**UNIVERSIDAD DE TARAPACÁ**



**FACULTAD DE INGENIERÍA**

Departamento de Ingeniería en Computación e Informática



Informe: Proyecto II

**A.C.E**

**“Asistente conversacional electrónico”**

**Autores:** Rodrigo Suaña

Anibal Laura

Mauricio Benavides

Felipe Crispin

**Profesor:** Diego Aracena P.

**Asignatura:** Proyecto 2

Arica, Chile 12 de Diciembre de 2022

# Historial de cambios

| **Fecha** | **Versión** | **Descripción** | **Autor(es)** |
| --- | --- | --- | --- |
| 08-09-2022 | 1.0 | Traspaso de la información ya definidas como la definición del problema y la solución. | Rodrigo Suaña |
| 13-09-2022 | 1.1 | Completación del panorama general y referencias | Rodrigo Suaña  Anibal laura  Felipe Crispin |
| 15-09-22 | 1.2 | Se finaliza el primer informe | Rodrigo Suaña  Anibal laura  Felipe Crispin |
| 27-09-2022 | 1.3 | Se finaliza las especificaciones de las funcionalidades principales del proyecto, al igual que los casos de uso | Rodrigo Suaña  Anibal Laura  Felipe Crispin |
| 14-10-2022 | 1.4 | Se finaliza el segundo avance del informe | Rodrigo Suaña  Anibal Laura  Felipe Crispin  Mauricio Benavides |
| 08-11-2022 | 1.5 | Corrección en base a la retroalimentación entregada por el profesor. | Mauricio Benavides  Anibal Laura |
| 22-11-2022 | 1.6 | Avance e integración de partes faltantes para la entrega final. | Mauricio Benavides  Anibal Laura  Rodrigo Suaña  Felipe Crispin |
| 07-12-2022 | 1.7 | Corrección y completación para la entrega final del informe. | Mauricio Benavides  Anibal Laura  Rodrigo Suaña  Felipe Crispin |

# 

# Tabla de contenidos

# 

[**Historial de cambios**](#_heading=h.gjdgxs) **2**

[**Tabla de contenidos**](#_heading=h.3znysh7) **3**

[**1. Panorama general**](#_heading=h.3dy6vkm) **5**

[**1.1. Resumen del proyecto:**](#_heading=h.1t3h5sf) **5**

[1.1.1. Introducción](#_heading=h.4d34og8) 5

[**1.1.2. Escenario del problema**](#_heading=h.5q8tqmf9t9t2) **6**

[1.1.3. Escenario de la solución](#_heading=h.17dp8vu) 7

[1.1.4. Propósito](#_heading=h.3rdcrjn) 8

[1.1.5. Objetivo general:](#_heading=h.rgz3ykogsit0) 8

[1.1.6. Objetivos específicos](#_heading=h.35nkun2) 8

[1.1.7. Restricciones](#_heading=h.44sinio) 8

[1.1.8. Entregables](#_heading=h.2jxsxqh) 8

[**3. Organización del proyecto**](#_heading=h.4i7ojhp) **9**

[3.1. Roles y responsabilidades](#_heading=h.1ci93xb) 9

[3.2. Mecanismos de comunicación](#_heading=h.3whwml4) 10

[**4. Planificación de los procesos de gestión**](#_heading=h.embpsfdug4nv) **11**

[4.1. Planificación inicial del proyecto](#_heading=h.3as4poj) 11

[4.1.1. Planificación de estimaciones](#_heading=h.1pxezwc) 11

[4.1.2. Planificación de recursos humanos](#_heading=h.49x2ik5) 11

[**4.2. Lista de actividades**](#_heading=h.tgdg6s7l3qsu) **13**

[4.2.1. Actividades de trabajo y asignación de tiempo](#_heading=h.147n2zr) 13

[4.2.3. Carta Gantt](#_heading=h.23ckvvd) 15

[4.3. Planificación de la gestión de riesgos](#_heading=h.ihv636) 16

[**5. Planificación de los procesos**](#_heading=h.w114xlr7f0mb) **17**

[**5.1 Análisis**](#_heading=h.y57555yyzqfu) **17**

[**5.2 Especificaciones**](#_heading=h.o2j3i9z4ycde) **17**

[**5.2.1 Funcionalidad 1: De voz a texto**](#_heading=h.qegr7p26zrms) **17**

[**1. El sistema capta la voz mediante el micrófono del smartphone.**](#_heading=h.ld7sixlwwv67) **17**

[**5.2.2 Funcionalidad 2: texto a voz**](#_heading=h.wue81i5fyzer) **17**

[**1. El sistema capta el texto escrito en pantalla.**](#_heading=h.jlmr74wl5z22) **17**

[**2. El sistema realiza el reconocimiento de texto.**](#_heading=h.sfkno2tw0x6) **17**

[**5.2.1 Funcionalidad 3: Chat**](#_heading=h.yw4v4b8sr4ss) **17**

[**1. El sistema capta las palabras escritas del teclado.**](#_heading=h.b61nfmnov559) **17**

[**5.3 Restricciones**](#_heading=h.qdazg75qm9e4) **17**

[**5.4 Casos de uso**](#_heading=h.qehkw7t6cefm) **18**

[**5.4.1 Descripción de los casos de uso**](#_heading=h.9n9jo7pmvxvk) **19**

[**5.4.2 Diagrama de secuencia**](#_heading=h.9g68sotddbwx) **21**

[**5.4.2.1 Funcionalidad 1: Diagrama de secuencia nvl 0.**](#_heading=h.crm7vxxosx0n) **22**

[**5.4.2.2 Funcionalidad 1: Diagrama de secuencia nvl 1.**](#_heading=h.n9enuqsf37iv) **23**

[**5.4.2.3 Funcionalidad 2: Diagrama de secuencia nvl 0.**](#_heading=h.ldw7y3lu1byo) **23**

[**5.4.2.4 Funcionalidad 2: Diagrama de secuencia nvl 1.**](#_heading=h.5mxyy7k7mtwy) **24**

[**5.4.2.5 Funcionalidad 3: Diagrama de secuencia nvl 0.**](#_heading=h.1s5akqyllxn5) **24**

[**5.4.2.6 Funcionalidad 3: Diagrama de secuencia nvl 1.**](#_heading=h.s13dchma61s3) **25**

[**5.5 Descripción de la arquitectura de sistema**](#_heading=h.ijzf3sd5n5so) **26**

[**5.5.1 Arquitectura del sistema:**](#_heading=h.ebvqcxkmu34d) **26**

[**5.5.2 Diagrama de flujo:**](#_heading=h.qp1a5yql4bgx) **27**

[**5.6 Requerimientos funcionales y no funcionales.**](#_heading=h.f4mrc35xqagg) **27**

[**5.6.1 Requerimientos funcionales:**](#_heading=h.ichr2lcqvfar) **27**

[**5.6.2 Requerimientos no funcionales:**](#_heading=h.iazy8vz409i5) **27**

[**5.7 Interfaz gráfica usuario (IGU)**](#_heading=h.oaozt4snii7u) **28**

[**5.8 Herramientas y técnicas**](#_heading=h.i6e02bjf3kwt) **32**

[**6. Planificación de procesos de soporte**](#_heading=h.et7nj7onylzx) **32**

[**6.1. Planificación de la documentación**](#_heading=h.vmim62xqtn6g) **32**

[**7. Implementación**](#_heading=h.pk1mdtlrfkm1) **33**

[**7.1. Plan de integración**](#_heading=h.mn18ca13maiu) **33**

[**7.2. Modelos de implementación**](#_heading=h.mc3pgm69v0i2) **33**

[**7.2.1. Traducción de voz a texto.**](#_heading=h.s4tc5krsawj1) **33**

[**7.2.2. Transformación de texto a voz.**](#_heading=h.5z25halfwp7o) **33**

[**8. Problemas y Soluciones**](#_heading=h.ghzcevmaoj3n) **34**

[**8.1 Problema N°1: Kivy MD**](#_heading=h.rnxhbqrhl2vm) **34**

[**Solución**](#_heading=h.typ7gtoxhai8) **34**

[**8.2 Problema N°2: React Native**](#_heading=h.aafd8y73xke3) **34**

[**Solución**](#_heading=h.vyklp4enuuw9) **34**

[**9. Implementación**](#_heading=h.97y5oiwumf5l) **35**

[**9.1 Módulos**](#_heading=h.jfsecvp74m76) **35**

[**9.1.1 Módulo speechToText**](#_heading=h.wie0fdkw38ev) **35**

[**9.1.2 Módulo textToSpeech**](#_heading=h.2x7nyeuk5l36) **36**

[**9.1.3 Módulo ChatText**](#_heading=h.9bfrw581tn6w) **37**

[**10. Pruebas**](#_heading=h.cz63fo6olip2) **38**

[**12. Conclusión**](#_heading=h.15otf5vlba8r) **41**

[**Referencias**](#_heading=h.geghxjibk8i) **42**

# 

# 1. Panorama general

## 1.1. Resumen del proyecto:

### 1.1.1. Introducción

La comunicación es trascendentalmente importante pero también es uno de los principales problemas que enfrentan las personas con alguna discapacidad auditiva, recurriendo a métodos poco eficientes o de difícil accesibilidad, de tal manera que el problema se agrava aún más.

