# APLICACIÓN PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID PARA USUARIOS CON DISCAPACIDAD VISUAL O CON INTERES EN APRENDER LA COMUNICACIÓN BASADA EN EL SISTEMA BRAILLE.

ERIKA MAYERLI GUZMAN RAMIREZ

JOHAN SEBASTIAN SUAREZ PEREZ

UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS

FACULTAD TECNOLOGICA

TECNOLOGIA EN SISTEMATIZACION DE DATOS

BOGOTA

2015

# APLICACIÓN PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID PARA USUARIOS CON DISCAPACIDAD VISUAL O CON INTERES EN APRENDER LA COMUNICACIÓN BASADA EN EL SISTEMA BRAILLE.

ERIKA MAYERLI GUZMAN RAMIREZ

JOHAN SEBASTIAN SUAREZ PEREZ

Trabajo de grado para optar por el título de Tecnólogo en Sistematización de Datos

Tutor

ING. JOSE VICENTE REYES MOZO

UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS

FACULTAD TECNOLOGICA

TECNOLOGIA EN SISTEMATIZACION DE DATOS

BOGOTA

2015

NOTA DE ACEPTACIÓN

## AGRADECIMIENTOS

A nuestras familias por ser nuestros puntos de apoyo, por su paciencia, amor y comprensión durante la carrera.

Al profesor José Vicente Reyes por su apoyo y asesorías en la elaboración de este proyecto.

Al Instituto Nacional para Ciegos INCI y al Centro de Rehabilitación para Adultos Ciegos CRAC por abrirnos sus puertas, brindarnos su apoyo y colaboración.

A Dios por ser siempre ese sentimiento de alegría, tranquilidad y serenidad en cada momento de esta etapa de vida que esta próxima a culminar y que hoy nos permite estar aquí.

Por ultimo a la Universidad Distrital Francisco José de Caldas pues sin ella no habría sido posible que este proyecto llegara a elaborarse.

“He tenido amigos que han actuado amablemente hacia mí, y ha sido mi buena fortuna causante de tener el poder de darles pruebas substanciales de mi gratitud”.

## DEDICATORIA

A mis padres, a mi mamá por el amor y confianza que ha depositado en mí, por ser mi punto de apoyo, motivación y fortaleza día a día, a mi familia (Mis hermanos, sobrina, mi tía y mi abuela) la fortuna más grande es tenerlos conmigo y el tesoro más valioso son todos y cada uno de ustedes, a mi compañero “Sebastián” por su confianza, paciencia y lo más valioso: su amistad a lo largo de la carrera, a Dios que todo lo hace posible, me llena de bendiciones día a día y que hoy me permite llegar hasta esta etapa.

Erika Mayerli Guzman Ramírez.

A mi tía, a mi papá, a mi mamá, a mis hermanas y hermanos, a mi familia hermosa, a Tania, a Mayerli y a mis queridos amigos; porque nunca uno logra algo de verdadera significancia sin la ayuda de quien sin importar nada siempre está ahí presente, a ustedes va dedicado este, mi primer gran logro.

Johan Sebastián Suarez Pérez.

**TABLA DE CONTENIDO.**

1 INTRODUCCIÓN: ..... 9
1.1 Titulo: ..... 9
1.2 Tema: ..... 9
1.3 PLANTEAMIENTO DEL PROBLEMA: ..... 9
1.3.1 Descripción: ..... 9
1.3.2 Formulación del problema: ..... 10
1.4 ALCANCES Y DELIMITACIONES ..... 10
1.4.1 Alcances ..... 10
1.4.2 Delimitaciones ..... 11
1.5 OBJETIVOS ..... 11
1.5.1 Objetivo General. ..... 11
1.5.2 Objetivos Específicos. ..... 12
1.6 JUSTIFICACIÓN. ..... 12
1.7 MARCO DE REFERENCIA ..... 13
1.7.1 MARCO HISTÓRICO ..... 13
1.7.2 MARCO TEÓRICO: ..... 18
- SISTEMA BRAILLE: ..... 18
- LIMITACIÓN VISUAL: ..... 19
- RECONOCIMIENTO TÁCTIL: ..... 20
- HABILIDAD MOTORA: ..... 20
- ENSEÑANAZA DEL SISTEMA BRAILLE: ..... 21
- SINTETIZADOR DE VOZ: ..... 24
- TIFLOTECNOLOGÍA: ..... 24
- LA LECTURA BRAILLE: ..... 25
- INCLUSIÓN SOCIAL: ..... 27

CICLO DE VIDA DE XP .....27
VALORES DE XP .....29
PRÁCTICAS DE LA XP: .....30
PRUEBAS .....34
1.7.3 MARCO CONCEPTUAL .....35
Aplicación móvil .....35
Android .....36
Java .....36
Android SDK manager .....36
JDK .....36
Eclipse .....37
TTS (Text to Speech-Texto a Voz) .....37
SQLite .....37
1.7.4 MARCO METODOLÓGICO .....38
1.7.4.1 SOLUCIÓN TECNOLÓGICA .....38
1.8 FACTIBILIDAD .....40
1.8.1 Factibilidad Técnica .....40
1.8.2 Factibilidad Operativa .....41
1.8.3 Factibilidad Económica .....43
1.8.4 Factibilidad Legal .....44
1.9 CRONOGRAMA. .....45
2 FASE DE ANÁLISIS .....46
2.1 Consulta a fuentes de información apropiadas e identificación de problemas. 46
2.1.2 Análisis de alternativas de solución. .....47
2.1.3 Establecimiento del papel del computador .....48
2.1.4 Selección de la Herramienta o lenguajes de Programación. .....48

2.2 DISEÑO.....50
2.2.1 Entorno para el diseño del MEC. .....50
2.2.2 Diseño.....52
Alcance: .....52
2.2.3 Diseño de Interfaz. .....53
2.2.4 Diseño computacional.....54
2.2.5 Preparación y evaluación del prototipo. .....54
2.2.6 FASE DE IMPLEMENTACIÓN. .....55
TARJETAS CRC (Clase - Responsabilidad – Colaborador). .....55
3. FASE DE PRUEBA PILOTO. .....66
4. FASE DE IMPLANTACIÓN. .....68
5. METODOLOGÍA.....69
Metodología de Desarrollo .....69
5.1 PLANEACIÓN.....69
5.2 DISEÑO .....71
5.3 CODIFICACION.....73
5.4 PRUEBAS.....74
5.5 ROLES:.....76
HISTORIAS DE USUARIO.....77
CONCLUSIONES.....99
RECOMENDACIONES. .....100
BIBLIOGRAFIA. .....101
INFOGRAFIA. .....102

# 1. INTRODUCCIÓN:

## 1.1 Titulo:

Aplicación para dispositivos móviles con sistema operativo Android para usuarios con discapacidad visual o con interés en aprender la comunicación basada en el sistema Braille.

## 1.2 Tema:

Este proyecto propone el desarrollo de una aplicación móvil para ayudar a los usuarios que tienen discapacidad visual como a usuarios interesados en aprender a manejar el sistema Braille mediante herramientas como respuestas por voz, vibración y teclado diseñado específicamente para el sistema Braille haciendo la comunicación muy natural y sencilla. Todo esto con los recursos móviles, en lenguaje java sobre Android.

## 1.3 PLANTEAMIENTO DEL PROBLEMA:

### 1.3.1 Descripción:

Hoy en día los dispositivos móviles han dejado de ser exclusivamente para recibir y enviar llamadas ha sido adaptado para realizar múltiples tareas. A medida que aumenta la tecnología en los dispositivos, aparecen nuevas generaciones de usuarios que alcanzan cada vez más afinidad con el auge de estas. Sin embargo para la población con discapacidades visuales resulta mucho más complicado a tal grado que se encuentran muy apartados de dichas tecnologías; por esta razón surge la necesidad de hacer que estas personas sientan que aunque el mundo va creciendo en tecnología, ellos también hacen parte de él y que esta se puede adaptar a sus necesidades, disminuyendo la brecha de desigualdad con otras personas sin discapacidad.

La Organización Mundial de la Salud el 9 de octubre en conmemoración al día mundial de la salud visual, publicó en su revista, que alrededor de 285 millones de personas sufren algún tipo de discapacidad visual en el mundo, al menos 39 millones son ciegas y 246 tienen baja visión, en Colombia 1,3 millones de colombianos no pueden acceder a las tecnologías debido su discapacidad visual.

En Colombia el gobierno nacional quiere eliminar este abismo entre la discapacidad visual y el acceso a la tecnología ya que se encuentra bastante adelantado en el legislativo, el *Proyecto de ley* estatutaria número 169 de 2014 Senado que entre otras

cosas, busca que en todos los establecimientos educativos públicos, bibliotecas, centros culturales, aeropuertos, terminales de transporte, cárceles y empresas sociales del Estado, se disponga de un “software lector de pantalla”, con los que las personas con discapacidad visual podrían hacer uso de las nuevas tecnologías. Según expreso el periódico el tiempo el pasado 13 de septiembre de 2014.

En la actualidad existen aplicaciones bajo Android como talkback diseñado por google que tiene como finalidad describir cada una de las funciones y aplicaciones del móvil, mediante respuestas por voz, en cierto sentido este proyecto ira guiado por este principio.

“Para muchas personas, la tecnología hace las cosas más fáciles. Para las personas con impedimentos, la tecnología hace las cosas posible” publicado por (IBM; 1991, p. 2). Con esta frase se hace una idea de lo que se quiere lograr. En definitiva, usar un teléfono móvil teniendo problemas de visión puede ser todo un desafío y este proyecto asume estas dificultades incorporado cambios muy relevantes en cuanto a la mejoras en la accesibilidad desarrollando métodos eficaces de lectura y escritura basado en el idioma Braille emulando un teclado con el cual se pueda realizar escritura y un sistema inteligente que procese el texto además de un módulo de entrenamiento para aquellos que quieren aprender este sistema con múltiples ayudas.

### 1.3.2 Formulación del problema:

¿Se puede realizar un software en el cual se pueda incluir a los usuarios con discapacidad visual a las tecnologías móviles de manera muy natural y autónoma, además de proporcionar entrenamiento a personas con la intención de aprender el sistema Braille?

## 1.4 ALCANCES Y DELIMITACIONES

### 1.4.1 Alcances

Este proyecto se centra en desarrollar una forma de comunicación escrita para población con discapacidad visual mediante la tecnología móvil.

En primer lugar. Se encuentra el módulo de escritura y lectura

- Se desarrolla un teclado para las personas con discapacidad visual, que cuenta con 6 puntos que implementa la sintaxis del sistema Braille.
- Con el teclado desarrollado el usuario con discapacidad visual, podrá escribir cualquier tipo de texto mediante el principio de la sintaxis punteada del sistema braille.

- Con la implementación de la librería TTS el usuario con discapacidad visual podrá saber qué y cómo está escribiendo su texto.

En segundo lugar. Se encuentra el módulo de aprendizaje

- Para el usuario común se desarrollan actividades de entretenimiento como juegos y trivias para la práctica del conocimiento del sistema Braille.
- Se desarrolla manuales de usuario en PDF y mediante voz para ambos usuarios.
- Se desarrolla teclados de prueba simulando el sistema Braille para el usuario común.

En tercer lugar. Se encuentra el módulo de evaluación

- Se desarrolla test en los que se evalúan los conocimientos adquiridos.

### 1.4.2 Delimitaciones

- Es necesario el desarrollo de manuales en PDF y mediante voz.
- Se deberá tomar en cuenta las características de hardware en las que se ejecute el proyecto.
- Es necesario la implementación de aplicaciones existentes como talk back ofrecida por google a dispositivos móviles, con sistema operativo Android.
- Es necesario que el proyecto trabaje de la mano con la librería extra para el sintetizador de voz TTS.
- El aplicativo es solo para dispositivos con sistema operativo Android.
- Se desarrolla una interfaz de video que expone todo el conocimiento del sistema Braille (tutoriales).

## 1.5 OBJETIVOS

### 1.5.1 Objetivo General.

Desarrollar una aplicación móvil para el aprendizaje y práctica del sistema Braille.

### 1.5.2 Objetivos Específicos.

- Recopilar y analizar la información correspondiente al sistema Braille.
- Diseñar un teclado que homologue el modo de escritura del sistema Braille.
- Implementar la librería TTS con el teclado, para lograr una ubicación espacial sobre este.
- Diseñar la base de datos que contendrá toda la información del sistema Braille, además de imágenes y sonidos.
- Implementar el método de enseñanza Bliseo para aplicar teoría del sistema Braille.
- Desarrollar interfaz de videos, para contribuir al proceso de enseñanza.
- Desarrollar interfaces graficas que contengan los juegos, test y actividades de entretenimiento a través de teclados, para aplicar el conocimiento del sistema Braille con el usuario interesado en aprenderlo.
- Desarrollar actividades para evaluar el proceso de aprendizaje.

## 1.6 JUSTIFICACIÓN.

"Para muchas personas, la tecnología hace las cosas más fáciles. Para las personas con impedimentos, la tecnología hace las cosas posible" IBM 1991. La tecnología ha resuelto muchos de los problemas de la humanidad, incluso en algunos casos ha generado cierta dependencia. Pero ¿qué ocurre con aquella población que no puede acceder a esta tecnología por impedimentos visuales?

Una de las principales dificultades que enfrentan los invidentes en Bogotá, más allá de los recursos o herramientas para su desarrollo, es "que falta mucho conocimiento de las demás personas, porque en realidad la inclusión no se da", asegura Verónica Gómez, estudiante invidente de la UPN y campeona nacional de natación, pues es usual que las personas videntes los excluyan de ciertos lugares, como las mismas aulas de clase, al considerarlos incapaces.

Tras la necesidad de introducir a la evolución tecnológica a las personas con discapacidad visual. Es de prioridad desarrollar un software que permita a dichas personas hacer mucho más sencilla la comunicación, entiéndase como vía de comunicación la lectura y la escritura de mensajes, archivos, documentos y todo lo relacionado con comunicación escrita.

Con la ayuda de un sintetizador de audio que se encargara de simular la voz humana, el software le permitirá al usuario acceder a toda la información que desee, mediante un teclado que se utiliza de tal manera que generaliza cualquier conocimiento del sistema braille ya que el teclado simula cómodamente su sintaxis de punteado. Se podrá integrar tanto la escritura como la lectura de manera conjunta, además de contar con un módulo de aprendizaje que le permitirá al usuario comprender y manejar correctamente el sistema de manera didáctica mediante juegos, videos, manuales etc. Como medio para desarrollar dicho software se utilizara el lenguaje java en el sistema operativo Android.

Las personas beneficiadas por el proyecto podrán romper esa barrera entre el uso de la comunicación escrita y el uso de la tecnología. Existen proyectos para aportar al cómodo manejo de la tecnología con las personas con discapacidad visual como shades, on the bus, móvil accesibilty y se centran en que el usuario se guie en cómo manejar su móvil o para ubicar el dispositivo y que quede listo para su uso e incluso recientemente salió un software que permite vincular un hardware al móvil mediante el bluetooth para la escritura, es decir ya se ha trabajado en el tema, pero no se ha pensado en la facilidad de conseguir el hardware necesario para su correcto funcionamiento.

## 1.7 MARCO DE REFERENCIA

### 1.7.1 MARCO HISTÓRICO

Para el desarrollo se consultaron diferentes fuentes de información, para buscar herramientas similares a la que se desarrolló, esta búsqueda arrojo la siguiente información:

1. **Dispositivo traductor de textos digitales al lenguaje Braille para ciegos como proyecto de emprendedurismo social. Autor Luis Hernández Ureña. ULACIT (Universidad Latino Americana de Ciencia y Tecnología). Abril 2014:** El proyecto fue concebido por un grupo de jóvenes egresados del CEDES Don Bosco quienes iniciaron la idea en 2007 en el marco de la EXPOTEC. La idea principal de este estudio era la de beneficiar a los centros educativos, las instituciones y las bibliotecas con el fin de colaborarles a las personas con discapacidad visual, así como establecer las características que un dispositivo traductor textos digitales al lenguaje Braille que se debe tener para que sea un proyecto social.

   El proyecto básicamente busca desarrollar un sistema en el cual desde un dispositivo (PC, celular, tableta, portátil) se pueda acceder a cualquier tipo de texto y posteriormente este sea traducido a un tipo de texto basado en el sistema braille

esto con el fin de que el usuario pueda acceder a este mediante la lectura tradicional del sistema braille.

Dentro de los estudios que se llevaron a cabo, se realizaron encuestas a una población de 50 personas de 29 mujeres y 21 hombres que contaba con problemas visuales elevados o totalmente. En la cual se encontraban preguntas como: Cuál era su nivel de escolaridad, la cual arrojo resultados del total de encuestados, la gran mayoría seleccionó como su nivel de escolaridad la universitaria con un total de 52% seguido del nivel profesional con un 28%, así mismo en 3er lugar, se encuentra secundaria con un 28%. En los últimos campos se tiene el rubro Otros (cursos libres y de idiomas) con 20% y un 2% con primaria.

La encuesta muestra los tipos de dispositivos que los usuarios tienden a utilizar regularmente, siendo la computadora de escritorio la más utilizada con un 52%, seguido muy de cerca por parte de los computadores portátiles con un 50% respectivamente, la razón principal, según los encuestados tienen la ventaja de ser más manejables respecto de los software que pueden ejecutar, son los más comunes en ambientes de trabajo y estudio, además de que los encuestados se sienten más familiarizados con su uso, por otra parte, se tiene los dispositivos móviles de los cuales el Smartphone presenta un 46%, la Tableta con un 26%, su ventaja radica en que pueden ser utilizados en cualquier parte, tienen funciones básicas, además que son livianos y fáciles de transportar.

A demás dentro de las preguntas hechas a los encuestados se preguntó con qué frecuencia utilizaban estos aparatos al mes y los porcentajes de uso de los dispositivos muestran indiscutiblemente un uso elevado por parte de los consumidores, siendo un 74% la opción de 15 veces o más en un mes, demostrando el uso intenso por parte de las personas ciegas, seguido en un número más modesto 22% de 11 a 15 veces por mes y finalmente de 1 a 5 veces por mes y de 6 a 10 veces por mes 2%, respectivamente.

Finalmente se les pregunto y se armó en una escala qué tanto están dispuestos a invertir en este tipo de dispositivos, siendo muy poco la opción con más porcentaje (60%) esto a que los precios a estos dispositivos que, actualmente, se ofrecen en el mercado son excesivamente caros, seguido de un 34% que escogieron invertir algo y en las últimas posiciones se encuentra que 4% invertirán mucho y casi nada con un 2%.

Este proyecto de investigación realizado desde la ULACIT abarca una idea de traductor de lenguaje a sistema braille utilizando varios recursos de hardware sin especificación pero tiene una gran desventaja la cual radica en que para

entregarle los resultados al usuario y que él pueda acceder a ellos, necesita de otro tipo de hardware adicional, debido a su condición de discapacidad visual.

El presente proyecto que se desea realizar abarca varios componentes importantes como lo son la lecto-escritura en la cual el usuario podrá escribir y acceder a los escritos mediante la librería sintetizadora de voz TTS, transformando los conocimientos del sistema braille en el lenguaje español como tal, no como propone el proyecto de investigación de la ULACIT de hacer una traducción del español al sistema braille, ya que esto limita la funcionalidad de la aplicación. A diferencia del proyecto de investigación, el presente proyecto no necesita de otro hardware para su correcto funcionamiento

2. **Nela: Aprende a escribir usando Braille Universidad Zaragoza. Trabajo de grado de Enrique Matías Sánchez, Febrero 2013:** Nela es un software que ayuda en el aprendizaje de la escritura usando el código Braille a los niños con discapacidad visual (visión reducida o ceguera).
Es multiplataforma (funciona tanto en Microsoft Windows como sobre sistemas de tipo Unix como Linux) y esta internacionalizado (preparado para traducirse a otros idiomas).

Nela se basa principalmente en el método presentado en el libro El Braille en la escuela. Una guía práctica para la enseñanza del Braille. Usando el teclado normal del ordenador, Nela simula una maquina Perkins, una maquina Perkins viene a ser una especie de máquina de escribir para ciegos, que solo tiene 6 teclas (una para cada uno de los seis puntos del cajetín Braille) además del espaciador, etc. El niño colocara las manos sobre la central del teclado, con los dedos índices sobre las teclas F y J, que tienen unas marcas en relieve. Si el niño tiene dificultades para notar ese relieve, se puede pegar un trozo de filtro sobre las teclas. Las teclas SDF y JKL se usan como si fueran una maquina Perkins.

Con los conocimientos del método utilizado para la enseñanza, la primera parte del software está encargada de hacer una inducción al usuario del funcionamiento de este mediante una interfaz con sintetizador de voz que básicamente lo introduce en la idea conceptual de la maquina Perkins, la segunda parte del programa tiene como misión transmitir el conocimiento del sistema braille basado en las reglas incluidas en el libro El Braille en la escuela, que tiene reglas como que en una primera etapa del aprendizaje, el niño deberá conocer un conjunto de palabras que contengan letras de estructura espacial simple. Tal es el caso de las letras que llevan puntos en una misma dirección o que en los extremos o que las

letras que llevan los puntos en la misma dirección se perciben más rápidamente, todo esto descrito paso a paso en el libro ya mencionado.

A diferencia del proyecto NELA, el presente proyecto busca enseñar el sistema braille a personas sin discapacidad visual pero utilizando la misma base ya que contaría con una interfaz gráfica en conjunto con un sintetizador de voz, que en este caso sería el TTS del SDK manager.

El presente proyecto va a trabajarse bajo el sistema operativo Android y el proyecto NELA puede manejarse por el usuario que tenga como recursos algunos de los sistemas operativos como Windows, Linux y UNIX, por lo que el presente proyecto tiene la diferencia de que va orientado hacia dispositivos móviles, transmitirá el conocimiento al usuario sobre el sistema braille con aplicaciones de entretenimiento como juegos de palabras, pero contará con un módulo en el que se podrá aplicar dicho conocimiento de forma en que el usuario podrá desarrollar escritos.

3. **JAWS 11.0 Como estrategia metodológica de inclusión educativa interactiva para la enseñanza a personas con discapacidad visual del centro de educación popular especial, 15 de abril del cantón Zaruma en el periodo 2011-2012, Universidad técnica de Machala del Ing. Polo Cacao Tito Javier:** El fin del proyecto es utilizar las jaws, que en definitiva vienen a ser una serie de comandos que tienen como fin sintetizar mediante voz, todo tipo de texto proveniente del paquete de office, internet entre otros. Además de controles básicos para el correcto manejo del PC, para incluir a todas las personas con discapacidad visual a las tecnologías de tal forma que se proporcionará al docente las instrucciones básicas para la Instalación del Jaws 11.0 en su equipo.
En estas circunstancias los contenidos a estudiarse en el módulo son un valioso apoyo para el invidente puesto que le permitirá interactuar y capacitarse para su inclusión social. El fin de utilizar a las jaws para que el usuario con discapacidad visual pudiese realizar nuevos documentos, abrir, guardar con contenidos que capaciten al estudiante invidente sobre lo que significa una computadora, como encender y apagar, escuchar el lector de pantalla y práctica sobre la correcta posición frente a la máquina entre otros, con el fin de familiarizarse para seguir las instrucciones y su aplicación futura, así mismo se capacitará para que en un futuro pueda acceder a una normal inclusión social como las personas sin ninguna discapacidad.

Dentro de las preguntas que se realizaron para la captura de datos fue la de ¿Qué estrategias metodológicas aplica usted en el desarrollo de los aprendizajes del sistema jaws 11.0? Hecha a 20 docentes la cual arrojo como resultados que el 59,8% contestaron que la mejor estrategia metodológica es la de formación de grupos, mientras que en el segundo lugar el 18,8% escogieron el trabajo individual, lo que significa que es viable utilizar las jaws como método de enseñanza.

El presente trabajo que se desea realizar utiliza el concepto del proyecto de las jaws 11.0 ya que utiliza el método de instrucciones mediante sintetizadores de voz para el manejo de un sistema y el de creación de documentos y edición de ellos, pero como diferencia tiene que el proyecto basado en jaws trata de que el usuario pueda desarrollar dichos documentos mediante el lenguaje español con metodología orientada a personas con discapacidad visual, y la intención del proyecto en proceso es desarrollar todo con conocimientos del sistema braille, es decir la escritura de documentos y edición de estos.

4. **Brailleteach, Universidad Santo Tomás Santander, Trabajo de grado Estudiantes de ingeniería mecatrónica Yulieth Pabón Moreno y Andrea Medina Ahumada:** El proyecto se basa en facilitar el aprendizaje de la lectoescritura del Braille a través de juegos y actividades entretenidas. El dispositivo consta de una caja electrónica con dos teclados, uno de lectura y otro de escritura, en el primero el alumno posiciona sus manos sobre este para sentir una combinación que le servirá para luego plasmarla en el otro teclado y en el otro el usuario debe oprimir los caracteres que sintió, para que posteriormente el sistema reconozca sí estuvo correcta o incorrecta.

   La caja tiene un plug USB que va conectado al computador para poder acceder a los recursos de software del proyecto, para el proyecto es de vital importancia que al usuario lo supervise durante las primeras sesiones un tutor, debido a que esto hace que las primeras experiencias del alumno sean agradables y posteriormente sea mayor la adaptación, el tutor podrá visualizar una interfaz gráfica que le permitirá avisarle al alumno sí estuvo o no correctamente y donde se localizó el error.

   Este proyecto obtuvo equipo proporcionado por parte del Tecno parque del Sena de Santander el cual les permitió desarrollar prototipos en 3D para desarrollar el hardware.
   El proyecto tuvo gran acogida en su población pronunciado por el coordinador del Comité Departamental de Discapacidad en Santander, Teófilo Rincón Mancilla, quien promovió el proyecto destacando la importancia de incluir a las personas en condición de discapacidad visual a las tecnologías, propiciando a que mejore la calidad de vida de dicha población. El proyecto Brailleteach fue probado sobre la población estudiantil de la Escuela Normal Superior de Piedecuesta y de la Escuela Taller para Ciegos de Bucaramanga.

   Una de las desventajas que posee el proyecto Brailleteach según puntualizo el instructor de la Escuela Taller para Ciegos de Bucaramanga Henry Alberto Sierra Saba, es que el programa va orientado a afianzar los conocimientos del sistema braille, al ser un método rígido y con exactitud, no permite el error en sus actividades de entretenimiento, por lo que el estudiante aprende mediante prueba

y error lo que hace mucho más eficiente la enseñanza con los métodos tradicionales.

Es claramente visible que el proyecto Brailleteach, resulta atractivo debido a sus actividades, con las cuales se les proporciona el conocimiento, pero carece de una forma práctica de enseñanza del manejo del software ya que es necesario de un tutor que esté pendiente durante los primeros pasos de la experiencia del software.

### 1.7.2 MARCO TEÓRICO:

- **SISTEMA BRAILLE:**

Es un sistema de lectura y escritura en relieve, inventado en el siglo XIX por el francés Louis Braille, se fundamenta en la combinación de seis puntos colocados en forma vertical, tres a la derecha y tres a la izquierda, que se conocen con el nombre de signo generador (es la unidad básica del sistema Braille, la persona con discapacidad visual debe manejarlo y familiarizarse con él en todas las formas y materiales posibles) y pueden ser distinguidos a través del tacto1. La combinación de estos seis puntos da sesenta y cuatro caracteres que permiten la escritura de las letras del alfabeto, signos de puntuación, los números y signos matemáticos.

Existe una asociación entre el conocimiento y uso del Braille y el grado de deficiencia visual y la existencia o no de otra deficiencia o limitación personal unida a la visual. Quienes más conocen y usan el Braille son los totalmente ciegos o que solo perciben luz, más que los que distinguen formas de objetos u objetos a una cierta distancia2.

**Alfabeto:**

- Los puntos negros pequeños son los puntos del signo generador que no están en relieve, sólo se usan para una mejor comprensión de cada símbolo.
- Los símbolos correspondientes a la primera fila ocupan sólo los cuatro puntos superiores del signo generador.

---

1 *http://www.inci.gov.co/inciparaninos/palabras-y-cuentas/sistema-de-lectoescritura-braille*

2 *http://www.giro.infor.uva.es/proyectos/analisisComponentesSWDiscapacitados.pdf*

- En la segunda fila son iguales a los de la primera, pero se le agrega el punto inferior izquierdo (menos en la letra "ñ" que es propia del idioma español).
- En los de la tercera se agregan los dos inferiores.

**Vocales acentuadas:**

Como no era posible colocar una tilde encima de los puntos correspondientes a estas vocales, se inventó un nuevo símbolo para cada una, que no suele coincidir para nada con la misma vocal sin tilde dentro del Alfabeto Braille.

**Símbolos Dobles:**

Tomando en cuenta los símbolos resultantes de los 6 puntos del signo generador, se inventaron los llamados símbolos dobles, que dan un nuevo significado a la letra que anteceden.

- El primer símbolo doble es el que antecede a otra letra para anunciar que se trata de una mayúscula.
- El otro símbolo doble es el que antecede a una letra para representar un número.

**Números:**
Los números se forman utilizando las primeras letras del alfabeto, de la (a) a la (j) (números 1 al 0), precedidas por el signo numerador.

**Signos de puntuación:**

En los signos de puntuación correspondientes a la admiración, interrogación y comillas no se diferencian los símbolos de abrir y cerrar3.

- **LIMITACIÓN VISUAL:**

Se refiere a condiciones caracterizadas por una limitación total o muy seria de la función visual. La limitación visual se puede presentar por factores internos y externos.

---

3*http://www.discapnet.es/Castellano/areastematicas/Accesibilidad/accesibilidadcomunicacion/lenguajebraille/Documents/Guias/Braille/alfabeto_braille_2.htm*

- Factores internos son los relacionados con la herencia que se transmite de los padres o por alteraciones genéticas.
- Factores externos: Los factores externos son aquellos que están en el medio ambiente como la exposición a radiaciones, sustancias tóxicas o procedimientos riesgosos en el momento del parto.

Se habla de personas con ceguera para referirse a aquellas que no ven nada en absoluto o solamente tienen una ligera percepción de luz (pueden ser capaces de distinguir entre luz y oscuridad, pero no la forma de los objetos), cuando se habla de personas con deficiencia visual son los que pueden ver o distinguir, aunque con gran dificultad, algunos objetos a una distancia muy corta, con un considerable esfuerzo y utilizando ayudas especiales.

Las personas con deficiencia visual, a diferencia de aquellas con ceguera, conservan todavía un resto de visión útil para su vida diaria4.

- **RECONOCIMIENTO TÁCTIL:**

El reconocimiento táctil es la forma en que las personas con discapacidad visual son capaces de recolectar información por sus canales sensoriales como oído, olfato o gusto.
El canal táctil, juega un papel importante, aunque no indispensable, en el proceso de aprendizaje de la lectoescritura Braille. Según el INCI (Instituto nacional para ciegos) en gran medida la construcción del mundo la hace la persona con limitación visual a partir del tacto, sentido que aprende a manejar con mayor propiedad si posee un ambiente rico en experiencias, que en este caso van desde el inicial reconocimiento de su cuerpo y el de otras personas, hasta el reconocimiento de los objetos cotidianos, las frutas y los animales.

- **HABILIDAD MOTORA:**

Los niños y niñas con discapacidad visual deben desarrollar algunas habilidades básicas que, sin ser diferentes a las que desarrollan otros niños y niñas, sí requieren un poco más de atención. Dentro de estas habilidades podemos mencionar:

---

4 *http://inclusionlimitacionvisual.blogspot.com.co/2012/11/lo-que-debemos-saber-de-la-discapacidad.html*

- Conocer y utilizar su cuerpo como una posibilidad o medio para aprender.
- Conocer y acercarse al cuerpo de los demás de manera respetuosa.
- Manejar sus dos manos como herramientas exploratorias de modo que le permitan conocer los objetos a su alrededor.
- Ubicarse y moverse en el espacio de manera segura.
- Utilizar un lenguaje comprensivo y expresivo como medio fundamental de comunicación.

El desarrollo motor y la habilidad de manejar el cuerpo para entablar relaciones con personas y distinguir objetos, juegan un papel muy importante en el proceso de aprendizaje; en el caso de los niños o niñas con limitación visual, el cuerpo es el instrumento más importante que poseen para conocer.
Por esta razón, estimular su desarrollo corporal conducirá a una lectura más completa del mundo y beneficiará el proceso de lectoescritura en cualquier sistema, incluido el Braille5.

El niño o niña con limitación visual, necesita que se le proporcionen la mayor cantidad de experiencias adecuadas para desarrollar conceptos tales como arriba, abajo, izquierda, derecha, cerca de, lejos de, encima, debajo, rápido, lento, etc.

- **ENSEÑANAZA DEL SISTEMA BRAILLE:**

Para la enseñanza del sistema braille, se deben tener en cuenta tres áreas importantes a desarrollar en la persona con discapacidad visua6l:

- Desarrollo Motor:

Son movimientos para lograr la flexibilidad en la muñeca y los dedos, se lleva a cabo por medio de actividades como la exploración de objetos tridimensionales, clasificación de tamaños (Grande-Pequeño), de esta manera también se desarrolla el lenguaje.

- Destrezas auditivas:

Las experiencias auditivas son de gran ayuda para la enseñanza del braille, para ellas es recomendable; repetir oraciones breves, aprender canciones cortas, escuchar y obedecer órdenes, para que así se desarrolle un lenguaje con significado, basado en experiencia.

---

5*http://www.icbf.gov.co/portal/page/portal/PortalICBF/RecursosMultimedia/Publicaciones/Editoriales1/CARTILLA-VISUAL-2.pdf*

6*http://www.icbf.gov.co/portal/page/portal/PortalICBF/RecursosMultimedia/Publicaciones/Editoriales1/CARTILLA-VISUAL-2.pdf*

- Desarrollo del Lenguaje.

**Métodos de enseñanza:**

Una vez conseguidas la capacidad táctil y destrezas generales, se puede empezar a abordar la enseñanza del sistema braille, los métodos más utilizados son7:

- Alborada:

Cartilla para el aprendizaje de la lectura. Presenta las letras en un orden lógico, con frases de creciente complejidad, resulta un método fácil de utilizar y motivador para los alumnos adultos ya que, desde las primeras páginas, leen palabras y frases con significado. El orden de presentación de las letras tiene en cuenta la sencillez o complejidad de los signos: a, o, u, e, l, p, i, b, m, s, n, v, d, ñ, g, t, f, ll, r, c, y, j, q, h, z, x, ch, k, punto, signo de mayúscula, sílabas trabadas, á, é, ó, coma, punto y coma, dos puntos, guion, í, ú, ü, w, interrogación, admiración y signo de número.

- Bliseo:

Es un método para aprendizaje del sistema braille para adultos alfabetizados. Empieza profundizando en el conocimiento especial del signo generador y va introduciendo las letras de la primera serie (de la «a» a la «j»), para seguir con la siguiente serie, añadiendo el punto 3 (de la «k» a la «t», excepto la «ñ») y las 5 últimas letras, añadiendo el punto 6.

- Pérgamo:

Método de alfabetización para personas ciegas adultas. Realiza una presentación de letras pensada para evitar confusiones y facilitar la percepción en braille. El orden de presentación de las letras es el siguiente: a, e, i, o, u, l, s, p, m, f, d, n, t, ñ, c, h, á, é, b, v, ll, y, r, í, ó, ú, g, j, z, mayúsculas, punto y coma. Al final, se introducen las letras que suelen aparecer menos: x, q, ch, k, w, ü. Después van apareciendo las sílabas trabadas (pl, cl, bl, dr, tr, etc.), el signo de número y signos de puntuación: guion, dos puntos, punto y coma, interrogación, admiración, comillas, paréntesis, etc.

