

**DESARROLLO DE UNA APLICACIÓN MÓVIL MEDIANTE EL DISEÑO DE ICONOS
HÁPTICOS PARA EL APRENDIZAJE EN LOS ESTUDIANTES CON DISCAPACIDAD
VISUAL DE LA UNIVERSIDAD DE PAMPLONA SEDE VILLA ROSARIO.**

**Autor:
Dicson Tarazona Tarazona**



Directora: Dra. Rocío De Belén Contreras Manrique

Universidad de Pamplona

Facultad de Ingenierías y Arquitectura

**Departamento de Ingenierías Electrónica, Eléctrica, Sistemas y Telecomunicaciones,
EEST**

Ingeniería de Sistemas

Villa del Rosario, Colombia.

2022

DEDICATORIA

Doy gracias a Dios y a la Virgen por haberme permitido llegar hasta este punto tan importante en mi formación profesional.

A mis padres Diorgen Tarazona Castilla y Sonia Tarazona por su apoyo incondicional, brindándome la confianza, oportunidad y recursos para lograrlo.

A mi hermano José Luis Tarazona por sus palabras en este camino de mis estudios y el apoyo necesario para formarme profesionalmente.

AGREDECIMIENTO

Agradezco en primer lugar a Dios quien me ha dado día a día la fortaleza y perseverancia necesaria para alcanzar mis metas y llegar hasta este punto.

A mi familia por motivarme para que culminara este ciclo que un día inicié y por su apoyo en todo momento.

A mi directora de tesis, Ingeniera Rocío de Belén Contreras Manrique por su asesoría, paciencia, colaboración y dedicación para que terminara este proyecto.

Mis amigos y compañeros con los cuales compartí momentos de tristeza, felicidad; pero que hoy se ven reflejados en un gran logro. Muchas gracias por ser un apoyo en este camino.

A todos los docentes de la carrera de Ingeniería de sistemas, que hicieron posible mi aprendizaje durante estos años de vida universitaria. De todos me llevo algo muy especial y sé que lo aprendido jamás se olvidara.

RESUMEN

En el presente trabajo de investigación se implementa una aplicación para uso estudiantil en condición de discapacidad visual basado en la plataforma Android, se pretende facilitar el proceso de aprendizaje a través del uso de las TIC. Por lo tanto, la funcionalidad consiste en que, mediante esta aplicación móvil, el estudiante pueda interpretar diversos documentos utilizando la transformación de archivos de texto a voz. Adicionalmente se incorporan iconos hápticos por medio de la vibración, con el fin de mejorar la experiencia de percepción en a la aplicación móvil. Por lo tanto, esto ayudara a mejorar la calidad de vida de los estudiantes con discapacidad visual en el programa de Derecho.

Para el desarrollo de esta aplicación se ha utilizado el IDE oficial de Android, llamado Android Studio, se ha utilizado el servidor Firebase ubicado en la nube para el almacenamiento de la base de datos y uso de los servicios desde la aplicación Android. Con la utilización de la percepción háptica(táctil) no depende de la visual; a través del sentido del tacto activo, es posible extraer la información necesaria para generar representaciones icónicas de los objetos. Es de gran utilidad para comprender mejor ese mundo visual de los medios de una forma de explorada de generar iconos mentales es mediante la percepción háptica o percepción táctil, la cual depende del sentido del tacto, y que en personas invidentes resulta de vital importancia a través de los procesos sensitivos.

Palabras claves:

App, TIC, Interpretar, Icono háptico, Percepción, Android, Firebase.

SUMMARY

In this research work an application for student use is implemented in a condition of visual disability based on the Android platform, it is intended to facilitate the learning process through the use of ICT. Therefore, the functionality is that, through this mobile application, the student can interpret various documents using the transformation of text-to-speech files. Additionally, haptic icons are incorporated through vibration, in order to improve the perception experience in the mobile application. Therefore, this will help to improve the quality of life of students with visual disabilities in the Law program.

For the development of this application the official IDE of Android, called Android Studio, has been used, the Firebase server located in the cloud has been used for the storage of the database and use of the services from the Android application. With the use of haptic perception (tactile) does not depend on the visual; Through the sense of active touch, it is possible to extract the necessary information to generate iconic representations of objects. It is very useful to better understand that visual world of the media in an explored way to generate mental icons is through haptic perception or tactile perception, which depends on the sense of touch, and which in blind people is of vital importance through of the sensitive processes.

Keywords:

App, ICT, Interpret, Haptic icon, Perception, Android, Firebase.

Tabla de Contenido

Contenido	
INTRODUCCIÓN.....	15
CAPÍTULO 1. PROBLEMA DE LA INVESTIGACIÓN.....	17
1.1. Planteamiento del Problema	17
1.1.1. Formulación del Problema.....	18
1.2. Justificación.....	19
1.3. Objetivos	20
1.3.1. Objetivo general	20
1.3.2. Objetivos específicos	20
1.3.3. Acotaciones.....	21
CAPÍTULO 2. MARCO TEÓRICO.....	22
2.1. Antecedentes	22
2.1.1. Antecedentes Nivel Regional.....	22
2.1.2. Antecedente Nivel Nacional	23
2.1.3. Antecedentes Nivel Internacional	23
2.2. Bases Teóricas.....	25
2.2.1. Las TIC como herramienta cognitiva.....	25
2.2.2. Inclusión y equidad educativa en el contexto de las TIC.....	25
2.2.4. Metodología Scrum	26
2.2.5. Artefactos en la metodología Scrum.....	27
2.2.5.1. Product Backlog.....	27
2.2.5.2. Sprint Backlog	27
2.2.5.3. Burndown del Sprint.....	27
2.2.5.4. Backlog de Impedimentos	27

2.2.6. Herramienta Android Studio	28
2.2.8. Herramientas CTV	28
2.2.9. Estructura de un conversor texto-voz	28
2.2.10. Icono Háptico	30
2.3. MARCO LEGAL	30
CAPITULO 3. MARCO METODOLOGICO	33
3.1. Enfoque de la Investigación	33
3.2. Diseño de la investigación	33
3.3. Definición de la población y selección de la muestra	34
CAPÍTULO 4. ANÁLISIS Y DISEÑO DEL SISTEMA	40
4.1.3.1. Interfaces de usuario.....	43
4.1.3.2. Interfaces de Hardware.....	43
4.1.3.3. Interfaces de Software.....	44
4.2.1.2.1. Descripción del Diagrama de Caso de Uso Estudiante.....	46
CAPÍTULO 5. DESARROLLO DEL PROYECTO	58
5.1. Determinar herramientas Tecnológicas en el procesamiento de conversor de formatos digital a audio para el proceso de aprendizaje en los estudiantes con discapacidad visual de la Universidad de Pamplona.....	58
5.1.1. Herramientas Tecnológicas para dispositivos (App)	58
5.1.1.1. Android Studio.....	59
5.1.1.2. Firebase.....	59
5.1.1.3. Software de Sintetización de Texto a Voz	59
5.1.1.5. TextToSpeech en Android	60
5.1.1.6. Herramienta de Autenticación Biométrica en Android	62
5.1.1.7. Herramienta de vibración en Android.....	64
5.2. Analizar las metodologías de desarrollo de aplicaciones móviles enfocadas en el diseño de sistemas utilizados por personas en condición de discapacidad visual.....	65
5.2.1.1. La Metodología Scrum, se seleccionó para la aplicación móvil Vision Access. A continuación se describen cada uno de sus componentes:	67
5.2.1.1.1. Product Backlog.....	67
5.2.1.1.2. Sprint Backlog.....	67

5.2.1.1.2.1. Tablas de Sprint Sprint 0 - Análisis de Requerimientos	68
5.2.1.1.3. Roles	75
5.3. Diseñar la estructura de la aplicación de iconos hápticos para el aprendizaje en estudiantes con discapacidad visual.	76
5.3.2. Configuración de las herramientas	76
5.3.3. Procesos del desarrollo de la aplicación.	76
5.3.3.1. Registro de Nuevo Usuario.....	76
5.3.3.1.1. Validación de los datos registro nuevo usuario	77
5.3.3.2.1. Validación de los datos iniciar sesión	80
5.3.3.3. Opción de Autenticación Biométrica local	83
5.3.3.3.1. Validaciones de acceso biométrico	84
5.3.3.3.2. Código de Validaciones de acceso biométrico	85
5.3.3.4. Interfaz Principal de usuario	89
5.3.3.5. Cerrar sesión de usuario.....	94
5.4. Validar el uso de la Aplicación móvil en estudiantes con discapacidad visual de la Universidad de Pamplona Sede Villa Rosario.....	95
CONCLUSIONES.....	106
CAPITULO 7. REFERENCIAS BIBLIOGRÁFICAS	107
ANEXOS.....	110
8.3. Manual de Usuario	111
8.3.1. Introducción	111
8.3.2. Requerimientos	111
8.3.3. Instalación	111
8.3.4. Uso de la aplicación VisionAccess	112
8.3.4.3. Menú principal	115

Lista de Ilustraciones

Ilustración 1. Metodología Scrum	26
Ilustración 2. Procesos necesarios para la conversión texto-voz.....	29
Ilustración 3. Actitud del estudiante de la Unipamplona con discapacidad visual en las herramientas Informáticas o TI.....	37
Ilustración 4. Uso de herramientas en los estudiantes de la universidad con discapacidad visual ...	38
Ilustración 5. Estrategias pedagógicas para la educación incluyente en estudiantes con discapacidad visual en la Unipamplona	39
Ilustración 6. Caso de Uso General Fuente: Propia.....	45
Ilustración 7. Caso de Uso Específico Fuente: Elaboración propia	46
Ilustración 8. Diagrama de Secuencia Estudiante con Discapacidad Visual Fuente: Propia	52
Ilustración 9. Estructura tipo JSON de los datos de la aplicación Fuente: Propia	53
Ilustración 10. Dependencias de FireBase	54
Ilustración 11. Añadiendo nuevo proyecto a Firebase	54
Ilustración 12. Notificación de proyecto agregado a Firebase	55
Ilustración 13. Reglas de Seguridad de la Base de datos en Firebase	55
Ilustración 14. Ubicación del Archivo JSON dentro del proyecto en Android.....	56
Ilustración 15. Comando FirebaseFirestore	56
Ilustración 16. Código CollectionReference	57
Ilustración 17. Herramientas de desarrollo	58
Ilustración 18. Burndown chart Sprint 0	68
Ilustración 19. Burdonw chart.....	69
Ilustración 20 Burdown chart Sprint 2	70
Ilustración 21. Burdown chart Sprint 3	71
Ilustración 22. Burdown chart Sprint 4	72
Ilustración 23. Burdown chart Sprint 5	73
Ilustración 24. Burdown chart Sprint 6	74
Ilustración 25. Burdown chart Sprint 7	75
Ilustración 26. Registro de Nuevo Usuario	76
Ilustración 27. Código de registro de nuevo usuario.....	78
Ilustración 28. Programación para Registrar Nuevo Usuario	78
Ilustración 29. Programación para emitir vibración (icono haptico).....	79
Ilustración 30. Iniciar Sesión.....	80
Ilustración 31. Código de Inicio de sesión loguearUsuario ()	81
Ilustración 32. Parte II de Código de Inicio de sesión loguearUsuario ()	82
Ilustración 33. Código de Incorporación del método vibrator en botón registrar usuario	83
Ilustración 34. Interfaz de autenticación biométrica	84
Ilustración 35. Código de declaración de Métodos usados en Validaciones de acceso biométrico..	85
Ilustración 36. Código de Método onSdkVersionNotSupported().....	86
Ilustración 37. Código de Método onBiometricAuthenticationNotSupported().....	86
Ilustración 38. Código de Método onBiometricAuthenticationNotAvailable().....	86
Ilustración 39. Código de Método onBiometricAuthenticationPermissionNotGranted().....	87
Ilustración 40. Código de Método onBiometricAuthenticationInternalError(String error)	87
Ilustración 41. Código de Método onAuthenticationFailed()	87
Ilustración 42. Código de Método onAuthenticationCancelled()	87
Ilustración 43. Código de Método onAuthenticationSuccessful()	88

Ilustración 44. Código de Método onAuthenticationHelp(int helpCode, CharSequence helpString)	88
Ilustración 45. Código de Método onAuthenticationError(int errorCode, CharSequence errString)	89
Ilustración 46. Interfaz Principal de usuario	89
Ilustración 47. Interfaz Leer Documento PDF	90
Ilustración 48. Interfaz de CHOOSE_PDF_FROM_DEVICE	91
Ilustración 49. Interfaz de extracción del texto del documento junto a su ruta de almacenamiento.	91
Ilustración 50. Código de extracción de texto del documento PDF	92
Ilustración 51 Interfaz Botón leer	92
Ilustración 52. Código de lectura de texto	93
Ilustración 53. Codificación de los métodos contenidos en la clase TTSTManager	94
Ilustración 54. Interfaz botón Cerrar sesión de usuario	94
Ilustración 55. Codificación botón Cerrar sesión de usuario	95
Ilustración 56. Encuesta realizada después de validación a la aplicación VisionAccess	96
Ilustración 57. Grafica representativa del resultado a la pregunta ¿Cómo aprecia el uso de la aplicación VisionAccess?	96
Ilustración 58. Grafica representativa del resultado a la pregunta ¿Le gusta la ayuda que brinda la aplicación para dar lectura a documentos digitales?	97
Ilustración 59. Grafico Representativo al ala pregunta ¿Qué opina de la aplicación VisionAccess?	98
Ilustración 60. Grafica representativa de la pregunta ¿Qué modificaciones le haría a la aplicación?	99
Ilustración 61. Manual usuario – Instalación	112
Ilustración 62. Manual usuario – Formulario login	112
Ilustración 63. Manual usuario – interfaz acceso con huella	113
Ilustración 64. Manual usuario – registro nuevo usuario	113
Ilustración 65. Manual usuario – mensaje campo vacío en contraseña	114
Ilustración 66. Manual usuario – mensaje campo vacío en correo	114
Ilustración 67. Manual usuario – mensaje falta carácter	114
Ilustración 68. Manual usuario – mensaje contraseña mayor a 6 dígitos	114
Ilustración 69. Manual usuario – mensaje de error al acceder	115
Ilustración 70. Menú principal	115
Ilustración 71. Interfaz Leer Documento	116
Ilustración 72. Acción cargar Pdf	117
Ilustración 73. Acción seleccionar Pdf	117
Ilustración 74. Resultado de la acción seleccionar Pdf	118
Ilustración 75. Interfaz de la acción leer	118
Ilustración 76. Interfaz de la acción cerrar sesión	118

Lista de Tablas

Tabla 1. Matriz de Operacionalización de las variables.....	36
Tabla 2. <i>Actitud del estudiante de la Unipamplona con discapacidad visual en las herramientas Informáticas o TI</i>	37
Tabla 3. Uso de herramientas en los estudiantes de la Unipamplona con discapacidad visual.	38
Tabla 4. Estrategias pedagógicas para la educación incluyente en estudiantes con discapacidad visual en la Unipamplona.....	39
Tabla 5. Requerimiento Funcional 1 - Registro usuarios.....	40
Tabla 6. Requerimiento Funcional 2 - Iniciar sesión	40
Tabla 7. <i>Requerimiento Funcional 3 - Reconocimiento texto a voz</i>	41
Tabla 8. Requerimiento Funcional 4- Interpretación de iconos hapticos.....	41
Tabla 9. <i>Requerimiento Funcional 5 -Escucha del texto procesado</i>	41
Tabla 10. Requerimiento Funcional 6 -Autenticación biométrica	42
Tabla 11. Requerimiento No Funcional - Interfaz sistema	42
Tabla 12. Requerimiento No Funcional - Disponibilidad	42
Tabla 13. Requerimiento No Funcional - Seguridad de la información	43
Tabla 14. Caso de uso general.....	45
Tabla 15. Diagrama de Caso de Uso Estudiante	47
Tabla 16 Caso de uso - Registrar usuario.....	47
Tabla 17. Caso de uso - Autenticación biométrica.....	48
Tabla 18. Caso de uso - iniciar sesión.....	49
Tabla 19. Caso de uso - Escuchar texto sintetizado	50
Tabla 20. Constantes más comunes que utiliza esta herramienta TextToSpeech	61
Tabla 21. Métodos más comunes que utiliza esta herramienta TextToSpeech.....	62
Tabla 22. Descripción métodos Autenticación biométrica	63
Tabla 23. Métodos principales de herramienta de vibratorManager.....	64
Tabla 24. Principales metodologías ágiles	65
Tabla 25. Descripción general de Product Backlog	67
Tabla 26. Sprint 0. Análisis de Requerimientos.....	68
Tabla 27. Tiempo en horas Sprint 0.	68
Tabla 28. Sprint 1.Base de Datos.	69
Tabla 29. Tiempo en Horas Sprint 1	69
Tabla 30. Sprint 2. Diseño de la Aplicación.....	70
Tabla 31. Tiempo en Horas Sprint 2.	70
Tabla 32. Sprint 3.Registro de Usuarios	71
Tabla 33. Tiempo en horas Sprint 3.	71
Tabla 34. Sprint 4.Autenticación de Usuarios.....	72
Tabla 35. Tiempo en horas Sprint 4	72
Tabla 36. Sprint 5.Reconocimiento de Voz	73
Tabla 37. Tiempo en horas Sprint 5.	73
Tabla 38. Sprint 6.Interpretacion de iconos haticos.	74
Tabla 39. Tiempo en horas Sprint 6.	74
Tabla 40. Sprint 7.Pruebas de Usabilidad	75
Tabla 41. Tiempo en horas Sprint 7	75
Tabla 42. Roles de Scrum para la aplicación	75
Tabla 43. Descripción de la validación de los datos registro nuevo usuario.....	77
Tabla 44. Validación de los datos en Inicio de Sesión.....	80
Tabla 45. Validaciones de acceso biométrico	84

Tabla 46. Prueba de caja negra - Registro de usuario	102
Tabla 47. Prueba de caja negra para iniciar sesión.....	103
Tabla 48. Prueba de caja negra para Autenticación Biométrica.....	104
Tabla 49. Población con discapacidad visual a nivel nacional.....	119
Tabla 50. Organizaciones de discapacidad visual a nivel regional	122

INTRODUCCIÓN

Una forma poco explorada de generar iconos mentales es mediante la percepción háptica o percepción táctil del cual depende del sentido del tacto, y que en personas con discapacidad visual resulta de vital importancia. Por lo tanto, las personas invidentes dependen principalmente de su sentido del tacto y del oído para obtener conocimiento de él mismo y de su medio ambiente. Es entendible esa falta de preocupación por las características propias de los medios denominados audiovisuales, ya que, por ser visuales, no se enfocan a lo táctil. Espinosa Castañeda, Raquel, & Medellín Castillo, Hugo Iván (2014).

Diferentes autores han propuesto en el desarrollo de la comunicación audiovisual para personas de con discapacidad visual, pero con la adición de la percepción virtual táctil de su entorno, lo cual se puede lograr mediante el uso de las tecnologías. además, se puede inferir que la percepción háptica permite obtener información del entorno, tanto a videntes como a invidentes, tomando en cuenta que en el caso de las personas con discapacidad visual “el cerebro tiene la capacidad de refuncionalizar áreas que están asociadas a funciones visuales, permitiéndoles emitir una respuesta a información táctil y auditiva”. Las dificultades de aprendizaje que experimentan con discapacidad visual guardan relaciones muy estrechas con el desarrollo cognitivo.

Por lo tanto, a pesar de que la visión desempeña un papel vital para el aprendizaje y que estos estudiantes necesitan equipos especiales o una adaptación de los procesos o de los materiales educativos para aprender, según Cruickshank (1986), hoy en día los avances tecnológicos nos permite intervenir en la problemática de los estudiantes con discapacidad visual en el aula; partiendo en el

desarrollo de una aplicación móvil mediante el diseño de iconos hápticos para el aprendizaje en los estudiantes con discapacidad visual. Esto permitirá al estudiante generar su propio material didáctico; donde se puedan utilizar una APP para leer un pdf o cualquier texto o documento; es decir puedes escuchar el audio de ese documento en voz alta.

CAPÍTULO 1. PROBLEMA DE LA INVESTIGACIÓN

1.1. Planteamiento del Problema

En la sociedad de la información y el conocimiento en que nos encontramos, el potencial de interacción individuo-máquina se está convirtiendo en una realidad preocupante para los ciudadanos que tienen dificultades de acceso a la tecnología. Hay una gran diferencia en los tiempos que marcan los avances en la tecnología estándar y los desarrollos específicos de ayudas técnicas para hacer accesible la información. En esta situación no cabe otra solución que el compromiso de la sociedad por trabajar en el desarrollo de productos desde los criterios del diseño para todos. Sin embargo, mientras esto no ocurra es inevitable seguir trabajando en el desarrollo de productos que permitan el acceso a la tecnología.

Es entonces que por medio de una aplicación móvil se hace buen uso de la incorporación de las TIC en esta comunidad de limitación visual, la cual, a través del teléfono celular, tendría la oportunidad de comunicarse e incorporarse de una mejor forma al proceso de enseñanza-aprendizaje. Por lo tanto, el uso de las herramientas tecnológicas nos da la posibilidad de poder utilizar vibro táctiles complejos se revela como una mejora notable en la usabilidad de los celulares, al ayudar a los usuarios a localizar los iconos en la pantalla de una forma más ágil y eficiente. Se puede inferir asimismo que el uso de un refuerzo del aprendizaje mejora de forma significativa los resultados obtenidos por todos los usuarios, resultando ser esta mejora especialmente relevante en los usuarios con discapacidad visual.

Por otra parte, poder utilizar convertidores de documentos a audio para los estudiantes invidentes de la universidad de pamplona, que no cuentan con un mecanismo efectivo de comunicación que permita desarrollar habilidades de aprendizaje en su entorno académico. Adicionalmente con esta tecnología va facilitar el proceso de aprendizaje y desenvolvimiento en su entorno y vida diaria.

1.1.1. Formulación del Problema

¿Cómo ayudar a los estudiantes en el proceso de aprendizaje mediante el diseño de iconos hápticos con personas con discapacidad visual de la universidad de pamplona?

1.2. Justificación

La universidad de pamplona actualmente está mejorando procesos institucionales que fomentan la inclusión con toda clase de la comunidad universitaria, permitiendo a los más vulnerables poder acceder a tecnologías que les permite realizar una serie de actividades que de otra manera presentarían grandes dificultades para ser realizadas de manera independiente. Es así que en el desarrollo de este proyecto se pretende aportar al mejoramiento de procesos académicos y de calidad de vida de personas con discapacidad visual mediante una aplicación móvil desarrollada, la cual contará un diseño que le permita tener accesibilidad a la interpretación por medio de audio, la cual le permitirá acceder a las diferentes extensiones de formatos para poder convertir los archivos de documentos en forma de audio.

Para facilitar la accesibilidad se pretende incorporar al diseño de la aplicación un modelo háptico ya que este interviene directamente con el tacto de la persona, ofreciendo privacidad, y permitiendo disponibilidad de los demás sentidos como la audición de la información del archivo y poder contribuir al mejoramiento de vida de las personas con discapacidad visual para los estudiantes de la universidad de pamplona sede villa Rosario.

1.3. Objetivos

1.3.1. Objetivo general

- Desarrollar una aplicación móvil mediante el diseño de iconos hápticos para el aprendizaje en los estudiantes con discapacidad visual de la Universidad de Pamplona Sede Villa Rosario.