Entonces como otra vía a la solución de este problema surge este proyecto, el cual consiste de una aplicación móvil que tiene como objetivo que las personas que presenten una discapacidad auditiva puedan comunicarse con otra persona, de una manera sencilla de tal manera que produzca que las demás personas se animen a probar este nuevo método y así la persona afectada no se sienta excluida del mundo exterior.

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

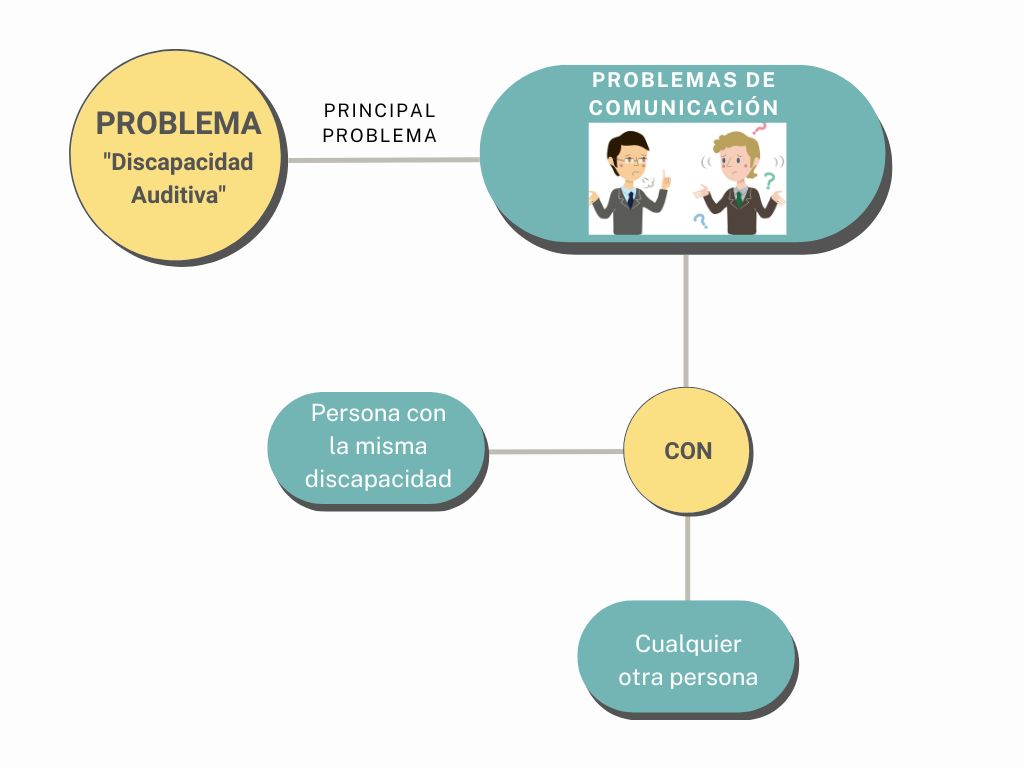
### 

### 

### 

### 1.1.2. Escenario del problema

Las personas que presentan discapacidad auditiva se enfrentan con distintos problemas al estar en lugares públicos, por ejemplo: una plaza, supermercado, una casa, entre otros. El problema principal al que se enfrentan estas personas es la “comunicación”, por ejemplo una persona que posea la misma discapacidad o cualquier otra persona. Si una persona normal le quiere hablar a la persona con discapacidad auditiva, el discapacitado no podrá comunicarse con esa persona desconocida, por distintas razones; una razón podría ser que el desconocido no tenga conocimiento del lenguaje de señas. Ahora si la persona con discapacidad auditiva quiere comunicarse con una persona que tenga su misma discapacidad, entonces no hay forma de entablar una conversación. Recalcar que la persona con discapacidad auditiva podría ser sordomudo.

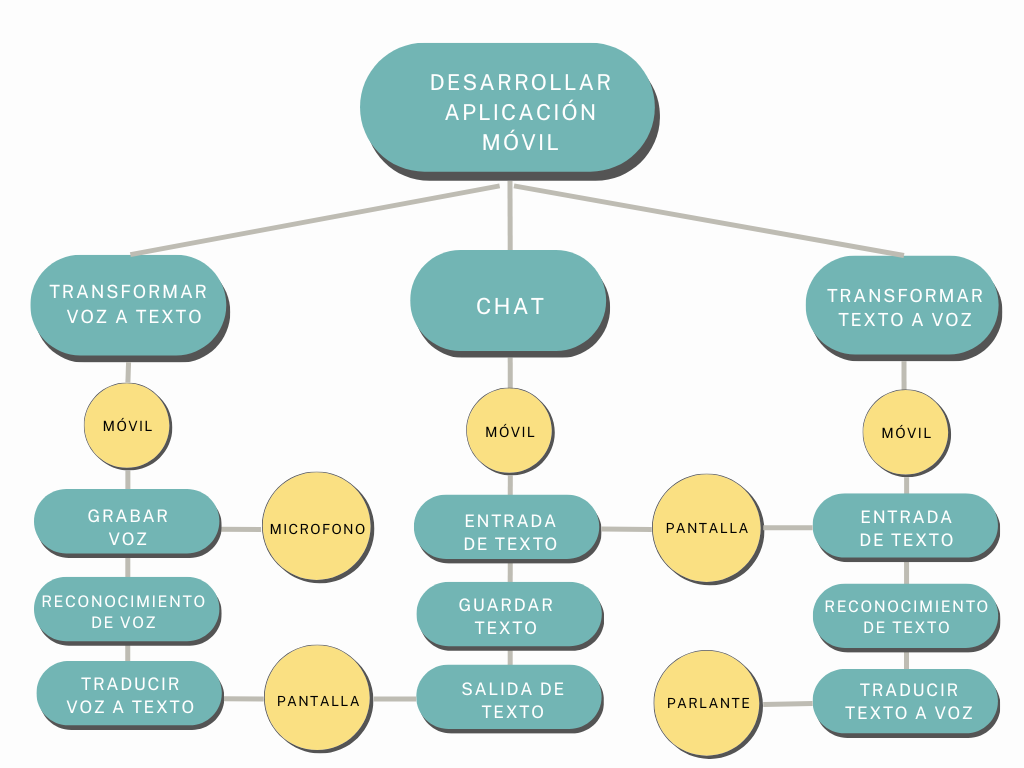


***Figura 1: Esquema del problema***

### 1.1.3. Escenario de la solución

Por esta razón, como propuesta de solución a este problema, se ha decidido desarrollar una aplicación móvil la cual sea más accesible para las personas con discapacidad auditiva. El cual cumpla con las siguientes funciones: transformar voz a texto, texto a voz y un chat de texto para el caso en el que se quieran comunicar dos personas sordomudas. De esta manera la persona con discapacidad auditiva lograría comunicarse con las demás personas.

Para esta solución se utilizan los diferentes periféricos que incluye un dispositivo móvil como: el micrófono para el emisor(una persona sin discapacidad) y la pantalla para el receptor que en este caso es el discapacitado, este último nos servirá para mostrar el texto ya traducido. También se podrá escribir texto el cual será transformado en audio. Finalmente el chat de texto. De esta manera se logra que la persona con una discapacidad auditiva pueda entablar una conversación con otra persona.



***Figura 2: Esquema solución***

### 1.1.4. Propósito

El propósito de este proyecto es el poder ayudar a las personas que presenten una discapacidad auditiva o sordomudo a que pueda comunicarse sin ningún problema con otra persona.

### 

### 1.1.5. Objetivo general:

Desarrollar una aplicación móvil para ayudar a las personas con discapacidad auditiva o sordo mudo, mediante la traducción del audio a una versión texto, traducción de texto a una versión de audio, y también el envío de mensajes en caso de que ambos participantes tengan alguna discapacidad auditiva.

