracteriza por ser robusto, versátil, potente y por su modularidad. En comparación con tecnologías similares. Las instrucciones son integradas directamente dentro del código HTML.

Comparado con ASP, la principal ventaja de PHP es que es multiplataforma, por otro lado, los programas de ASP resultan ser más lentos y pesados y a la vez menos estable. En los entornos de la plataforma

de Microsoft la ventaja de ASP es que los servidores web soportan directamente ASP sin realizar ninguna instalación adicional.

PHP es un lenguaje de los más utilizados actualmente en el desarrollo de aplicaciones web y tiene un constante crecimiento en su nivel de utilización en Internet. (Ángel Cobo, 2005)

Una página en PHP se crea en principio como página HTML y se guarda con la extensión PHP. Al insertar código PHP este puede ir en cualquier parte, también dentro de un elemento HTML, delante de la etiqueta <head> o detrás de la etiqueta </body>. El código PHP estará incluido dentro de los símbolos como apertura “<?php” y como cierre “?>”, de esta manera podemos identificar código PHP. (Spona, 2010)

### **2.15. Lenguaje JavaScript**

JavaScript, es un lenguaje interpretado y basado en líneas que son directamente integrados en el código HTML, El código se transfiere al cliente para que el cliente lo pueda interpretar y cargar la página. Con JavaScript no se pueden crear programas independientes.

Las principales características de este lenguaje son:

- Es un lenguaje interpretado.
- No necesita compilación.

- Multiplataforma.
- Lenguaje de alto Nivel.
- Permite programación estructurada.
- Programación Basado en Objetos
- Maneja los eventos que se pueden ejecutar sobre la página web.
- No se necesita ningún kit o entorno de desarrollo. (Ángel Cobo, 2005)

El programa en JavaScript reconoce muchos eventos uno de ellos es el onClick, que se refiere a que al pulsar con el cursor en la zona asociada este evento realizara una acción. Es por ello que podemos crear formularios que verifiquen la validez de la información sobre contenidos y código HTML sin necesidad de entablar alguna comunicación en red. (S.L., 2001)

### **2.16. Diseño web responsive**

El Diseño Web Responsive soluciona los problemas del diseño ya que existe una gran diversidad de dispositivos y resoluciones. Este enfoque se centra en el cliente, contenido y experiencia de usuario, de tal manera que no habría dificultar si se navega en una misma página web ya sea en un computador, Tablet o smartphone.

El Diseño Web Responsive quiere eliminar que se haga diferentes diseños y desarrollos para distintas resoluciones, lo que sugiere es que el desarrollo de soporte y responda a la necesidad del contexto que se está ejecutando, considerando los parámetros como el tipo de dispositivo, orientación y el

tamaño de pantalla, de esta manera que la página web se adapte a cada dispositivo creando una solución única. (Adrián Vega,2013)

### **2.17. JSON**

Es un lenguaje de intercambio de datos que es fácil de utilizar y analizar para ordenadores. JSON es compatible con el interior de JavaScript y el más adecuado para las aplicaciones de JavaScript, así proporcionando un rendimiento significativo sobre XML, que requiera bibliotecas adicionales para la recuperación de datos del documento de modelo de objetos (DOM). Se estima que JSON analiza cien veces más rápido que XML en los navegadores modernos. JSON usa contexto para evitar la ambigüedad al igual que la programación de idiomas. (Nurzhan Nurseitov,2015)

### **2.18. Metodología AGILE SCRUM**

Las metodologías ágiles de desarrollo están concebidas para proyectos con requerimiento cambiantes y adaptativos a nuevos cambios en los requerimientos esto sin afectar en gran manera el desarrollo del proyecto, esto es gracias a que las metodologías ágiles disgrega en pequeños grupos los requerimientos resolviendo problemas concretos modulando para una implementación exitosa del software a desarrollar.

Scrum es un marco de referencia que toma en cuenta que el equipo de desarrolladores puede tener errores y nuevas ideas en el proceso de desarrollo que ayudan a garantizar una construcción de software más robusta y sin tener una rigidez en el proceso, en scrum se crea una pila de

requerimientos los cuales se ordenan por problema y nivel de prioridad las cuales se tomas de las características que tendrá el producto final. Al realizarse pequeñas tareas en ciclos iterativos llamados SPRINT se puede identificar si alguna característica o funcionalidad pueda estar mal enfocada y solucionarla en el momento ayuda a identificar cuellos de botellas con anticipación.

Finalmente, al culminar con la pila de requerimientos y sprints se tendrá un producto final el cual pasara una serie de pruebas que ayudaran a asegurar la integridad del software desarrollado. (Dimes, 2015)

### The Agile: Scrum Framework at a glance

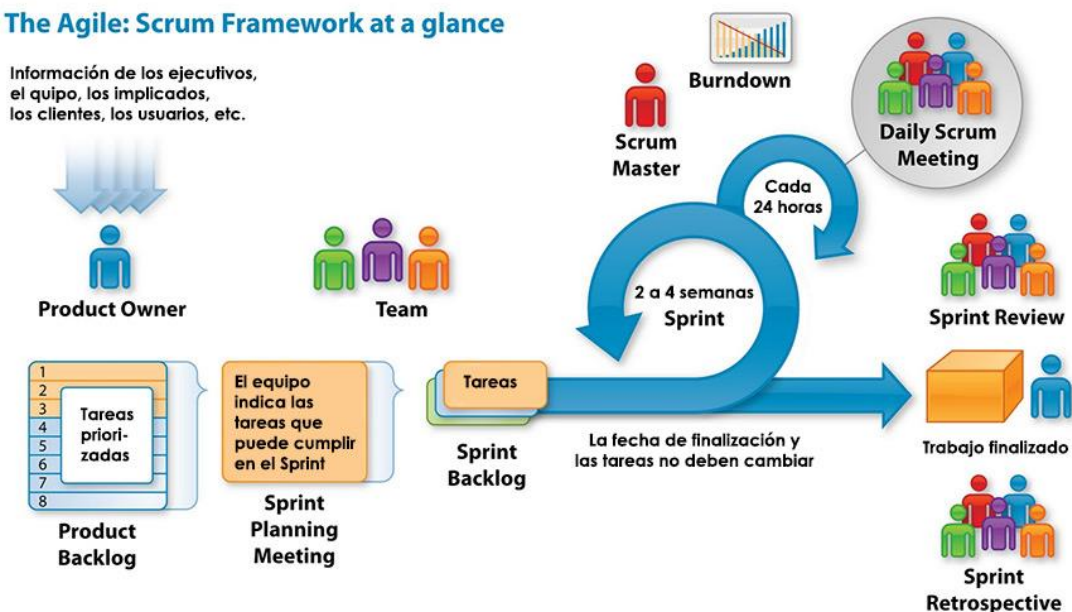


Figura 2.7. Representación de Metodología Agile Scrum

Fuente: (Torras, 2015)

## CAPITULO 3: IDENTIFICACION Y ESPECIFICACIÓN DE REQUERIMIENTOS

En el presente capítulo se identificarán y definirán los requerimientos que permitirán tener una guía en el desarrollo del sistema a diseñar durante su ciclo de vida. Se seguirán etapas para la realización del sistema, que son: análisis, diseño, desarrollo, pruebas e implementación.

Se realizará el modelado de requerimientos para determinar la información requerida y necesaria que ayuda a la construcción del software para especificar, analizar y evaluar las propiedades del producto. Es necesario identificar la problemática para poder capturar los requerimientos de manera eficaz y que nos permita conocer el proceso a seguir.

### **3.1. Análisis de situación actual**

En la actualidad los asistentes de voz desarrollados por grandes empresas tecnológicas están enfocadas a dar órdenes al móvil o dispositivo para que realice ciertas acciones. Si bien la interacción entre usuario y aplicación es por voz este se limita a solo mostrar datos por pantalla o confirmar una acción solicitada por el usuario dejando de lado a personas con discapacidad visual o personas con habilidades diferentes. Nuestra aplicación web tiene como propósito el crear un lazo entre el usuario y el ordenador, con la finalidad de realizar búsquedas de información, que no se limitarán a solo mostrar por pantalla, sino que también las reproducirá para que el usuario pueda escuchar los resultados dando la opción de elegir un

resultado en particular y mostrar el contenido completo de esta opción elegida.

Se trabajará con el motor de búsqueda de Google, directamente personalizando este buscador a nuestras necesidades, de modo que nos brinden un breve resumen y título principal de las 10 primeras búsquedas con alta relevancia para mostrar las respuestas que los usuarios están buscando: De esta manera, el asistente pueda leer los resultados del usuario.

### 3.2. Definición de requerimientos

Se identificarán los requerimientos generales para la solución de problema, mediante una serie de preguntas y encuestas las cuales ayudarán a recopilar las necesidades principales que tiene los usuarios.

#### 3.2.1. Identificar actores

El presente trabajo está diseñado para un tipo de actor:

Usuario, el cual podrá interactuar directamente con la aplicación por medio de la voz detallando sus privilegios en la tabla 3.1.

Tabla 3.1. Privilegios de actores

	Usuario
Interacción con la Aplicación	X
Búsquedas específicas de contenido	X

Fuente: Propia

### 3.3. Catalogación de requerimientos

Después de analizar e investigar sobre la existencia de asistentes especializados en personas con visión subnormal se podrá identificar los requisitos, definiendo y catalogando los requerimientos en funcionales y no funcionales que deberán satisfacer las necesidades del sistema, indicando sus prioridades.

#### 3.3.1. Requerimientos funcionales

Los requerimientos funcionales se refieren a procedimientos, actividades y consultas necesarias para llegar a un objetivo específico a continuación, se muestra la figura 3.1 los requisitos funcionales de la solución web.

##### **RFU01. Reconocimiento de Voz**

**RFU1.1** La aplicación deberá reconocer las palabras que se está dictando por medio del micrófono.

**RFU1.2** La librería de reconocimiento de voz deberá ser capaz de reconocer las palabras en español.

**RFU1.3** Se deberá mostrar en la interfaz las predicciones del reconocimiento de voz que se está obteniendo al momento de usar el micrófono.

##### **RFU02. Sintetizador de Voz**

**RFU2.1** La aplicación deberá reproducir las palabras que se envíen para poder ser reproducida en los parlantes.

**RFU2.2** La librería de sintetizador de voz deberá ser capaz de reconocer las palabras en español.

**RFU2.3** Se deberá mostrar en la interfaz el texto que se está reproduciendo.

### **RFU03. Consulta de búsqueda**

**RFU3.1** La aplicación deberá ser capaz de obtener una cadena de texto que se utilizará para la búsqueda.

**RFU3.2** Se deberá mostrar en la interfaz la búsqueda realizada y los resultados obtenidos de la búsqueda.

### **RFU04. Secuencia de palabras claves**

**RFU4.1** La aplicación deberá reconocer las palabras clave o palabras reservadas que ayudarán a renacer cuando parar, volver a buscar o cerrar la aplicación.

## **3.3.2. Requerimientos no funcionales**

Los requerimientos no funcionales se refieren a calidad, disponibilidad, prestaciones, restricciones y o cualidades que el sistema web debe tener.

### **RNF01 Rendimiento**

**RNF1.1** La aplicación solución web deberá realizar las búsquedas de manera óptimas en un rango no mayor a 30sec. Evaluando el desempeño en la medición tomada por Chrome y Lighthouse Audit.

**RNF1.2** La aplicación solución web deberá devolver los resultados de acuerdo a lo solicitado por el usuario teniendo como máximo las 10 más relevantes.

#### **RNF04 Disponibilidad**

**RNF4.1** La aplicación solución web deberá estar disponible 24/7 durante todo el año para realizar búsquedas en todo momento.

**RNF4.2** La aplicación solución web deberá contar con backup en caso de alguna pérdida de datos.

#### **RNF05 Usabilidad**

**RNF5.1** La aplicación solución web deberá tener como premisa el uso sea sencillo y sin complicaciones para el usuario final evaluándose con Lighthouse Audit una extensión que verifica la buenas practicas.

**RNF5.2** La aplicación solución web deberá contar con una interfaz intuitiva evaluándose con Lighthouse Audit una extensión que verifica la buenas practicas

**RNF2.1** La aplicación solución web deberá ser sencilla y amigable para los usuarios finales se evaluará a través de Lighthouse Audit proporcionado por Chrome.

**RNF2.2** La aplicación solución web deberá mostrar el contenido obtenido de la búsqueda de manera rápida y ordenada se medirá la rapidez según los marcadores que indica el navegador Chrome.

### **RNF06 Concurrencia**

**RNF6.1** La aplicación solución web deberá estar disponible y soportar el alto índice de concurrencia (máx.1000/m usuarios al mismo tiempo) en peticiones de búsqueda para realizar en todo momento la cual se podrá monitorear desde la consola del servidor de aplicaciones.

**RNF6.2** La aplicación solución web si presenta alta concurrencia (exceder las 1000/m peticiones) deberá rechazar peticiones de forma aislada para no afectar a los demás usuarios.

### **RNF07 Portabilidad**

**RNF7.1** La aplicación solución web deberá estar disponible en cualquier equipo que cuente un navegador web Chrome y una conexión a internet.

**RNF7.2** La aplicación solución web puede ser consumida desde Chrome en cualquier SO que tenga este navegador instalado.

## **3.4. Análisis de factibilidad para implementación web**

Para el análisis de factibilidad se consideró los puntos anteriores los cuales contribuirán a que el sistema web funcione al culminar su desarrollo ya que se cuenta con el compromiso de terminar dicho sistema web asegurando la factibilidad operacional.

El sistema web es factible técnicamente ya que las tecnologías que se están empleando son actualmente usadas en el mercado. En concreto, se considera en la implementación de una arquitectura Cliente-Servidor (Figura

3.1) que se integra con los servicios de búsqueda de Google para la obtención de datos esta interacción se dará a través de una interfaz diseñada en HTML5 y por el lado servidor PHP.

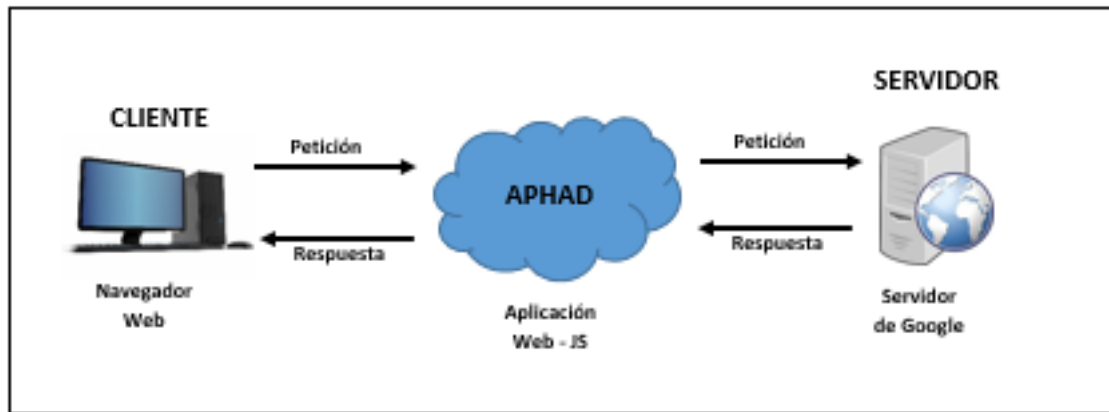


Figura 3.1. Representación de la tecnología usada en el sistema web

Fuente: Propia

### 3.5. Análisis de casos de uso

#### 3.5.1. Elaboración de Caso de Uso Ingreso de valor de Búsqueda a petición del Usuario

**Descripción:** permite al usuario interactuar con el sistema web por medio de la voz.

**Actores:** Usuario

**Precondiciones:** el usuario tiene que tener instalada la extensión del sistema web, debe tener un micrófono operativo y parlantes.

**Flujo del evento:** Diagrama del caso de uso figura 3.2

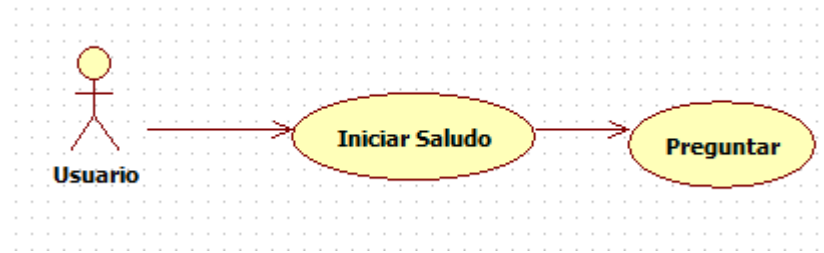


Figura 3.2. Caso de uso búsqueda de petición del usuario.