1.3.2. Objetivos específicos

- Determinar herramientas Tecnológicas en el procesamiento de conversor de formatos digital a audio para el proceso de aprendizaje en los estudiantes con discapacidad visual de la Universidad de Pamplona.
- Analizar las metodologías de desarrollo de aplicaciones móviles enfocadas en el diseño de sistemas utilizados por personas en condición de discapacidad visual.
- Diseñar la estructura de la aplicación de iconos hápticos para el aprendizaje en estudiantes con discapacidad visual.
- Validar el uso de la aplicación móvil con interfaz mediante el diseño de iconos hápticos para la comunicación en estudiantes con discapacidad visual del programa de derecho de la Universidad de Pamplona Sede Villa Rosario.

1.3.3. Acotaciones

- Enfocado a estudiantes de derecho con discapacidad visual de la Universidad de Pamplona sede Villa del Rosario.
- Desarrollado en el entorno Android Studio 3.5
- Lenguaje de codificación utilizado; Java
- Aplicación realizada para dispositivos móviles con sistema operativo Android.

CAPÍTULO 2. MARCO TEÓRICO

2.1. Antecedentes

Cada antecedente tiene como propósito aportar una comprensión de lo que representan las nuevas alternativas de aprendizaje a través aplicación móvil mediante el diseño de iconos hápticos para el aprendizaje en los estudiantes con discapacidad visual.

2.1.1. Antecedentes Nivel Regional

Según **Cardenas, L.** (2015). Población con discapacidad está conectada con las TIC, de la Oficina de Prensa de la gobernación de norte de Santander. Allí se evidencia el apoyo del gobierno a la implantación de las TIC en personas con discapacidades visuales por medio de aplicativos y a la capacitación de comunidad educativa para socializar la inclusión de dichas personas.

Basado en el relato de **César Martínez**, el invidente que les enseña tecnología a los reclusos de Cúcuta (2015). Se puede apreciar como un joven cucuteño con discapacidad visual es apoyado por el Ministerio de Tecnologías de la Información y las Comunicaciones (MinTIC) para fomentar el aprendizaje de la tecnología en comunidades con poco acceso a esta, como lo son las personas privadas de la libertad. Para ello el gobierno dispone de una plataforma llamada Convertic el cual brinda un software lector de pantalla.

2.1.2. Antecedente Nivel Nacional

Orientación de Pasajeros con Discapacidad Visual dentro del Sistema de Transporte Masivo Transmilenio, mediante Geolocalización Satelital. **Garay F, Camargo J, Segura D, Rincón N.**

Aunque el sistema de transporte masivo Transmilenio posee medios de audio informativo, éstos son insuficientes para permitir que pasajeros en condición de discapacidad visual, puedan desplazarse con mayor facilidad, ubicar puertas de acceso y rutas de desplazamiento, en este artículo se propone la utilización de un dispositivo portátil de bajo costo acondicionado con geolocalización satelital buscando mejorar la autonomía de desplazamiento de ese segmento de población dentro del sistema Transmilenio. Método: Se realizó la revisión de los desarrollos que hasta ahora se han hecho para mejorar el nivel de autonomía de las personas en condición de discapacidad visual, un análisis necesario para definir la tecnología más práctica para el desarrollo de un dispositivo aplicable en la ciudad de Bogotá, posteriormente se establecieron los módulos funcionales del dispositivo, incluyendo el sistema de almacenamiento de archivos, el de reproducción de sonido, la interfaz de usuario y el de geolocalización y ubicación dentro de Transmilenio.

2.1.3. Antecedentes Nivel Internacional

Ignacio cano nos presenta el **artículo** Fomento de la movilidad de invidentes mediante cartografía táctil 2. De la Universidad Politécnica de Valencia, Maestro I. La aspiración fundamental de estos dos artículos es fomentar el interés por la resolución de los problemas de movilidad de las personas discapacitadas desde una perspectiva cartográfica, poco abordada hasta el momento. Es cierto que en los últimos años, se ha puesto un empeño creciente en lo que es la mejora de la accesibilidad a las personas discapacitadas (prueba de ello es que el año 2003.

ha sido declarado Año Europeo de las Personas con Discapacidad). No obstante, quizás se ha olvidado que también se mejoraría de forma notable la movilidad de estas personas si se facilitara el conocimiento de ese espacio por el cual han de desplazarse. Es en ese campo en el cual pretende actuar este tipo de trabajos. En concreto, la aplicación que aquí se expone trata de servir como apoyo

a trabajadores, estudiantes y visitantes de la Universidad Politécnica de Valencia que posean alguna deficiencia visual severa.

Según **Galdon P.** (2014). Mejora en el interfaz de un dispositivo móvil para personas con discapacidad visual mediante el diseño de iconos hápticos. **Tesis doctoral** de la Universidad de Málaga. La mayoría de los interfaces de los actuales dispositivos móviles se basan en la visión para comunicar la información al usuario. Sin embargo, debido a que el usuario no se encuentra continuamente mirando la pantalla del dispositivo, y al reducido tamaño de las mismas, la interacción visual con dichos dispositivos suele complementarse con el uso de sonidos. No obstante, si el entorno es ruidoso el uso de sonidos resulta ineficaz, como también resulta ineficaz su uso si se requiere estar en silencio o se necesita el sentido del oído para otras tareas. En estas situaciones, el sentido del tacto puede convertirse en una alternativa para comunicar la información al usuario.

Vintimilla, G (2014). “Desarrollo e implementación de una aplicación móvil que traduzca el abecedario del lenguaje de señas a texto para ayuda a discapacitados auditivos mediante dispositivos móviles Android”, de la Universidad de las fuerzas armadas, Sangolquí-Ecuador, Ingeniería Electrónica en Redes y Comunicación de Datos. Este trabajo se basa en la forma de encontrar la arquitectura más óptima, que sea capaz de aprender y devolver una respuesta correcta por medio de redes neuronales para llegar al objetivo de reconocer letras y números. Basados en dos procedimientos, el tratamiento de la imagen y el entrenamiento de la red.

Aporte de **tesis** “diseño y desarrollo de prototipo de sistema de traducción instantánea de habla y transmisión en tiempo real, sobre el protocolo rtp utilizando tecnologías de reconocimiento de voz”. Según **Rojas, N; Araya, E. & Vidal, L.** (2006). Universidad Austral de Chile. La tesis propone un sistema capaz de capturar el flujo de voz del emisor integrando tecnologías de reconocimiento de voz avanzadas, traducción instantánea, y comunicación sobre un protocolo de Internet dedicado al transporte para enviar en tiempo real la información al receptor.

2.2. Bases Teóricas

2.2.1. Las TIC como herramienta cognitiva

Los nuevos ámbitos educativos que exigen las sociedades de la información y el conocimiento, destacan la importancia de repensar los elementos básicos de los **procesos de enseñanza y aprendizaje**, de ahí la necesidad de transformar estos en un marco de acción donde el educando, aprenda a aprender, a hacer, a ser y a convivir, erigiéndose el acompañamiento responsable de procesos de aprendizaje, el **desarrollo permanente de habilidades cognitivas** y la implementación de nuevas tecnologías como agentes externos de alta importancia educativa y garantes de un enfoque equitativo, pertinente e inclusivo.

2.2.2. Inclusión y equidad educativa en el contexto de las TIC

Según UNESCO (2003), la inclusión educativa se concibe como un proceso de gestión y respuesta a la **diversidad de necesidades** de todos los aprendices a través de la participación en la cultura, las comunidades y el propio aprendizaje. La inclusión implica cambios y modificaciones en contenido, enfoques, estructuras y estrategias, con la visión común que cubre a todos los niños de un rango apropiado de edad y la **convicción de que es responsabilidad** de cada sistema educativo, **formar a todos sin ningún tipo de discriminación y en óptimas condiciones de equidad y calidad.**

2.2.4. Metodología Scrum

La ingeniería de software juega un papel importante en el desarrollo, portabilidad, mantenibilidad, funcionalidad, fiabilidad y productividad del software, igualmente, las metodologías ágiles son procesos para desarrollar software **de manera rápida con gran facilidad de** adopción por los equipos de trabajo.

Scrum se centra la gestión de proyectos en situaciones en las que **es difícil planificar el futuro**, con mecanismos de control “proceso empírico”, donde **los bucles de realimentación constituyen el elemento central**. El software es desarrollado por un equipo de auto-organización **en incrementos (llamados “sprints”)**, empezando por la planificación y finalizando con un comentario. Las características que deben aplicarse en el sistema **se registran en un backlog**. Entonces, el dueño del producto decide qué elementos del backlog se deben desarrollar en el sprint siguiente. Los miembros del equipo coordinan su trabajo en un diario de stand-up de la reunión.

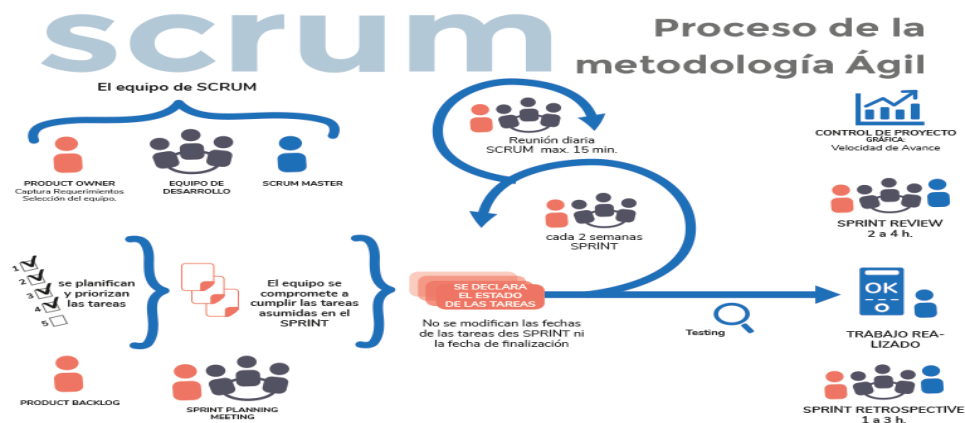


Ilustración 1. Metodología Scrum

Fuente: Stack Overflow

2.2.5. Artefactos en la metodología Scrum

2.2.5.1. Product Backlog

El product Backlog, es conocido como la lista de tareas a desarrollar definidas y especificadas en la primera reunión. Dado que se consideran los requerimientos emergentes, lo que significa que no se conoce ni se puede conocer de antemano todas y cada una de las características que se desea que contenga el producto, se considera el Product Backlog como un documento viviente, que requiere una constante preparación (grooming) para mantenerlo actualizado y útil con el paso del tiempo.

2.2.5.2. Sprint Backlog

Es el tablero de tareas, en el que se plasman las tareas del **product Backlog**, para verificar el estado en que se encuentran, este tablero, es una emulación semejante a la del tablero Kanban, en el que se identifican las columnas de Tareas por hacer, las que están en ejecución, en prueba y las terminadas, y, en el que, además, como es la filosofía de Kanban se puede identificar los cuellos de botella o represamiento de tareas.

2.2.5.3. Burndown del Sprint

El gráfico de burndown de tareas del sprint tiene por objeto ayudar al equipo en la monitorización de su progreso y ser el indicador principal que informa las **posibilidades de alcanzar su compromiso al finalizar el sprint**. El formato clásico requiere que el equipo estime la duración de cada tarea en **horas de forma diaria**. El burndown deberá completarse de forma tal que grafica **cuántas horas de trabajo restan para concluir el sprint**.

2.2.5.4. Backlog de Impedimentos

El backlog de impedimentos consiste en la lista de situaciones que están impidiendo que el equipo progrese. Estas son situaciones que el Scrum Máster debe eliminar con el fin de que el equipo trabaje de la mejor manera posible. Se debe entender que los impedimentos son de cualquier tipo, sea cual

sea la magnitud de lo que representan. Es evidente que de una rápida solución dependerá notoriamente el rendimiento del equipo, de allí que un buen Scrum Máster suela tener la meta de resolver estas situaciones en un tiempo máximo de 24 horas.

2.2.6. Herramienta Android Studio

Android Studio es un entorno de desarrollo integrado (IDE), basado en IntelliJ IDEA de la compañía JetBrains, que proporciona varias mejoras con respecto al plugin ADT (Android Developer Tools) para Eclipse. Android Studio utiliza una licencia de software libre Apache 2.0, está programado en Java y es multiplataforma. Además, fue presentado por Google el 16 de mayo del 2013 en el congreso de desarrolladores Google I/O, con el objetivo de crear un entorno dedicado en exclusiva a la programación de aplicaciones para dispositivos Android, proporcionando a Google un mayor control sobre el proceso de producción.

2.2.8. Herramientas CTV

Basado en la investigación (agustin,inaki,derro,eva,Inma), El sistema de **Conversión Texto a Voz de Código Abierto** Para Lenguas Ibéricas. Un sistema de conversión texto a voz (CTV) es un sistema que **convierte una entrada de texto en una salida en forma de señal de audio** cuyo contenido se corresponde con el mensaje del texto de entrada. Los sistemas CTV actuales pueden descomponerse en **dos módulos**. El primero, fuertemente **dependiente del idioma**, toma como entrada el texto a sintetizar y genera a su **salida etiquetas que describen lingüística, fonética** y prosódicamente dicha entrada. Estas etiquetas alimentan el segundo módulo, **el motor de síntesis** que en sí mismo es independiente del idioma. Este módulo emplea voces entrenadas a partir de bases de datos que son dependientes del idioma para sintetizar la señal de voz de salida en función de las etiquetas de entrada.

2.2.9. Estructura de un conversor texto-voz

El grupo de investigación tecnológica de la universidad técnica de Madrid describen los procesos que componen el tratamiento **lingüístico-prosódico**, como aquellos que se **encargarán de analizar el texto de entrada**, con el objetivo de **extraer** la información adecuada para la correcta y natural lectura del texto. Esta fase de la conversión texto-voz, es sobre la que se ha trabajado e investigado, durante la realización de este proyecto fin de carrera. El procesado acústico, el cual partirá de los

datos obtenidos en los procesos mencionados anteriormente, será necesario para generar la señal de voz. Por tanto, podemos decir, que todo sistema de conversión de texto a voz constará de dos grandes bloques: Procesado lingüístico-prosódico y Procesado acústico.

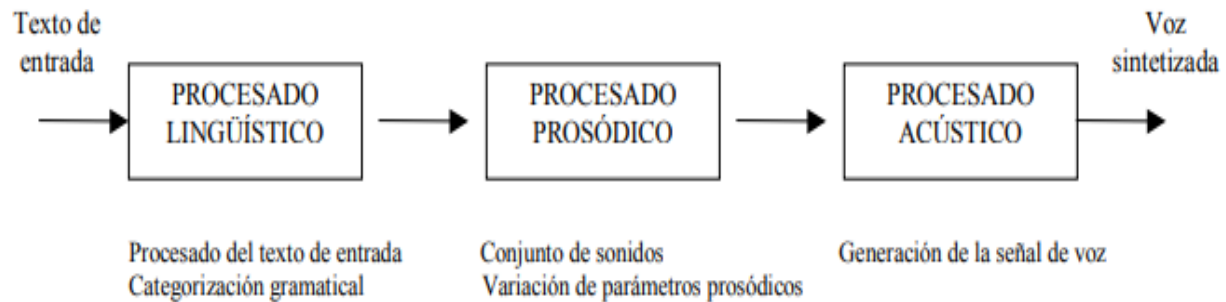


Ilustración 2. Procesos necesarios para la conversión texto-voz

Fuente: grupo de investigación tecnológica de la universidad técnica de Madrid

(a) Procesado lingüístico: El objetivo principal del procesado lingüístico es determinar los sonidos que hay que producir, y cómo producirlos para la correcta lectura del texto de entrada.

(b) Procesado prosódico: Este último módulo dentro del procesado lingüístico, es el encargado de recoger la información segmental y suprasegmental obtenida de los bloques anteriores, y así generar, a partir de ella, tres parámetros prosódicos: ritmo, entonación y pausado. Estas informaciones, ayudan a la generación de una prosodia más correcta, y así se puede dar la impresión de que el conversor "entiende" lo que está leyendo, aumentando la naturalidad de la voz sintética recibida por el oyente

(c) Procesado acústico: El objetivo de este bloque, es generar la forma de onda asociada a la voz sintetizada a partir de la cadena fonética y las variables de control prosadico..

2.2.10. Icono Háptico

Según **Espinosa Castañeda**, R. y Medellín Castillo, H. I. (2014) presentan la definición de icono que significa imagen o representación, un icono es un signo visual que representa otro objeto semejante a él. Es entonces que el icono puede sustituir a lo que representa. La imagen por su parte es una representación visual, aunque también se puede aplicar como extensión a otros tipos de percepción, como imágenes auditivas, olfativas, táctiles, kinestésicas, entre otros. Por su parte, el icono no sólo es un signo visual, sino sensorial que permite la representación de los objetos en la mente del que se expone a ese signo.

Adicionalmente Merleau-Ponty (1960). agregó “que cuando el niño se acerca a la etapa de operaciones formales, alcanza un nivel de resolución igual al de los niños videntes” Este concepto es amplio, pues implica que la representación mental puede lograrse, aunque la persona no tenga la posibilidad de ver, lo cual confirma que los ciegos tienen la capacidad de crear iconos mentales. Y es complementario a la definición de la palabra háptico que hace referencia a el nombre que se le da al proceso de sentir o tocar objetos virtualmente.

2.3. MARCO LEGAL

Según la ley 1341 de 2009 ; ley de TIC en Colombia, se definen las Tecnologías de la Información y la Comunicación (TIC), como el conjunto de recursos, herramientas, equipos, programas informáticos, aplicaciones, redes y medios, que permiten la compilación, procesamiento, almacenamiento y transmisión de información como voz, datos, texto, video e imágenes.

Por otra parte, en La Constitución Política de Colombia de 1991, en sus artículos 13, 47, 54 y 68, se contempla la responsabilidad del estado en dar atención especial a las personas con discapacidad y basándose en ella, el Congreso de la República incluyó en el Artículo 47 de la Ley 115 de 1993 o Ley Marco de la Educación, que determina la obligación de brindar, adecuadamente, las condiciones para la reivindicación del derecho a la educación a toda la población y específicamente, se refiere al derecho a la educación en aulas regulares para personas en condición de discapacidad, especialmente en el título III, capítulo 1.

Según el decreto 2082 de 1996, se refiere específicamente a la discapacidad, el cual reglamenta la atención educativa para personas con limitaciones o capacidades excepcionales. En la ley 361 de 1997⁴⁶ o Ley Marco de la Discapacidad en Colombia, se establecen mecanismos de integración social de las personas con limitación.

La Constitución Política de Colombia (Asamblea Nacional Constituyente, 1991) contempla la igualdad de todos los seres humanos, la inalienabilidad de los derechos de las personas sin discriminación alguna; la protección especial a personas que, por su condición económica, física o mental, se encuentren en condición de protección especial.

La Ley 115 (Congreso de Colombia, 1994) dedica el capítulo 1 del título III a la educación para personas con limitaciones o capacidades excepcionales y establece en el Artículo 46° que “la educación para personas con limitaciones físicas, sensoriales, psíquicas, cognoscitivas, emocionales o con capacidades intelectuales excepcionales es parte integrante del servicio público educativo” (párr. 1), y en el mismo Artículo 46° plantea que “los establecimientos educativos organizarán directamente o mediante convenio, acciones pedagógicas y terapéuticas que permitan el proceso de integración académica y social de dichos educandos” (párr. 2).

El Decreto 2082 de 1996 (Presidencia de la República de Colombia, 1996) afirma que la atención de la población con discapacidad y con capacidades o talentos excepcionales es de carácter formal, no formal (hoy en día formación para el trabajo y desarrollo humano) e informal y que será ofrecida en instituciones educativas estatales y privadas de manera directa o mediante convenio.

El Decreto 3020 de 2002 (Ministerio de Educación Nacional, Colombia, 2002), en el Artículo 11, establece que la entidad territorial debe atender los criterios y parámetros establecidos por el Ministerio de Educación Nacional para “fijar la planta de personal de los establecimientos que atienden a estudiantes con necesidades educativas especiales” (p. 2).

El Decreto 366 (Ministerio de Educación Nacional, Colombia, 2009), a través del cual se organizan los servicios de apoyo pedagógico para la atención del estudiantado con discapacidad y con capacidades o talentos excepcionales, en el marco de la educación inclusiva.

La Ley 1346 de 2009 (Congreso de Colombia, 2009) por medio de la cual se aprueba y adopta la Convención de Derechos de las Personas con Discapacidad.

La Ley Estatutaria 1618 de 2013 (Congreso de Colombia, 2013): establece las disposiciones para garantizar el pleno ejercicio de los derechos de las personas con discapacidad.

CAPITULO 3. MARCO METODOLOGICO

3.1. Enfoque de la Investigación

La investigación tiene como enfoque cuantitativo, no probabilístico; siguiendo la metodología de Hernández, Fernández y Baptista (2003), “el enfoque cuantitativo utiliza la recolección y el análisis de datos para contestar preguntas de investigación y probar hipótesis previamente hechas, confía en la medición numérica, el conteo y frecuentemente en el uso de estadística para establecer con exactitud patrones de comportamiento en una población” (p.5).

En todo trabajo investigativo es necesario determinar el tipo de investigación que se aplicará, de acuerdo al nivel o grado de profundidad necesario para su desarrollo, esto indica que dependiendo del objetivo que se busca se puede establecer el enfoque, para Hernández; Fernández & Baptista (2008) El tipo de investigación cuantitativo no experimental es aquel en el cual no se aplica una manipulación deliberada de las variables, pudiéndose observar el fenómeno tal como se da en su contexto natural, para después analizarlo. En la investigación propuesta de evaluar el uso de las tecnologías de la información y comunicación (TIC) en los procesos de su práctica pedagógica de los profesores de la facultad de ciencias empresariales de la Universidad Francisco de Paula Santander, siendo el tipo de investigación cuantitativo en el desarrollo del trabajo investigativo.

3.2. Diseño de la investigación

El diseño de la investigación es descriptivo, según Tamayo (2004) es aquella que comprende la descripción, registro, análisis e interpretación de la naturaleza actual, composición o procesos de los fenómenos. Este enfoque trabaja sobre las realidades de los hechos y sus características fundamentales, para presentarnos una interpretación correcta. (p. 54).

Además, la investigación es de tipo Exploratoria - Descriptiva, de enfoque cuantitativo, no probabilístico, con un muestreo intencional Fidias (2004) “consiste en la caracterización de un hecho, fenómeno o grupo con el fin de establecer su estructura o comportamiento. Los resultados de este tipo de investigación se ubican en un nivel intermedio en cuanto a la profundidad de los

conocimientos se refiere”. (p. 22). Según Selltiz (1980) y Fidias (2004) la investigación exploratoria es aquella que se efectúa sobre un tema u objeto desconocido o poco estudiado, por lo que sus resultados constituyen una visión aproximada de dicho objeto, es decir un nivel superficial de conocimiento. También el estudio puede ser dirigido a la formulación más precisa de un problema de investigación.

3.3. Definición de la población y selección de la muestra

Según Méndez (2001) el muestreo permite que el investigador seleccione la (muestra) de un grupo de datos (población) a fin de hacer algunas inferencias sobre el grupo total; para que una muestra sea útil debe reflejar las similitudes y diferencias encontradas en la totalidad del grupo. La muestra A fue cinco (5) estudiantes con discapacidad visual en la Universidad de Pamplona.

3.4. Instrumentos de recolección de información

La Encuesta. “Las encuestas consisten en un conjunto más o menos amplio de preguntas o cuestiones que se consideran relevantes para el rasgo, característica o variables que son objeto de estudio”. (Bisquerra, 1989, p. 88). Básicamente, podemos decir que los dos pilares sobre los cuales se construye la encuesta son el marco teórico y los objetivos de la investigación. De esta manera, al construir la encuesta se toman en cuenta una serie de aspectos, ideas o creencias del investigador ante el problema a estudiar atendiendo a la revisión bibliográfica y otras investigaciones sobre el mismo tema, realizadas en otros contextos, así como los objetivos planteados en el estudio.