### 1.1.6. Objetivos específicos

* Estudiar la problemática y evaluar las distintas soluciones propuestas.
* Analizar la solución seleccionada.
* Analizar librerías del lenguaje de programación.
* Desarrollar la aplicación.
* Realizar pruebas de funcionamiento de la aplicación.

### 1.1.7. Restricciones

En este apartado se mencionan algunas restricciones que se deben cumplir.

* La aplicación solo estará disponible en dispositivos Android.
* El equipo estará conformado por 4 integrantes.
* La aplicación debe cumplir con los requerimientos deseados.
* Las bitácoras se deben entregar en un archivo word.
* La entrega debe ser en la fecha estimada.

### 1.1.8. Entregables

### 

* Bitácoras semanales
* Avances del Informe
* Presentaciones (PPT)
* Carta Gantt
* Manual de usuario
* Documentación
* Informe final
* Producto final

# 3. Organización del proyecto

## 3.1. Roles y responsabilidades

En esta tabla se muestra la descripción de los roles según área de trabajo y los encargados

para dichos roles.

**Tabla 1: roles y responsables**

| Rol | Responsable | Descripción |
| --- | --- | --- |
| Líder de proyecto | Rodrigo Suaña | El jefe de proyecto se encargará de planificar, ejecutar y monitorear las acciones que se realizarán durante el proyecto. |
| Informes y Bitácoras | Mauricio Benavides, Felipe Crispin, Anibal Laura, Rodrigo Suaña. | Serán los encargados de documentar los avances y planificaciones del grupo semanalmente (Bitácora), y la revisión de los informes cuando sea necesario. |
| Programadores | Mauricio Benavides, Felipe Crispin, Anibal Laura, Rodrigo Suaña. | Será el encargado de escribir, de depurar y de revisar todo del código fuente de un software informático para que lleve a cabo determinadas tareas que se requieren en un proyecto. |
| Diseñadores de Interfaz | Mauricio Benavides, Felipe Crispin, Anibal Laura, Rodrigo Suaña. | Encargado de maquetar y darle estilo a un diseño en específico de un programa software. |
| Responsable de la Wiki | Anibal Laura - Mauricio Benavides | Será el encargado de hacer el registro de toda la información del proyecto que se realiza en el sitio web “redmine”. |
| Carta Gantt | Rodrigo Suaña, Mauricio Benavides y Felipe Crispin | Se encargará de crear y actualizar la carta gantt para que la carta gantt de “Redmine” refleje el trabajo del equipo. |
| Enviar documentos vía redmine | Anibal Laura | Este será el encargado de enviar todos los documentos entregables al sitio web “Redmine”. |

## 3.2. Mecanismos de comunicación

1. **Whatsapp:** El equipo de trabajo contará con un grupo de Whatsapp el cual se utilizará principalmente para determinar los horarios en los que se realizarán las reuniones de trabajo, así como también para comunicar cualquier inconveniente para asistir a las mismas que pueda presentar uno o más miembros del equipo.
2. **Discord:** Se cuenta con un servidor de Discord en el que se realizan las reuniones de trabajo del equipo, siendo también otro medio de comunicación para la planificación de las siguientes reuniones y actividades a realizar. También es el medio donde se comparten enlaces y archivos relevantes para el proyecto.
3. **Google Drive:** Se tendrá una carpeta en Google Drive en donde cada uno de los integrantes tendrá acceso y esta carpeta servirá como alojamiento de los archivos importantes relacionados con el proyecto como: bitácoras, diferentes etapas del informe, presentaciones, etc.
4. **Redmine:** Herramienta web de gestión de proyectos en la cuál se subirán todos los entregables referentes al proyectos. El equipo de trabajo podrá a su vez editar la Carta Gantt que proporciona la aplicación, planificar las actividades y administrar la wiki del proyecto, entre otros.

# 4. Planificación de los procesos de gestión

## 4.1. Planificación inicial del proyecto

### 4.1.1. Planificación de estimaciones

Esta tabla muestra los recursos Hardware-Software que se utilizan para el desarrollo del proyecto, luego se realizará una estimación de costo por cada recurso y una estimación total.

**Tabla 2: costo por recursos Hardware-Software**

| Recursos | Costo Individual | Cantidad | Costo Total |
| --- | --- | --- | --- |
| Notebook | 1.000.000 | 4 | 4.000.000 CLP |
| Teléfono móvil | 200.000 | 4 | 800.000 CLP |
| Android Studio | Gratis | 4 | 0 CLP |
| GitHub | 19.000 CLP | 4 | 76.000 CLP |
| Total | | | 4.876.000 CLP |

### 4.1.2. Planificación de recursos humanos

En este punto se realizará la estimación total de los recursos humanos, para el cálculo se realizarán las siguientes tablas que describen todos los datos a considerar para el cálculo final del costo del proyecto.

**Tabla 3: costos según área de trabajo(rol).**

| Rol | Costo Individual por Hora | N° de Integrantes |
| --- | --- | --- |
| Programador | 7.000 CLP | 4 |
| Diseñador | 5.000 CLP | 2 |
| Líder | 6.000 CLP | 1 |
| Administrador | 6.000 CLP | 1 |

**Tabla 4: horas de trabajo semanal**

| Rol | Horas por semana |
| --- | --- |
| Programador | 4 |
| Diseñador | 2 |
| Líder | 2 |
| Administrador | 2 |

**Tabla 5: estimación de costo semanal**

| Responsable | Rol (es) | Horas totales por semana | Costo Total por semana |
| --- | --- | --- | --- |
| Anibal Laura | programador, Administrador | 5 | 40.000 CLP |
| Rodrigo Suaña | programador, Líder | 6 | 40.000 CLP |
| Mauricio Benavides | programador, diseñador | 6 | 38.000 CLP |
| Felipe Crispin | programador, diseñador | 6 | 38.000 CLP |
| Total por equipo semana | | | 156.000 CLP |

A continuación se realizará la estimación mensual con respecto a la tabla 3 y una estimación total con una duración de 4 meses.

**Tabla 6: estimación total para los recursos humanos**

| Total por mes | 4\*156.000 = 624.000 CLP |
| --- | --- |
| Total | 4\*624.000 = 2.496.000 CLP |

Finalmente se realizará una tabla con el precio total del proyecto, incluyendo los recursos hardware-software más la estimación final de los recursos humanos.

**Tabla 7: costo del proyecto**

| Recursos | Costos |
| --- | --- |
| Hardware y software | 4.876.000 CLP |
| Recursos Humanos | 2.496.000 CLP |
| Total | 7.372.000 CLP |

## 

## 4.2. Lista de actividades

### 4.2.1. Actividades de trabajo y asignación de tiempo

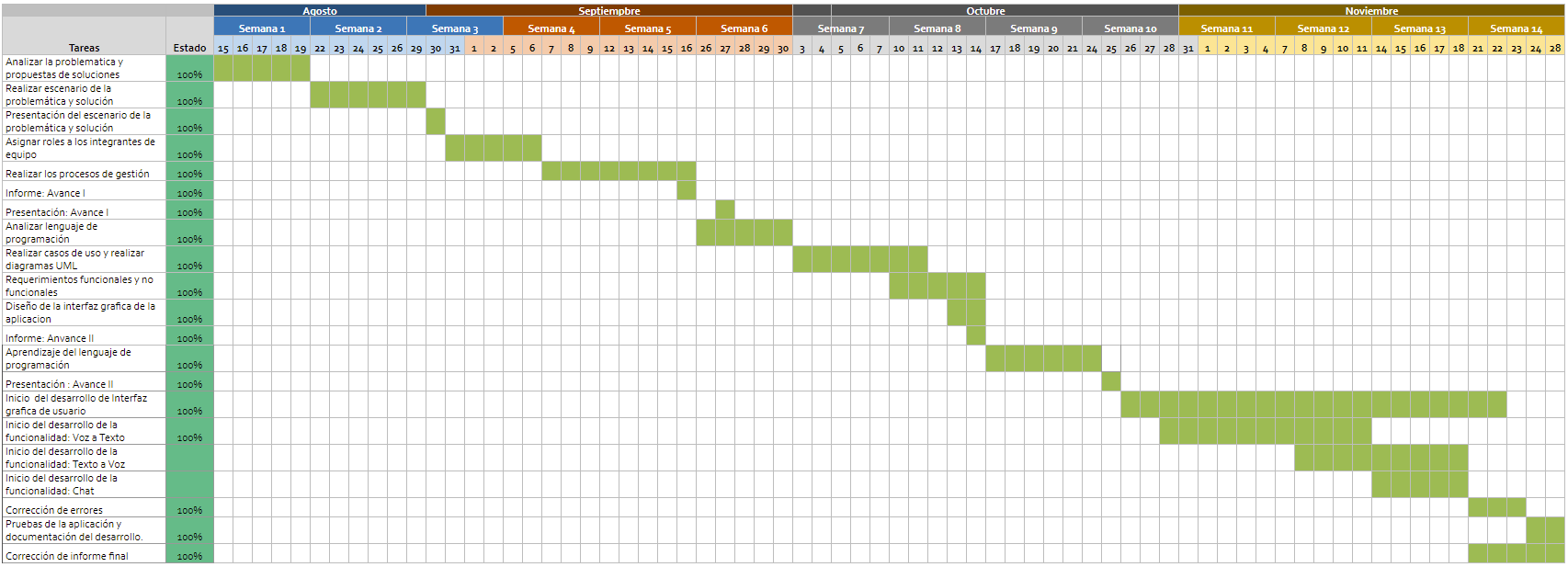
En este punto se tiene la descripción de las actividades a realizar y los tiempos asignados para dicha actividad y también se incluye a los responsables. A medida que se van realizando las actividades, se estará haciendo un informes para cada fase, presentaciones y entre otros entregables.