---

7*http://www.ite.educacion.es/formacion/materiales/129/cd/unidad_5/m5_metodo_ensenanza.htm*

- Punto a punto:

Se presenta en castellano y catalán. Consta de dos series de 5 y 4 tomos, respectivamente. La primera serie presenta un programa de prelectura y pre escritura y la segunda se dedica a la enseñanza del sistema braille.

Primera serie: en los tres primeros tomos se ofrecen una serie de ejercicios de prelectura, para el reconocimiento de formas (cuadrado, círculo, triángulo, rectángulo) y tamaños, líneas horizontales y verticales, y seguimiento de líneas y orientación espacial con cuadrados, líneas y series de varios elementos (conjuntos de puntos), para proceder a su discriminación, aun sin darles un significado. En el cuarto tomo es cuando se inicia el aprendizaje del sistema braille: el signo generador, en grandes caracteres con el objetivo de ir disminuyendo el tamaño e ir reconociendo las diferentes posiciones y las primeras letras: a, b, l, e, o. También se comienza la pre escritura. En el último tomo se hacen consideraciones metodológicas para el profesor.

Segunda serie: en los tres primeros tomos, se van presentando una a una todas las letras del alfabeto, se hacen ejercicios de reconocimiento táctil, identificación y discriminación, combinando con las letras ya sabidas, para pasar a la lectura de sílabas, palabras y frases. El orden de presentación de las letras es el siguiente: a, o, u, e, l, p, b, m, n, f, i, signo de mayúscula y punto, r, s, apóstrofe, t, ll, c, admiración, d, interrogación, g, j, á, í, ú, v, coma, x, h, q, punto y coma, ñ, z, dos puntos, é, ó, ü, t y k; (en la versión en catalán, se añaden las letras è, ò, ç, ï) El cuarto tomo está dedicado a los maestros, donde se les explica en qué consiste el método y cómo utilizarlo. Se acompaña de dibujos en relieve para motivar al alumno y de ejercicios para reforzar la discriminación de las letras.

- Tomillo:

Es un método de iniciación a la lectura braille dirigido, especialmente, a la población infantil. Apoya la presentación de contenidos significativos, al mismo tiempo que respeta las peculiaridades de la exploración táctil. Además, se adecua a la edad a la que va dirigido, utilizándose palabras y frases cortas con sentido, con estructuras lingüísticas familiares para el niño. Se emplean materiales atractivos para estas edades, con representaciones en relieve. Además ofrece ejercicios. La secuencia de presentación de grafemas es: a, o, u, e, l, p, á, b, c, d, m, signo de mayúscula, punto, i, n v, ó, s g, t, f, r í, ll, j, z, ñ, é, h, y, ch, ú, q, rr, r, gu. Es decir, se presentan, en primer lugar, las letras que se perciben más fácilmente al tacto, las que no presentan dificultades fonéticas y se evita unir letras simétricas. Se van introduciendo desde el principio las vocales con tilde. Utilizan doble espacio para facilitar la lectura y el cambio de línea.

Existen diferentes factores que inciden en la capacidad lectora: la motivación, la cantidad de estímulos lectores que recibe el niño, la edad de comienzo de la lectura, el grado de desarrollo de las destrezas previas, la maduración del alumno, los apoyos que encuentra en el entorno y la metodología empleada. En función de todas estas variables es que se debe elegir el método a utilizar.

- **SINTETIZADOR DE VOZ:**

Un sintetizador de voz es un dispositivo que se encarga de la producción artificial del habla humana.

Los sintetizadores de voz son considerados periféricos de salida que suelen incluir un microprocesador, memoria ROM con programas y datos, un conversor D/A, un amplificador de audiofrecuencia y altavoz. La mayor parte de los dispositivos sintetizadores de voz tienen memorizados digitalmente cada uno de los fonemas o palabras que son capaces de emitir. Los datos que recibe un sintetizador procedentes del ordenador corresponden a la identificación de los fonemas o palabras a emitir. Una vez que se analiza el dato, se activa una rutina encargada de generar el sonido correspondiente8.

Para lograr la más completa claridad en el habla, lo adecuado es almacenar pronunciaciones de palabras completas. Los sintetizadores de habla son muy útiles para ser usados por aquellas personas con discapacidades; por ejemplo, se utiliza un sintetizador de habla en programas de asistencia como los lectores de pantalla.

- **TIFLOTECNOLOGÍA:**

La capacitación y el uso adecuado de las tecnologías actuales determinan el desarrollo personal y social de las personas, pero no todos los dispositivos tecnológicos pueden ser utilizados fácilmente por cualquier persona, pero esta situación afecta a las personas con discapacidad visual.

La tiflotecnología, del griego «tiflos», que significa ciego es el conjunto de técnicas, conocimientos y recursos para brindar a las personas con discapacidad visual los medios oportunos para la correcta utilización de la tecnología[9].

---

8 *http://www.cavsi.com/preguntasrespuestas/que-un-sintetizador-de-voz/*

9 *http://www.ite.educacion.es/formacion/materiales/129/cd/unidad_10/m10_tiflotecnologia.htm*

Proporciona los instrumentos auxiliares, ayudas o adaptaciones tecnológicas, creadas o adaptadas específicamente para posibilitar a las personas con limitación visual o sordo ceguera la correcta utilización de la tecnología, contribuyendo a su autonomía personal y plena integración social, laboral y educativa.

Existen una gran variedad de dispositivos tiflotécnicos, que se agrupan en dos grupos:

- Los que facilitan o permiten el acceso a la información del ordenador (sistemas de reconocimiento óptico o inteligente de caracteres, sistemas de reconocimiento táctil, revisores de pantalla, etc.)
- Los que pueden conectarse al ordenador para intercambiar información, aun cuando también funcionan de forma autónoma y tienen su propia utilidad (sistemas portátiles de almacenamiento y procesamiento de la información, impresoras braille, aparatos de reproducción y grabación, calculadoras parlantes, diccionarios y traductoras parlantes, periódicos electrónicos adaptados para personas con discapacidad visual, programas de gestión bibliotecaria y de acceso a Internet, ampliación de la imagen, códigos de barras comprimidos para información de consumo y audio descripción, etc.)

Por tanto la tiflotecnología, es la adaptación y accesibilidad de las tecnologías de la información y comunicación para su utilización y aprovechamiento por parte de las personas con limitación visual.

- **LA LECTURA BRAILLE:**

La lectura es un proceso que implica diversos niveles de análisis de la información, existen diversos modelos con los que se intenta explicar el proceso de la lectura en las personas con limitación visual (ascendente, descendente, interactivo).

En estas personas el área táctil no abarca más allá de la superficie de contacto del dedo lector y el papel, por lo que la velocidad lectora de un buen lector ciego será siempre inferior a la de un buen lector vidente.

Todo proceso de lectoescritura debe partir de lo básico hacia lo complejo, la lectura y pre escritura se combinan para desarrollar actividades y ejercitar los movimientos digito manuales hacia la exploración táctil, en el proceso de lectoescritura es conveniente incorporar la máquina de escribir braille (Perkins) de forma que lectura y escritura sean simultáneas. En la escritura con la máquina Perkins la lectura puede

realizarse inmediatamente sobre lo escrito, la simpleza de la máquina y su fácil manejo la convierten en un instrumento similar al lápiz.

Teniendo en consideración el movimiento de las manos sobre el texto en braille pueden establecerse dos tipos de lectura[10:]

- Lectura unimanual.
- Lectura bimanual: Es mucho más eficiente y dinámica, con esta se evitan pérdidas de tiempo y continuidad en la lectura, mejorando así la comprensión. Para alcanzar una buena velocidad lectora es necesario además utilizar otras estrategias de mejoramiento (des verbalización, búsqueda de imágenes, ideas, situaciones, acciones).

La adquisición de una correcta técnica lectora ayudará a que la lectura en braille sea una actividad gratificante.

La lectura Braille es táctil y móvil, ya que son factores esenciales para hacer posible la lectura en sistema braille, el movimiento de las manos es diferente en cada caso y puede ser de tres tipos[11]:

- Movimiento de cambio de línea: El dedo lector de la mano derecha capta los caracteres finales de la línea, mientras que el dedo de la mano izquierda que busca el inicio de la siguiente, explora en el margen el principio de ésta.
- Barrido Progresivo: Con movimientos continuos, sin variar la velocidad y sin que el dedo lector se separe en ningún momento de la línea de texto.
- Repasos: Son más frecuentes según la habilidad lectora, aumenta el número de grafemas (Unidad mínima de la escritura de una lengua) de una palabra.

En la lectura Braille el proceso de lectura es analítico-asociativo, teniendo el grafema y no la palabra como unidad fundamental, por tanto, el reconocimiento de la lectura se hace letra a letra; el almacenamiento de la información se lleva a cabo en la lectura braille por medio del almacenamiento de grafemas[12].

---

10 *http://sid.usal.es/idocs/F8/FDO5148/desafios_didacticos_lectura_braille.pdf*

11*http://educacion.once.es/appdocumentos/educa/prod/Guia%20didactica%20lectoescritura%20braille.pdf*

12 *http://www.once.es/new/servicios-especializados-en-discapacidad-visual/braille/documentos/DOCUMENTO%20TECNICO%20B%2011%20DIDACTICA%20DEL%20BRAILLE%20V1.pdf*

- **INCLUSIÓN SOCIAL:**

El Instituto Nacional para Ciegos INCI tiene en ejecución un PLAN ESTRATEGICO (2011- 2014) con el cual pretende promover la inclusión educativa, social y cultural de la población con discapacidad visual, mediante la gestión de políticas públicas inclusivas en Colombia.

En Colombia según cifras reveladas por el INCI, el 2% de la población con discapacidad visual accede a Internet, frente al 38.5% de la población en general. Finalmente, el 80% de las personas con discapacidad visual vive en condiciones de pobreza y presentan serias dificultades para acceder a bienes y servicios. Situación aún más crítica en pequeños municipios y en el área rural de Colombia.Por análisis poblacionales como los anteriores, en 1955 el gobierno nacional creó el INCI como Entidad adscrita al Ministerio de Educación Nacional.

Hoy, el INCI cuenta con un equipo Institucional que le permite posicionarse a nivel nacional como la Entidad que brinda asesoría y asistencia técnica a entidades del nivel nacional y territorial en la formulación, ejecución, seguimiento y evaluación de las políticas públicas educativas, sociales y culturales para la inclusión de la población con discapacidad visual.

En consecuencia, el INCI  acoge desde la perspectiva de los derechos humanos, la inclusión social entienda como un proceso que promueve la igualdad de acceso y oportunidades para todos a las estructuras sociales, económicas y políticas de la sociedad.

En lo educativo, busca responder a las necesidades de todos, garantizando la igualdad de oportunidades y apoyando a los estudiantes para que desarrollen sus potencialidades dentro de una comunidad educativa13.

- **CICLO DE VIDA DE XP**

El ciclo de vida recomendado para XP tiene seis fases: Exploración, Planificación de la Entrega (Release), Iteraciones, Producción, Mantenimiento y Muerte del Proyecto14.

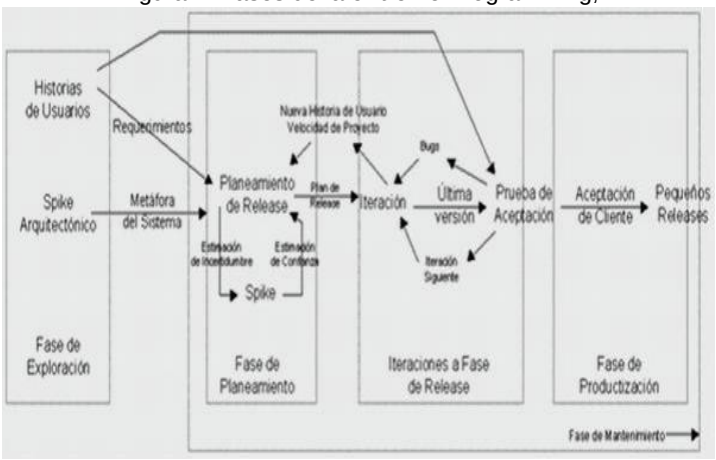
---

13 *www.mineducacion.gov.co/cvn/1665/articles-270320_archivo_doc.doc*

14 *http://programacionextrema.tripod.com/fases.htm*

A continuación se describen brevemente cada una de ellas:

Figura 1. Fases de la eXtreme Programming,




Tomado de *http://ingsoftware072301.obolog.es/metodologia-xp-2012877*

- **Fase de Exploración:**

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto.

Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto.

La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología.

- **Fase de Planteamiento (Release):**

En esta fase el cliente establece la prioridad de cada historia de usuario y los programadores realizan una estimación del esfuerzo necesario en cada una de ellas.

- **Fase de Iteraciones:**

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado, el plan de entrega se compone por iteraciones de no más de tres semanas, al final de la última iteración el sistema estará listo para entrar en producción.

Algunos elementos que se tienen en cuenta durante la elaboración del plan de la Iteración son: historias de usuario, velocidad del proyecto, pruebas de aceptación.

- **Fase de Producción:**

La fase de producción se basa en pruebas adicionales y revisiones de rendimiento antes de que el sistema sea entregado al cliente. Es posible que se rebaje el tiempo que toma cada iteración, de tres a una semana.

- **Fase de Mantenimiento:**

Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones, la fase de mantenimiento puede requerir cambios en su estructura.

- **Fase de muerte del Proyecto:**

Es cuando el cliente no tiene más historias para ser incluidas en el sistema, entonces se debe buscar satisfacer al cliente en aspectos como rendimiento y confiabilidad del sistema.

Se genera la documentación final del sistema y no se realizan más cambios en la arquitectura.

- **VALORES DE XP**

Para que el proyecto a desarrollar sea finalizado con éxito, esta metodología se basa en cuatro valores que debe tener el equipo15:

15 *http://repositorio.utp.edu.co/dspace/bitstream/11059/794/1/0053E18cp.pdf*

- Comunicación: Muchos de los problemas que existen en proyectos de software son debido a la comunicación entre las personas, por esta razón en XP la comunicación debe y es permanente durante todo el proyecto, como la documentación es poca, el diálogo cara a cara, entre desarrolladores y el cliente es el medio de comunicación.

- Simplicidad: Como toda metodología ágil esta se basa en la sencillez, en los procesos, diseño, código (mediante la refactorización), etc. La programación XP no utiliza sus recursos para la realización de actividades complejas, sólo se desarrolla lo que el cliente demanda, de la forma más sencilla.

- Retroalimentación: Debe estar presente desde el inicio del proyecto, debe darse por parte del equipo de trabajo hacia el cliente, con el fin de brindarle información sobre la evolución del sistema y desde el cliente hacia el equipo en los aportes a la construcción del proyecto.

- Coraje: El equipo debe tener coraje cuando se encuentran problemas serios en el diseño, o en cualquier otro aspecto, para darles solución, sin importar la dificultad como por ejemplo cambiar totalmente una parte del código sin importar el tiempo que se ha tardado en desarrollarlo.

- **PRÁCTICAS DE LA XP:**

La programación XP tiene doce prácticas fundamentales, las cuales se plantean a partir de los valores, de forma aislada, cualquier práctica individual de Xp tiene poco sentido, pero en conjunto, unas complementan las carencias que las otras puedan tener, a continuación se describen cada una de ellas16:

---

16 *http://www.davidvalverde.com/blog/introduccion-a-la-programacion-extrema-xp/*

1. **El juego de la planificación:** Desde el inicio el grupo y el cliente deben tener una visión clara del proyecto, la planificación nunca será perfecta y variará según varíen las necesidades del negocio, la planificación es iterativa: un representante del proyecto decide al comienzo de cada iteración qué características concretas se van a implementar.

Hacen parte de esta práctica las historias de usuario que son las que describen la funcionalidad del proyecto y son las más importantes, se realiza también la división de responsabilidades, teniendo en cuenta las necesidades del negocio en todo momento. Dentro de esta división, el representante del cliente tiene las siguientes responsabilidades:

- Decidir qué se implementa en cada release o iteración.
- Fijar las fechas de fin de la release, recortando unas características o añadiendo otras.

Las responsabilidades del equipo de desarrollo son las siguientes:

- Estimar cuánto tiempo llevará una historia y cuales historias se deben incluir en una iteración.
- Estimar el riesgo de cada historia.
- Decidir el orden de desarrollo de historias dentro de la iteración.
- Proporcionar información sobre el coste de utilizar distintas opciones tecnológicas.
- Organizar el equipo.

2. **Pruebas Continuas (Testing):** Los tests son los que dan seguridad a los programadores, se hacen cambios más o menos importantes sin miedo a problemas inesperados, los desarrolladores escriben pruebas unitarias y los clientes especifican pruebas funcionales, estos son elementos claves en la metodología, existen diferentes tipos de test como lo son:

- Test internos o de unidad, garantizan que es correcto al ejecutarse.

- Test de aceptación, garantizan que el código hace lo que debe hacer. Son responsabilidad del cliente en cuanto a que este los define pero no es necesario que los implemente.

Si no hay tests, las cosas sólo funcionan en apariencia, si un test no está automatizado, no se le considera como tal. El objetivo de los tests no es corregir errores, sino prevenirlos.

3. **Programación en parejas:** En la programación XP nadie programa en solitario, siempre hay dos personas delante del ordenador, debido a las siguientes razones:

   - El hecho de que todas las decisiones las tomen al menos dos personas proporciona un mecanismo de seguridad.
   - Es muy difícil que dos personas se salten tareas por descuido o negligencia.
   - El código siempre está siendo revisado por otra persona. La revisión de código es el método más eficaz de conseguir código de calidad.
   - En contra de lo que pueda parecer, los dos desarrolladores no hacen lo mismo: mientras el que tiene el teclado adopta un rol más táctico, el otro adopta un rol más estratégico, preguntándose constantemente si lo que se está haciendo tiene sentido desde un punto de vista global.

4. **Refactorización ((Refactoring):**

   Uno de los objetivos de la XP es mantener la curva de costes tan plana como sea posible, por lo que existen una serie de mecanismos destinados a mantener el código en buen estado, modificándolo activamente para que conserve claridad y sencillez a este proceso se le llama refactorización.

   Esta no sólo sirve para mantener el código legible y sencillo sino que también se utiliza cuando resulta conveniente modificar código existente para hacer implementar más fácil una nueva funcionalidad sin cambiar el funcionamiento del mismo.

5. **Diseño simple:**

   Otra práctica fundamental es utilizar diseños tan simples como sea posible. La XP soporta modificaciones gracias a los tests automatizados, estos permiten hacer cambios importantes, la refactorización hace que el código existente sea claro y sencillo, también ayuda a hacer factibles las modificaciones.

   Se plasma el diseño del sistema en tarjetas CRC (Clase – Responsabilidad-Colaboración), no se implementan características que no son necesarias, con esta técnica, las clases descubiertas durante el análisis pueden ser filtradas para determinar qué clases son realmente necesarias para el sistema.

6. **Propiedad colectiva del código:**

   Todo el mundo tiene autoridad para hacer cambios a cualquier código, cada cuál es responsable de las modificaciones que haga. El principio básico es "tú lo rompes, tú lo arreglas, no importa si está en el código propio o en el de otros".

7. **Integración Continua:**

   La Programación Extrema hace que la integración sea permanente, con lo que todos los problemas se manifiestan de forma inmediata, en lugar de durante una fase de integración más o menos remota.

   Los cambios se integran en el código base varias veces por día, todos los casos de prueba se deben pasar antes y después de la integración, se dispone de una máquina para la integración y se realizan test funcionales en donde participa el cliente.

8. **Cliente en el equipo:**

   El equipo de desarrollo tiene acceso todo el tiempo al cliente, el cual está disponible para responder preguntas, fijar prioridades, etc. Esto no siempre se consigue; Un cliente muy Junior no sirve y un cliente muy Sénior no es disponible. "Lo ideal es un cliente Analista".

9. **Releases pequeñas:**

   Para dar el máximo valor posible en cada momento, se intenta liberar nuevas versiones de las aplicaciones con frecuencia. Éstas deben ser tan pequeñas como sea posible, aunque deben añadir suficiente valor como para que resulten valiosas para el cliente.

10. **Semanas laborales de 40 horas:**

    Una semana de 40 horas (no más) en las que se dedica la mayor parte del tiempo a tareas que den un avance y hace innecesario recurrir a sobreesfuerzos (excepto en casos extremos), porque estos llevan a un rendimiento menor y a un deterioro de la moral de todo el equipo.

11. **Estándares de codificación:**

    Todo el código debe estar escrito de acuerdo a un estándar de codificación que es necesario para soportar otras prácticas de la metodología, para conseguir que el código se encuentre en buen estado y que cualquier persona del equipo pueda modificar cualquier parte del código.

12. **Uso de Metáforas:**

    La comunicación fluida es uno de los valores más importantes de la programación extrema, para conseguir que la comunicación sea fluida es importante utilizar el vocabulario del negocio. También es fundamental huir de definiciones abstractas. Dentro de este enfoque es fundamental buscar continuamente metáforas que comuniquen intenciones y resulten descriptivas, enfatizando el qué por delante del cómo.

    Cada Proyecto es guiado por una historia simple de cómo funciona el sistema en general, reemplaza a la arquitectura y debe estar en lenguaje común, entendible para todos (Cliente y Desarrolladores), esta puede cambiar permanentemente.

- **PRUEBAS**

- **Pruebas unitarias:**

  Las pruebas unitarias son una de las piedras angulares de XP. Todos los módulos deben pasar las pruebas unitarias antes de ser liberados o publicados, esto es lo que habilita que funcione la propiedad colectiva del código, las pruebas deben ser definidas antes de realizar el código ("Test-driven programming").

  En este sentido, el sistema y el conjunto de pruebas debe ser guardado junto con el código, para que pueda ser utilizado por otros desarrolladores, en caso de tener que corregir, cambiar o recodificar parte del mismo.

- **Detección y corrección de errores:**

Cuando se encuentra un error ("bug"), éste debe ser corregido inmediatamente, y se deben tener precauciones para que errores similares no vuelvan a ocurrir, se generan nuevas pruebas para verificar que el error haya sido resuelto.

- **Pruebas de aceptación:**

Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Las pruebas de aceptación son consideradas como "pruebas de caja negra" ("Black box system tests"). Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. Dado que la responsabilidad es grupal, es recomendable publicar los resultados de las pruebas de aceptación, de manera que todo el equipo esté al tanto de esta información.

### 1.7.3 MARCO CONCEPTUAL

- **Aplicación móvil**

Una aplicación móvil o app es una aplicación informática diseñada para ser ejecutada en teléfonos inteligentes, tabletas y otros dispositivos móviles. Por lo general se encuentran disponibles a través de plataformas de distribución, operadas por las

compañías propietarias de los sistemas operativos móviles como Android, iOS, BlackBerry OS, Windows Phone, entre otros17.

- **Android**

Es un sistema operativo basado en el kernel de Linux diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas que puede ser adoptado por cualquier fabricante y permite realizar tareas que se asemejan a una PC, como navegar la web, leer emails, descargar aplicaciones, etc18.

- **Java**

Es un lenguaje de programación con el que se puede realizar cualquier tipo de programa. En la actualidad es un lenguaje muy extendido y cada vez cobra más importancia tanto en el ámbito de Internet como en la informática en general. Está desarrollado por la compañía Sun Microsystems con gran dedicación y siempre enfocado a cubrir las necesidades tecnológicas más punteras19.

Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si se desarrolla un programa en Java podrá funcionar en cualquier ordenador del mercado. Es una ventaja significativa para los desarrolladores de software, pues antes se tenía que hacer un programa para cada sistema operativo, por ejemplo Windows, Linux, Apple, etc. Esto lo consigue porque se ha creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente.

- **Android SDK manager**

SDK responde a las siglas Software Development Kit, lo que viene a ser un kit de desarrollo de software. Con él se puede desarrollar aplicaciones y ejecutar un emulador del sistema Android de la versión que sea. Todas las aplicaciones Android se desarrollan en lenguaje Java con este kit20.

- **JDK**

Java Development Kit o (JDK), es un software que provee herramientas de desarrollo para la creación de programas en java. Puede instalarse en una computadora local o

---

17 *http://sesmark.mx/servicios/desarrollo-de-aplicaciones-moviles*

18 *http://es.slideshare.net/Yalvin/sistema-operativo-41336411*

19 *http://www.desarrolloweb.com/articulos/497.php*

20 *http://www.androidpit.es/sdk-android*

en una unidad de red, el JDK tiene el compilador que permite convertir código fuente en bytecode, es decir, el código "maquina" de la máquina virtual de Java al código21.

- **Eclipse**

Es un entorno de desarrollo integrado, de Código abierto y Multiplataforma. Es una potente y completa plataforma de Programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java22.

No es más que un entorno de desarrollo integrado (IDE) en el que se encuentran todas las herramientas y funciones necesarias para tu trabajo, recogidas además en una atractiva interfaz que lo hace fácil y agradable de usar.

- **TTS (Text to Speech-Texto a Voz)**

Esta tecnología permite la conversión de un texto a una voz artificial o sintética. Está disponible a través de varios software, orientados hacia, personas no videntes, con baja visión, ancianos o analfabetos que podrán acceder a los textos gracias a que podrán oírlos23.

La modificación del mensaje implica sencillamente modificar el texto. No implica volver a grabar, como cuando se utiliza mensajes pregrabados, lo cual reduce costes y facilita la tarea.

- **SQLite**

Es una pequeña librería programada en lenguaje C que implementa un completo motor de base de datos multiplataforma que no precisa configuración. Se distribuye bajo licencia de dominio público. Es muy rápido y la ventaja fundamental es que permite utilizar un amplio subconjunto del lenguaje estándar SQL.

SQLite destaca también por su versatilidad. Combina el motor y el interfaz de la base de datos en una única biblioteca, y almacena los datos en un único archivo de texto plano.

Esto hace que cada usuario pueda crear tantas bases de datos como desee sin la necesidad de la intervención de un administrador de bases de datos que gestione los espacios de trabajo, usuarios y permisos de acceso.

---

21 *https://es.wikipedia.org/wiki/Java_Development_Kit*

22 *http://gl-epn-programacion-ii.blogspot.com.co/2010/03/jvm-jdk-jre-conceptos-fundamentales-de.html*

23 *https://es.wikipedia.org/wiki/Conversor_texto-voz*

El hecho de almacenar toda la base de datos en un único archivo, facilita la portabilidad de los datos, y solamente tiene la restricción del espacio de disco asignado al usuario en el servidor24.

### 1.7.4 MARCO METODOLÓGICO

#### 1.7.4.1 SOLUCIÓN TECNOLÓGICA

- **TTS en Android**

Android.speech.tts.TextToSpeech.OnInitListener:

Esta será la librería que se utilizara para:

- Sintetizar mediante voz cualquier tipo de carácter y texto del sistema.
- Sintetizar mediante voz todo el contenido del manual de usuario.
- Sintetizar mediante voz cada una de las ayudas de cada uno de los ítems a elegir de los menús.

- **Plataforma de Eclipse**

Eclipse es la plataforma de desarrollo sobre el que se ejecuta el plug-in SDK, además de ser la herramienta para elaborar:

- Interfaz gráfica.
- Métodos para realizar los conjuntos de instrucciones definidas dentro de las clases.

- **Herramientas desarrolladoras de Android**

SDK (*Software Development Kit),* lo que viene a ser un kit de desarrollo de software. Incluye las herramientas desarrolladoras de Android (el plug-in de Eclipse), que se utilizaran como emulador del dispositivo Android:

---

24 *http://recursostic.educacion.es/usuarios/web/preguntas-frecuentes/45-sqlite-*

- Extensiones .XML
- Emulador de las diferentes versiones de ANDROID.
- Librerías
- Métodos
- Entre Otros.

- **Código fuente**

Los fragmentos de código que se utilizaran incluye:

- Intent Receiver—Esto demuestra la implementación de un Intent Receiver, que es la clase que procesa las intenciones según se promociona por la etiqueta Intent Filter en el archivo AndroidManifest.xml.

- SaySomething.java — Esto implementa una actividad Android, el punto de entrada principal para la misma aplicación.

- Main.xml — Esto contiene los elementos visuales o recursos para el uso de las actividades Android, como menús y elementos gráficos.

- R.java — Este archivo se genera automáticamente por las herramientas desarrolladoras de Android y "conecta" los recursos visuales para el código fuente Java.

- AndroidManifest.xml — Entrega los permisos para interactuar con otros recursos del hardware, ejemplo: cámara, micrófono etc.

- AndroidManfest.xml complete, que es el que se encarga de dar una descripción de cada uno de los elementos importantes.

- **SQLite.**

Este motor de bases de datos será en el cual se desarrollaran todas las bases de datos para el proyecto:

- Teoría del sistema braille.
- Videos con conocimiento del sistema Braille.
- Información de usuario.

- Imágenes.
- Información de actividades de entretenimiento.
- Entre otras.

- **JDK 7.3.1**

Proveerá las herramientas de desarrollo para el sistema.

## 1.8 FACTIBILIDAD

A continuación se desarrolla un estudio para estudiar la factibilidad del proyecto teniendo en cuenta su nivel técnico, operativo, legal y económico.

### 1.8.1 Factibilidad Técnica

La factibilidad técnica se realizó con el fin de recolectar y analizar la información sobre los componentes tecnológicos necesarios para desarrollar el presente proyecto de manera óptima y eficaz, para dicho fin se plantearon dos aspectos, hardware y software.

**Requerimientos de hardware:**

| **Nombre** | **características** |
|---|---|
| Microprocesador | Procesador AMD Single-Core E-240 con tecnología VISION |
| **Memoria** | Memoria de sistema DDR3 de 1GB (1 DIMM) |
| **Gráficos de video** | Gráficos AMD Radeon HD 6310M Discrete-Class con memoria de gráficos total de hasta 373 MB |
| **Disco duro** | Disco duro de 320 GB (5400 RPM) |
| **Unidad multimedia** | Unidad SuperMulti DVD±R/RW 8x con tecnología LightScribe compatible con doble capa |
| **Pantalla** | Pantalla LED de alta definición de 14,0 pulgadas en diagonal con tecnología HP BrightView (1366 x 768) |

| | |
|---|---|
| **Tarjeta de red** | Lector de medios digitales para tarjetas Secure Digital y Multimedia |
| **Ranuras para PC Card** | Lector de medios digitales para tarjetas Secure Digital y Multimedia |
| **Teclado** | Teclado de tamaño normal USB idioma español-latino |
| **Mouse** | Dos botones USB |

**Hardware móvil a usar:**

Para las pruebas del presente proyecto se contará con dos tipos de móvil, ASUS NEXUS 7 con versión de Android 4.2 y con un móvil Avvio 765 con versión 4.4 de Android.

**Recursos de software:**

En cuanto a los recursos de software necesarios para el correcto funcionamiento del presente proyecto se encuentra:

- JDK 7.3.1
- SQLite.
- SDK manager *Software Development Kit.*
- Plataforma de Eclipse (móvil).
- Windows 7 32 bits.
- Adobe Reader (PDF).
- Google Chrome.
- Nela.

De lo anterior se puede concluir que resulta viable el presente proyecto en cuanto a la factibilidad técnica.

### 1.8.2 Factibilidad Operativa

Resulta necesario para poder iniciar con el presente proyecto que se cuente con los siguientes factores humanos

A continuación se presenta la información del personal humano que requiere este proyecto y sus respectivos conocimientos y habilidades

| **Personal operativo** | **Características de sus funciones** |
|---|---|
| Para correcto desarrollo del presente proyecto son necesarios como mínimo 2 programadores, preferentemente que cuenten con título de tecnólogo o que se encuentren en últimos semestres de este | • Conocimientos en bases de datos SQLite.<br>• Manejo de sistema operativo Windows y Android.<br>• Conocimientos del lenguaje java.<br>• Conocimientos de manejo de la plataforma eclipse en conjunto con las herramientas SDK manager.<br>• Conocimientos básicos sobre manejo de aplicaciones móviles. |
| | Habilidades<br><br>• Trabajo en equipo.<br>• Innovación y creatividad.<br>• Capacidad de rápido aprendizaje.<br>• Capacidad de comunicación.<br>• Capacidad de transmitir conocimiento.<br>• Correcta interrelación con otros. |
| Tutor de proyecto | Brinda un proceso de acompañamiento integral al grupo de programadores para la correcta incorporación den el desarrollo de las competencias de innovación y la integración en procesos de investigación. |

Teniendo en cuenta la información anterior se puede concluir, que el proyecto resulta factible operativamente ya que como punto de partida cuenta con el personal requerido, para el inicio de su desarrollo.

### 1.8.3 Factibilidad Económica

A continuación se muestra la información de los recursos económicos que se necesitaran a lo largo del desarrollo del presente proyecto

| Recursos | | Proveedor | Costos | Total |
|---|---|---|---|---|
| **Software** | Eclipse luna | Software libre | $0 | $0 |
| | SQLite | Software libre | | |
| | SDK manager | Software libre | | |
| | Windows 7 S.O | Equipo de programación | $ 80.000 | $ 80.000 |
| **Hardware** | 2 Ordenadores necesarios | Equipo de programación | $1'200.000 | $2'400.000 |
| | 2 Dispositivos móviles | Equipo de programación | $300.000 | $300.000 |
| **Talento humano** | Desarrolladores | | $20.000 cada hora<br>$2'880.000 Mensualmente | $28'800.000 |
| | Tutorías | | $31.000 cada hora<br>$744.000 Mensualmente | $ 7.440.000 |
| **Otros** | Papelería, transportes, medios magnéticos de almacenamiento | | $120.000 Mensualmente | $1.200.000 |

| | |
|---|---|
| **Total** | $40.220.000 |

Partiendo desde la información presentada anteriormente es correcto afirmar que el presente proyecto resulta factible en cuanto a su nivel económico requerido.