Fuente: Propia

**Flujos alternos:** micrófono no está operativo.

**Acción:** si no se recibe ninguna entrada por medio de la voz la aplicación seguirá en espera para obtener el valor a buscar.

### 3.5.2. Elaboración de Caso de Uso Confirmar Búsqueda a petición del Usuario

**Descripción:** Permite al usuario realizar la búsqueda en el sistema web por medio de la voz.

**Actores:** Usuario

**Precondiciones:** El usuario tiene que tener instalada la extensión del sistema web, debe tener un micrófono operativo y parlantes.

**Flujo del evento:** Diagrama del caso de uso figura 3.3

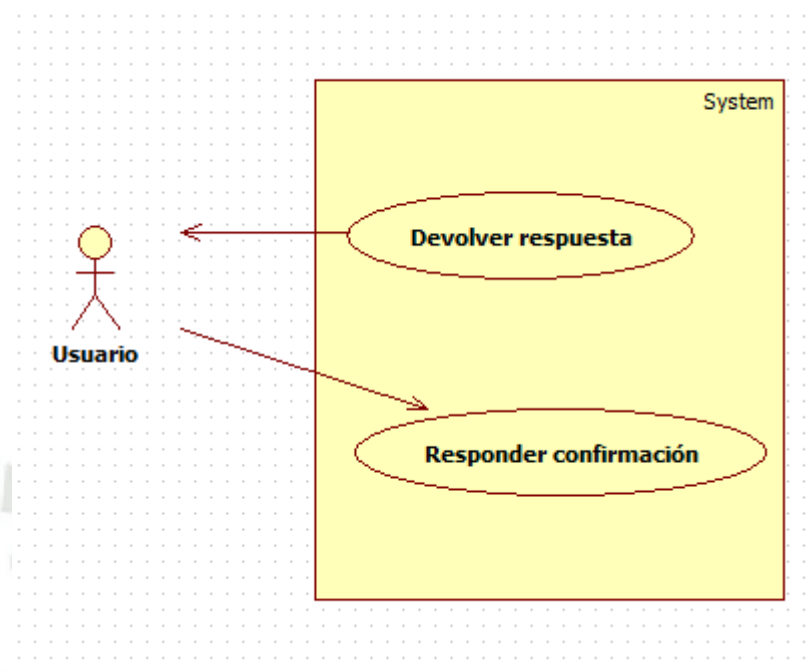


Figura 3.3. Caso de uso búsqueda de petición del usuario.

Fuente: Propia

**Flujos alternos:** Micrófono no está operativo.

**Acción:** Al recibir la confirmación del valor a buscar por el usuario se desencadena el evento de búsqueda en motores de Google.

### 3.5.3. Elaboración de Caso de Uso Elección de respuesta a detallar a petición del Usuario

**Descripción:** Permite al usuario escuchar los resultados de búsqueda y escoger que detalle una de ella interactuando con el sistema web por medio de la voz.

**Actores:** Usuario

**Precondiciones:** El usuario tiene que tener instalada la extensión del sistema web, debe tener un micrófono operativo y parlantes.

**Flujo del evento:** Diagrama del caso de uso figura 3.4

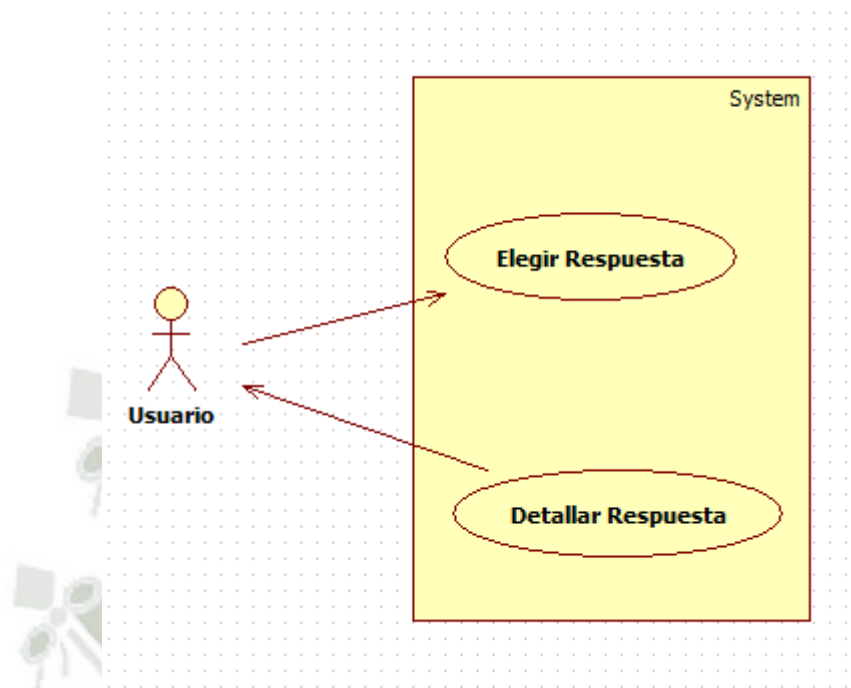


Figura 3.4. Caso de uso búsqueda de petición del usuario.

Fuente: Propia

**Flujos alternos:** Micrófono no está operativo.

**Acción:** El usuario recibe los resultados de la búsqueda por medio de un audio y este puede escoger que opción de búsqueda desea detallar.

#### 3.5.4. Elaboración de Caso de Uso recepción auditiva de la búsqueda detallada a petición del Usuario

**Descripción:** Permite al usuario recepcionar de forma auditiva el resultado al interactuar con el sistema web por medio de la voz.

**Actores:** Usuario

**Precondiciones:** El usuario tiene que tener instalada la extensión del sistema web, debe tener un micrófono operativo y parlantes.

**Flujo del evento:** Diagrama del caso de uso figura 3.5

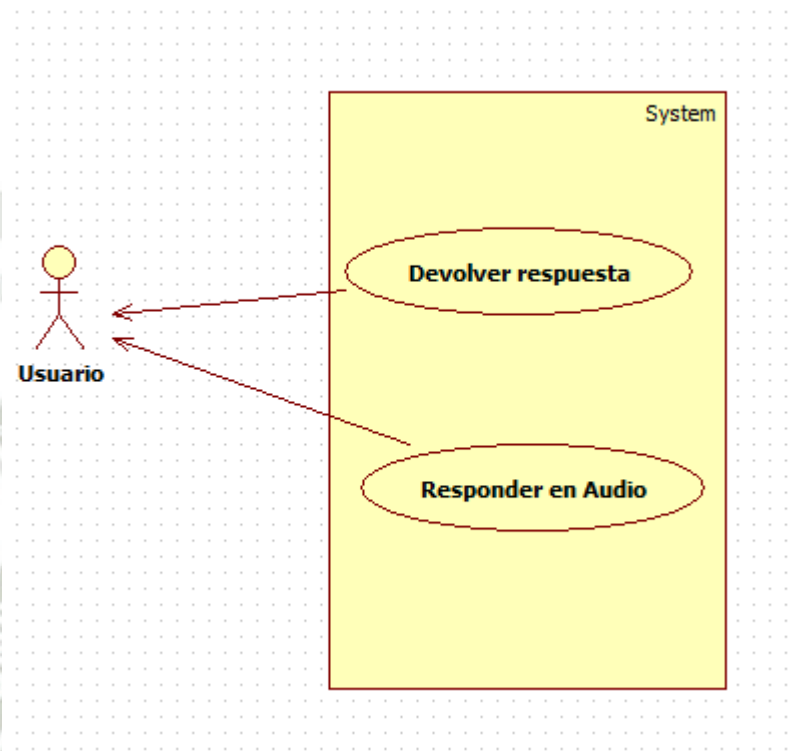


Figura 3.5. Caso de uso búsqueda de petición del usuario.

Fuente: Propia

**Flujos alternos:** Micrófono no está operativo.

**Acción:** El usuario desencadena el evento de búsqueda detallada por la opción escogida y el sistema web devuelve el resultado por medio auditivo.

## CAPITULO 4: DISEÑO E IMPLEMENTACION DE SOLUCION WEB

### 4.1. Arquitectura de sistema

#### 4.1.1. Arquitectura lógica

La arquitectura lógica elegida es una de las más usadas actualmente en el desarrollo de aplicaciones web denominada arquitectura de capas. Este estilo de arquitectura lógica separa de manera lógica aspectos de presentación de la aplicación, la lógica del negocio y la gestión de los datos tal como se muestra en la siguiente figura 4.1

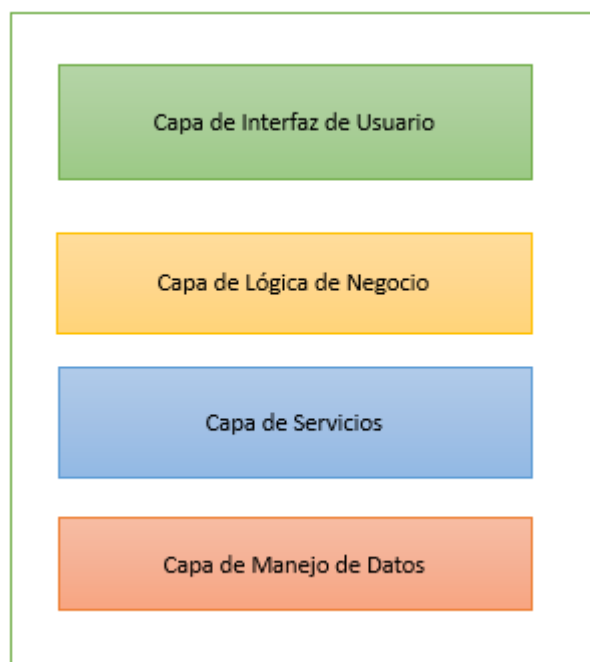


Figura 4.1. Patrón de Diseño por capas

Fuente: Propia

La capa de interfaz de usuario es la encargada de manejar el ingreso de datos mostrando por pantalla lo que se está dictando además de mostrar

el resultado de la búsqueda de forma ordenada para su posterior lectura.

Esta capa se comunica únicamente con la capa de lógica de negocios.

La capa de lógica de negocio tiene como responsabilidad manejar la funcionalidad de la solución web con interacción de las librerías JavaScript se procede a construir los textos dictados por el usuario para su posterior búsqueda a la misma vez una vez recibida la respuesta del servidor de google esta capa se encargará de transformar el texto a voz para poder ser dictado por los parlantes del equipo.

Esta capa lógica se comunica con la capa de interfaz de usuario, para recibir las solicitudes y presentar los resultados interactuando con la capa de manejo de datos que solicitará la búsqueda de la petición y recuperar los datos de la búsqueda realizada.

La capa de manejo de datos en este caso será la encargada de devolver la búsqueda solicitada por la capa de lógica de negocio limitándose a presentar los 10 primeros resultados con relevancia alta esta clasificación designada por Google.

La solución web empleará el estilo de capas (ver en la figura 4.2) la cual se organizará con la capa de interfaz, la capa de lógica de negocio y la

capa de manejo de datos. La capa de interfaz la cual interactúa directamente con el usuario final, se implementó en HTML5, CCS3, JS. La capa de lógica de negocios contendrá un conjunto de procedimientos para recibir las peticiones del usuario y sean enviados para ser procesados en los motores de búsqueda de Google a través de nuestros sistemas web. Este se compondrá principalmente de PHP, JSON, JS. Finalmente la capa de manejo de datos, la cual se encargará de devolver las búsquedas enviadas como peticiones desde los motores de búsqueda de Google.

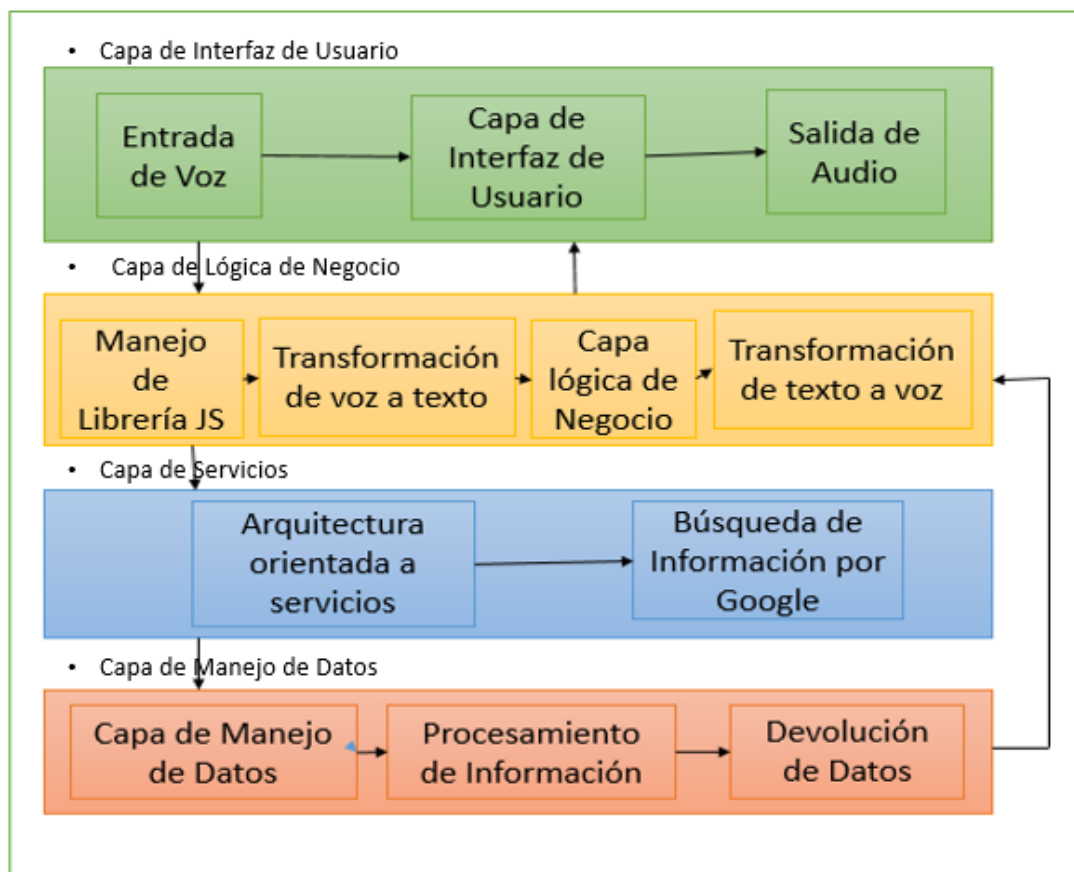


Figura 4.2. Patrón detallado de Diseño por capas

Fuente: Propia

#### 4.1.2. Arquitectura física

La arquitectura física tomada para esta solución web, primero se debe considerar que usuario tenga una PC con la aplicación web (Software) y diferentes equipos de computación además de periféricos de salida y entrada como parlantes y micrófonos respectivamente (hardware). La arquitectura física del sistema es representada con la siguiente figura 4.3.

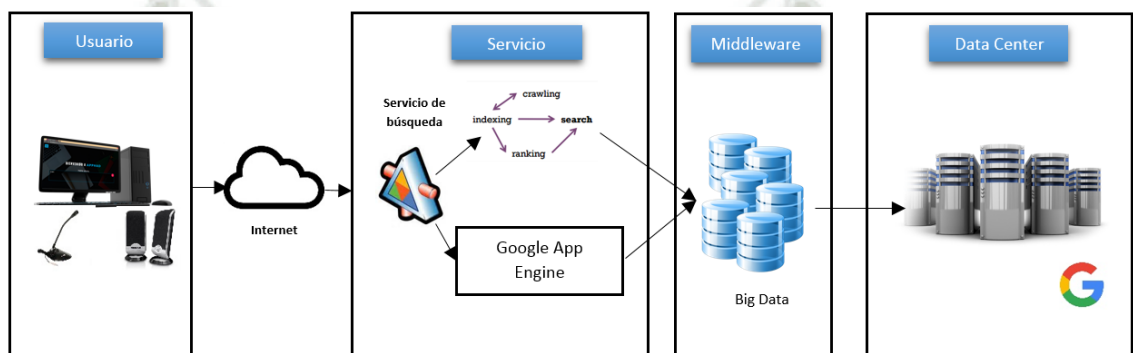


Figura 4.3. Arquitectura Física de la Solución Web

Fuente: Propia

#### 4.2. Secuencia de Solución Web

La secuencia de la solución web se diseña con la finalidad de tener presente como el usuario final podrá interactuar con la solución web y esta pueda realizar la búsqueda y mostrar los datos correctos al usuario. A continuación, en la figura 4.4 se detalla la secuencia propuesta para la solución web.