**Tabla 8: actividades y responsables**

| Nro | Actividad | Tiempo | Responsables |
| --- | --- | --- | --- |
| 1 | Fase inicial de ideas propuestas y asignación de roles | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 2 | Fase analizar y escoger una idea propuesta | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 3 | Realizar una descripción del problema y posible solución | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 4 | Realizar escenario del problema -solución, realizar primera presentación | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 5 | Analizar lenguajes de programación | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 6 | Aprendizaje del lenguaje | 2 semanas | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 7 | Analizar requerimientos y realizar modelos gráficos en UML | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 8 | Borradores de diseño gráfico | 1 semana | Rodrigo suaña y  Mauricio benavides |
| 9 | Inicio del desarrollo de Interfaz de usuario | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 10 | Inicio del desarrollo de la funcionalidad 1 (voz a texto) | 2 semanas | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 11 | Inicio del desarrollo de la funcionalidad 2 (texto a voz) | 1 semanas | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 12 | Inicio del desarrollo de la funcionalidad 2 (texto a texto) | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 13 | Pruebas de la aplicación y documentación del desarrollo. | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |
| 14 | Corrección de errores | 1 semana | Anibal Laura y Rodrigo Suaña. |
| 15 | Corrección de informe final y presentación. | 1 semana | Anibal Laura, Rodrigo Suaña, Mauricio Benavides y Felipe Crispin. |

### 4.2.3. Carta Gantt

La imagen muestra la asignación de tiempo en una carta gantt hecha en excel.



***Figura 3: Carta gantt***

## 4.3. Planificación de la gestión de riesgos

En el transcurso del desarrollo del proyecto existen ciertos riesgos que pueden interrumpir su desarrollo, a continuación, se describe la categoría de los riesgos y los posibles riesgos que se pueden presentar.

**Tabla 9: categorías de riesgos**

| Categoría de Riesgos | |
| --- | --- |
| 1 | Catastrófico |
| 2 | Crítico |
| 3 | Marginal |
| 4 | Despreciable |

**Tabla 10: Riesgos**

| Riesgo | Probabilidad de ocurrencia | Nivel de impacto | Acción remedial |
| --- | --- | --- | --- |
| Ausencia de algún integrante del equipo | 30 % | 2 | Se organizara las tareas entre los integrantes presentes y luego se hablará con el integrante ausente para tomar medidas |
| Problemas técnicos con los equipos | 20% | 3 | Tener equipo de respaldo y en caso de no tener apoyarse con un integrante el cual cuente con un equipo. |
| Problemas con conexion a internet | 30% | 2 | Tener una conexión de respaldo, como datos móviles. |
| Pérdida del codigo completo | 20% | 1 | Todos los integrantes tendrán un respaldo de las versiones del código. |
| Error en la ejecución del código | 70 % | 3 | Revisión completa del código o cierta parte en donde puede estar el posible error. |

# 5[.](#_heading=h.1hmsyys) Planificación de los procesos

## 5.1 Análisis

En esta sección se realizará el análisis profundo del proyecto, identificando requerimientos para el desarrollo. Para tener un mejor análisis se realizará diagramas de casos de uso y diagramas de secuencia.

El propósito de este análisis, es tener claro la estructura del proyecto y los pasos a seguir para su desarrollo.

## 5.2 Especificaciones

En esta sección se realizará las especificaciones correspondientes a las funcionalidades

principales del proyecto.

### 5.2.1 Funcionalidad 1: De voz a texto

### El sistema capta la voz mediante el micrófono del smartphone.

1. El sistema realiza el reconocimiento de voz.
2. El sistema traduce la voz a texto.
3. El sistema muestra el texto en pantalla de la voz traducida.

### 5.2.2 Funcionalidad 2: texto a voz

### El sistema capta el texto escrito en pantalla.

### El sistema realiza el reconocimiento de texto.

1. El sistema traduce el texto a voz.
2. El sistema reproduce el audio por medio de los parlantes.

### 5.2.1 Funcionalidad 3: Chat

### El sistema capta las palabras escritas del teclado.

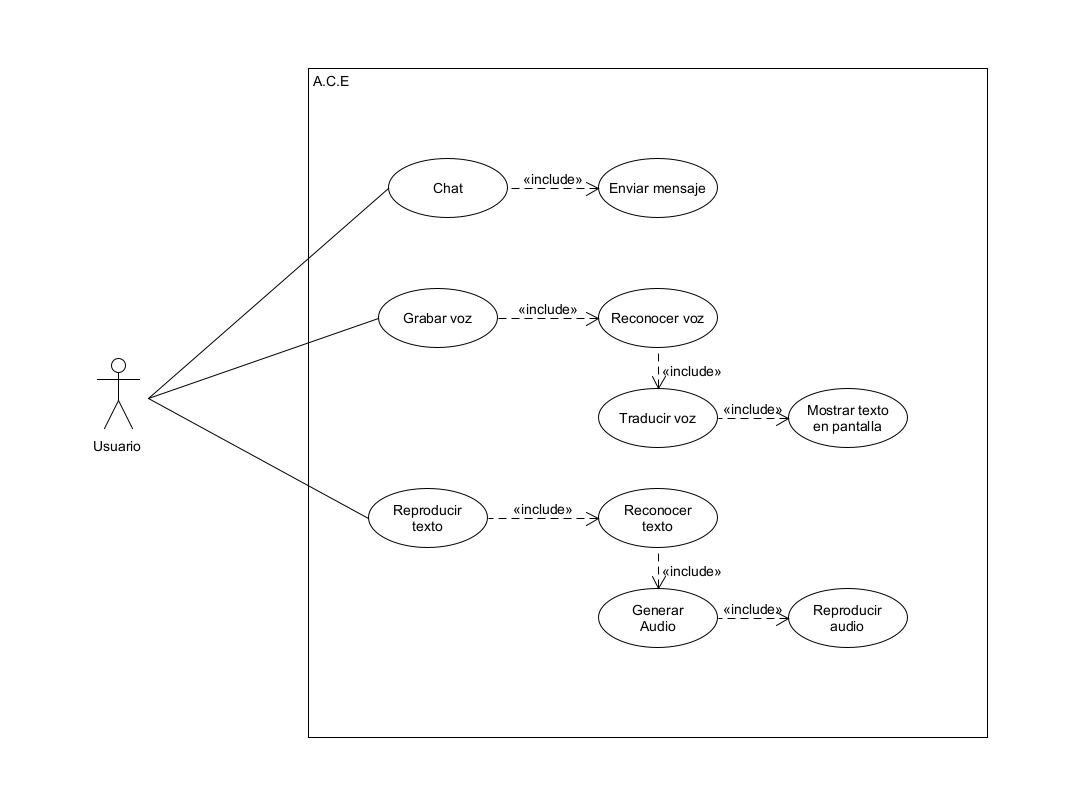
1. El sistema muestra el texto escrito por el usuario.

## 5.3 Restricciones

* El sistema cuenta con una distancia próxima para la detección adecuada de voz, 2 metros aproximado.
* El sistema requiere el uso permiso de los parlantes del smartphone.

## 5.4 Casos de uso

En esta sección se mostrará el diagrama de casos de uso y la descripción de los casos de uso.