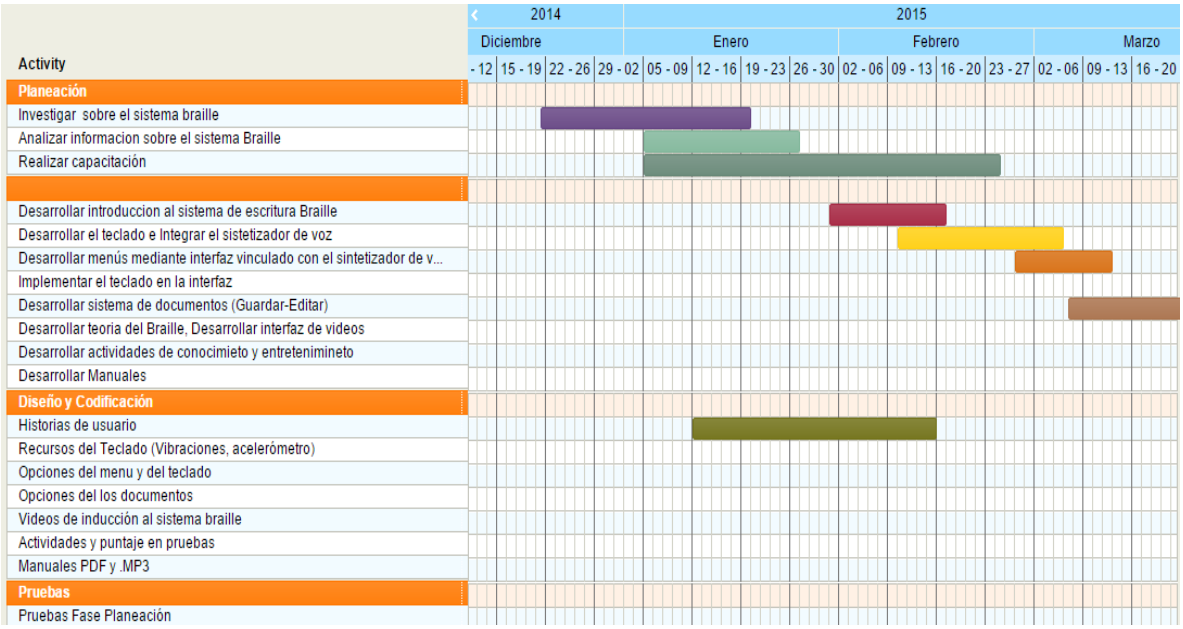
### 1.8.4 Factibilidad Legal

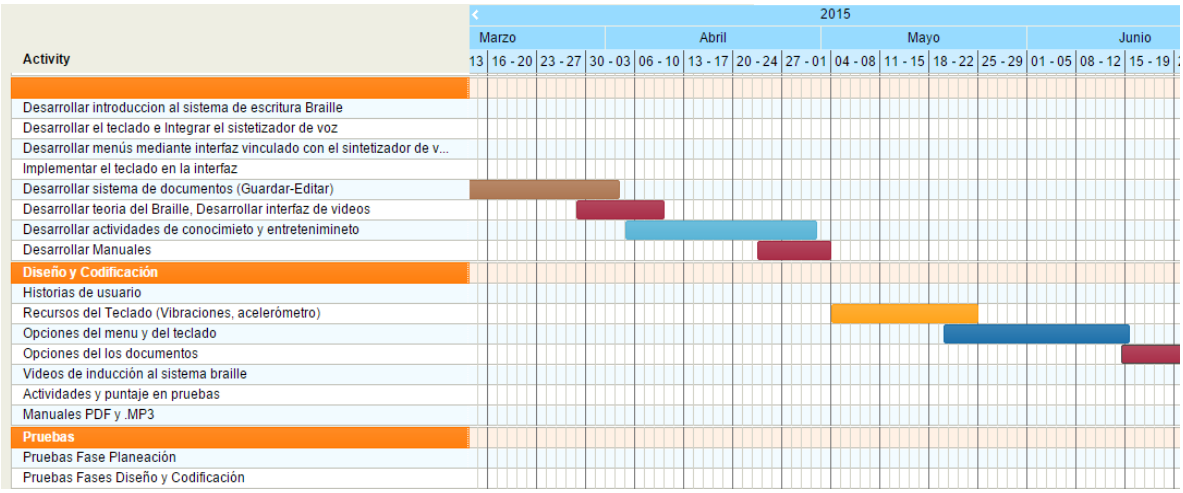
En cuanto al nivel legal las normas que pudiesen generar alguna contrariedad frente al desarrollo del proyecto según la investigación que se realizó fueron:

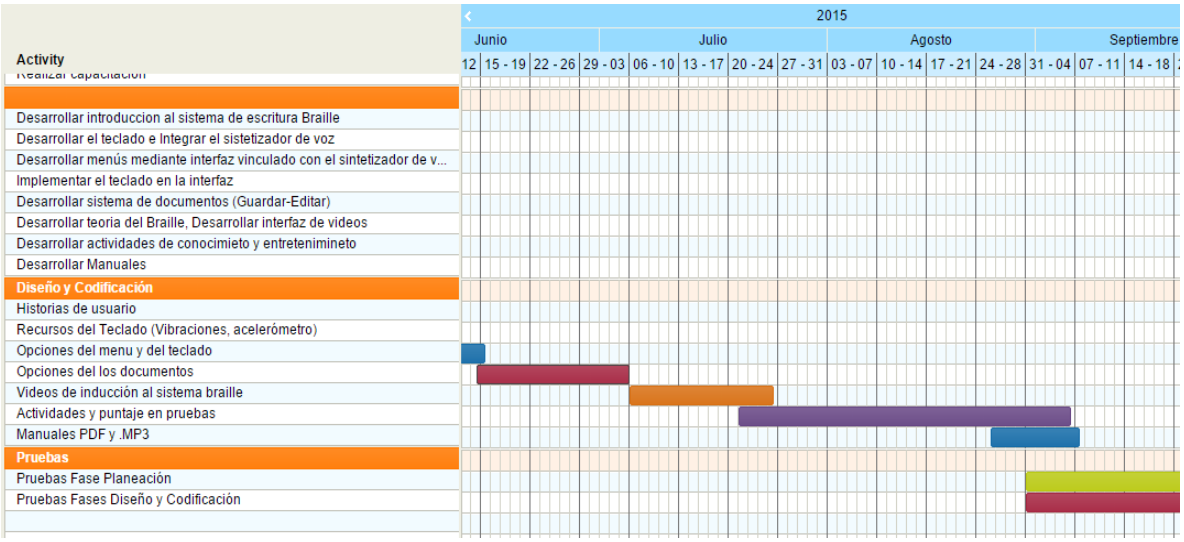
- En cuanto a licencias de las herramientas que se utilizaran como eclipse, SQ Lite, SDK manager no tienen licencia por lo que no implica algún infringimiento legal.
- En cuanto a Windows, se cuenta con la licencia.
- Para cada una de las pruebas de software que se realizaran en móviles, se cuenta con cada una de sus respectivas facturas y registros.
- Las instituciones como centros de aprendizaje para estudiantes en condición de discapacidad visual, de manera general, no tienen ningún tipo de reglamento que impida la ejecución e implantación de este proyecto en las organizaciones de este tipo.

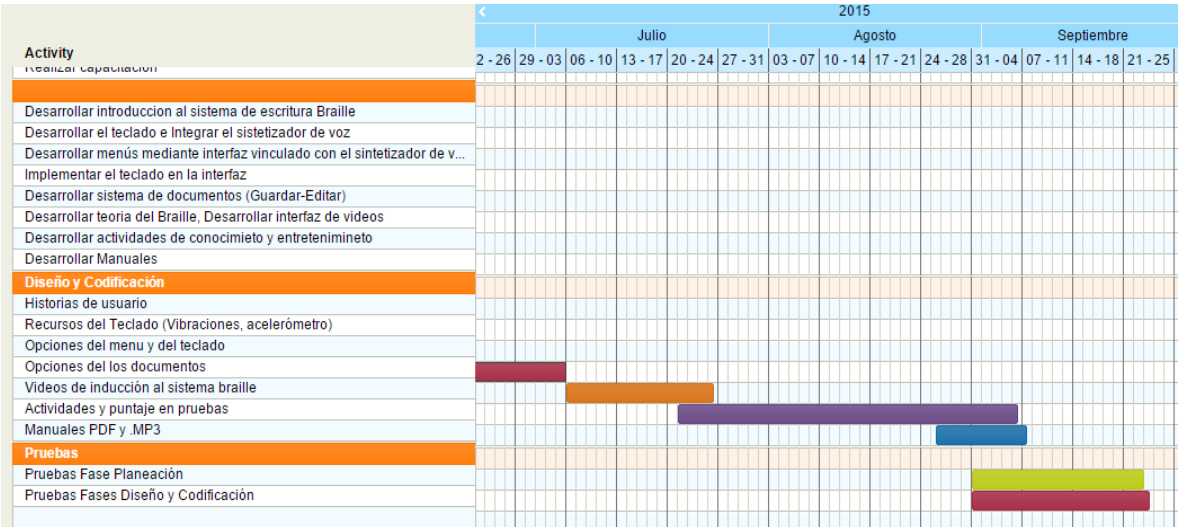
De acuerdo a la información presentada anteriormente se puede llegar a la conclusión de que el presente proyecto es viable en cuanto al aspecto legal.

## 1.9 CRONOGRAMA.






## 2. FASE DE ANÁLISIS

### 2.1 Consulta a fuentes de información apropiadas e identificación de problemas.

Con el uso de nuevas tecnologías en la sociedad, las personas buscan la manera de capacitarse en el uso de las mismas, a diferencia de las personas con limitaciones visuales, las que lo son deben aprender a utilizar “sistemas” alternativos de comunicación, estos se dividen en:

- Alfabéticos: Exigen que los usuarios conozcan el sistema de lectura y/o escritura Braille.

- No alfabéticos: Son el lenguaje oral (habla).

Es por esto que las personas con discapacidad visual deben interactuar con su entorno para dar uso a sus otros sentidos este es un proceso mental, intelectual y perceptivo.

El primer problema que se evidencia es que el usuario no este familiarizado con el uso de dispositivos móviles o tabletas, ya que se le dificultaría el uso del aplicativo. El ritmo de aprendizaje y de ejecución de tareas para las personas con discapacidad visual suele ser más lento frente a las que no lo son.

Una debilidad es que la persona no cuente con el conocimiento necesario del sistema Braille, pero no es de mayor trascendencia ya que con el aplicativo se busca que las personas aprendan del mismo, otra debilidad evidenciada es la edad de las personas que interactúan con el aplicativo, ya que en algunas ocasiones las personas de edades más jóvenes tienen mayor capacidad para utilizar los dispositivos móviles.

### 2.1.2 Análisis de alternativas de solución.

Debido a que las personas con discapacidad visual tienen ayudas informáticas para algunas de sus discapacidades como las herramientas de software: **Magic** que es un software que da ayuda óptica a personas con deficiencia visual o con miopía que quieren trabajar, estudiar o jugar con un computador y pasar largos períodos de tiempo frente a la pantalla lo que principalmente hace es aumentar hasta 16 veces el tamaño de las letras o de los objetos a percibir en la pantalla25; el software **Jaws** que es una herramienta que permite a las personas en condición de discapacidad visual transformar en sonido los textos del computador, trabaja con aplicaciones como Microsoft Word, Excel y PowerPoint, al igual que con documentos en PDF, diferentes navegadores de Internet, correo, servicios de llamadas y mensajería instantánea como Skype, redes sociales y reproductores de video y audio como VLC Media Player o el reproductor de Windows Media26; pero no tienen aplicaciones para dispositivos móviles entonces se planteó un aplicativo enfocado principalmente en estas personas y a la falta de accesibilidad que ellas tienen, en donde las personas con discapacidad visual pudieran en primera instancia aprender los símbolos del sistema Braille, para luego de adquirir esos conocimientos redactaran cualquier texto y lo copiaran y

25 *http://www.ver.com.co/hogar/magic.html*

26 *http://www.guiaacademica.com/educacion/personas/cms/pregrado/conexionu/ARTICULO-WEB-NEW_NOTA_GUIA-13478695.html*

pegaran en aplicaciones de su dispositivo móvil, por último la realización de test en donde se evalúa si el usuario está aprendiendo o no, permitiendo de este modo el uso de estas nuevas tecnologías que son tan importantes en la sociedad hoy en día y que sin duda alguna es una herramienta que les será de gran ayuda

### 2.1.3 Establecimiento del papel del computador

Para cualquier persona las tecnologías digitales no son hoy ya un lujo sino una herramienta para el trabajo, el ocio y la comunicación. El ordenador supone para la sociedad de hoy lo que en su día supuso la luz eléctrica: ¿nos podemos imaginar un mundo sin ella? hasta ahora el manejo de las tecnologías en los centros educativos ha estado limitado por cuestiones de equipamiento, formación del profesorado e insuficiencia de software educativo. A todas estas trabas hay que añadirle una para los niños con discapacidad y es la falta de accesibilidad de los programas.

Las tecnologías hacen más fácil la integración: los alumnos toman notas con mayor facilidad en un Braille hablado o en un ordenador portátil que en la antigua máquina de escribir Braille; los apuntes pueden ser enviados por correo electrónico a los maestros para que los transcriban, las impresoras permiten que lo que el niño escribe en Braille lo imprima en tinta para su maestro. Hay programas y páginas web como correos electrónicos que son perfectamente manejables con revisores de pantallas, por lo que también se les ha facilitado la comunicación. También los que tienen algo de visión tienen diferentes posibilidades de configurar la pantalla de forma que los textos y los iconos aumenten de tamaño, que los colores varíen en función de sus necesidades.

Además todo esto hace que su autoestima y sensación de competencia suba a la par que aumenta sus posibilidades de integración, pues puede comunicarse con sus compañeros, hablar de las mismas cosas, aprender de ellos y enseñarles también a ellos, intercambiarse apuntes, correos electrónicos27.

### 2.1.4 Selección de la Herramienta o lenguajes de Programación.

| Lenguaje de Programación | Creador | ¿Qué es? | Ventajas | Desventajas |
|---|---|---|---|---|

27 *http://nodejesdeinnovar.blogspot.com.co/*

| | | | | |
|---|---|---|---|---|
| Java | James Gosling (Sun Micro Systems) | Es un lenguaje orientado a objetos de propósito general.<br><br>Es considerado como el mejor lenguaje de desarrollo de aplicaciones<br><br>Permite escribir programas de interfaz gráfica o textual. | Puede ser utilizado en cualquier plataforma (Windows, Android, iOS, Linux,etc)<br>Se considera el lenguaje más simple en comparación a otros lenguajes de programación.<br>Las aplicaciones desarrolladas con este lenguaje se pueden ejecutar en cualquier plataforma de hardware y software. | Los programas hechos en Java no tienden a ser muy rápidos |
| Eclipse | A finales de 2001 IBM, junto a Borland, crearon la fundación sin ánimo de lucro Eclipse.<br>. | Es uno de los entornos Java más utilizados a nivel profesional. El paquete básico de Eclipse se puede expandir mediante la instalación de plugins para añadir funcionalidades a medida que se vayan necesitando. Es una plataforma de código abierto y multiplataforma | Tiene pruebas unitarias con JUNIT<br><br>La compilación es en tiempo real. | Un inconveniente que tiene en común con otros IDEs es el consumo de recursos del sistema.<br><br>Permite crear aplicaciones en diferentes lenguajes (PHP, Python,Rubi) |
| Python | Guido van Rossum en el año 1990. | Es un lenguaje de programación dinámico y orientado a objetos.<br><br>Su objetivo principal es la facilidad, tanto de lectura, como de diseño. | Soporte e integración con otros lenguajes Y herramientas.<br><br>La escritura de python promueve código limpio y legible. | Si se crean aplicaciones más complejas el código es más complicado de escribir.<br>Los programas interpretados son más lentos que los compilados. |

| | | | | |
|---|---|---|---|---|
| | | | Es portable y se puede ejecutar en varios sistemas (multiplataforma). | |
| **PHP** | Desarrollado en el año 1994 por Rasmus Lerdorf | Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML y ejecutado en un servidor. | Es un lenguaje multiplataforma.<br><br>Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.<br><br>Programación segura y confiable ya que el código fuente es invisible al navegador y al cliente.<br><br>Conexión con gran mayoría de los motores de base de datos que se utilizan en la actualidad. | Como es un lenguaje de código abierto, todas las personas pueden ver el código fuente, y si hay errores, la gente puede utilizar estas debilidades de codificación Se necesita instalar un servidor web. |

## 2.2 DISEÑO

### 2.2.1 Entorno para el diseño del MEC.

- **Destinatarios:**

Esta aplicación es destinada a personas con cualquier tipo de discapacidad visual y para personas que no la tengan; pero que estén interesadas en aprender el alfabeto Braille.

- **Área de contenido:**

La aplicación cuenta con tres módulos (enseñanza, práctica y test) dentro de estos está el contenido del sistema Braille, lo que se quiere lograr primeramente es que las personas aprendan la simbología del sistema Braille, para que luego logren digitar textos, puedan copiarlos y pegarlos en una aplicación diferente y por último se evalué el aprendizaje por medio de los test.

- **Necesidad educativa:**

Debido a que hoy en día la mayoría de personas con discapacidad visual aprenden con una regleta Braille y un punzón lo que se quiere es que con ayuda del aplicativo logren aprenderlo de una forma más lúdica.

El aplicativo tiene como fin que los usuarios finales aprendan terminología del sistema Braille, para que así luego de que tenga una gran cantidad de conocimiento del mismo pueda escribir por medio de alguno de los dos teclados (vertical y horizontal) que tendrá disponible en la aplicación.

Para las personas que no tienen ninguna discapacidad visual, se plantean los mismos módulos y teclados, en donde al igual que el usuario con discapacidad visual, esta puede aprender la simbología del Braille, escribir y realizar los test correspondientes del material.

- **Limitaciones:**

  - Es necesario el desarrollo de manuales en PDF y mediante voz.
  - Se deberá tomar en cuenta las características de hardware en las que se ejecute el proyecto.
  - Es necesario la implementación de aplicaciones existentes como talk back ofrecida por google a dispositivos móviles, con sistema operativo Android.
  - Es necesario que el proyecto trabaje de la mano con la librería extra para el sintetizador de voz TTS.
  - El aplicativo es solo para dispositivos con sistema operativo Android.
  - La aplicación debe instalarse en el dispositivo, sea celular o Tablet.

- **Recursos para los usuarios:**

Los usuarios cuentan con recursos de voz (sintetizador TTS), en todo momento en que el usuario interactúa con el aplicativo tiene ayudas auditivas, cuando se desliza por los módulos (ViewPager) el sintetizador de voz le dice en donde se encuentra

ubicado al igual que cuando ingresa a uno de los módulos, por último para mayor comodidad del usuario con discapacidad visual se inhabilitaron las teclas de atrás y opciones del dispositivo móvil.

- **Soporte lógico que se va a utilizar:**

Para el desarrollo del aplicativo se trabajó con: Adroid Studio, SQLite (Base de Datos), ViewPager para que el usuario se desplazara de manera más rápida por los menús, sintetizador de voz (TTS).

### 2.2.2 Diseño.

**Alcance:**

Este proyecto se centra en desarrollar una forma de comunicación escrita para población con discapacidad visual mediante la tecnología móvil.

En primer lugar. Se encuentra el módulo de escritura y lectura

- Se desarrolla un teclado para las personas con discapacidad visual, que cuenta con 6 puntos que implementa la sintaxis del sistema Braille.
- Con el teclado desarrollado el usuario con discapacidad visual, podrá escribir cualquier tipo de texto mediante el principio de la sintaxis punteada del sistema Braille.
- Con la implementación de la librería TTS el usuario con discapacidad visual podrá saber qué y cómo está escribiendo su texto.
- El usuario con discapacidad visual podrá copiar y pegar los documentos escritos.
- El usuario con discapacidad visual podrá editar los textos guardados.

En segundo lugar. Se encuentra el módulo de aprendizaje

- Se desarrolla manuales de usuario en PDF y mediante voz para ambos usuarios.
- Se desarrolla teclados de prueba simulando el sistema Braille para el usuario común.

En tercer lugar. Se encuentra el módulo de evaluación

- Se desarrolla test en los que se evalúan los conocimientos adquiridos.

**Contenido:**

El aplicativo se elabora para que en su contenido se encuentren tres módulos diferentes (aprendizaje, escritura y test), en cada uno de los módulos el usuario final lograra interactuar de manera fluida y fácil con cada uno de ellos.

En el módulo de aprendizaje el usuario encontrara la terminología del Braille, empezando por las letras del abecedario, los números, signos matemáticos, diéresis, vocales acentuadas.

En el módulo de escritura el usuario cuenta con dos tipos de teclado, el teclado vertical y el teclado horizontal, en cualquiera de los dos el usuario podrá escribir el texto que desee, teniendo la opción de espacio, borrar y salto de línea, también puede saber el texto que ha digitado manteniendo prolongadamente el dedo oprimido sobre la pantalla ya que el sintetizador de voz le dictara lo que tiene escrito, por ultimo tiene la opción de copiar este texto y pegar en otra aplicación diferente (Facebook, WhatsApp).

El módulo de test cuenta con juegos, en el primero se le dice un símbolo del Braille y él debe oprimir con sus dedos el total de puntos con los que se forma esa letra, en el segundo el usuario debe escuchar los números y luego repetirlos, en el último se le dirá que escriba algún símbolo del alfabeto Braille y el deberá digitarlo; la finalidad es evaluar si el usuario está aprendiendo la simbología del Braille.

### 2.2.3 Diseño de Interfaz.

Para que el usuario pueda “comunicarse” con el aplicativo necesitara de un dispositivo móvil en este caso será un celular o una Tablet que tenga sistema operativo Android, para la interfaz de entrada se utilizaron ViewPager para que deslizarse por cada menú fuera más fácil, el sintetizador de voz lo guía diciéndole en donde se encuentra ubicado, que debe hacer y como ubicara cada punto; al momento de que el usuario desee ingresar en alguna de las categorías solo debe oprimir prolongadamente la pantalla, si en algún momento quiere salir entonces debe hacer dos toques sobre la pantalla, de esta manera es que el usuario interactúa con la aplicación.

En el módulo de aprendizaje encontrara 7 categorías diferentes (letras, números, signos ortográficos, vocales acentuados, signos matemáticos, diéresis), en el módulo de escritura tendrá dos teclados (vertical y horizontal) en los que tendrá tres opciones de tiempo (lento, medio y rápido) que es el tiempo de respuesta del aplicativo al momento en que digita la combinación, por último en el módulo de test tendrá a su disposición 3 tipos de juegos en el primero se le dice un símbolo del Braille y él debe oprimir con sus dedos el total de puntos con los que se forma esa letra, en el segundo

el usuario debe escuchar los números y luego repetirlos, en el último se le dirá que escriba algún símbolo del alfabeto Braille y el deberá digitarlo; todo esto con ayuda del sintetizador de voz.

### 2.2.4 Diseño computacional.

Las personas con discapacidad visual tienen dificultad más o menos graves para recibir e interpretar cualquier información visual, entre otras, la lengua escrita. En una sociedad como la nuestra, dominada por los medios de comunicación, esto supone una barrera decisiva para la inclusión, quienes tienen una discapacidad visual parcial necesitan ayudas ópticas para leer. Los ciegos sólo pueden leer los textos escritos en el sistema de escritura Braille, en la actualidad también les es posible leer cualquier texto informatizado si cuentan con un programa de traducción a voz de la lengua escrita. Es nuestro deber facilitarles los recursos que necesitan para que la lengua escrita les resulte accesible. También tienen especiales problemas para desplazarse con autonomía en un mundo cuyas condiciones de accesibilidad dejan mucho que desear.28

Debido a que la mayoría de personas con discapacidad visual no saben Braille como generalmente se cree, lo primero que se desea que el aplicativo pueda apoyar es que tanto estas personas como las que no tiene discapacidad visual aprendan la simbología del Braille siendo el aplicativo un material de apoyo tanto como para el maestro que enseña el Braille como para la personas con discapacidad visual, para que luego en otro de los módulos de la aplicación logre practicar escribiendo textos y así poder copiar y pegar en otra aplicación del celular o Tablet en la que esté trabajando, ayudando así a los maestros que como en el CRAC dictan materias como elementos de la comunicación (Fomenta la participación en actividades de aprendizaje, aplicación del conocimiento y comunicación a través del uso de productos y tecnologías disminuyendo las barreras en los factores ambientales individuales)29.

### 2.2.5 Preparación y evaluación del prototipo.

---

28 *http://sid.usal.es/idocs/F8/FDO19306/orientaciones_visual_UCA.pdf*

29 *http://www.cracolombia.org/rehabilitacion-integral-para-personas-ciegas/*

Debido a que conjuntamente se trabajó con el Centro de Rehabilitación para Adultos Ciegos CRAC, cuando se finalizó el prototipo se llevó la aplicación para que fuese evaluada allí, fue ensayada por personas con discapacidad visual, en ella pudimos detectar que las personas de edades más jóvenes tenían mayor capacidad para adaptarse al dispositivo y al uso de la aplicación.

Para las personas que tenían conocimientos previos en el sistema Braille, fue mas sencillo el escrito de textos.

Hablando con ellos y las profesoras de diferentes áreas del centro, muchos se encontraron a gusto con la aplicación, mostrando interés en instalarla en sus dispositivos e incluso en dejarlo en el centro para que los maestros tuvieran a disposición esta herramienta que no solo será de ayuda para enseñar el Braille, sino que también les servirá de refuerzo en varias áreas, podrán digitar textos para luego copiarlos en una aplicación diferente.

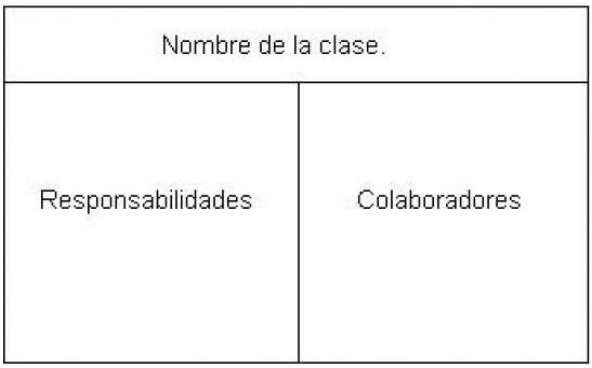
### 2.2.6 FASE DE IMPLEMENTACIÓN.

**TARJETAS CRC (Clase - Responsabilidad – Colaborador).**

Estas tarjetas se dividen en tres secciones que contienen la información del nombre de la clase, sus responsabilidades y sus colaboradores. (Figura 2)

- Clase: Es cualquier persona, cosa, evento, concepto, pantalla o reporte.
- Responsabilidades: Son sus atributos y métodos.
- Colaboradores: Son las demás clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades.

*Figura 2, Modelo de la tarjeta CRC*




Fuente: *http://www.humbertocervantes.net/homepage/itzamna/DOCUMENTACION/Doc2.html*.

## TARJETAS CRC

| **Confirmacion** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - Paso : int<br>- Carácter : String<br>- referencia_x : int<br>- referencia_y : int<br>- pos_ahora_x : int<br>- pos_ahora_y: int<br>- numero : int<br>- totalNumeros : int<br>- veces_oprimidas : int<br>- i : int<br>- estado_letra : String<br>- array_po_unos : ArrayList <Integer><br>- num : int<br>- combinacion : String<br>- combinacion2 : String<br>- nombre_tabla : String<br>- contador : int<br>- real : int<br>- no_su : int<br>- dato : String<br>- andar : String<br>- index_array : ArrayList <String><br>- acumulador : int<br>- detener : boolean<br>- fecha : String | - onCreate(Bundle) : void<br>- click(View) : void<br>- encontrar_primer_boton(): void<br>- localizar_primera_coordenada() : void<br>- encontrar_cada_boton(): void<br>- encontrar_total_unos(): void<br>- setear(): void<br>- animar(View): void<br>- devolverse(): void<br>- pasar_confirmacion(): void<br>- consulta_base_datos(): void<br>- buscar_index(): void<br>- arrancarConteo(): void<br>- pasar_PorTiempo(): void<br>- consultar() : void<br>- onKeyDown(int, KeyEvent) : boolean | - Manager: Tablas<br>- cursor : Cursor<br>- cursor2 : Cursor<br>- adapter: SimpleCursorAdapter<br>- ttsManager: TTSManager<br>- stringBuilder : StringBuilder<br>- mp: MediaPlayer<br>- combinación_cargada: String<br>- b1 : Button<br>- b2 : Button<br>- b3 : Button<br>- b4 : Button<br>- b5 : Button<br>- b6 : Button<br>- letra_mostrada: TextView<br>- MainLayout : RelativeLayout |

| **EleccionTeclado** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - pagina : int<br>- numero : int<br>- NumberOfPages : int<br>- res : String []<br>- backgroundcolor : int []<br>- backgroundcolor2 : int []<br>- bandera : boolean<br>- teclado1 : boolean<br>- teclado2 : boolean<br>- paso : int | - onCreate (Bundle): void<br>- pasar (View): void<br>- arrancar_conteo_inicio(): void<br>- onKeyDown (int, KeyEvent): boolean<br>- atrás(): void<br>- devolverse(): void | - ttsManager : TTSManager<br>- viewPager : ViewPager<br>- myPagerAdapter : MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListene |

| **Base_de_datos** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - DB_NAME: String<br>- DB_SCHEME_VERSION : int | - onCreate: (SQLiteDataBase) : void<br>- onUpgrade (SOLiteDatabase, int, int) : void | - Tablas : class |

| **EleccionModulos** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - pagina :int<br>- numero : int<br>- NumberOfPages : int<br>- res : String []<br>- backgroundcolor : int []<br>- backgroundcolor2 : int [] | - onCreate: (Bundle): void<br>- pasar (View): void<br>- arrancar_conteo_inicio(): void<br>- onKeyDown (int, KeyEvent): boolean | - ttsManager : TTSManager<br>- viewPager : ViewPager<br>- myPagerAdapter : MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListener |

| **Grilla** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - Letters : String[] | - onCreate(Bundle) : void<br>- consulta_base_datos : void | - Manager: Tablas |

| **Splash** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | - Manager: Tablas |

| | | |
|---|---|---|
| | - onCreate(Bundle) : void | - cursor : Cursor<br>- ttsManager: TTSManager |

| **EleccionVelocidad** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - pagina : int<br>- numero : int<br>- NumberOfPages : int<br>- res : String[]<br>- backgroundcolor : int[]<br>- backgroundcolor2 : int[]<br>- bandera : boolean<br>- teclado1 : boolean<br>- teclado2 : boolean<br>- paso : int | - onCreate:(Bundle): void<br>- pasar(View) :void<br>- arrancar_conteo_inicio(): void<br>- onKeyDown(int, KeyEvent): boolean<br>- atrás(): void<br>- devolverse(): void | - ttsManager : TTSManager<br>- viewPager : ViewPager<br>- myPagerAdapter : MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListene |

| **Escritura_segundo** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - bandera : boolean<br>- Letra_a_reproducirse: String<br>- estado_btn_1 : int<br>- estado_btn_2 : int<br>- estado_btn_3 : int<br>- estado_btn_4 : int<br>- estado_btn_5 : int<br>- estado_btn_6 : int<br>- estado_run : int<br>- resultado_combinacion : String<br>- cadena_binaria : String<br>- numero_prueba : int<br>- cadena: String<br>- MainLayout : RelativeLayout<br>- paso : int | - onCreate(Bundle) : void<br>- onCreateOptionsMenu(Menu) : boolean<br>- hacer_click(View) : void<br>- enviar() : void<br>- mostrar_cadena(String) :void<br>- esperar() : void<br>- esperar_para_enter() : void<br>- correr_hilo(View) : void<br>- acciones() :void<br>- concatenar (int, int, int, int, int, int) : String<br>- consulta_base_datos(): void<br>- hablar(String) : void<br>- concatenar_palabra(String): void<br>- atras() : void<br>- devolverse() : void<br>- animar(View) : void | - Manager : Tablas<br>- cursor : Cursor<br>- adapter: SimpleCursorAdapter<br>- ttsManager: TTSManager<br>- b1 : Button<br>- b2 : Button<br>- b3 : Button<br>- b4 : Button<br>- b5 : Button<br>- b6 : Button<br>- boton_espacio : Button<br>- boton_borrar : Button<br>- mostrar : TextView<br>- hilo_prueba : TextView<br>- letra_mostrada: TextView |

| **Letras** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - numero : int<br>- a : ArrayList<String><br>- paso : int<br>- res : String[]<br>- backgroundcolor : int[]<br>- backgroundcolor2 : int[]<br>- pagina : int<br>- nom_letra : String | - onCreate: (Bundle): void<br>- consulta_base_datos() : void<br>- pasar (View): void<br>- pasar2 (View): void<br>- consulta_base_datos_inicio(View) : void<br>- atrás() : void<br>- devolverse (): void<br>- arrancar_conteo_inicio() : void<br>- onKeyDown (int, KeyEvent) : boolean | - ttsManager : TTSManager<br>- manager : Tablas<br>- cursor : Cursor<br>- cursor1 : Cursor<br>- viewPager : ViewPager<br>- myPagerAdapter : MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListene |

| **Llaves** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - mp : MediaPlayer<br>- combinación_cargada: String<br>- nombre_tabla : String<br>- combinacion : String<br>- real : int<br>- referencia_x : int<br>- referencia_y : int<br>- pos_ahora_x : int<br>- pos_ahora_y: int<br>- numero : int<br>- totalNumeros : int<br>- veces_oprimidas : int<br>- i : int<br>- num : int<br>- veces_fallidas : int<br>- fecha : String<br>- estado_letra : String<br>- array_po_unos : ArrayList<Integer><br>- cantidad_logrado : int<br>- contador : int<br>- detener : boolean | - onCreate(Bundle) : void<br>- click(View) : void<br>- encontrar_primer_boton() : void<br>- localizar_primera_coordenada() : void<br>- encontrar_cada_boton() : void<br>- encontrar_total_unos() : void<br>- consulta_base_datos_inicio() : void<br>- setear() : void<br>- animar(View) : void<br>- arrancarConteo() : void<br>- pasar_PorTiempo() : void<br>- onKeyDown(int, KeyEvent) : boolean | - Manager: Tablas<br>- cursor : Cursor<br>- stringBuilder : StringBuilder<br>- b1 : Button<br>- b2 : Button<br>- b3 : Button<br>- b4 : Button<br>- b5 : Button<br>- b6 : Button<br>- letra_mostrada: TextView<br>- MainLayout : RelativeLayout |

| MenuAprendizaje | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - paso : int<br>- numero : int<br>- NumberOfPage : int<br>- res : String []<br>- backgroundcolor : int[]<br>- backgroundcolor2 : int[] | - onCreate: (Bundle): void<br>- pasar (View): void<br>- pasar_numeros(View) : void<br>- pasar_vocales(View) : void<br>- pasar_matem(View) : void<br>- pasar_ortogr(View) : void<br>- pasar_dieresis(View) : void<br>- arrancar_conteo_inicio() : void<br>- onKeyDown (int, KeyEvent) : boolean<br>- atrás() : void<br>- devolverse (): void | - ttsManager : TTSManager<br>- manager : Tablas<br>- viewPager : ViewPager<br>- myPagerAdapter : MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListene |

| Numeros | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - Valor_texto : String<br>- Numero_nuevo : String<br>- numero : int<br>- paso : int<br>- pagina : int<br>- res : String []<br>- backgroundcolor : int []<br>- backgroundcolor2 : int []<br>- a : ArrayList<String> | - onCreate: (Bundle): void<br>- pasar (View) : void<br>- sonar (int) : void<br>- atras () : void<br>- devolverse () : void<br>- arrancar_conteo_inicio() : void<br>- onKeyDown (int, KeyEvent) : boolean | - ttsManager : TTSManager<br>- manager : Tablas<br>- viewPager: ViewPager<br>- myPagerAdapter: MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListene |

| Operaciones_menu | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |

| | | |
|---|---|---|
| - Cantidad_valores : int<br>- Valor_texto : String<br>- Numero_nuevo : String<br>- numero : int<br>- a: ArrayList<String><br>- paso : int<br>- res : String[]<br>- backgroundcolor : int []<br>- backgroundcolor2 : int []<br>- nom_letra : String<br>- pagina : int | - onCreate: (Bundle): void<br>- consulta_base_datos() : void<br>- pasar (View): void<br>- pasar2 (View): void<br>- consulta_base_datos_inicio (View) : void<br>- atras (): void<br>- devolverse (): void<br>- arrancar_conteo_inicio() : void<br>- onKeyDown (int, KeyEvent) : boolean | - ttsManager : TTSManager<br>- manager : Tablas<br>- cursor : Cursor<br>- cursor1 : Cursor<br>- viewPager: ViewPager<br>- myPagerAdapter: MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListene |

| **Ortograficos** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - paso : int<br>- res : String[]<br>- backgroundcolor : int[]<br>- backgroundcolor2 : int[]<br>- pagina : int<br>- nom_letra: String<br>- numero : int<br>- a: ArrayList<String> | - onCreate: (Bundle): void<br>- cosulta_base_datos(): void<br>- pasar (View):void<br>- pasar2 (View):void<br>- cosulta_base_datos_inicio(View): void<br>- atras () : void<br>- devolverse (): void<br>- arrancar_conteo_inicio() : void<br>- onKeyDown (int, KeyEvent) : boolean | - ttsManager : TTSManager<br>- manager : Tablas<br>- cursor:Cursor<br>- cursor1:Cursor<br>- viewPager: ViewPager<br>- myPagerAdapter: MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListene |

| **Letras** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |

| | | |
|---|---|---|
| - Cantidad_valores : int<br>- Valor_texto : String<br>- Numero_nuevo : int<br>- paso : int<br>- a: ArrayList<String><br>- res : String []<br>- backgroundcolor : int []<br>- backgroundcolor2 : int []<br>- nom_letra : String<br>- pagina :int | - onCreate: (Bundle): void<br>- consulta_base_datos() : void<br>- pasar (View):void<br>- pasar2 (View):void<br>- consulta_base_datos_inicio(View) : void<br>- atrás() : void<br>- devolverse (): void<br>- arrancar_conteo_inicio() : void<br>- onKeyDown (int, KeyEvent) : boolean | - ttsManager : TTSManager<br>- manager : Tablas<br>- cursor : Cursor<br>- cursor1: Cursor<br>- viewPager: ViewPager<br>- myPagerAdapter: MyPagerAdapter<br>- myOnPageChangeListener : OnPageChangeListene |

| **Tablas** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - TABLE_NAME: String<br>- BR_ID : String<br>- BR_COMBINACION: String<br>- BR_NAME : String<br>- BR_WIN : String<br>- CREATE_TABLE : String<br>- TABLE_NAME_G: String<br>- GR_ID : String<br>- GR_WIN : String<br>- CREATE_TABLE_G : String<br>- TABLE_NAME_N : String<br>- NR_ID : String<br>- NR_COMBINATION : String<br>- NR_NAME : String<br>- NR_WIN : String<br>- CREATE_TABLE_NUM : String<br>- TABLE_NAME_V : String<br>- VR_ID : String<br>- VR_COMBINATION : String<br>- VR_NAME : String<br>- VR_WIN : String<br>- CREATE_TABLE_V : String<br>- TABLE_NAME_S : String<br>- SR_ID : String<br>- SR_COMBINATION : String<br>- SR_NAME : String<br>- SR_WIN : String | - generarContentValues (String, String, String) : ContentValues<br>- insertar (String, String, String) : void<br>- actualizar (String, String) : void<br>- actualizar_por_index(String, String) : void<br>- buscarCombinacion (int) : Cursor<br>- buscarPorEstado (String) : Cursor<br>- actualizar_estado() : void<br>- buscarCombinacion1(String) : Cursor<br>- generarContentValues(String) : ContentValues<br>- insertar_g (String) : void<br>- actualizar_g (String, String) : void<br>- buscarEstado (String) : Cursor<br>- generarContentValues_N (String, String, String ) : ContentValues<br>- insertar_N (String, String , String) : void<br>- actualizar_N(String, String) : void<br>- actaulizar_por_numero_N(String, String) : void<br>- buscarCombinacion_N (int) : Cursor<br>- buscarPorEstado_N (String) : Cursor<br>- generarContentValues_V(String, String, String ) : ContentValues<br>- insertar_V (String, String , String) : void<br>- actualizar_V(String, String) : void | - helper : Base_de_datos<br>- db : SQLiteDatabase |

| | | |
|---|---|---|
| - CREATE_TABLE_S : String<br>- TABLE_NAME_O : String<br>- OR_ID : String<br>- OR_COMBINATION : String<br>- OR_NAME : String<br>- OR_WIN : String<br>- CREATE_TABLE_O: String<br>- TABLE_NAME_E : String<br>- ER_ID : String<br>- ER_COMBINATION : String<br>- ER_NAME : String<br>- ER_DATE : String<br>- ER_TOUCH : String<br>- ER_TIME : String<br>- ER_TABLE : String<br>- ER_WIN : String<br>- CREATE_TABLE_E : String<br>- TABLE_NAME_D : String<br>- DR_ID : String<br>- DR_COMBINATION : String<br>- DR_NAME : String<br>- DR_WIN : String<br>- CREATE_TABLE_D : String | - actualizar_por_index_V(String, String) : void<br>- buscarCombinacion_V(int) : Cursor<br>- buscarPorEstado_V (String) : Cursor<br>- actualizar_estado_V() : void<br>- generarContentValues_S(String, String, String ) : ContentValues<br>- insertar_S (String, String , String) : void<br>- actualizar_S(String, String) : void<br>- actualizar_por_index_S(String, String) : void<br>- buscarCombinacion_S(int) : Cursor<br>- buscarPorEstado_S (String) : Cursor<br>- actualizar_estado_S() : void<br>- generarContentValues_O(String, String, String ) : ContentValues<br>- insertar_O(String, String , String) : void<br>- actualizar_O(String, String) : void<br>- actualizar_por_index_O(String, String) : void<br>- buscarCombinacion_O(int) : Cursor<br>- buscarPorEstado_O (String) : Cursor<br>- actualizar_estado_O() : void<br>- generarContentValues_E(String, String, String ) : ContentValues<br>- insertar_E(String, String , String, String, String , String) : void<br>- buscarPorTabla_E(String) : Cursor<br>- generarContentValues_D(String, String, String ) : ContentValues<br>- insertar_D(String, String , String) : void<br>- actualizar_D(String, String) : void<br>- actualizar_por_index_D(String, String) : void<br>- buscarCombinacion_D(int) : Cursor<br>- buscarPorEstado_D (String) : Cursor<br>- actualizar_estado_D() : void | |

| **Teclado_aprendizaje** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | - Manager : Tablas |
| - paso : int | - onCreate(Bundle) : void | |

| | | |
|---|---|---|
| - tipo : String<br>- letra_mostrada : TextView<br>- adapter : SimpleCursorAdapter<br>- real : int<br>- encontrar_siguiente_3350 : int<br>- num : int<br>- estado_letra : String<br>- referencia_x : int<br>- referencia_y : int<br>- pos_ahora_x : int<br>- pos_ahora_y: int<br>- numero : int<br>- totalNumeros: int<br>- veces_oprimidas: int<br>- i : int<br>- combinación_cargada : String<br>- array_po_unos : ArrayList<Integer><br>- stringBuilder : StringBuilder<br>- dato : String<br>- encontrar_siguiente : String h : int | - click(View) : void<br>- encontrar_primer_boton(): void<br>- localizar_primera_coordenada() : void<br>- encontrar_cada_boton(): void<br>- encontrar_total_unos(): void<br>- consulta_base_datos_inico(int): void<br>- setear(): void<br>- animar(View): void<br>- atras(View) : void<br>- devolverse(): void<br>- consulta_base_datos(): void<br>- consulta_base_datos_final(): void<br>- pasar_confirmacion(): void<br>- traer (int) : void<br>- arrancarConteo(String, String): void<br>- hacer_sonar() : void<br>- onKeyDown(int, KeyEvent) : boolean | - cursor : Cursor<br>- ttsManager: TTSManager<br>- b1 : Button<br>- b2 : Button<br>- b3 : Button<br>- b4 : Button<br>- b5 : Button<br>- b6 : Button<br>- MainLayout : RelativeLayout<br>- mp : MediaPlayer |

| **Confirmacion** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - Paso: int<br>- Carácter: String<br>- referencia_x : int<br>- referencia_y : int<br>- pos_ahora_x : int<br>- pos_ahora_y: int<br>- numero : int<br>- totalNumeros: int<br>- veces_oprimidas: int<br>- i : int<br>- estado_letra : String<br>- array_po_unos : ArrayList<Integer><br>- num : int<br>- combinacion : String<br>- combinacion2 : String<br>- nombre_tabla : String<br>- contador : int | - onCreate(Bundle) : void<br>- click(View) : void<br>- encontrar_primer_boton(): void<br>- localizar_primera_coordenada() : void<br>- encontrar_cada_boton(): void<br>- encontrar_total_unos(): void<br>- setear(): void<br>- animar(View): void<br>- devolverse(): void<br>- pasar_confirmacion(): void<br>- consulta_base_datos(): void<br>- buscar_index(): void<br>- arrancarConteo(): void<br>- pasar_PorTiempo(): void<br>- consultar() : void<br>- onKeyDown(int, KeyEvent) : boolean | - Manager: Tablas<br>- cursor : Cursor<br>- cursor2 : Cursor<br>- adapter: SimpleCursorAdapter<br>- ttsManager: TTSManager<br>- stringBuilder : StringBuilder<br>- mp: MediaPlayer<br>- combinación_cargada: String<br>- b1 : Button<br>- b2 : Button<br>- b3 : Button<br>- b4 : Button<br>- b5 : Button<br>- b6 : Button |

| | | |
|---|---|---|
| - real:int<br>- no_su:int<br>- dato : String<br>- andar : String<br>- index_array : ArrayList<String><br>- acumulador : int<br>- detener : boolean<br>- fecha : String | | - letra_mostrada: TextView<br>- MainLayout : RelativeLayout |

| **TTSManager** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - isLoaded : boolean | - Init(Context) : void<br>- shutDown() : void<br>- addQueue(String) : void<br>- initQueue(String) : void | - mTts : TexToSpeech |

| **Escritura_primer** | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - Letra_a_reproducirse : String<br>- estado_btn_1 : int<br>- estado_btn_2 : int<br>- estado_btn_3 : int<br>- estado_btn_4 : int<br>- estado_btn_5 : int<br>- estado_btn_6 : int<br>- estado_run : int<br>- resultado_combinacion: String<br>- cadena_binaria : String<br>- numero_prueba : int<br>- cadena: String | - onCreate(Bundle) : void<br>- onCreateOptionsMenu (Menu) : boolean<br>- hacer_click(View) : void<br>- enviar() : void<br>- mostrar_cadena(String): void<br>- esperar(): void<br>- correr_hilo(View) : void<br>- acciones() : void<br>- concatenar (int, int, int, int, int, int) : String<br>- consulta_base_datos(): void<br>- hablar(String) : void<br>- concatenar_palabra(String): void<br>- atras () : void<br>- devolverse () : void<br>- onKeyDown (int, KeyEvent) : boolean<br>- animar(View) : void<br>- esperar_para_enter() : void | - Manager : Tablas<br>- cursor : Cursor<br>- cursor2 : Cursor<br>- adapter : SimpleCursorAdapter<br>- ttsManager : TTSManager<br>- stringBuilder : StringBuilder<br>- b1 : Button<br>- b2 : Button<br>- b3 : Button<br>- b4 : Button<br>- b5 : Button<br>- b6 : Button<br>- boton_espacio : Button<br>- boton_borrar : Button<br>- mostrar : TextView<br>- hilo_prueba : TextView<br>- letra_mostrada : TextView<br>- MainLayout : RelativeLayout |

| Singo_generador | | |
|---|---|---|
| Responsabilidades | | Colaboradores |
| Atributos | Métodos | |
| - tipo : String<br>-combinación_cargada: String<br>- referencia_x : int<br>- referencia_y : int<br>- pos_ahora_x : int<br>- pos_ahora_y: int<br>- numero : int<br>- totalNumeros : int<br>- veces_oprimidas: int<br>- i : int<br>- num : int<br>- veces_fallidas : int<br>- fecha : String<br>- estado_letra : String<br>- array_po_unos : ArrayList<Integer><br>- cantidad_logrado : int<br>- contador : int<br>- detener : boolean<br>- array_po_unos : ArrayList<Integer > | - onCreate(Bundle) : void<br>- click(View) : void<br>- encontrar_primer_boton(): void<br>- localizar_primera_coordenada() : void<br>- encontrar_cada_boton(): void<br>- encontrar_total_unos(): void<br>- consultar_base_datos_inicio(Strig): void<br>- setear(): void<br>- animar(View): void<br>- arrancarConteo_inicio(): void<br>- arrancarConteo(): void<br>- pasarPorTiempo(): void<br>- onKeyDown(int, KeyEvent) : boolean | - mp: MediaPlayer<br>- Manager: Tablas<br>- cursor : Cursor<br>- ttsManager: TTSManager<br>- stringBuilder : new StringBuilder<br>- b1 : Button<br>- b2 : Button<br>- b3 : Button<br>- b4 : Button<br>- b5 : Button<br>- b6 : Button<br>- letra_mostrada : TextView<br>- MainLayout : RelativeLayout |

## 3. FASE DE PRUEBA PILOTO.

Para esta fase se realizaron pruebas al prototipo, con dos tipos de destinatarios, personas con alguna discapacidad visual y personas que no tenían discapacidad visual.

### Personas con Discapacidad Visual.

*Imagen 1 – Ubicación del Dispositivo*

*Fuente: Autores.*

*Imagen 2 – Manejo de la aplicación*

*Fuente: Autores.*

*Imagen 3 – Prueba de la aplicación*

*Fuente: Autores*

## 4. FASE DE IMPLANTACIÓN.

Incluir los requerimientos de software y de hardware para la instalación y posterior puesta en marcha.

**Requerimientos:**

| | |
|---|---|
| **Sistema operativo** | Android ® (igual o superior al api 15) |
| **Memoria** | Se recomienda 5 MB |
| **Pantalla** | Se necesita una pantalla para celulares de mínimo 3.5”. |
| **Aplicaciones** | - Lector de archivos PDF (por ejemplo Acrobat Reader).<br>- Reproductor de sonido.<br>- Internet, el dispositivo debe tener acceso a Internet, ya que la aplicación detecta si el dispositivo tiene o no el sintetizador de voz, si no lo tiene entonces lo descarga automáticamente.<br>- Talk back |

# 5. METODOLOGÍA

## Metodología de Desarrollo

Para el desarrollo del presente trabajo se eligió la metodología ágil Extreme Programming (XP) fundamentada en una serie de valores y prácticas de buenas maneras que tiene como objetivo aumentar la productividad a la hora de desarrollar programas.

En febrero de 2001, en las montañas Wasatch de Utah, se reunieron 17 desarrolladores convencidos de que era necesario un cambio en las metodologías "clásicas" de desarrollo de software. Entre ellos se encontraban los creadores de XP, SCRUM, DSDM, Adaptive Software Development, Crystal, Feature-Driven Development y Pragmatic Programming. Juntos proclamaron lo que se ha dado a conocer como el "Manifesto for Agile Software Development", estableciendo en él cuatro principios30:

- Se valora a los individuos y las interacciones sobre los procesos y herramientas.
- Se valora a las aplicaciones que funcionan sobre la documentación exhaustiva.
- Se valora la colaboración del cliente sobre las negociaciones contractuales.
- Se valora la respuesta al cambio sobre el seguimiento de un plan.

Esto significa que, sin desconocer el valor de los segundos ítems, se valoran más los primeros.

Las fases que se plantearon para el presente proyecto son:

## 5.1 PLANEACIÓN

La planeación es la etapa inicial de todo proyecto en XP. En este punto se comienza a interactuar con el cliente y el resto del grupo de desarrollo para descubrir los requerimientos del sistema. En este punto se identifican el número y tamaño de las

30 *http://eisc.univalle.edu.co/materias/WWW/material/lecturas/xp.pdf*

iteraciones al igual que se plantean ajustes necesarios a la metodología según las características del proyecto.

En este apartado se tendrán en cuenta cuatro elementos, los cuales son los siguientes. Historias de usuario, velocidad del proyecto, iteraciones y entregas pequeñas.

Historias de usuario

**Iteración 1** (Dos semanas)

- Desarrollar introducción al sistema de escritura, mediante curso de manejo introductorio.

**Iteración 2** (Tres semanas)

- Desarrollar el teclado.
- Vincular el sintetizador de voz con el teclado.

**Iteración 3** (Dos semanas)

- Desarrollar menús mediante interfaz vinculado con el sintetizador de voz.
- Implementar el teclado en la interfaz.

**Iteración 4** (Tres semanas)

- Vincular ayudas para que el usuario con discapacidad visual se pueda mover con comodidad y correctamente por el sistema.

**Iteración 5** (Dos semanas)

- Añadir teoría del sistema Braille.

**Iteración 6** (Cuatro semanas)

- Desarrollar cada una de las actividades de entretenimiento basado en el sistema Braille.
- Desarrollar pruebas de conocimiento para el usuario.

**Iteración 7** (Una semana y cuatro días)

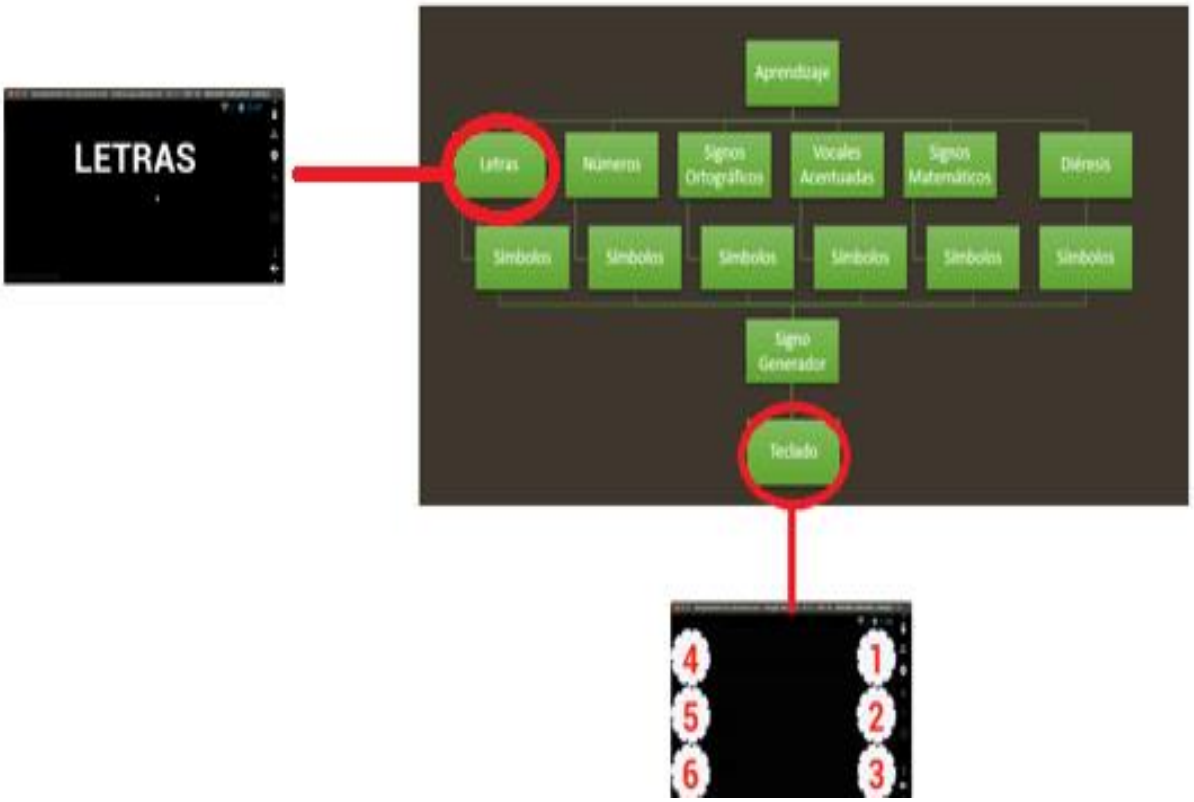
- Desarrollar manuales en PDF y mediante voz.

## 5.2 DISEÑO

En XP solo se diseñan aquellas historias de usuario que el cliente ha seleccionado para la iteración actual por dos motivos: por un lado se considera que no es posible tener un diseño completo del sistema y sin errores desde el principio. El segundo motivo es que dada la naturaleza cambiante del proyecto, el hacer un diseño muy extenso en las fases iniciales del proyecto para luego modificarlo, se considera un desperdicio de tiempo.

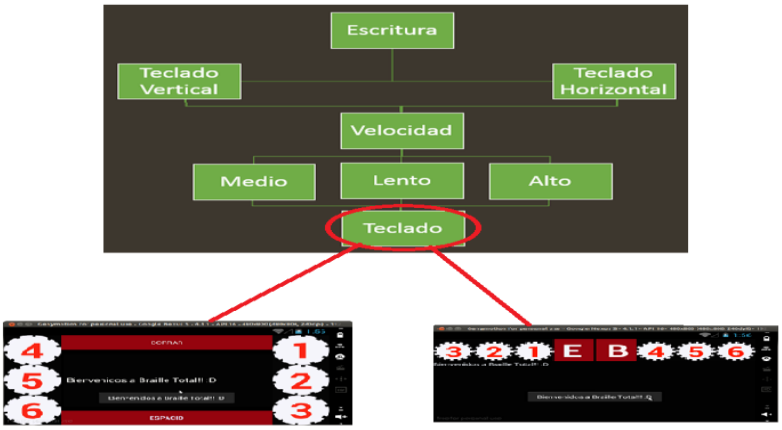
Es importante resaltar que esta tarea es permanente durante la vida del proyecto partiendo de un diseño inicial que va siendo corregido y mejorado en el transcurso del proyecto. Para la fase de diseño se plantearon tres estructuras, una para cada módulo de la aplicación.

*Imagen 4 - Estructura modulo aprendizaje.*



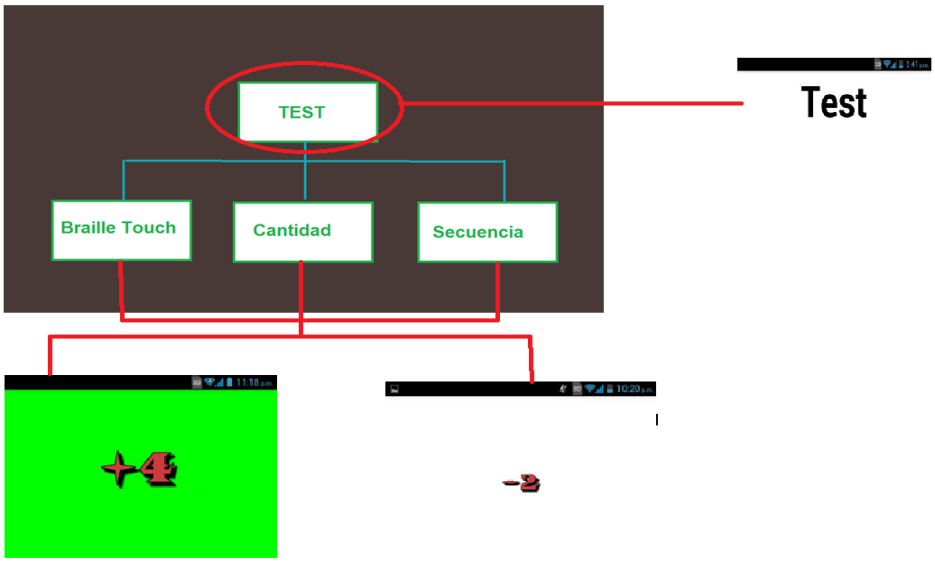

Fuente: Autores

Imagen 5 - Estructura Modulo Escritura




Fuente: Autores

Imagen 6 - Estructura modulo test.




Fuente: Autores

## 5.2 CODIFICACION

La codificación es un proceso que se realiza en forma paralela con el diseño y la cual está sujeta a varias observaciones por parte de XP consideradas controversiales por algunos expertos tales como la rotación de los programadores o la programación en parejas.

Además de los mencionados temas, el lector encontrará a continuación una descripción de los siguientes temas: cliente siempre presente, codificar primero la prueba, integración secuencial e integraciones frecuentes.

Implementación de fase de diseño y codificación.

**Iteración 1**

Es de importancia realizar correctamente la implementación de la librería TTS para que el curso de manejo introductorio resulte dinámico y fácil de comprender.

**Iteración 2** (Tres semanas)

Es necesario que el teclado llame varios recursos que el hardware proporciona.

**Iteración 3** (Dos semanas)

Es necesario que los menús de la interfaz no tengan muchas opciones, con esto se previene que se haga la adaptación del software más complicada ante el usuario.

Es conveniente que el teclado adaptado a la interfaz, se adapte a la posición a la que se encuentre el móvil para que el sistema de escritura no se vea afectado.

**Iteración 4** (Tres semanas)

Para la edición de documentos es importante que el usuario pueda interactuar con el documento mediante el sintetizador de voz, es decir que a medida que el usuario elija el texto, el sintetizador de voz le haga saber el contenido de este.

Las ayudas para el usuario tienen que estar siempre en cada uno de los módulos de la interfaz.

**Iteración 5** (Dos semanas)

La información que se ingrese en el sistema (todo lo relacionado con el sistema braille) tiene que ir referenciada.

**Iteración 6** (Cuatro semanas)

Las actividades que se desarrollen tienen que manejar conceptos del sistema Braille con todas las reglas de sintaxis de dicho sistema.

Las pruebas tienen que ir con puntajes que el usuario va a ingresar conforme va desarrollando los test.

**Iteración 7** (Una semana y cuatro días)

Los manuales tienen que estar en formato PDF y en formato Mp3.

### 5.4 PRUEBAS

XP enfatiza mucho los aspectos relacionados con las pruebas, clasificándolas en diferentes tipos y funcionalidades específicas, indicando quién, cuándo y cómo deben ser implementadas y ejecutadas.

Cuando se tienen bien implementadas las pruebas no habrá temor de modificar el código del otro programador en el sentido que si se daña alguna sección, las pruebas mostrarán el error y permitirán encontrarlo. Uno de los elementos que podría obstaculizar que un programador cambie una sección de código funcional es precisamente hacer que esta deje de funcionar. Si se tiene un grupo de pruebas que garantice su buen funcionamiento, este temor se mitiga en gran medida.

Según XP se debe ser muy estricto con las pruebas. Sólo se deberá liberar una nueva versión si esta ha pasado con el cien por ciento de la totalidad de las pruebas. En caso contrario se empleará el resultado de estas para identificar el error y solucionarlo con mecanismos ya definidos.

| Acción | Datos | Resultado esperado |
|---|---|---|
| **Escuchar curso introductorio** | Archivo mp3 | Archivo mp3 valido y es posible reproducir pausar y parar. |
| **Escribir en teclado** | Texto | El texto va acorde con la información del sistema Braille. |
| | Texto | Cada letra que se escribe tiene que ir de la mano con el sintetizador de audio. |
| | Texto | El teclado tiene que tener en cada botón un hilo que permitirá el uso óptimo de la escritura. |
| **Moverse por los menús** | Interfaz | La interfaz tiene que tener la ayuda sintetizada por voz. |
| | Interfaz | Cada uno de los recursos gráficos de la interfaz tiene que tener sintetizado por voz |

| | | |
|---|---|---|
| | | para cada una de las acciones. |
| | Interfaz | La interfaz no puede tener muchas opciones para la comodidad del usuario. |
| | Interfaz | Para cada paso de una interfaz a otra el sintetizador de voz tiene que mostrar las acciones sintetizadas. |
| **Almacenar documentos (elegir tipo de memoria)** | Espacio de memoria | Tiene que haber espacio en memoria. |
| | Espacio de memoria | La memoria tiene que estar disponible. |
| **Editar documentos** | Documento | El nombre y ruta del documento debe ser comunicado al usuario mediante el sintetizador de voz. |
| | Documento | Cada una de las partes del sistema a la que el usuario tenga acceso tiene que ser informada mediante el sintetizador de voz. |
| | Documento | El usuario accede a la parte que el mismo desee mediante el tacto. |
| | Documento | El usuario edita el documento accediendo al teclado. |
| | Documento | El usuario guarda la actualización del documento en el espacio de memoria y nombre que elija. |
| **Acceder a ayuda** | Archivo mp3 | Cada archivo mp3 seleccionado se escucha. |
| **Acceder a actividad** | Actividad entretenimiento | La actividad permite puntajes. |
| | Actividad entretenimiento | La actividad se abre normalmente. |
| | Actividad entretenimiento | La actividad interactúa correctamente con los |

| | | |
|---|---|---|
| | | recursos de hardware (vibración, acelerómetro, etc.) |
| **Abrir manual** | Archivo mp3 | El archivo mp3 se reproduce e interactúa normalmente con el usuario (reproduce, pausa, para). |
| | Archivo PDF | El archivo PDF se abre. |

**5.5 ROLES:**

En Xp existen diferentes roles y responsabilidades para llevar a cabo las tareas y los planes durante el proceso de desarrollo:

- Programmer (Programador): Es el responsable del diseño, programación y realización de las pruebas; decisiones técnicas y construcción del sistema, no tiene distinción entre analista, diseñador o codificador.

Sebastián Suarez y Erika Guzman fueron los encargados de diseñar, escribir y realizar las pruebas a la aplicación.

- Customer (Cliente): Es parte del equipo, en todo proyecto debe existir un cliente; tiene como tarea establecer las pruebas de aceptación que son las que determinan si el sistema cumple con sus requerimientos**.**

Sebastián Suarez, Erika Guzman, fueron los encargados de realizar las historias de usuarios y las pruebas funcionales.

- Coach (Entrenador): Es el principal responsable del proceso y líder del equipo, toma las decisiones importantes y tiende a estar en un segundo plano a medida que el equipo madura.

José Vicente Reyes fue el responsable del proceso, brindando asesorías y seguimiento al desarrollo del proyecto.

- Tester (Encargado de las pruebas): Ayuda al cliente con las pruebas funcionales, asegurándose de que sean ejecutadas**.**

Los programadores, INCI, CRAC fueron los encargados de la realización de las pruebas de la aplicación.

## HISTORIAS DE USUARIO

Representan una breve descripción del comportamiento del sistema, emplea terminología del cliente sin lenguaje técnico, se realiza una por cada característica principal del sistema, se emplean para hacer estimaciones de tiempo y para el plan de lanzamientos, reemplazan un gran documento de requisitos y presiden la creación de las pruebas de aceptación.

| | **Historia de Usuario** |
|---|---|
| **Número:** 1 | **Nombre:** Diseño de la aplicación |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Baja | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Baja | **Puntos Reales:** |
| **Descripción** | |
| Se plantea el desarrollo de una aplicación móvil que contenga la simbología del sistema Braille, en donde el usuario cuente con tres módulos (aprendizaje, escritura, test), la interfaz para el usuario final será agradable y de fácil manejo, se manejaran los colores blanco y negro para las pantallas y las letras. | |
| **Observaciones** | |
| - Que tenga incluida la base de datos. | |

| | **Historia de Usuario** |
|---|---|
| **Número**: 2 | **Nombre:** Crear Base de Datos |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 1 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| Crear las tablas necesarias, definir llaves primarias y foráneas y por ultimo definir las relaciones entre las tablas. | |
| **Observaciones** | |
| Incluir la base de datos en la aplicación. | |

| | |
|---|---|
| | **Historia de Usuario** |
| **Número**: 3 | **Nombre:** Creación de los teclados para escritura |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 3 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| El usuario encontrara diferentes opciones de teclado (2), en las cuales tendrá un diseño que será de fácil uso dependiendo de la opción de teclado que se acomode a su gusto.<br>Se carga la opción elegida desde la base de datos, el teclado se encuentra en el modo de escritura indicado para el sistema Braille; si el usuario toca algún lado que no sea el botón correspondiente se le notificara con direcciones (derecha, izquierda, arriba, abajo) en donde se encuentra localizado el botón.<br>Cuando lo oprima correctamente se le notificara con voz el número del botón que oprimió. | |
| **Observaciones** | |
| El usuario contara con ayudas por medio de sonidos.<br>El botón cambiara de color cada vez que se oprima. | |

| | |
|---|---|
| | **Historia de Usuario** |
| **Número:** 4 | **Nombre:** Teclado horizontal de escritura |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 3 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:**2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| Es la segunda opción de teclado con la que contara el usuario, tiene ocho botones ubicados horizontalmente en la parte superior.<br>Contando así con los botones del símbolo generador (seis), el botón de borrado, espacio.<br>Cuando el usuario ingresa a esta opción debe elegir la velocidad de respuesta con la que se sienta más cómodo (lento 7 segundos , medio 5 segundos y rápido 3 segundos)<br>Cada vez que oprime uno de los botones este cambiara de color a verde, si el usuario desea saber que escribió solo debe oprimir unos segundos la pantalla y esta le deletreara lo que tiene escrito, si quiere copiar el texto debe también oprimir la pantalla hasta que le sea notificado que el texto fue copiado. | |
| **Observaciones** | |
| El usuario contara con ayudas por medio de sonidos.<br>El usuario debe estar ubicado en el menú de escritura específicamente en el teclado horizontal.. | |

| | **Historia de Usuario** |
|---|---|
| **Número**: 5 | **Nombre:** Búsquedas |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:**1 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| La aplicación debe buscar la combinación correcta para las letras, números, signos, etc. Al momento en el que el usuario ingrese y/o quiera digitar alguna de las combinaciones existentes. | |
| **Observaciones** | |
| Incluir la base de datos. | |

| | **Historia de Usuario** |
|---|---|
| **Número**: 6 | **Nombre:** Aprendizaje |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 3 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| Al momento en que el usuario ingrese al módulo de aprendizaje, encontrara una interfaz cómoda y de fácil acceso, en la que por medio de ViewPager se deslizara para encontrar las diferentes opciones (letras, números, signos ortográficos, vocales acentuados, signos matemáticos); con ayuda del sintetizador de voz recibirá las indicaciones de la opción que desee, luego saldrá en pantalla un teclado que simula los seis puntos del sistema Braille, cuando este elija un símbolo en el menú esta será buscada en la base de datos, al momento de ingresar a alguna de las opciones encontrara los puntos que generan el símbolo, por ejemplo si ingresa a la letra a, encontrara el símbolo generador que son los puntos 1,2,3,4,5 y 6, luego de que los puntos sean oprimidos le pedirá que oprima el punto uno que es la letra a. | |
| **Observaciones** | |
| - Cuando se mantiene prolongadamente oprimida la pantalla, se elige la opción deseada.<br>- El usuario contara con ayudas por medio de sonidos.<br>- El usuario tendrá el teclado de puntos laterales que será solo para el módulo de aprendizaje.<br>- Al digitar una combinación, el usuario es notificado con un sonido; de si es la correcta o no. | |

| | **Historia de Usuario** |
|---|---|
| **Número**: 7 | **Nombre:** "Navegar" por los menús |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 2 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| Cuando el usuario empieza a manejar la aplicación cuenta con ayudas auditivas, al momento de navegar por cada menú, el sintetizador lo guiara para que oprima el botón correcto y cuando este sea elegido con éxito se producirá un sonido. | |
| **Observaciones** | |
| Si el dispositivo en el que se corre la aplicación no cuenta con la librería del sintetizador de voz, esta será descargada automáticamente. | |

| | **Historia de Usuario** |
|---|---|
| **Número**: 8 | **Nombre:** Inicio de la aplicación |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 4 |
| **Prioridad en Negocio (Alta/Media/Baja):** Media | **Puntos Estimados:** |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Media | **Puntos Reales:** |
| **Descripción.** | |
| Cuando se inicia la aplicación por primera vez, aparece una pantalla divida en 3 opciones (Escritura, lectura y ayuda); se oye un mensaje de ayuda, "Deslice el dedo para escuchar sus opciones”, cuando se va deslizando se oye la opción en la que se está ubicado. | |
| **Observaciones** | |
| La aplicación debe estar corriéndose en el dispositivo. | |