3.5. Técnicas de procesamiento y análisis de datos

Arias (2004) señala que: Se describen las distintas operaciones a las que serán sometidos los datos que se obtengan: clasificación, registro, tabulación y codificación si fuera el caso”. Se recolectarán los datos mediante entrevistas estructurada, y encuestas; y analizados bajo una escala Licker con cinco opciones de respuesta de juicio valor, sobre cada uno de los indicadores en cada dimensión de la variable, actitud, uso de herramienta, interés del profesor, la información se tabulará y se realizará un análisis de carácter estadístico. Definidos los parámetros de valores de frecuencia y en forma de porcentajes (p. 99). Los datos recopilados se organizan y se analizan para cada ítem utilizando la estadística descriptiva para elaborar tablas de frecuencia que a su vez dieron lugar a las figuras que se presentan en el análisis de resultados. El análisis de datos cuantitativos en este tipo de operación se efectúa naturalmente, con toda la información numérica resultante de la investigación, la cual luego del procedimiento sufrido se presentó como un conjunto de cuadros, tablas, calculando sus porcentajes y otorgándoles forma definitiva.

Anexo A. Matriz de operacionalización de las variables

Tabla 1. Matriz de Operacionalización de las variables

OBJETIVO GENERAL: Desarrollar una aplicación móvil mediante el diseño de iconos hápticos para el aprendizaje en los estudiantes con discapacidad visual de la Universidad de Pamplona sede Villa Rosario

OBJETIVOS ESPECIFICOS	VARIABLE	DIMENSION	INDICADOR	ITEMS
Determinar herramientas Tecnológicas en el procesamiento de conversor de formatos digital audio para el proceso de aprendizaje en los estudiantes con discapacidad visual de la universidad de Pamplona	Actitud del estudiante de la Unipamplona con discapacidad visual en las herramientas Informáticas o TIC	Herramientas Tecnológicas en el procesamiento de conversor de formatos digital audio para el proceso de aprendizaje en los estudiantes con discapacidad visual de la universidad de Pamplona	# Entrevistas que permite dar un valor de juicios mediante la tabla de liker dentro del indicador de evaluación de interés de formación e interpretación de las TIC, en los estudiantes con discapacidad visual. 1.Considera ud. viable el desarrollo de una aplicación móvil que ayude en sus procesos de aprendizaje en la universidad? 2.para ud. es importante el uso de las TIC en sus procesos de aprendizaje?	1,2
Analizar las metodologías de desarrollo de aplicaciones móviles enfocadas en el diseño de sistemas utilizados por personas en condición de discapacidad visual.	Uso de herramientas en los estudiantes de la Unipamplona con discapacidad visual		# estudiantes estaría dispuesto a utilizar un teléfono móvil como herramienta de estudio? ¿De qué forma o que método utiliza para estudiar fuera de las aulas de clase?	4,5
	Estrategias pedagógicas para la educación incluyente en estudiantes con discapacidad visual en la Unipamplona		¿Qué actividad se le dificulta en su entorno académico? al escribir o tomar apuntes ¿Colocan a un tutor que les lea documentos de estudio para poderlos memorizar?	3, 6

Análisis de Resultados Entrevista

1. Actitud del estudiante de la Unipamplona con discapacidad visual en las herramientas Informáticas o TI.

Tabla 2. Actitud del estudiante de la Unipamplona con discapacidad visual en las herramientas Informáticas o TI

ITEMS	SI	NO
1. Considera ud. viable el desarrollo de una aplicación móvil que ayude en sus procesos de aprendizaje en la universidad?	5	0
2. ¿Para ud. es importante el uso de las TIC en sus procesos de aprendizaje?	5	0

Gráfica 1. Actitud del estudiante de la Unipamplona con discapacidad visual en las herramientas Informáticas o TI

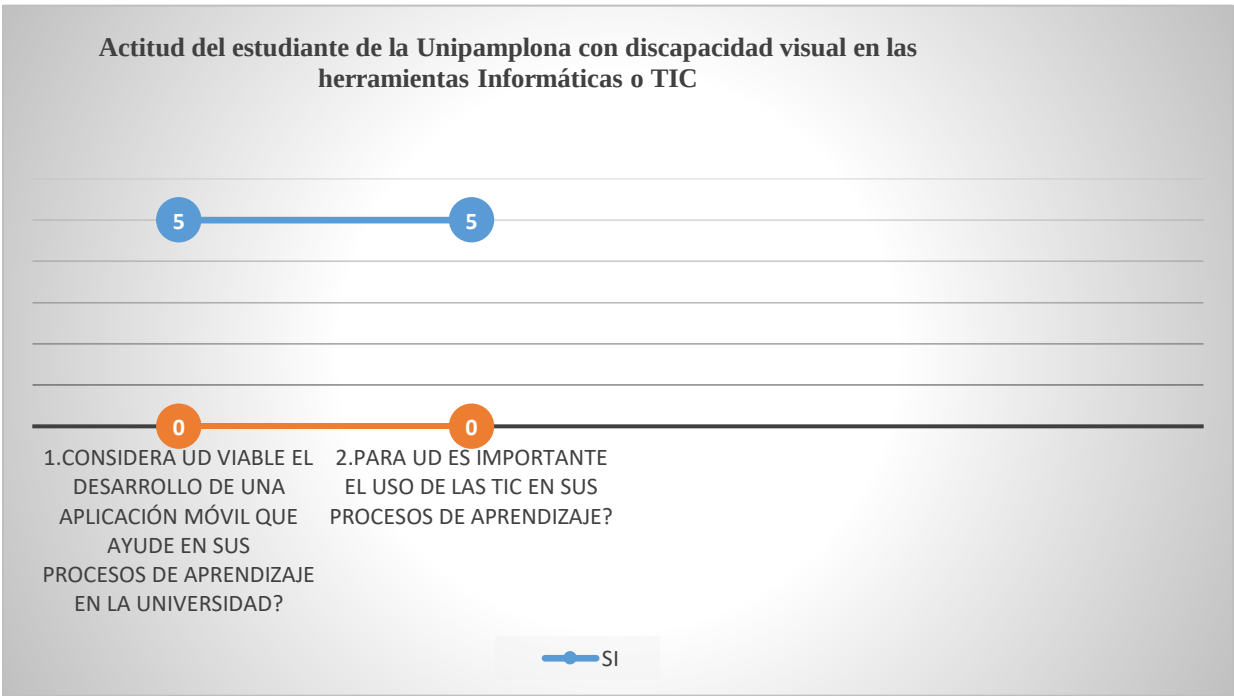


Ilustración 3. Actitud del estudiante de la Unipamplona con discapacidad visual en las herramientas Informáticas o TI

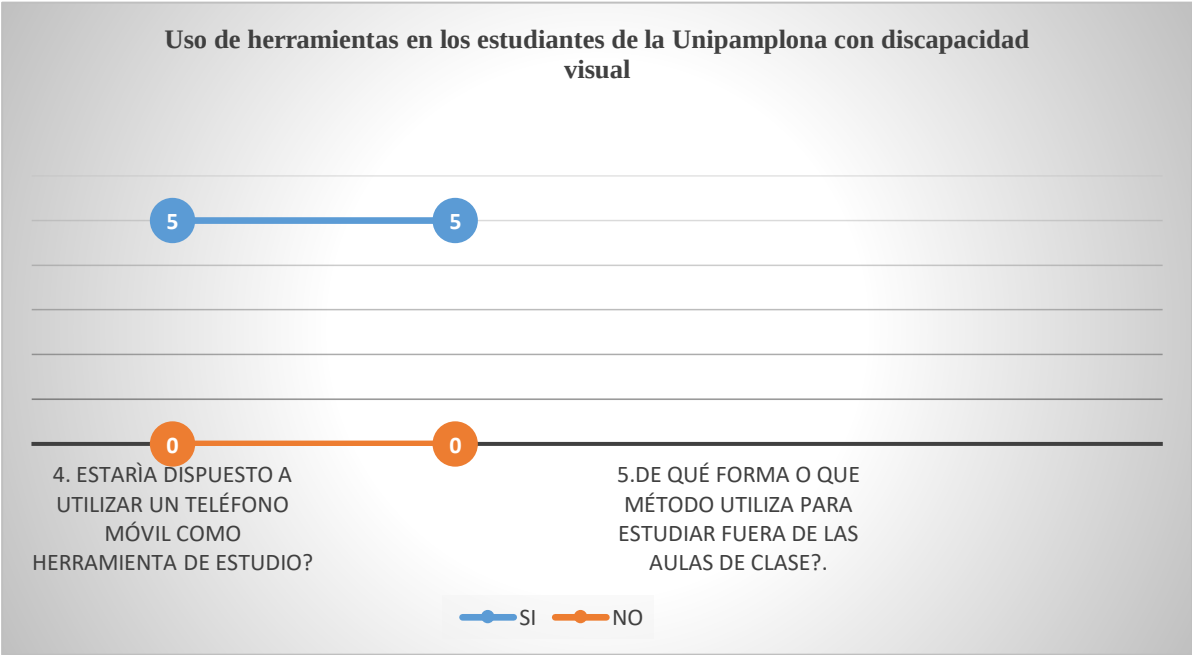
Un 100 % en los estudiantes con discapacidad visual considera que es importante utilizar las TIC en la aplicación móvil que ayuda en los procesos de aprendizaje y creen que si es viable que la implemente en la Unipamplona.

3. Uso de herramientas en los estudiantes de la Unipamplona con discapacidad visual

ITEMS	SI	NO
4. Estaría dispuesto a utilizar un teléfono móvil como herramienta de estudio?	5	0
5. utiliza un método para estudiar fuera de las aulas de clase?.	5	0

Tabla 3. Uso de herramientas en los estudiantes de la Unipamplona con discapacidad visual.

Gráfica 2. Uso de herramientas en los estudiantes de la universidad con discapacidad visual



Un

Ilustración 4. Uso de herramientas en los estudiantes de la universidad con discapacidad visual

100% en los estudiantes con discapacidad visual, utilizan herramientas de estudio como el brailer, grabadora periodista, computador para la investigación y el software JOW; por lo tanto, se resalta en 4 estudiantes que utilizan teléfono móvil. Como consecuencia de mayor relevancia para la muestra estudiada es necesario utilizar las TIC y el móvil como herramienta de apoyo en el aprendizaje, en la interacción con la comunidad universitaria. En cuanto a la seguridad de huella, expresan que no tienen ya que es un móvil de alta gama.

4. Estrategias pedagógicas para la educación incluyente en estudiantes con discapacidad visual en la Unipamplona

Tabla 4. Estrategias pedagógicas para la educación incluyente en estudiantes con discapacidad visual en la Unipamplona

Tabla 4. Estrategias pedagógicas para la educación incluyente en estudiantes con discapacidad visual en la Unipamplona

ITEMS	SI	NO
3. ¿Se le dificulta en su entorno académico? al escribir o tomar apuntes	5	0
6. Colocan a un tutor que les lea documentos de estudio para poderlos memorizar?.	0	5

Gráfica 3. Estrategias pedagógicas para la educación incluyente en estudiantes con discapacidad visual en la Unipamplona

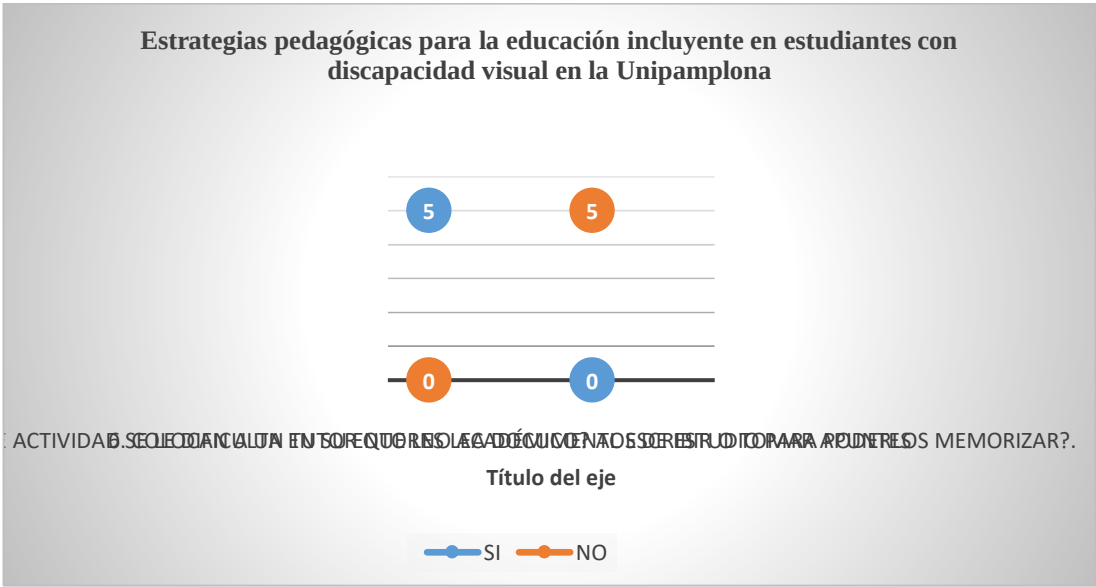


Ilustración 5. Estrategias pedagógicas para la educación incluyente en estudiantes con discapacidad visual en la Unipamplona

Un 100% en los estudiantes se le dificulta en tomar apuntes, es bueno que se implemente un libro táctil para las notificaciones en la clase, teniendo una vibración específica para facilitar la información en la clase, la comunicación rápida. En los estudiantes con discapacidad visual es primordial que se incluya la población sorda – ciego ya que ellos se comunican con la escritura en la mano, siendo costumbre el desarrollo del tacto. En cuanto a un tutor para los estudiantes con discapacidad visual, no se cuenta con este apoyo educativo.

CAPÍTULO 4. ANÁLISIS Y DISEÑO DEL SISTEMA

4.1 ANÁLISIS DEL SISTEMA

4.1.1. Requerimientos Funcionales

4.1.1.1. Tabla Descriptiva del requerimiento funcional - Registro usuarios

Tabla 5. Requerimiento Funcional 1 - Registro usuarios

Identificación del requerimiento	RF01
Nombre del Requerimiento	Registro de Usuarios
Características	El sistema permitirá el registro de usuarios.
Descripción del requerimiento	Permite el registro de datos de los nuevos usuarios como un id, nombre, apellido, y datos personales.
Prioridad del requerimiento	Alta

Fuente: Elaboración propia

4.1.1.2. Tabla Descriptiva del requerimiento funcional - Iniciar sesión

Tabla 6. Requerimiento Funcional 2 - Iniciar sesión

Identificación del requerimiento	RF02
Nombre del Requerimiento	Iniciar Sesión
Características	Los usuarios deberán registrarse permitirles el acceso a la aplicación que está protegida por un usuario y contraseña.
Descripción del requerimiento	El sistema podrá ser usado por cualquier usuario que este registrado.
Prioridad del requerimiento	Alta

Fuente: Elaboración propia

4.1.1.3. Tabla Descriptiva del requerimiento funcional - Reconocimiento texto a voz

Tabla 7. Requerimiento Funcional 3 - Reconocimiento texto a voz

Identificación del requerimiento:	RF03
Nombre del Requerimiento:	Reconocimiento de texto a voz
Características:	El sistema permite reconocimiento de texto en documentos, procesando la información contenida y emitiendo una señal de voz reconocida por el usuario.
Descripción del requerimiento:	El sistema convertirá en voz el texto de los documentos usados en las clases para los estudiantes con discapacidad visual.
Prioridad del requerimiento:	Alta

Fuente: Elaboración propia

4.1.1.4. Tabla Descriptiva del requerimiento funcional - Interpretación de iconos hapticos

Tabla 8. Requerimiento Funcional 4- Interpretación de iconos hapticos

Identificación del requerimiento:	RF04
Nombre del Requerimiento:	Interpretación de iconos hápticos
Características:	El sistema permite una herramienta en el dispositivo para emitir vibraciones y poder ser interpretadas por el usuario.
Descripción del requerimiento:	Los usuarios registrados en el sistema podrán identificar las diversas vibraciones por medio del uso de la aplicación.
Prioridad del requerimiento	Alta

Fuente: Elaboración propia

4.1.1.5. Tabla Descriptiva del requerimiento funcional - Escucha del texto procesado

Tabla 9. Requerimiento Funcional 5 -Escucha del texto procesado

Identificación del requerimiento:	RF05
Nombre del Requerimiento:	Escucha del Texto Procesado
Características:	El sistema permite al estudiante oír el texto en su dispositivo móvil por medio del proceso de síntesis de voz.
Descripción del requerimiento:	Los usuarios registrados en el sistema podrán escuchar el texto procesado de sus documentos.
Prioridad del requerimiento	Alta

Fuente: Elaboración propia

4.1.1.5. Tabla Descriptiva del requerimiento funcional - Autenticación biométrica

Tabla 10. Requerimiento Funcional 6 -Autenticación biométrica

Identificación del requerimiento:	RF06
Nombre del Requerimiento:	Autenticación Biométrica
Características:	El sistema permite acceder mediante autenticación biométrica
Descripción del requerimiento:	Los usuarios podrán ingresar al sistema, a través de su huella dactilar registrada en el dispositivo.
Prioridad del requerimiento	Alta

Fuente: Elaboración propia

4.1.2. Requerimientos No Funcionales

4.1.2.1. Tabla Descriptiva del requerimiento no funcional - Interfaz sistema

Tabla 11. Requerimiento No Funcional - Interfaz sistema

Identificación del requerimiento:	RNF01
Nombre del Requerimiento:	Interfaz del sistema.
Características:	El sistema presentará una interfaz de usuario sencilla para que sea de fácil manejo a los usuarios del sistema.
Descripción del requerimiento:	El sistema debe tener una interfaz de uso intuitiva y sencilla
Prioridad del requerimiento	Alta

Fuente: Elaboración propia

4.1.2.2. Tabla Descriptiva del requerimiento no funcional - Disponibilidad

Tabla 12. Requerimiento No Funcional - Disponibilidad

Identificación del requerimiento:	RNF02
Nombre del Requerimiento:	Disponibilidad
Características:	El sistema tendrá que estar en funcionamiento las 24 horas de los 7 días de la semana. Ya que es una aplicación móvil diseñada para el uso continuo de sus usuarios.
Descripción del requerimiento:	La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 7 días por 24 horas.
Prioridad del requerimiento	Alta

Fuente: Elaboración propia

4.1.2.3. Tabla Descriptiva del requerimiento no funcional - Seguridad de la información

Tabla 13. Requerimiento No Funcional - Seguridad de la información

Identificación del requerimiento:	RNF03
Nombre del Requerimiento:	Seguridad en la información
Características:	El sistema garantizará a los usuarios seguridad en cuanto a la información que se procede en el sistema.
Descripción del requerimiento:	Garantizar la seguridad del sistema con respecto a la información y datos que se tratan como por ejemplo contraseñas.
Prioridad del requerimiento	Alta

Fuente: Elaboración propia

4.1.3. Requerimientos de interfaz

4.1.3.1. Interfaces de usuario.

La interfaz con el usuario consistirá en un conjunto de Activitys con botones y campos de texto. Esta deberá ser construida específicamente para el sistema propuesto y, será visualizada desde un dispositivo móvil.

4.1.3.2. Interfaces de Hardware.

Disponer de un dispositivo móvil Smartphone con sistema Operativo Android.

Servicios: 1) Internet (WIFI o Datos móviles),

2) Permiso de Internet.

3) Permiso de Acceso a memoria del dispositivo.

4.1.3.3. Interfaces de Software.

Se utilizó en sistema operativo Android igual o superior a la versión 6.0 (Marshmallow). Eso debido a que se requiere del uso de los diferentes compones que esa versión nos ofrece, entre los más destacados está el soporte a los sensores de huellas dactilares que se necesitan ejecutar en el desarrollo de la aplicación móvil para poder dar acceso al usuario por autenticación biométrica, Además de ser compatible con la gran mayoría de marcas y modelos de dispositivos existentes en el mercado como Samsung, Huawei, Lg, entre otros.

Por lo tanto, este nuevo sistema que intentará minimizar los wakelocks cuando el dispositivo no se está usando de forma activa porque por ejemplo estemos durmiendo. Las aplicaciones aumentarán la frecuencia de sincronización cuando el dispositivo reconoce que estamos haciendo un uso intenso. También «App Standby» no habrá más consumo de batería de aquellas aplicaciones que rara vez utilizamos.

4.2. DISEÑO DEL SISTEMA

4.2.1. Diagramas de Casos de Uso

Los diagramas de caso de uso son de tipo UML (lenguaje de modelado unificado) de comportamiento y se usa frecuentemente para analizar diversos sistemas. Adicionalmente Permiten que el analista se centre en las necesidades del usuario quien espera lograr utilizar el sistema lo que facilita la priorización del requisito.

A continuación se presentan los diagramas de uso general y especifico de la aplicación VisionAccess desarrollada.

4.2.1.1. Caso de Uso General

El caso de uso general nos permite ver los procesos que se realizan de forma global en todo el sistema de forma ilustrada. (Ver ilustración 6)

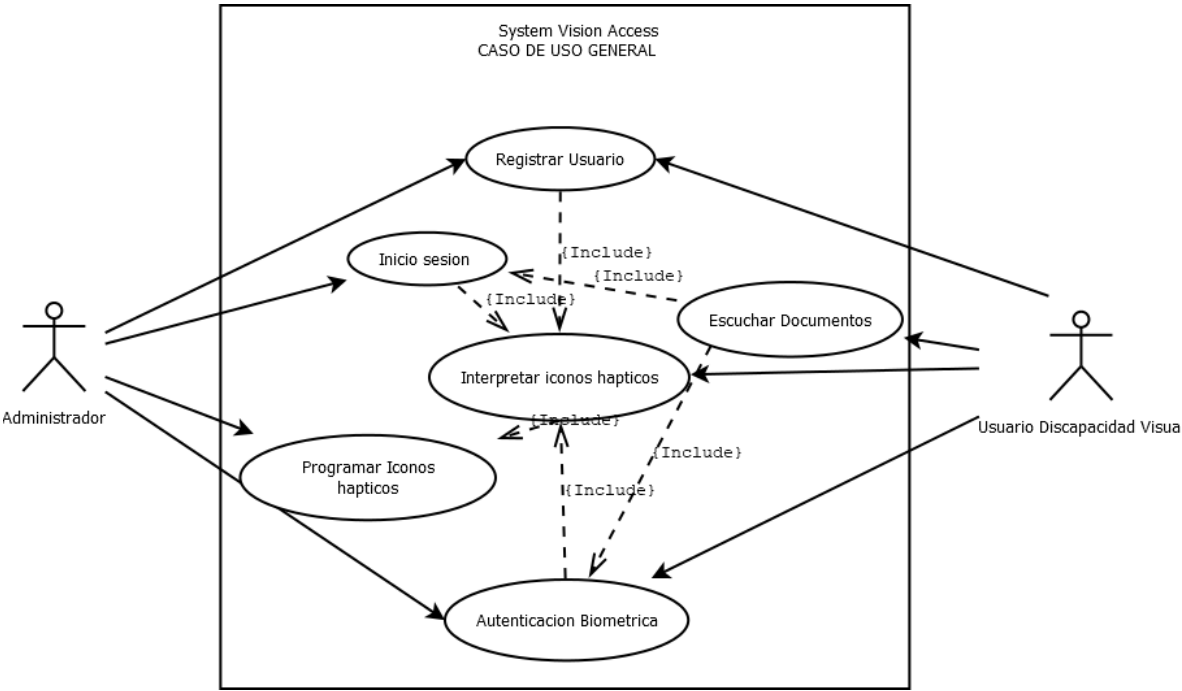


Ilustración 6. Caso de Uso General
Fuente: Propia

Tabla 14. Caso de uso general

CÓDIGO	CASO DE USO
CU1	Iniciar Sesión
CU2	Registrar Usuario
CU3	Programar iconos hápticos
CU4	Escuchar Texto Sintetizado
CU5	Interpretar iconos hápticos

Fuente: Elaboración propia

4.2.1.2. Caso de Uso Específico

El caso de uso específico nos proporciona detalles textuales que permiten comprender mejor los procesos internos que realiza el usuario al interactuar con el sistema. (Ver Ilustración 7.)