***Figura 4: Diagrama de casos de uso***

### 5.4.1 Descripción de los casos de uso

**Tabla 11: descripción caso de uso voz a texto**

| Nombre caso de uso: Voz a texto | |
| --- | --- |
| Autor: Mauricio Benavides, Rodrigo Suaña - 06/12/2022 | |
| Descripción: El sistema permite grabar la voz y convertirlo a texto. | |
| Actor: Usuario objetivo, Usuario con discapacidad auditiva | |
| Actor:   1. El usuario discapacitado presiona el botón “grabar” al momento de que el usuario objetivo quiera entregarle un mensaje, así le dará paso a la grabación de su smartphone. 2. El usuario objetivo deberá transmitir su mensaje verbalmente. 3. El usuario discapacitado podrá leer en la pantalla de su smartphone el mensaje transmitido por el usuario objetivo. | Sistema:   1. El sistema recibirá el mensaje grabado, lo reconocerá y procederá a traducirlo a texto. 2. El sistema devolverá el mensaje en forma de texto y lo mostrará en la pantalla del smartphone. |
|
| Flujo Alternativo: | 1. Si el sistema no reconoce la voz de la grabación, mostrará un mensaje informando que “No entendí. Dilo de nuevo”. |
| Postcondición: Mensaje transmitido correctamente. | |

### 

**Tabla 12: descripción del caso de uso texto a voz**

| Nombre caso de uso: Texto a voz | |
| --- | --- |
| Autor: Anibal laura, Rodrigo Suaña - 06/12/2022 | |
| Descripción: Este caso de uso permite que el sistema transforme el texto a voz. | |
| Actor: Usuario con discapacidad sordomuda. | |
| Actor:   1. El usuario escribe en la caja de texto que se encuentra en pantalla. 2. El usuario presiona el botón reproducir texto, el cual tendrá una forma de parlante. | Sistema:   1. El sistema captura el texto ingresado por el usuario. 2. El sistema genera el archivo de audio y lo reproduce mediante los parlantes del dispositivo. |
|
| Postcondición: Mensaje transmitido correctamente. | |

### 

**Tabla 13: descripción del caso de uso chat**

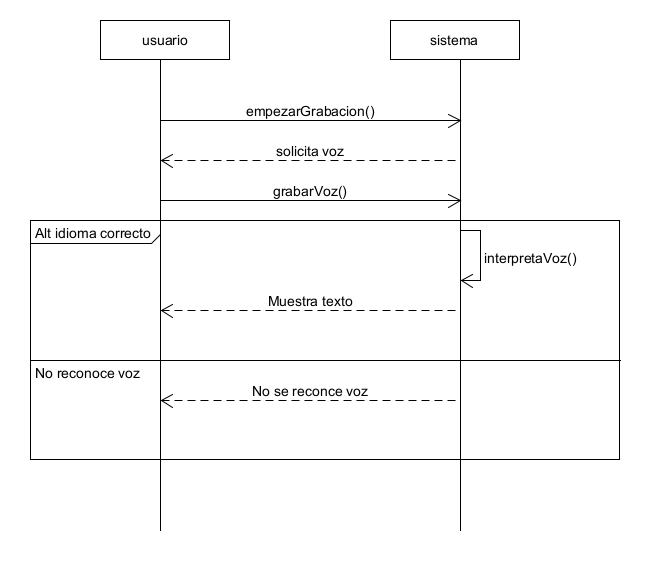
| Nombre caso de uso: Chat | |
| --- | --- |
| Autor: Anibal Laura - 25/11/2022 | |
| Descripción: Este caso de uso permite al usuario escribir texto y mostrarlo en pantalla | |
| Actor: Usuario. | |
| Actor:   1. El usuario ingresa texto en una caja de entrada. 2. El usuario presiona el botón de enviar, el botón está representado por una imagen de un teclado. | Sistema:   1. El sistema captura el texto ingresado. 2. El sistema muestra el texto en una caja de salida, el texto se mantendrá en pantalla mientras dure la interacción. |
|
| Postcondición: Mensaje transmitido correctamente. | |

### 5.4.2 Diagrama de secuencia

A Continuación se mostrarán los diagramas de secuencia, los cuales están conformados por los siguientes niveles:

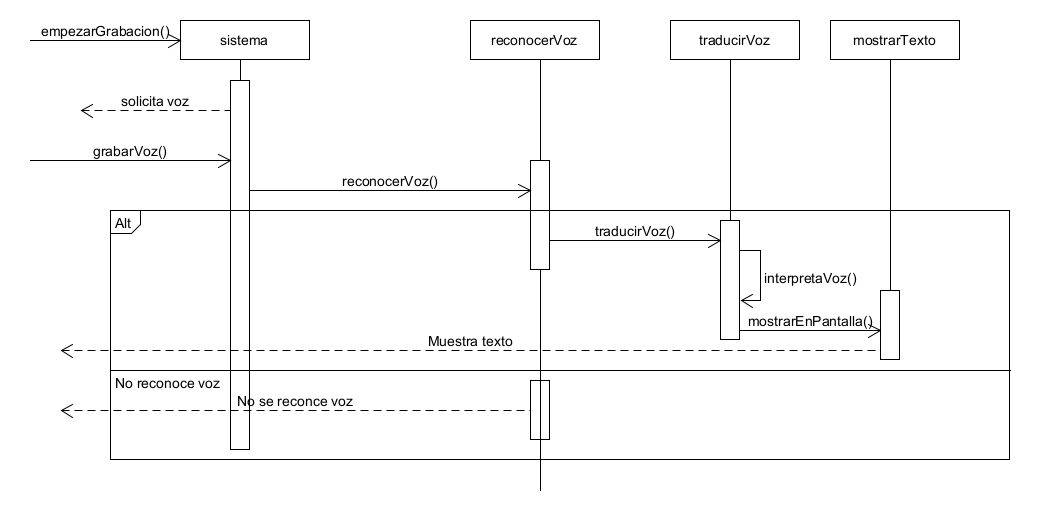
* Nivel 0, se muestra la interacción del usuario con el sistema, haciendo llamadas a funciones del sistema.
* Nivel 1, muestra la interacción del sistema con las clases o funciones que se requieren para cada funcionamiento del sistema, ejemplo grabar voz requiere otras funciones para su ejecución.

#### 5.4.2.1 Funcionalidad 1: Diagrama de secuencia nvl 0.



***Figura 5: Diagrama de secuencia nivel 0***

#### 5.4.2.2 Funcionalidad 1: Diagrama de secuencia nvl 1.



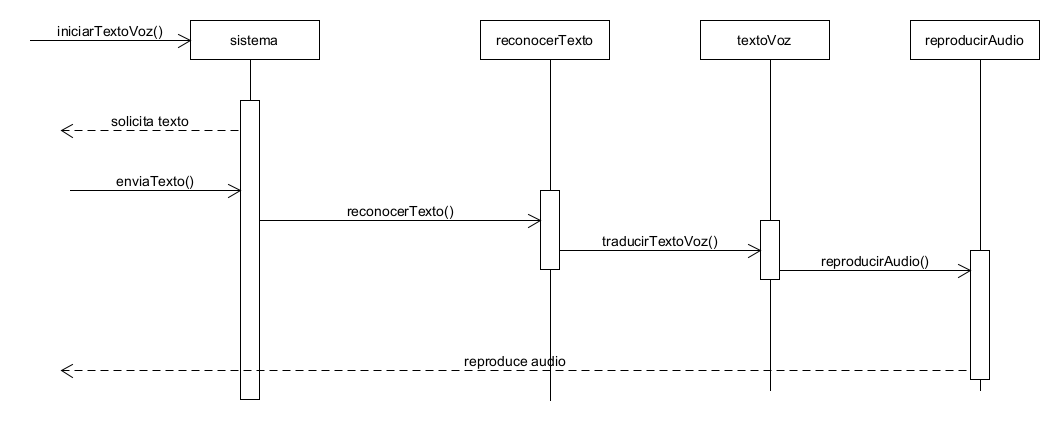
***Figura 6: Diagrama de secuencia nivel 1***

#### 5.4.2.3 Funcionalidad 2: Diagrama de secuencia nvl 0.

## 

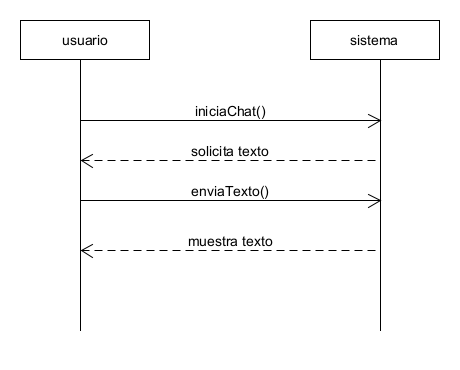
***Figura 7: Diagrama de secuencia nivel 0***

