

UNIVERSIDAD DE TARAPACÁ



FACULTAD DE INGENIERÍA



Departamento de Ingeniería en Computación e Informática

Plan de proyecto

Cámara lectora para gente invidente

Autor(es): Christian Cáceres

Ernesto García

Daniel Ramírez

Asignatura: Proyecto 2

Profesor(es): Diego Alberto Aracena Pizarro

ARICA, 22 de Diciembre 2020

Historial de Cambios

Dueño del documento: Grupo "Ingenieros Anónimos"

Fecha	Versión	Descripción	Autor(es)
06/10/2020	1.0	Versión preliminar del formato	Christian Cáceres Ernesto García Daniel Ramírez
20/10/2020	1.1	Modificación del informe plan de proyecto	Christian Cáceres Ernesto García Daniel Ramírez
27/10/2020	1.2	Revisión del informe plan de proyecto	Christian Cáceres Ernesto García Daniel Ramírez
29/10/2020	1.3	Modificación del informe plan de proyecto	Christian Cáceres Ernesto García Daniel Ramírez
02/11/2020	1.4	Versión final del informe plan de proyecto	Christian Cáceres Ernesto García Daniel Ramírez
18/12/2020	1.5	Modificación del informe de avance	Christian Cáceres Ernesto García Daniel Ramírez
19/12/2020	1.6	Modificación del informe de avance	Christian Cáceres Ernesto García Daniel Ramírez
22/12/2020	1.7	Modificación del informe de avance	Christian Cáceres Ernesto García

Tabla de Contenidos

Historial de Cambios	2
1. Panorama General.....	5
1.1. Resumen del proyecto:	5
1.1.1. Introducción:	5
1.1.2. Escenario:.....	5
1.1.3. Propósito:	5
1.1.4. Alcance:.....	5
1.1.5. Objetivo General:	6
1.1.6. Objetivos Específicos:.....	6
1.1.7. Suposiciones:	6
1.1.8. Restricciones:	6
1.1.9. Entregables del Proyecto:	6
2. Organización del Proyecto	7
2.1. Personal y entidades internas	7
2.2. Roles y responsabilidades	7
2.3. Mecanismos de comunicación	7
3. Planificación de los procesos de gestión.....	8
3.1. Planificación inicial del proyecto.....	8
3.1.1. Planificación de estimaciones	8
3.1.2. Planificación de Recursos Humanos.....	8
3.2. Lista de actividades	9
3.2.1. Actividades de trabajo.....	9
3.2.2. Carta Gantt	10
3.2.3. Asignación de tiempo.....	10
3.3. Planificación de la gestión de riesgos	11
4. Referencias.....	11
5. Planificación de procesos técnicos	12
5.1. Modelo de Proceso	12
5.1.1. Diagrama de Caso de Uso General	12
5.1.2. Diagrama de Caso de Uso De Sistema.....	12
5.1.3. Diagrama de Secuencia.....	14
5.1.3.1. Inicia Aplicación	14
5.1.3.2. Capturar Video	14
5.1.3.3. Reconocer texto	15

5.1.3.4.	Transformar texto en audio	15
5.1.3.5.	Reproducir Audio.....	15
5.1.4.	Modelo de Clases	16
5.1.5.	Descripción de la Arquitectura	17
5.1.6.	Diseño de la Interfaz de Usuario	18
5.1.7.	Especificación de requerimientos.....	19
5.2.	Herramientas y técnicas.....	19
6.	Planificación de procesos de soporte.....	20
6.1.	Planificación de la documentación	20
7.	Problemas Encontrados.....	20
8.	Soluciones Propuestas	20
9.	Conclusión	21

1. Panorama General

1.1. Resumen del proyecto:

1.1.1. Introducción:

En la actualidad las personas invidentes o con problemas de vista pueden recurrir a la tecnología para facilitar su día a día. Es por eso que el presente informe tiene como fin mostrar la planificación del proyecto “Cámara lectora para gente invidente”, el cual tiene pensado hacer una aplicación móvil para ayudar a las personas con problemas de visión a leer etiquetas en los productos que consumen.