| | **Historia de Usuario** |
|---|---|
| **Número:** 9 | **Nombre:** Opciones de Escritura |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 2 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Media | **Puntos Reales:** |
| **Descripción.** | |
| En el momento en que el usuario empieza a escribir el sintetizador le dirá la letra que se va escribiendo, si desea borrar alguna letra, este también le dirá la letra borrada.<br>Al oprimir prolongadamente sobre la pantalla se escucha lo que se escribió. | |

| Observaciones |
| --- |
| La velocidad al momento en el que se digita la letra es decidida por el usuario. |

| | | Historia de Usuario |
| --- | --- | --- |
| **Número:** 10 | **Nombre:** Elección del modulo | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 2 | |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados: 3** | |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** | |
| **Descripción** | | |
| Se le notifica mediante voz al usuario que ha ingresado al módulo de elección (enseñanza, aprendizaje, test) se muestra mediante una sucesión de paginasen las que el usuario puede avanzar o retroceder según la elección, a medida que pasa las paginas se le va diciendo el ítem en el que se encuentra. | | |
| **Observaciones** | | |
| - Con el paso de las páginas se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla. | | |

| | | Historia de Usuario |
| --- | --- | --- |
| **Número:** 11 | **Nombre:** Aprendizaje – Letras | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 2 | |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 2 | |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** | |
| **Descripción** | | |
| Si es la primera vez que el usuario ingresa a este ítem, aparecerá el signo generador el cual tendrá que digitar en un tiempo de 70 segundos luego podrá empezar con la enseñanza de las letras, la aplicación le dirá en donde se encuentran ubicados los botones, por ejemplo si es la letra a, se le güira para que encuentre arriba a la derecha el botón 1. | | |
| **Observaciones** | | |
| - Con el paso de las páginas se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla. | | |

| |
|---|
| - Al pulsar un botón, su color cambiara a verde, debido a que si la persona tiene baja visión lograra distinguirlo cuando lo oprima. |

| | | |
|---|---|---|
| | | **Historia de Usuario** |
| **Número:** 12 | **Nombre:** Aprendizaje – Números | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | | **Iteración Asignada:** 2 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | | **Puntos Estimados: 1** |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | | **Puntos Reales:** |
| **Descripción** | | |
| Si es la primera vez que el usuario ingresa a este ítem, aparecerá el signo generador "llave" el cual tendrá que digitar para empezar con los números, la aplicación le dirá en donde se encuentran ubicados los botones, por ejemplo si es el número 1, se le güira para que encuentre arriba a la derecha el botón 1. | | |
| **Observaciones** | | |
| - Cuando es superado un número se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla.<br>- Al pulsar un botón, su color cambiara a verde, debido a que si la persona tiene baja visión lograra distinguirlo cuando lo oprima. | | |

| | | |
|---|---|---|
| | | Historia de Usuario |
| **Número:** 13 | **Nombre:** Teclado de Confirmación (Signo generador) | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Media | | **Puntos Estimados:** 1 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Media | | **Puntos Reales:** |
| **Descripción** | | |
| Aparece una sola vez, cuando se ingresa por primera vez a un carácter, se le notificara al usuario que ingreso al símbolo generador y le pedirá al usuario que ingrese la combinación del mismo ósea los seis botones, cuenta con un límite de tiempo que al acabarse o al pulsar correctamente la combinación pasa al carácter elegido. | | |
| **Observaciones** | | |
| - Interfaz a blanco y negro (intercalándose) por si la persona tiene baja visión.<br>- No tiene la opción de regresar. | | |

| |
|---|
| - Al pulsar un botón, su color cambiara a verde, debido a que si la persona tiene baja visión lograra distinguirlo cuando lo oprima. |

| | | |
|---|---|---|
| | | **Historia de Usuario** |
| **Número:** 14 | **Nombre:** Desarrollo de las actividades de entretenimiento | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | | **Iteración Asignada:** 3 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | | **Puntos Reales:** |
| **Descripción** | | |
| Se desarrollan tres tipos de actividades, cantidad, secuencia y confirmación; todas son trabajadas por ViewPager, dependiendo de si el usuario acierta o falla entonces se le suman o restan puntos. | | |
| **Observaciones** | | |
| - Con el paso de las páginas se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla. | | |

| | |
|---|---|
| **Historia de Usuario** | |
| **Número**: 15 | **Nombre:** Desarrollo de manuales PDF |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 1 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| Se desarrollan el manual de usuario, para que el usuario final pueda familiarizarse con la aplicación, entienda como es el proceso de instalación, su funcionalidad, requisitos, etc. y se desarrolla el manual del programador para que si alguien desea tener acceso al código del aplicativo pueda hacerlo, también se le dan especificaciones del mismo | |
| **Observaciones** | |
| - Para regresar debe oprimir dos veces la pantalla.<br>- Cuando pasa de página a página, se le notifica por voz en que ítem se encuentra ubicado. | |

| | |
|---|---|
| **Historia de Usuario** | |
| **Número**: 16 | **Nombre:** Escritura |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 1 |

| | |
|---|---|
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 1 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| Dependiendo de la elección del teclado de escritura, mediante voz se le notificara al usuario que ha ingresado a la opción de teclado, al igual que en módulos anteriores se manejan los colores blanco y negro y que se invierten según la página, cuando va atrás llega a la elección del módulo, y cuando ingresa llega a la elección de la velocidad. | |
| **Observaciones** | |
| - Para regresar debe oprimir dos veces la pantalla.<br>- Cuando pasa de página a página, se le notifica por voz en que ítem se encuentra ubicado. | |

| | |
|---|---|
| | **Historia de Usuario** |
| **Número:** 17 | **Nombre:** Teclado vertical de escritura |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 3 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción.** | |
| Es la primera opción de teclado con la que contara el usuario, tiene ocho botones ubicados en la parte derecha del dispositivo encontrara los botones 1,2 y 3 de arriba hacia abajo del signo generador; en la parte izquierda los botones 4,5 y 6 de arriba hacia abajo por ultimo en el centro superior de pantalla encontrara el botón de borrar y en la parte inferior central el botón de espacio.<br>Cuando el usuario ingresa a esta opción debe elegir la velocidad de respuesta con la que se sienta más cómodo (lento 7 segundos, medio 5 segundos y rápido 3 segundos).<br>Cada vez que oprime uno de los botones este cambiara de color a verde, si el usuario desea saber que escribió solo debe oprimir unos segundos la pantalla y esta le deletreara lo que tiene escrito, si quiere copiar el texto debe también oprimir la pantalla hasta que le sea notificado que el texto fue copiado. | |
| **Observaciones** | |
| El usuario contara con ayudas por medio de sonidos.<br>El usuario debe estar ubicado en el menú de escritura. | |

| | |
|---|---|
| | **Historia de Usuario** |
| **Número:** 18 | **Nombre:** Aprendizaje – Signos Ortográficos |
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción** | |

| |
|---|
| Para encontrar los diferentes signos ortográficos el usuario deberá deslizar el dedo de derecha a izquierda, si es la primera vez que el usuario ingresa a este ítem, aparecerá el signo generador “llave” el cual tendrá que digitar en un tiempo límite de 70 segundos para empezar con los números, después de que digite correctamente el símbolo generador podrá empezar con el signo deseado (la aplicación le dirá en donde se encuentran ubicados los botones), cundo supera correctamente un símbolo se le pide que lo confirme de nuevo y luego continua con otro. |
| **Observaciones** |
| - Cuando es superado un signo se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla.<br>- Al pulsar un botón, su color cambiara a verde, debido a que si la persona tiene baja visión lograra distinguirlo cuando lo oprima.<br>- Pantalla negra, letras blancas. |

| | | **Historia de Usuario** |
|---|---|---|
| **Número:** 19 | **Nombre:** Aprendizaje – Vocales acentuadas | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | | **Puntos Estimados:** 1 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | | **Puntos Reales:** |
| **Descripción** | | |
| Para encontrar las vocales acentuadas el usuario deberá deslizar el dedo de derecha a izquierda, cuando ingrese a uno de las opciones se le dirá los botones que debe digitar para generar la vocal acentuada, después de que digite correctamente los botones, se le pide confirmación y al confirmarlo correctamente se guardara como superado y podrá continuar con la siguiente vocal acentuada. | | |
| **Observaciones** | | |
| - Cuando es superado una vocal se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla.<br>- Al pulsar un botón, su color cambiara a verde, debido a que si la persona tiene baja visión lograra distinguirlo cuando lo oprima.<br>- Se alterna la Pantalla negra o blanca, con un contraste en las letras, dependiendo del color del fondo. | | |

| | **Historia de Usuario** |
|---|---|
| **Número:** 20 | **Nombre:** Aprendizaje – Signos Matemáticos |

| | |
|---|---|
| **Usuario:** Usuario | |
| **Modificación de Historia Número:** | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | **Puntos Estimados:** 1 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | **Puntos Reales:** |
| **Descripción** | |
| Para encontrar los signos matemáticos el usuario deberá deslizar el dedo de derecha a izquierda (ViewPager), cuando ingrese a uno de las opciones se le dirá los botones que debe digitar para generar el signo matemático deseado, después de que digite correctamente los botones, se le pide confirmación y al confirmarlo correctamente se guardara como superado y podrá continuar con el siguiente. | |
| **Observaciones** | |
| - Cuando es superado un signo, se van alternando los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro los botones serán blancos.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla.<br>- Al pulsar un botón, su color cambiara a verde, debido a que si la persona tiene baja visión lograra distinguirlo cuando lo oprima. | |

| | | |
|---|---|---|
| | | **Historia de Usuario** |
| **Número:** 21 | **Nombre:** Test - Cantidad | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | | **Iteración Asignada:** 2 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | | **Puntos Reales:** |
| **Descripción** | | |
| Es la primera actividad con la que cuenta el usuario, consiste en que cuando el usuario ingresa se le pregunta cuantos puntos tiene algún símbolo del sistema Braille, lo que debe hacer el usuario es poner la cantidad de dedos que forman el símbolo, por ejemplo, si se le pregunta cuantos puntos tiene “signo generador”, el debo poner sobre la pantalla seis dedos.<br>Tiene un teclado similar al vertical de escritura la única diferencia es que en este no encuentra los botones de espacio y borrado, por lo que lograra ubicarse rápidamente en él pues ya lo ha trabajado anteriormente. | | |
| **Observaciones** | | |
| - Con el paso de las páginas se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Cuando el usuario acierta se le suman dos puntos, la pantalla se pone verde y sale un +2, cuando falla la pantalla se pone roja y se le restan dos puntos.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla. | | |

| | | |
|---|---|---|
| | | **Historia de Usuario** |
| **Número:** 22 | **Nombre:** Test - Secuencia | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | | **Iteración Asignada:** 2 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | | **Puntos Reales:** |
| **Descripción** | | |
| Es la segunda actividad con la que cuenta el usuario, consiste en que cuando el usuario ingresa se le pide que escuche una secuencia de números (algún símbolo del sistema Braille), lo que debe hacer el usuario es repetir la secuencia de los números que se le dijeron al principio.<br>Tiene un teclado similar al vertical de escritura la única diferencia es que en este no encuentra los botones de espacio y borrado, por lo que lograra ubicarse rápidamente en él pues ya lo ha trabajado anteriormente. | | |
| **Observaciones** | | |
| - Con el paso de las páginas se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Cuando el usuario acierta se le suman dos puntos, la pantalla se pone verde y sale un +2, cuando falla la pantalla se pone roja y se le restan dos puntos.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla. | | |

| | | |
|---|---|---|
| | | **Historia de Usuario** |
| **Número:** 24 | **Nombre:** Diseño del logo | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | | **Iteración Asignada:** 1 |
| **Prioridad en Negocio (Alta/Media/Baja):** Baja | | **Puntos Estimados:** 1 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Baja | | **Puntos Reales:** |
| **Descripción** | | |
| Para el momento en que la aplicación sea instalada en el dispositivo se plantea el logo de la aplicación que serán unas manos sosteniendo un celular. | | |
| **Observaciones** | | |
| | | |

| | | |
|---|---|---|
| | | **Historia de Usuario** |
| **Número:** 23 | **Nombre:** Test – Confirmación | |
| **Usuario:** Usuario | | |
| **Modificación de Historia Número:** | | **Iteración Asignada:** 2 |
| **Prioridad en Negocio (Alta/Media/Baja):** Alta | | **Puntos Estimados:** 2 |
| **Riesgo en Desarrollo (Alta/Media/Baja):** Alta | | **Puntos Reales:** |
| **Descripción** | | |
| Es la última actividad con la que cuenta el usuario, consiste en que cuando el usuario ingresa se le pide que escriba algún símbolo del sistema Braille, tiene un teclado similar al vertical de escritura la única diferencia es que en este no encuentra los botones de espacio y borrado, por lo que lograra ubicarse rápidamente en él pues ya lo ha trabajado anteriormente. | | |
| **Observaciones** | | |
| - Con el paso de las páginas se alternan los colores negro y blanco, para tener un contraste entre ellas, por ejemplo si el fondo es negro las letras serán blancas.<br>- Para ingresar a uno de los módulos el usuario debe oprimir por unos segundos la pantalla.<br>- Cuando el usuario acierta se le suman dos puntos, la pantalla se pone verde y sale un +2, cuando falla la pantalla se pone roja y se le restan dos puntos.<br>- Para salir de un módulo el usuario debe oprimir dos veces la pantalla. | | |

## PRUEBAS DE ACEPTACIÓN

Permite confirmar que la historia ha sido implementada correctamente

| **CASO DE PRUEBA DE ACEPTACIÓN** | |
|---|---|
| Código: 1 | Historia de Usuario: 1 |
| Nombre: Diseño de la Aplicación. | |
| Descripción: La interfaz para el usuario final debe ser agradable y de fácil manejo. | |
| Condiciones de Ejecución: La aplicación debe iniciarse en el dispositivo. | |
| Entrada/Pasos de ejecución:<br>1. Se debe ejecutar la aplicación en el dispositivo.<br>2. Al realizar las pruebas finales se debe evidenciar que la aplicación puede ser manejada y es agradable para el usuario que la usa. | |
| Resultado esperado:<br>Cuando la aplicación este totalmente finalizada se debe evidenciar el cumplimiento de la historia de usuario número 1. | |
| Evaluación de la prueba: Positiva. | |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 2 | Historia de Usuario: 2 |
| Nombre: Creación de la Base de Datos. | |
| Descripción: La aplicación debe tener una base de datos para el uso de la misma. | |
| Condiciones de Ejecución:<br>- La aplicación debe iniciarse en el dispositivo.<br>- La base de datos debe estar creada.<br>- La base de datos debe estar conectada con la aplicación. | |
| Entrada/Pasos de ejecución:<br>1. La aplicación debe estar ejecutándose en el dispositivo.<br>2. Cuando el usuario digite una combinación, esta debe estar en la base de datos, de esta forma será rectificada.<br>3. Debe incluirse en la base de datos, todos los “símbolos” manejados en el sistema Braille. | |
| Resultado esperado:<br>La base de datos debe tener un buen funcionamiento y debe contar con la terminología necesaria del sistema Braille. | |
| Evaluación de la prueba: Positiva | |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 3 | Historia de Usuario: 3 |
| Nombre: Creación de los teclados de escritura. | |
| Descripción: La aplicación tendrá 2 teclados para la escritura. | |
| Condiciones de Ejecución: La aplicación debe ejecutarse en el dispositivo.<br>El usuario escogerá el teclado que mejor se adapte a su comodidad. | |
| Entrada/Pasos de ejecución:<br>1. La aplicación debe estar ejecutándose en el dispositivo.<br>2. Cuando el usuario se desplace por los menús ingresara a la opción de escritura.<br>3. Luego seleccionara el teclado que desee utilizar (vertical u horizontal).<br>4. Siempre tendrá ayudas por medio de voz. | |
| Resultado esperado:<br>El usuario deberá acoplarse al teclado con el que mejor se sienta, aprenderá y afianzara los conocimientos en la escritura del sistema Braille. | |
| Evaluación de la prueba:<br>Positiva | |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 4 | Historia de Usuario: 4 |

| |
|---|
| Nombre: Teclado horizontal de escritura. |
| Descripción: Implementación del teclado horizontal. |
| Condiciones de Ejecución: La aplicación debe ejecutarse en el dispositivo.<br>El usuario debió elegir el teclado horizontal de escritura. |
| Entrada/Pasos de ejecución:<br>1. El usuario ingresara al módulo de escritura.<br>2. Elegir el teclado horizontal de escritura.<br>3. Luego seleccionara el tiempo de respuesta (lento, medio o rápido)<br>4. Cuando ingrese al teclado horizontal vera una pantalla con ocho botones ubicados en la parte superior de la pantalla. (1,2,3,4,5,6, espacio y borrar)<br>5. El usuario empezara a digitar su texto, si en algún momento quiere saber que ha escrito solo debe presionar con el dedo la pantalla del dispositivo por unos segundos y el sintetizador de voz le dirá lo que tiene escrito.<br>6. Si quiere guardar el texto entonces deberá oprimir la pantalla hasta que la aplicación le diga que fue copiado.<br>7. Si en algún momento desea salir del módulo, debe oprimir dos veces la pantalla del dispositivo. |
| Resultado esperado:<br>El usuario deberá acoplarse al teclado, afianzando los conocimientos en la escritura del sistema Braille, lograra escribir textos para luego copiarlos en una aplicación diferente (whatsapp, facebook). |
| Evaluación de la prueba:<br>Positiva |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 5 | Historia de Usuario: 5 |
| Nombre: Búsqueda | |
| Descripción: La aplicación deberá buscar las combinaciones de la simbología Braille. | |
| Condiciones de Ejecución: La aplicación debe ejecutarse en el dispositivo. | |
| Entrada/Pasos de ejecución:<br>1. Al momento en el que el usuario ingrese a cualquiera de los módulos de la aplicación, esta debe buscar la combinación que forma el carácter. | |
| Resultado esperado:<br>El usuario contara con la terminología del sistema Braille, tendrá a su disposición letras, números, signos matemáticos, vocales acentuadas, signos ortográficos, diéresis. | |
| Evaluación de la prueba:<br>Positiva | |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 6 | Historia de Usuario: 6 |
| Nombre: Aprendizaje | |
| Descripción: Desarrollo del módulo de aprendizaje. | |
| Condiciones de Ejecución: El usuario debe ingresar al módulo de aprendizaje. | |
| Entrada/Pasos de ejecución:<br>1. El usuario ingresara a la aplicación.<br>2. Se desplazara por las páginas y se ubicara en el módulo de aprendizaje.<br>3. Para ingresar al módulo debe oprimir unos segundos la pantalla.<br>4. Encontrará divido en letras, números, signos ortográficos, vocales acentuadas, signos matemáticos y diéresis la simbología del sistema Braille.<br>5. Cuando ingrese a alguna de las opciones en el caso de letras, números y signos ortográficos tendrá que digitar el signo generador de cada uno.<br>6. Al momento de digitar un símbolo deberá confirmarlo. | |
| Resultado esperado:<br>Se busca que el aprendizaje sea de manera fácil y rápida, por lo que se dividió en seis categorías, cada vez que ingrese i es necesario se le pedirá que digite el símbolo que genera y luego el carácter que esta aprendiendo, cuando logre digitarlo se le pedirá que lo confirme, ósea que vuelva a digitar los puntos que lo generan. | |
| Evaluación de la prueba:<br>Positiva | |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 7 | Historia de Usuario: 7 |
| Nombre: Navegar por los menús | |
| Descripción: Desarrollo del módulo de aprendizaje. | |
| Condiciones de Ejecución: El usuario debe ingresar al módulo de aprendizaje. | |
| Entrada/Pasos de ejecución:<br>1. El usuario ingresara a la aplicación.<br>2. Se desplazara por las páginas y se ubicara en el módulo de aprendizaje.<br>3. Para ingresar al módulo debe oprimir unos segundos la pantalla.<br>4. Encontrará divido en letras, números, signos ortográficos, vocales acentuadas, signos matemáticos y diéresis la simbología del sistema Braille.<br>5. Cuando ingrese a alguna de las opciones en el caso de letras, números y signos ortográficos tendrá que digitar el signo generador de cada uno.<br>6. Al momento de digitar un símbolo deberá confirmarlo. | |
| Resultado esperado: | |

| |
|---|
| Se busca que el aprendizaje sea de manera fácil y rápida, por lo que se dividió en seis categorías, cada vez que ingrese i es necesario se le pedirá que digite el símbolo que genera y luego el carácter que está aprendiendo, cuando logre digitarlo se le pedirá que lo confirme, ósea que vuelva a digitar los puntos que lo generan. |
| Evaluación de la prueba:<br>Positiva |

| **CASO DE PRUEBA DE ACEPTACIÓN** | |
|---|---|
| Código: 8 | Historia de Usuario: 8 |
| Nombre: Inicio de la aplicación | |
| Descripción: Describir interfaz de inicio. | |
| Condiciones de Ejecución: El usuario debe iniciar la aplicación. | |
| Entrada/Pasos de ejecución:<br>1. Cuando el usuario ingrese a la aplicación encontrara una pantalla de inicio con una frase en Braille “Aerodinámicamente el abejorro no debería volar, pero no lo sabe, por lo que va a volar de todas formas” y el logo. | |
| Resultado esperado:<br>Que la interfaz de inicio sea cómoda y agradable a la vista para el usuario. | |
| Evaluación de la prueba:<br>Positiva | |

| **CASO DE PRUEBA DE ACEPTACIÓN** | |
|---|---|
| Código: 9 | Historia de Usuario: 9 |
| Nombre: Opciones de escritura. | |
| Descripción: Describir las opciones que el usuario tiene al momento de escribir. | |
| Condiciones de Ejecución: El usuario debe estar en el módulo de escritura. | |
| Entrada/Pasos de ejecución:<br>1. Cuando el usuario ingrese al módulo de escritura, seleccionara alguno de los dos teclados (vertical, horizontal).<br>2. Luego de seleccionar alguno de los dos teclados selecciona el tiempo que se ajuste a su comodidad.<br>3. Empezara a digitar su texto.<br>4. Si quiere saber que ha escrito solo debe presionar la pantalla unos segundos.<br>5. Si el usuario quiere guardar lo que digito entonces debe oprimir la pantalla hasta que se le notifique que el texto fue copiado. | |

| |
|---|
| Resultado esperado:<br>Que el usuario tenga a disposición las diferentes opciones de teclado, como también el guardado y copiado de los escritos. |
| Evaluación de la prueba:<br>Positiva |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 10 | Historia de Usuario: 10 |
| Nombre: Elección modulo | |
| Descripción: Elección módulos de aplicación. | |
| Condiciones de Ejecución: La aplicación debe estar en ejecución. | |
| Entrada/Pasos de ejecución:<br>1. Para que el usuario escoja alguna de las opciones de los módulos basta con ubicarse en la que desee<br>2. Luego mantener oprimida la pantalla por unos segundos hasta que ingrese a la opción deseada<br>3. Sera notificado por voz en que ítem se encuentra y a cual ingresa.<br>4. El color del fondo y la letra del aplicativo varían con cada cambio de página (blanco y negro). | |
| Resultado esperado:<br>Lograr que el usuario realice la elección de uno de los módulos de la aplicación de una forma fácil, debido a que en el caso de que tenga algún tipo de discapacidad visual, será más comodo para él. | |
| Evaluación de la prueba:<br>Positiva | |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 11 | Historia de Usuario: 11 |
| Nombre: Aprendizaje letras | |
| Descripción: Módulo aprendizaje – Letras | |
| Condiciones de Ejecución: El usuario ingresara en el módulo de aprendizaje y luego en el sub módulo de letras. | |
| Entrada/Pasos de ejecución:<br>1. Ingresar a la aplicación y luego al módulo de aprendizaje. | |

| |
|---|
| 2. Cuando se encuentre en el módulo de aprendizaje deberá ingresar a la opción de letras.<br>3. Cuando ingrese a una letra por primera vez, tendrá que digitar el símbolo generador que son los botones (1, 2, 3, 4,5 y 6) en un tiempo mínimo de 70 segundos.<br>4. Luego de digitar correctamente el signo generador, tendrá que digitar los botones que generan la letra.<br>5. Cuando el usuario digita bien la combinación de puntos, se le pide que confirme el carácter de nuevo.<br>6. El fondo de la pantalla y las letras se intercalan en blanco y negro, los botones son de color blanco y al momento en que se oprimen cambian de color a verde (por si la persona tiene baja visión podrá distinguirlo). |
| Resultado esperado:<br>El usuario debe interactuar de una manera sencilla con el módulo de letras, se le pide que digite el signo generador y luego los botones que forman el carácter, cuando los digita bien se le pide que lo confirme debido a que según el método de Bliseo se aprende repitiendo y esta es la mejor opción de repasar la letra que se va digitando. |
| Evaluación de la prueba:<br>Positiva |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 12 | Historia de Usuario: 12 |
| Nombre: Teclado confirmación – signo generador | |
| Descripción: Módulo aprendizaje – números | |
| Condiciones de Ejecución: El usuario ingresara en el módulo de aprendizaje y luego en el sub módulo de números. | |
| Entrada/Pasos de ejecución:<br>1. Ingresar a la aplicación y luego al módulo de aprendizaje.<br>2. Cuando se encuentre en el módulo de aprendizaje deberá ingresar a la opción de números (oprimiendo unos segundos la pantalla del dispositivo).<br>3. Cuando ingrese a un número por primera vez, tendrá que digitar el símbolo generador que son los botones (3, 4,5 y 6).<br>4. Luego de digitar correctamente el signo generador de los números, tendrá que digitar los botones que generan el número.<br>5. Cuando el usuario digita bien la combinación de puntos, se le pide que confirme el carácter de nuevo.<br>6. Cuando digita un numero correctamente, seguirá con el siguiente.<br>7. Es importante recordar que los números del 0 al 9 son los mismos puntos de las letras de la a a la j. | |

| |
|---|
| 8. El fondo de la pantalla y las letras se intercalan en blanco y negro, los botones son de color blanco y al momento en que se oprimen cambian de color a verde (por si la persona tiene baja visión podrá distinguirlo). |
| Resultado esperado:<br>El usuario debe interactuar de una manera sencilla con el módulo de números, se le pide que digite el signo generador y luego los botones que forman el carácter, cuando los digita bien se le pide que lo confirme debido a que según el método de Bliseo se aprende repitiendo y esta es la mejor opción de repasar el número que se va digitando. |
| Evaluación de la prueba:<br>Positiva |

| **CASO DE PRUEBA DE ACEPTACIÓN** | |
|---|---|
| Código: 13 | Historia de Usuario: 13 |
| Nombre: Desarrollo de las actividades de | |
| Descripción: Módulo aprendizaje – números | |
| Condiciones de Ejecución: El usuario ingresara en el módulo de aprendizaje y luego en el sub módulo de números. | |
| Entrada/Pasos de ejecución:<br>1. Ingresar a la aplicación y luego al módulo de aprendizaje.<br>2. Cuando se encuentre en el módulo de aprendizaje deberá ingresar a la opción de números (oprimiendo unos segundos la pantalla del dispositivo).<br>3. Cuando ingrese a un número por primera vez, tendrá que digitar el símbolo generador que son los botones (3, 4,5 y 6).<br>4. Luego de digitar correctamente el signo generador de los números, tendrá que digitar los botones que generan el número.<br>5. Cuando el usuario digita bien la combinación de puntos, se le pide que confirme el carácter de nuevo.<br>6. Cuando digita un numero correctamente, seguirá con el siguiente.<br>7. Es importante recordar que los números del 0 al 9 son los mismos puntos de las letras de la a a la j.<br>8. El fondo de la pantalla y las letras se intercalan en blanco y negro, los botones son de color blanco y al momento en que se oprimen cambian de color a verde (por si la persona tiene baja visión podrá distinguirlo). | |
| Resultado esperado:<br>El usuario debe interactuar de una manera sencilla con el módulo de números, se le pide que digite el signo generador y luego los botones que forman el carácter, cuando los digita bien se le pide que lo confirme debido a que según el método de Bliseo se aprende repitiendo y esta es la mejor opción de repasar el número que se va digitando. | |
| Evaluación de la prueba: | |

| |
|---|
| Positiva |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 14 | Historia de Usuario: 14 |
| Nombre: Desarrollo de las actividades de entretenimiento | |
| Descripción: Creación de los juegos para evaluar conocimiento. | |
| Condiciones de Ejecución: El usuario ingresara en el módulo de test. | |
| Entrada/Pasos de ejecución:<br>1. Se desarrolla el módulo de test, para evaluar al usuario sobre los conocimientos adquiridos<br>2. Se crean tres tipos de test, cada uno diferente al resto.<br>3. Test cantidad: Se le pregunta al usuario cuantos puntos tiene algún símbolo del sistema Braille, él debe poner la cantidad de dedos que forman ese símbolo.<br>4. Test Secuencia; el usuario debe escuchar una secuencia números y luego repetirlos.<br>5. Test confirmación; se le dirá al usuario que escriba alguno de los símbolos del sistema Braille.<br>6. Para el desarrollo de los test se maneja un teclado similar al vertical de escritura, la diferencia es que aquí no tiene los botones de espacio y borrar.<br>7. Cuando el usuario tenga aciertos serán acumulados, la pantalla se pondrá color verde y aparecerá un +2.<br>8. Cuando el usuario falle la pantalla se pondrá roja y se le restaran puntos. | |
| Resultado esperado:<br>El usuario debe interactuar de una manera sencilla con el módulo de números, se le pide que digite el signo generador y luego los botones que forman el carácter, cuando los digita bien se le pide que lo confirme debido a que según el método de Bliseo se aprende repitiendo y esta es la mejor opción de repasar el número que se va digitando. | |
| Evaluación de la prueba:<br>Positiva | |

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Código: 15 | Historia de Usuario: 15 |
| Nombre: Desarrollo de manuales PDF | |
| Descripción: Creación de los manuales correspondientes a la aplicación | |
| Condiciones de Ejecución: | |

| |
|---|
| Entrada/Pasos de ejecución:<br>1. Crear los manuales de usuario y del programador para usuarios interesados en el uso del aplicativo y programadores.<br>2. El manual de usuario tendrá especificaciones de requerimientos, instalación y manejo de la aplicación.<br>3. El manual del programador tendrá a disposición instrucciones de acceso y especificaciones del código.<br>4. Los manuales estarán disponibles en formato PDF. |
| Resultado esperado:<br>Se encontrara a disposición de quien lo requiera los manuales del usuario y del programador, especificando en ellos todo lo referente a la aplicación. |
| Evaluación de la prueba:<br>Positiva |

Características que debe tener la aplicación:

| Número de Historia | Historia de Usuario | Tarea |
|---|---|---|
| **1** | Crear una aplicación de uso fácil y que sea amigable al usuario. | Lograr que la aplicación tenga un diseño amigable y sea de fácil uso para los usuarios finales.<br>Diseñar módulos, desarrollar animaciones, implementar sonidos. |
| **2** | Crear la Base de datos para inserción, consulta y actualización de datos. | Creación y diseño de la base de datos. |
| **3** | Desarrollo del teclado para las personas con discapacidad visual. | Diseño e implementación del módulo de escritura y lectura. |
| | Implementación de la librería TTS. | Importar la librería TTS (Android) y utilizar los métodos que esta ofrece. |
| **14** | Desarrollo de actividades de entretenimiento para práctica y evaluación del conocimiento del sistema Braille. | Diseño e implementación de test para el aprendizaje obtenido sobre el sistema; juegos y trivias para la práctica del sistema Braille. |

| | | |
|---|---|---|
| **15** | Desarrollo de manuales PDF y mediante voz. | Elaboración de los respectivos manuales para usuarios finales. |
| | Desarrollo de teclados de prueba. | Diseño e implementación de teclados de prueba para uso del usuario común. |
| **15** | Elaborar manuales técnicos y de usuario. | Elaboración de ayudas necesarias para el buen manejo y usabilidad de la aplicación. |

## CONCLUSIONES.

- Todos los módulos se encuentran desarrollados, se está trabajando con el CRAC (Centro de Rehabilitación para Adultos Ciegos) con el que se están realizando las pruebas correspondientes a la aplicación, anteriormente se trabajó con el INCI (Instituto Nacional para Ciegos).
- Se debe aprovechar el incremento en las nuevas tecnologías para adaptar la enseñanza tradicional Braille hacia los dispositivos móviles, facilitando el acceso de la población con discapacidad visual hacia el aprendizaje de la escritura de dicho sistema.
- El diseño de los prototipos de teclados de escritura han sido resultado de la experiencia que se ha tenido con los usuarios y las instituciones con las que se ha trabajado.
- Se requiere de la asesoría de personal capacitado en la comunicación escrita basada en el sistema Braille o licenciados en educación especial con experiencia en población invidente para que las personas con discapacidad visual, tengan un mejor manejo de la aplicación y la simbología del Braille.
- Aunque no fue fácil, porque no se encuentra mucha información, la experiencia trabajando con la metodología XP, fue satisfactoria, en el momento en que se eligió y aplico al desarrollo del proyecto.
- Motivar a los usuarios con discapacidad visual a buscar que las nuevas tecnologías puedan trabajar de la mano con las herramientas tradicionales.
- El aplicativo cuenta con un método de puntos, el cual puede ser utilizado como recurso de evaluación por los maestros para analizar los avances constantes del usuario.

## RECOMENDACIONES.

La investigación y resultados que arrojó el trabajo de campo realizado a lo largo del proyecto permiten establecer un conjunto de recomendaciones:

- Ayudar a que los usuarios con edad avanzada dejen de ver lo dispositivos móviles como tabú, motivándolos al uso de nuevas tecnologías.

- Es recomendable brindarle el apoyo necesario de acompañamiento a los usuarios con discapacidad visual para poder aprovechar altamente la potencialidad del aplicativo.

- Para el correcto uso del aplicativo el usuario debe tener muy presente que todo el aplicativo posee sus interfaces graficas en posición horizontal.

- La aplicación debe ser instalada solo en dispositivos con Sistema Operativo Android.

Si se quisiera realizar futuras versiones del aplicativo es aconsejable:

- El aplicativo cuenta con tres módulos que implementan la escritura con el sistema braille, aun no se ha implementado un módulo de lectura, que sería el modulo que debería desarrollarse como continuación.

- El aplicativo está orientado a objetos por lo que si se requiere realizar una actualización a la estructura del código, es recomendable, documentarse con el manual de programador que se encuentra adjunto en este cd.

- El aplicativo cuenta con un módulo de videos que contribuye con el aprendizaje de la escritura del sistema braille, por lo que si se desarrollan nuevos módulos, es necesario agregar nuevos videos para que se abarquen los nuevos temas a implementar.

## BIBLIOGRAFIA.


1. Manifesto for Agile Software Development , Kent Beck, Mike Beedle, Arie William L Heward, Michael D Orlanssky, Biblioteca de la educación especial II. Programas de educación especial, ediciones ceac.

2. Margarita Córdoba Pérez, Julio Cabero Almenara, Francisco Javier Soto Pérez, Buenas prácticas de aplicación. Las TIC para la igualdad, Ediciones de la U.

## INFOGRAFIA.


1. van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas [en línea]. *<http://agilemanifesto.org/>, citado el 11 de Septiembre de 2014.*

2. Reglas y Prácticas en eXtreme Programming [en línea] *<http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>, citado el 10 de Octubre de 2014.*

3. El alfabeto Braille. [en línea]. <http://www.discapnet.es/Castellano/areastematicas/Accesibilidad/accesibiliadcomunicacion/lenguajebraille/Documents/Guias/Braille/alfabeto_braille_2.htm>.[Sin fecha], citado el 25 de septiembre de 2014.

