

UNIVERSIDAD TECNOLÓGICA ISRAEL



CARRERA DE ELECTRÓNICA Y TELECOMUNICACIONES

**ESTUDIO, DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DE
ESCRITURA BRAILLE PARA ENVIAR SMS VÍA TELÉFONO CELULAR.**

**TRABAJO DE GRADUACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES**

ESTUDIANTE: DANIEL ANDRÉS NOBOA GALLEGOS

TUTOR: ING. MAURICIO ALMINATI V.

Quito D. M., Noviembre de 2013

UNIVERSIDAD TECNOLÓGICA ISRAEL

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Graduación certifico:

Que el trabajo de graduación **“ESTUDIO, DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DE ESCRITURA BRAILLE PARA ENVIAR SMS VÍA TELÉFONO CELULAR”**, presentado por NOBOA GALLEGOS DANIEL ANDRÉS, estudiante de la Carrera de Electrónica y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito D. M., Noviembre de 2013

TUTOR

Ing. Mauricio Alminati V.

UNIVERSIDAD TECNOLÓGICA ISRAEL**AUTORÍA DE TESIS**

El abajo firmante, en calidad de estudiante de la Carrera de Electrónica y Telecomunicaciones, declaro que los contenidos de este Trabajo de Graduación, requisito previo a la obtención del Grado de Ingeniería en Electrónica y Telecomunicaciones, son absolutamente originales, auténticos y de exclusiva responsabilidad legal y académica del autor.

Quito D.M., Noviembre de 2013

DANIEL ANDRÉS NOBOA GALLEGOS
CC: 0603784349

UNIVERSIDAD TECNOLÓGICA ISRAEL

APROBACIÓN DEL TRIBUNAL DE GRADO

Los miembros del Tribunal de Grado, aprueban la tesis de graduación de acuerdo con las disposiciones reglamentarias emitidas por la Universidad Tecnológica Israel para títulos de pregrado.

Quito D.M., Noviembre de 2013

Para constancia firman:

TRIBUNAL DE GRADO

PRESIDENTE

MIEMBRO 1

MIEMBRO 2

AGRADECIMIENTO

Agradezco a Dios por haberme guiado en escoger y culminar mis estudios en la Carrera de Electrónica y Telecomunicaciones y permitirme llegar a este punto tan importante de mi vida. Además, agradezco a mis padres y a mi hermano que han estado junto a mi en todo momento, con su cariño y dándome fuerzas para conseguir este objetivo tan anhelado.

DEDICATORIA

Va dedicado a mi padre, mi madre mi hermano, a mi novia y a las personas que han estado cerca y que me han ayudado a superarme cada día, además a los profesores Electrónica de la UISRAEL que con su apoyo y pasión por la carrera han sabido guiarme a lo largo de mis estudios. Además, dedicada a personas que estén comprometidas en desarrollar nueva tecnología dirigida a personas con capacidades especiales.

ÍNDICE GENERAL

CAPÍTULO 1 PROBLEMATIZACIÓN.....	1.
1.1 Antecedentes.....	1.
1.2 Problema Investigado.....	3.
1.3 Problema Principal.....	4.
1.4 Problemas Secundarios.....	5.
1.5 Justificación.....	5.
1.6 Objetivos.....	6.
1.6.1 Objetivo Principal.....	6.
1.6.2 Objetivos Específicos.....	6.
1.7 Metodología.....	7.
CAPÍTULO 2 MARCO TEÓRICO.....	8.
Introducción.....	8.
2.1 Deficiencia Visual.....	8.
2.1.1 Clasificación de la deficiencia visual.....	8.
2.1.2 Causas de la deficiencia visual.....	9.
2.2 Programación de aplicaciones para dispositivos móviles.....	9.
2.3 Teclado Braille.....	10.
2.4 Universal Serial Bus (USB).....	11.

CAPÍTULO 3 ESTUDIO, DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO DE SISTEMA DE ESCRITURA BRAILLE PARA ENVIAR SMS VÍA TELÉFONO CELULAR.....	13.
Introducción.....	13.
3.1. Estudio de las características de los dispositivos que serán utilizados para la correcta funcionalidad del sistema al momento de enviar los mensajes de texto.....	13.
3.1.1. Teclado Braille Portset.....	14.
3.1.1.1. Características técnicas teclado Braille Portset.....	15.
3.1.2. USB (On-The-Go).....	15.
3.1.2.1. Características técnicas USB OTG.....	16.
3.1.3. Software de programación App Inventor.....	17.
3.1.3.1. Características de App Inventor.....	17.
3.1.4. Tabla comparativa para selección de Smartphone.....	18.
3.1.4.1. SONY Xperia Go.....	18.
3.2. Diseño del prototipo de sistema de escritura Braille para enviar SMS vía celular.....	20.
3.2.1. Bloque 1.....	21.
3.2.2. Bloque 2.....	21.