En este informe se mostrará el escenario considerado para el proyecto, su propósito, alcance y objetivos que fueron considerados llevar a cabo el proyecto. Además, se observa la distribución de roles entre el personal del proyecto, junto con la planificación de los procesos de gestión del mismo.

1.1.2. Escenario:



Ilustración 1: Escenario del proyecto

1.1.3. Propósito:

En la actualidad la tecnología de asistencia a videntes está aumentando gracias a todos los avances hechos en las últimas décadas, una persona invidente puede tener una vida y trabajo normales gracias a las nuevas tecnologías. Es por esto que en el presente proyecto se ha determinado desarrollar un software que pueda reconocer el texto frente a la cámara frontal (de un smartphone) y reproducirlo en voz alta, ayudando a los invidentes (o personas con problemas a la vista) a estar más conectados con el mundo.

1.1.4. Alcance:

La aplicación utilizará la cámara frontal del móvil para localizar texto y luego reproducirlo en forma de audio. Será desarrollado en el lenguaje de programación “Python”, versión 3.6.8.

1.1.5. Objetivo General:

Desarrollar una aplicación que reconozca texto y lo reproduzca en voz alta.

1.1.6. Objetivos Específicos:

1. Estudiar y definir el problema.
2. Proponer la solución al problema y sus alternativas.
3. Desarrollar la aplicación de acuerdo a la solución seleccionada.
4. Realizar pruebas a la aplicación.
5. Entregar la aplicación finalizada.

1.1.7. Suposiciones:

1. Se asume que las personas invidentes que utilicen la aplicación tienen un celular con una cámara frontal funcional.
2. Se asume que las personas no tienen problemas auditivos.
3. Se presume que el medio por el cual se distribuirá la aplicación será por Google Play.

1.1.8. Restricciones:

1. El proyecto debe ser realizado en el plazo determinado por el usuario.
2. La programación del software será realizada en el lenguaje de programación Python.
3. La aplicación debe poder ser usada por una persona no vidente.

1.1.9. Entregables del Proyecto:

1. Bitácoras semanales.
2. Informe de Plan de Proyecto
3. Presentación del Escenario.
4. Informe de avance.
5. Informe final.
6. Presentación final.
7. Manual de usuario.
8. Wiki del proyecto.
9. Producto final.

2. Organización del Proyecto

2.1. Personal y entidades internas

Jefe de proyecto, Programadores, Diseñadores, Secretario.

2.2. Roles y responsabilidades

Jefe de proyecto: Se encarga de la coordinación del trabajo del grupo y de la comunicación con el usuario. El responsable es: Daniel Ramírez Díaz.

- Responsable de programar y desarrollar el software en Python que implemente las funciones de leer y reproducir texto para la aplicación.
- Responsable de diseñar la interfaz visual de usuario para la aplicación móvil.
- Responsable de la documentación del proyecto, de esta forma, realizan las bitácoras del proyecto y formulan los informes requeridos.

2.3. Mecanismos de comunicación

El grupo se comunica principalmente por la aplicación Discord, la cual se trata de un chat de voz y texto gratuito con una interfaz muy moderna e intuitiva, en la cual se pueden notificar con mensajes de texto o canales de voz. Alternativamente el grupo se comunicará por WhatsApp. Para realizar los informes y bitácoras se utilizará Microsoft Word a través de OneDrive, que permite a los miembros del grupo trabajar simultáneamente en los documentos.

3. Planificación de los procesos de gestión

3.1. Planificación inicial del proyecto

3.1.1. Planificación de estimaciones

Recurso	Valor	Cantidad
Notebook	\$500.000	3
Smartphone	\$180.000	3
Software de desarrollo Python	De libre acceso	3
Microsoft Office	\$56.000	1
Sueldo del desarrollador	\$938.000	3
Costo Total del proyecto	\$1.674.000	

3.1.2. Planificación de Recursos Humanos

Responsable de programar: Ernesto García.

Responsable de diseñar: Daniel Ramírez.

Responsable de documentar: Christian Cáceres.