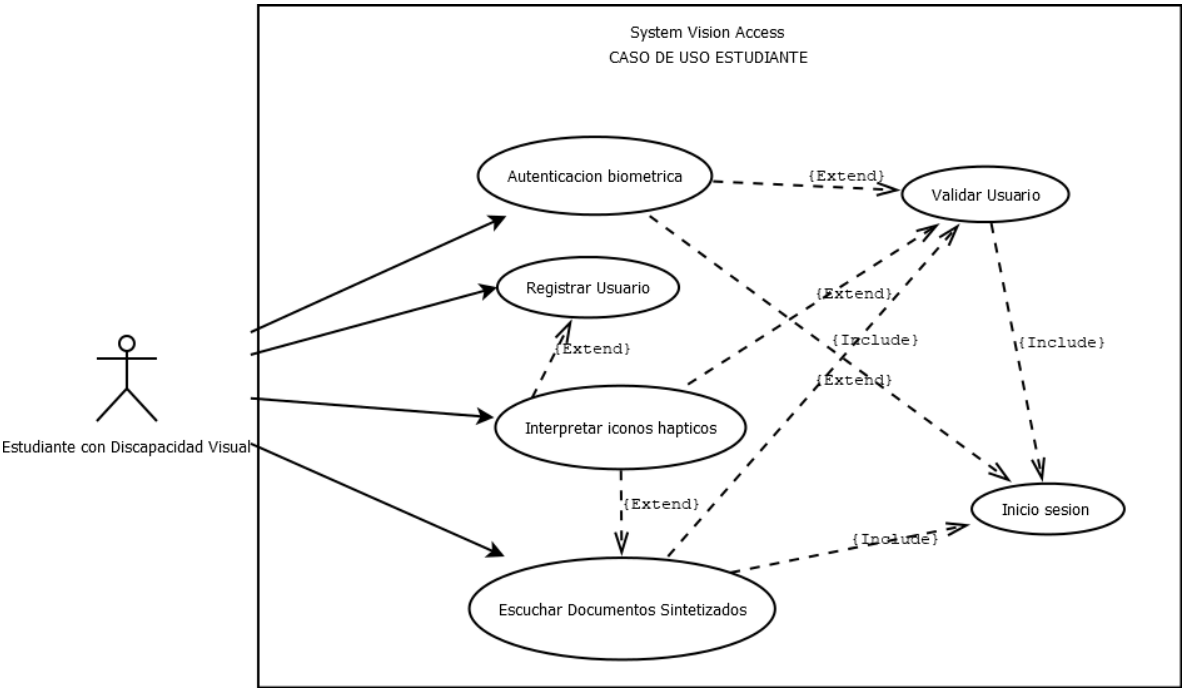


Ilustración 7. Caso de Uso Específico Fuente: Elaboración propia

4.2.1.2.1. Descripción del Diagrama de Caso de Uso Estudiante

A continuación se presentan mediante una tabla descriptiva los diversos componentes que forman la estructura de caso de uso en el estudiante que va a utilizar la aplicación. (Ver tabla 15 a la tabla 19).

Tabla 15. Diagrama de Caso de Uso Estudiante

CÓDIGO	CASO DE USO
CU1	Registrar usuario Iniciar Sesión
CU2	Autenticación biométrica
CU3	Programar iconos hápticos
CU4	Escuchar Texto Sintetizado
CU5	Interpretar iconos hápticos

Fuente: Elaboración propia

4.2.1.2.2. Tabla Registrar usuario

Tabla 16 Caso de uso - Registrar usuario

CASO DE USO – CU1	
Nombre : Registrar Usuario	
Descripción	El sistema permite crear nuevos usuarios para que puedan acceder a la aplicación.
Requerimiento	RF01
Precondición	El id del usuario solo se podrá registrar una sola vez.
Flujo Normal	
El usuario ingresa a la aplicación y selecciona el botón “Registrar usuario”, ingresa los datos requeridos en el formulario como (usuario, contraseña) y por último selecciona el botón de registrarse.	
Actor	Sistema
1. El usuario ingresa los datos en el formulario de registro.	3. El sistema verifica los datos.
2. Selecciona el botón “registrarse”.	4. Hace el registro del nuevo usuario.
	5. muestra un mensaje de usuario registrado.

Flujo Alterno

Si al momento de verificar si los datos con el correo del usuario ya existen, el sistema muestra un mensaje de “Usuario ya registrado”.

Postcondición Muestra un mensaje de “Usuario Registrado”.

Fuente: Elaboración propia

4.2.1.2.3. Tabla de caso de uso - Autenticación biométrica

Tabla 17. Caso de uso - Autenticación biométrica

CASO DE USO – CU2

Nombre : Autenticación Biométrica

Descripción El sistema permite al usuario autenticar su huella dactilar

Requerimiento RF06

Precondición Haber registrado la huella en el dispositivo .

Flujo Normal

El usuario toca el sensor de huella para ser autenticado.

Actor

1. El usuario estudiante
Selecciona el botón Iniciar con huella.
2. Desliza su huella dactilar sobre el sensor del móvil

Sistema

3. Se visualiza la respectiva interfaz de acceso
4. Se validan los métodos de autenticación biométrica para dar acceso al usuario.

Flujo Alterno

Si el usuario no toca el sensor de huella del móvil no podrá ser autenticado para acceder al sistema.

Postcondición Visualizar los respectivos documentos disponibles.

Fuente: Elaboración propia

4.2.1.2.4. Tabla de Caso de uso - iniciar sesión

Tabla 18. Caso de uso - iniciar sesión

CASO DE USO – CU3	
Nombre : Iniciar Sesión	
Descripción	El sistema valida si el usuario, contraseña son correctos para autorizar el uso de la funcionalidad en el aplicación.
Requerimiento	RF01
Precondición	El usuario que va a iniciar sesión debe estar registrado en el sistema. O en caso de huella estar registrada en el móvil.
Flujo Normal	
El usuario digita el nombre de “usuario y contraseña”, se validan los datos y si estos son correctos se les autoriza el uso de la funcionalidad Leer documentos.	
Actor	Sistema
1. El usuario digita el “usuario” y la “contraseña”	3. El sistema valido que los datos ingresados sean correctos para mostrar la interfaz correspondiente.
2. Selecciona el botón “Iniciar Sesión”.	
Flujo Alterno	
Si el usuario y contraseña no coinciden con los ya registrados en el sistema se mostrará un mensaje de “usuario o contraseña incorrectos”.	
Postcondición	Muestra la interfaz principal.

Fuente: Elaboración propia

4.2.1.2.5. Tabla de Caso de uso - Escuchar texto sintetizado

Tabla 19. Caso de uso - Escuchar texto sintetizado

CASO DE USO – CU4

Nombre : Escuchar texto sintetizado

Descripción

El sistema permite al usuario escuchar el texto sintetizado; es decir la traducción de texto a voz.

Requerimiento

RF05

Precondición

El usuario puede escuchar el texto sintetizado de formato pdf cuyo contenido sean caracteres tipo texto .

Flujo Normal

El usuario escucha la voz del texto que va apareciendo del respectivo documento seleccionado, para una mejor comprensión de los temas a estudiar.

Actor

1. El usuario escucha en la pantalla del móvil el texto contenido en el documento que se produce de la sintetización de este mismo.

Sistema

2. El sistema lee el texto contenido en el documento.

Flujo Alterno

Si el usuario no selecciona ningún documento no va haber síntesis de texto a voz por lo cual el estudiante tampoco podrá escuchar ningún texto.

Postcondición

Ayuda en la comprensión de los temas de las clases y mejora la organización de estudio.

Fuente: Elaboración propia

4.2.1.2.6. Tabla de caso de uso - Interpretar iconos hápticos

CASO DE USO – CU5	
Nombre : Interpretar iconos hápticos	
Descripción	El sistema permite al usuario interpretar las vibraciones diferenciales emitidas de cada icono cuando esté utilizando la aplicación.
Requerimiento	RF04
Precondición	El usuario debe ser capacitado para identificar las diversas vibraciones emitidas en el sistema.
Flujo Normal	
El usuario debe realizar una acción en la aplicación como oprimir un botón para que se inicie el proceso de vibración.	
Actor	Sistema
1. El usuario realiza una acción en la aplicación	6. El sistema identifica la acción del usuario
2. tocar un botón	7. El sistema emite un patrón de vibración específico dependiendo de la acción que realice el usuario.
3. tocar sensor de huella	
4. tocar cuadro de texto	
5. escribir sobre el cuadro de texto	
Flujo Alterno	
Si el usuario no realiza ninguna acción en la aplicación no se activará el sistema de vibración en sus iconos.	
Postcondición	Permite al usuario percibir de forma sensorial la navegación y utilización de la aplicación.

Fuente: Elaboración propia

4.2.2. Diagramas de Secuencia.

Los diagramas de secuencia son una solución de modelado en UML porque se centran específicamente en líneas de vida o en los procesos y objetos que coexisten simultáneamente, y los mensajes intercambiados entre ellos para ejecutar una función antes de que la línea de vida termine.

Diagrama de Secuencia Estudiante con Discapacidad Visual

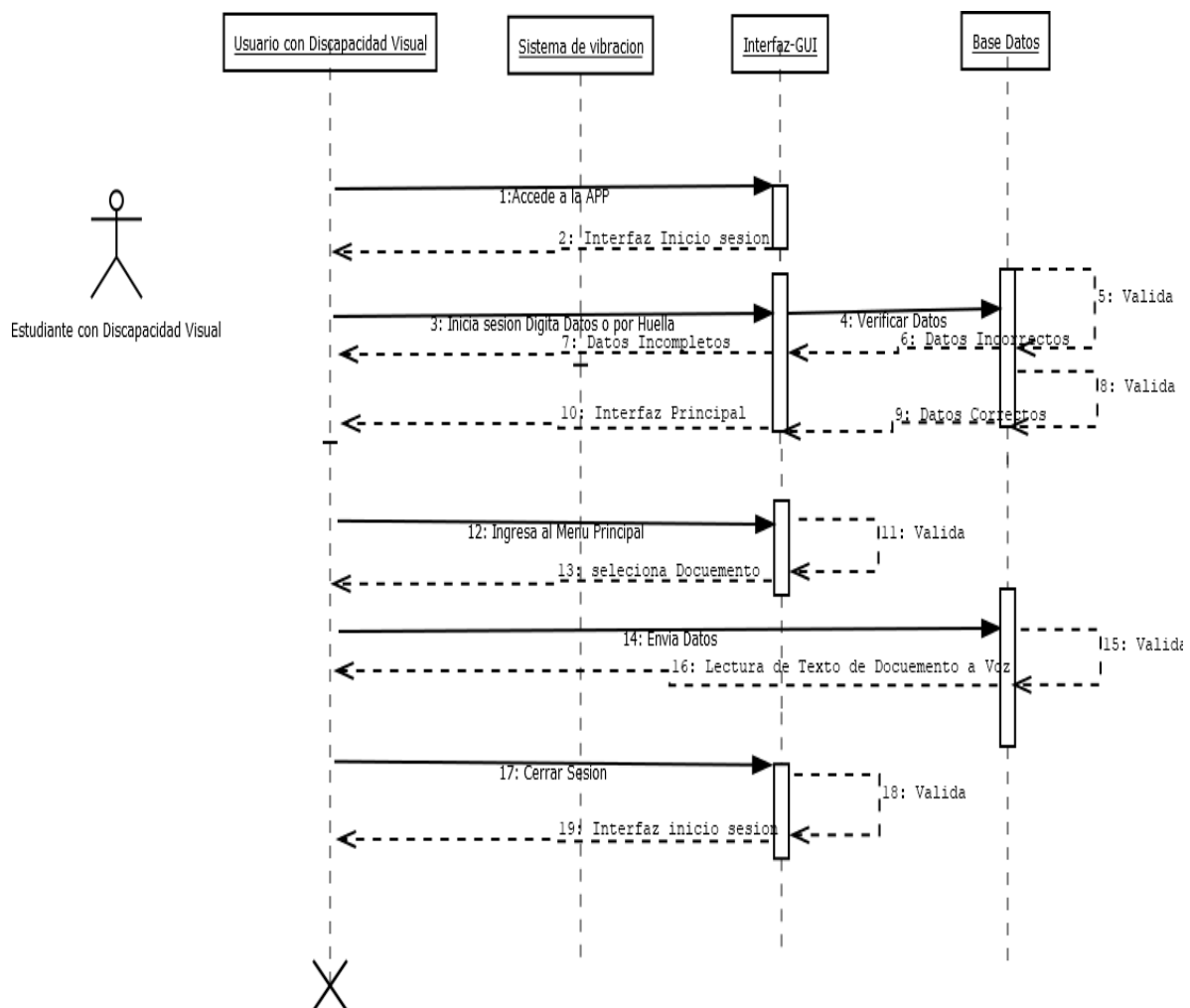


Ilustración 8. Diagrama de Secuencia Estudiante con Discapacidad Visual

Fuente: Propia

4.3 Diseño Base Datos

Para la implementación de la base de datos se trabaja con Firebase Database, la cual es una base de datos en tiempo real alojada en la nube, por lo cual se puede contar con alta disponibilidad; además de esto brinda una interfaz de fácil comprensión y relativamente sencilla la cual se adapta de manera eficaz a la aplicación. Además la base de datos no relacional utiliza una estructura diferente a la base de datos relacionales, esta estructura está basada en formato de tipo JSON.

Por lo tanto, como todos los datos se almacenan en formato JSON, es por esto que la estructura de la base puede verse como un árbol JSON. También el siguiente árbol muestra la estructura de los datos con los cuales trabaja la aplicación.

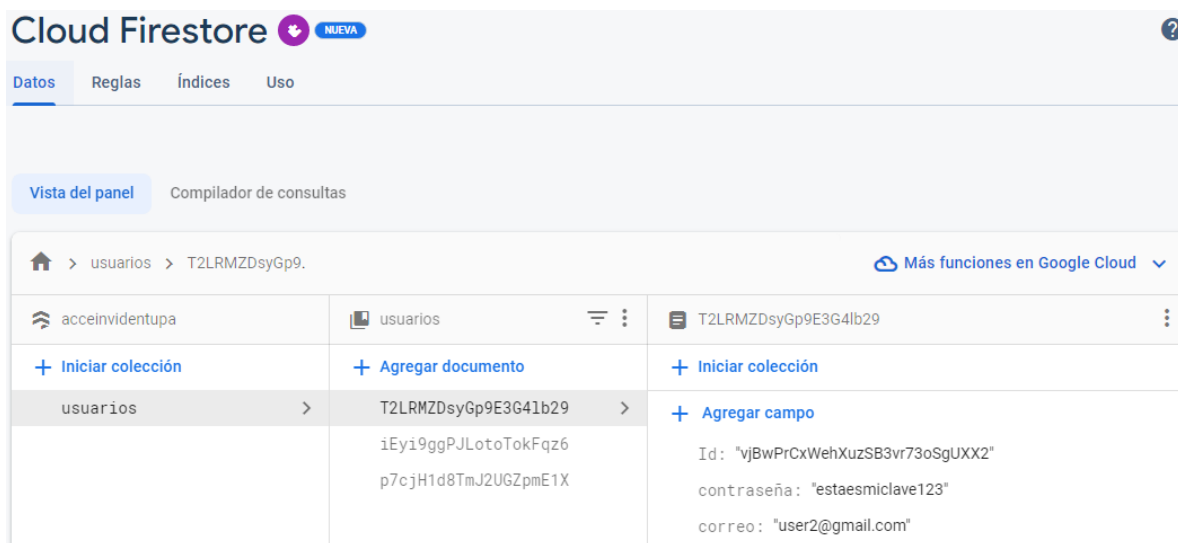


Ilustración 9. Estructura tipo JSON de los datos de la aplicación

Fuente: Propia

en este caso como se puede observar en la ilustración, el árbol se basa en el nodo “Usuarios” que es donde se almacenarán todos los datos de cada usuario, cada vez que un nuevo usuario se registra en la base de datos se le asocia un identificador único y se registra en dicho nodo.

Como se puede observar los datos almacenan un tipo de valor específico, sin embargo, en este tipo de bases de datos no relacionales estos tipos no se especifican dentro de la estructura sino en la plataforma de programación (Android Studio) ya que pueden ser cualquier tipo nativo que corresponda a los tipos de JSON disponibles.

Para la configuración de la base de datos dentro de la aplicación móvil es necesario incluir dentro de las dependencias de la aplicación Android las siguientes líneas de código:



Ilustración 10. Dependencias de FireBase

Fuente: Propia

Esto permite la importación de las funciones de Firebase dentro del proyecto de Android a través de las dependencias y sirve de puente para la conexión con Android.

Es importante que la comunicación sea exitosa para ello debemos añadir la aplicación móvil a la base de datos por medio de la consola de Firebase, de la siguiente forma:



Ilustración 11. Añadiendo nuevo proyecto a Firebase

Fuente: Propia

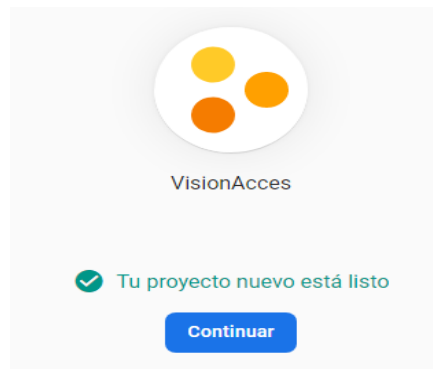


Ilustración 12. Notificación de proyecto agregado a Firebase

Fuente: Propia

A continuación se establecen las reglas de seguridad por defecto en cloud firestore donde se elige si se denegará o permitirán las operaciones de lectura y escritura de terceros en la aplicación.



Ilustración 13. Reglas de Seguridad de la Base de datos en Firebase

Fuente: Propia

En nuestro caso habilitamos con true el permiso para la escritura (.write) y lectura (.read) de los usuarios que van a utilizar de nuestra aplicación.

Luego se descarga un archivo de configuración generado por Firebase que se añade al proyecto en Android. Este archivo es de tipo JSON, el cual contiene la información necesaria del proyecto para

poder realizar la conexión adecuada con la aplicación registrada anteriormente por medio de la consola.

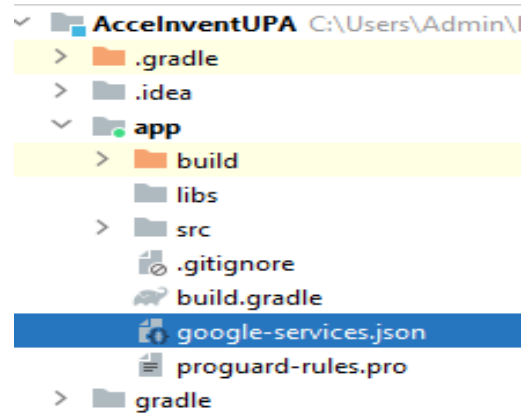


Ilustración 14. Ubicación del Archivo JSON dentro del proyecto en Android

Fuente: Propia

En el entorno de la plataforma android studio se hace uso de la referencia FirebaseFirestore mediante el comando siguiente:

```
private FirebaseFirestore mFirestore; |  
  
private FirebaseFirestore dbFirestore ;
```

Ilustración 15. Comando FirebaseFirestore

Fuente: Propia

Después de inicializado se puede usar para guardar la información que se desee en la base de datos de firestore (dbFirestore) , en este caso en el nodo usuarios. También se utiliza el parámetro DocumentReference que hace referencia a la ubicación de un documento en una base de datos de Firestore y se puede usar para escribir o leer la ubicación. El documento en la ubicación a la que se hace referencia puede existir o no, ya que si no existe directamente lo crea.

```
dbFirestore.collection(collectionPath: "usuarios").  
    add(map).addOnSuccessListener(new OnSuccessListener<DocumentReference>() {  
        @Override  
        public void onSuccess(DocumentReference documentReference) {  
            Log.d(LOG_TAG, msg: "USUARIO AGREGADO CON ID: " + documentReference.getId());  
        }  
    })
```

Ilustración 16. Código CollectionReference

Fuente: Propia

Una DocumentReference también se puede usar para crear una CollectionReference para una subcolección.

CAPÍTULO 5. DESARROLLO DEL PROYECTO

5.1. Determinar herramientas Tecnológicas en el procesamiento de conversor de formatos digital a audio para el proceso de aprendizaje en los estudiantes con discapacidad visual de la Universidad de Pamplona.

5.1.1. Herramientas Tecnológicas para dispositivos (App)

A continuación se presenta un esquema generalizado que permite identificar las características más relevantes en cuanto a herramientas de desarrollo.

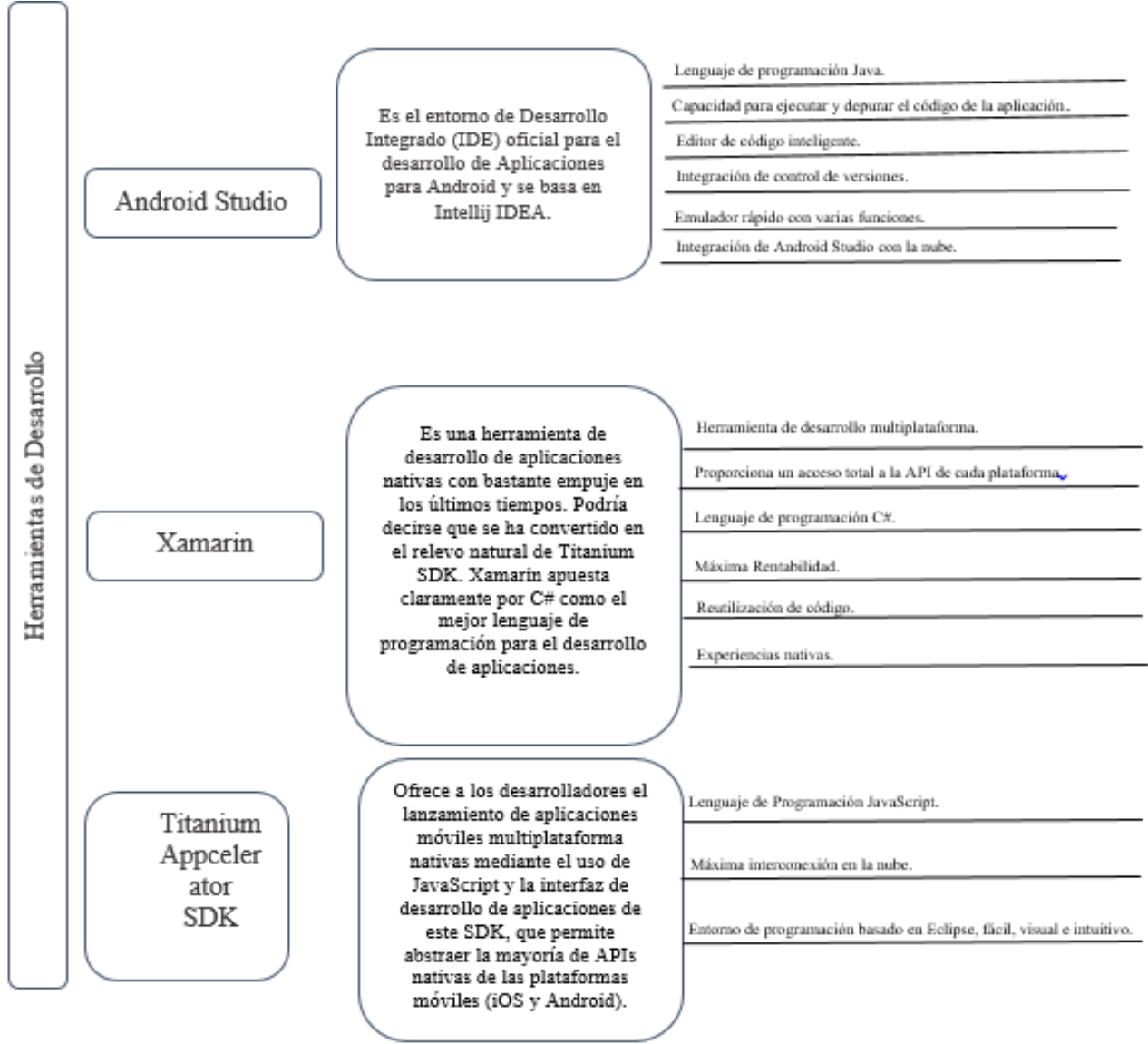


Ilustración 17. Herramientas de desarrollo
Fuente: Propia

Por las características anteriormente descritas se eligió Android Studio como el entorno de desarrollo de la aplicación.