#### 5.4.2.4 Funcionalidad 2: Diagrama de secuencia nvl 1.



***Figura 8: Diagrama de secuencia nivel 1***

#### 5.4.2.5 Funcionalidad 3: Diagrama de secuencia nvl 0.



***Figura 9: Diagrama de secuencia nivel 0***

#### 5.4.2.6 Funcionalidad 3: Diagrama de secuencia nvl 1.

## 

***Figura 10: Diagrama de secuencia nivel 1***

## 

## 

## 

## 

## 

## 

## 5.5 Descripción de la arquitectura de sistema

### 5.5.1 Arquitectura del sistema:

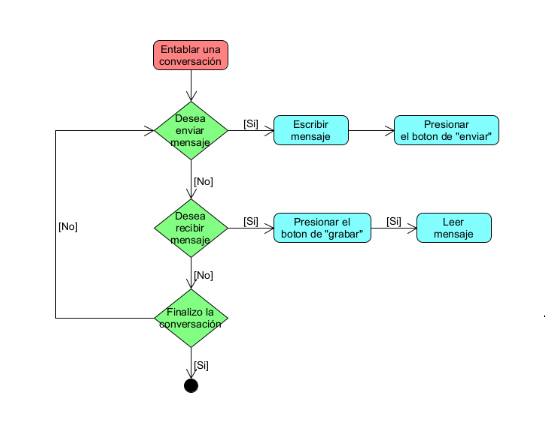


***Figura 11: Arquitectura de Sistema***

1. Smartphone con sistema operativo android, la interfaz está compuesta por botones, inputs y campos donde se mostrará texto, ya sea traducido (Voz a Texto) o mensajes (Chat).
2. Parlantes del dispositivo funcionando, para la reproducción del texto transformado a voz.
3. Permisos del sensor del audio del smartphone, capta la voz del hablante para luego traducirlo a texto (para el discapacitado).

### 5.5.2 Diagrama de flujo:

Este diagrama muestra el inicio, y las distintas “rutas” que puede tomar la ejecución de la aplicación.



***Figura 12: Diagrama de flujo.***

## 

## 5.6 Requerimientos funcionales y no funcionales.

### 5.6.1 Requerimientos funcionales:

* Voz a texto: El sistema permitirá recibir audio en su dispositivo móvil para que este luego sea reconocido, traducido y mostrado en pantalla.
* Texto a voz: El sistema permitirá traducir el texto ingresado con anterioridad a audio, para ser reproducido por medio de los parlantes.
* Chat: El sistema permitirá a dos usuarios establecer una conversación local mediante el intercambio de mensajes(texto).
* Vistas: El sistema permitirá al usuario moverse entre las diferentes vistas que representan las distintas funcionalidades de la aplicación (chat, voz a texto, texto a voz).

### 5.6.2 Requerimientos no funcionales:

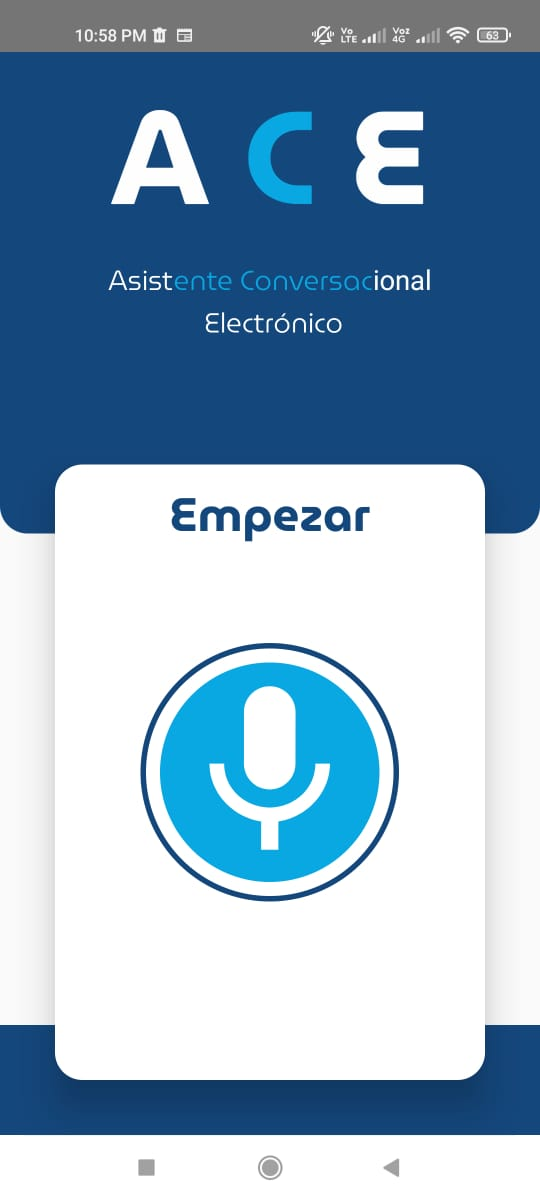
* Interfaz: El sistema debe presentar una interfaz amigable y de fácil comprensión.
* Portabilidad: La aplicación solo se encuentra disponible para dispositivos que cuenten con el sistema operativo Android.
* Idioma: La aplicación solo estará disponible para el idioma español.

## 5.7 Interfaz gráfica usuario (IGU)

**Interfaz 1.**

Esta pantalla es la de inicio de la aplicación. En la parte superior de la derecha, hay un

botón de ayuda para el usuario (más información en el manual).



***Figura 13: interfaz inicio.***

**Interfaz 2.**

Pantalla de voz a texto: En esta interfaz se muestra la funcionalidad de voz a texto.



***Figura 14: interfaz voz a texto.***

**Interfaz 3.**

Pantalla de texto a voz: En esta interfaz se muestra la funcionalidad de texto a voz.



***Figura 15: interfaz texto a voz.***

**Interfaz 4.**

Pantalla de texto a voz: En esta interfaz se muestra la funcionalidad de texto a voz.



***Figura 16: chat local.***

## 5.8 Herramientas y técnicas

**Herramientas**: Android Studio (lenguaje de programación Java), Google Drive, Google Docs, Microsoft Office, Github.

# 6. Planificación de procesos de soporte

## 6.1. Planificación de la documentación

* **Manual de usuario**: Es un documento que contiene las diferentes instrucciones de instalación y la descripción de todas las funcionalidades que posee la aplicación.
* **Bitacoras:** Un conjunto de documentos que describen las principales tareas realizadas en una reunión de trabajo de equipo, así como las tareas planeadas para la siguiente reunión, sugerencias a desarrollar y los temas posibles a tratar.
* **Carta Gantt:** Es una línea de tiempo de la planificación y desarrollo de futuras tareas relacionadas con el avance del proyecto, cada una de las cuales sean futuras, pasadas o en progreso.
* **Wiki del proyecto:** Es un blog presente en la página de Redmine que detalla de forma general la realización del proyecto, sus aspectos que lo componen y su propósito.
* **Informe final de proyecto:** Es el presente documento que describe todo el apartado de planificación del proyecto, así como el análisis, diseño e implementación de la aplicación software desarrollada

# 7. Implementación

#### **7.1. Plan de integración**

La aplicación móvil presentada tiene como propósito servir de asistente a personas que tengan alguna discapacidad ya sea sordomudo o auditiva que le impida comunicarse normalmente.Esto se pretende lograr mediante el uso tanto de una funcionalidad de voz a texto, la cual recibirá el audio de una persona hablando y mediante la utilización de una API el audio será transformado en un texto que será mostrará en la pantalla del dispositivo, la aplicación también contará con la funcionalidad de texto a voz, la cual mediante el uso de APIs se utilizara una voz generada por software para “hablar” lo se haya escrito en la pantalla. Finalmente una última funcionalidad es el chat local, esta funcionalidad está pensada para una conversación entre personas sordomudas.

#### **7.2. Modelos de implementación**

##### **7.2.1. Traducción de voz a texto.**

Para este paso se utilizara una API de google, de voz a texto, la cual recibe audio, lo procesa y luego lo transforma en un string de caracteres el cual luego será mostrado en la pantalla.

##### **7.2.2. Transformación de texto a voz.**

Para este paso también se utilizara una API, la cual tomará el texto ingresado y transforma, mediante un generador de voces, un audio que luego será reproducido por los parlantes del dispositivo.

**7.2.3 Chat local**

Esta funcionalidad captura el texto escrito en pantalla por el usuario, luego lo muestra en una caja de texto haciendo referencia a que usuario le pertenece el texto. Hay dos botones para enviar, uno hace referencia al usuario de la derecha y el otro a la izquierda.