3.2. Lista de actividades

3.2.1. Actividades de trabajo

- Crear diseño básico de la aplicación.
Descripción: Se crea el diseño base de la aplicación.
Responsable: Daniel Ramírez.
- Realizar el informe de plan de proyecto.
Descripción: Se realiza el informe de plan de proyecto
Responsable: Christian Cáceres.
- Investigar sobre el software OCR.
Descripción: Se investigan sobre los tipos de software OCR y lo que pueden hacer.
Responsable: Ernesto García.
- Investigar la programación en Python.
Descripción: Se investiga la programación en Python y se hacen cursos.
Responsable: Daniel Ramírez.
- Diseñar la interfaz de usuario de la aplicación.
Descripción: Se diseña una interfaz más avanzada para la aplicación.
Responsable: Daniel Ramírez.
- Implementar el reconocimiento de texto.
Descripción: Se implementa el reconocimiento de texto en Python.
Responsable: Daniel Ramírez.
- Realizar pruebas de reconocimiento de texto.
Descripción: Se realizan pruebas de reconocimiento de texto.
Responsable: Ernesto García.
- Realizar el informe de avance de proyecto.
Descripción: Se realiza el informe de avance de proyecto.
Responsable: Christian Cáceres
- Implementar la lectura de texto.
Descripción: Se implementa la lectura de texto en Python.
Responsable: Ernesto García.
- Realizar pruebas de lectura de texto.
Descripción: Se realizan pruebas de reconocimiento de texto.
Responsable: Ernesto García.
- Programar la interfaz de la aplicación.
Descripción: Se programa la interfaz de usuario con el software Kivy.
Responsable: Daniel Ramírez.
- Realizar pruebas de funcionalidad.
Descripción: Se realizan pruebas de funcionalidad de la aplicación.
Responsable: Ernesto García.

- Realizar documentación de los códigos.
Descripción: Se realiza la documentación.
Responsable: Christian Cáceres.
- Realizar el informe final de proyecto.
Descripción: Se realiza el informe final del proyecto.
Responsable: Christian Cáceres
- Realizar la presentación final del proyecto.
Descripción: Se realiza la presentación final de proyecto.
Responsable: Daniel Ramírez

3.2.2. Carta Gantt

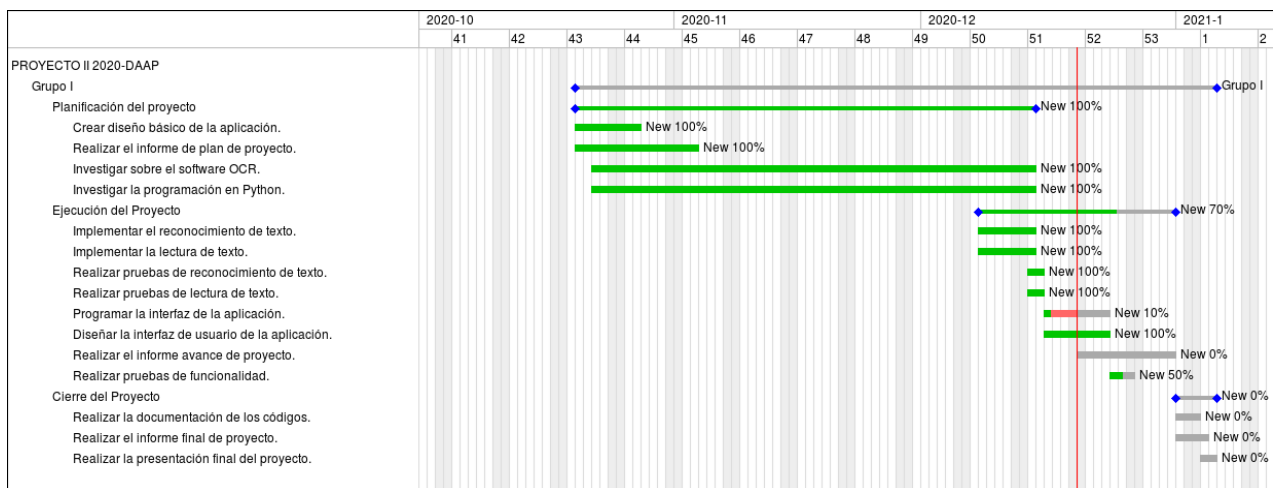


Ilustración 2: Carta Gantt del proyecto

3.2.3. Asignación de tiempo

- Planificación del proyecto: 4 semanas.
- Ejecución del proyecto: 5-6 semanas.
- Cierre de proyecto: 1-2 semanas.