5.1.1.1. Android Studio

Android Studio es el IDE oficial de Android. Este IDE incorpora herramientas especiales para editar, depurar y probar código de desarrollo de aplicaciones Android. Entre sus características destacan (Android Developers): (a). Instant Run: Cuando hagas clic en Run o Debug, la función Instant Run de Android Studio aplicará los cambios en el código y los recursos en tu aplicación en ejecución. Esta interpreta de manera inteligente los cambios sin compilar de nuevo la app. (b). Editor de código inteligente: El editor de código inteligente te permite escribir un código más eficaz, trabajar más rápido y ser más productivo. A medida que escribes, Android Studio proporciona sugerencias en una lista desplegable. (c). Emulador rápido y cargado de funciones: Android Emulador se instala e inicia tus apps más rápido que un dispositivo real. También te permite crear prototipos de tu app y probarlos en todas las configuraciones de dispositivos Android: teléfonos, tablets y dispositivos Android Wear y Android TV. Por lo tanto; las tres características aquí descritas son muy importantes a la hora de agilizar el desarrollo de la aplicación, especialmente el uso del editor inteligente para disponer de la documentación necesaria a medida que se está programando.

5.1.1.2. Firebase

5.1.1.3. Software de Sintetización de Texto a Voz

El sistema de Conversión Texto a Voz de Código Abierto Para Lenguas Ibéricas. Un sistema de conversión texto a voz (CTV) es un sistema que convierte una entrada de texto en una salida en forma de señal de audio cuyo contenido se corresponde con el mensaje del texto de entrada. Los sistemas CTV actuales pueden descomponerse en dos módulos. El primero, fuertemente dependiente del idioma, toma como entrada el texto a sintetizar y genera a su salida etiquetas que describen lingüística, fonética y prosódicamente dicha entrada. Estas etiquetas alimentan el segundo módulo, el motor de síntesis que en sí mismo es independiente del idioma. Este módulo emplea voces

entrenadas a partir de bases de datos que son dependientes del idioma para sintetizar la señal de voz de salida en función de las etiquetas de entrada.

5.1.1.4. Google Text-to-Speech

Servicios de voz de Google es una aplicación oficial de Google que nos permitirá hacer que otras aplicaciones de nuestro terminal Android 'nos hablen', dictando todo el texto que aparezca en pantalla. Además; es importante tener en cuenta que Servicios de voz de Google no es compatible con todas las aplicaciones de Android, sino tan sólo con unas pocas. Entre las más destacadas encontramos Google Translator que nos dará la posibilidad de escuchar las traducciones. Por consiguiente; para activar el motor de síntesis de voz de Google es necesario seguir el siguiente procedimiento: acceder a los ajustes del sistema, seleccionar 'idioma y entrada de texto', y por último elegir 'Síntesis de voz'. Ahí tendremos que seleccionar el motor de síntesis de voz de Google como motor de preferencia.

5.1.1.5. TextToSpeech en Android

Esta herramienta se encarga de sintetizar el habla a partir del texto para su reproducción inmediata una instancia de TextToSpeech solo se puede usar para sintetizar texto una vez que haya completado su inicialización. Implementando el atributo TextToSpeech.OnInitListener para ser notificado de la finalización de la inicialización. Cuando se haya terminado de usar la instancia de TextToSpeech, se hace un llamado al shutdown() ,método para liberar los recursos nativos que usa el motor de TextToSpeech. Por consiguiente; entre las constantes más comunes que utiliza esta herramienta tenemos las que se presentan a continuación:

Tabla 20. Constantes más comunes que utiliza esta herramienta TextToSpeech

Constantes	Descripción
ACTION_TTS_QUEUE_PROCESSING_COMPLETED	Acción de transmisión: el sintetizador TextToSpeech ha completado el procesamiento de todo el texto en la cola de voz.
ERROR	Denota una falla de operación genérica.
LANG_NOT_SUPPORTED	Indica que el idioma no es compatible.
QUEUE_ADD	Modo de cola donde la nueva entrada se agrega al final de la cola de reproducción.
ERROR_OUTPUT	Indica una falla relacionada con la salida (dispositivo de audio o un archivo)
QUEUE_FLUSH	Modo de cola en el que todas las entradas de la cola de reproducción (medios que se reproducirán y texto que se sintetizará) se eliminan y reemplazan por la nueva entrada
STOPPED	Indica una parada solicitada por un cliente.
SUCCESS	Denota una operación exitosa.
ERROR_NETWORK	Indica una falla causada por problemas de conectividad de red.
ERROR_SYNTHESIS	Indica una falla de un motor TTS para sintetizar la entrada dada.
LANG_MISSING_DATA	Indica que faltan datos de idioma.

Fuente: Propia

La herramienta TextToSpeech también cuenta con métodos nativos muy útiles entre los cuales se encuentran:

Tabla 21. Métodos más comunes que utiliza esta herramienta TextToSpeech

Método	Descripción
shutdown()	Libera los recursos utilizados por el motor TextToSpeech.
getVoices()	Consulta al motor sobre el conjunto de voces disponibles.
isSpeaking()	Comprueba si el motor TTS está ocupado hablando.
setLanguage(Locale loc)	Establece el idioma de texto a voz.
stop()	Interrumpe la expresión actual (ya sea reproducida o procesada en un archivo) y descarta otras expresiones en la cola.
getMaxSpeechInputLength()	Límite de longitud de la cadena de entrada pasada para hablar y sintetizar en archivo.
getDefaultEngine()	Obtiene el nombre del paquete del motor de síntesis de voz predeterminado.

Fuente: Propia

5.1.1.6. Herramienta de Autenticación Biométrica en Android

Un método para proteger la información además de prestar facilidad y agilidad en la App es mediante el uso de autenticación biométrica. Una clase que proporciona información del sistema relacionada con la biometría es *BiometricManager* que es la encargada de contener funciones y atributos a utilizar en nuestra App para el correcto proceso de autenticación biométrica. También contamos con la clase *BiometricPrompt* que administra un diálogo biométrico proporcionado por el sistema para interactuar con el usuario. Entre los métodos más comunes tenemos:

Tabla 22. Descripción métodos Autenticación biométrica

Método	Descripción
AUTHENTICATION_RESULT_TYPE_BIOMETRIC	Se da cuando el usuario se autenticó al presentar algún tipo de biometría (por ejemplo, huella digital).
BIOMETRIC_ACQUIRED_IMAGER_DIRTY	La imagen biométrica tenía demasiado ruido debido a que se sospechaba o se detectaba suciedad en el sensor.
BIOMETRIC_ACQUIRED_PARTIAL	Solo se detectó una imagen biométrica parcial. Durante el registro, se debe informar al usuario sobre lo que debe suceder para resolver este problema, por ejemplo, "presione firmemente el sensor". (para huella digital)
BIOMETRIC_ACQUIRED_TOO_FAST	La imagen biométrica estaba incompleta debido al movimiento rápido. Por ejemplo, esto también podría suceder si el usuario se movió durante la adquisición. Se debe pedir al usuario que repita la operación más lentamente.
BIOMETRIC_ERROR_HW_NOT_PRESENT	El dispositivo no tiene un sensor biométrico.
BIOMETRIC_ERROR_LOCKOUT	La operación se canceló porque la API está bloqueada debido a demasiados intentos. Esto ocurre después de 5 intentos fallidos y dura 30 segundos.
BIOMETRIC_ERROR_NO_BIOMETRICS	El usuario no tiene datos biométricos registrados.

Fuente: Propia

5.1.1.7. Herramienta de vibración en Android

Android cuenta con una herramienta nativa llamada *vibrator* y la clase *VibratorManager* que administra funciones que permiten emitir vibración en el dispositivo móvil. Es importante recalcar que si su proceso sale, cualquier vibración que haya iniciado en el dispositivo se detendrá.

A continuación se describen los métodos que esta herramienta de Android administra:

Tabla 23. Métodos principales de herramienta de vibratorManager

Método	Descripción
<code>abstract void cancel()</code>	Apague todos los vibradores.
<code>abstract Vibrator getDefaultVibrator()</code>	Devuelve el vibrador predeterminado para el dispositivo.
<code>abstract Vibrator getVibrator(int vibratorId)</code>	Recuperar un solo vibrador por id.
<code>abstract int[] getVibratorIds()</code>	Enumere todos los identificadores de vibradores disponibles y devuelva una posible lista vacía.
<code>final void vibrate(CombinedVibration effect)</code>	Vibrar con una determinada combinación de efectos.
<code>final void vibrate(CombinedVibration effect, VibrationAttributes attributes)</code>	Vibrar con una determinada combinación de efectos.

Fuente: Propia

5.2. Analizar las metodologías de desarrollo de aplicaciones móviles enfocadas en el diseño de sistemas utilizados por personas en condición de discapacidad visual.

5.2.1. Principales metodologías ágiles

Para el desarrollo de aplicaciones móviles las principales metodologías son: (a) XP, (b) Mobile-D, (c) Scrum. A continuación se describe mediante una tabla de tipo comparativa para identificar las características y los aspectos más relevantes de cada una de ellas.

Tabla 24. Principales metodologías ágiles

	Programación Extrema XP	Mobile-D	Scrum
Definición	Es una metodología de desarrollo ligero (o ágil) basada en una serie de valores y de prácticas de buenas maneras que persigue el objetivo de aumentar la productividad a la hora de desarrollar programas.	Esta metodología está basada en diversas tecnologías como Rational Unified Process, Extreme Programming y Crystal Mehodologies, y su finalidad es intentar obtener pequeños ciclos de desarrollo de forma rápida en dispositivos pequeños.	Es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

	Programación XP	Extrema	Mobile-D	Scrum
Características	● Desarrollo iterativo e incremental. ● Pruebas unitarias. ● Programación en parejas. ● Corrección de todos los errores antes de añadir nueva funcionalidad. ● Refactorización del código. ● Simplicidad en el código. ● Propiedad del código compartida.		● Un costo bajo al realizar un cambio en el proyecto. ● Entrega de resultados de manera rápida. ● Asegura el software adecuado en el momento adecuado.	● Metodología ágil y flexible ● Resultados anticipados. ● Flexibilidad y adaptación. ● Retorno de inversión. ● Mitigación de riesgos. ● Productividad y calidad. ● Alineamiento entre cliente y equipo.
Fases	● Fase planificación del proyecto. ● Fase de diseño. ● Fase de codificación. ● Fase de pruebas		● Fase de Exploración. ● Fase de inicialización. ● Fase de producción. ● Fase de estabilización. ● Fase de prueba.	● Reunión de planificación de Sprint. ● Scrum diario. ● Trabajo de desarrollo durante el sprint. ● Revisión del sprint. ● Retrospectiva del sprint
Equipos que componen los procesos.	● Programador. ● Tester. ● Tracker. ● Entrenador. ● Consultor. ● Gestor.		● Cliente. ● Equipo de desarrollo.	● Product owner. ● Scrum master. ● Scrum Team. ● Cliente.

5.2.1.1. La Metodología Scrum, se seleccionó para la aplicación móvil Vision Access. A continuación se describen cada uno de sus componentes:

5.2.1.1.1. Product Backlog

A continuación se describen en términos generales mediante una tabla la utilidad que presentan los componentes de Product Backlog

Tabla 25. Descripción general de Product Backlog

	ITEM	DESCRIPCION	PRIORIDAD
1	Análisis de Requerimientos	El análisis de requerimiento será esencial para saber qué es lo que se desea lograr con el desarrollo de la aplicación	1
2	Base de Datos para la aplicación	Elegir qué base de datos implementar para guardar los registros de usuarios.	1
3	Diseño de la aplicación	Modelado UML y las interfaces de usuario.	2
4	Registro de Usuarios	Permite el registro de datos de los nuevos usuarios como un id, y nombre	1
5	Autenticación de Usuarios	El sistema podrá ser usado solo por aquellos usuarios que estén registrados.	1
6	Sintetización de Texto a Voz	La aplicación podrá convertir el texto de documentos en voz para los estudiantes con discapacidad visual.	1
7	Interpretación de iconos hápticos.	Por medio de la aplicación los usuarios podrán interpretar sensiblemente las vibraciones emitidas por el móvil..	2
8	Prueba de usabilidad	Las pruebas de usabilidad permiten saber el éxito de la aplicación, es decir si su uso es satisfactorio.	1

Fuente: Propia

5.2.1.1.2. Sprint Backlog

A continuación se describen mediante una tabla en términos generales la utilidad que presentan los componentes de Sprint Backlog

5.2.1.1.2.1. Tablas de Sprint Sprint 0 - Análisis de Requerimientos

Tabla 26. Sprint 0. Análisis de Requerimientos

		Sprint 0				
Ítem Backlog		Tareas	Fecha Inicio	Fecha Fin	Total Semanas	Completada al finalizar el Sprint
1. Análisis de Requerimientos	1	Identificar las necesidades para el desarrollo de la aplicación	12/08/19	20/08/19	1	Si
	2	Toma de Requerimientos	21/08/19	11/09/19	3	Si
	3	Encuesta realizada a los estudiantes con discapacidad visual.	12/09/19	19/09/19	1	Si
	4	Análisis de la información Recolectada	20/09/19	04/10/19	1	Si

Fuente: Propia.

Tabla 27. Tiempo en horas Sprint 0.

Sprint 0					
Tarea	Estimado Horas	Semana 1	Semana 2	Semana 3	Total Horas
1	24	24	0	0	24
2	60	20	20	20	60
3	4	4	0	0	4
4	8	8	0	0	8
Horas Restantes	96	40	20	0	
Horas Estimadas Restantes	96	64.0	32.0	0.0	

Fuente: Propia.

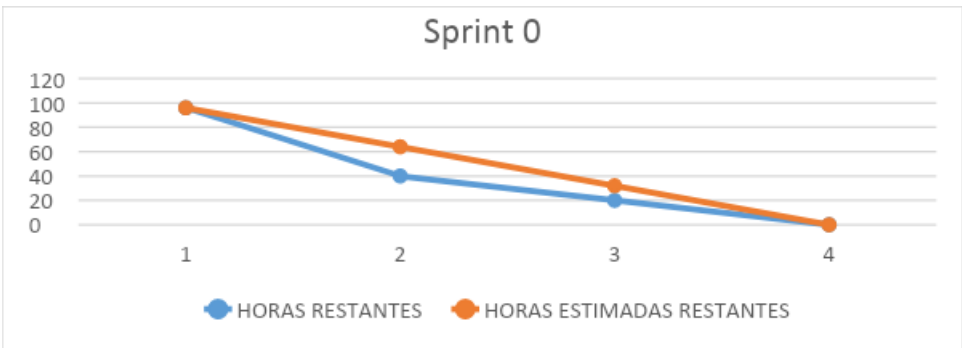


Ilustración 18. Burndown chart Sprint 0

Fuente: Propia..

5.2.1.1.2.2. Tablas de Sprint Sprint 1 - Base de Datos.

Tabla 28. Sprint 1.Base de Datos.

Sprint 1						
Ítem Backlog		Tareas	Fecha inicio	Fecha fin	Total Semanas	Completada al finalizar el Sprint
2. Base de Datos	1	Elección de la base de datos.(Firebase)	08/07/22	15/09/22	1	Si
	2	Conexión de Android Studio con Firebase.	16/07/22	23/09/22	1	Si
	3	Pruebas de la conexión de la base de datos.	16/07/20	23/09/22	1	Si

Fuente: Propia.

Tabla 29. Tiempo en Horas Sprint 1

Sprint 1					
Tarea	Estimado	Semana 1	Semana 2	Semana 3	Total Horas
1	20	20	0	0	20
2	24	24	0	0	24
3	24	24	0	0	24
Horas Restantes	68	0	0	0	
Horas Estimadas Restantes	68	45.3	22.7	0.0	

Fuente: Propia.

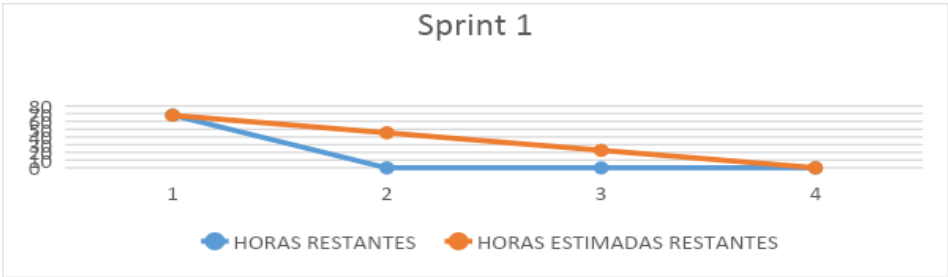


Ilustración 19. Burdonw chart.

Fuente: Propia

5.2.1.1.2.3. Tablas de Sprint Sprint 2 - Diseño de la Aplicación

Tabla 30. Sprint 2. Diseño de la Aplicación

Sprint 2						
Ítem Backlog		Tareas	Fecha inicio	Fecha fin	Total Semanas	Completada al finalizar el Sprint
3. Diseño de la aplicación	1	Modelado UML de la aplicación	24/07/22	15/18/22	3	Si
	2	Diseño de las Interfaces de usuario	18-08-22	28-08-22	1	Si
	3	Diseño de la base de Datos	01-09-22	15-09-22	3	Si

Fuente: Propia.

Tabla 31. Tiempo en Horas Sprint 2.

Sprint 2						
Tarea	Estimado	Semana 1	Semana 2	Semana 3	Total Horas	
1	60	20	20	20	60	
2	32	32	0	0	32	
3	44	15	14	15	44	
Horas Restantes	136	69	35	0		
Horas Estimadas Restantes	136	90.7	45.3	0.0		

Fuente: Propia.

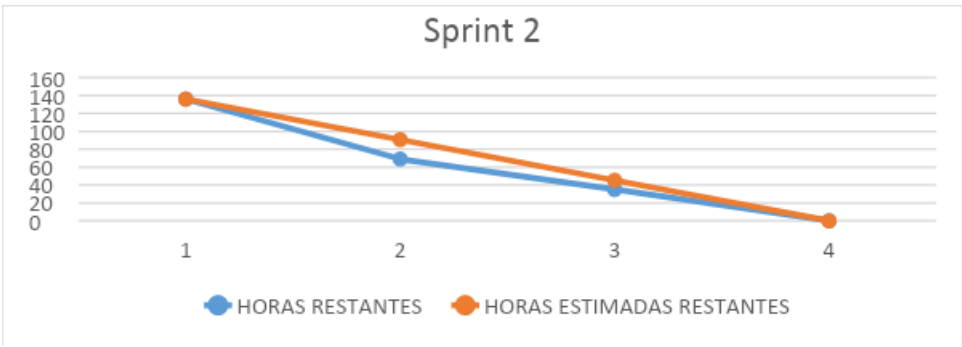


Ilustración 20 Burdown chart Sprint 2

Fuente: Propia.

5.2.1.1.2.4. Tablas de Sprint Sprint 3 - Registro de Usuarios

Tabla 32. Sprint 3.Registro de Usuarios

Sprint 3						
Ítem Backlog		Tareas	Fecha inicio	Fecha fin	Total Semanas	Completada al finalizar el Sprint
4. Registro de Usuarios	1	Diseño de la interfaz de registro.	16/09/22	22/10/22	1	Si
	2	Programación del registro de usuarios.	16/09/22	22/10/22	1	No
	3	Pruebas en registro de usuarios.	16/09/22	22/10/22	1	No

. Fuente: Propia.

Tabla 33. Tiempo en horas Sprint 3.

Sprint 3						
	Tarea	Estimado	Semana 3	Semana 2	Semana 1	Total Horas
	1	24	0	0	24	24
	2	24	0	12	24	36
	3	24	0	12	24	36
	Horas Restantes	72	72	48	-24	
	Horas Estimadas Restantes	72	48.0	24.0	0.0	

Fuente: Propia.

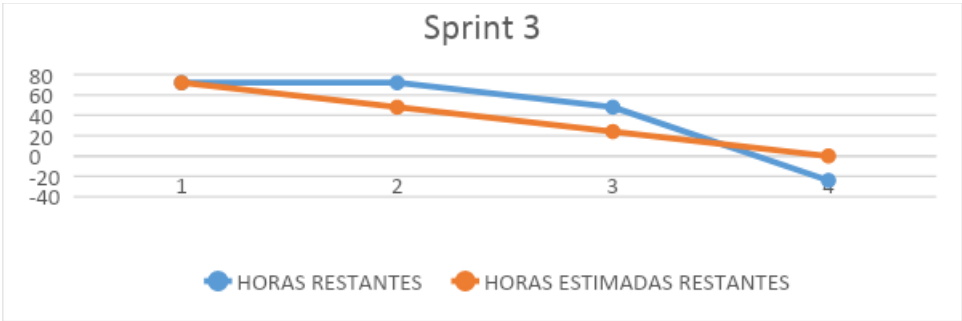


Ilustración 21. Burdown chart Sprint 3

Fuente: Propia.

5.2.1.1.2.5. Tablas de Sprint Sprint 4 - Autenticación de Usuarios

Tabla 34. Sprint 4. Autenticación de Usuarios

Sprint 4						
Ítem Backlog		Tareas	Fecha inicio	Fecha fin	Total Semanas	Completada al finalizar el Sprint
5. Autenticación de Usuarios	1	Definir el método de autenticación.	23/10/22	30/10/22	1	Si
	2	Programación de la autenticación de usuarios.	23/10/22	30/10/22	1	Si
	3	Control de acceso.	23/10/22	30/10/22	1	Si

. Fuente: Propia.

Tabla 35. Tiempo en horas Sprint 4

Sprint 4						
Tarea	Estimado	Semana 1	Semana 2	Semana 3	Total Horas	
1	36	36	0	0	36	
2	36	36	0	0	36	
3	36	36	0	0	36	
Horas Restantes	108	0	0	0		
Horas Estimadas Restantes	108	72.0	36.0	0.0		

. Fuente: Propia.

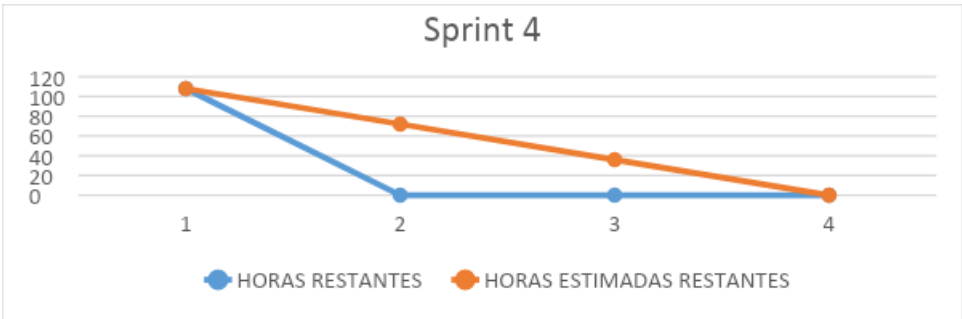


Ilustración 22. Burdown chart Sprint 4 Fuente: Propia.