3.2.3. Bloque 3.....	22.
3.2.4. Diseño de software.....	23.
3.2.5. Descripción del software de programación.....	24.
3.2.5.1. Pasos para la instalación del software App Inventor.....	25.
3.2.5.2 descripción de los elementos del software App Inventor.....	28.
3.2.5.3. Editor de bloques de programación.....	31.
3.3. Montaje.....	32.
3.3.1. Bloques de programación.....	32.
3.3.1.1. Bloques de inicialización.....	32.
3.3.1.2. Bloque mostrar pantalla enviar SMS y seleccionar contactos.....	33.
3.3.1.3. Bloque mostrar pantalla ingresar y guardar contactos.....	33.
3.3.1.4. Bloque regresar a pantalla de inicio.....	34.
3.3.1.5. Bloque ver contactos y editar mensaje.....	35.
3.3.1.6. Bloque cancelar y mostrar pantalla de inicio.....	35.
3.3.1.7. Bloque Ingreso De Contactos.....	36.
3.3.1.8. Bloque de selección de contactos en la lista.....	37.
3.3.1.9. Bloque enviar mensaje.....	38.
3.3.1.10. Bloque realizar llamada.....	39.
3.3.2. Pruebas Iniciales.....	40.

3.3.2.1. Pruebas iniciales de la aplicación.....	40.
3.3.2.2. Pruebas iniciales del teclado Braille.....	46.
3.3.3 Pruebas de Verificación.....	49.
3.4. Implementación del prototipo de sistema de escritura Braille para enviar SMS por medio de teléfono celular, de acuerdo a las características del proyecto.....	52.
3.4.1. Procedimiento para descargar la aplicación.....	52.
3.4.2. Pasos para la instalación de la aplicación.....	53.
3.4.3. Instalación de Talkback.....	56.
3.4.4 Configuración de Smart Connect.....	57.
3.5 Validación del funcionamiento del sistema de escritura Braille para enviar SMS vía teléfono celular.....	58.
3.5.1. Pruebas de confirmación de implementación del sistema.....	58.
3.5.2. Pruebas de Funcionamiento.....	60.
CAPÍTULO 4 RESULTADOS Y COSTOS DEL PROYECTO.....	63.
Introducción.....	63.
4.1. Análisis de los Resultados.....	63.
4.2. Matriz FODA.....	66.
4.3. Costos del Proyecto.....	67.
CAPÍTULO 5 CONCLUSIONES Y RECOMENDACIONES.....	68.
5.1. Conclusiones.....	68.

5.2. Recomendaciones.....	69.
Bibliografía.....	70.
Anexos.....	72.

ÍNDICE DE TABLAS

Tabla 2.1 Descripción de pines.....	12.
Tabla 3.1 Características técnicas teclado Braille Portset.....	15.
Tabla 3.2 Características técnicas USB OTG.....	16.
Tabla 3.3: Tabla comparativa selección Smartphone.....	18.
Tabla 3.4: Especificaciones generales Xperia Go.....	20.
Tabla 3.5. Lista de chequeo de operatividad y conectividad.....	59.
Tabla 3.6. Pruebas de manejo del sistema.....	60.
Tabla 3.7. Pruebas de funcionamiento.....	61.
Tabla 4.1. Matriz FODA.....	66.
Tabla 4.2. Costos del proyecto.....	67.