4. Orientaciones generales para la enseñanza del sistema de lectoescritura braille. [en línea]. <http://www.colombiaaprende.edu.co/recursos/software/palabrasycuentas/OrientacionesBraille.pdf> [2000], citado el 8 de septiembre de 2014.

5. Limitación Visual [en línea]. <http://inclusionlimitacionvisual.blogspot.com>.[Lunes 26 Noviembre de 2012], citado el 11 de septiembre de 2014.

6. Tiflotecnología [en línea]. <http://www.ite.educacion.es/formacion/materiales/129/cd/unidad_10/m10_tiflotecnologia.htm>. [Sin fecha], citado el 11 de septiembre de 2014.

7. Alfabeto Braille. [en línea]. <http://www.fbu.edu.uy/informacion/alfabeto/alfabeto.htm [Sin fecha], citado el 20 de octubre de 2014.

8. Metodología XP. [en línea]. <http://ingsoftware072301.obolog.es/metodologia-xp-201287> [8 de Julio de 2012], citado el 8 de Julio de 2015.

9. Eclipse [en línea]. *<http://www.edukanda.es/mediatecaweb/data/zip/678/XW07_M2101_02708/web/main/m1/v9_10_1.html>* [sin fecha], citado el 8 de Julio de 2015.

10. PHP [en línea]. *<http://php.net/manual/es/history.php.php>* [sin fecha], citado el 8 de Julio de 2015.

**11.** ¿Qué es PHP? [en línea]. <*http://www.maestrosdelweb.com/phpintro/*> [sin fecha], citado el 8 de Julio de 2015.

**12.** Magic [en línea]. <*http://www.ver.com.co/hogar/magic.html*> [2013], citado el 8 de Septiembre de 2015.

**13.** *Jaws* [en línea]. <*http://www.guiaacademica.com/educacion/personas/cms/pregrado/conexion u/ARTICULO-WEB-NEW_NOTA_GUIA-13478695.html*> [*Septiembre 16 de 2014*], citado el 8 de Septiembre de 2015.

**14.** *Inclusión digital* [en línea]. <*http://nodejesdeinnovar.blogspot.com.co/*> [martes, 28 de abril de 2015], citado 22 Julio de 2015.

**15.** Orientaciones para la inclusión de las personas con discapacidad visual en la vida universitaria [en línea]. <*http://sid.usal.es/idocs/f8/fdo19306/orientaciones_visual_uca.pdf*> [Sin fecha], citado el 14 de Abril de 2015.

**16.** Rehabilitación Integral para Personas Ciegas [en línea]. <*http://www.cracolombia.org/rehabilitacion-integral-para-personas-ciegas/*>[Sin fecha], citado el 16 de Septiembre de 2015.

**17.** Programación Extrema [en línea]. <*http://eisc.univalle.edu.co/materias/WWW/material/lecturas/xp.pdf*>[Sin fecha], citado el 16 de Septiembre de 2015.

**18.** Iteraciones[en línea]. <*http://oness.sourceforge.net/proyecto/html/ch08.html*> [Sin fecha], citado el 10 de Mayo de 2015.

**19.** Alumnos con discapacidad visual necesidades y respuesta educativa [en línea].<*http://educacion.once.es/appdocumentos/educa/prod/Necesidades%20y%20respuesta%20educativa.pdf*> [Sin fecha], citado el 17 de Julio de 2015.

**20.** Tecnología y ceguera [en línea]. < *http://www.uncu.org.uy/tecnologia_y_ceguera.htm*> [Sin fecha], citado el 4 de Julio de 2015.

**21.** Smartphones y accesibilidad [en línea]. <*http://www.omicrono.com/2013/01/smartphones-y-accesibilidad-soluciones-para-sordos-ciegos-mayores-y-gente-con-movilidad-reducida/*> [Sin fecha], citado el 6 de Julio de 2015.

**22.** Programación Extrema [en línea]. <*http://es.slideshare.net/edgarespinoza/programacion-extrema*> [13 de julio de 2008], citado el 12 de Marzo de 2015.

**23.** Metodología Ágil XP [en línea]. <*http://procesosoftware.wikispaces.com/M%C3%A9todo+%C3%81gil+XP*> [Sin fecha], citado el 7 de Julio de 2015.

**24.** Acceso a las tic para alumnos con discapacidad visual. [en línea]. <*http://ares.cnice.mec.es/informes/17/contenido/11.htm*> [Sin fecha], citado el 7 de Julio de 2015.

**25.** Ciclo de vida [en línea]. <*http://oness.sourceforge.net/proyecto/html/pt03.html*> [Sin fecha], citado el 8 de Enero de 2015.

**26.** Fases de la programación Extrema [en línea]. <*http://programacionextrema.tripod.com/fases.htm*> [Sin fecha], citado el 20 de Enero de 2015.

**27.** La discapacidad visual y las TIC en la etapa escolar [en línea]. <*http://recursostic.educacion.es/observatorio/web/es/cajon-de-sastre/38-cajon-de-sastre/322-la-discapacidad-visual-y-las-tic-en-la-etapa-escolar*> [Sin fecha], citado el 6 de Febrero de 2015.

**ANEXOS.**

MANUAL DEL PROGRAMADOR.

APLICACIÓN PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO

ANDROID PARA USUARIOS CON DISCAPACIDAD VISUAL O CON INTERES EN APRENDER LA COMUNICACIÓN BASADA EN EL SISTEMA BRAILLE.

ERIKA MAYERLI GUZMAN RAMIREZ JOHAN
SEBASTIAN SUAREZ PEREZ

UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS

FACULTAD TECNOLOGICA

TECNOLOGIA EN SISTEMATIZACION DE DATOS

BOGOTA

2015

TABLA DE CONTENIDO.

1.INTRODUCCIÓN..... .....107
2. REQUERIMIENTOS. …….. 107
3. DESCARGA E INSTALACIÓN DE REQUERIMIENTOS……3
4. CONFIGURACIONES ESPECIALES…… .....113
5. DEFINICIÓN A TENER EN CUENTA: 115
6. DOCUMENTACIÓN TÉCNICA 116
6.1. Descripción de clases utilizadas. .....116
6.2. Descripción de actividades utilizadas. .....116
6.3. Descripción de recursos adicionales: .....119
7. FUNDAMENTOS DE PROGRAMACIÓN: 125
7.1. Fundamentos modulo aprendizaje:.....125
7.2. Fundamentos modulo escritura: .....126
7.3. Fundamentos modulo test: .....126
7.4. Actividades y clases extra:.....127
8. IMPLEMENTACIÓN DE CÓDIGO DE IMPORTANCIA: 127
8.1. clases:.....127
8.1.1 TTSManager:.....127
8.1.2 Display:.....127
8.1.3 Base_de_datos:.....128
8.1.4 Tablas:.....129
9. ACTIVIDADES: 130
9.1. Cantidad: .....130
9.2. Teclado_Aprendizaje: .....132
9.3. EleccionModulo: .....134
9.4. EscrituraPrimer: .....135
9.5. Encontrar: .....136
9.6. Splash:.....136

## 1. INTRODUCCIÓN:

El propósito de este manual del programador, es dar a conocer al lector los códigos fuentes del programa realizado. Para ello tratamos de forma amena y concisa cada uno de los códigos junto con la programación utilizada en el desarrollo del software, esto con el fin de que el usuario pueda acceder a cada uno de los valores y parámetros de las funciones que se encuentran expuestas en la programación de la Aplicación móvil bajo el sistema operativo Android para usuarios con discapacidad visual o usuarios interesados en la comunicación basada en el sistema Braille.

## 2. **REQUERIMIENTOS.**

Para el desarrollo del aplicativo se utilizaron las siguientes herramientas, cada una con su link oficial de descarga.

1. **Java Development Kit (JDK):** contiene el entorno de desarrollo de Java **http://www.oracle.com/technetwork/java/javase/downloads/jdk8**http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html?ssSourceSiteId=otnes**downloads-2133151.html?ssSourceSiteId=otnes**
2. **Android Studio:** Entorno de programación de Android **http://developer.android.com/intl/es/sdk/index.html**
3. **Genymotion:** Emulador utilizado para testeo de la aplicación **https://www.genymotion.com/#!/**
4. **SQLite:** librería que implementa un sistema autónomo, en este caso Android Studio

## 3. DESCARGA E INSTALACIÓN DE REQUERIMIENTOS.

**Java Development Kit (JDK):**

1. Se abre el link de descarga.
2. Se selecciona la opción que corresponde al Sistema Operativo y aceptar la licencia (imagen 1)

*Imagen 1 (selección de sistema operativo)*

**Java SE Development Kit 8u11**

**You must accept the Oracle Binary Code License Agreement for Java SE to download this software.**

**Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.**

| Product / File Description | File Size | Download |
|---|---|---|
| Linux x86 | 133.58 MB | jdk-8u11-linux-i586.rpm |
| Linux x86 | 152.55 MB | jdk-8u11-linux-i586.tar.gz |
| Linux x64 | 133.89 MB | jdk-8u11-linux-x64.rpm |
| Linux x64 | 151.65 MB | jdk-8u11-linux-x64.tar.gz |
| Mac OS X x64 | 207.82 MB | jdk-8u11-macosx-x64.dmg |
| Solaris SPARC 64-bit (SVR4 package) | 135.66 MB | jdk-8u11-solaris-sparcv9.tar.Z |
| Solaris SPARC 64-bit | 96.14 MB | jdk-8u11-solaris-sparcv9.tar.gz |
| Solaris x64 (SVR4 package) | 135.7 MB | jdk-8u11-solaris-x64.tar.Z |
| Solaris x64 | 93.18 MB | jdk-8u11-solaris-x64.tar.gz |
| Windows x86 | 151.81 MB | jdk-8u11-windows-i586.exe |
| Windows x64 | 155.29 MB | jdk-8u11-windows-x64.exe |

*Fuente: Autores*

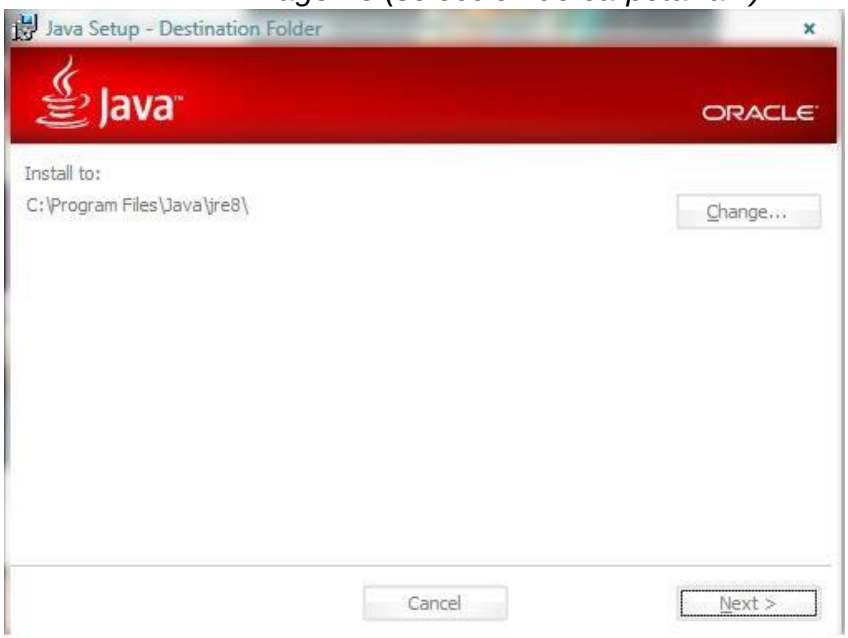
3. Una vez descargado tienes que ir a la carpeta donde se descargó el archivo
4. Ejecutarlo como administrador 5. Cuando se abra esta ventana dar clic en siguiente (imagen

*Imagen 2 (inicio de instalación)*

*Fuente: Autores*

5. Dar siguiente en las posteriores ventanas
6. Elegir la carpeta donde se instalara java y dar clic en siguiente (imagen 3)

*Imagen 3 (selección de carpeta raíz)*




*Fuente: Autores*

7. Una vez terminada la instalación dar clic en cerrar (imagen 4)

*Imagen 4 (finalización de instalación)*

*Fuente: Autores*

- **Android Studio:**

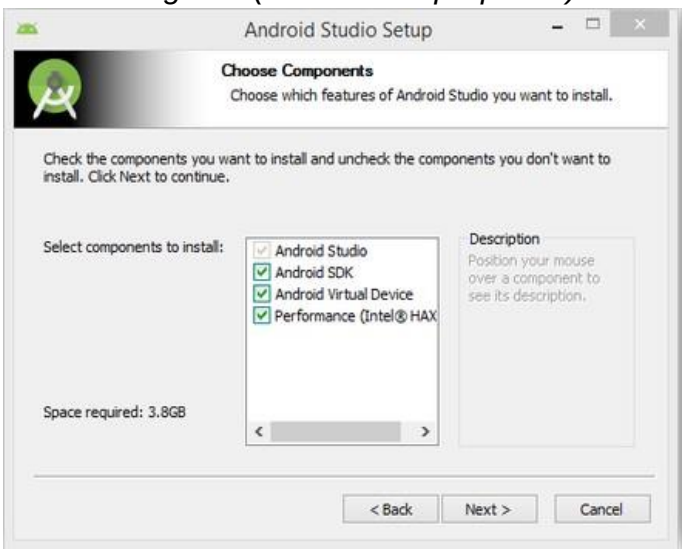
1. Dirigirse al link de descarga, el tamaño promedio de descarga es de 450MB y contiene todas las librerías necesarias para comenzar a crear aplicaciones
2. Dirigirse a la carpeta donde quedo guardado el instalador y abrirlo, aparecerá el asistente (imagen 5).

*Imagen 5 (inicio asistente)*

*Fuente: Autores*

3. Elegir componentes a instalar, lo más recomendable es seleccionar todas las opciones (imagen 6)

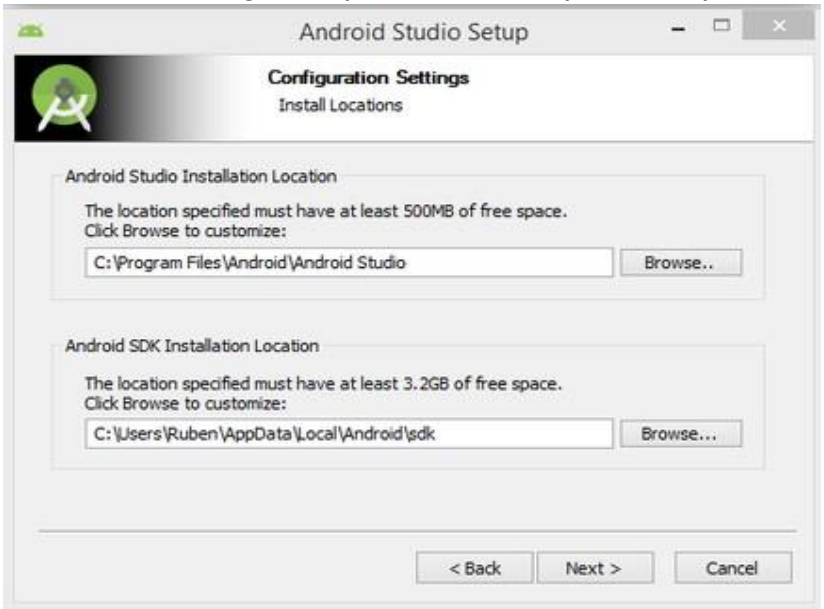
*Imagen 6 (elección de paquetes)*




*Fuente: Autores*

4. Elegir ruta de instalación de Android Studio y ruta de instalación de SDK.

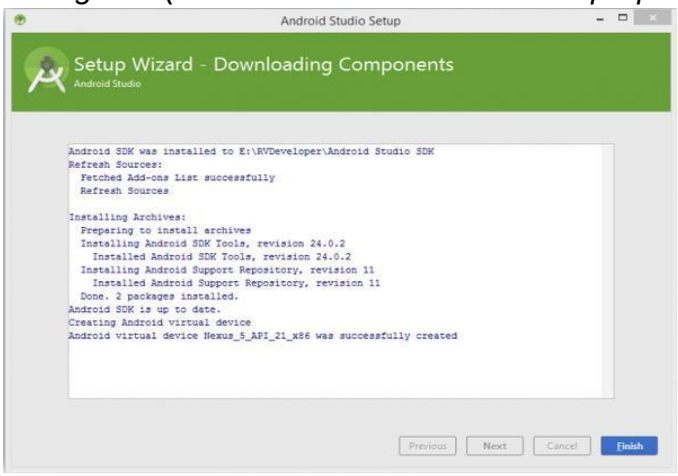
*Imagen 7 (elección de carpetas raíz)*




*Fuente: Autores*

5. Una vez finalice la instalación Android Studio se conectará a Internet y descargará los elementos del SDK necesarios para funcionar correctamente (imagen 8).

*Imagen 8 (finalización de instalación de paquetes SDK)*




*Fuente: Autores*

6. Una vez terminado le damos finalizar y listo.

- **Genymotion:**

    1. Abrir link de descarga.
    2. Registrarse en la página oficial.
    3. Elegir Sistema Operativo (imagen 9)

*Imagen 9 (Elección de Sistema Operativo)*

*Fuente: Autores*

4. Con la herramienta Genymotion no son muchos los pasos a confirmar, tan solo uno le damos siguiente a la bienvenida. (imagen 10)

*Imagen 10 (ventana de inicio del asistente)*

*Fuente: Autores*

5. Dar clic en finalizar y marcar la casilla launch si se desea abrir cuando finalice

*Imagen 11 (finalización de instalación de paquetes)*

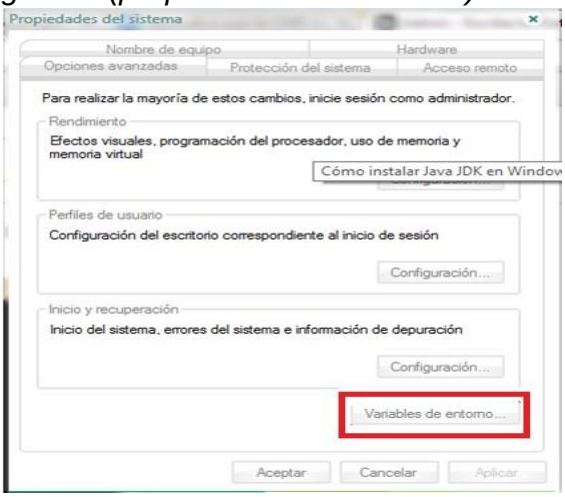
*Fuente: Autores*

## 4. CONFIGURACIONES ESPECIALES

- **JDK:** es necesario configurar las variables de entorno para poder utilizar correctamente el JDK.

1. Abrir las propiedades de Sistema y seleccionar variables de entorno (imagen 12).

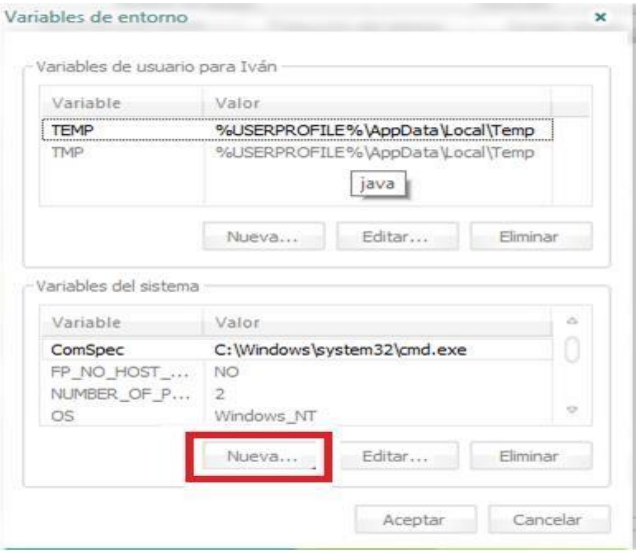
*Imagen 12 (propiedades del sistema)*




*Fuente: Autores*

2. La ventana de variables de entorno aparecerá. Damos nueva variable (imagen 13).

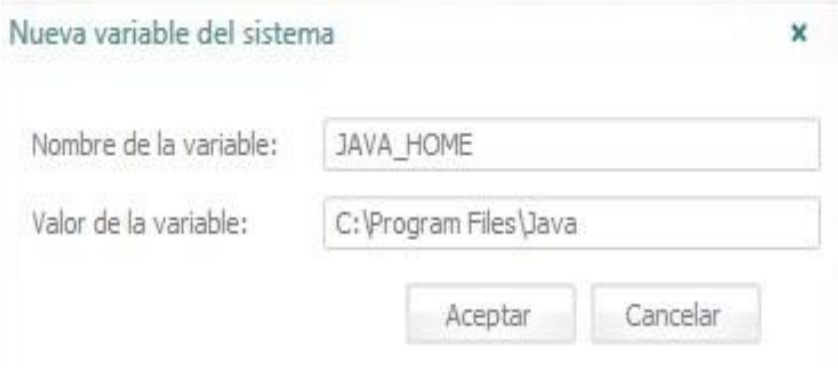
*Imagen 13 (elección de nueva variable)*




*Fuente: Autores*

3. En Nueva variable del sistema, agregaremos una nueva variable con el nombre "JAVA_HOME" y su valor será la ruta de la carpeta raíz que se requiera (imagen 14).

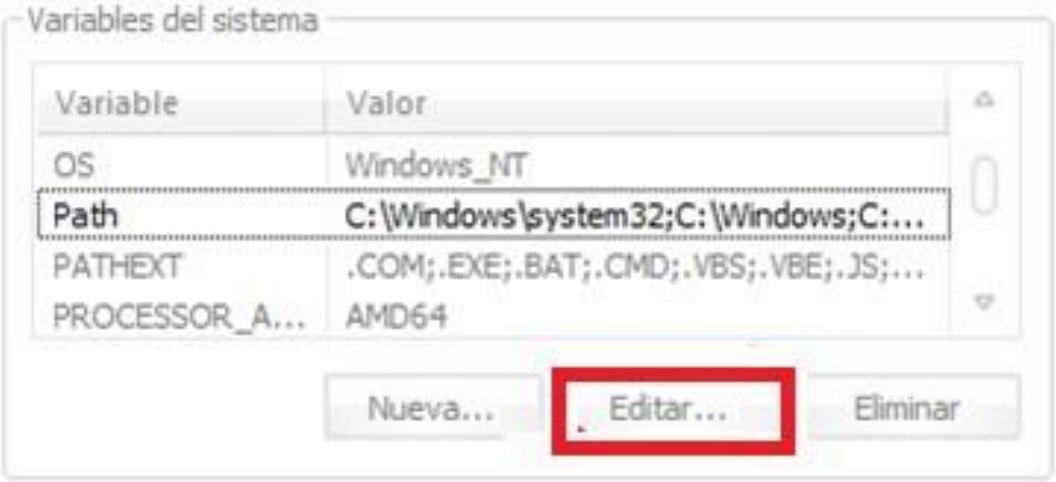
*Imagen 14 (elección de carpeta raíz)*




*Fuente: Autores*

4. Volveremos a la ventana Variables de entorno. Ahora buscaremos una Variable que se llama "Path", y daremos clic en el botón de "Editar" imagen

*Imagen 14 (edición variable path)*




*Fuente: Autores*

5. Agregaremos lo siguiente SIN ELIMINAR NADA en el cuadro Valor de la variable:
   ; JAVA_HOME%jdk1.8.0_11
6. Dar clic en aceptar.

## 5. DEFINICIÓN A TENER EN CUENTA:

- **ViewPager:**

Esta vista se ha vuelto muy popular entre las aplicaciones que han aparecido en los últimos meses. Gracias a esta vista se puede desplazar por los diferentes menús de la aplicación con un simple gesto de un dedo, es lo que en inglés se llama *swipe* (deslizar en español). El uso de esta vista proporciona a la aplicación un aspecto moderno, atractivo y más limpio, ya que no se tiene que incluir botones o enlaces a los diferentes menús de la aplicación.

Los ViewPager se utilizaron como herramienta para cada uno de los menús de la aplicación notificando por voz el paso de páginas y el contenido de cada una.

# 6. DOCUMENTACIÓN TÉCNICA

## 6.1 Descripción de clases utilizadas.

En este apartado se procede a la descripción de todos los archivos contenidos en la aplicación, cada celda contiene el nombre real de cada una de las clases y una breve descripción de cada una según corresponda.

| NOMBRE | DESCRIPCIÓN |
|---|---|
| TTSManager | Proporciona los métodos necesarios para poder acceder a los recursos de sintetizador de voz de los dispositivos Android. |
| Display | Proporciona los métodos para poder acceder a las dimensiones de los dispositivos móviles. |
| BaseDeDatos | Proporciona los métodos para poder crear la base de datos en SQLite y cada una de las tablas que se le enviar por parámetros. |
| Tablas | Proporciona los métodos para poder instanciar, eliminar, actualizar y consultar cada una de las tablas de la base de datos. |

## 6.2 Descripción de actividades utilizadas.

En este apartado se procede a la descripción de todos los archivos contenidos en la aplicación, cada celda contiene el nombre real de cada una de las actividades y una breve descripción de cada una según corresponda.

| NOMBRE | DESCRIPCIÓN |
|---|---|
| Cantidad | Permite al usuario |
| Confirmación | Muestra interfaz que simula el carácter que anteriormente el usuario eligió, tiene la finalidad de que el usuario confirme que comprendió la combinación de puntos a la que accedió en actividades anteriores. |

| | |
|---|---|
| Diéresis | Muestra ViewPager en el que proporciona al usuario cada uno de los items de diéresis (de la ä a la ü), para poder ingresar a la opción según corresponda. |
| EleccionModulos | Muestra ViewPager en el que proporciona al usuario las opciones a elegir respecto a cada uno de los módulos (aprendizaje, escritura, test o instrucciones). |
| EleccionVelocidad | Muestra ViewPager en el que proporciona al usuario las opciones a elegir respecto a la velocidad de respuesta (lento, medio, rápido) de la actividad escrituraPrimer o escrituraSegundo según corresponda. |
| EleccionTeclado | Muestra ViewPager en el que proporciona al usuario las opciones a elegir respecto al tipo de teclado para realizar escritos (teclado vertical o teclado horizontal). |
| Encontrar | Muestra interfaz que permite al usuario ingresar una combinación de puntos con respecto a algún carácter que aleatoriamente se le asigne. |
| EscrituraPrimer | Muestra interfaz en la que se encuentran todos los botones ubicados verticalmente en la parte lateral derecha y lateral izquierda, con la finalidad de que pueda realizar escritos y posteriormente, si se desea, poder exportarlos a diferentes aplicaciones. |
| EscrituraSegundo | Muestra interfaz en la que se encuentran todos los botones ubicados horizontalmente en la parte superior, con la finalidad de que pueda realizar escritos y posteriormente, si se desea, poder exportarlos a diferentes aplicaciones. |
| Letras | Muestra ViewPager en el que proporciona al usuario cada uno de los items de letras (de la a a la z), para poder ingresar a la opción según corresponda. |

| | |
|---|---|
| Llaves | Muestra interfaz en la que se encuentran todos los botones ubicados verticalmente en la parte lateral derecha y lateral izquierda, con la finalidad de que pueda acceder a la combinación pertinente a las llaves de los caracteres especiales, por ejemplo los números con la llave correspondiente a los botones 3, 4,5 y 6. |
| MenuPrincipal | Muestra ViewPager en el que proporciona al usuario cada uno de los ítems de selección de información (letras, números, signos ortográficos, vocales, acentuadas, signos matemáticos y diéresis), para poder ingresar a la opción según corresponda. |
| Numeros | Muestra ViewPager en el que proporciona al usuario cada uno de los ítems de selección de números (del 0 al 9), para poder ingresar a la opción según corresponda. |
| OperacionesMenu | Muestra ViewPager en el que proporciona al usuario cada uno de los items de selección de signos matemáticos (ejemplo: +, - >, <, etc.), para poder ingresar a la opción según corresponda. |
| Ortograficos | Muestra ViewPager en el que proporciona al usuario cada uno de los items de selección de signos ortográficos (ejemplo: ? ( ,“ , etc. ), para poder ingresar a la opción según corresponda. |
| Secuencia | Muestra interfaz en la que se le proporciona al usuario una secuencia de puntos de manera dinámica, para que el posteriormente pueda realizarla, con notificaciones de elección correcta y errónea. |
| SignoGenerador | Muestra interfaz que simula al signo generador que le permite al usuario realizar una selección de secuencia de 6 botones en 70 segundos. |
| Splash | Muestra interfaz que exhibe una animación y notifica al usuario del correcto inicio de la aplicación. |

| | |
|---|---|
| TecladoAprendizaje | Muestra interfaz que simula la combinación de puntos del respectivo carácter que el usuario eligió en actividades anteriores, cuenta con |
| | ayudas de voz para localizar en pantalla cada uno de los puntos. |
| TecladoAprendizajeNumero | Muestra interfaz que simula la combinación de puntos del respectivo carácter especial que el usuario eligió en actividades anteriores, cuenta con ayudas de voz para localizar en pantalla cada uno de los puntos. |
| Test | Muestra ViewPager en el que proporciona al usuario cada uno de los ítems de selección de actividades (cantidad, secuencia, encontrar), para poder ingresar a la opción según corresponda. |
| Vocales | Muestra ViewPager en el que proporciona al usuario cada uno de los ítems de selección de vocales (de la a a la e), para poder ingresar a la opción según corresponda. |
| Instrucciones | Muestra ViewPager en el que proporciona al usuario cada uno de los ítems de selección de instrucciones (instrucciones por voz), para poder ingresar a la opción según corresponda. |

**Nota:** Al entrar y al salir de cada una de las actividades mencionadas anteriormente el usuario será notificado mediante voz del correcto flujo de cada actividad de manera específica dependiendo de la naturaleza de cada una.

### 6.3 Descripción de recursos adicionales:

6.3.1 **carpeta anim:** Se encarga de almacenar archivos xml que se encargaran de animar las diferentes vistas a las que se le asocien.

6.3.1.1. **cerca:** Permite acercar durante 1000 milisegundos la vista a la que se asocie desde lo más pequeño a lo más grande dependiendo de su respectivo tamaño original.

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
android:interpolator="@android:anim/linear_interpolator">
   <scale
android:fromXScale="1.0"
android:toXScale="0.0"
android:fromYScale="1.0"
android:toYScale="0.0"
android:pivotX="50%"
android:pivotY="50%"
android:duration="1000" />  </set>
```

6.3.1.2 **girar:** Permite girar durante 222 milisegundos la vista a la que se le asocie desde 0 grados a 360 grados.

```
<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android">
   <rotate
android:duration="222"
android:fromDegrees="0"
android:toDegrees="360"
android:pivotX="50%"
android:pivotY="50%"
      ></rotate>
</set>
```

6.3.1.3 **rotate:** Permite girar durante 1700 milisegundos la vista a la que se le asocie desde 0 grados a 360 grados.

```
<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android">
<rotate
android:duration="1700"
android:fromDegrees="0"
android:toDegrees="360"
```

```
android:pivotX="50%"
android:pivotY="50%"
   ></rotate>
</set>
```

6.3.2 **drawable:** Se encarga de almacenar los archivos xml que va a interactuar con las vistas de las actividades de manera visual.

6.3.2.1 **borde:** Permite dar un borde de tipo rectangular, de color blanco con interior negro a la vista a la que se le asocie.

```
<layer-list xmlns:android="http://schemas.android.com/apk/res/android" >
<item>
      <shape android:shape="rectangle">
         <stroke android:width="2dp" android:height="0dp"
                     android:color="#FFFFFF" />
         <solid android:color="#000000" />
         <padding android:top="1dp" android:bottom="1dp" />
</shape></item></layer-list>
```

6.3.2.2 **botonCuadrado:** Asocia dos xml uno en el caso de que el usuario de click encima (boton_cuadrado_presionado) o en el caso contrario

(boton_cuadrado_defecto ).

```
<?xml version="1.0" encoding="utf-8"?>

<selector xmlns:android="http://schemas.android.com/apk/res/android">
   <item android:state_pressed="true"
android:drawable="@drawable/boton_cuadrado_presionado"/>
<item android:drawable="@drawable/boton_cuadrado_defecto"/>
</selector>
```

6.3.2.3 **boton_cuadrado_presionado:** Le proporciona a la vista asociada un color verde fosforescente con código hexadecimal #ff00ff20 al momento de ser oprimido.

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<shape xmlns:android="http://schemas.android.com/apk/res/android"
android:shape="rectangle">     <solid android:color="#ff00ff20" />
</shape>
```

6.3.2.4 **boton_cuadrado_defecto:** Le proporciona a la vista asociada un color vino tinto con código hexadecimal #ff95040e en el estado no-oprimido.

```
<?xml version="1.0" encoding="utf-8"?>

<shape xmlns:android="http://schemas.android.com/apk/res/android"
android:shape="rectangle">     <solid android:color="#ff95040e" />
</shape>
```

6.3.2.5 **botonCirculo:** Asocia dos xml uno en el caso de que el usuario de click encima (boton_circulo_presionado) o en el caso contrario

(boton_cuadrado_defecto)

```
<?xml version="1.0" encoding="utf-8"?>

<selector xmlns:android="http://schemas.android.com/apk/res/android">
<item android:state_pressed="true"
android:drawable="@drawable/boton_circulo_presionado"/>
<item android:drawable="@drawable/boton_circulo_defecto"/>
</selector>
```

6.3.2.6 **boton_circulo_presionado**: Le proporciona a la vista asociada un color verde fosforescente con código hexadecimal #ff00ff20 , un ancho y alto de 90 dp, un valor geométrico en forma de ovalo, un bordeado de tipo stroke, todo lo anterior cuando se encuentra en el estado oprimido.

```
<?xml version="1.0" encoding="utf-8"?>

<shape xmlns:android="http://schemas.android.com/apk/res/android"
android:shape="oval">
   <solid android:color="#ff00ff20" />
   <size android:height="90dp" android:width="90dp"/>
   <stroke
android:width="10dp"
```

```
android:color="#ff00ff20"
android:dashGap="5dp"
        android:dashWidth="5dp"
        />

</shape>
```

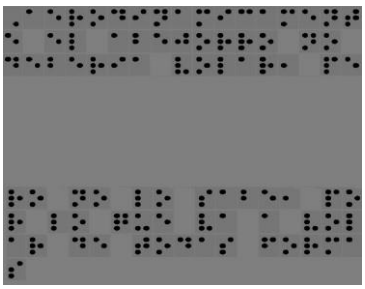
6.3.2.7 **boton_circulo_defecto**: Le proporciona a la vista asociada un color gris asociado  con código hexadecimal #ff00ff20 , un ancho y alto de 90 dp, un valor geometrico en forma de ovalo, un bordeado de tipo stroke, todo lo anterior cuando se encuentra  en el estado no – oprimido.

```
<?xml version="1.0" encoding="utf-8"?>

<shape xmlns:android="http://schemas.android.com/apk/res/android"
android:shape="oval">
    <solid android:color="#fff5f5fb" />
    <size android:height="90dp" android:width="90dp"/>
    <stroke
android:width="10dp"
android:color="#fff6f5fd"
android:dashGap="6dp"
        android:dashWidth="20dp"
        />

</shape>
```

6.3.2.8 **carpeta mipmap:** Se encarga de almacenar los archivos .png y .jpg que va a interactuar con las vistas de las actividades de manera visual.

| NOMBRE | FIGURA | DESCRIPCIÓN |
|---|---|---|
| Fondo |  | Fondo de splash tiene la frase:<br>Aerodinámicamente el abejorro no debería volar, pero no lo sabe, por lo que va a volar de todas formas.<br>-Mary Kay Ash. |

| | | |
|---|---|---|
| Logo | BRAILLE TOTAL | Logo de la aplicación este será el lanzador que el usuario elige para ingresar al aplicación. |
| Mas | +4 | Aparece cuando en cualquiera de las actividades del módulo test realiza correctamente la tarea asignada. |
| Menos | -2 | Aparece cuando en cualquiera de las actividades del módulo test realiza incorrectamente la tarea asignada. |
| Ying | | Imagen que será animada en el splash de la aplicación. |

6.3.2.9 **carpeta raw: Se** encarga de almacenar los archivos .mp3 que va a interactuar con las vistas de las actividades en forma de audio.