3.3. Planificación de la gestión de riesgos

RIESGOS	PROBABILIDAD DE OCURRENCIA	NIVEL DE IMPACTO	ACCIÓN REMEDIAL
Un integrante del equipo tiene problemas de salud.	10%	1	Repartir trabajo entre los integrantes restantes
La cámara o altavoz del Smartphone se dañan	25%	1	Reponer Smartphone.
La pantalla táctil del Smartphone esté defectuosa	15%	2	Reemplazar pantalla táctil.
La batería del Smartphone deja de funcionar.	30%	3	Cargar el Smartphone
Los programas se pierden debido a un error en el dispositivo de almacenamiento.	10%	1	Respaldar los datos periódicamente

Nivel de impacto:

- 1: Catastrófico
- 2: Crítico
- 3: Marginal
- 4: Despreciable

4. Referencias

Apuntes del Curso "Proyecto 2".

Apuntes del Curso "Taller de Técnicas de Programación".

Materiales del Curso "Formato Bitacora.pdf".

Materiales del Curso "Propuesta formato plan de proyecto 2.docx".

Materiales del Curso "Jit 1 ImageProcessing.pdf".

Materiales del Curso "Jit 2 Filtros y Rasgos.pdf".

Materiales del Curso "Entregables.pdf".

Materiales del Curso "Proyecto II Piloto 2do Semestre 2020v1.0.pdf".

5. Planificación de procesos técnicos

5.1. Modelo de Proceso

5.1.1. Diagrama de Caso de Uso General

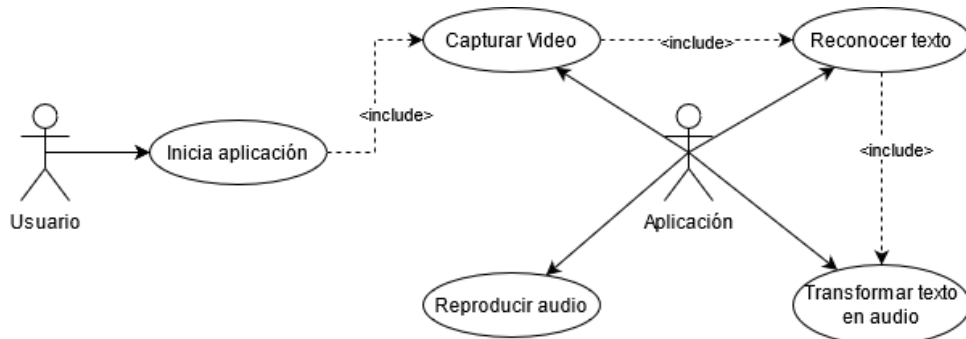


Ilustración 3: Caso de Uso del proyecto

5.1.2. Diagrama de Caso de Uso De Sistema

Nombre:	Iniciar aplicación
Autor:	Ernesto García
Fecha:	22/12/2020
Descripción:	Permite al usuario iniciar la aplicación.
Actor:	Usuario.
Flujo Normal:	
Usuario	Sistema
1.-El usuario presiona el icono de la aplicación	2.-El sistema inicia la aplicación

Nombre:	Capturar video
Autor:	Christian Cáceres
Fecha:	22/12/2020
Descripción:	Permite al usuario capturar un video que contenga texto que se desea leer.
Actor:	Usuario.
Precondiciones:	La aplicación debe tener acceso a la cámara
Flujo Normal:	La aplicación detecta texto
Usuario	Aplicación
4.-El usuario presiona la pantalla	1.-La aplicación captura video 2.-Incluye el C.U.S Reconocer texto 3.-La aplicación manda un mensaje de voz "Se ha encontrado texto" 5.-Incluye C.U.S Reproducir Audio
Flujo Alternativo:	La aplicación no detecta texto