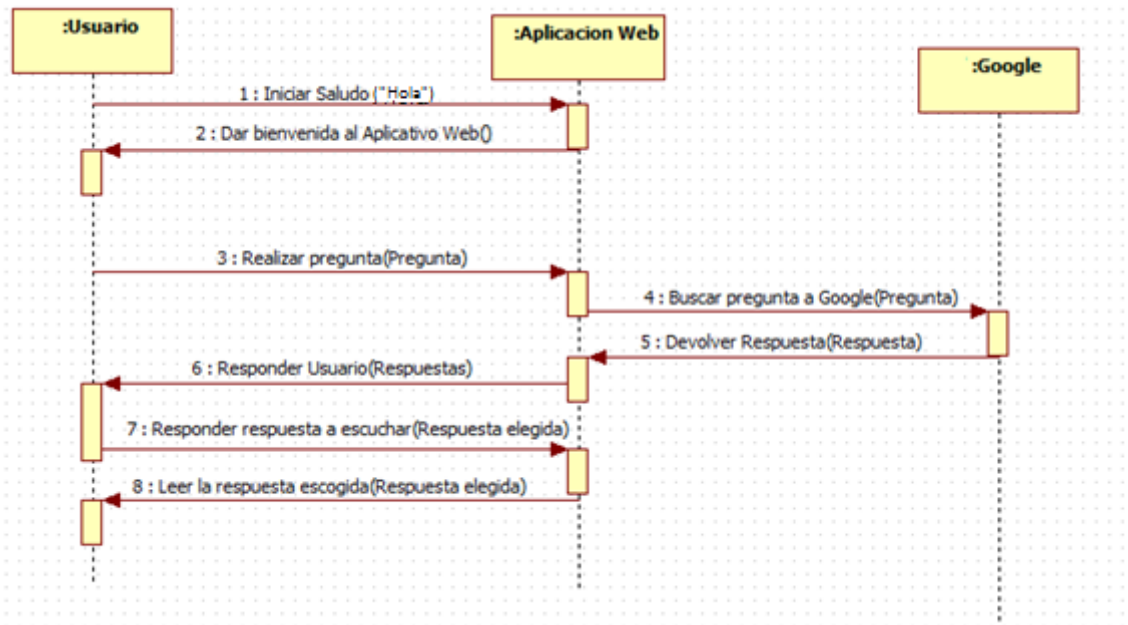


Figura 4.4. Diagrama de secuencia de la Solución Web

Fuente: Propia

### 4.3. Actividades de Solución Web

En la solución web se ha definido un conjunto de actividades que se tiene que realizar para la ejecución correcta de la aplicación y que esto no afecte la interacción con el usuario final, se muestra en la figura 4.5 al detalle en conjunto de actividades que realizará la solución web.

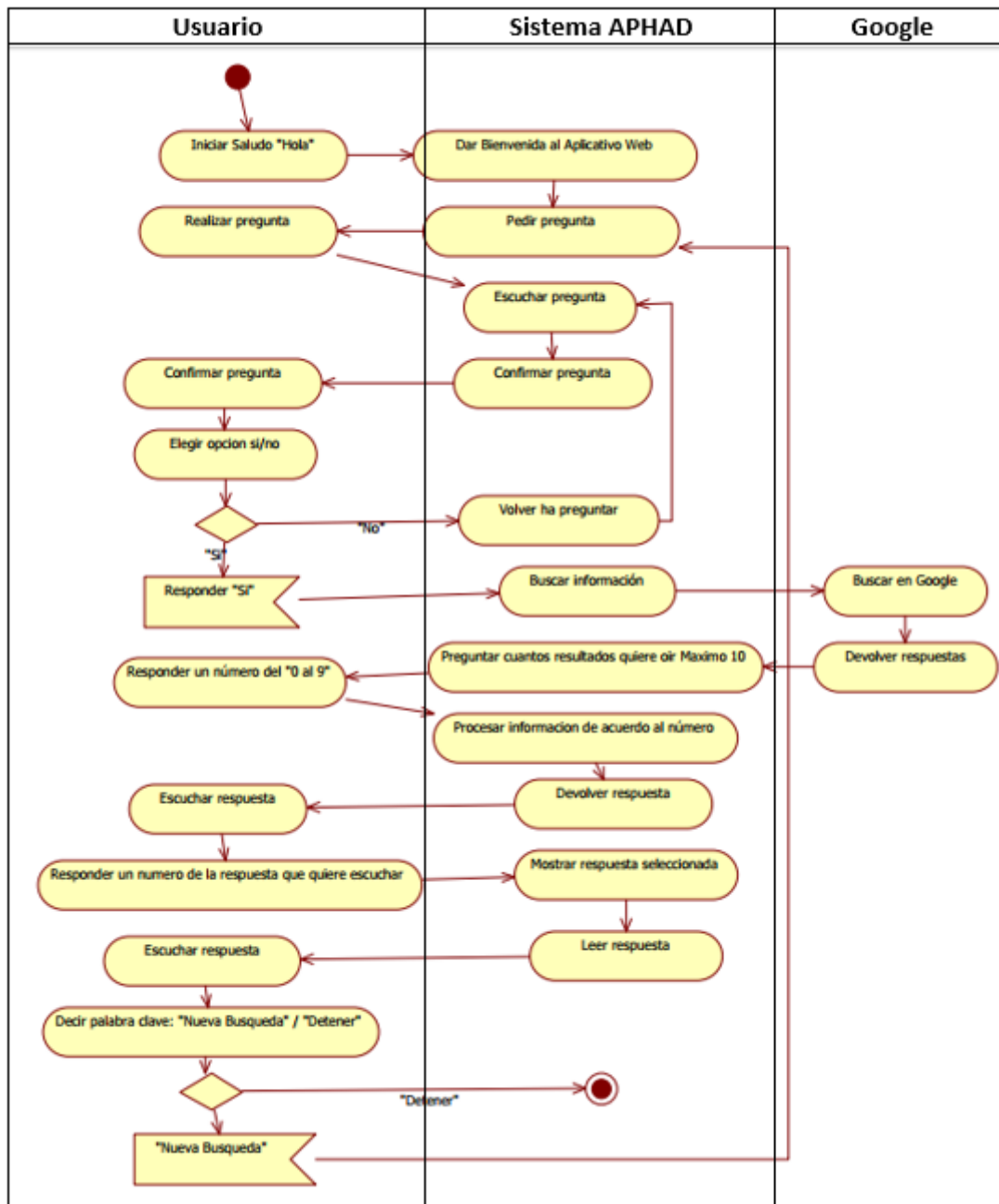


Figura 4.5. Diagrama de actividades de la Solución Web

Fuente: Propia

#### **4.4. Métodos principales**

##### **4.4.1. Método Start**

El método Start es llamado al inicio de la aplicación, representa el momento en el que la aplicación web desea comenzar el reconocimiento. El momento que la entrada de voz se trasmite a través de un flujo de medios de entrada (micrófono) esta llamada al método da inicio a escuchar y el servicio debe comenzar a reconocer lo que se está dictando.

##### **4.4.2. Método Stop**

El método Stop es llamado cuando el servicio de reconocimiento deja de escuchar más audios y este intenta devolver los resultados de los reconocimientos. El usuario final puede presionar o decir una palabra clave para llamar a este método.

##### **4.4.3. Método Abort**

El método Abort es llamado cuando el servicio presenta algún problema al escuchar o devolver los resultados del reconocimiento de voz. Este método detiene todo proceso o evento que se esté ejecutando en ese momento.

#### **4.5. Eventos Principales**

Los eventos se manejan con el modelo de eventos DOM para el reconocimiento de voz. Los métodos de la interfaz EventTarjet deben utilizarse para registrar a los oyentes de eventos. Para todos los eventos definidos se le agregó un atributo timeStamp definido en la interfaz de

eventos por DOM el cual establece en la mejor estimación posible el tiempo por el cual el evento se debe ejecutar. También para hacer la búsqueda más rápida podemos acelerar con la barra espaciadora, así tendremos resultados más rápidos.

#### **4.5.1. Evento audiostart**

Este evento es disparado cuando la aplicación ha comenzado a capturar el audio del usuario.

#### **4.5.2. Evento soundstart**

Este evento es disparado cuando se ha detectado algún sonido, generalmente es activado con baja latencia.

#### **4.5.3. Evento speechstart**

Este evento es disparado cuando se inicia la entrada de audio de manera fluida para el reconocimiento de voz.

#### **4.5.4. Evento speechstop**

Este evento es disparado cuando la entrada de voz ha concluido después de una latencia en espera, este evento se ejecutará solo si speechstart fue disparado con anticipación.

#### **4.5.5. Evento soundend**

Este evento se dispara cuando la aplicación detecta que ya no hay entrada de sonido por parte del usuario.

#### **4.5.6. Evento audioend**

Este evento se dispara cuando se ha terminado de capturar el sonido.

#### **4.5.7. Evento error**

Este evento se dispara cuando se produce un error en la etapa de reconocimiento de voz

#### **4.5.8. Evento Start**

Este evento se dispara cuando el servicio de reconocimiento de voz ha comenzado a escuchar audio con la intención de reconocer.

#### **4.5.9. Evento End**

Este evento se dispara cuando el servicio de ha desconectado.

### **4.6. Frases reservadas**

Las frases reservadas son aquellas que desencadenaran eventos en la aplicación web para realizar una serie de interacciones por parte del usuario con la aplicación web.

#### **4.6.1. “Hola”**

Esta frase reservada desencadenará una serie de eventos que alistará a la aplicación web para escuchar al usuario y reconocer el flujo de audio.

#### **4.6.2. “Si”**

Esta frase reservada desencadenará una serie de eventos que realizará la petición de búsqueda en la aplicación web con el motor de búsqueda de Google para obtener resultados.

#### **4.6.3. “No”**

Esta frase reservada desencadenará una serie de eventos que realizará la petición de búsqueda en la aplicación web con el motor de búsqueda de Google para obtener resultados.

#### **4.6.4. “Detener”:**

Esta frase reservada desencadenará una serie de eventos que detendrá la lectura de los resultados obtenidos por la búsqueda realizada en la aplicación web.

#### **4.6.5. “Repetir”:**

Esta frase reservada desencadenará una serie de eventos que alistarán el contenido que se va a volver a leer en la aplicación web.

#### **4.6.6. “Nueva Búsqueda”:**

Esta frase reservada desencadenará un evento que detiene todos los servicios de reconocimiento de voz de la aplicación web para posteriormente se cierre la aplicación.

### **4.7. Diseño de interfaces**

La interacción con la solución web es a través de navegador web (Browser), las interfaces que se generarán son páginas web que estarán alojadas en un servidor en la nube. El lenguaje de programación utilizado para elaborar las páginas es HTML5, el cuál puede ser interpretado de diferentes maneras por los navegadores. Para esto se incluirán hojas de estilo CSS que darán un estilo único a la solución web.

También se tomó en cuenta que la solución web será utilizada por diferentes usuarios con nivel de conocimiento informáticos variables por lo cual nuestra interfaz debe ser intuitiva y de fácil uso al usuario final.

#### 4.7.1. Diseño de interfaz pantalla inicial



Figura 4.6. Pantalla principal de la Solución Web

Fuente: Propia

- Etiqueta 1: Zona de notificaciones, indica la acción activa en ese momento.
- Etiqueta 2: Caja de texto que muestra el texto capturado por medio del micrófono
- Etiqueta 3: Icono que indica si el micrófono está escuchando o está desactivado
- Etiqueta 4: Barra de menú para opciones y más información sobre el proyecto

#### 4.7.2. Diseño de interfaz resultado búsqueda

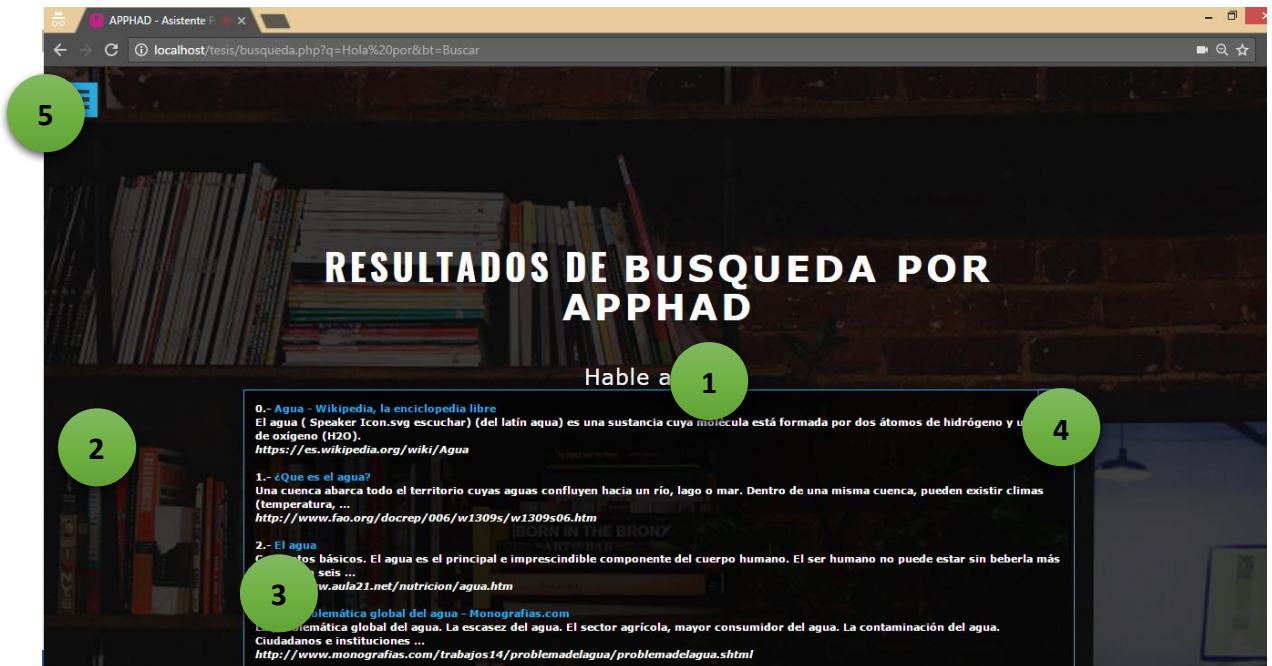


Figura 4.7. Pantalla de resultado de búsqueda de la Solución Web

Fuente: Propia

- Etiqueta 1: Zona de notificaciones, indica la acción activa en ese momento.
- Etiqueta 2: Caja de texto que muestra el resultado de la búsqueda realizada ordenada con enlaces integrados por cada resultado (10 primeros resultados)
- Etiqueta 3: Enlace de hipervínculo agregada para cada resultado de la búsqueda y detalle del mismo
- Etiqueta 4: Icono que indica si el micrófono está escuchando o está desactivado

- Etiqueta 5: Barra de menú para opciones y más información sobre el proyecto

#### 4.7.3. Diseño de interfaz resultado específico



Figura 4.8. Pantalla de interfaz de resultado específico de la Solución Web

Fuente: Propia

- Etiqueta 1: Zona de notificaciones, indica la acción activa en ese momento.
- Etiqueta 2: Caja de texto que muestra el resultado de la búsqueda específica realizada por el resultado escogido.
- Etiqueta 3: Contenido de la página que se buscó.
- Etiqueta 4: Icono que indica si el micrófono está escuchando o está desactivado

- Etiqueta 5: Barra de menú para opciones y más información sobre el proyecto

## 4.8. Desarrollo de solución web

### 4.8.1. FrontEnd

Para el desarrollo de la solución web se inicia con la interfaz mostrada al usuario final esta se divide en 3 pantallas (ver figuras 4.9, 4.10 y 4.11) que se encargarán de mostrar y obtener datos obtenidos por la aplicación.

Para la realizar el front end se utilizó HTML5 por la compatibilidad con distintos navegadores, para darle un formato y estilo a las páginas se usó CSS3 que uniformiza el formato de la página para que el usuario final pueda navegar.