# 8.Problemas y Soluciones

Durante el desarrollo del proyecto a lo largo del semestre nos hemos ido topando con distintos problemas, algunos de ellos fueron:

## 8.1 Problema N°1: Kivy MD

Para poder hacer implementación de la funcionalidad de voz a texto una de las primeras idea de lenguaje y librerías para poder realizarlo fue Kivy MD, al momento de hacer uso de esta se hizo notorio la dificultad que tiene para hacer el paso a APK además del tamaño de este el cual es muy grande.

### Solución

La solución con la cual fue posible salir de este problema fue el cambio a otro lenguaje de programación el cual a diferencia de Kivy MD el paso a APK fuera menos complicado y el APK resultante fuera de un menor tamaño. El software escogido para solucionar este problema fue Android Studio, en el cual se hace uso del lenguaje de programación Java.

## 8.2 Problema N°2: React Native

En este caso el problema encontrado fue que no se pudo lograr la implementación de la funcionalidad de voz a texto.

### Solución

Fue necesario hacer el cambio a un software en el cual fuera posible la implementación.De la misma forma que en el problema anterior el software que soluciono este problema fue Android Studio.

# 9.Implementación

## 9**.1 Módulos**

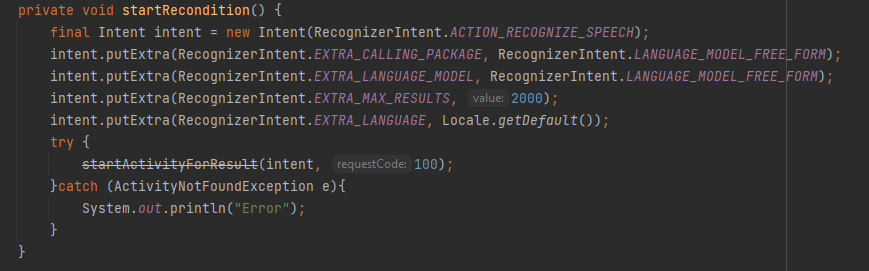
### 9.1.1 Módulo speechToText

* ACTION\_RECOGNIZE\_SPEECH: Inicia una actividad que pedirá al usuario que hable y la envíe a través de un reconocedor de voz.
* LANGUAGE\_MODEL\_FREE\_FORM: Utilice un modelo de lenguaje basado en el Reconocimiento de voz de forma libre.
* EXTRA\_LAGUAJE\_MODEL: Informa al reconocedor qué modelo de voz preferir cuando se realiza
* ACTION\_RECOGNIZE\_SPEECH. El reconocedor utiliza esta información para ajustar los resultados. Este extra es obligatorio. Las actividades que implementan.
* ACTION\_RECOGNIZE\_SPEECH pueden interpretar los valores como mejor les parezca
* EXTRA\_MAX\_RESULTS: Límite opcional en el número máximo de resultados a devolver.
* EXTRA\_LANGUAJE: Etiqueta de idioma IETF opcional, por

ejemplo, "es-CL".

**Método startRecondition**

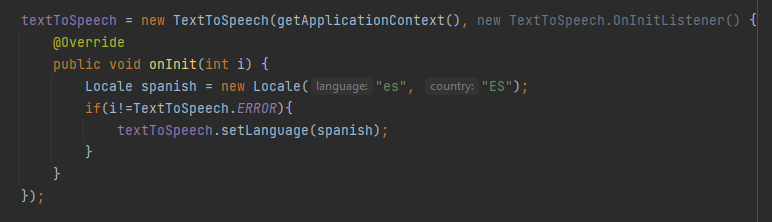
Esta función inicializa el reconocimiento de voz que solicita hablar al usuario usuario.

******

***Figura 17: método reconocimiento de voz***

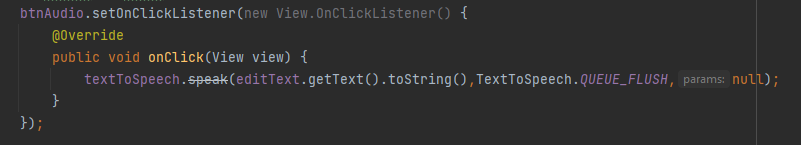
### 9.1.2 Módulo textToSpeech

* TextToSpeech: Es el constructor de la clase TextToSpeech, el cual inicializa el motor TTS (TextToSpeech).
* onInit: Con este método de la Interfaz OnInitListener que es invocado una vez que ha sido inicializado el motor TTS, nos permite saber si la inicialización fue exitosa y definimos el lenguaje con el que se va a reproducir el texto.

******

***Figura 18: Método para inicializar el motor TTS y define el idioma.***

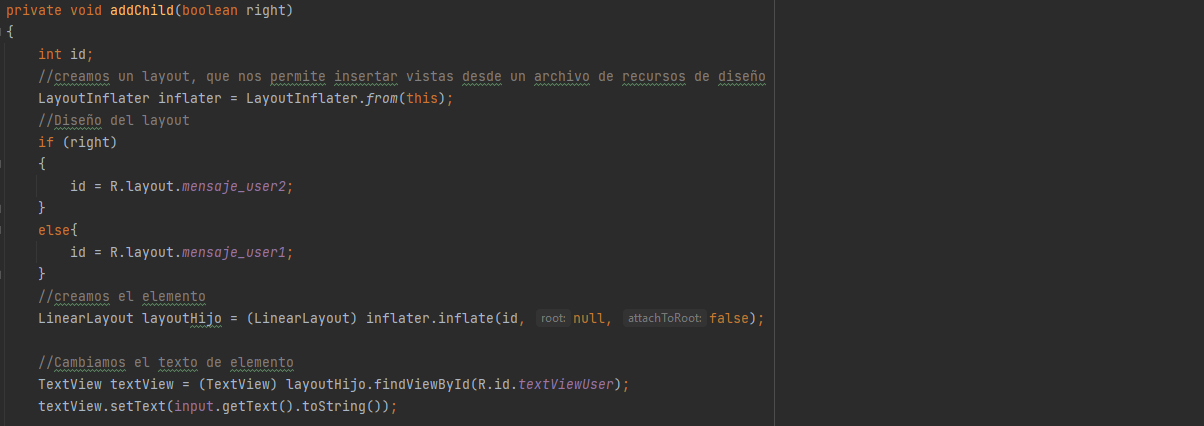
* speak: Método que usa el motor para leer cualquier texto en este caso el texto que se encuentra en el editText que es donde el usuario ingresa el texto.
* TextToSpeech.QUEUE\_FLUS: Parámetro que define la cola de reproducción en este caso al recibir una nueva entrada esta elimina la entrada anterior en la cola y la reemplaza por la nueva.

******

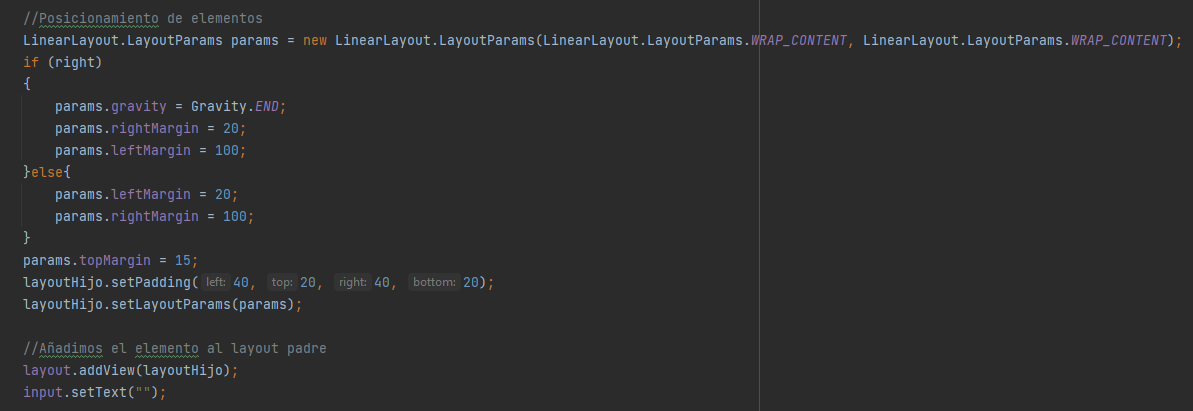
***Figura 19: Método para reproducir audio.***

### 9.1.3 Módulo ChatText

* addChild: Método con el cual añadimos y posicionamos un elemento (mensaje) en un contenedor padre. Como atributo recibe un valor booleano, en donde si es true indica que debe ser posicionado al lado derecho de la pantalla y si es false los posiciona al lado izquierdo.