Usuario	Sistema
	2.1.-La aplicación manda un mensaje de voz "No se encuentra texto"
Postcondiciones: Se genera una imagen	

Nombre:	Reconocer texto
Autor:	Ernesto García
Fecha:	22/12/2020
Descripción: Comprueba si se encuentra texto en la imagen	
Actor: Aplicación.	
Precondiciones: La aplicación debe tener acceso a los archivos del smartphone. Debe haber un haber una imagen capturada.	
Flujo Normal: La aplicación detecta texto	
Aplicación	Sistema
1.-La aplicación busca la imagen capturada en el C.U.S Capturar Video 2.-La aplicación es procesada	
Postcondiciones: Se genera un archivo de texto	

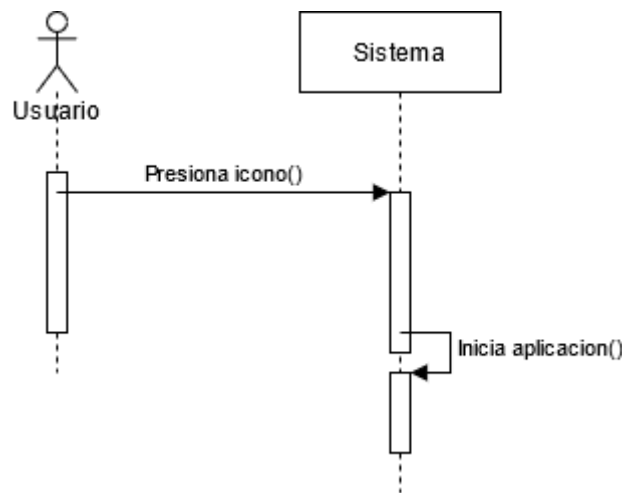
Nombre:	Transformar texto en audio
Autor:	Christian Cáceres
Fecha:	22/12/2020
Descripción: La aplicación transforma el texto encontrado a formato audio.	
Actor: Aplicación.	
Precondiciones: La aplicación debe tener acceso a los archivos del smartphone. Debe existir un archivo de texto.	
Flujo Normal: La aplicación detecta texto	
Aplicación	Sistema
1.- Busca el archivo de texto que se generó el C.U.S Reconocer texto 2.- El texto es procesado en audio.	
Postcondiciones: Se genera un archivo de audio	

Nombre:	Reproducir audio
Autor:	Christian Cáceres
Fecha:	22/12/2020
Descripción: La aplicación reproduce el archivo de audio con el texto que se desea leer.	
Actor: Usuario.	
Precondiciones: Debe existir un archivo de audio con el texto que se pretende leer.	
Flujo Normal:	
Usuario	Aplicación

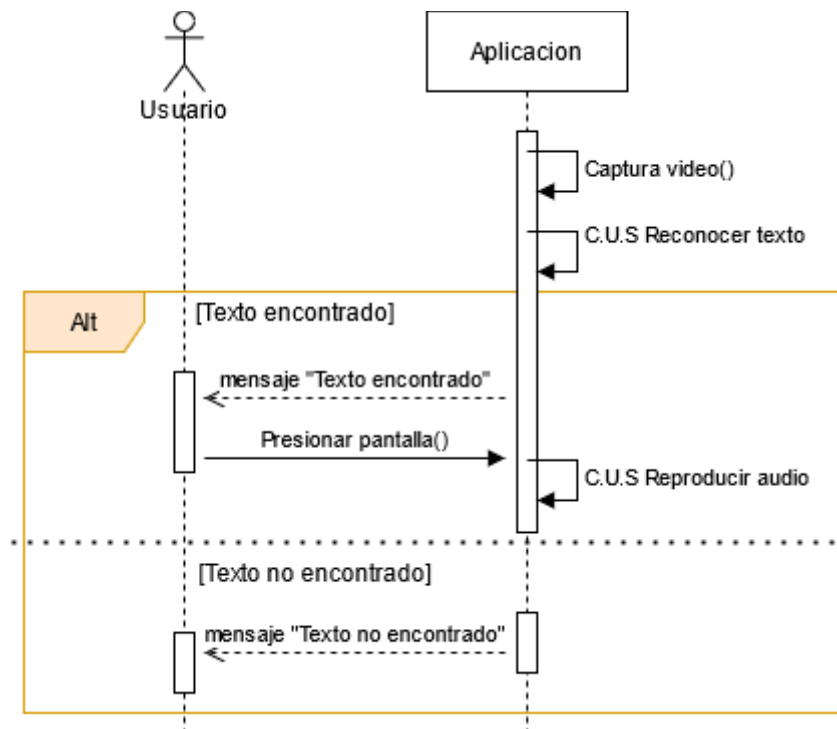
	1.- La aplicación busca el audio generado en el C.U.S Transformar texto en audio 2.-La aplicación reproduce el audio
Flujo Alternativo: El usuario detiene el audio	
Usuario	Aplicación
2.1- El usuario presiona la pantalla	2.2.- Detiene la reproducción del audio.