Figura 4.9. Pantalla de interfaz inicial

Fuente: Propia

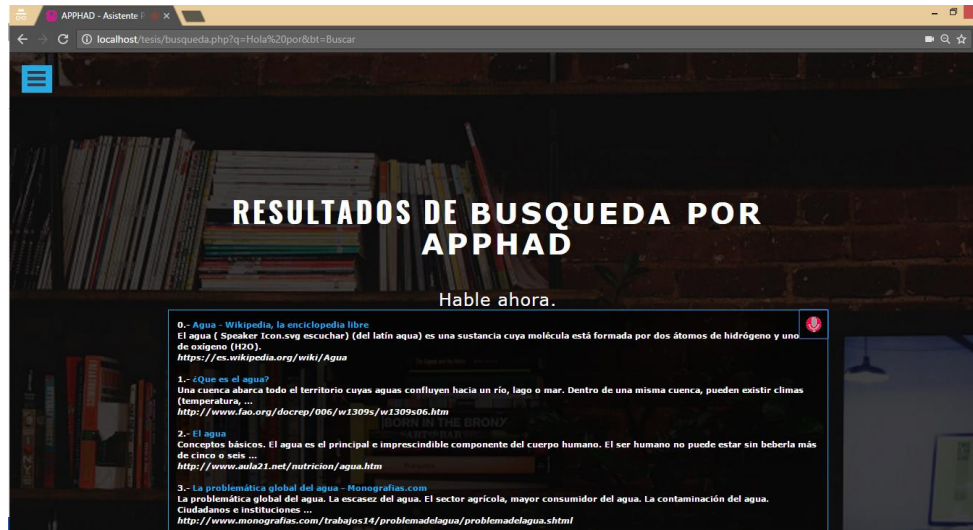


Figura 4.10. Pantalla de interfaz de resultado búsqueda

Fuente: Propia

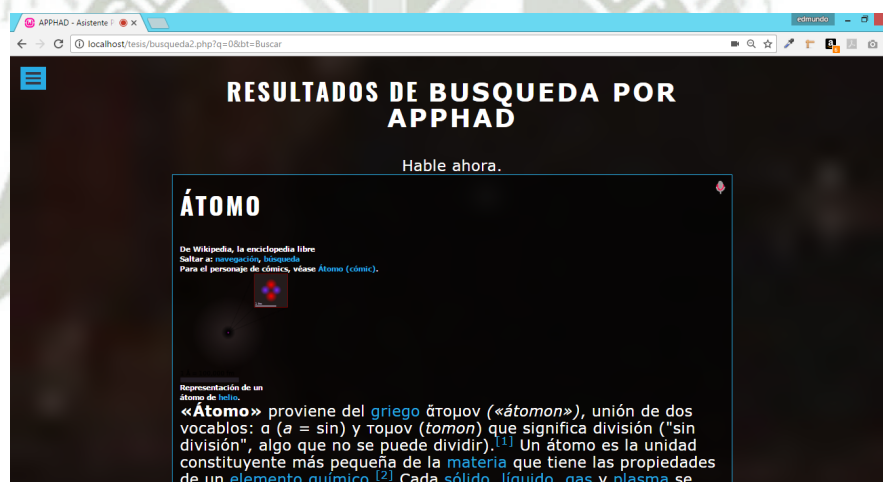


Figura 4.11. Pantalla de interfaz de resultado específico

Fuente: Propia

#### 4.8.2. BackEnd

Para el desarrollo de la solución web se debe tener una capa lógica que se puede denominar backend que se encarga de realizar las peticiones de búsqueda entre la aplicación y los motores de búsqueda de Google. El

usuario ingresará por medio del micrófono una pregunta o texto a buscar y el back end recibirá esta entrada en formato texto y realizará la búsqueda en los motores de búsqueda de Google obteniendo los 10 primeros resultados más relevantes. Posteriormente el back end se comunicará con el front end para mostrar los resultados e interactuar con el usuario por medio de una salida de audio.

Definimos nuestra función de búsqueda (ver figura 4.12) la cual recibe en la variable `urlencode($aQue)` que contiene el texto a buscar. Finalmente se arma la URL a buscar y se almacena en la variable `$url`.

```
function buscar($aQue){  
    $aUrl = "https://www.googleapis.com/customsearch/";  
    $aKey = "&key=AizaSyARBVdohtNeGLDWQAdy-O_08423B7raGqc";  
    $url = $aUrl."v1?q=".urlencode($aQue)."&cX=006116565395137965508:1fccn3hwhbs". $aKey;
```

Figura 4.12. Fragmento de código definición de variables URL búsqueda

Fuente: Propia

Seguidamente realizamos la búsqueda y obtenemos el resultado del mismo por medio de un JSON enviado por los motores de Google (ver figura 4.13).

```
$data = file_get_contents($url); // Obtenemos el resultado  
$obj = json_decode($data); // Decodificamos el json  
$results = $obj->{'items'}; // Entramos al array de los resultados
```

Figura 4.13. Fragmento de código ejecución de búsqueda y recuperación de resultado

Fuente: Propia

Para recorrer el JSON almacenamos el resultado en un array llamado \$results el cual será recorrido con un FOR para añadir dinámicamente el resultado en las variables \$tmpHTML y \$resultados las cuales serán usadas posteriormente para ingresar a un resultado en específico que el usuario escoja (ver figura 4.14).

```
$tmpHTML='';
$resultados='';
# Hacemos un bucle para obtener los 4 resultados que muestra
for ($i=0; $i<sizeof($results); $i++) {
    $tmp = $results[$i];
    $tmpHTML.= $i.".- "."
```

Figura 4.14. Fragmento de código recorrer JSON para dar un formato a los resultados

Fuente: Propia

### 4.8.3. Librerías JS

Las librerías de JS empleadas ayudan a la interacción entre el usuario y la aplicación tanto para la transformación de audio a texto y viceversa. Estas librerías de interacción ayudan a predecir y definir las palabras captadas por medio del micrófono.

La función de starbutton() desencadena una serie de eventos en los cuales está la activación del reconocimiento de voz. La aplicación está escuchando lo que diga el usuario (ver figura 4.15).

```
function startButton(event) {
  if (recognizing) {
    recognition.stop();
    return;
  }
  final_transcript = '';
  recognition.lang = select_dialect.value;
  recognition.start();
  ignore_onend = false;
  final_span.innerHTML = '';
  interim_span.innerHTML = '';
  start_img.src = 'img/mic-slash.gif';
  showInfo('info_allow');
  //showButtons('none');
  start_timestamp = event.timestamp;
}
```

Figura 4.15. Fragmento JS, función que inicia el evento de escuchar.

Fuente: Propia

La función showinfo() es usada para añadir texto al objeto textbox que es mostrada al usuario y esta pueda cambiar dinámicamente según lo que envíe esta función (ver figura 4.16).

```
function showInfo(s) {
  if (s) {
    for (var child = info.firstChild; child; child = child.nextSibling) {
      if (child.style) {
        child.style.display = child.id == s ? 'inline' : 'none';
      }
    }
    info.style.visibility = 'visible';
  } else {
    info.style.visibility = 'hidden';
  }
}
```

Figura 4.16. Fragmento JS, función que se encarga de mostrar el contenido.

Fuente: Propia

La función verification() es la encargada de inicializar el servicio de síntesis de voz enviando el texto plano a la función SpeechSynthesisUtterance() la cual procesará el texto y devolverá en

audio la respuesta. En el fragmento asignamos las variables por DOM al elemento `final_span` que es la caja de texto mostrada al usuario (ver figura 4.17).

```
function verificacion()
{
  var utterance1 = new SpeechSynthesisUtterance('Tu pregunta fue');
  var utterance2 = new SpeechSynthesisUtterance(document.getElementById("final_span").innerHTML);
  var utterance3 = new SpeechSynthesisUtterance('¿es correcto?');
  window.speechSynthesis.speak(utterance1);
  window.speechSynthesis.speak(utterance2);
  window.speechSynthesis.speak(utterance3);
  var mensaje = document.getElementById("final_span").innerHTML;
  msjbuscar = document.getElementById("final_span").innerHTML;
  startButton(event);
}
```

Figura 4.17. Fragmento JS, función que se encarga de inicializar la síntesis de voz.

Fuente: Propia

La función `buscar()` recibe un valor extraído de la caja de texto mostrada al usuario y este realiza la petición de búsqueda por medio de método GET hacia otra página de la aplicación (ver figura 4.18).

```
function buscar(valor)
{
  location.href="busqueda.php?q="+valor+"&bt=Buscar";
}
```

Figura 4.18. Fragmento JS, función que se encarga de enviar el contenido a buscar.

Fuente: Propia

En este fragmento de código estamos declarando variables de lenguaje que utilizará la aplicación para el reconocimiento de voz. Además del proceso de sintetizar texto plano a audio, relocalizamos esta

inicialización de variables con el fin de ser más precisos en el reconocimiento de voz y sintetizar la voz (ver figura 4.19).

```
var langs =
[
  ['Español', ['es-PE', 'Perú']],
  ['Lingua latina', ['la']]
];
for (var i = 0; i < langs.length; i++) {
  select_language.options[i] = new Option(langs[i][0], i);
}
select_language.selectedIndex = 0;
updateCountry();
select_dialect.selectedIndex = 0;
showInfo('info_start');

function updateCountry() {
  for (var i = select_dialect.options.length - 1; i >= 0; i--) {
    select_dialect.remove(i);
  }
  var list = langs[select_language.selectedIndex];
  for (var i = 1; i < list.length; i++) {
    select_dialect.options.add(new Option(list[i][1], list[i][0]));
  }
  select_dialect.style.visibility = list[1].length == 1 ? 'hidden' : 'visible';
}
var meibucan = 1;
```

Figura 4.19. Fragmento JS, variables y función que define el tipo de lenguaje.

Fuente: Propia

En este fragmento de código estamos definiendo variables auxiliares para almacenar ciertos datos como el resultado obtenido del reconocimiento de voz, la transcripción final, si se inicializó de manera correcta el evento de reconocimiento, y si no es así se activa nuevamente dicho evento (ver figura 4.20).

```
var msjbuscar='';
var create_email = false;
var final_transcript = '';
var recognizing = false;
var ignore_onend;
var start_timestamp;
if (!('webkitSpeechRecognition' in window)) {
  upgrade();
} else {
  start_button.style.display = 'inline-block';
  var recognition = new webkitSpeechRecognition();
  recognition.continuous = true;
  recognition.interimResults = true;
  recognition.onstart = function() {
    recognizing = true;
    showInfo('info_speak_now');
    start_img.src = 'img/mic-animate.gif';
  };
};
```

Figura 4.20. Fragmento JS, variables que inicializan el reconocimiento de voz.

Fuente: Propia

En este fragmento de código estamos controlando si hay algún error al momento de inicializar el evento de reconocimiento de voz, como: el error de encontrar el dispositivo de entrada (micrófono), no dar permiso de usar el micrófono, no se detecta ningún sonido después de un tiempo de espera, etc. Esta función muestra por pantalla por medio de una notificación el error presentado mayor detalle en la (figura 4.21).

```
recognition.onerror = function(event) {  
  if (event.error == 'no-speech') {  
    start_img.src = 'img/mic.gif';  
    showInfo('info_no_speech');  
    ignore_onend = true;  
  }  
  if (event.error == 'audio-capture') {  
    start_img.src = 'img/mic.gif';  
    showInfo('info_no_microphone');  
    ignore_onend = true;  
  }  
  if (event.error == 'not-allowed') {  
    if (event.timestamp - start_timestamp < 10000) {  
      showInfo('info_blocked'); /*moificando */  
    } else {  
      showInfo('info_denied');  
    }  
    ignore_onend = true;  
  }  
};
```

Figura 4.21. Fragmento JS, función de captura de error al reconocer la voz.

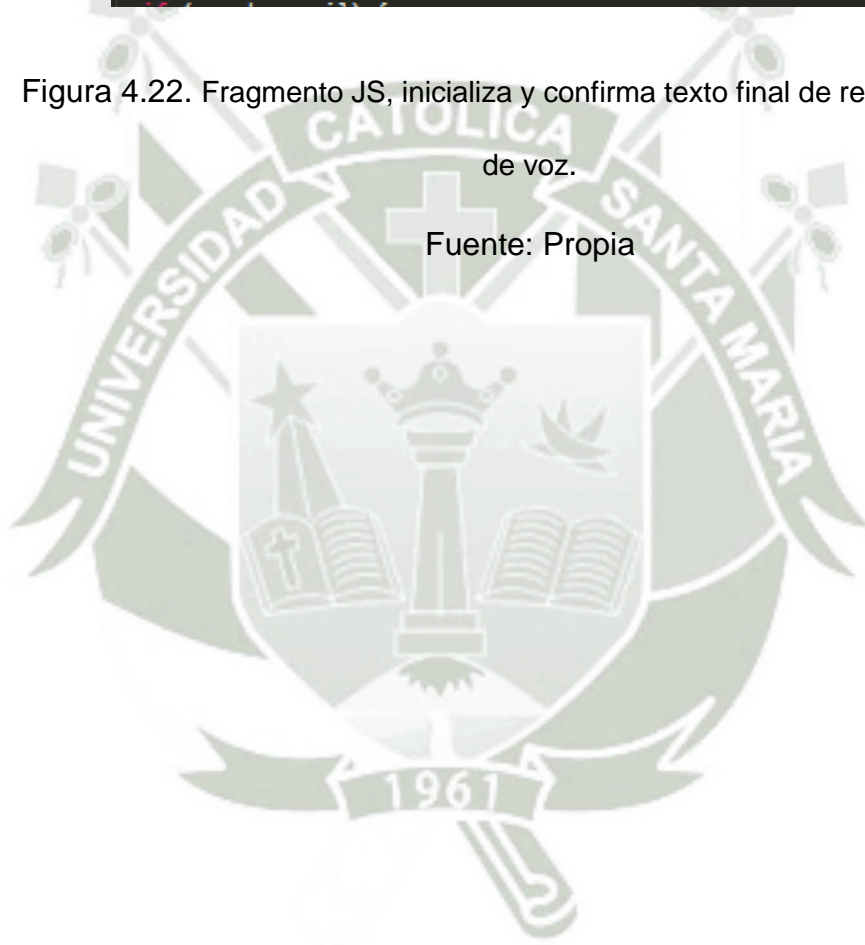
Fuente: Propia

Este fragmento de código se ejecuta cuando ya se tiene un texto definido en la caja de texto después de realizar el reconocimiento de voz. El texto obtenido lo procesa y llama a la función de verification() que realizará la síntesis del texto enviado (ver figura 4.22).

```
recognition.onend = function() {  
  recognizing = false;  
  if (ignore_onend) {  
    return;  
  }  
  start_img.src = 'img/mic.gif';  
  if (!final_transcript) {  
    showInfo('info_start');  
    return;  
  }  
  showInfo('');  
  if (window.getSelection()) {  
    window.getSelection().removeAllRanges();  
    var range = document.createRange();  
    range.selectNode(document.getElementById('final_span'));  
    window.getSelection().addRange(range);  
  }  
}
```

Figura 4.22. Fragmento JS, inicializa y confirma texto final de reconocimiento de voz.

Fuente: Propia



## CAPITULO 5: VALIDACIÓN DE SOLUCION WEB

Este capítulo tiene como finalidad, presentar la ejecución de la aplicación web desarrollada frente a nuestros usuarios finales tanto como personas con visión subnormal y otras personas, con el objetivo de obtener y verificar los resultados provenientes del Asistente para Personas con Habilidades Diferentes (APHAD).

### 5.1. Escenarios previos a puesta en producción

El Asistente para Personas con Habilidades Diferentes (APHAD) tiene como objetivo principal incluir a personas con visión subnormal al mundo tecnológico y que puedan a través de comandos de voz, informarse tanto de noticias, cultura general, etc. Para poder medir el uso y la información que aporta este asistente a las personas con visión subnormal, es necesario realizar capacitaciones en las diversas Instituciones en la Ciudad de Arequipa, sobre el uso del Asistente para Personas con Habilidades Diferentes (APHAD). Para ello, al finalizar las capacitaciones, se toma una encuesta elaborada de acuerdo al funcionamiento del asistente, para poder obtener los resultados.