5.2.1.1.2.6. Tablas de Sprint Sprint 5 - Reconocimiento de Voz

Tabla 36. Sprint 5.Reconocimiento de Voz

		Sprint 5				
Ítem Backlog		Tareas	Fecha inicio	Fecha fin	Total Semanas	Completada al finalizar el Sprint
6. Sintetización de Texto a Voz	1	Elegir la API para el reconocimiento de texto a voz.	01/11/22	08/11/22	1	Si
	2	Programación de la Api de reconocimiento de texto a voz.	08/11/22	15/11/22	1	No
	3	Pruebas del correcto funcionamiento de la Api del proyecto.	08/11/22	15/11/22	1	No

. Fuente: Propia.

Tabla 37. Tiempo en horas Sprint 5.

Sprint 5					
Tarea	Estimado	Semana 3	Semana 2	Semana 1	Total Horas
1	20	0	0	20	20
2	28	0	16	28	44
3	28	0	16	12	28
Horas Restantes	76	76	44	-16	
Horas Estimadas Restantes	76	50.7	25.3	0.0	

Fuente: Propia.

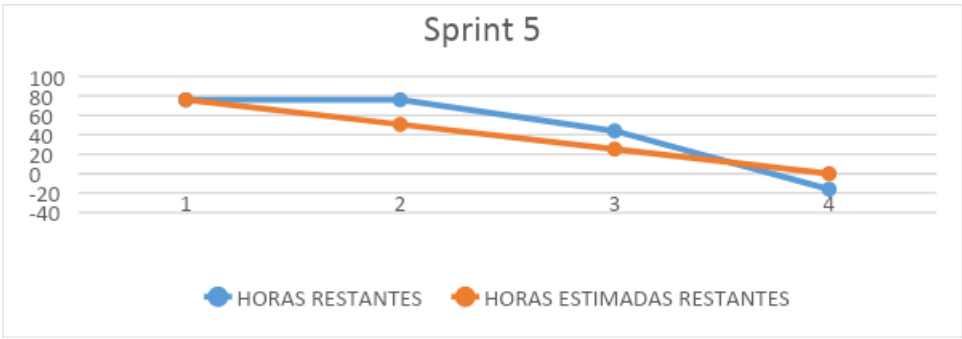


Ilustración 23. Burdown chart Sprint 5

Fuente: Propia.

5.2.1.1.2.7. Tablas de Sprint Sprint 6 - Reconocimiento de Voz

Tabla 38. Sprint 6. Interpretación de iconos hápticos.

Sprint 6						
Ítem Backlog		Tareas	Fecha inicio	Fecha fin	Total Semanas	Completada al finalizar el Sprint
7. Interpretación de iconos hápticos.	1	Interfaz del sistema.	16/11/22	20/11/22	1	Si
	2	Programación de la herramienta	20/11/22	22/11/22	1	Si
	3	Pruebas	22/11/22	23/11/22	1	Si

Fuente: Propia.

Tabla 39. Tiempo en horas Sprint 6.

Sprint 6						
Tarea	Estimado	Semana 1	Semana 2	Semana 3	Total Horas	
1	8	8	0	0	8	
2	20	20	0	0	20	
3	20	20	0	0	20	
Horas Restantes	48	0	0	0		
Horas Estimadas Restantes	48	32.0	16.0	0.0		

Fuente: Propia.

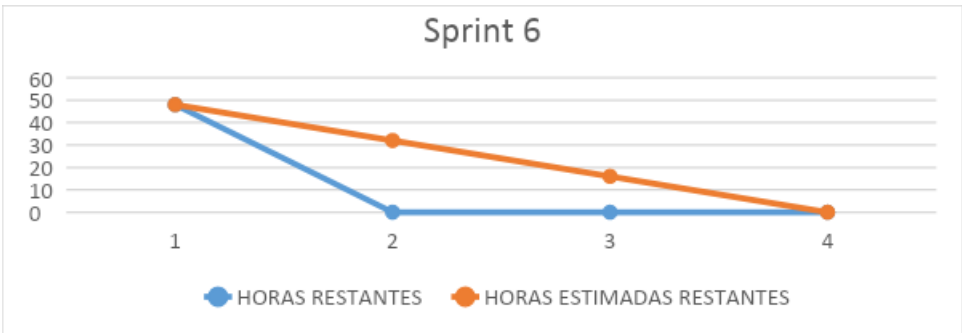


Ilustración 24. Burdown chart Sprint 6

Fuente: propia

5.2.1.1.2.8. Tablas de Sprint Sprint 7 - Pruebas de Usabilidad

Tabla 40. Sprint 7.Pruebas de Usabilidad

		Sprint 7			
Ítem Backlog	Tareas	Fecha inicio	Fecha fin	Total Semanas	Completada al finalizar el Sprint
8. Pruebas de Usabilidad en la aplicación	1 Realizar la respectiva prueba de la aplicación a los estudiantes con discapacidad visual.	24/11/22	25/11/22	2	si

Fuente: Propia.

Tabla 41. Tiempo en horas Sprint 7

Sprint 7					
Tarea	Estimado	Semana 1	Semana 2	Semana 3	Total Horas
1	2	2	0	0	2
Horas Restantes	6	0	0	0	
Horas Estimadas Restantes	6	3.0	0.0	0.0	

Fuente: Propia.

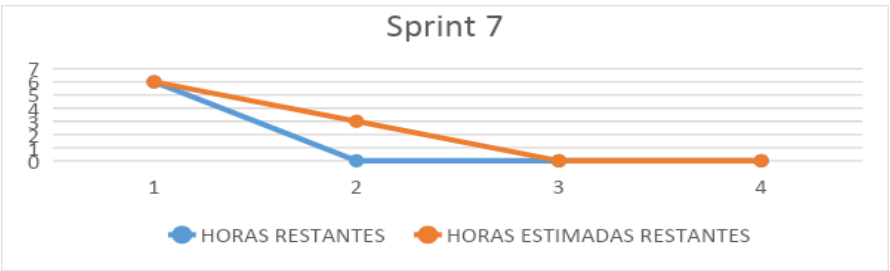


Ilustración 25. Burdown chart Sprint 7

Fuente: propia.

5.2.1.1.3. Roles

Tabla 42. Roles de Scrum para la aplicación

Rol	Persona	Contacto
Scrum Master	Dicson Tarazona Tarazona	Tarazonadicson@gmail.com
Team		
Produc Owner	Ing. Rocío De Belén Contreras	Rocio.contreras@yahoo.co
Customer	Estudiantes con discapacidad visual	Universidad de Pamplona

Fuente: Propia

5.3. Diseñar la estructura de la aplicación de iconos hápticos para el aprendizaje en estudiantes con discapacidad visual.

5.3.1. Desarrollo de la aplicación móvil

Desarrollar una aplicación móvil (App) que sirva como apoyo en el proceso de aprendizaje al estudiante con discapacidad visual de la Universidad de Pamplona sede Villa Del Rosario.

5.3.2. Configuración de las herramientas

Para el desarrollo de la aplicación se hizo necesaria la configuración de las siguientes herramientas: (a) Android Studio: Entorno de desarrollo integrado oficial para la plataforma Android, (b) Java JDK: JDK es el acrónimo de “Java Development Kit”, es decir Kit de desarrollo de Java. Se puede definir como un conjunto de herramientas, utilidades, documentación y ejemplos para desarrollar aplicaciones, (c) Permisos; De conexión a internet (permite la conexión con Firebase), Permiso para que la aplicación pueda acceder a información sobre redes. Permiso para el uso de la memoria de almacenamiento del dispositivo para el reconocimiento de archivos de texto. Permiso del sensor de huella del móvil como alternativa de acceder a la aplicación. Permiso para usar el sistema de vibración del móvil.

5.3.3. Procesos del desarrollo de la aplicación.

5.3.3.1. Registro de Nuevo Usuario



Ilustración 26. Registro de Nuevo Usuario

Fuente: Propia.

En este formulario se solicitan los datos personales del usuario que va a registrarse en la aplicación para hacer uso de la misma. (Correo y Contraseña). (Ver Ilustración 26)

5.3.3.1.1. Validación de los datos registro nuevo usuario

Tabla 43. Descripción de la validación de los datos registro nuevo usuario

REGISTRARSE	MENSAJE
Campo vacío en contraseña	Muestra , lee o vibra el siguiente mensaje: 
Campo vacío en correo	Muestra , lee o vibra el siguiente mensaje: 
La contraseña debe ser mayor a seis dígitos.	Si la contraseña no cumple con los requisitos ,Muestra , lee o vibra el siguiente mensaje: 

Fuente: Propia.

5.3.3.1.2. Código registro de nuevo usuario:

Creamos el método RegistrarUsuario(), dentro de este se hace el registro en la tabla de autenticación por correo y contraseña por medio de auth.createUserWithEmailAndPassword. con la variable instanciada userid, realizamos la búsqueda en Users del hijo y con el hashMap envío la información que deseo guardar en la base de datos. El hasMap es una colección de objetos, con el detalle que a diferencia de Arrays estos no tienen orden.

Para finalizar el registro se utiliza add (map) para que me inserte los hash en la base de datos.

```
//creando nuevo usuario
firebaseAuth.createUserWithEmailAndPassword(email, password)

.addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {
    @Override
    public void onComplete(@NonNull Task<AuthResult> task) {
        //checking if success
        if(task.isSuccessful()){

            FirebaseUser firebaseUser= firebaseAuth.getCurrentUser();
            assert firebaseUser !=null;
            String usuarioid=firebaseUser.getUid();

            Map<String,Object> map =new HashMap<>();

            map.put("Id", usuarioid);
            map.put("correo", email);
            map.put("contraseña", password);
```

Ilustración 27. Código de registro de nuevo usuario

Fuente: Propia

```
//VALIDO SI EL USUARIO FUE CREADO EN BD:
dbFirestore.collection( collectionPath: "usuarios").
    add(map).addOnSuccessListener(new OnSuccessListener<DocumentReference>() {
        @Override
        public void onSuccess(DocumentReference documentReference) {
            Log.d(LOG_TAG, msg: "USUARIO AGREGADO CON ID: " + documentReference.getId());
        }
    });
    Toast.makeText( context: MainActivity.this, text: "Se ha registrado el usuario con el email: "+ TextEmail.getText()
} else {
    if (task.getException() instanceof FirebaseAuthUserCollisionException) { //si se presenta una colisión
        Toast.makeText( context: MainActivity.this, text: "Este usuario ya existe ", Toast.LENGTH_SHORT).show();
        ttsManager.initQueue( text: "Este usuario ya existe ");
    } else{

        Toast.makeText( context: MainActivity.this, text: "No se pudo registrar el usuario ", Toast.LENGTH_LONG).show();
        ttsManager.initQueue( text: "No se pudo registrar el usuario ");
    }
}
progressDialog.dismiss();
});
```

Ilustración 28. Programación para Registrar Nuevo Usuario

Fuente: Propia.

Adicionalmente se pensó en diferenciar esta acción de registrar nuevo usuario mediante la asignación de patrones de vibración, en este caso al oprimir botón registrar el móvil realiza las vibraciones indicadas mediante una cadena de valores que sigue el orden de un tiempo en milisegundos de retraso, espera y vibración. Para así formar un patrón que sea identificable al usuario que utiliza la aplicación.

```
@Override
public void onClick(View v){

    switch (v.getId()) {

        case R.id.bottonRegistrarUsuario:

            // Mili-sg, 0 es por retraso, x VIBRA , x DUERME...
            long[] pattern= {0,300,500,300};
            //-1 NO REPETIR
            vibratorr.vibrate(pattern, repeat: -1);

            String text=btnRegistrar.getText().toString();
            ttsManager.initQueue(text);

            //Invocamos al método:
            registrarUsuario();

            break;
```

Ilustración 29. Programación para emitir vibración (icono haptico)

Fuente: Propia

5.3.3.2. Iniciar Sesión

En este formulario debemos ingresar correo, contraseña; tenemos también un Registrar usuario para insertar uno nuevo.



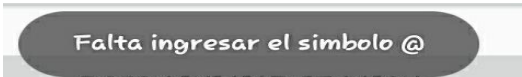
Ilustración 30. Iniciar Sesión

Fuente: Propia

5.3.3.2.1. Validación de los datos iniciar sesión

A continuación se describen mediante una tabla las diversas respuestas emitidas de la aplicación al usuario cuando este realiza alguna acción respecto a las funciones que realiza la aplicación.

Tabla 44. Validación de los datos en Inicio de Sesión

INICIAR SESION	MENSAJE
Campo vacío en contraseña	Muestra y lee el siguiente mensaje: 
Campo vacío en correo	Muestra y lee el siguiente mensaje: 
Si digito todos los campos pero al correo le hizo falta el @	Muestra y lee el siguiente mensaje: 
La contraseña debe ser mayor a seis dígitos.	Muestra y lee el siguiente mensaje:

Su contraseña debe tener mas de 6 caracteres

Si completo todos los datos del formulario , pero el usuario no se encuentra registrado en la base de datos

Muestra y lee el siguiente mensaje:

No se pudo iniciar el usuario

Fuente: Propia.

5.3.3.1.2. Código Inicio de sesión loguearUsuario () :

```
private void loguearUsuario() {  
    //Obtenemos el email y la contraseña desde las cajas de texto  
    final String email = TextEmail.getText().toString().trim();  
    String password = TextPassword.getText().toString().trim();  
  
    //Verificamos que las cajas de texto no esten vacías:  
  
    if (TextUtils.isEmpty(email)) {  
        Toast.makeText( context: this, text: "Se debe ingresar un email", Toast.LENGTH_LONG).show();  
        ttsManager.initQueue( text: "Se debe ingresar un email ");  
        return;  
    }  
  
    if (TextUtils.isEmpty(password)) {  
        Toast.makeText( context: this, text: "Falta ingresar la contraseña", Toast.LENGTH_LONG).show();  
        ttsManager.initQueue( text: "Falta ingresar la contraseña");  
        return;  
    }  
  
    progressDialog.setMessage("Realizando consulta en linea...");  
    progressDialog.show();  
}
```

Ilustración 31. Código de Inicio de sesión loguearUsuario ()

Fuente propia

```

//loguear usuario
firebaseAuth.signInWithEmailAndPassword(email, password)
    .addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            //checking satisfactorio
            if (task.isSuccessful()) {

                int pos = email.indexOf("@");
                String user = email.substring(0, pos);
                Toast.makeText( context: MainActivity.this, text: "Bienvenido: " + TextEmail.getText(), Toast.LENGTH_LONG).

                //ABRO PRINCIPAL
                Intent intencion = new Intent(getApplicationContext(), Principal.class);
                intencion.putExtra(Principal.user, user);
                startActivity(intencion);

            } else {
                if (task.getException() instanceof FirebaseAuthUserCollisionException) { //si se presenta una colisión
                    Toast.makeText( context: MainActivity.this, text: "Ese usuario ya existe ", Toast.LENGTH_SHORT).show();
                    ttsManager.initQueue( text: "Ese usuario ya existe");
                }
            }
        }
    })

```

Ilustración 32. Parte II de Código de Inicio de sesión loguearUsuario ()

Fuente propia

El SDK de Firebase Authentication proporciona métodos para crear y administrar usuarios que usan sus direcciones de correo electrónico y contraseñas para iniciar sesión.

En el método loguearUsuario() implementamos auth.signInWithEmailAndPassword donde enviamos los parámetros de usuario y contraseña y así validamos que se encuentre registrado en la base de datos. Si los datos ingresados son correctos nos mostrará la interfaz principal.

```

@Override
public void onClick(View v){

    switch (v.getId()) {

        case R.id.botonRegistrarUsuario:

            // Mili-sg, 0 es por retraso, x VIBRA , x DUERME...
            long[] pattern= {0,300,500,300};
            //-1 NO REPETIR
            vibratorr.vibrate(pattern, repeat: -1);

            String text=btnRegistrar.getText().toString();
            ttsManager.initQueue(text);

            //Invocamos al método:
            registrarUsuario();

            break;
    }
}

```

Ilustración 33. Código de Incorporación del método vibrator en botón registrar usuario

Fuente: Propia

También se utiliza un método llamado `Vibrator()` el cual se usa para dar vibración al móvil cuando el usuario realiza alguna acción en la app como en este caso tocar el botón nombrado (`botonRegistrarUsuario`). Esto permite identificar y diferenciar mediante diversos patrones de vibración la acción que está realizando.

5.3.3.3. Opción de Autenticación Biométrica local

Esta opción de autenticación permite al usuario acceder a la aplicación de forma más ágil y con un alto nivel de seguridad ya que tiene patrones únicos de cada persona como lo es la huella dactilar. También los mensajes de aviso son leídos al usuario invidente para que logre interpretar la acción a realizar. A continuación se aprecia la interfaz de solicitud de Autenticación biométrica para el usuario:



Ilustración 34. Interfaz de autenticación biométrica

Fuente: Propia

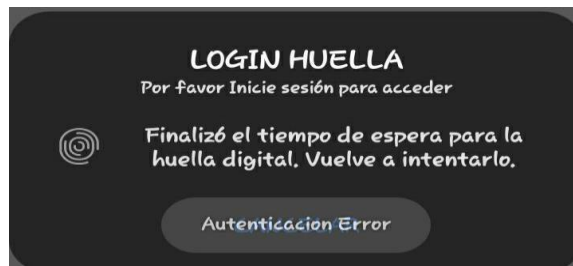
5.3.3.3.1. Validaciones de acceso biométrico

Tabla 45. Validaciones de acceso biométrico

Acceso biométrico	Mensaje
	Muestra y lee el siguiente mensaje:
Huella incompleta	
	Muestra y lee el siguiente mensaje:
Huella incorrecta	
	Muestra y lee el siguiente mensaje:
No identifica Ninguna Huella	

Máximo tiempo de inactividad sin registrar
huella

Muestra y lee el siguiente mensaje:



Máximo de intentos fallidos (4) con Huella
incorrecta

Muestra y lee el siguiente mensaje:



Fuente: Propia.

5.3.3.3.2. Código de Validaciones de acceso biométrico

En la clase BimetricCallback() declaramos los siguientes métodos de devolución de llamada que usaremos más adelante:

```
public interface BiometricCallback {  
  
    void onSdkVersionNotSupported();  
  
    void onBiometricAuthenticationNotSupported();  
  
    void onBiometricAuthenticationNotAvailable();  
  
    void onBiometricAuthenticationPermissionNotGranted();  
  
    void onBiometricAuthenticationInternalError(String error);  
  
    void onAuthenticationFailed();  
  
    void onAuthenticationCancelled();  
  
    void onAuthenticationSuccessful();  
  
    void onAuthenticationHelp(int helpCode, CharSequence helpString);  
  
    void onAuthenticationError(int errorCode, CharSequence errString);  
}
```

Ilustración 35. Código de declaración de Métodos usados en Validaciones de acceso biométrico

Fuente: Propia

Posteriormente; en la clase Main Activity desarrollamos los métodos de llamada para el acceso biométrico en la aplicación, a continuación se define la utilidad de cada uno de ellos:

Utilidad: Se llamará si la versión sdk del dispositivo no es compatible con la autenticación biométrica

```
@Override
public void onSdkVersionNotSupported() {
    Toast.makeText(context: MainActivity.this, text: " sdk del dispositivo no es compatible con la autenticación " +
        "biométrica", Toast.LENGTH_LONG).show();

    ttsManager.initQueue(text: "sdk del dispositivo no es compatible con la autenticación");
}
```

Ilustración 36. Código de Método onSdkVersionNotSupported()

Fuente: Propia

Utilidad: Se llamará si el dispositivo no contiene ningún sensor de huellas dactilares

```
@Override
public void onBiometricAuthenticationNotSupported() {
    Toast.makeText(context: MainActivity.this, text: " el dispositivo no contiene ningún sensor de huellas dactilares",
        ttsManager.initQueue(text: "el dispositivo no contiene ningún sensor de huellas dactilares");
}
```

Ilustración 37. Código de Método onBiometricAuthenticationNotSupported()

Fuente: Propia

Utilidad: se hace el llamado si el dispositivo no tiene datos biométricos registrados.

```
@Override
public void onBiometricAuthenticationNotAvailable() {
    Toast.makeText(context: MainActivity.this, text: "El dispositivo no tiene datos biométricos registrados en el dispositivo",
        ttsManager.initQueue(text: "El dispositivo no tiene datos biométricos registrados en el dispositivo");
}
```

Ilustración 38. Código de Método onBiometricAuthenticationNotAvailable()

Fuente: Propia

Utilidad: hace el llamado si el permiso android.permission.USE_BIOMETRIC no se otorga a la aplicación.

```

@Override
public void onBiometricAuthenticationPermissionNotGranted() {
    Toast.makeText( context: MainActivity.this, text: " Permiso Denegado", Toast.LENGTH_LONG).show();
    ttsManager.initQueue( text: " Permiso Denegado ");
}

```

Ilustración 39. Código de Método onBiometricAuthenticationPermissionNotGranted()

Fuente: Propia

Utilidad: Este método se llama si uno de los campos como el título, subtítulo, descripción o el texto del botón negativo está vacío

```

@Override
public void onBiometricAuthenticationInternalError(String error) {
    Toast.makeText( context: MainActivity.this, text: " Error en Autenticacion interna por campo vacios", Toast.LENGTH_LONG)
    ttsManager.initQueue( text: " Error en Autenticacion interna por campo vacios ");
}

```

Ilustración 40. Código de Método onBiometricAuthenticationInternalError(String error)

Fuente: Propia

Utilidad: Cuando la huella digital no coincide con ninguna de las registradas en el dispositivo, entonces se activará esta devolución de llamada.

```

@Override
public void onAuthenticationFailed() {
    Toast.makeText( context: MainActivity.this, text: " Huella No Coincide", Toast.LENGTH_LONG).show();
    ttsManager.initQueue( text: " Huella No Coincide ");
}

```

Ilustración 41. Código de Método onAuthenticationFailed()

Fuente: Propia

Utilidad: este llamado se realiza cuando la autenticación es cancelada por el usuario.

```

public void onAuthenticationCancelled() {
    Toast.makeText( context: MainActivity.this, text: " Autenticacion Cancelada", Toast.LENGTH_LONG).show();
    ttsManager.initQueue( text: " Autenticacion Cancelada ");
}

```

Ilustración 42. Código de Método onAuthenticationCancelled()

Fuente: Propia

Utilidad: Cuando la huella dactilar se ha emparejado con una registrada en el dispositivo con éxito, entonces se activará esta devolución de llamada y le da acceso a la aplicación.

```
@Override
public void onAuthenticationSuccessful() {
    Toast.makeText(context: MainActivity.this, text: " Authentication EXITOSA", Toast.LENGTH_LONG).show();
    ttsManager.initQueue(text: " Authentication EXITOSA ");

    // Abre layout Principal

    Intent intencion = new Intent(getApplicationContext(), Principal.class);
    intencion.putExtra(user, user);
    startActivity(intencion);
}
```

Ilustración 43. Código de Método onAuthenticationSuccessful()

Fuente: Propia

Utilidad: Este método se llama cuando se ha producido un error no fatal durante la autenticación. La devolución de llamada se proporcionará con un código de ayuda para identificar la causa de la falla.

```
@Override
public void onAuthenticationHelp(int helpCode, CharSequence helpString) {
    Toast.makeText(context: MainActivity.this, text: "Autenticacion de Ayuda:", Toast.LENGTH_LONG).show();
    ttsManager.initQueue(text: " Autenticacion de Ayuda ");
}
```

Ilustración 44. Código de Método onAuthenticationHelp(int helpCode, CharSequence helpString)

Fuente: Propia

Utilidad: Cuando se ha encontrado un error irrecuperable y el proceso de autenticación ha se ha completado sin éxito, entonces se activará esta devolución de llamada. Se proporciona la devolución de llamada con un código de error para identificar la causa del error, junto con el mensaje de error.

```

@Override
public void onAuthenticationError(int errorCode, CharSequence errString) {
    Toast.makeText( context: MainActivity.this, text: "Autenticacion Error", Toast.LENGTH_LONG).show();
    ttsManager.initQueue( text: " Autenticacion Fatal Error ");
}

```

Ilustración 45. Código de Método `onAuthenticationError(int errorCode, CharSequence errString)`

Fuente: Propia

5.3.3.4. Interfaz Principal de usuario

En la interfaz principal del usuario encontramos el proceso que puede realizar: TEXTO A VOZ



Ilustración 46. Interfaz Principal de usuario

Fuente: Propia

El primer proceso del usuario al oprimir botón `TEXTO A VOZ` será redirigirlo a layout que le permite seleccionar el documento de tipo pdf que desea abrir al oprimir el Botón (Cargar Pdf) para proceder a efectuar el proceso de lectura mediante el motor de síntesis de voz.