6.3.2.10

| NOMBRE | DESCRIPCIÓN |
|---|---|
| Bambú | Se activa cuando el usuario da click sobre la pantalla en las actividades del módulo test. |
| Correcto | Se activa cuando el usuario acierta en las actividades del módulo test. |
| Incorrecto | Se activa cuando el usuario no acierta en las actividades del módulo test. |
| Lose | Se activa cuando el usuario pierde en las actividades del módulo test. |
| Uno | Se activa cuando el usuario da click sobre el punto número uno en los teclados que simulan los símbolos del sistema Braille. |
| Dos | Se activa cuando el usuario da click sobre el punto número dos en los teclados que simulan los símbolos del sistema Braille. |
| Tres | Se activa cuando el usuario da click sobre el punto número tres en los teclados que simulan los símbolos del sistema Braille. |
| Cuatro | Se activa cuando el usuario da click sobre el punto número cuatro en los teclados que simulan los símbolos del sistema Braille. |
| Cinco | Se activa cuando el usuario da click sobre el punto número cinco en los teclados que simulan los símbolos del sistema Braille. |
| Seis | Se activa cuando el usuario da click sobre el punto número seis en los teclados que simulan los símbolos del sistema Braille. |

## 7. FUNDAMENTOS DE PROGRAMACIÓN:

La aplicación cuenta con tres módulos principales (escritura, aprendizaje y test).

### 7.1 Fundamentos modulo aprendizaje:

Las clases y actividades incluidas dentro de este módulo son las siguientes:

- Confirmacion
- Dieresis
- Display

- letras
- llaves
- menuPrincipal
- numeros
- operacionesMenu
- Ortograficos
- SignoGenerador
- tablas
- TecladoAprendizaje
- TecladoAprendizajeNumero
- TTSManager
- Vocales

**7.2 Fundamentos modulo escritura:**

Las clases y actividades incluidas dentro de este módulo son las siguientes:

- Display
- EleccionVelocidad
- EleccionTeclado
- EscrituraPrimer
- EscrituraSegundo
- Tablas
- TTSManager
- Llaves

**7.3 Fundamentos modulo test:**

Las clases y actividades incluidas dentro de este módulo son las siguientes:

- Cantidad
- MultiTouchTest
- Secuencia
- Test
- TTSManager

### 7.4 Actividades y clases extra:

Las siguientes clases y actividades no están dentro de los 4 módulos anteriores pero hacen parte vital de la aplicación.

- Base_de_datos
- EleccionModulos
- Splash

## 8. IMPLEMENTACIÓN DE CÓDIGO DE IMPORTANCIA:

### 8.1. Clases:

#### 8.1.1 TTSManager:

**metodo initQueue:** Recibe como parámetro un String llamado texto e invoca a un método de la superclase TextToSPeech llamado speak lo que permite acceder al sintetizador de voz del dispositivo móvil y reproducir por voz el String texto.

```
public void initQueue(String text0) {
if (isLoaded)
        mTts.speak(texto, TextToSpeech.QUEUE_FLUSH, null);
else
        Log.e("error", "TTS Not Initialized");
}
```

#### 8.1.2 Display:

**metodo getScreenSiseW:** Recibe como parámetro el contexto de la actividad que quiera acceder a este y devuelve un int con  el tamaño del ancho de la pantalla a la que se le relacione el contexto, esto es vital para que el contenido de las vistas puedan re evaluar el tamaño de alto y ancho.

```
public static int getScreenSizeW(Context context) {
    if (screenSizeW == -1) {
```

```
        DisplayMetrics displayMetrics = context.getResources().getDisplayMetrics();
try {
            screenSizeW =
DisplayMetrics.class.getField("widthPixels").getInt(displayMetrics);
        } catch (Exception e) {
            screenSizeW = 320;
        }
    }
    return screenSizeW;
}
```

**8.1.3 Base_de_datos:**

**metodo OnCreate:** Este método recibe una referencia de tipo SQLiteDatabase y mediante el método execSQL de la superclase SQLiteOpernHelper que permite crear las tablas de la base de datos que va a manejar la aplicación.

```
@Override
public void onCreate(SQLiteDatabase db) {
db.execSQL(Tablas.CREATE_TABLE);
db.execSQL(Tablas.CREATE_TABLE_G);
db.execSQL(Tablas.CREATE_TABLE_NUM);
db.execSQL(Tablas.CREATE_TABLE_V);
db.execSQL(Tablas.CREATE_TABLE_S);
db.execSQL(Tablas.CREATE_TABLE_O);
db.execSQL(Tablas.CREATE_TABLE_E);
db.execSQL(Tablas.CREATE_TABLE_D);
db.execSQL(Tablas.CREATE_TABLE_C);
}
```

**Atributo DB_NAME:** De tipo String toma el nombre que la base de datos va a tener.

```
private static final String DB_NAME = "base_braille.sqlite";
```

**Atributo DB_SCHEME_VERSION:** De tipo int toma el valor de la versión en la que se encuentra la base de datos, en caso de tener que realizar cambios en la estructura de la base de datos se sobre-escribe el valor del esquema y no se pierden los datos de la estructura anterior.

```
private static final int DB_SCHEME_VERSION = 1;
```

**8.1.4 Tablas:**

**método generarContentValues:** Es de tipo ContentValues, recibe 4 valores de tipo String la combinación de letra por ejemplo: “110000” que corresponde a la b, el nombre de la letra por ejemplo “b”, y el estado de la letra “aprobado o no aprobado”, devuelve un contenedor con cada uno de los valores anteriormente mencionadas.

```
public ContentValues generarContentValues(String combi_letra, String
nom_letra, String estado_letra){

   ContentValues valores= new ContentValues();
valores.put(BR_COMBINATION ,combi_letra);
valores.put(BR_NAME,nom_letra);
valores.put(BR_WIN,estado_letra); return
valores;
}
```

**metodo insertar:** De tipo void permite insertar datos en la tabla letras con sus respectivos datos, y lo inserta en un contentValues que contendrá todos los datos.

```
public void insertar (String combi_letra, String nom_letra, String
estado_letra){ db.insert(TABLE_NAME, null,
generarContentValues(combi_letra, nom_letra, estado_letra));
}
```

**método buscarPorEstado:** De tipo Cursor recibe un String que sera el estado de las letras que vamos a buscar, si se hará un filtro con las letras que ya se han aprobado o las que aún no se han aprobado, retorna el resultado de la consulta.

```
public Cursor buscarPorEstado (String estado){

   String [] columnas = new String
```

```
[]{BR_ID,BR_COMBINATION,BR_NAME,BR_WIN};
   return db.query(TABLE_NAME,columnas,BR_WIN+ "=?", new

String[]{(estado)},null,null,null);

}
```

**metodo actualizar_por_index:** De tipo void permite actualizar el estado de un carácter con respecto al index en el que se encuentra almacenada en la tabla de la base de datos, recibe un número (el index) y un win_nuevo (el estado nuevo) que actualizara al carácter de estado aprobado a estado no aprobado según corresponda.

```
public void actualizar_por_index(String numero ,String win_nuevo){

   int l= Integer.parseInt(numero);

   db.execSQL("UPDATE " + TABLE_NAME + " SET '" + BR_WIN + "' = '" +
win_nuevo + "' WHERE " + BR_ID + " = " + l + " ;");

}
```

## 9. ACTIVIDADES:

### 9.1 Cantidad:

**método onTouchEvent:** De tipo booleano recibe un evento (up, down, etc.) que permite guardar las coordenadas X y Y del touch sobre la pantalla y guardarlo en vectores de 6 posiciones (máximo), logrando que se pueda realizar actividades con estos datos, posteriormente edita los vectores a medida que se deja de tocar o se toca la pantalla.

```
public boolean onTouchEvent( MotionEvent event) {

        corx = (int) event.getX();
cory = (int) event.getY();

        fondo.invalidate();

        int action= event.getAction() & MotionEvent.ACTION_MASK;

        int pointerIndex = ( event.getAction() &
```

```
MotionEvent.ACTION_POINTER_ID_MASK) >>
            MotionEvent.ACTION_POINTER_ID_SHIFT;
/* Se marcan tachadas porque fueron renombradas por index  pero si las
escribimos con index no soportara versiones anteriores*/          int
pointerId = event.getPointerId(pointerIndex);          switch(action) {
case MotionEvent.ACTION_DOWN:
            /*
            if( siHay=false) {
atras();                    paso++;
            }else {*/
                mp = MediaPlayer.create(Cantidad.this, R.raw.bambu);
mp.start();

            mp.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {

                public void onCompletion(MediaPlayer mp) {

                    mp.release();

                }

                });

                if(bandera ==0 ) {
bandera=1;                          evaluar();
                }
atras();
paso++;
             //   atras();
               // paso++;
           //  }
          case MotionEvent.ACTION_POINTER_DOWN:
            tocado[pointerId]=true;
x[pointerId]=(int) event.getX(pointerIndex);
y[pointerId]=(int) event.getY(pointerIndex);
mas++;                corx=  x[0];cory=  y[0];
cor2x=  x[1];cor2y=  y[1];                cor3x=
x[2];cor3y=  y[2];                cor4x=  x[3];cor4y=  y[3];
cor5x=  x[4];cor5y=  y[4];                cor6x=
x[5];cor6y=  y[5];                break;            case
```

```
MotionEvent.ACTION_UP:            case
MotionEvent.ACTION_POINTER_UP:

            // Toast.makeText(getApplicationContext(), " id: "+pointerId,
Toast.LENGTH_SHORT).show();            case
MotionEvent.ACTION_CANCEL:
tocado[pointerId]=false;              menos++;
break;            case
MotionEvent.ACTION_MOVE:
            int pointerCount = event.getPointerCount();
for(int i=0; i < pointerCount; i++) {

                pointerIndex=i;                pointerId =
event.getPointerId(pointerIndex);
x[pointerId]=(int) event.getX(pointerIndex);
y[pointerId]=(int) event.getY(pointerIndex);
            }

            corx=  x[0];cory=  y[0];
cor2x=  x[1];cor2y=  y[1];
cor3x=  x[2];cor3y=  y[2];
cor4x=  x[3];cor4y=  y[3];
cor5x=  x[4];cor5y=  y[4];
cor6x=  x[5];cor6y=  y[5];
            break;
        }

        return true;
    }  }
```

**9.2 Teclado_Aprendizaje:**

**método consulta_base_datos:** De tipo void permite consultar cada una de las tablas de la base de datos de los caracteres que ya hayan sido aprobados y almacenando los cuatro atributos en arrays para su consulta aleatoria posteriormente. **public void consulta_base_datos(){**

```
    cursor= manager.buscarPorEstado("TRUE");
```

```
        while ( cursor . moveToNext ()) {

combinacion_array.add(cursor.getString(cursor.getColumnIndex(manager.BR_
COMBINATION)));

letra_array.add(cursor.getString(cursor.getColumnIndex(manager.BR_NAME)));
        }
        try {
        }catch(Exception e){
        }
        cursor= manager.buscarPorEstado_V("TRUE");
        while ( cursor . moveToNext ()) {

combinacion_array.add(cursor.getString(cursor.getColumnIndex(manager.VR_
COMBINATION)));

letra_array.add(cursor.getString(cursor.getColumnIndex(manager.VR_NAME)));
        }
        try {
        }catch(Exception e){
        }
        cursor= manager.buscarPorEstado_D("TRUE");
        while ( cursor . moveToNext ()) {

combinacion_array.add(cursor.getString(cursor.getColumnIndex(manager.DR_
COMBINATION)));
letra_array.add(cursor.getString(cursor.getColumnIndex(manager.DR_NAME)));
        }
        try {
        }catch(Exception e){
        }
        cursor= manager.buscarPorEstado_S("TRUE");
        while ( cursor . moveToNext ()) {

combinacion_array.add(cursor.getString(cursor.getColumnIndex(manager.SR_CO
MBINATION)));

letra_array.add(cursor.getString(cursor.getColumnIndex(manager.SR_NAME)));
        }
```

```
        try {
        }catch(Exception e){
        }
        cursor= manager.buscarPorEstado_O("TRUE");
        while ( cursor . moveToNext () ) {

combinacion_array.add(cursor.getString(cursor.getColumnIndex(manager.OR_CO
MBINATION)));

letra_array.add(cursor.getString(cursor.getColumnIndex(manager.OR_NAME)));
         }
        try {
        }catch(Exception e){
        }
       cursor= manager.buscarPorEstado("TRUE");
        while ( cursor . moveToNext () ) {

combinacion_array.add(cursor.getString(cursor.getColumnIndex(manager.BR_CO
MBINATION)));


letra_array.add(cursor.getString(cursor.getColumnIndex(manager.BR_NAME)));

         }

        try {
        }catch(Exception e){

        }

    if(combinacion_array.size()==0 ){

        siHay=false;

    } }
```

**9.3 EleccionModulo:**

**metodo deslizar :** De tipo void permite guardar los parámetros de la página en la que se encuentre para poder poner el texto del array res, además de poder llamar al método initQueue que hará que el sintetizador de voz notifique al usuario del contenido de cada página.

```
String[] res = {"Aprendizaje","Escritura","Test",”Instrucciones”}; public

void deslizar(){

   ViewPager.OnPageChangeListener myOnPageChangeListener =
         new ViewPager.OnPageChangeListener(){
            @Override
            public void onPageScrollStateChanged(int state) {
numero =  viewPager.getCurrentItem();
            }
            @Override
            public void onPageScrolled(int position,
                                       float positionOffset, int positionOffsetPixels) {
               numero =  viewPager.getCurrentItem();
               ttsManager.initQueue("" + res[numero]);
            }
            @Override
            public void onPageSelected(int position) {
            }}; }
```

**9.4 EscrituraPrimer:**

**método portapapeles:** De tipo void permite al usuario guardar en el portapapeles el texto que se halla hecho con el simulador de signo generador (plantilla de seis botones), notifica al usuario que ha sido copiado visualmente y por voz.

```
public void portapapeles() {
      MainLayout.setOnLongClickListener(new View.OnLongClickListener() {
         @Override
         public boolean onLongClick(View v) {
            crear_archivo();
String text;
            text = letra_mostrada.getText().toString();
myClip = ClipData.newPlainText("text", text);
myClipboard.setPrimaryClip(myClip);
Toast.makeText(getApplicationContext(),
"Copiado",Toast.LENGTH_SHORT).show();
```

```
            ttsManager.initQueue(letra_mostrada.getText() + "");
copiar();
            //   guardar_ultima_cadena();
            //Toast.makeText(getApplicationContext(),"la ultima palabra
es"+ultima_palabra, Toast.LENGTH_SHORT).show();
return true;
        }
}
```

**9.5 Encontrar:**

**método generarCadena:** Permite generar aleatoriamente una secuencia binaria de 6 posiciones , validando que nunca va a tener una cadena de seis ceros.

```
public void generarCadena(){
   combinacion="";
   for(int i=0;i<6;i++){
      combinacion=combinacion+""+ran.nextInt(2);
   }
   if(combinacion.equals("000000")){
      generarCadena();
   }
}
```

9.6 Splash:

**metodo insertar:** De tipo void permite insertar 10 registros con los números, la combinación de puntos y el estado de aprobación a la base de datos dentro de la tabla números, esto se puede realizar debido a que se crea un objeto de la clase tablas y se accede al método insertar_N que habilita ingresar nuevos registros a dicha tabla.

```
manager = new Tablas(this); public

void insertar (){

      manager.insertar_N("100000", "1", "FALSE");
      manager.insertar_N("110000", "2", "FALSE");
      manager.insertar_N("100100", "3", "FALSE");
      manager.insertar_N("100110", "4", "FALSE");
      manager.insertar_N("100010", "5", "FALSE");
```

```
        manager.insertar_N("110100", "6", "FALSE");
        manager.insertar_N("110110", "7", "FALSE");
        manager.insertar_N("110010", "8", "FALSE");
        manager.insertar_N("010100", "9", "FALSE");
        manager.insertar_N("010110", "0", "FALSE");
    }
```

**Nota:** La anterior es la estructura para ingresar datos en la base de datos de cada una de las tablas, se utilizó como ejemplo los números, pero es equivalente para cada uno de los caracteres pertenecientes a la aplicación.

**método crearArchivo:** Permite crear un archivo con el puntaje logrado en dicha actividad, lo realiza en modo privado, es decir que sobrescribe cada uno de los datos que anteriormente habían estado en el txt, en caso de no poder realizar la sobre escritura de los datos el usuario será notificado.

```
public void crearArchivo(){
    try
    {
        OutputStreamWriter fout=
new OutputStreamWriter(
                    openFileOutput("puntos.txt", Context.MODE_PRIVATE));
fout.write("" + puntos);
        fout.close();

    }
    catch (Exception ex)
    {
        Toast.makeText(getApplicationContext(),"Error al escribir fichero a memoria
interna", Toast.LENGTH_SHORT).show();
        //  Log.e("Ficheros", "Error al escribir fichero a memoria interna");
    } }
```

**ANEXO 2.**

MANUAL DEL USUARIO.

APLICACIÓN PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID PARA USUARIOS CON DISCAPACIDAD VISUAL O CON INTERES EN APRENDER LA COMUNICACIÓN BASADA EN EL SISTEMA BRAILLE.

ERIKA MAYERLI GUZMAN RAMIREZ
JOHAN SEBASTIAN SUAREZ PEREZ

UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS
FACULTAD TECNOLOGICA
TECNOLOGIA EN SISTEMATIZACION DE DATOS
BOGOTA
2015

CONTENIDO.

INTRODUCCION..... 3
REQUERIMIENTOS DEL DISPOSITIVO.....3
INSTALACION DE LA APLICACIÓN. .....3
NAVEGANDO POR BRAILLE TOTAL .....7
MODULOS DE BRAILLE TOTAL .....8
MODULO DE APRENDIZAJE. .....8
Aprendizaje .....8
MODULO DE ESCRITURA..... 15
Teclado Horizontal ..... 16
Teclado Vertical ..... 19
MODULO DE TEST. ..... 20
MODULO DE MANUAL. ..... 23

## 1. **INTRODUCCION.**

La aplicación Braille Total es una aplicación para dispositivos móviles (celulares o Tablet) con sistema operativo Android (igual o superior al api 15) que ayuda a usuarios que tienen discapacidad visual como a usuarios interesados en aprender el sistema Braille mediante herramientas como respuestas por voz, teclados en los que podrá realizar textos y test en donde se evaluara su conocimiento del sistema Braille.

## 2. REQUERIMIENTOS DEL DISPOSITIVO.

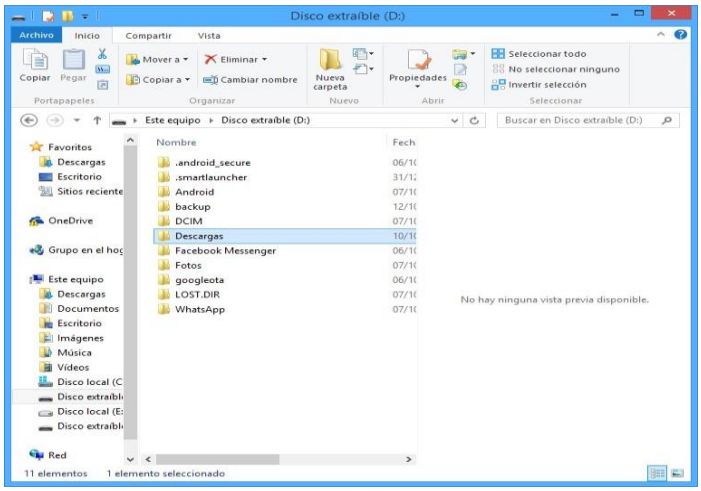
Para buen uso del aplicativo se recomienda:

| **Sistema operativo** | **Android ® (igual o superior al api 15)** |
|---|---|
| **Memoria** | Se recomienda 5 MB para el buen uso. |
| **Pantalla** | Se necesita una pantalla para celulares de mínimo 3.5” |
| **Aplicaciones Instaladas** | - Lector de archivos PDF (por ejemplo Acrobat Reader).<br>- Reproductor de sonido.<br>- Internet, el dispositivo debe tener acceso a Internet, ya que la aplicación detecta si el dispositivo tiene o no el sintetizador de voz, si no lo tiene entonces lo descarga automáticamente.<br>- Talk back: Es una herramienta que cuando está activa, el dispositivo habla y describe cada uno de los elementos que son seleccionados o activados. Se trata de una ayuda por voz que es capaz de leer lo que ocurre en el teléfono e ir guiando al usuario en su navegación por los menús y las aplicaciones. |

## 3. INSTALACION DE LA APLICACIÓN.

Para la instalación de la puede pasarla al dispositivo por medio bluetooth, correo electrónico o siguiendo los pasos a continuación, para esto debe conectar el dispositivo en un computador y acceder a la carpeta descargas o download del dispositivo.

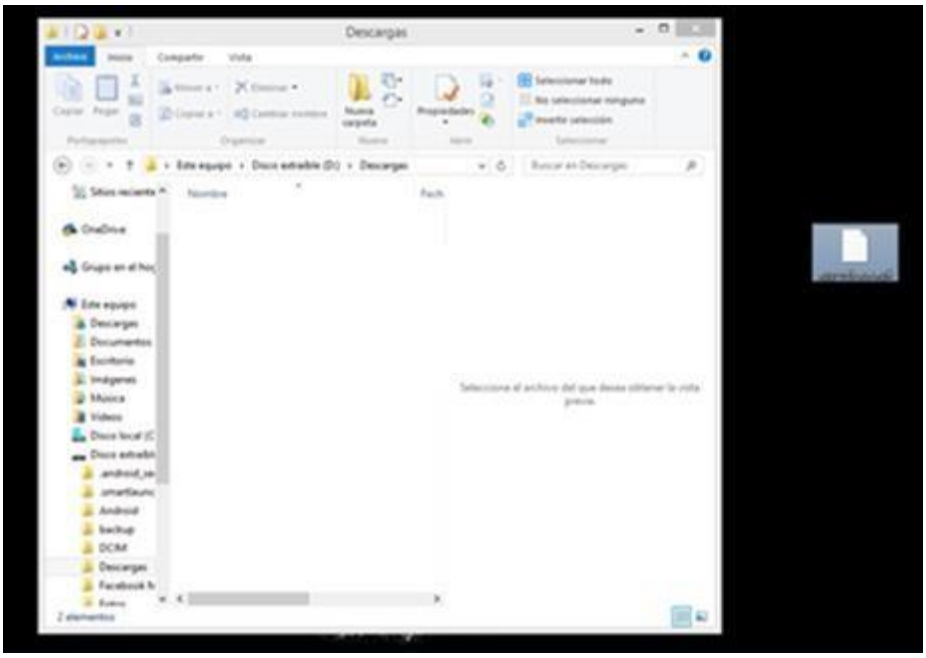
*Imagen 1 – Carpeta descargas*




*Fuente: Autores.*

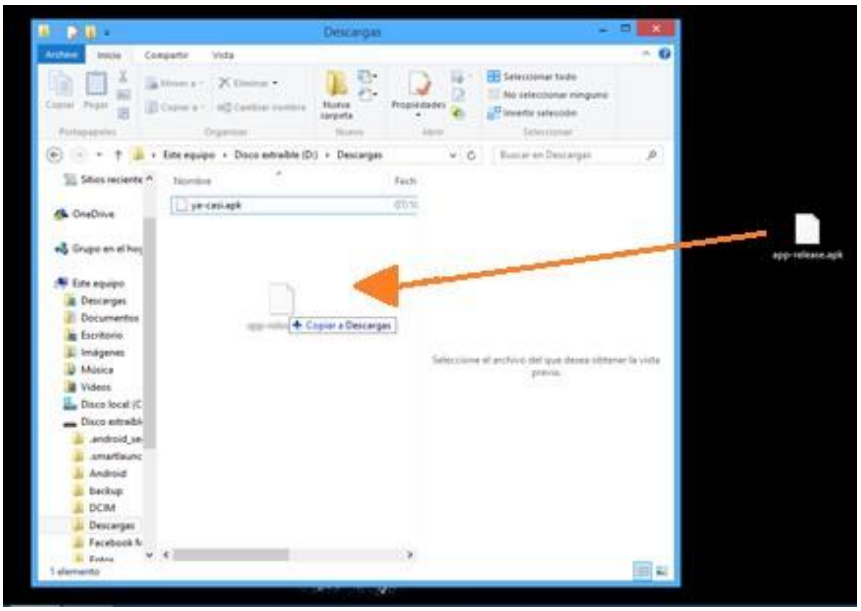
Luego copie el apk de la aplicación (que está ubicado en el computador) a esta carpeta

*Imagen 2 – Selección del apk*




*Fuente: Autores.*

*Imagen 3 - Copia apk*




*Fuente: Autores.*

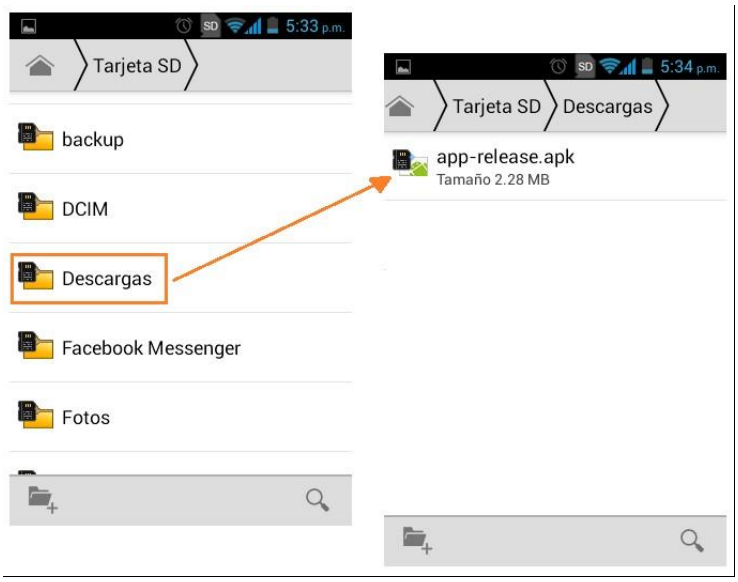
Después vaya al dispositivo móvil y ubique la carpeta llamada “Mis Archivos”

*Imagen 4 – Carpeta Archivos*

*Fuente: Autores.*

Ingrese y ubique la carpeta “Descargas” o “Download”, allí encontrara el apk que acabo de copiar al dispositivo

*Imagen 5 _Ubicación apk en carpeta Descargas*




*Fuente: Autores.*

Al oprimir en el apk le preguntara si desea instalar la aplicación, cuando acepte espere un momento mientras se instala, será notificado cuando finalice la instalación

*Imagen 6 – Confirmación e instalación de la aplicación*

*Fuente: Autores.*

Cuando explore el menú del dispositivo, encontrara la aplicación

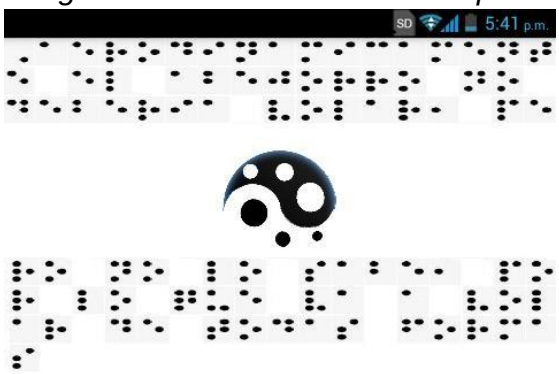
*Imagen 7- Ubicar aplicación Instalada*

*Fuente: Autores.*

## 4. NAVEGANDO POR BRAILLE TOTAL

Al ingresar a Braille total encontrara una pantalla de inicio con una frase en Braille "Aerodinámicamente el abejorro no debería volar, pero no lo sabe, por lo que va a volar de todas formas.-Mary Kay Ash" y el logo.

*Imagen 8 – Pantalla inicio de la aplicación.*




*Fuente: Autores.*

El menú principal consta de tres opciones, la primera corresponde al módulo “Aprendizaje” donde podrá acceder a la simbología del sistema Braille (dividida en categorías); la segunda opción, “Escritura” en la que tendrá dos opciones de teclado (vertical y horizontal); la tercera “Test” donde encontrara tres diferentes opciones de juegos y por último el módulo de “Manual” en donde podrá encontrar el manual del usuario.

*Imagen 9 – Módulos de la Aplicación*

*Fuente: Autores.*

Nota:

Para ingresar alguno de los módulos de la aplicación solo debe mantener oprimida por unos segundos la pantalla.

Para salir debe oprimir dos veces la pantalla.

## 5. MODULOS DE BRAILLE TOTAL

En la aplicación encontrara cuatro módulos:

### a. MODULO DE APRENDIZAJE:

- **Aprendizaje:** Para su comodidad el modulo fue divido en seis categorías

- Letras:

*Imagen 10 - Modulo Letras*

*Fuente: Autores.*

Aquí están las 27 letras del abecedario, cuando ingrese a la letra “a”, aparecerá el signo generador que son los puntos 1, 2, 3, 4, 5 y 6 los cuales deberá oprimir en un tiempo límite de 70 segundos, cuando el botón no se oprima correctamente el usuario tendrá ayudas auditivas y en pantalla para que ubique el botón.

*Imagen 11 - Signo Generador*

*Fuente: Autores.*

Después de esto y si lo digito bien encontrara la letra a, si la digita bien entonces le pedirá que confirme de nuevo el carácter, al confirmar le permitirá avanzar a la siguiente letra.

*Imagen 12 – Ejemplo letras*

*Fuente: Autores.*

o Números:

Encontrará los números del 1 al 0 (que son los mismos puntos de las letras de la “a” a la “j”)

*Imagen 13 - Modulo Números*

*Fuente: Autores.*

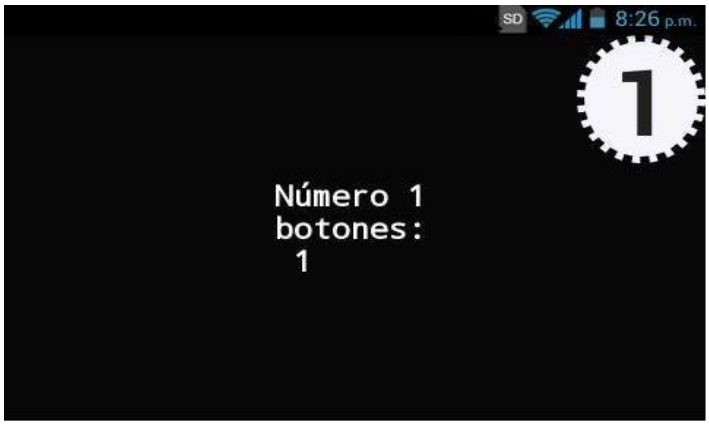
Siempre que ingrese a un número este se verá en pantalla luego pedirá que digite la “llave” que genera los números en este caso los botones 3, 4, 5 y 6.

*Imagen 14 - Ejemplo numero*

*Fuente: Autores.*

Luego de digitarla aparecerá el número:

*Imagen 15 – Ejemplo Numero*

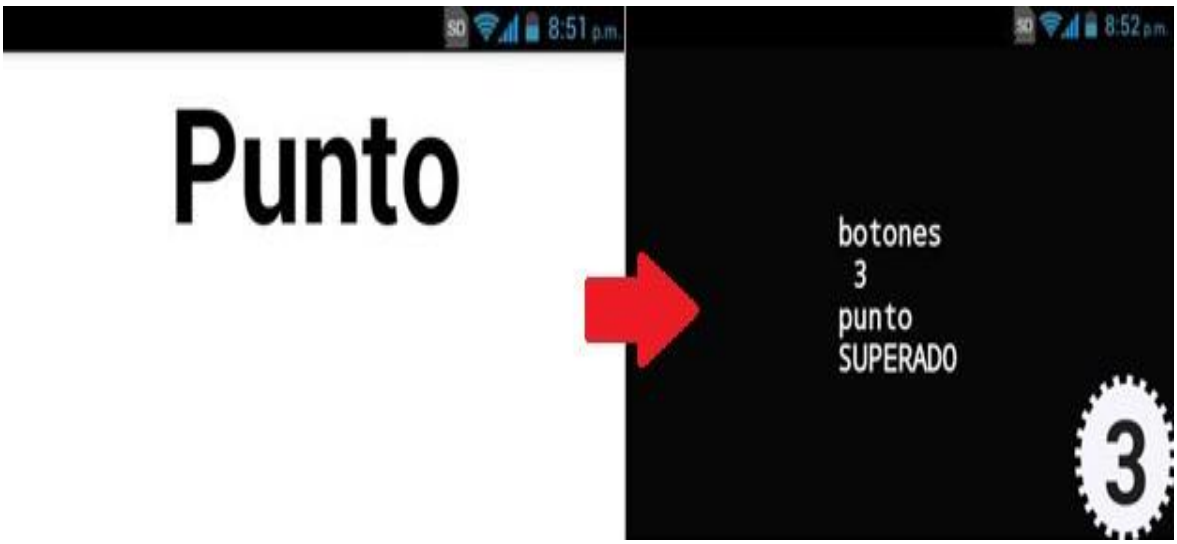



*Fuente: Autores.*

- Signos Ortográficos:

El funcionamiento es similar al de los módulos anteriores, ingresa al signo ortográfico y pide que digite el botón que lo genera; ejemplo: punto, le pide que digite el botón (3)

*Imagen 16 - Ejemplo Signos Ortográficos*




*Fuente: Autores.*

Luego de digitarlo correctamente, pide confirmación:

*Imagen 17 – Confirmación de Carácter*

*Fuente: Autores.*

- Vocales Acentuadas:

Al igual que en los signos ortográficos solo digitara los puntos que generan la vocal acentuada, luego tendrá que confirmarla y al pasar una podrá continuar con la siguiente.

*Imagen 18 - Vocales Acentuadas.*

*Fuente: Autores.*

*Imagen 19 - Ejemplo Vocal acentuada.*

*Fuente: Autores.*

- Signos Matemáticos:

Tiene el mismo funcionamiento que los módulos de signos ortográficos y vocales acentuadas.

*Imagen 20 – Signos Ortograficos.*

*Fuente: Autores.*

Digitara los puntos que generan el signo matemático, luego tendrá que confirmarlo y al pasar uno podrá continuar con el siguiente.