Como el asistente para Personas con Habilidades Diferentes (APHAD) concierne al ámbito con personas con visión subnormal, fue necesario apoyarse a Instituciones que atiende este tipo de casos, – Nuestra Señora del Pilar (CEBE) (Niños de la edad de 8 a 12 años), Centro de Rehabilitación para Ciegos Adultos (CERCIA) (Adultos entre 27 a 71 años ),

CEPTRO “Nuestra Señora de los Ángeles” (Adultos y niños de 12 a 60 años) y otras personas que nos apoyaron con sus docentes especialistas en estos casos de docentes de Informática, quien nos ayudó a realizar la evaluación del asistente que fue aplicada a la muestra de forma escrita detallada en la tabla

**Tabla 5.1. Encuestas aplicadas**

Institución	8 – 12 años	24 - 29 años (Adultos Jóvenes)	32 – 55 años (Adultos)	56 –78 años (Adultos Mayores)
CEBE – Nuestra Señora del Pilar	8			
CERCIA – Centro de Rehabilitación para ciegos adultos		2	4	2
CEPTRO Nuestra Señora de los Ángeles	1	1	1	1
Otras Personas		2	4	4
	9	5	7	7
Total: 30				

Fuente Propia



Figura 5.1. Capacitación CEBE – Nuestra Señora del Pilar

Fuente Propia



Figura 5.2. Capacitación CERCIA – Centro de Rehabilitación para ciegos  
adultos

Fuente Propia



Figura 5.3. Capacitación CEPTRONuestra Señora de los Ángeles

Fuente Propia

## 5.2. Publicación de solución web

Con fecha 5 de mayo 2017, se llevó a cabo el lanzamiento oficial en nuestros servidores del Asistente para Personas con Habilidades Diferentes (APHAD), con la concurrencia de 30 personas, haciendo capacitaciones y visitando las Instituciones para enseñarles el funcionamiento del Asistente, asimismo dejando libre acceso para estas instituciones y en público en general.

## 5.3. Escenario posterior a puesta en producción

Una vez presentado el Asistente para Personas con Habilidades Diferentes (APHAD) se llevó de 2 a 3 sesiones (ver tabla 5.2) la capacitación en el uso y manejo del asistente. Muchos de ellos solo requirieron una sesión para entender el funcionamiento de este asistente, y hacer uso de él cuándo lo necesiten, dejando el acceso libre para que puedan utilizar en la institución como en su casa.

Tabla 5.2. Numero de Sesiones de entrenamiento realizadas

Institución	Persona	Edad	Número de Sesiones
CEBE – Nuestra Señora del Pilar	1	8	3 Sesiones
CEBE – Nuestra Señora del Pilar	2	8	3 Sesiones
CEBE – Nuestra Señora del Pilar	3	8	3 Sesiones
CEBE – Nuestra Señora del Pilar	4	10	3 Sesiones
CEBE – Nuestra Señora del Pilar	5	10	3 Sesiones
CEBE – Nuestra Señora del Pilar	6	10	3 Sesiones
CEBE – Nuestra Señora del Pilar	7	10	3 Sesiones

CEBE – Nuestra Señora del Pilar	8	10	3 Sesiones
CERCIA – Centro de Rehabilitación para ciegos adultos	9	24	2 Sesiones
CERCIA – Centro de Rehabilitación para ciegos adultos	10	29	2 Sesiones
CERCIA – Centro de Rehabilitación para ciegos adultos	11	35	2 Sesiones
CERCIA – Centro de Rehabilitación para ciegos adultos	12	42	2 Sesiones
CERCIA –	13	43	2 Sesiones

Centro de Rehabilitación para ciegos adultos			
CERCIA – Centro de Rehabilitación para ciegos adultos	14	49	2 Sesiones
CERCIA – Centro de Rehabilitación para ciegos adultos	15	56	2 Sesiones
CERCIA – Centro de Rehabilitación para ciegos adultos	16	71	2 Sesiones
CEPTRO Nuestra Señora de los Ángeles	17	60	1 Sesión
CEPTRO Nuestra Señora	18	12	1 Sesión

de los Ángeles			
CEPTRO Nuestra Señora de los Ángeles	19	27	1 Sesión
CEPTRO Nuestra Señora de los Ángeles	20	32	1 Sesión
Otras Personas	21	23	1 sesión
Otras Personas	22	24	1 sesión
Otras Personas	23	50	2 Sesiones
Otras Personas	24	32	1 sesión
Otras Personas	25	38	1 sesión
Otras Personas	26	40	1 sesión
Otras Personas	27	66	2 Sesiones
Otras Personas	28	70	2 Sesiones
Otras Personas	29	70	2 Sesiones
Otras Personas	30	78	2 Sesiones

Fuente Propia

## RESULTADOS

Inicialmente se plantearon requerimientos no funcionales los que se verificaron su cumplimiento y desempeño de acuerdo a nuestra aplicación publicada.

### Rendimiento

Por medio de Chrome se evaluó el tiempo de despliegue de la aplicación para que esté operativa y lista para usarse.



Figura 5.4. Tiempo de carga de la aplicación

Fuente Lighthouse Audit – Chrome

### Disponibilidad

Se encuentra alojado en servidores de GoDaddy teniendo una disponibilidad del 99% garantizado según contrato pactado.

## Usabilidad

Por medio de Chrome se realizó una auditoria a la aplicación web y se verificó que cumpla con los lineamientos y estándares de buenas prácticas para desarrollo web publicado por google.

- ▶ **Elements Use Attributes Correctly**  
Screen readers and other assistive technologies require annotations to understand otherwise ambiguous content.

---

- ▶ `[accesskey]` values are unique. ✓

---

- ▶ `<audio>` elements contain a `<track>` element with `[kind="captions"]`. ✓

---

- ▶ Image elements have `[alt]` attributes. ✓

---

- ▶ `<input type="image">` elements have `[alt]` text. ✓

---

- ▶ No element has a `[tabindex]` value greater than 0. ✓

---

- ▶ Cells in a `<table>` element that use the `[headers]` attribute only refer to other cells of that same table. ✓

---

- ▶ `<th>` elements and elements with `[role="columnheader"/"rowheader"]` have data cells they describe. ✓

Figura 5.5. Elementos utilizan correctamente atributos HTML

Fuente Lighthouse Audit – Chrome

- ▶ **ARIA Attributes Follow Best Practices**  
Screen readers and other assistive technologies require annotations to understand otherwise ambiguous content.
  - ▶ [aria-\*] attributes match their roles. ✓
  - ▶ [role]s have all required [aria-\*] attributes. ✓
  - ▶ Elements with [role] that require specific children [role]s, are present. ✓
  - ▶ [role]s are contained by their required parent element. ✓
  - ▶ [role] values are valid. ✓
  - ▶ [aria-\*] attributes have valid values. ✓
  - ▶ [aria-\*] attributes are valid and not misspelled. ✓

Figura 5.6. Mejores prácticas en uso de atributos HTML

Fuente Lighthouse Audit - Chrome

- ▶ **Elements Have Discernable Names**  
Screen readers and other assistive technologies require annotations to understand otherwise ambiguous content.
  - ▶ Buttons have an accessible name. ✓
- ▶ **Elements Describe Contents Well**  
Screen readers and other assistive technologies require annotations to understand otherwise ambiguous content.
  - ▶ The page contains a heading, skip link, or landmark region. ✓
  - ▶ Document has a <title> element. ✓
  - ▶ <frame> or <iframe> elements have a title. ✓
  - ▶ Form elements have associated labels. ✓
  - ▶ Presentational <table> elements avoid using <th>, <caption> or the [summary] attribute. ✓
  - ▶ <object> elements have [alt] text. ✓
  - ▶ <video> elements contain a <track> element with [kind="captions"]. ✓
  - ▶ <video> elements contain a <track> element with [kind="description"]. ✓

Figura 5.7. Uso correcto de nombres en atributos HTML

Fuente Lighthouse Audit - Chrome

- ▶ **Elements Are Well Structured**  
Screen readers and other assistive technologies require annotations to understand otherwise ambiguous content.
  - ▶ `<d1>`'s contain only properly-ordered `<dt>` and `<dd>` groups, `<script>` or `<template>` elements. ✓
  - ▶ Definition list items are wrapped in `<d1>` elements. ✓
  - ▶ `[id]` attributes on the page are unique. ✓
  - ▶ Lists contain only `<li>` elements and script supporting elements (`<script>` and `<template>`). ✓
  - ▶ List items (`<li>`) are contained within `<ul>` or `<ol>` parent elements. ✓
- ▶ **Page Specifies Valid Language**  
Screen readers and other assistive technologies require annotations to understand otherwise ambiguous content.
  - ▶ `<html>` element has a `[lang]` attribute. ✓
  - ▶ `<html>` element has a valid value for its `[lang]` attribute. ✓
  - ▶ `[lang]` attributes have a valid value. ✓

Figura 5.8. Estructuración correcta y lenguaje especificado por pagina

Fuente Lighthouse Audit - Chrome

- ▶ **Meta Tags Used Properly**  
Screen readers and other assistive technologies require annotations to understand otherwise ambiguous content.
  - ▶ The document does not use `<meta http-equiv="refresh">`. ✓
  - ▶ `[user-scalable="no"]` is not used in the `<meta name="viewport">` element and the `[maximum-scale]` attribute is not less than 5. ✓

Figura 5.9. Uso correcto de etiquetas portabilidad en dispositivos

Fuente Lighthouse Audit - Chrome

▶ Properly size images	
▶ Offscreen images	
▶ Enable text compression	
▶ Avoids enormous network payloads: Total size was 450 KB (target: < 1.600 KB)	100
▶ Avoids an excessive DOM size: 106 nodes (target: < 1.500 nodes)	100
▶ User Timing marks and measures: 0	

Figura 5.10. Tiempo de carga de la aplicación

Fuente Lighthouse Audit – Chrome

### Concurrencia

Muestra del dashboard de control de grafico de nuestra aplicación en los servidores, mostrando siempre una respuesta óptima.

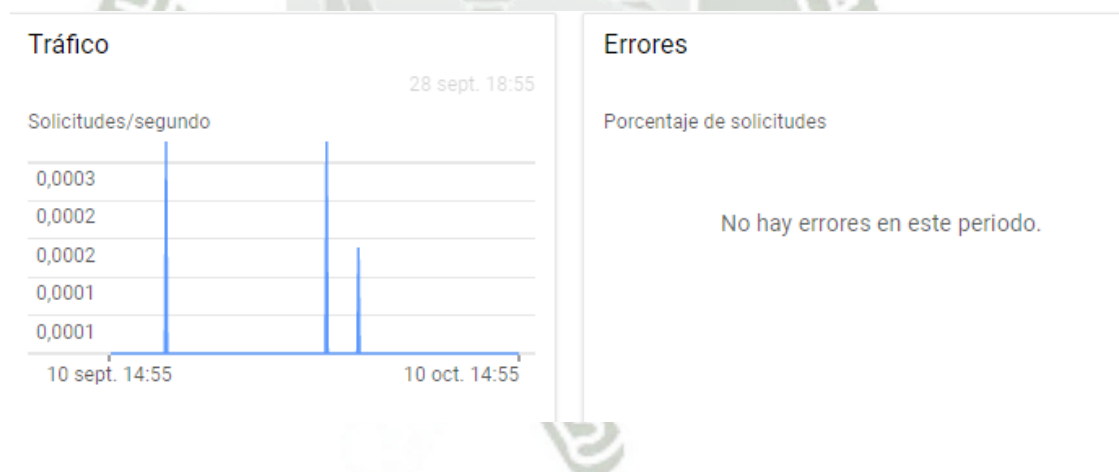


Figura 5.11. Trafico de la aplicación en el periodo 10 Set – 10 Oct

Fuente Lighthouse Audit – Chrome

## Portabilidad

Capturas de pantalla de los SO más comerciales en la actualidad, mostrando la aplicación en funcionamiento.

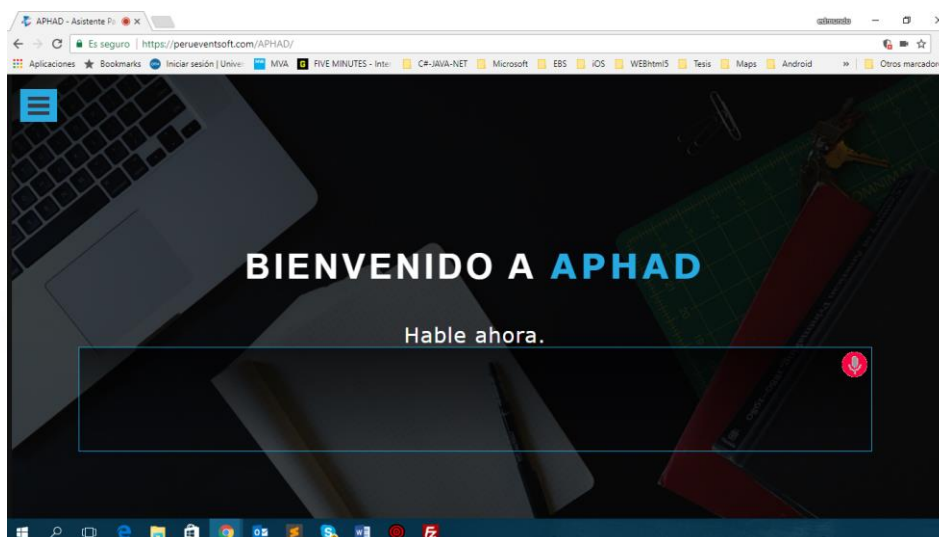


Figura 5.12. APHAD ejecutándose en Windows

Fuente Propia

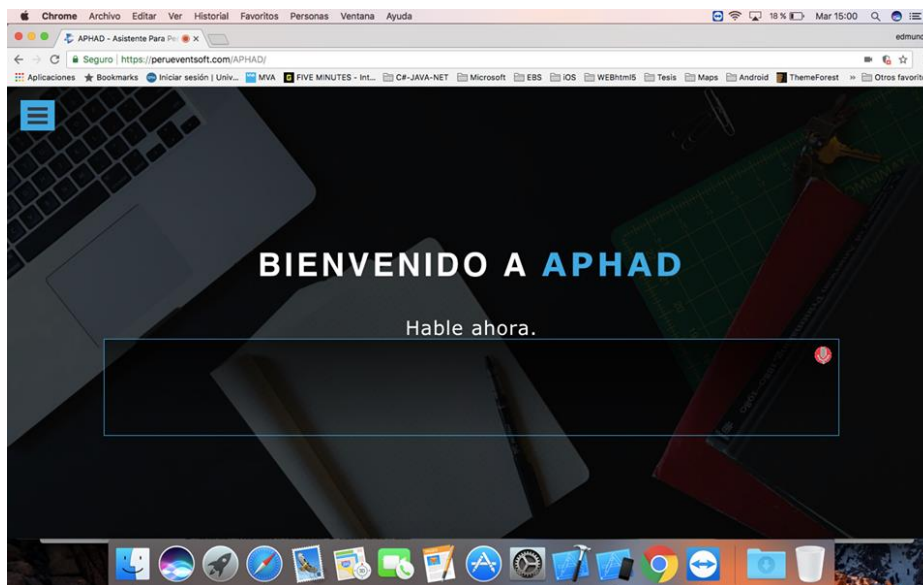


Figura 5.13. APHAD ejecutándose en iOS Sierra

Fuente Propia

Para lograr el objetivo planteado en el inicio de la tesis, se hizo un análisis estadístico de las encuestas que fueron recolectadas de las personas que hicieron uso de este Asistente para Personas con Habilidades Diferentes (APHAD), al finalizar las capacitaciones en las diferentes Instituciones donde se hicieron las pruebas.

La encuesta fue aplicada a diversos grupos de personas con diferentes edades. Con la finalidad de tener una opinión general de los encuestados tras haber asistido a las sesiones de aprendizaje del asistente web APHAD, en cómo influye en su labor diaria y en su educación. Los resultados obtenidos de la encuesta aplicada obtuvimos lo siguiente:

Se evaluó la satisfacción del usuario con la iteración de la aplicación teniendo como resultados que de 30 encuestas aplicadas el 63.3% estaba muy de acuerdo y el 36.7% estaba de acuerdo que la interacción era favorable para el usuario final.