5.1.3. Diagrama de Secuencia

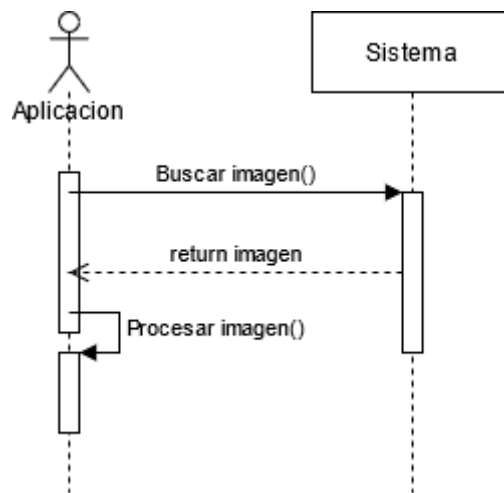
5.1.3.1. Inicia Aplicación



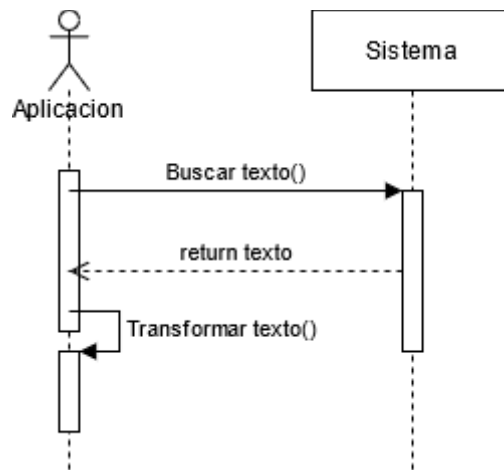
5.1.3.2. Capturar Video



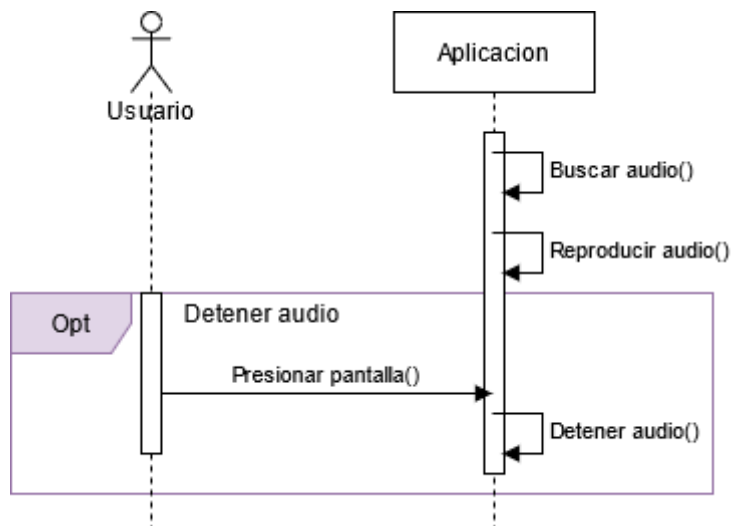
5.1.3.3. Reconocer texto



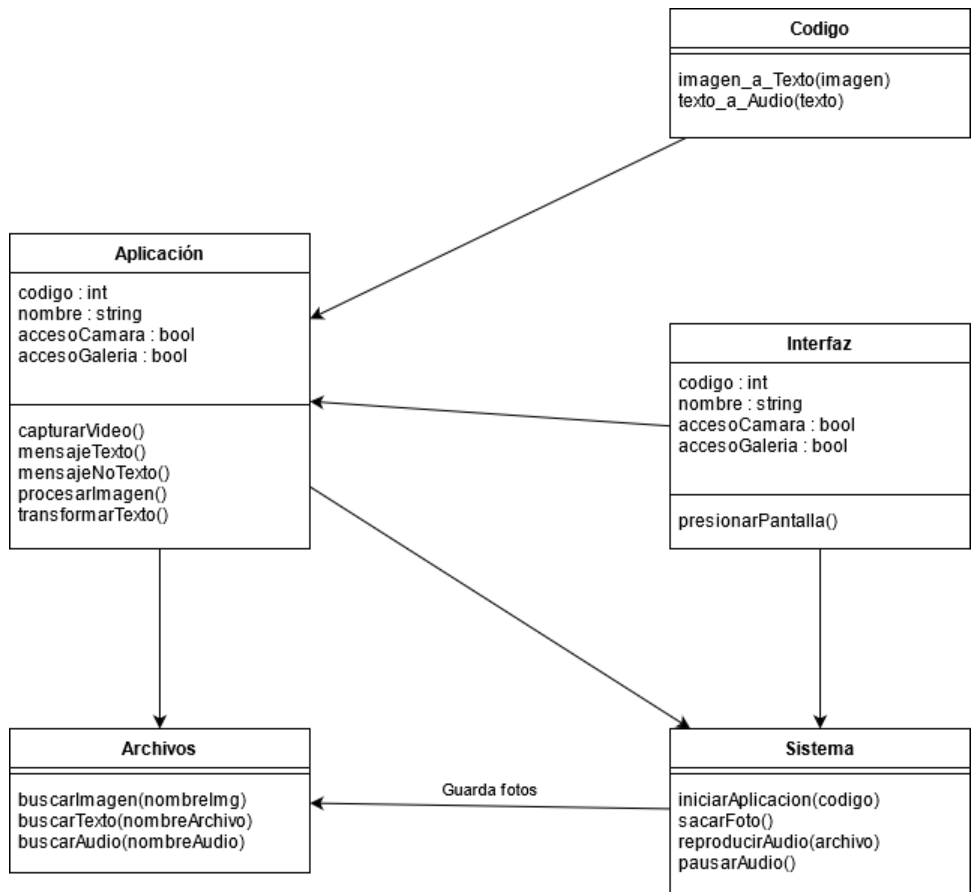
5.1.3.4. Transformar texto en audio



5.1.3.5. Reproducir Audio



5.1.4. Modelo de Clases



5.1.5. Descripción de la Arquitectura

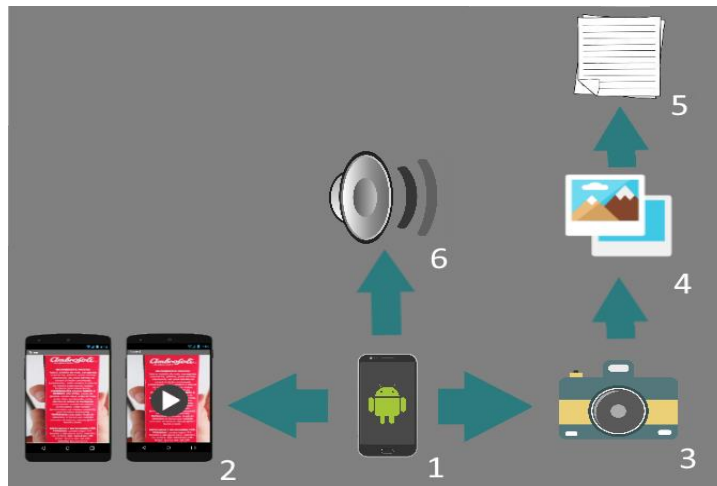


Ilustración 4: Arquitectura del proyecto

- Dispositivo móvil que utiliza un software con el que el cliente puede reproducir un audio del texto reconocido en una foto.
- Interfaz de un software con el que se controla la aplicación.
- Cámara del dispositivo móvil con la que se captura la foto a analizar.
- Foto tomada por la cámara.
- Texto extraído de la foto.
- Dispositivo de reproducción de audio con el que el software reproducirá el texto.