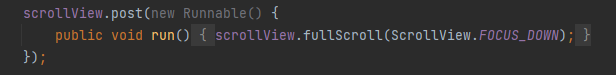
***Figura 20***



***Figura 21***

***Figura 20 y 21: Método para añadir y posicionar elementos dentro de un contenedor.***

* run: Método el cual mueve el scroll del contenedor de mensajes hacia la parte inferior de éste, de tal manera que se muestran los últimos mensajes insertados.



***Figura 22: Método trasladar el scroll hacia la parte inferior del contenedor.***

# 10. Pruebas

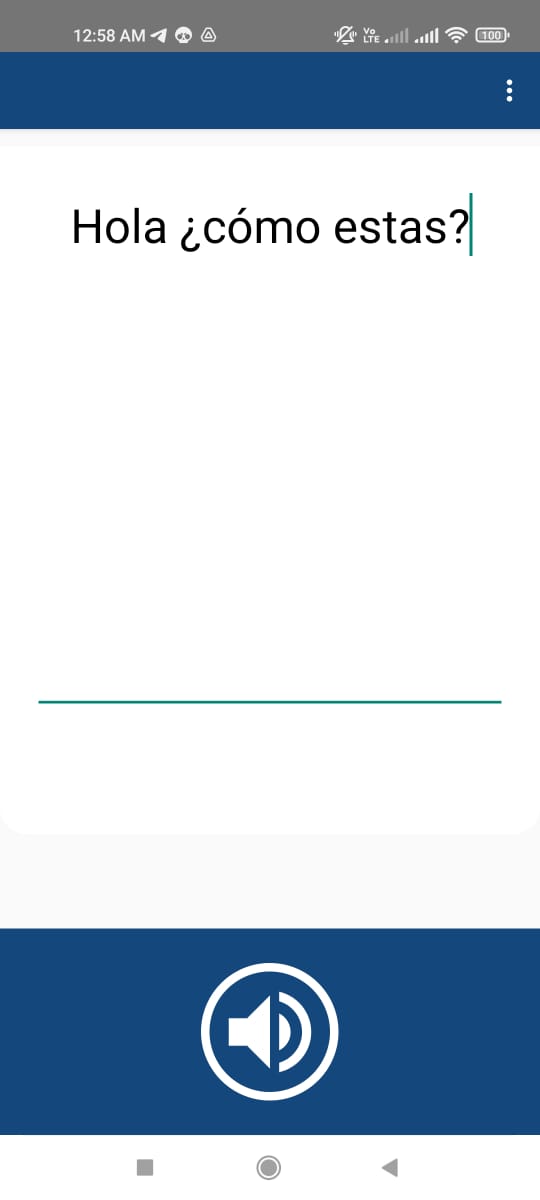
**10.1 Primera prueba: voz a texto**

****

***Figura 23: Imagen de prueba, Voz a Texto.***

En esta prueba se experimentó con el smartphone, simulando el caso de que una persona con discapacidad auditiva desee entablar una conversación con una persona normal. Los resultados fueron exitosos, la conversación se entablo satisfactoriamente.

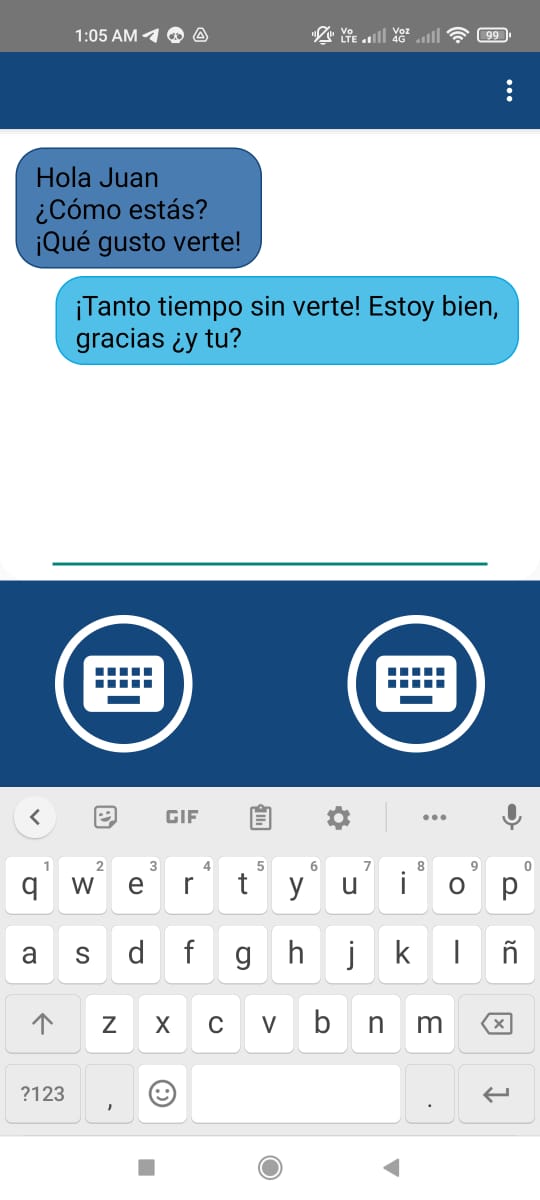
**10.2 Segunda prueba: texto a voz**

****

***Figura 24: Imagen de prueba, Texto a Voz.***

En esta prueba se hizo la simulación interpretando a una persona sordomudo, es decir la persona con esa discapacidad no tiene la capacidad de hablar y gracias a nuestra app el puede escribir texto en la pantalla y el receptor no tiene problemas en escuchar el audio. El resultado fue bueno.

**10.3 Tercera prueba: chat local.**

****

***Figura 25: Imagen de prueba, chat local.***

Para esta prueba se hizo la simulación de una conversación entre dos personas con discapacidad auditiva, la aplicación cumplio con el resultado esperado, no surgio ningun problema.

# 12. [Conclusión](#_heading=h.1hmsyys)

Luego de abordar las tareas planificadas, con el fin de continuar la progresión del proyecto, se estima que en cuanto a especificaciones del mismo, así como objetivos, el alcance a donde se pretende llegar, la planificación, entre muchos apartados más, son fundamentales que se encuentren, desde un inicio, bien especificados si se quiere progresar sin mayores problemas y así alcanzar el éxito. Se tienen etapas para dedicar tiempo a la revisión sobre la correcta definición de cada uno de estos puntos, por lo que se espera seguir progresando de una buena manera en el desarrollo del objetivo.

Durante el desarrollo del proyecto, se encontraron problemas los cuales no generaron un mayor retraso ni inconveniente, la solución para estos problemas fue generada gracias al trabajo en equipo y a las distintas ideas de cada uno de los integrantes. Con una mirada hacia el futuro se espera poder poner en práctica lo aprendido en el desarrollo de este proyecto, además del crecimiento que esta experiencia significó para el grupo completo.

El producto resultante de este proyecto cumplio con las expectativas que el grupo tuvo desde el comienzo del mismo, fue posible implementar todas las funcionalidades propuestas, además de agregar mejoras que fueron apareciendo en el camino, si bien el producto se encuentra en un estado avanzado este aun no se encuentra terminado en su totalidad ya que siempre hay espacio para una posible mejora.

# Referencias

1. Algunas de las ideas se extrajeron de los documentos que publicó el profesor D. Aracena Pizarro en Google Drive.

<https://drive.google.com/drive/u/2/my-drive>

1. IPS junto al Ministerio de Desarrollo Social y Familia dan a conocer ley que favorece a personas sordas enlace: <https://www.ips.gob.cl/servlet/internet/noticia/1421811586067/ips-mindes-dan-a-conocer-ley-que-favorece-a-personas-sordas#:~:text=En%20Chile%20existen%20712.005%20personas,las%20personas%20con%20discapacidad%20auditiva>
2. Android studio:

<https://developer.android.com/studio?hl=es-419&gclid=Cj0KCQiA7bucBhCeARIsAIOwr-_CrnezguQYrQEmlFAdLXv1CkoFc11swdOZlIPafs7XYATrLYJrQrwaAoYzEALw_wcB&gclsrc=aw.ds>

1. Canva, sitio web para diseño de los esquemas problema solución:

<https://www.canva.com/>

1. Figma, sitio web para el diseño de la interfaz:

<https://www.canva.com/>

1. RecognizerIntent: API de google para el módulo voz a texto

<https://developer.android.com/reference/android/speech/RecognizerIntent>

1. Text to Speech

<https://developer.android.com/reference/android/speech/tts/TextToSpeech>