**Tabla Resultado Encuesta Pregunta 1**

**¿Te gusta la interacción con la aplicación?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy de acuerdo	19	63,3	63,3	63,3
	De acuerdo	11	36,7	36,7	100,0
	Indeciso	0	0	0	0
	En desacuerdo	0	0	0	0
	Muy en desacuerdo	0	0	0	0
	Total	30	100,0	100,0	

Fuente Propia

■ Muy de acuerdo  
■ De acuerdo

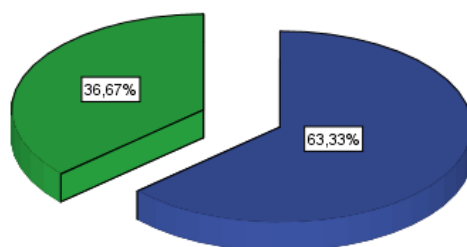


Figura 5.14. Gráfico resultado encuesta pregunta 1

Fuente Propia

Se evaluó si era necesario un entrenamiento previo para su uso obteniendo como resultado del 100%

**Tabla Resultado Encuesta Pregunta 2**

**¿Necesitó que se le indique como usar la aplicación?**

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido Si	30	100,0	100,0	100,0
No	0	0	0	0

■ Si

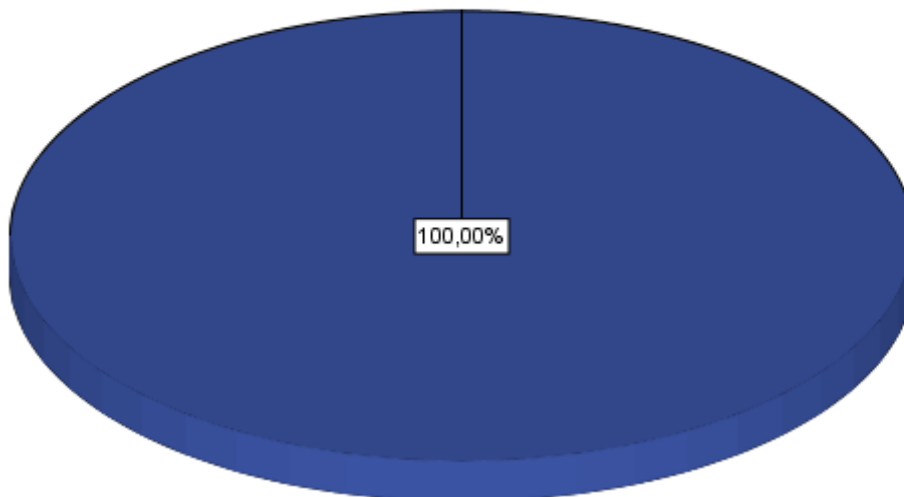


Figura 5.15. Grafico resultado encuesta pregunta 2

Fuente Propia

Se evaluó la facilidad que tuvo el usuario al interactuar con la aplicación teniendo como resultado un 66.7% que estaba muy de acuerdo con la facilidad de interacción y un 33.3% que estaban de acuerdo.

**Tabla Resultado Encuesta Pregunta 3**

**¿Fue fácil interactuar con la aplicación?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy de acuerdo	20	66,7	66,7	66,7
	De acuerdo	10	33,3	33,3	100,0
	Indeciso	0	0	0	0
	En desacuerdo	0	0	0	0
	Muy en desacuerdo	0	0	0	0
	Total	30	100,0	100,0	

■ Muy de acuerdo  
■ De acuerdo

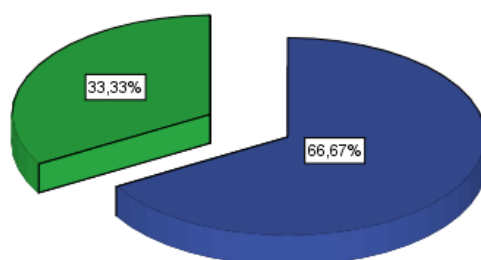


Figura 5.16. Gráfico resultado encuesta pregunta 3

Fuente Propia

Se evaluó si los usuarios cambiarían algo a la aplicación o agregarían algo nuevo. Obtuvimos un 23.3% que indicó realizar un cambio y un 76.7% no realizar cambio a la aplicación.

**Tabla Resultado Encuesta Pregunta 4**

**¿Cambiaría algo en la aplicación?**

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido Si	7	23,3	23,3	23,3
No	23	76,7	76,7	100,0
Total	30	100,0	100,0	

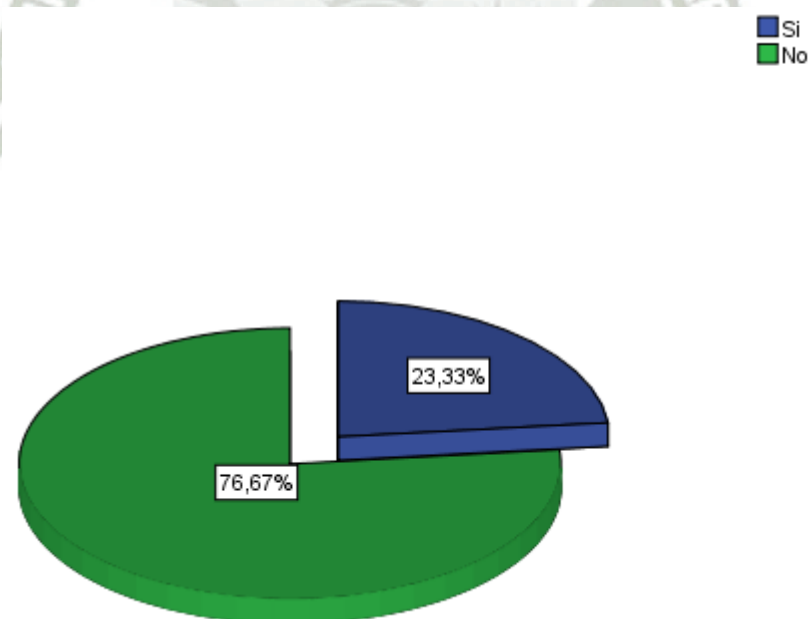


Figura 5.17. Gráfico resultado encuesta pregunta 4

Fuente Propia

Se evaluó si el usuario final recomendaría el uso de dicha aplicación a familiares o amigos obteniendo como resultados el 100% que si lo recomendarían.

**Tabla Resultado Encuesta Pregunta 5**

**¿Recomendaría su uso a familiares y amigos?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Si	30	100,0	100,0	100,0
	No	0	0	0	0

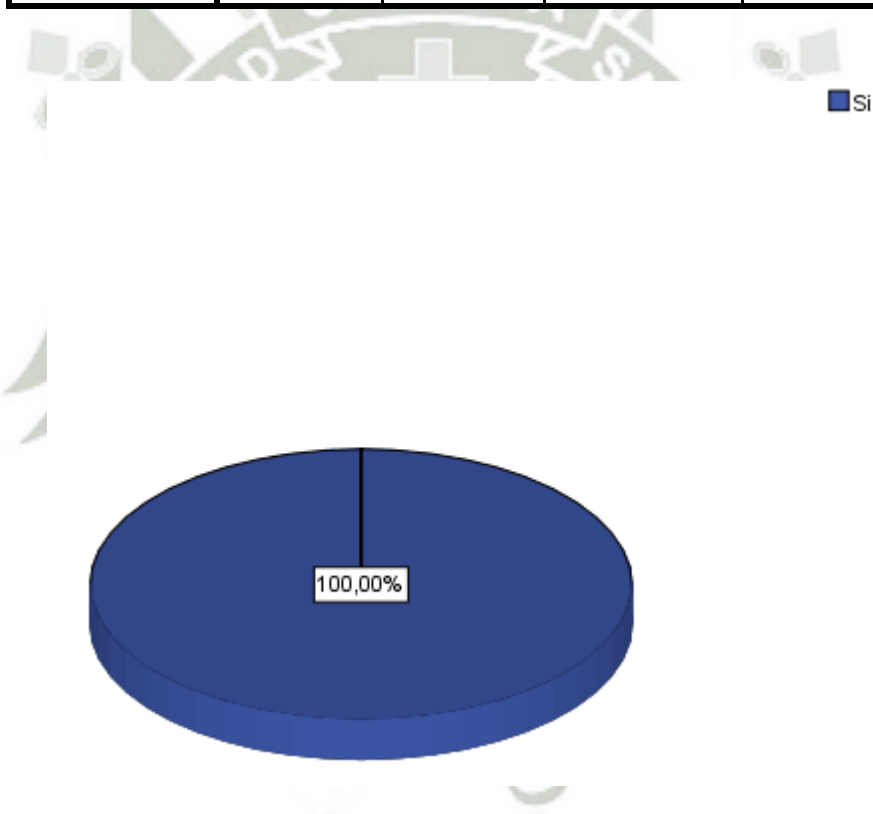


Figura 5.18. Gráfico resultado encuesta pregunta 5

Fuente Propia

Se evaluó la frecuencia con la que utilizaría la aplicación obteniendo como resultados un 56.7% siempre que se conecten a internet, 33.3% por lo general lo utilizarían al conectarse a internet y un 10% que lo haría esporádicamente.

**Tabla Resultado Encuesta Pregunta 6**

**¿Con que frecuencia utilizaría la aplicación?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Siempre	17	56,7	56,7	56,7
	Por lo general	10	33,3	33,3	90,0
	De vez en cuando	3	10,0	10,0	100,0
	Rara vez	0	0	0	0
	Nunca	0	0	0	0
	Total	30	100,0	100,0	

■ Siempre  
■ Por lo general  
■ De vez en cuando

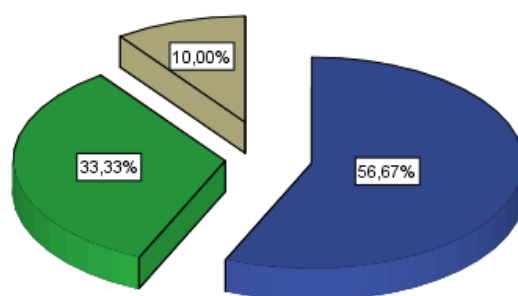


Figura 5.19. Gráfico resultado encuesta pregunta 6

Fuente Propia

Se evaluó si la aplicación lo consideraban eficaz y se obtuvo como resultado que un 56.7% estaba muy de acuerdo con la eficacia y un 43.3% estaba de acuerdo.

**Tabla Resultado Encuesta Pregunta 7**

**¿Consideras que la aplicación es eficaz?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy de acuerdo	17	56,7	56,7	56,7
	De acuerdo	13	43,3	43,3	100,0
	Indeciso	0	0	0	0
	En desacuerdo	0	0	0	0
	Muy en desacuerdo	0	0	0	0
	Total	30	100,0	100,0	

■ Muy de acuerdo  
■ De acuerdo

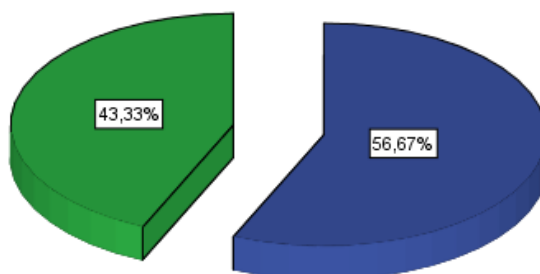


Figura 5.20. Gráfico resultado encuesta pregunta 7

Fuente Propia

Se evaluó si la aplicación le ayudaba a resolver sus dudas obteniendo como resultados 53.3% muy de acuerdo y un 46.7 de acuerdo a que la aplicación les ayuda.

**Tabla Resultado Encuesta Pregunta 8**

**¿La aplicación resuelve tus dudas?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy de acuerdo	16	53,3	53,3	53,3
	De acuerdo	14	46,7	46,7	100,0
	Indeciso	0	0	0	0
	En desacuerdo	0	0	0	0
	Muy en desacuerdo	0	0	0	0
	Total	30	100,0	100,0	

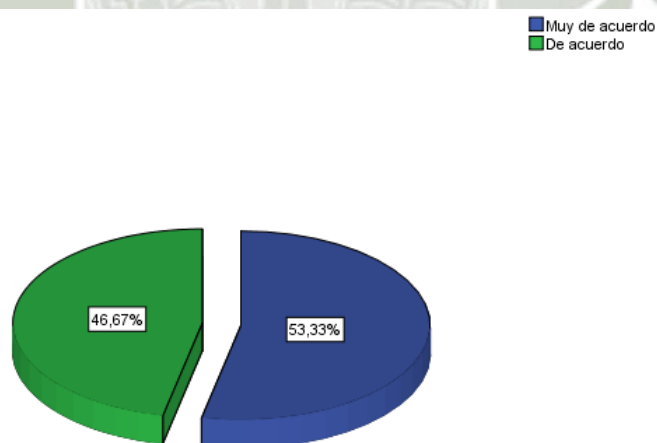


Figura 5.21. Gráfico resultado encuesta pregunta 8

Fuente Propia

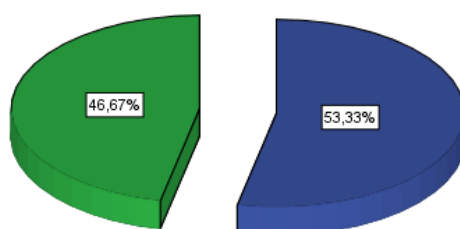
Se evaluó si para los usuarios les ayudaba a informarse de una manera fácil sobre búsquedas realizadas en internet. Se obtuvo como resultado que el 53.3% está muy de acuerdo que los ayuda de manera fácil y un 46.7% que está de acuerdo.

**Tabla Resultado Encuesta Pregunta 9**

**¿Crees que esta aplicación ayudará a informarte de una manera fácil?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy de acuerdo	16	53,3	53,3	53,3
	De acuerdo	14	46,7	46,7	100,0
	Indeciso	0	0	0	0
	En desacuerdo	0	0	0	0
	Muy en desacuerdo	0	0	0	0
	Total	30	100,0	100,0	

■ Muy de acuerdo  
■ De acuerdo



**Figura 5.22. Gráfico resultado encuesta pregunta 9**

Fuente Propia

Se evaluó si los usuarios consideran que la aplicación es importante para su uso diario obteniendo como resultados 56.7% que consideran que es muy importante, un 40% que si es importante para su uso y un 3.3% que es moderadamente importante.

**Tabla Resultado Encuesta Pregunta 10**

**¿Cree que la aplicación es importante para el uso diario?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy importante	17	56,7	56,7	56,7
	Importante	12	40,0	40,0	96,7
	Moderadamente importante	1	3,3	3,3	100,0
	Poca importancia	0	0	0	0
	Sin importancia	0	0	0	0
	Total	30	100,0	100,0	

■ Muy importante  
■ Importante  
■ Moderadamente importante

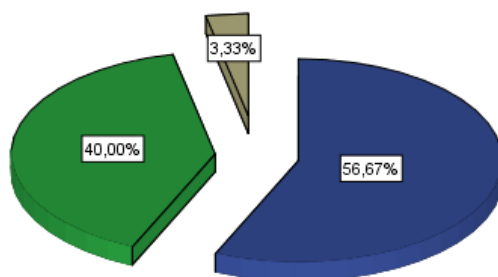


Figura 5.23. Gráfico resultado encuesta pregunta 10

Fuente Propia

Se evaluó si consideran que la aplicación es fácil de usar obteniendo como resultado un 60% que estaba muy de acuerdo y un 40% que si está de acuerdo que la aplicación es fácil de usar.

**Tabla Resultado Encuesta Pregunta 11**

**¿Cree que la aplicación es fácil de utilizar?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy de acuerdo	18	60,0	60,0	60,0
	De acuerdo	12	40,0	40,0	100,0
	Indeciso	0	0	0	0
	En desacuerdo	0	0	0	0
	Muy en desacuerdo	0	0	0	0
	Total	30	100,0	100,0	

■ Muy de acuerdo  
■ De acuerdo

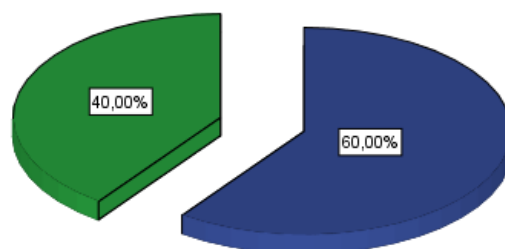


Figura 5.24. Grafico resultado encuesta pregunta 11

Fuente Propia

Se evaluó la utilidad de los comandos predeterminados obteniendo como resultados 60% que está muy de acuerdo con su utilidad, un 33.3% que está de acuerdo y un 6.7% que está indeciso a responder la pregunta.