5.1.6. Diseño de la Interfaz de Usuario



Ilustración 5: Vista inicial

En la Ilustración 5 se aprecia la interfaz de vista inicial de la aplicación, en esta vista la aplicación enviará un mensaje de voz cuando haya encontrado texto, luego el usuario podrá tocar una vez la pantalla para sacar reproducir el texto.



Ilustración 6: Vista de reproducir audio

En la Ilustración 6 se aprecia la interfaz de vista al detectar texto de la aplicación, en esta vista el usuario podrá tocar una vez la pantalla para detener la reproducción del audio correspondiente.

5.1.7. Especificación de requerimientos

Requerimiento Funcional	Descripción
La aplicación debe utilizar la cámara del smartphone	La aplicación debe poder acceder a la cámara del smartphone y debe ser capaz de tomar fotos y guardar archivos.
La aplicación debe poder reconocer texto	La aplicación debe tener la capacidad de reconocer texto a través de la cámara
La aplicación debe poder reproducir audio	La aplicación debe tener la capacidad de reproducir el texto a través del altavoz o audífonos

Requerimiento no Funcional	Descripción
La aplicación debe estar hecha en Python 3.6.8	La aplicación debe ser programada utilizando la versión de Python 3.6.8
La aplicación debe utilizar software OpenCV 4.0	Las imágenes que se capturen con la aplicación deben ser procesadas por el software OpenCV 4.0
La aplicación debe reconocer texto en un tiempo determinado.	La aplicación debe poder reconocer texto en menos de 1 minuto.
La aplicación debe reconocer texto correctamente.	La aplicación debe reconocer el texto que se quiere leer de manera correcta y fidedigna.
La aplicación debe reproducir texto en forma de audio correctamente.	La aplicación debe reproducir el texto que se quiere leer mediante un audio claro y entendible.
La aplicación debe ser instalada en Android.	La aplicación debe ser instalada en un dispositivo móvil Android 5.0.2 o superior.

5.2. Herramientas y técnicas

Herramientas: Google Docs, Microsoft Office, OneDrive, Visual Studio Code, Pycharm.

Técnicas: Prueba y error, Dividir para conquistar, etc.

6. Planificación de procesos de soporte

6.1. Planificación de la documentación

Manual de usuario: Usuario e Instalación. Este documento contendrá las indicaciones para que el usuario pueda manejar la aplicación de manera correcta.

Wiki del proyecto: Blog donde se documenta detalladamente la realización del proyecto y su propósito.

Documentación del código: Este documento detallara la finalidad de cada función dentro del código que compone la aplicación.

7. Problemas Encontrados

Durante la investigación de la API de Google "Cloud Vision" que permite reconocer texto, nos encontramos con un problema al momento de utilizarla, ya que esta API es de pago, por lo que el grupo tuvo que decidir si seguir utilizando dicha API o elegir otra opción.

8. Soluciones Propuestas

Eventualmente el grupo decidió a optar por utilizar Tesseract, un software OCR gratis que también permite reconocer texto, casi igual de competente que la API de Google.

9. Conclusión

Con la finalidad de orientar al lector de este informe a tener un resumen conciso de los temas más destacados encontrados en este informe se pueden destacar los siguientes puntos.

El uso de OCR es tan variado como complejo cuando se propone implementar en diversos temas. Dependiendo de la finalidad con la que se pretende implementarlo puede encontrarse con diversas complejidades.

Además, es importante que se haga una buena investigación al inicio del proyecto para tener una buena base y evitar retrasos.