ÍNDICE DE FIGURAS

Fig. 2.1. Java J2ME.....	10.
Fig. 2.2. Teclado Braille.....	11.
Fig. 2.3. Tipos de conectores USB.....	12.
Fig. 3.1. Teclado Braille Portset.....	14.
Fig. 3.2. USB OTG.....	15.
Fig. 3.3. Conexión USB OTG.....	16.
Fig. 3.4. SONY Xperia Go.....	18.
Fig. 3.5. Diagrama de bloques.....	20.
Fig. 3.6. Teclado Braille Portset.....	21.
Fig. 3.7. Cable USB On The Go.....	22.
Fig. 3.8. Conector USB Hembra.....	22.
Fig. 3.9. Conector micro USB.....	22.
Fig. 3.10. Celular Sony Xperia Go Advanced.....	23.
Fig. 3.11. Diagrama de flujo del software.....	24.
Fig. 3.12. Pantalla de inicio de App Inventor.....	25.
Fig. 3.13. Pantalla setup.....	26.
Fig. 3.14. Pantalla App Inventor para instalación Java.....	26.
Fig. 3.15. Pantalla instalación Java.....	27.
Fig. 3.16. Pantalla App Inventor y selección sistema operativo.....	27.

Fig. 3.17. Pantalla Gmail e ingreso a App Inventor.....	28.
Fig. 3.18. Pantalla App Inventor.....	28.
Fig. 3.19. Pantalla crear proyecto.....	29.
Fig. 3.20. Pantalla de diseño.....	29.
Fig. 3.21. Pantalla editor bloques.....	32.
Fig. 3.22. Bloques de inicialización.....	33.
Fig. 3.23. Bloque mostrar tabla ver contactos.....	33.
Fig. 3.24. Bloque mostrar tabla ingresar contactos.	34.
Fig. 3.25. Bloque botón regresar.....	34.
Fig. 3.26. Bloque botón ver contactos y editar SMS.....	35.
Fig. 3.27. Bloque botón cancelar.....	35.
Fig. 3.28. Bloque botón guardar contactos.....	37.
Fig. 3.29. Bloque selección de contacto.....	38.
Fig. 3.30. Bloque botón enviar SMS.....	39.
Fig. 3.31. Bloque botón Llamar.	40.
Fig. 3.32. Ícono de la aplicación.....	41.
Fig. 3.33. Pantalla de inicio de la aplicación.....	41.
Fig. 3.34. Pantalla agregar contactos.....	42.
Fig. 3.35. Contacto a agregar.....	42.
Fig. 3.36. Mensaje de advertencia.	43.

Fig. 3.37. Pantalla seleccionar contactos.....	43.
Fig. 3.38. Lista de contactos.	44.
Fig. 3.39. Pantalla contacto seleccionado.....	44.
Fig. 3.40. Advertencia de mensaje enviado.....	45.
Fig. 3.41. Subrutina carga de contactos.	46.
Fig. 3.42. Bloque final de inicialización.....	46.
Fig. 3.43. Conexión cable OTG.....	47.
Fig. 3.44. Cable OTG conectado al teléfono celular.....	47.
Fig. 3.45. Teclado USB reconocido.....	48.
Fig. 3.46. interactuando con el teclado Braille.	48.
Fig. 3.47. Teclado USB funcionando con la aplicación.....	48.
Fig. 3.48. Pantalla agregar contactos.....	49.
Fig. 3.49. Ingreso de datos de contacto.....	50.
Fig. 3.50. Contacto guardado.....	50.
Fig. 3.51. Lista de contactos.	51.
Fig. 3.52. Contacto seleccionado.....	51.
Fig. 3.53. Advertencia de mensaje enviado.	52.
Fig. 3.54. Página SkyDrive.	53.
Fig. 3.55. Instalador TesisFinal.apk.....	53.
Fig. 3.56. Carpeta de destino.....	54.

Fig. 3.57. Ventana de verificación.....	54.
Fig. 3.58. Ventana de instalación.	55.
Fig. 3.59. Ventana de finalización de instalación.....	55.
Fig. 3.60. Aplicación TalkBack en la Play Store.....	56.
Fig. 3.61. Activación de TalkBack.....	57.
Fig. 3.62. Utilidad Smart Connect.....	58.

RESUMEN

Este proyecto de grado es basado en el uso tecnología que permita a personas con deficiencia visual comunicarse mediante el envío de SMS vía teléfono celular permitiendo enlazar éste mediante una interfaz física a un teclado Braille y realizar la escritura, es por eso que este proyecto pretende a que se incursione mas en el uso de diferente tecnología para su implementación y el desarrollo de aplicaciones móviles destinadas a personas con capacidades especiales.

En el capítulo uno se muestra la problematización, se describen los antecedentes del proyecto, seguido por los objetivos propuestos y la metodología utilizada para la investigación del proyecto. El capítulo dos muestra el marco teórico donde se explican conceptos básicos necesarios para la elaboración del proyecto. En el capítulo 3 se describe paso a paso el estudio, diseño e implementación del sistema, seguido por las pruebas de funcionamiento realizadas para validar el sistema.