**Tabla Resultado Encuesta Pregunta 12**

**¿Cree que los comandos de detener y nueva búsqueda son útiles?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy de acuerdo	18	60,0	60,0	60,0
	De acuerdo	10	33,3	33,3	93,3
	Indeciso	2	6,7	6,7	100,0
	En desacuerdo	0	0	0	0
	Muy en desacuerdo	0	0	0	0
	Total	30	100,0	100,0	

■ Muy de acuerdo  
■ De acuerdo  
■ Indeciso

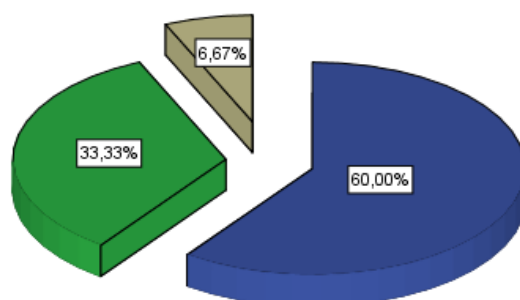


Figura 5.25. Gráfico resultado encuesta pregunta 12

Fuente Propia

Se evaluó que tan eficiente creen los usuarios que es la aplicación obteniendo como resultados 50% que es Muy Buena, 46.7% que es buena y un 3.3% que es regular.

**Tabla Resultado Encuesta Pregunta 13**

**¿Cree que la aplicación es eficiente?**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	Muy buena	15	50,0	50,0	50,0
	Buena	14	46,7	46,7	96,7
	Regular	1	3,3	3,3	100,0
	Mala	0	0	0	0
	Muy Mala	0	0	0	0
	Total	30	100,0	100,0	

■ Muy buena  
■ Buena  
■ Regular

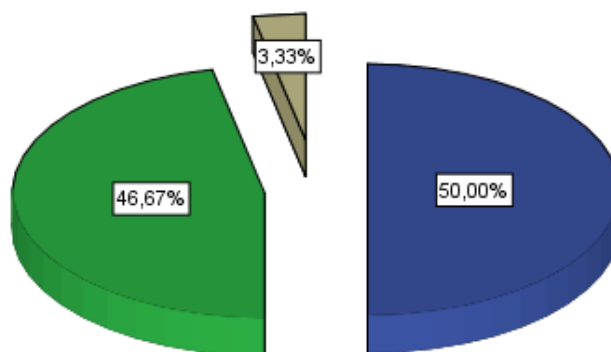


Figura 5.26. Gráfico resultado encuesta pregunta 13

Fuente Propia

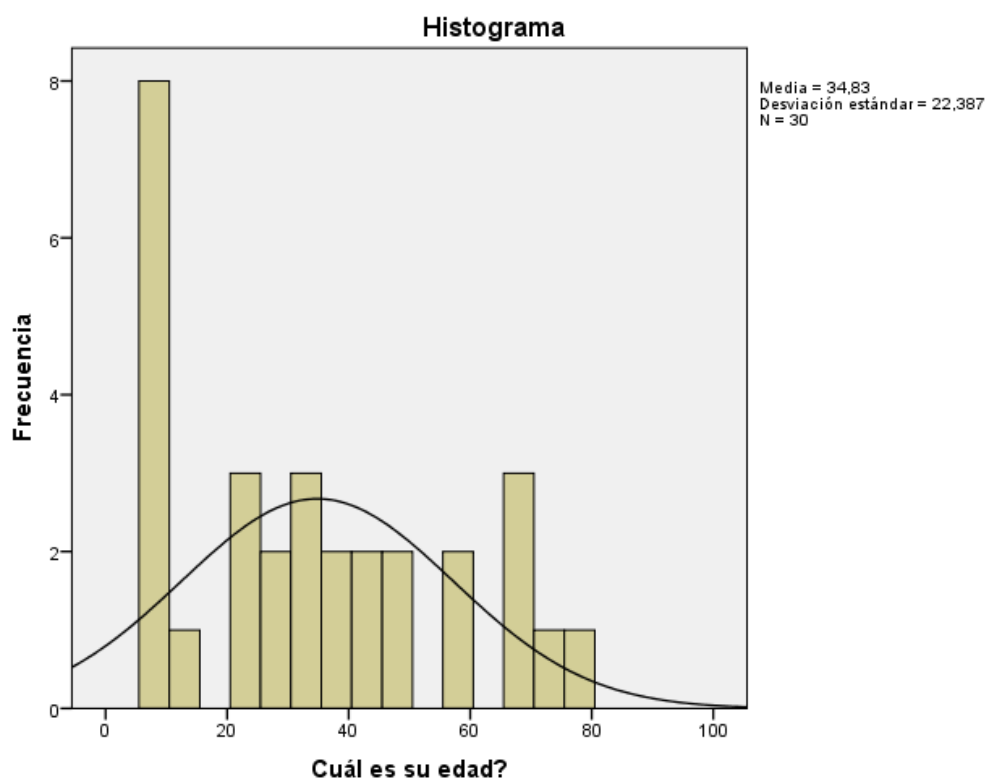
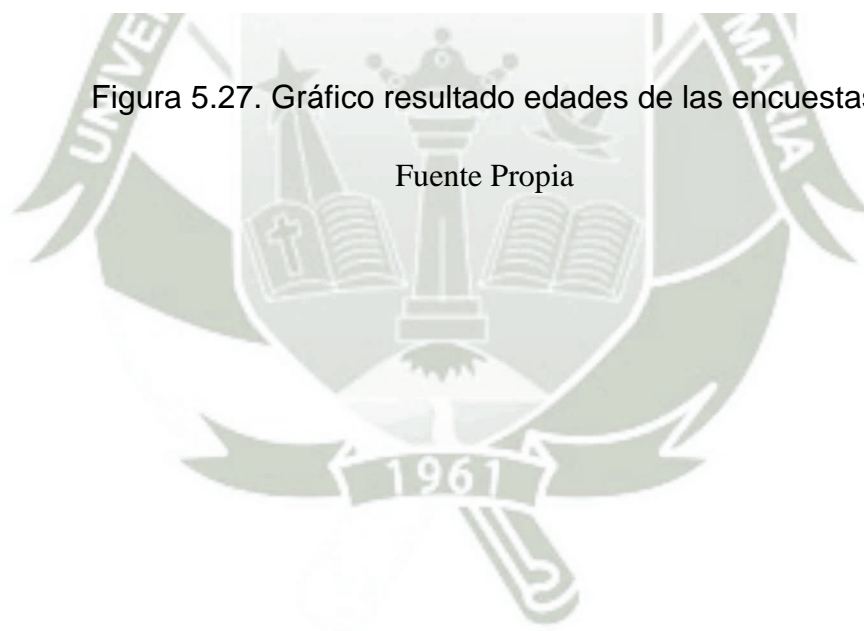


Figura 5.27. Gráfico resultado edades de las encuestas

Fuente Propia



## CONCLUSIONES

### PRIMERA:

Se implementó el Asistente de voz basado en web para personas con visión subnormal, incluyendo así a estas personas a la era tecnológica por medio de búsquedas por voz en internet.

### SEGUNDA:

Se diseñó y desarrolló un Asistente de voz basado en Web para Personas con visión subnormal y se logró integrar de manera satisfactoria la búsqueda con el motor de Google, la API de sintetizador de voz y la API de reconocimiento de voz obteniendo una interacción e integración fluida de cara al usuario. También se agregó un diseño responsive para el Asistente de voz basado en Web y se logró que este asistente se encuentra alojado en servidores alquilados de EEUU con una disponibilidad 24/7 para el uso de personas con visión subnormal y personas en general.

### TERCERA:

Se diseñó una interfaz sencilla y fácil de utilizar para el asistente de voz basado en web sin tener complicaciones cuando el usuario final utilice el asistente logando pasar de manera satisfactoria la auditoria de ARIA (Accessible Rich Internet Applications Suite) y mejores prácticas de SEO (Search Engine Optimization).

**CUARTA:**

Se desarrolló e implementó el asistente de voz basado en web de manera que se logró la interacción con las palabras claves sea entendible y fluida a la vez para el usuario final.

**QUINTA:**

Se realizó el despliegue de dicho asistente de voz basado en web con la finalidad de realizar pruebas de funcionalidad y eficiencia con las personas encuestadas, y evaluarlas para determinar su nivel de satisfacción al analizar el asistente de voz basado en web.

**SEXTA:**

Se ayudó y logro que las personas adquieran una mayor autonomía en la búsqueda de información y navegación por la web de una manera interactiva y fácil.

## RECOMENDACIONES

### PRIMERA

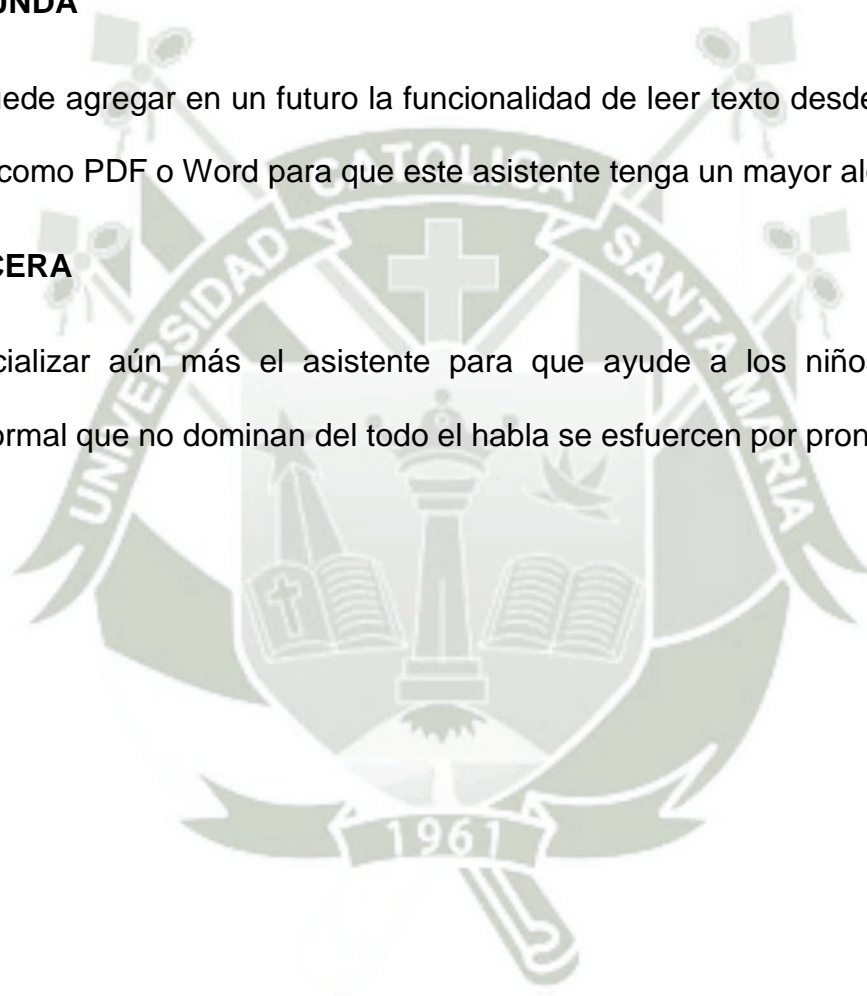
Si bien este asistente de voz basado en web es responsive, en algunos móviles tiene la dificultad en la interacción para lo cual se podría implementar a futuro una versión nativa para móviles.

### SEGUNDA

Se puede agregar en un futuro la funcionalidad de leer texto desde archivos de texto como PDF o Word para que este asistente tenga un mayor alcance.

### TERCERA

Especializar aún más el asistente para que ayude a los niños con visión subnormal que no dominan del todo el habla se esfuercen por pronunciar bien.



## BIBLIOGRAFIA

1. Adrián Alonso Vega (2013). Responsive Web Design: Interfaces Web Adaptables al dispositivo empleando HTML5 y CSS3.
2. Agüero, P. D. (2012). Síntesis de voz aplicada a la traducción voz a voz.
3. Anaya, R. (2006). Una visión de la enseñanza de la Ingeniería de Software como apoyo al mejoramiento de las empresas de software . Revista Universidad EAFIT , 60-76.
4. Andrés, G. (2011). Sintetizador de Voz Basado en LPC.
5. Cobo Ángel, P. G. (2005). PHP Y MYSQL Tecnologías para el desarrollo de aplicaciones Web. Ediciones Días de Santos.
6. Cornejo, J. E. (11 de 22 de 2016). DocIRS. Obtenido de DocIRS: <http://www.docirs.com/uml.htm>
7. D'Haro L.F., R. C.-S.-G. (2005). Plataforma de generación semiautomática de sistemas de diálogo multimodales y multilingües: Proyecto GEMINI.
8. Dimes, T. (2015). Conceptos basucos de Scrum: Desarrollo de software agile y manejo de proyectos agile. BabelCube.
9. Enrique, H. O. (2002). El Lenguaje Unificado de Modelado. Valencia.
10. Flores, L. V. (2001). Síntesis en Español Mexicano con el Método de Selección de Unidades de Longitud Variable.

11. Flores, L., Vargas, A., Olivier, A., Kirschning, I., & Cervantes, O. (2001). Síntesis en Español Mexicano con el Método de Selección de Unidades de Longitud Variable.
12. Florian Hinterleitner, S. Z.-P. (2015). On the use of automatic speech recognizers for the quality and intelligibility prediction of synthetic speech. Quality and Usability Lab, 10-13.
13. Freeman, J., & Skapura. (1993). Redes Neuronales. Algoritmos, aplicaciones y técnicas de propagación. Addison-Wesley, 306.
14. Fuentes, C. V. (2012). "Diseño Y Desarrollo De Un Software De Síntesis De Voz Para El Español De Colombia Aplicado A La Comunicación A Través De Dispositivos Móviles.
15. Glen Shires, Hans Wennborg. (2012). Web Speech API. Web Speech API, 2-4.
16. James Rumbaugh (2007). El Lenguaje Unificado del Modelado. Addison Wesley Edición 2.
17. Lawrence P., S. (2002). Ingeniería de software. Teoría y práctica. Prentice Hall.
18. Martin Fowler(2000). Uml Gota A Gota. Prentice Hall
19. Martin Fowler, Kedall Scott (2003). UML Distilled: A Brief Guide to the Standard Object Modeling Language. Pearson Education (Us); Edición: 3 ed.

20. Mayer, Bunge. (2004). Panorama de la industria del software en latinoamerica. Brasil: Brasil Informatica LTDA.
21. Nurzhan Nurseitov, Michael Paulson, Randall Reynolds, Clemente Izurieta (2015). Comparison of JSON and XML Data Interchange Formats: A Case Study Department of Computer Science Montana State University - USA
22. Pressman, R. S. (2006). Ingeniería del software: un enfoque práctico. En R. S. Pressman, Ingeniería del software: un enfoque práctico (págs. 50-51). Mexico: Mcgraw-HILL Interamericana Editores, S.A.
23. Pressman, R. S. (2010). Ingeniería del software: un enfoque práctico. En R. S. Pressman, Ingeniería del software: un enfoque práctico (págs. 6-7). Mexico: Mcgraw-Hill Interamericana Editores, S.A.
24. S.L., I. y. (2001). JavaScript. Innova.
25. Spona, H. (2010). Programación de bases de datos con MYSQL y PHP. marcombo.
26. Tordera Yllescas, J. C. (2011). Lingüística computacional. Tratamiento del habla. Valencia: Universtitat de València.
27. Tulsa. (2000). Statistica for Windows [Computer program manual].
28. Valdes, L. E. (2004). Situación Actual de la Informática en Colombia. Bogota: Informe del CATI.
29. Vértice. (2009). Diseño Básico de Páginas Web en HTML. Editorial Vértice.





## GLOSARIO DE TERMINOS

### TERMINOS

### CONCEPTO/DEFINICION

#### **Tecnología**

Se define como el conjunto de conocimientos y técnicas que, aplicados de forma lógica y ordenada, permiten al ser humano modificar su entorno material o virtual para satisfacer sus necesidades, esto es, un proceso combinado de pensamiento y acción con la finalidad de crear soluciones útiles.

#### **Plataforma Web**

Una plataforma incluye elementos adicionales a la página web tales como aplicaciones, formularios, convertidores y alguna otra solución específica para la necesidad del cliente y el mercado.

#### **Metodología Ágil**

Método que permite incorporar cambios con rapidez en el desarrollo de software. En muchas ocasiones, los modelos de gestión tradicionales no sirven para afrontar un reto que hoy en día resulta fundamental: incorporar cambios con rapidez y en cualquier fase del proyecto.