Ilustración 47. Interfaz Leer Documento PDF

Fuente: Propia

Se utiliza un primer TextView (Referencia) para validar la ruta donde está almacenado el archivo pdf. y un segundo TextView donde se mostrará el texto extraído de este mismo documento. Por consiguiente; después de proceder a cargar documento nos aparecerá la siguiente interfaz de tipo CHOOSE_PDF_FROM_DEVICE :

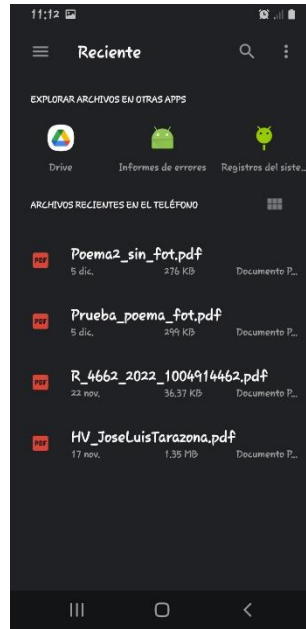


Ilustración 48. Interfaz de CHOOSE_PDF_FROM_DEVICE

Fuente: Propia

Después de seleccionarlo el documento se ejecuta internamente el proceso para extraer el texto de este mismo. Como por ejemplo a continuación se observa la ruta del archivo junto con el contenido del documento:



Ilustración 49. Interfaz de extracción del texto del documento junto a su ruta de almacenamiento

Fuente: Propia

5.3.3.4.1. Código de extracción de texto del documento PDF

El método `extractTextPdfFile()` es el que nos permite realizar esta extracción de texto y está codificado de la siguiente manera:

```
private void extractTextPdfFile (Uri uri){
    try {
        inputStream = TextoVoz.this.getContentResolver().openInputStream(uri);
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    }
    new Thread(new Runnable() {
        @Override
        public void run() {
            String filecontent = "";
            StringBuilder builder = new StringBuilder();
            PdfReader reader = null;
            try {
                if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP){
                    reader = new PdfReader(inputStream);
                    int pages = reader.getNumberOfPages();
                    // FOR PARA BUSCAR TODO LO QUE CONTENGA EL PDF
                    for(int i=1;i<=pages;i++){
                        filecontent = PdfTextExtractor.getTextFromPage(reader,i);
                        builder.append(filecontent);
                    }
                }
                reader.close();
                //Toast.makeText(MainActivity.this, "Se cargo el PDF con éxito", Toast.LENGTH_SHORT).show();
                runOnUiThread() -> txt_contenido_pdf.setText(builder.toString());
            }catch (IOException e){
                //Toast.makeText(MainActivity.this, "Error al cargar el PDF", Toast.LENGTH_SHORT).show();
            }
        }
    }).start();
}
```

Ilustración 50. Código de extracción de texto del documento PDF

Fuente: Propia

Finalmente el motor de síntesis de texto procede a dar lectura al contenido extraído mediante la acción pulsar el botón LEER



Ilustración 51 Interfaz Botón leer

Fuente: Propia

5.3.3.4.2. Código de lectura de texto

En la acción de oprimir este botón realizamos el llamado de método `initQueue()` de la clase `TTSManager`

```
ttsManager=new TTSManager();

hablarAhoraBoton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        String text=editText2.getText().toString(); // OBTIENE TEXTO DE EDITEXT
        ttsManager.initQueue(text); // ENVIA TEXTO A METODO INITQUEUE
    }
});
```

Ilustración 52. Código de lectura de texto

Fuente: Propia

A continuación se muestra la codificación de los métodos contenidos en la clase `TTSManager` incluido el Método `initQueue()` que es el que realiza la acción de lectura al texto.

```
public class TTSManager {

    // DECLARACION DE VARIABLES:

    private TextToSpeech mTts = null;
    private boolean isLoading = false;

    public void init(Context context) {
        try {
            mTts = new TextToSpeech(context, onInitListener); // ASIGNACION DE VARIABLE A METODO TextToSpeech
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```

private TextToSpeech.OnInitListener onInitListener = new TextToSpeech.OnInitListener() {
    @Override
    public void onInit(int status) {
        Locale spanish = new Locale( language: "es", country: "ES"); // DECLARACION DE TIPO DE IDIOMA ESPAÑOL
        if (status == TextToSpeech.SUCCESS) {
            int result = mTts.setLanguage(spanish); //ENVIA IDIOMA A VARIABLE RESULT
            isLoading = true;

            if (result == TextToSpeech.LANG_MISSING_DATA || result == TextToSpeech.LANG_NOT_SUPPORTED) {
                //SI IDIOMA ES INCOMPATIBLE

                Log.e( tag: "error", msg: "Este Lenguaje no esta permitido");
            }
        } else {
            Log.e( tag: "error", msg: "Fallo al Inicilizar!");
        }
    }
};

public void addQueue(String text) {
    if (isLoading)
        mTts.speak(text, TextToSpeech.QUEUE_ADD, params: null); //AÑADE PARAMETRO TEXTO A METODO TextToSpeech
    else
        Log.e( tag: "error", msg: "TTS No inicializado");
}

public void initQueue(String text) {
    if (isLoading)
        mTts.speak(text, TextToSpeech.QUEUE_FLUSH, params: null); //LEE PARAMETRO RECIBIDO
    else
        Log.e( tag: "error", msg: "TTS No inicializado");
}
}

```

Ilustración 53. Codificación de los métodos contenidos en la clase TTSManger

Fuente: Propia

5.3.3.5. Cerrar sesión de usuario

Al usuario oprimir el botón de cerrar sesión mostrado a continuación:



Ilustración 54. Interfaz botón Cerrar sesión de usuario

Fuente: Propia

Se ejecuta en esta acción la siguiente codificación:

```
btnCerrarSesion=(Button)findViewById(R.id.buttonCerrarSesion);
btnCerrarSesion.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        firebaseAuth.getInstance().signOut(); // DA DE BAJA LA SESION
        Intent i1 = new Intent( packageContext: Principal.this, MainActivity.class); // ENVIA A ACTIVITY LOGIN
        startActivity(i1);
        Toast.makeText( context: Principal.this, text: "SESION CERRADA", Toast.LENGTH_SHORT).show();
        ttsManager.initQueue( text: "SESION CERRADA, GRACIAS POR USAR VISION READING");
    }
});
```

Ilustración 55. Codificación botón Cerrar sesión de usuario

Fuente: Propia

Finalmente allí mediante el atributo `signOut()` propio de Firebase Authentication se da de baja a la sesión y se redirecciona al layout inicial de logueo de usuario. También el usuario puede interpretar una sensación de tipo vibratoria que le ayuda a identificar la acción que está realizando en la interfaz de la aplicación. Al ejecutar este método cierra todos los procesos internos que el usuario haya realizado.

5.4. Validar el uso de la Aplicación móvil en estudiantes con discapacidad visual de la Universidad de Pamplona Sede Villa Rosario.

5.4.1. Análisis de las encuestas realizadas para la aplicación móvil VisionAccess

Una de las formas para validar el uso de la aplicación es mediante la implementación de una encuesta, ya que estas resultan de gran importancia para conocer el impacto que genera en la accesibilidad a aquellas personas que lo requieren, sabiendo las actividades y acciones que el usuario realiza ayudan a mejorar aspectos de retroalimentación.

5.4.1.1. Encuesta realizada a personas con discapacidad visual de la universidad de pamplona después de las pruebas de usabilidad de la aplicación.

- 1. ¿Cómo aprecia el uso de la aplicación VisionAcces?
 - a) fácil
 - b) difícil
- 2. ¿Le gusta la ayuda que brinda la aplicación para dar lectura a documentos digitales?
 - a) Si
 - b)No
- 3. En términos generales, ¿Qué opina de la aplicación VisionAcces?
- 4. ¿Qué modificaciones le haría a la aplicación?

Ilustración 56. Encuesta realizada después de validación a la aplicación VisionAccess

Fuente propia

5.4.1.2. Resultado de la Encuesta de los Estudiantes

1. ¿Cómo aprecia el uso de la aplicación VisionAccess?

- a) fácil
- b) difícil

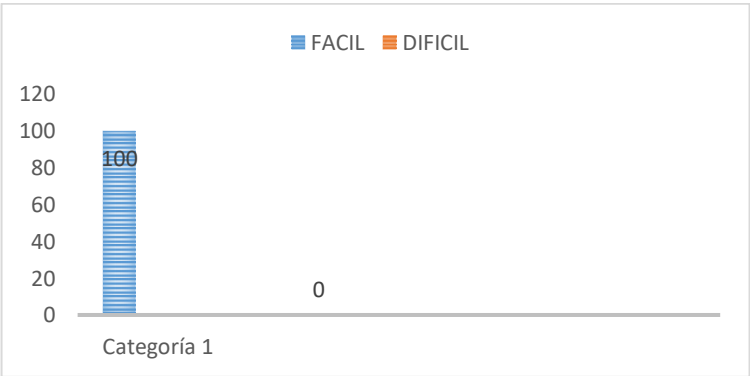


Ilustración 57. Grafica representativa del resultado a la pregunta ¿Cómo aprecia el uso de la aplicación VisionAccess?

Fuente propia

Un 100 % en los encuestados respondieron que es fácil la aplicación VisionAccess

2. ¿Le gusta la ayuda que brinda la aplicación para dar lectura a documentos digitales?

a) Si

b) No

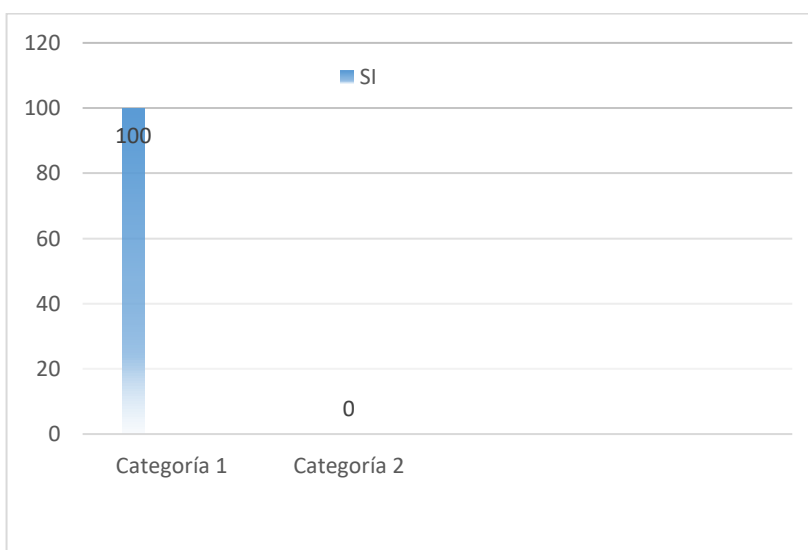


Ilustración 58. Grafica representativa del resultado a la pregunta ¿Le gusta la ayuda que brinda la aplicación para dar lectura a documentos digitales?

Un 100 % en los encuestados seleccionaron que si le gusto la aplicación para dar lectura a documentos digitales

3. En términos generales, ¿Qué opina de la aplicación VisionAccess?

1 Respuesta: Buena

2 Respuesta: Bien

3 Respuesta: Positiva

4 Respuesta: Incluyente

5 Respuesta: De ayuda

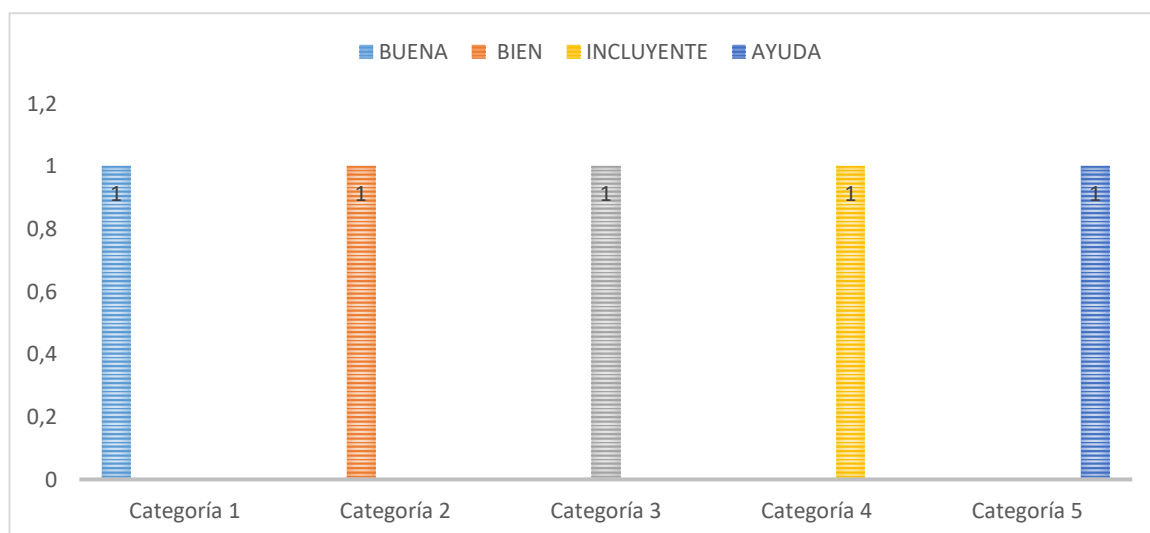


Ilustración 59. Grafico Representativo al ala pregunta ¿Qué opina de la aplicación VisionAccess?

Fuente propia

Los estudiantes con discapacidad visual respondieron a la pregunta ¿Qué opina de la aplicación VisionAccess? En el cual un 20 % respondió Buena, un 20 % respondió Bien, Un 20 % respondió Positiva, Un 20 % respondió Incluyente y un 20 % respondió De ayuda, dando como resultado el 100 % de respuestas en los encuestados.

La pregunta. ¿Qué opina de la aplicación VisionAccess?. Cada uno de los encuestados dio su apreciación en la aplicación VisionAccess con las siguientes respuestas: buena, bien, positiva, incluyente, de ayuda.

4. ¿Qué modificaciones le haría a la aplicación?

Ninguna

Voz más lenta

Ampliar cantidad de archivos

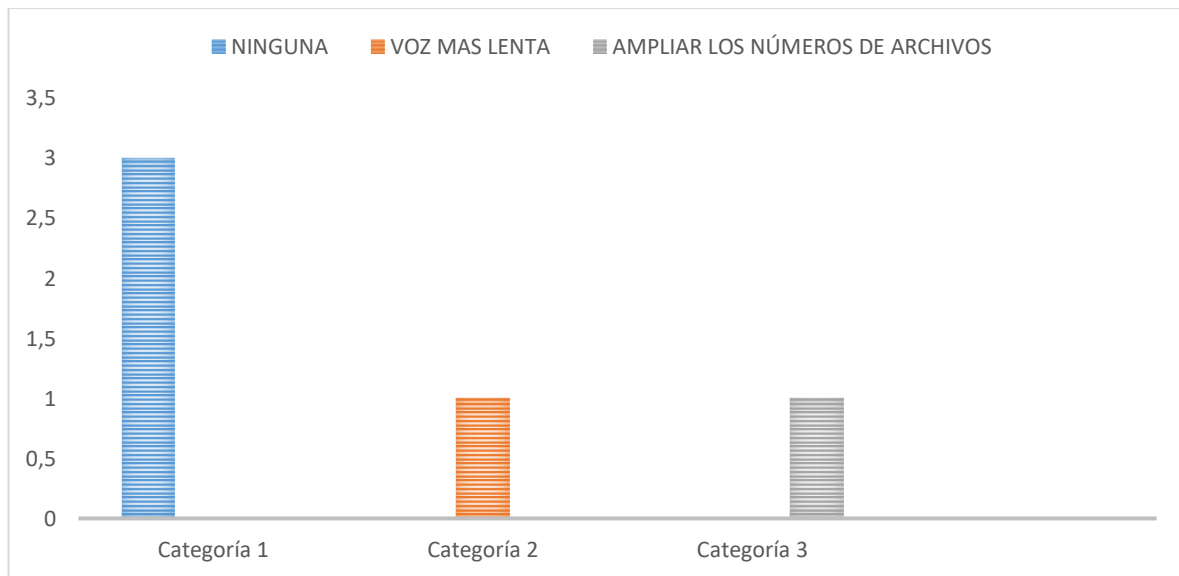


Ilustración 60. Grafica representativa de la pregunta ¿Qué modificaciones le haría a la aplicación?

Fuente propia

Un 90% en los encuestados seleccionaron que no le harían ninguna modificación a la aplicación, seguidamente, un 0,5% respondieron que modificarán en la voz porque es muy lenta y por último un 0.5% que le modificarán en la ampliación de los números de archivos.

5.4.2. Pruebas de Usabilidad

La usabilidad es un factor importante en este desarrollo de software, la cual tiene que ver con la facilidad que tienen los usuarios con limitación visual al ingresar a la aplicación y poder conseguir lo que están buscando dentro de ella. es por ello que se creó de tal forma que resulte cómoda de que ellos logren navegar por la aplicación interpretando por sonidos y vibraciones las acciones que van a desarrollar.

Para poder saber el grado de usabilidad en los usuarios que van a utilizar la aplicación se realizó un estudio a través de encuestas después de haber utilizado la aplicación, estas pruebas realizadas se hicieron con estudiantes en condición de discapacidad visual de la universidad de Pamplona.

5.4.2.1. Componentes de la usabilidad

Para lograr mejorar la usabilidad existen varios componentes que permiten, además de medir la facilidad de uso, mejorar el proceso de diseño; con la intención de que los usuarios puedan tener una mejor experiencia, y se puedan sentir totalmente cómodos cuando utilicen la interfaz.

5.4.2.1.1. Facilidad de aprendizaje

Se trata de un punto muy importante, ya que esto es lo que va a permitir descubrir qué tan fácil es para las personas llevar a cabo todas las acciones que necesiten al momento de acceder a la aplicación. Por ello se pensó en la autenticación biométrica ya que facilita la acción de acceder a la aplicación con una simple pulsación de la huella dactilar.

5.4.2.1.2. Eficiencia

Una vez que los usuarios hayan ingresado a la interfaz y ya conocen el diseño de la misma, el paso a seguir es ofrecerle la rapidez necesaria para que puedan realizar las acciones y tareas sin ningún inconveniente que los limite a poder hacerlo. Es entonces que la estructura del diseño de la aplicación es muy sencilla ya que fue pensada para ir directamente a lo que se necesita realizar como lo es la lectura de documentos digitales.

5.4.2.1.3. Memorabilidad

Este punto está relacionado con la facilidad que existe en recordar el sistema cuando pasa un determinado tiempo sin ingresar en él. Esto también tiene que ver con la facilidad que tenga el usuario con discapacidad visual para interactuar con la aplicación. Un elemento que se incorporó fue el uso de iconos hápticos que da una experiencia sensorial única lo cual tiende a aumentar el desempeño de aprendizaje en cuanto a la navegación por la aplicación. Ya que a medida que el usuario realiza una acción se lee la descripción de esta y a su vez se emite una vibración única que la identifica lo cual permite que al usarla en repetidas ocasiones se mejore el nivel de memorabilidad en la exploración de la App.

5.4.2.1.4. Errores

Este componente permite saber cuáles son los posibles errores que pueden existir; algo que se ve reflejado a través del comportamiento que tiene el usuario cuando utiliza la aplicación. Además, es importante ya que permite tener una retroalimentación directa con la finalidad de mejorar aspectos a futuro.

5.4.2.1.5. Satisfacción

El punto de satisfacción está relacionado a la experiencia de usuario que tienen al utilizar la aplicación. Esto determina la calidad que se ofrece a las personas que utilizan la aplicación. Es por ello que en este proyecto se hace uso de diversos elementos que usa el dispositivo móvil como los motores de lectura, sistema de vibración, sistema biométrico entre otros, para colocarlos a disposición del usuario y más en este caso donde estos requieren de ayuda sensorial para realizar exitosamente sus tareas.

5.5. Instrucciones para el estudiante con discapacidad visual

- Instalar la App en su dispositivo móvil.
- Interpretar las vibraciones al realizar una acción en la aplicación
- Registrarse en la aplicación.
- Registrar huella biométrica en el móvil.
- Iniciar sesión.
- Seleccionar el archivo que desea sintetizar a voz
- Escuchar el texto del contenido del documento.
- Cerrar sesión en la aplicación

5.5. Pruebas de Caja Negra

Las pruebas de caja negra, conocidas también como black box testing, son una técnica donde se busca la verificación de las funcionalidades de nuestra aplicación analizada, sin tomar como referente la estructura del código interno, las rutas de tipo internas ni la información referente a la implementación. Esto quiere decir que la prueba se lleva a cabo con desconocimiento del funcionamiento del sistema interno, debido a que se enfoca en la entrada y salida del sistema.

De manera que se puede asegurar que el objetivo de las pruebas de caja negra está relacionado con la validación de los recursos funcionales de la aplicación.

A continuación se presentan los datos de entrada y salida de la aplicación.

5.5.1. Tabla de Prueba de caja negra para el registro de usuario

Tabla 46. Prueba de caja negra - Registro de usuario

REGISTRO DE USUARIO				
ID	Descripción	N° casos	Datos de Entrada	Datos de Salida
1	Validar campos vacíos	1	No digitar ningún campo	Muestra mensaje de error
		2	Con tan solo un campo que falte por digitar	Muestra mensaje de error
		3	Si digita todos los campos correctamente	Registra el usuario
2	Validar si el usuario ya existe	1	Si digita un usuario que ya se registró antes y selecciona el mismo rol	Muestra mensaje de error
		2	Si digita un usuario que ya se registró pero selecciona otro rol	Muestra el siguiente mensaje, y deja registrar al usuario.

		3	Si digita un usuario registrado anteriormente con los dos roles	Muestra Mensaje de error
3	Validar las contraseñas	1	Si las contraseñas no coinciden	Muestra mensaje de error
		2	Si las contraseñas no contienen por lo menos una mayúscula, una minúscula, un número, un carácter especial y >7 dígitos	Muestra mensaje de error
		3	Si las contraseñas contienen los parámetros correctos	Se Registra al Usuario
4	Validar el correo	1	Si digito todos los campos pero al correo le hizo falta el @	Muestra un mensaje de error
		2	Si digito los campos completos	Se Registra al Usuario

. Fuente: Propia.