El capítulo 4 muestra el análisis de resultados del sistema, una matriz FODA y finalmente el costo total del proyecto implementado. En el capítulo 5 se muestra únicamente las conclusiones y recomendaciones después de haber implementado el sistema.

ABSTRACT

This project grade is based on using technology to help visually impaired people to communicate by sending SMS via cell phone by allowing it to bind a physical interface to a keypad and make writing Braille, is why this project aims to that incursione more on using different technologies for implementation and development of mobile applications aimed at people with special abilities.

In chapter one shows the problematization, describes the background of the project, followed by the objectives and the methodology used for the research project. Chapter two shows the theoretical framework that explains basic concepts necessary for the development of the project. Chapter 3 describes step by step the study, design and implementation of the system, followed by functional tests performed to validate the system.

Chapter 4 shows the results of the system analysis, SWOT matrix and finally the total cost of the project implemented. Chapter 5 shows only the conclusions and recommendations after implementing the system.

CAPÍTULO 1

PROBLEMATIZACIÓN

1.1. Antecedentes

Uno de los problemas sociales en el Ecuador es la atención a las personas con capacidades especiales que en años atrás ha sido de deficiente calidad y de baja cobertura.

Hace aproximadamente medio siglo se empieza con un progresivo ordenamiento de las organizaciones de personas con capacidades especiales mediante la iniciativa de padres de familia y algunas organizaciones privadas mediante criterios de beneficencia y caridad con lo cual se inicia la atención a sus asociados.

Entre las décadas del 40 al 60 se crearon varias escuelas de educación especial para personas discapacitadas a través de instituciones del Ministerio de Educación.

El sector privado también interviene en la atención a este sector poblacional como el Instituto Nacional del Niño y la Familia (INNFA), la creación de varios centros de rehabilitación y escuelas de educación especial. Entre algunas ONGs que se destacaron por su gran labor en beneficio de las personas con capacidades especiales son: CEBYCAM, OLIMPIADAS ESPECIALES, ASENIR, SERLI, FUNDACIÓN GENERAL ECUATORIANA, FASINARM, ADINEA, FUNDACIÓN HERMANO MIGUEL, FUNAPACE, entre otras.

Un cambio muy importante en la década pasada es el ordenamiento y racionalización de la atención a las personas con capacidades especiales que da como resultado la aplicación de la Ley 180 de Discapacidades de 1992 y la aparición del Consejo Nacional de Discapacidades (CONADIS), se dispone la creación de un Sistema Nacional de Prevención de Discapacidades y de atención e integración de las personas con capacidades especiales y donde al CONADIS tiene la facultad de dictar políticas, impulsar investigaciones y coordinar acciones.¹

Ahora la inclusión laboral de personas con capacidades es ley desde 2006, donde las empresas deben tener en su nómina por lo menos el 4 % de personas con esta condición, además el uso de la tecnología debe permitir incluir con mucho más énfasis a las personas con deficiencia visual, las cuales aprenden el alfabeto braille donde les permite leer y escribir en un teclado especial.

El sistema actual de lectura a través del tacto evolucionó desde un código de ocho puntos conocida también como "Lectura en la Oscuridad" creado por Charles Barbier para la Armada de Francia en el año 1825. Este sistema no solamente utilizó puntos en relieve sino también utilizó una línea horizontal que los ciegos pronto descubrieron que era más difícil de leer que el sistema por puntos en relieve enteramente compuestos.

El Sistema Braille se basa en seis puntos con alto relieve que permite leer a las personas no videntes mediante el sentido del tacto. Este sistema, que representa números, letras, notaciones musicales, etc, mediante la combinación de estos

¹ http://www.conadis.gob.ec/index.php?option=com_content&view=article&id=57%3Aantecedentes-historicos&catid=34%3Aentidad&Itemid=1, Febrero 2013

puntos ha ganado, a través de sus más de 170 años de historia, un reconocimiento a nivel mundial y ha tenido varios esfuerzos para su educación, incluso para defenderse frente a propuestas distintas que nunca se llegaron a concretar.

En países desarrollados, la Signografía Braille ha tenido que adecuarse para satisfacer una gran variedad de necesidades.²

1.2. Problema Investigado

En este mundo globalizado, la comunicación es indispensable y el uso de la tecnología debe permitir incluir con mucho más énfasis a las personas con deficiencia visual.