<b>API</b>	Es un conjunto de subrutinas, funciones y procedimientos que ofrecen bibliotecas para ser utilizados por otros softwares como una capa de abstracción.
<b>Reconocimiento de voz</b>	Es una disciplina de la inteligencia artificial (IA) que tiene como objetivo principal permitir la comunicación hablada entre seres humanos y computadoras.
<b>Interacción</b>	Es un vocablo que describe una acción que se desarrolla de modo recíproco entre dos o más organismos, objetos, agentes, unidades, sistemas, fuerzas o funciones.
<b>Retrospectiva</b>	Se hace referencia a observar hacia atrás, por lo tanto, es aquello que tiene en cuenta un desarrollo o un trabajo que se realizó en el pasado.
<b>Software</b>	Es un conjunto de programas, instrucciones y reglas informáticas que permiten ejecutar distintas tareas en una computadora.

**Voz Sintetizada**

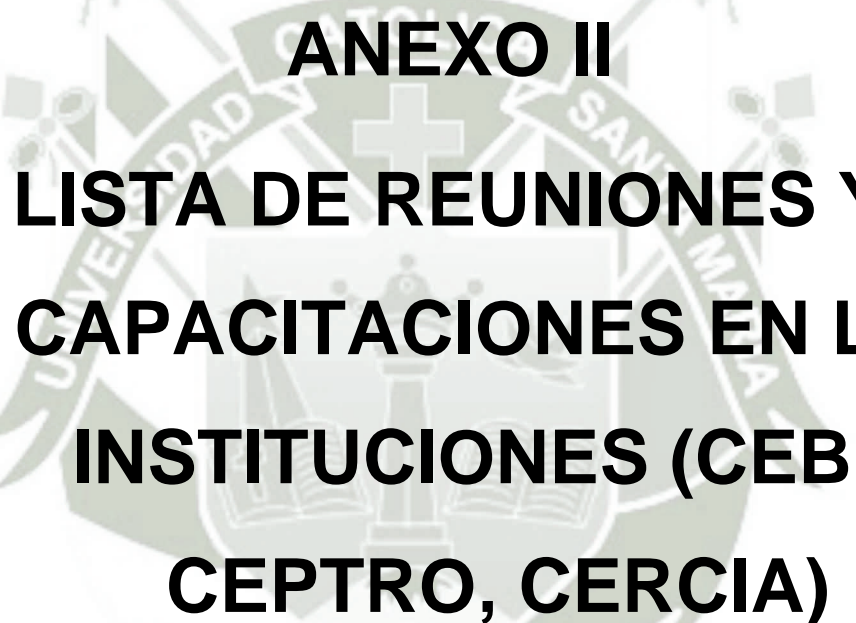
Es la producción artificial del habla, el sistema computarizado que es usado con este propósito es llamado computadora de habla o sintetizador de voz que puede ser implementados en productos software o hardware, normalmente convierte el lenguaje de texto normal en habla, otros sistemas recrean la representación simbólica lingüística como transcripciones fonéticas en habla.

**Búsqueda por Voz**

Le permite al usuario usar comandos de voz para buscar en internet o en un dispositivo portátil por medio de un micrófono, en la actualidad la búsqueda de voz es bien usada, en un sentido estricto, en asistencia de directorio o búsqueda local.

**Visión Subnormal**

Se define como baja visión a la agudeza visual disminuida o al campo visual restringido, como consecuencia de algún desorden en el sistema visual, el cual no puede ser corregido con un antejo convencional ni con intervención quirúrgica.



**ANEXO II**  
**LISTA DE REUNIONES Y**  
**CAPACITACIONES EN LAS**  
**INSTITUCIONES (CEBE,**  
**CEPTRO, CERCIA)**

**Reunión 1:**

<b>Fecha:</b>	20/06/2017
<b>Duración:</b>	1 hora
<b>UE:</b>	Srta. Lucy
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Instituciones CEBE – Nuestra Señora del Pilar
<b>Objetivos:</b>	Explicación y Autorización para realizar las pruebas con los niños de la Institución.

En esta reunión nos pusimos en contacto con la señorita Lucy de la Institución Nuestra Señora del Pilar, para explicarle y enseñarle sobre el Asistente APHAD así enseñándole su funcionamiento, La señorita Lucy quedo muy encantada así quedando los días los cuales realizaremos las siguientes capacitaciones con los niños para enseñarles el funcionamiento del sistema, así coordinando las 3 fechas.

**Reunión 2:**

<b>Fecha:</b>	20/06/2017
<b>Duración:</b>	1 hora
<b>UE:</b>	Sra. directora Ana Luz Chávez Machado
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Instituciones CERCIA – Centro de Rehabilitación para Adultos Ciegos
<b>Objetivos:</b>	Explicación y Autorización para realizar las pruebas con las personas de la Institución.

En esta reunión nos pusimos en contacto con la directora la Sra. Ana Luz Chávez Machado de la Institución Centro de Rehabilitación para adultos ciegos, para explicarle y enseñarle sobre el asistente APHAD así enseñándole su funcionamiento, la Sra. Ana Luz quedo muy satisfecha con el asistente, así

llamando al Profesor de Informática para poder coordinar las fechas para las capacitaciones de las personas con visión subnormal, así coordinando las 3 fechas.

### Reunión 3:

<b>Fecha:</b>	23/06/2017
<b>Duración:</b>	4 hora
<b>UE:</b>	Srta. Lucy, niños con visión subnormal
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Instituciones CEBE – Nuestra Señora del Pilar
<b>Objetivos:</b>	Primera capacitación en la Institución.

En esta reunión se coordinó con el profesor de informática así brindándonos acceso a sus computadoras, primero se formó grupos de acuerdo con la Srta. Lucy.

El primer grupo fueron niños de 8 años los cuales se les dio una charla sobre el funcionamiento y como podían hacer uso de este Asistente, había niños con visión baja y niños con visión subnormal, después de la charla se pasó a las pruebas con el asistente los niños empezaron a interactuar y al ver que el asistente APHAD le contestaban se quedaron sorprendidos porque no habían visto un asistente como ese que responda a sus necesidades, muchos de los niños hicieron preguntas como quien era Rayo McQueen y el asistente les daba información de acuerdo a su pregunta, así creando que los niños se aprendan las palabras claves del asistente para su interacción.

El segundo grupo eran más niños de 8 años, de igual manera se les dio la charla sobre el funcionamiento y pasamos a hacer las pruebas de igual manera los niños empezaron a preguntar como rayo McQueen o que es el día del campesino porque era la fecha que se aproximaba así tratando que los niños se aprendan las palabras claves.

El tercer grupo eran niños de 12 años, haciendo la charla repitiendo dos a tres veces ya que estos niños no solo tenían la discapacidad de la visión subnormal sino tenían problemas con la retención de lo explicado, se hicieron las pruebas respectivas y de igual manera preguntaron sobre el día del campesino y así tratando que los niños vayan aprendiendo las palabras claves del asistente.

Luego de concluir la primera sesión en la institución coordinamos con la Srta. Lucy para la próxima reunión.

#### Reunión 4:

<b>Fecha:</b>	26/06/2017
<b>Duración:</b>	1:30 hora
<b>UE:</b>	Sra. directora Ana Luz Chávez Machado
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Instituciones CERCIA – Centro de Rehabilitación para Adultos Ciegos
<b>Objetivos:</b>	Primera Capacitación.

En esta reunión se coordinó con el profesor de informática así brindándonos acceso a sus computadoras.

Así empezando con las personas con visión subnormal adultas dándoles una charla explicando el funcionamiento del Asistente, luego de ello pasamos hacer

las pruebas con cada uno de ellos, uno de ellos empezó realizando la búsqueda de poemas y al ver que el asistente le respondía se quedó muy asombrado así dando la confianza a las demás personas de empezar con la interacción, así cada uno de ellos fueron haciendo sus preguntas y quedándose admirados del asistente, luego de terminar con la capacitación muchos de ellos querían traer sus laptops para tener acceso al asistente, uno de los participantes hablando por medio del grupo nos dio las gracias por este asistente, ya que era mucha ayuda para ellos ya que el Jaws (un sintetizador que va leyendo por donde vas pasando el mouse) no era tan útil, y así quedando para la siguiente reunión el refuerzo de la primera capacitación.

#### Reunión 5:

<b>Fecha:</b>	27/06/2017
<b>Duración:</b>	1 hora
<b>UE:</b>	Sra. directora Ana Luz Chávez Machado
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Instituciones CERCIA – Centro de Rehabilitación para Adultos Ciegos
<b>Objetivos:</b>	Segunda Capacitación.

En la segunda reunión de igual manera conversamos con las personas con visión subnormal indicando que todo estaba claro hicimos algunas pruebas, muchas de las personas trajeron sus laptops para ponerles como acceso directo el Link de ingreso al Asistente, así pidiendo llevar acabo la encuesta la cual fue muy satisfactoria, igual manera las personas nos volvieron a dar las gracias por el gran apoyo que les estábamos brindando.

**Reunión 6:**

<b>Fecha:</b>	04/07/2017
<b>Duración:</b>	4 hora
<b>UE:</b>	Srta. Lucy, niños con visión subnormal
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Instituciones CEBE – Nuestra Señora del Pilar
<b>Objetivos:</b>	Segunda capacitación en la Institución.

En esta reunión se coordinó con el profesor de informática así brindándonos acceso a sus computadoras, primero se formó grupos de acuerdo con la Srta. Lucy.

El primer grupo fueron niños de 8 años y empezaron a interactuar con el asistente APHAD así reforzando la primera capacitación haciéndoles recuerdo de las palabras claves, muchos de los niños empezaron a interactuar mejor con el asistente.

El segundo grupo eran más niños de 8 años, de igual manera empezaron a interactuar con el asistente APHAD así reforzando la primera capacitación haciéndoles recuerdo de las palabras claves, muchos de los niños empezaron a interactuar mejor con el asistente.

El tercer grupo eran niños de 12 años, empezaron a interactuar con el asistente APHAD así reforzando la primera capacitación haciéndoles recuerdo de las palabras claves, muchos de los niños empezaron a interactuar mejor con el asistente.

Luego de concluir la segunda sesión en la institución coordinamos con la Srta. Lucy para la próxima reunión.

### Reunión 7:

<b>Fecha:</b>	05/07/2017
<b>Duración:</b>	4 hora
<b>UE:</b>	Srta. Lucy, niños con visión subnormal
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Instituciones CEBE – Nuestra Señora del Pilar
<b>Objetivos:</b>	Segunda capacitación en la Institución.

En esta reunión se coordinó con el profesor de informática así brindándonos acceso a sus computadoras, primero se formó grupos de acuerdo con la Srta. Lucy.

El primer grupo fueron niños de 8 años y empezaron a interactuar con el asistente APHAD así reforzando la segunda capacitación haciéndoles recuerdo de las palabras claves, hubo buenos resultados ya que los niños aprendieron el funcionamiento y las palabras claves del asistente.

El segundo grupo eran más niños de 8 años, de igual manera empezaron a interactuar con el asistente APHAD así reforzando la segunda capacitación haciéndoles recuerdo de las palabras claves, hubo buenos resultados ya que los niños aprendieron el funcionamiento y las palabras claves del asistente.

El tercer grupo eran niños de 12 años, empezaron a interactuar con el asistente APHAD así reforzando la segunda capacitación haciéndoles recuerdo de las

palabras claves, hubo buenos resultados ya que los niños aprendieron el funcionamiento y las palabras claves del asistente.

Luego de concluir la tercera sesión de la capacitación dimos por concluida el aprendizaje del asistente así pudiendo aplicar la encuesta y dándonos respuestas satisfactorias.

### Reunión 8:

<b>Fecha:</b>	18/07/2017
<b>Duración:</b>	1 hora
<b>UE:</b>	Sr. Carlos
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Institución CEPTRONuestra Señora de los Ángeles
<b>Objetivos:</b>	Coordinación para la capacitación en la Institución.

En esta reunión se conversó con el Director Carlos de la institución así quedando la fecha para poder llevar acabo la capacitación con las personas con visión subnormal, así coordinando para la primera capacitación.

### Reunión 9:

<b>Fecha:</b>	20/07/2017
<b>Duración:</b>	2 horas
<b>UE:</b>	Sr. Carlos, personas con visión subnormal
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Institución CEPTRONuestra Señora de los Ángeles
<b>Objetivos:</b>	Primera capacitación

En esta reunión el director Carlos nos brindó el apoyo del profesor de informática.

Empezamos dando la charla sobre el funcionamiento del asistente, luego se hicieron las pruebas correspondientes con las personas, en este grupo había una niña, jóvenes y personas adultas, así entendiendo muy rápido el funcionamiento de este asistente y quedando muy agradecidos porque ahora podrían hacer uso de este asistente y poder informarse solos de una manera fácil.

Luego de ello aplicamos las encuestas a las personas dándonos resultados muy satisfactorios.

#### Reunión 10:

<b>Fecha:</b>	23/07/2017
<b>Duración:</b>	2 horas
<b>UE:</b>	Otras Personas
<b>IS:</b>	Edmundo Aparicio, Katherin Collado
<b>Lugar:</b>	Casa de uno de los integrantes
<b>Objetivos:</b>	Primera capacitación

Se realizó pruebas con personas jóvenes, adultos, adultos mayores y personas de tercera edad así obteniendo en las encuestas buenos resultados, así indicando que este asistente web no solo es para personas con visión subnormal sino para todo público.



# **ANEXO III**

## **ENCUESTA APLICADA**

## ENCUESTA APLICADA

### Encuesta

Edad: \_\_\_\_ Sexo: \_\_\_\_

La presente encuesta tiene como finalidad obtener el nivel de satisfacción y ayuda al usuario al haber usado APHAD como un asistente de búsqueda por voz.

Llene y marque con letra legible y con total sinceridad cada pregunta.

1. ¿Te gusta la iteración con la aplicación?
  - a. Muy de acuerdo
  - b. De acuerdo
  - c. Indeciso
  - d. En desacuerdo
  - e. Muy en desacuerdo
2. Que comandos quisiera añadir a la aplicación \_\_\_\_
3. ¿Necesitó que le indiquen como usar la aplicación?
  - a. Sí
  - b. No
4. ¿Cuánto tiempo le tomó entender el funcionamiento de la aplicación? \_\_\_\_
5. ¿Le fue fácil interactuar con la aplicación?
  - a. Muy de acuerdo
  - b. De acuerdo
  - c. Indeciso
  - d. En desacuerdo
  - e. Muy en desacuerdo
6. ¿Cuál fue su primera búsqueda  
\_\_\_\_\_
7. ¿Cambiaría algo en la aplicación?
  - a. Sí
  - b. No
8. Si la respuesta fue Sí. ¿Qué le gustaría cambiar en la aplicación?

- 
9. ¿Recomendaría su uso a familiares y amigos?
- Si
  - No
10. ¿Con que frecuencia utilizaría la aplicación?
- Siempre
  - Por lo general
  - De vez en cuando
  - Rara vez
  - Nunca
11. ¿Consideras que la aplicación es eficaz?
- Muy de acuerdo
  - De acuerdo
  - Indeciso
  - En desacuerdo
  - Muy en desacuerdo
12. ¿La aplicación resuelve tus dudas?
- Muy de acuerdo
  - De acuerdo
  - Indeciso
  - En desacuerdo
  - Muy en desacuerdo
13. ¿Crees que esta aplicación ayudara a informarte de una manera fácil?
- Muy de acuerdo
  - De acuerdo
  - Indeciso
  - En desacuerdo
  - Muy en desacuerdo
14. ¿Cree que la aplicación es importante para el uso diario?
- Muy importante
  - Importante
  - Moderadamente importante

- d. Poca importancia
  - e. Sin importancia
15. ¿Cree que la aplicación es fácil de utilizar?
- a. Muy de acuerdo
  - b. De acuerdo
  - c. Indeciso
  - d. En desacuerdo
  - e. Muy en desacuerdo
16. ¿Cree que los comandos de detener y nueva búsqueda son útiles?
- a. Muy de acuerdo
  - b. De acuerdo
  - c. Indeciso
  - d. En desacuerdo
  - e. Muy en desacuerdo
17. ¿Cree que la aplicación es eficiente?
- a. Muy buena
  - b. Buena
  - c. Regular
  - d. Mala
  - e. Muy mala



### Sesiones Aplicadas en CEBE Nuestra Señora del Pilar





Sesiones en CERCIA





### Sesiones realizadas en CEPTRÓ Luz y Alegría