*Imagen 21 – Confirmación*

*Fuente: Autores.*

- Diéresis:

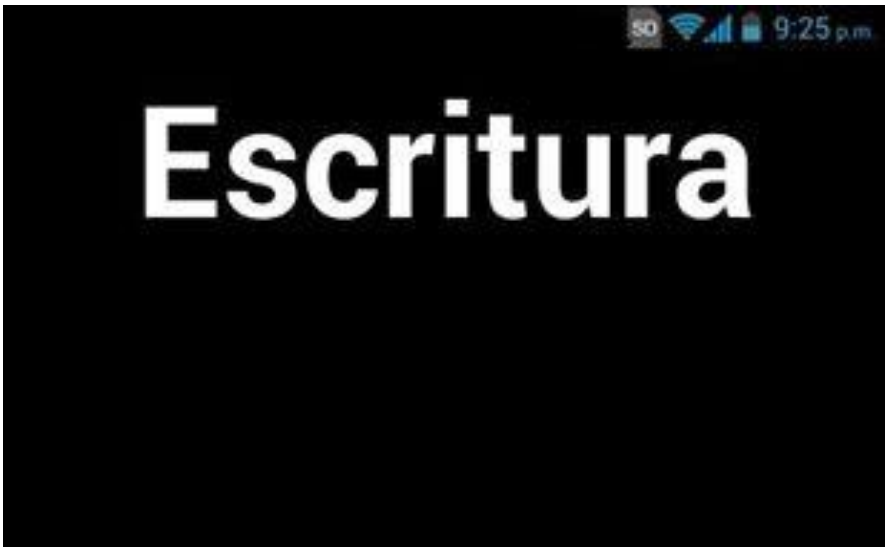
*Imagen 22 - Diéresis*

*Fuente: Autores.*

**NOTA:** Cada vez que el usuario oprime un botón la aplicación le dirá cual presiono, ejemplo, si presiona por ejemplo el botón uno, la aplicación le dirá “uno”; siempre cuenta con ayudas auditivas.

**5.2 MODULO DE ESCRITURA:**

*Imagen 23 –Modulo Escritura.*




*Fuente: Autores.*

En este módulo podrá hacer cualquier escrito, para su comodidad tiene a disposición dos tipos de teclados (vertical y horizontal).

**TECLADO HORIZONTAL:**

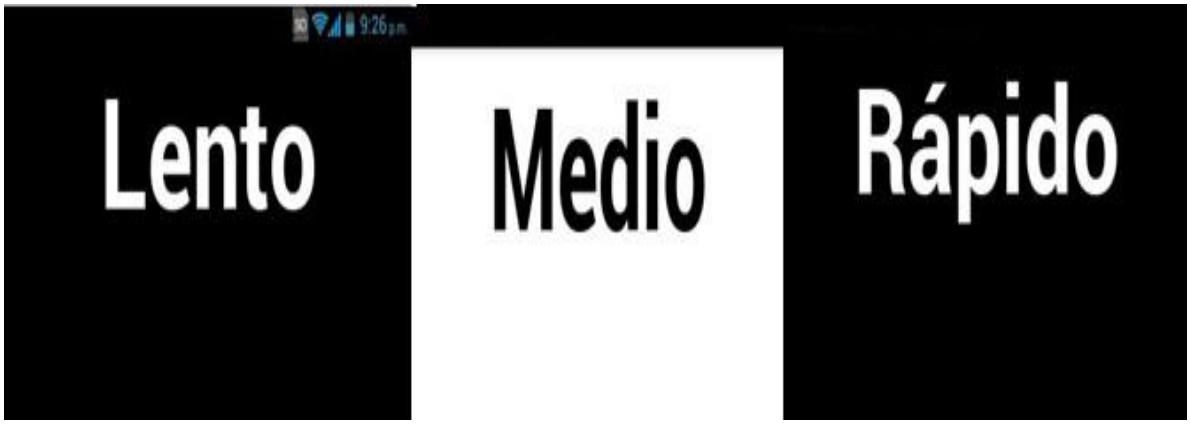
Al momento de ingresar, encontrara la siguiente pantalla

*Imagen 24 – Teclado Horizontal.*

*Fuente: Autores.*

Se le pedirá que escoja el tiempo de respuesta para el momento en el que este escribiendo su texto, tiene tres opciones: lento (7 segundos), medio (5 segundos) y rápido (3 segundos).

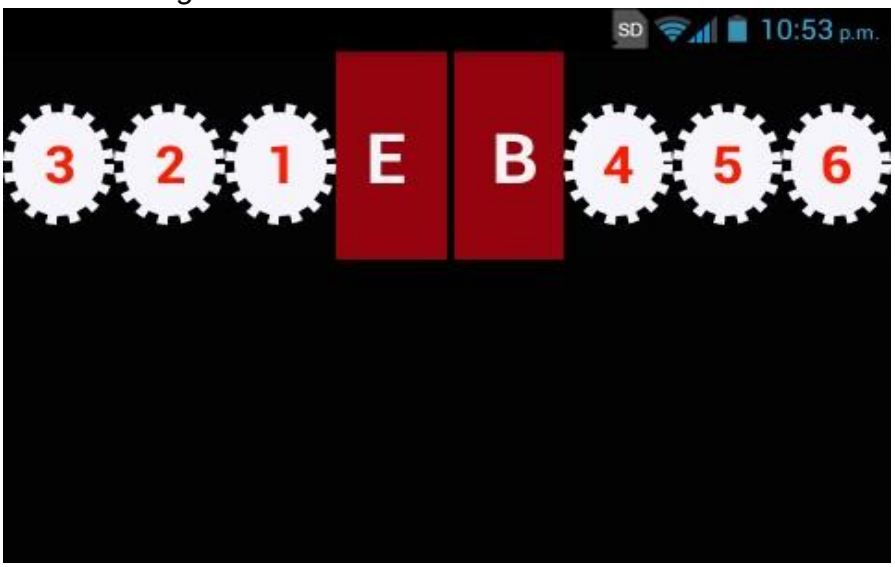
*Imagen 25 - Opciones de Tiempo.*




*Fuente: Autores.*

En la pantalla encontrara 8 botones ubicados horizontalmente en la parte superior, al lado izquierdo se encuentran los tres primeros botones del símbolo generador (3, 2 y 1) a la derecha de estos los botones de espacio y borrar y a la derecha de estos los siguientes tres botones del símbolo generador (4,5 y 6).

*Imagen 26 - Teclado Horizontal.*




*Fuente: Autores.*

Cuando empiece a digitar su texto el sintetizador de voz le dirá lo que vaya digitando según el tiempo de respuesta de su elección.

*Imagen 27 – Ejemplo texto.*

*Fuente: Autores.*

Cuando mantiene oprimido el dedo sobre la pantalla, copiara el texto.

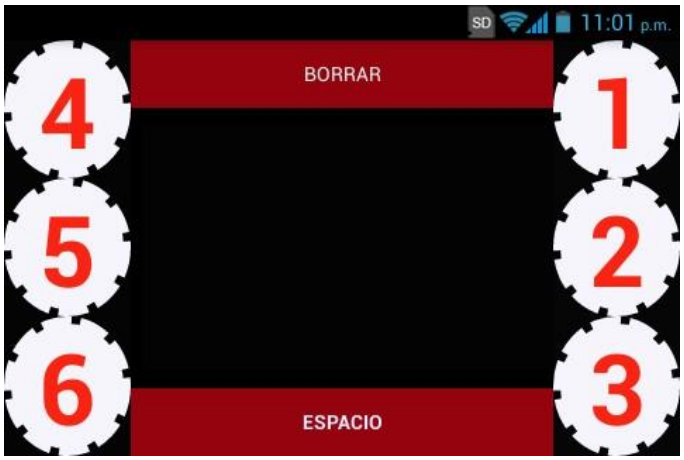
*Imagen 28 – Texto Digitado.*

*Fuente: Autores.*

**TECLADO VERTICAL:**

Su funcionamiento es igual al del teclado horizontal. Este teclado simula los 6 puntos del sistema braille de tal forma que los puntos 1,2 y 3 se encuentran al costado lateral derecho de su pantalla hallados de forma vertical de arriba hacia abajo y los puntos 4,5 y 6 al costado lateral izquierdo de su pantalla hallados de forma vertical de arriba hacia abajo, en el centro superior de la pantalla encontrara el botón de borrar y en el centro inferior el botón de espacio**.**

*Imagen 29 – Teclado Vertical.*

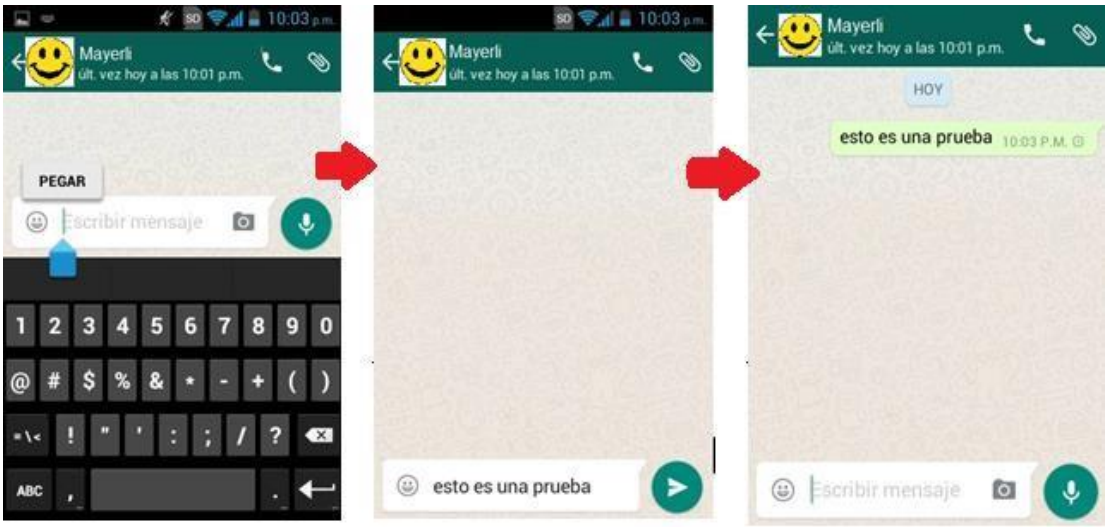



*Fuente: Autores.*

Después de copiar el texto vamos a otra aplicación diferente ejemplo WhatsApp o Facebook, y lo podemos copiar.

NOTA: Si en algún momento desea saber todo el texto que tiene escrito solo debe mantener oprimida la pantalla, si le da click prolongado el texto se guarda y se notifica mediante voz.

*Imagen 30 – Copiado de Texto.*




*Fuente: Autores.*

**5.3 MODULO DE TEST.**

*Imagen 31 –Modulo Test*

*Fuente: Autores*

En este módulo encontrara tres juegos disponibles:

o Cantidad:

*Imagen 32 – Test Cantidad.*

*Fuente: Autores*

En este test se le pregunta al usuario cuantos puntos tiene algún símbolo del alfabeto Braille, lo que debe hacer es oprimir con los dedos la cantidad de puntos que forman el símbolo.

- Secuencia:

*Imagen 33 – Test Secuencia.*

*Fuente: Autores*

Aquí el usuario debe escuchar los puntos que la aplicación dice, luego debe digitarlos en un teclado similar el vertical de escritura, la diferencia es que en este no tiene los botones de borrar y espacio.

- Confirmación:

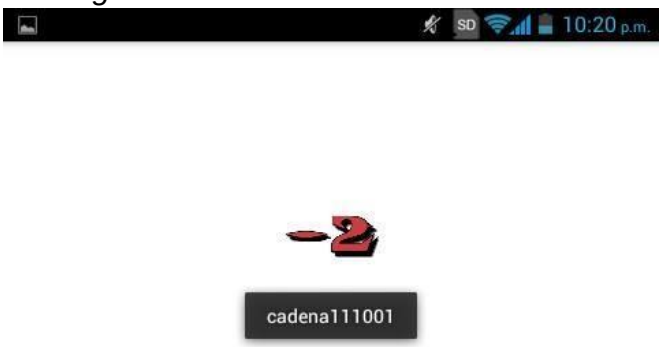
*Imagen 34 – Test Confirmación.*

*Fuente: Autores*

En este módulo la aplicación le dirá un símbolo, con el cual el usuario debe digitar los puntos que lo generan. Cuando el usuario se equivoca la pantalla es blanca y se le descuentan puntos al usuario, cuando acierta se pone verde y se le suman.

**Error:**

*Imagen 35 – Pantalla Error*




*Fuente: Autores*

**Acierto:**

*Imagen 36 – Pantalla Acierto*

*Fuente Autores.*

## 5.4 MODULO DE MANUAL.

En este módulo encontrara el manual del usuario de la aplicación, con las indicaciones de instalación, explicación del aplicativo, etc., solo que este se encuentra disponible para ser escuchado por las personas que tienen discapacidad visual.

## ANEXO 3

Para saber si las personas se encontraban familiarizadas con el uso de dispositivos móviles, el sistema de lectoescritura Braille y si estarían interesadas en una aplicación basada en el mismo, se realizó la siguiente encuesta:

Nombre: _____

1. ¿Tiene algún tipo de discapacidad visual?, si ☐ no ☐ (Ceguera parcial o total)
2. ¿Está familiarizado con los dispositivos móviles? si ☐ no ☐, ¿Con cuántos cuenta? _____
3. De 1 a 10 que tanto conoce sobre el sistema Braille, siendo uno muy poco o nada y diez alto.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|

4. ¿Alguna vez se le ha impartido alguna clase sobre el sistema Braille? si ☐ no ☐

Si su respuesta a la pregunta 4 fue si, conteste la pregunta 4.1, de lo contrario continúe con la pregunta 5.

4.1 ¿De uno a diez que tan cómoda y fluida resultó?, siendo uno mala y 10 excelente.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|

5. ¿Conoce algún software móvil que ayude al aprendizaje del sistema Braille?, si ☐ no ☐
6. De 1 a 10 ¿Qué tan interesado estaría en aprenderlo, si lo encontrara disponible en un dispositivo móvil? , recuerde que 1 es el menor rango y 10 el mayor.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|

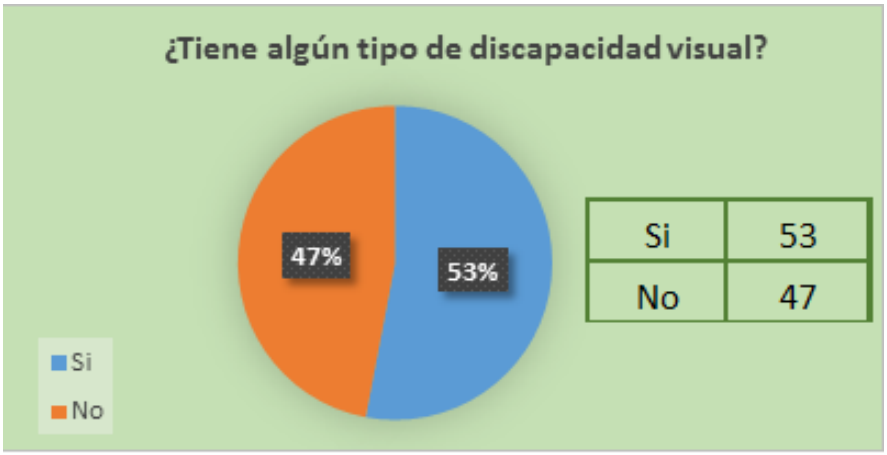
7. De estos ítems cual le gustaría tener dentro de una aplicación basada en el sistema Braille: (Marque con una X)

| | |
|---|---|
| | Aprendizaje |
| | Escritura |
| | Test |
| | Todas las anteriores |
| | Ninguna de las anteriores |

Fue aplicada a un total de 100 personas las cuales tenían algún tipo de discapacidad visual y otras quienes no presentaban discapacidad, por cuestiones de seguridad e integridad personal el listado de encuestados no se anexa, pero si en algún momento es requerido se podrá mostrar a quien así lo disponga.

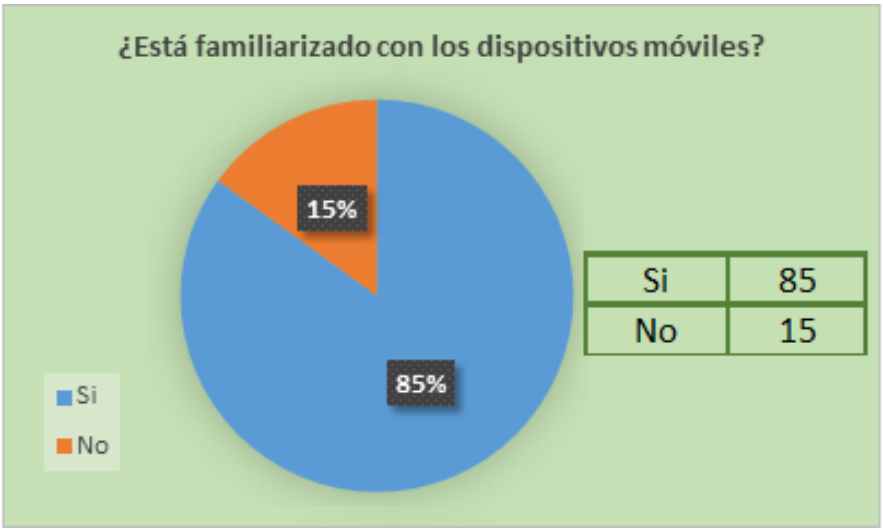
**RESULTADOS OBTENIDOS:**

1. ¿Tiene algún tipo de discapacidad visual?



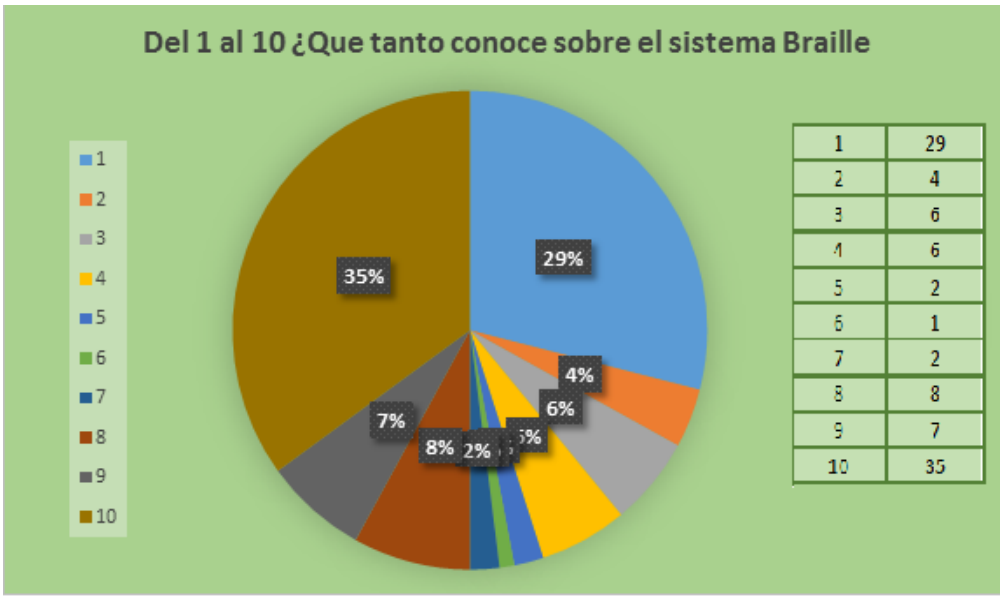

El 53% de personas quienes contestaron a la pregunta uno afirma que tiene discapacidad, teniendo en cuenta que la mayoría la tiene pero que algunas con mayor severidad que las demás.

2. ¿Está familiarizado con los dispositivos móviles?



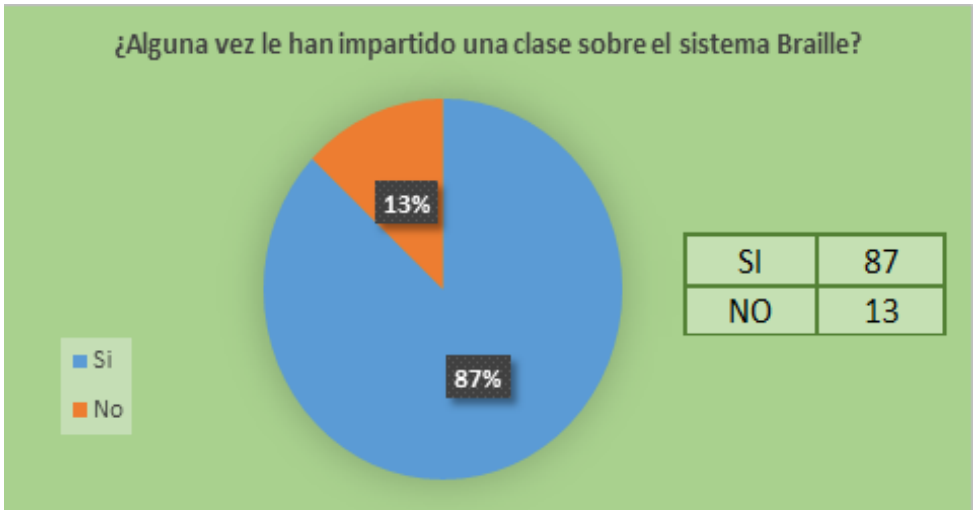

El 85% de los encuestados afirma que está familiarizado, conoce y sabe manipular estos dispositivos.

3. Del 1 al 10 ¿Que tanto conoce sobre el sistema Braille, siendo 1 el menor y 10 el mayor?




Realizando un rango entre las respuestas, el 39% de personas encuestadas tienen poco conocimiento o nada del sistema Braille, el 11% tiene conocimientos medios del Braille y el 50% tiene conocimientos más avanzados o conocen en totalidad el sistema.

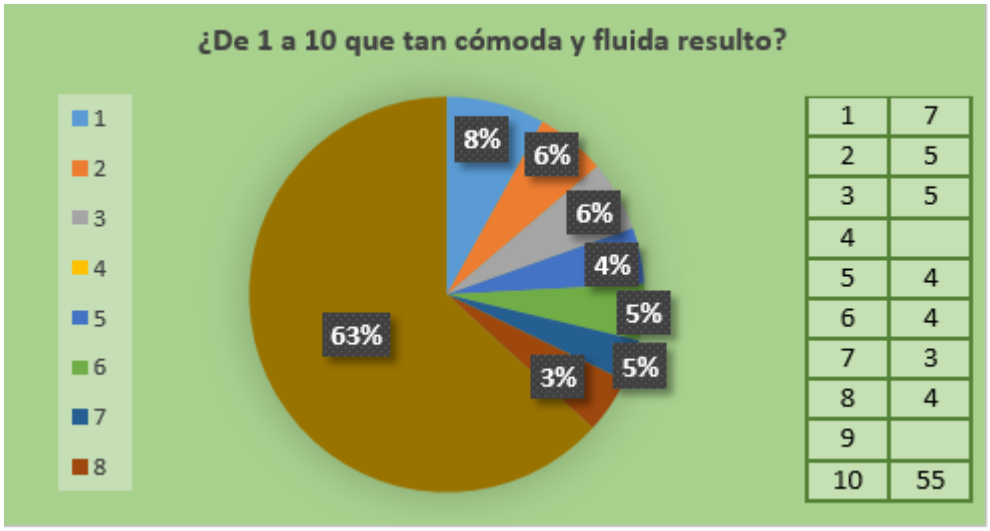
4. ¿Alguna vez le han impartido una clase sobre el sistema Braille?



Respondiendo a la pregunta de si alguna vez se le impartió una clase sobre el sistema Braille el 87% de personas respondió que si, a diferencia del 13% a quienes nunca se les ha impartido una clase sobre el sistema Braille.

Si su respuesta a la pregunta 4 fue si, conteste la pregunta 4.1, de lo contrario continúe con la pregunta 5.

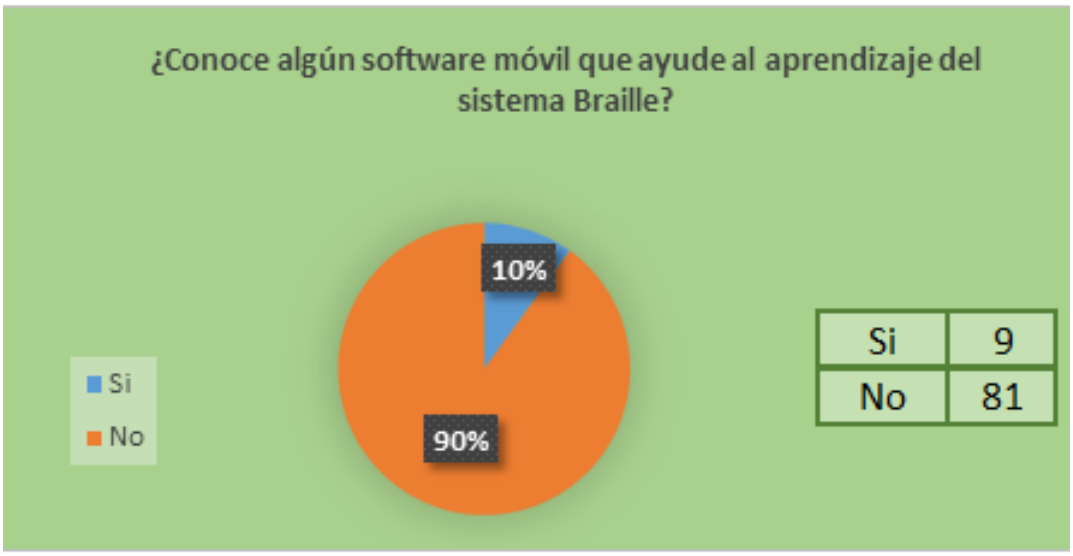
4.1 De uno a diez ¿Qué tan cómoda y fluida resulto?, siendo 1 el menor y 10 el mayor.




| | |
|---|---|
| 1 | 7 |
| 2 | 5 |
| 3 | 5 |
| 4 | |
| 5 | 4 |
| 6 | 4 |
| 7 | 3 |
| 8 | 4 |
| 9 | |
| 10 | 55 |

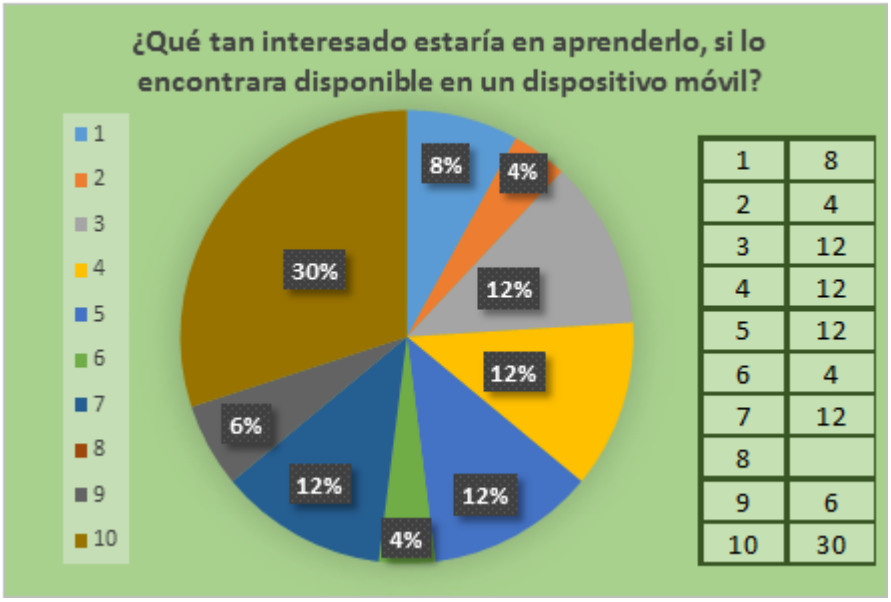
Dado que esta pregunta depende de la anterior, indicando que de los encuestados iniciales solo el 87% respondió esta pregunta, podemos decir que para el 62% la clase resulto cómoda y fluida, para el 8% estuvo en un nivel medio, y para el 17% no fue cómoda y fluida.

5. ¿Conoce algún software móvil que ayude al aprendizaje del sistema Braille?



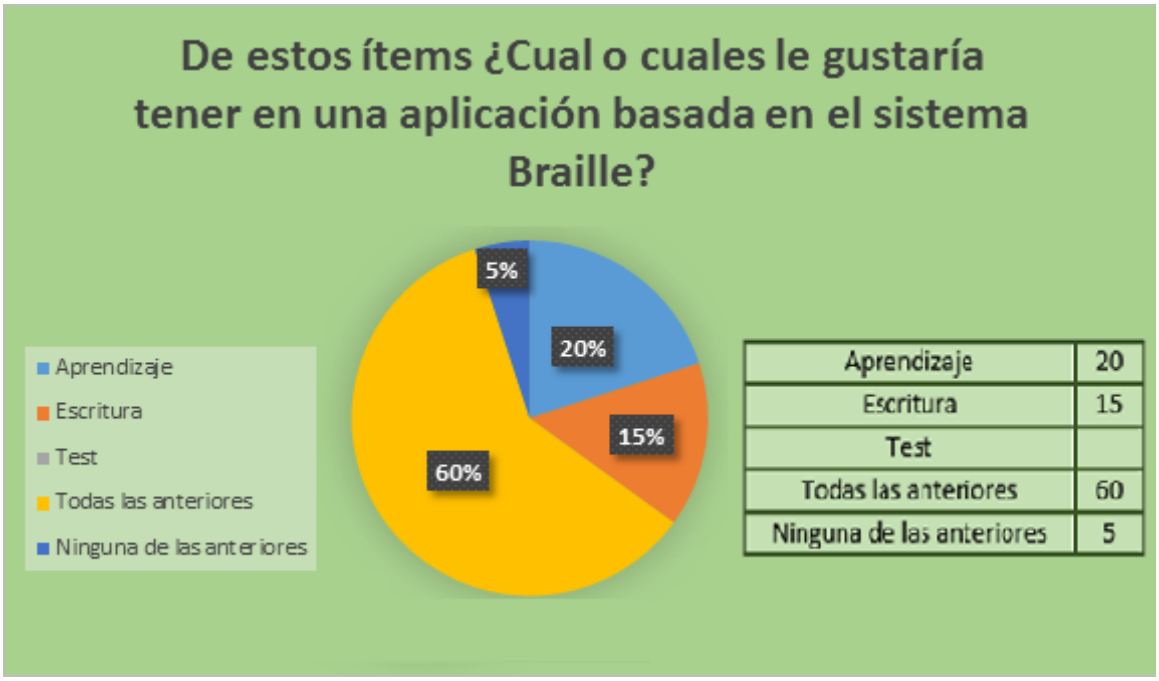

En respuesta a la pregunta de si conoce algún software móvil que ayude al aprendizaje del sistema Braille, el 9% contesto si, y el 81% de los encuestados que no.

6. De 1 a 10 ¿Qué tan interesado estaría en aprenderlo, si lo encontrara disponible en un dispositivo móvil? , recuerde que 1 es el menor rango y 10 el mayor.



Contestando a la pregunta de qué tan interesado estaría en aprenderlo si lo encontrara disponible en un dispositivo móvil el 36% estaría muy interesado, el 40% tendría un interés medio en aprenderlo y el 24% no estaría interesado.

7. De estos ítems ¿Cual o cuales le gustaría tener en una aplicación basada en el sistema Braille?




| | |
|---|---|
| Aprendizaje | 20 |
| Escritura | 15 |
| Test | |
| Todas las anteriores | 60 |
| Ninguna de las anteriores | 5 |

Basados en la respuesta a la pregunta anterior, aunque no les gustaría mucho aprender, se evidencia que a un total del 60% si les gustaría tener una aplicación con estas características.

## ANEXO 4

### SOLUCIÓN TECNOLÓGICA

En primer lugar, se encuentra el módulo de escritura y lectura.

Para este primer módulo se cuenta con una interfaz que contendrá el teclado que homologa la escritura del sistema Braille, contara con seis botones, que se dividirán en dos grupos de tres que harán las siguientes acciones:

1. Cada vez que el usuario pulse una combinación de botones específica, el sistema será capaz de comunicarle cual fue la combinación que realizo, es decir que letra del alfabeto significa dicha combinación con respecto al sistema Braille.
2. A medida que el usuario va ingresando caracteres al sistema, con dichas combinaciones, se ira guardando lo que este valla ingresando, esto con el fin de que el mismo pueda solicitar en cualquier momento dicha información, mediante el pulso prolongado de la pantalla.
3. Al ser re direccionado a la información que el usuario ingreso el sistema será capaz de comunicarle la información mediante un sintetizador de voz.

Para el desarrollo del teclado que homologa el sistema Braille se necesita:

a. <gap: plugin name=”org.tt.plugin.tts” versión=”1” />
   Plugin para javascritp que permitirá acceder al sintetizador de voz del dispositivo móvil, esto con el fin de que el usuario sepa que letra a ingresado con el teclado.
b. <plugin name=”vibrator” value=”org.phonegap.vibrator”>
   </Plugin>: plugin para Javascritp que permitirá acceder al vibrador del dispositivo móvil, esto con el fin de que el usuario sepa cuando ha ingresado un nuevo carácter con el teclado.
c. HTML5: Se utilizara para definir la estructura del teclado.
d. CSS3: Se utilizara para diseñar el contenido de la interfaz del teclado.
e. JavaScritp: Se utilizara para realizar todos los procesos relacionados con la interpretación de las combinaciones que ingresa el usuario en relación al sistema Braille.

En segundo lugar, se encuentra el módulo de aprendizaje

Para este segundo módulo se contara con una serie de interfaces para el aprendizaje del sistema Braille.

- Para el usuario común se desarrollan actividades de entretenimiento como juegos y trivias para la práctica del conocimiento del sistema Braille.
- Se desarrolla manuales de usuario en PDF y mediante voz para ambos usuarios.
- Se desarrolla teclados de prueba simulando el sistema Braille para el usuario común.

En principio se visualizara un menú que contara con la siguiente estructura

1. Teoría

1.1 La interfaz teoría que ira de la mano entre la teoría y la adaptación contara con:

1.1.1 Interfaz de información del sistema Braille que contara con un menú que re direccionara hacia los diferentes ítems que se manejan para la enseñanza del sistema Braille que serán los siguientes:

1.1.1.1. Letras
1.1.1.2. Letras Acentuadas
1.1.1.3. Signos de puntuación.
1.1.1.4. Números.
1.1.1.5. Exploración del teclado: Esta opción contara con el teclado homologado con el sistema Braille que tomara la función de adaptación para el usuario a asimilarlo, contara con las funciones del teclado del primer módulo.

De cada una de las opciones del menú teoría contendrá toda la información en un scrollbar además de un menú desplegable con los temas a tratar que lo re direccionara a cualquiera de estos o al menú.

2. ACTIVIDADES

2.1 La interfaz de actividades estará dividida por temas de aprendizaje de la siguiente forma:

2.1.1 LETRAS: La interfaz contara en esencia con el teclado del primer módulo, pero con la función de que se homologue el vocabulario la letra que el sistema dice mediante el sintetizador de voz, si se hace de forma correcta se pasara al siguiente, de lo contrario se le hara saber al usuario y se repetirá el procedimiento.

2.1.2.2 PRACTICA NUMERICA: La interfaz le solicitara al usuario que ingrese un resultado de una operación matemática.

Las anteriores actividades contaran con una opción que los re direccionaran a la opción de actividades.

3. ESCRITURA: Esta opción re direccionara al usuario al primer módulo.

Las herramientas que se utilizaran para el segundo módulo serán las siguientes

1. SQLite: Se utilizara este gestor de bases de datos para almacenar cada uno de los datos de la sintaxis del sistema Braille, además de sonidos e imágenes.
2. HTML5: Se utilizara para definir la estructura de cada una de las piezas de la interfaz gráfica.
3. CSS3: Se utilizara para diseñar el contenido de la interfaz gráfica.
4. JavaScritp: Se utilizara para realizar todos los procesos relacionados con:
    a. Las consultas a la base de datos.
    b. La configuración de hilos para el audio e imágenes.

En tercer lugar, se encuentra el módulo de evaluación

- Se desarrolla test en los que se evalúan los conocimientos adquiridos.