Hoy en día millones de mensajes se envían diariamente a través del servicio de mensajes cortos (SMS) disponibles en teléfonos móviles. Se trata de un servicio para enviar mensajes de texto en forma práctica, fácil y económica, gracias a esto se puede mandar mensajes cortos que no justifican una llamada telefónica.

Si una persona no vidente quisiera enviar un mensaje de texto vía celular, con la tecnología actual no podría hacerlo por su propia cuenta ya que no hay dispositivos que tengan esta funcionalidad y necesitaría de una persona que le ayude a enviar el mensaje y podría llegar a ser muy molesto para algunas personas proporcionar esta ayuda; además se perdería la privacidad y confiabilidad al momento de redactar el texto.

² [http://es.wikipedia.org/wiki/Braille_\(lectura\)](http://es.wikipedia.org/wiki/Braille_(lectura)), Febrero 2013

Las personas con discapacidad visual ya sea por una anomalía congénita o un accidente aprenden el alfabeto braille donde les permite leer y escribir en un teclado especial, pero un cierto número de personas con esta deficiencia aprenden a escribir en un teclado con el alfabeto del idioma castellano pero no les permite escribir con rapidez como lo podrían hacer con un teclado desarrollado para su necesidad.

Existen sistemas digitalizados de escritura en braille que son capaces de imprimir en la misma escritura; para transcribir de Braille a escritura en castellano, este sistema requiere una impresora común. Para el transcriptor de escritura convencional a Braille se requiere de un teclado QWERTY convencional. En el autoaprendizaje tiene una importante ayuda, como es el habla digitalizada, que permite verbalizar los nombres de cada una de las teclas, comandos, errores, el estado y conexión de la impresora.

Es una máquina liviana, de excelente calidad de impresión Braille en papel común. Otorga por lo tanto, una comunicación fluida y eficaz entre personas que tienen conocimiento del sistema de lectura-escritura Braille, con personas que no conocen este tipo de escritura y permitiendo a cualquier persona dar una revisión rápida de lo escrito ya sea en Braille o en escritura normal.

1.3. Problema Principal

Las personas con discapacidad visual no cuentan con un sistema que les permita escribir un mensaje de texto y que sea enviado mediante un teléfono celular.

1.4. Problemas Secundarios

- No existe un estudio sobre las características que debe tener este tipo de sistemas para enviar SMS.
- Se carece de un diseño que envíe mensajes de texto con las especificaciones requeridas.
- No se sabe que tecnología se va a utilizar para la implementación del sistema propuesto.
- Falta un procedimiento adecuado para verificar la funcionalidad del sistema.

1.5. Justificación

Este proyecto servirá para investigar los diferentes tipos de comunicaciones usados en celulares y poder implementar un sistema adecuado que además permita incluir el alfabeto braille y mejorar la investigación para este tipo de sistemas.

Este prototipo permitirá que las personas con discapacidad visual usen un sistema que les va a permitir escribir en un teclado especial y así poder comunicarse de una manera eficaz enviando mensajes de texto vía teléfono celular y con un costo relativamente bajo.

Con la elaboración del sistema se tratará de ayudar a las personas con discapacidad visual puedan enviar SMS sin ayuda de otras personas y tengan privacidad al momento de enviar el mensaje y sin errores.

Además es un tema de actualidad porque hoy en día la comunicación es

indispensable y el uso de la tecnología debe permitir incluir con mucho mas énfasis a las personas con deficiencia visual.

1.6. Objetivos

1.6.1. Objetivo Principal

Estudiar, Diseñar e Implementar un prototipo de Sistema de escritura braille para enviar SMS vía teléfono celular.

1.6.2. Objetivos Específicos

- Estudiar las características de los dispositivos para la correcta funcionalidad del sistema al momento de enviar los mensajes de texto.
- Diseñar un prototipo de Sistema de escritura braille para enviar SMS vía celular.
- Implementar el prototipo de Sistema de escritura braille para enviar SMS por medio de teléfono celular, de acuerdo a las características del proyecto.
- Validar el funcionamiento del sistema de escritura braille para enviar SMS vía teléfono celular.

1.7. Metodología

El proyecto se realizó mediante cuatro etapas de investigación, que se describen a continuación:

En la Primera etapa se usó los métodos de Análisis y Síntesis para la recopilación de los datos necesarios en la investigación del proyecto.