5.5.2. Tabla de Prueba de caja negra para iniciar sesión

Tabla 47. Prueba de caja negra para iniciar sesión.

INICIO DE SESION				
ID	Descripción	N° casos	Datos de Entrada	Datos de Salida
		1	No digitar ningún campo	Muestra mensaje de error
		2	Si digita el correo pero no la contraseña	Muestra mensaje de error
		3	Si digita la contraseña pero no el correo	Muestra mensaje de error

1	Validar campos vacíos	4	Si digita el correo, la contraseña	Inicia Sesión
2	Validar contraseña	1	Si las contraseñas más de 6 dígitos	Muestra mensaje de error
		2	Si las contraseñas contienen los parámetros correctos	Inicia Sesión
3	Validar Correo	1	Si digito el correo pero le hizo falta el @	Muestra un mensaje de error
		2	Si digito todos los campos	Inicia Sesión
4	Validar Usuario	1	Si el usuario no existe en la base de datos	Muestra mensaje de error
		2	Si el usuario existe en la base de datos	Inicia Sesión

Fuente: Propia.

5.5.3. Tabla de Prueba de caja negra en la autenticación biométrica

Tabla 48. Prueba de caja negra para Autenticación Biométrica.

AUENTICACION BIOMETRICA				
ID	Descripción	N° casos	Datos de Entrada	Datos de Salida
		1	La Huella es incompleta	Muestra mensaje de error
		2	La Huella es incorrecta	Muestra mensaje de error

1	Validar Acción con sensor biométrico	3	No identifica Ninguna Huella	Muestra mensaje de error
		4	Máximo tiempo de inactividad sin registrar ninguna huella	Muestra mensaje de error
		5	Si la huella es correcta	Inicia Sesión
		6	Máximo de intentos fallidos con Huella incorrecta (más de 4)	Muestra mensaje de error
		7	Desliza la huella muy rápido sobre el sensor	Muestra mensaje de error
		1	Si el usuario no existe en la base de datos	Muestra mensaje de error
		2	Si el usuario existe en la base de datos	Inicia Sesión

Fuente: Propia.

CONCLUSIONES

La aplicación permitió el fortaleciendo de conocimiento de desarrollo móvil bajo la plataforma Android Studio donde se exploró la búsqueda de métodos útiles que se pudieran incorporar al desarrollo del proyecto y poder adaptarlo a las necesidades requeridas por la persona en condición de discapacidad visual. Como el estudio de documentos digitales para poder ser interpretados con la ayuda de las herramientas que proporciona un teléfono móvil.

La implementación de Firebase en el proyecto fue de gran aprovechamiento ya que cuenta con una variedad de extensiones como Firebase autenticación que trabaja de forma eficaz todo el proceso de logueo en usuarios, y cuenta con el respaldo de grandes corporaciones en el desarrollo de software como Google que transmite confianza y viabilidad de seguir utilizándolo en el futuro para el desarrollo de nuevos proyectos.

La aplicación VisionAccess es una herramienta para usuarios con discapacidad visual que se incorpora en beneficio de las poblaciones más vulnerables y busca fomentar inclusión que deben tener este tipo de personas buscando las mismas posibilidades y oportunidades para realizarse. Independientemente de sus características, habilidades o discapacidades.

CAPITULO 7. REFERENCIAS BIBLIOGRÁFICAS

Agudelo, W. (2001). Las TIC como herramientas potenciadoras de equidad, pertenencia e inclusión educativa, Líder de Programa Programas Educativos Especiales de Secretaría de Educación de Medellín. Medellín - Colombia; willmes2001@yahoo.es

Amaya, B. (2013). Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles.

Agustin A. , Sainz I. , Erro D., Navas E. , . & Hernaez I. (2013). Sistema de Conversión Texto a Voz de Código Abierto Para Lenguas) Ibéricas; University of the Basque Country (UPV/EHU), Bilbao,

Delia, L; Galdámez; Thomas, P & Pesado, P. (2013). Un Análisis Experimental de Tipo de Aplicaciones para Dispositivos Móviles. Disponible: http://sedici.unlp.edu.ar/bitstream/handle/10915/32397/Documento_completo.pdf?sequence=1&isAllowed=y

Espinosa Castañeda, R. y Medellín Castillo, H. I.(2014): Análisis y evaluación de la generación de iconos mentales en personas invidentes a partir de la percepción virtual táctil utilizando realidad virtual y sistemas hápticos, Icono 14, volumen (12), pp. 295-317. doi: 10.7195/ri14.v12i2.695

Grupo de investigación de la universidad técnica de Madrid (2013). Estructura de un conversor texto –voz. Disponible: <http://lorien.die.upm.es/juancho/pfcs/DPF/capitulo3.pdf>

Garay F, Camargo J, Segura D, & Rincón N. (2003). Orientación de Pasajeros con Discapacidad Visual dentro del Sistema de Transporte Masivo Transmilenio, mediante Geolocalización Satelital.

Galdon P. (2014). Mejora en el interfaz de un dispositivo móvil para personas con discapacidad visual mediante el diseño de iconos hápticos. Tesis doctoral de la Universidad de Málaga.

Grupo de investigación de la universidad técnica de Madrid (2013). Estructura de un conversor texto –voz. Disponible: <http://lorien.die.upm.es/juancho/pfcs/DPF/capitulo3.pdf>

Garay F, Camargo J, Segura D, & Rincón N. (2003). Orientación de Pasajeros con Discapacidad Visual dentro del Sistema de Transporte Masivo Transmilenio, mediante Geolocalización Satelital.

Galdon P. (2014). Mejora en el interfaz de un dispositivo móvil para personas con discapacidad visual mediante el diseño de iconos hápticos. Tesis doctoral de la Universidad de Málaga.

Griselda. (2011).Base de datos. Disponible: <http://www.griselda.blogspot.com/2011/03/base-de-datos.html>

Higuera, J. ; Durán, C. & Torres,O. (2014): Scrum: a través de una aplicación móvil ISSN: 2344-8288 Vol. 2 No. 2

ISO International Standard, ISO 9241-11 (1998): Ergonomic requirements for office work with visual display terminals (VDTs.) - Part 11: Guidance on usability.

IBM (2012). El desarrollo de aplicaciones móviles nativas, Web o híbridas. Disponible: ftp://ftp.software.ibm.com/la/documents/gb/commons/27754_IBM_WP_Native_Web_or_hybrid_2846853.pdf

Muñoz, J (2012). Las TIC y la discapacidad visual, Centro de Investigación Desarrollo y Aplicación Tiflotécnica (ONCE-CIDAT) jmsv@once.es

Disponible: <https://development.grupogaratu.com/metodologia-scrum-desarrollo-software/>

Maestro I. (2003). Fomento de la movilidad de invidentes mediante cartografía táctil 2. De la Universidad Politécnica de Valencia, La aspiración fundamental de estos dos artículos es fomentar el interés por la resolución de los problemas de movilidad de las personas discapacitadas desde una perspectiva cartográfica, poco abordada hasta el momento.

Postparaprogramadores. (2018). Que es Firebase? Disponible: <https://postparaprogramadores.com/que-es-firebase/>

Proyectos ágiles (2003). Historia de Scrum. Disponible: <https://proyectosagiles.org/historia-de-scrum/>

Rojas, N; Araya,E. & Vidal, L. (2006). Aporte de tesis “diseño y desarrollo de prototipo de sistema de traducción instantánea de habla y transmisión en tiempo real, sobre el protocolo rtp utilizando tecnologías de reconocimiento de voz”. Universidad Austral de Chile.

Spain Basque Foundation for Science (IKERBASQUE), Bilbao, Spain
{agustin,inaki,derro,eva,inma}@aholab.ehu.es

Sanz, Saucedo, & Torralbo. Introducción a Android. Disponible: <https://openlibra.com/es/book/download/introduccion-a-android>

Servisoftcorp. (2019). Definición y cómo funcionan las aplicaciones móviles. Disponible: <https://www.servisoftcorp.com/definicion-y-como-funcionan-las-aplicaciones-moviles/>

Sims, C., & Johnson, H. (2011). Elements of Scrum. Foster City, California: Dymaxicom.

Sims, C., & Johnson, H. (2012). SCRUM: A breathtakingly brief and agile introduction.

Tiempo de inactividad de la sesión ISO 27002 anexo A 11.5.5, (2015). Disponible: <http://isoedith18.blogspot.com/2015/06/11-control-del-acceso.html>

Ventura R, V; Negrete E; Salinas J, M. (2013). Aprendiendo a desarrollar aplicaciones para Android con la metodología ágil Scrum: un caso de estudio. Disponible: <http://congreso.investiga.fca.unam.mx/docs/xviii/docs/8.01.pdf>

Vintimilla, G (2014). “Desarrollo e implementación de una aplicación móvil que traduzca el abecedario del lenguaje de señas a texto para ayuda a discapacitados auditivos mediante dispositivos móviles Android”, de la Universidad de las fuerzas armadas, Sangolqui-Ecuador, Ingeniería Electrónica en Redes y Comunicación de Datos.

ANEXOS

8.1. Socialización de la Aplicación VisionAccess

En la socialización a las personas con discapacidad visual de la universidad de Pamplona se dio a conocer el desarrollo del proyecto VisionAccess en donde se realizaron entrevistas de tipo verbal basándose en la experiencia de uso de esta aplicación. A continuación se muestra evidencia fotográfica de dicha socialización.



Ilustración Socialización de la Aplicación Móvil con las personas invidentes de la universidad de pamplona

8.2. Evidencia de Grabaciones verbales registradas a las personas con discapacidad visual de la universidad de pamplona.

A continuación se comparte el link para tener acceso a los archivos de grabación tipo encuesta realizados a las personas con discapacidad visual de la universidad de pamplona:

https://drive.google.com/drive/folders/1OxmYR0fqSETZ0hTrRQKhM38_Uw01LUU5?usp=share_link

8.3. Manual de Usuario

8.3.1. Introducción

VisionAccess es una aplicación que sirve como apoyo a las personas con discapacidad visual en la universidad de pamplona. Allí el usuario podrá percibir por vibraciones la exploración en la app además de dar lectura artificial a documentos de tipo pdf a través de un dispositivo móvil.

8.3.2. Requerimientos

Los requerimientos mínimos para que la aplicación VisionAccess funcione correctamente, son los siguientes:

- Sistema operativo Android 6.0 en adelante.
- Conexión a internet.
- Dispositivo con sensor de huella dactilar

8.3.3. Instalación

Una vez que se instala la aplicación en el teléfono, se procede a ubicar el icono de la app, para asegurarnos que se instaló correctamente. Para hacer uso de la aplicación VisionAccess, es necesario tocar el icono para abrirla.

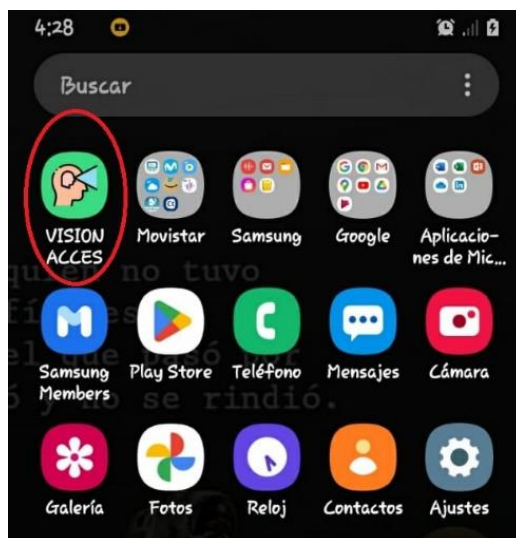


Ilustración 61. Manual usuario – Instalación

Fuente propia

8.3.4. Uso de la aplicación VisionAccess

8.3.4.1. Formulario de Login

Acceder 1 : Digitar correo electrónico, contraseña y tocamos el botón de iniciar sesión. En caso de no tener un correo registrado anteriormente debemos tocar el botón de Registrar Usuario.

Ilustración 62. Manual usuario – Formulario login

Fuente propia

Acceder 2 : una opción cómoda y segura para tener acceso a la aplicación es mediante la autenticación de huella biométrica. Allí el usuario deberá deslizar su huella dactilar por el sensor biométrico del dispositivo móvil.



Ilustración 63. Manual usuario – interfaz acceso con huella

Fuente propia

8.3.4.2. Formulario de Registro Nuevo Usuario

En este formulario el usuario tiene la opción de registrarse, mediante los campos de un correo electrónico y una contraseña. Luego de llenar los datos y para finalizar la acción Deberá oprimir el botón registrar usuario.

Ilustración 64. Manual usuario – registro nuevo usuario

Fuente propia

Si el registro fue exitoso se mostrara y leera un mensaje de bienvenido

De lo contrario y dependiendo de la posible acción errónea que está realizando el usuario ,se le notificaran una serie de mensajes que son leídos por el móvil como los presentados a continuación:

Acción de Campo vacío en contraseña

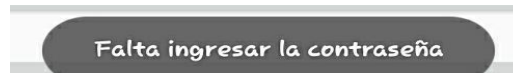


Ilustración 65. Manual usuario – mensaje campo vacío en contraseña

Fuente propia

Acción de Campo vacío en correo



Ilustración 66. Manual usuario – mensaje campo vacío en correo

Fuente propia

Acción de Si digito todos los campos pero al correo le hizo falta el @



Ilustración 67. Manual usuario – mensaje falta carácter

Fuente propia

Acción de digitar contraseña menor a seis dígitos.



Ilustración 68. Manual usuario – mensaje contraseña mayor a 6 dígitos

Fuente propia

Acción de completar todos los datos del formulario, pero el usuario no se encuentra registrado en la base de datos



No se pudo iniciar el usuario

Ilustración 69. Manual usuario – mensaje de error al acceder

Fuente propia

8.3.4.3. Menú principal

Después de haberse identificado en el inicio de sesión exitosamente le dará acceso al menú principal donde podrá hacer uso de la herramienta lectura de documentos al oprimir el botón leer documento pdf. Observar los archivos de tipo pdf y la opción de cerrar sesión.



Ilustración 70. Menú principal

Fuente propia

8.3.4.4. Interfaz Leer Documento

Esta interfaz muestra el botón Leer y el botón Seleccionar Pdf , allí el usuario al oprimir cualquiera de estos dos botones emite una vibración que los diferencia y permite identificar junto con la lectura artificial del nombre de dichos botones ya que complementa la navegación en la aplicación.



Ilustración 71. Interfaz Leer Documento

Fuente propia

8.3.4.5. Acción cargar Pdf

Al usuario realizar la acción de oprimir el botón cargar pdf lo enviara a elegir el documento que desea leer por medio del dispositivo.

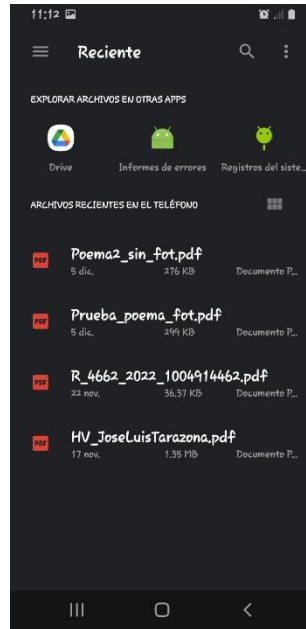


Ilustración 72. Acción cargar Pdf

Fuente propia

Después de seleccionado el documento, se iniciará el proceso de extracción de texto contenido en este y se mostrara en un área de texto junto con la ruta donde se encuentra almacenado.

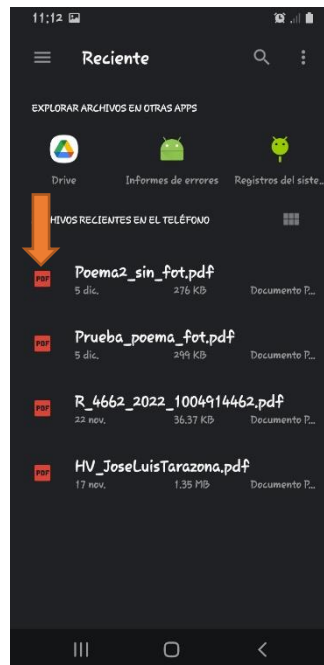


Ilustración 73. Acción seleccionar Pdf

Fuente propia

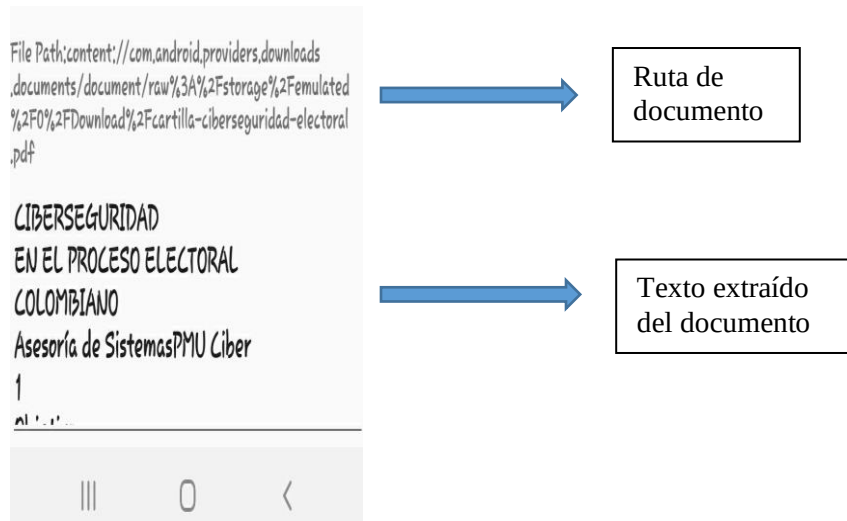


Ilustración 74. Resultado de la acción seleccionar Pdf

Fuente propia

8.3.4.6. Acción leer



Ilustración 75. Interfaz de la acción leer

Fuente propia

Cuando el usuario realiza la acción de oprimir el botón leer inicia el proceso de síntesis del texto donde el dispositivo mediante los motores de voz lee de forma artificial el texto contenido del documento.

8.3.4.4. Acción Cerrar Sesión

Finalmente Cuando el usuario realiza la acción de oprimir el botón cerrar sesión finaliza cualquier proceso que se esté realizando lo trasfiere a la interfaz de inicio de sesión.



Ilustración 76. Interfaz de la acción cerrar sesión

Fuente propia

8.4. Población con Discapacidad Visual a Nivel Nacional y Regional

8.4.1. Población con Discapacidad Visual a Nivel Nacional:

A nivel nacional se cuentan el Instituto Nacional Para Ciegos INCI, Esta organización busca capacitar sobre estrategias pedagógicas para la inclusión educativa de la población ciega y con baja visión.

INCI Realizó su Audiencia Pública de Rendición de Cuentas, en la que se presentaron los avances de la vigencia 2021 en beneficio de los **1.948.332** colombianos ciegos o con baja visión que representan el 43,5 % del total de discapacitados del país.

También se logró antes del inicio de confinamiento por pandemia en año 2019 identificar a los departamentos en donde están ubicadas las personas con discapacidad visual, según el Registro de Localización y Caracterización de Personas con Discapacidad del Ministerio de Salud y Protección Social, Donde a continuación se muestra la clasificación por departamentos.

Tabla 49. Población con discapacidad visual a nivel nacional

Departamento	Cantidad de personas con Discapacidad Visual	Porcentaje
ANTIOQUIA	50.302	10%
ATLÁNTICO	12.457	2%
BOGOTÁ, D.C.	105.559	21%
BOLÍVAR	14.388	3%
BOYACÁ	16.627	3%
CALDAS	8.279	2%
CAQUETÁ	5.635	1%
CAUCA	10.049	2%
CESAR	13.104	3%
CÓRDOBA	11.248	2%
CUNDINAMARCA	17.279	3%
CHOCÓ	1.871	0%

Departamento	Cantidad de personas con Discapacidad Visual	Porcentaje
HUILA	20.571	4%
LA GUAJIRA	6.690	1%
MAGDALENA	15.995	3%
META	7.727	2%
NARIÑO	24.616	5%
NORTE DE SANTANDER	13.535	3%
QUINDIO	8.120	2%
RISARALDA	11.909	2%
SANTANDER	29.487	6%
SUCRE	12.347	2%
TOLIMA	15.980	3%
VALLE DEL CAUCA	52.430	10%
ARAUCA	3.256	1%
CASANARE	6.335	1%
PUTUMAYO	5.072	1%
SAN ANDRÉS, PROVIDENCIA Y SANTA CATALINA	463	0,1%
AMAZONAS	930	0,2%
GUAINÍA	89	0,0%
GUAVIARE	728	0,1%
VAUPÉS	218	0,0%
VICHADA	325	0,1%
NO DEFINIDO	788	0,2%

Fuente: Registro de Localización y Caracterización de Personas con Discapacidad del Ministerio de Salud y Protección Social

Adicionalmente Debido al aislamiento en ese entonces, se implementó una Plataforma de Aprendizaje con ocho cursos dirigidos a maestros, familias y personas con discapacidad. Ese año se desarrollaron dos ediciones de dichas capacitaciones. De igual forma, en el transcurso del año, la Biblioteca Virtual para Ciegos estructuró 579 documentos digitales accesibles y la Imprenta Nacional para Ciegos imprimió 147.917 unidades de libros, cartillas y material para personas con discapacidad visual.

8.4.1.1. Inclusión educativa de las personas con discapacidad

Las mayores brechas del nivel educativo entre la población con y sin discapacidad entre quienes no tienen ningún nivel educativo y en los niveles básicos primarios y media: el 16,2% de las personas en situación de discapacidad no tiene ningún nivel educativo, frente al 2,6% de las personas sin discapacidad.

También el 37,7% de las personas con discapacidad tienen educación básica primaria, mientras que este porcentaje es de 26,9% para las personas sin discapacidad y el porcentaje de personas con discapacidad con educación media (16,5%) es 9,2 puntos porcentuales menores que el de personas sin discapacidad (25,7%). Por otra parte, el 14,3% de las personas con discapacidad tiene educación superior en comparación con el 21,0% de las personas sin discapacidad.

Estas cifras son basadas en registros administrativos, principalmente el Registro de Localización y Caracterización de Personas con Discapacidad (RLCPCD) del Ministerio de Salud y Protección Social, que parte de una visión de registro voluntario acompañado de un diagnóstico médico.

8.4.2. Asociaciones con Discapacidad Visual a Nivel Regional:

A nivel regional, Norte de Santander se cuenta con tres organizaciones de personas en condición de discapacidad visual representada en el siguiente cuadro informativo:

Tabla 50. Organizaciones de discapacidad visual a nivel regional

Nombre	Descripción	Dirección	Forma Jurídica	NIT
Asociación Norte Santandereana de Ciegos (ASOCIEGOS)	Organización que representa a la población invidente a nivel departamental. Donde busca por medio de diversas actividades inclinarse por el bienestar general de su colectividad.	Avenida 2 A 13 06 c Pasaje Moreno en Cúcuta, norte de Santander.	Entidad sin ánimo de lucro	8905020186
Fundación Forjando Empresa Inclusiva (FOREMIN)	Busca apoyar principalmente el desarrollo de negocios de emprendimiento en personas discapacitadas. Cuenta con más de mil invidentes registrados.	Avenida 11AN 11 AE 100 Torres del centenario, Cúcuta, norte de Santander.	Entidad sin ánimo de lucro	9007959063

Asociación	Infunden	capacitación	Calle 37 5 10	Entidad	
Integral	De	sobre	estrategias	Barrio la sabana,	sin
Personas	En	pedagógicas	y de	Los Patios, norte	ánimo de
Condición	De	bienestar	para la	de Santander.	lucro
Discapacidad		inclusión	de la		9010965491
Del Municipio		población ciega y con			
De Los Patios		baja visión bajo la			
(ASINDISCAP)		administración			
		municipal de los patios.			

Fuente propia