En la Segunda etapa se tomó en cuenta el método de Modelación que ayudó a realizar el diseño del proyecto que se basó en la realidad.

En la Tercera y Cuarta etapa se utilizó el método experimental que permitió verificar el comportamiento del proyecto que permitió ponerlo en marcha.

CAPÍTULO 2

MARCO TEÓRICO

Introducción

En este capítulo se explican conceptos relevantes en los cuales se basará para realizar el diseño del sistema de escritura braille, además se muestran varios gráficos que ayudarán en el desarrollo del proyecto.

2.1. Deficiencia Visual

La deficiencia visual se refiere a las condiciones que las personas presentan, estas pueden ser pérdida parcial de la visión, que se refiere a personas que tiene una pequeña percepción de la luz, pudiendo así distinguir la luz y la oscuridad mas no distinguir formas de los objetos y la pérdida total de la visión es donde las personas no ven nada.³

2.1.1. Clasificación de la deficiencia visual

Se puede clasificar desde la ceguera hasta una deficiencia visual mínima y se dividen en:

- **Visión Parcial:** Cuando la percepción de los objetos se dificulta ya sea por falla de uno o ambos ojos, para corregir esta dificultad se requiere el uso de lentes.

³ <http://www.once.es/new/servicios-especializados-en-discapacidad-visual/discapacidad-visual-aspectos-generales/concepto-de-ceguera-y-deficiencia-visual>, Agosto 2013.

- **Visión Escasa:** Cuando la persona solo puede percibir objetos a pocos centímetros.
- **Ceguera parcial:** La persona solo puede captar la luz y los objetos sin formas observados como bultos.
- **Ceguera:** Las personas que no perciben nada o apenas pueden distinguir algo de luz.

2.1.2 Causas de la deficiencia visual

La deficiencia visual puede estar dada por diferentes causas, se describen a continuación las más importantes.

- Hereditarias.
- Congénitas.
- Adquiridas Accidentalmente.
- Víricas.⁴

2.2. Programación de aplicaciones para dispositivos móviles

Actualmente los teléfonos celulares tienen la capacidad de ejecutar aplicaciones de diferentes desarrolladores, estas aplicaciones son creadas en lenguaje de programación JAVA desarrollada por SUN Microsystems que es basada en programación orientada a objetos.

Debido a la limitación de Hardware de los teléfonos móviles se optimizó una versión de JAVA llamada J2ME (JAVA 2 MICRO EDITION), éste estándar es limitado solo para teléfonos móviles y da soporte de desarrollo que permite a la aplicación creada

⁴ <http://www.psicopedagogia.com/articulos/?articulo=459>, Agosto 2013.

controlar distintas funciones del Smartphone como: GPS, Bluetooth, Cámara entre otras.

Para que los teléfonos celulares sean capaces de abrir las aplicaciones deben cumplir con un estándar desarrollado por SUN Microsystems (Fig. 2.1).



Fig. 2.1. Java J2ME.⁵

2.3 Teclado Braille

Es un dispositivo electrónico de entrada que consta de 6 u 8 teclas principales y algunas auxiliares, donde las teclas principales permiten el ingreso del código braille y las auxiliares permiten el ingreso de varios comandos requeridos dependiendo del teclado utilizado.⁶

Gracias al avance de la tecnología estos dispositivos se pueden conectar mediante USB o Bluetooth permitiendo la conexión con dispositivos móviles e interactuar con ellos.

⁵ <http://www2.elo.utfsm.cl/~iwg101/ClaseCelulares.pdf>, Agosto.

⁶ <http://accesibilidadweb.dlsi.ua.es/?menu=tec-braille>, Agosto 2013.

El funcionamiento de este tipo de teclado es similar al de un teclado convencional, las teclas que se usan son un mecanismo electro-mecánico que permite la detección de la pulsación, para mostrar un caracter se deben presionar las teclas correspondientes para generar el caracter requerido (Fig. 2.2).⁷



Fig. 2.2. Teclado Braille.

2.4 Universal Serial Bus (USB)

Es un estándar desarrollado a mediados de los noventa donde define cables, conectores y protocolos utilizados en un bus para conectar, comunicar y proveer de energía eléctrica entre ordenadores y dispositivos electrónicos.

Fue diseñado para estandarizar la conexión de diferentes dispositivos como por ejemplo mouse, teclados, teléfonos móviles, impresoras entre otros, algunos dispositivos necesitan una mínima potencia para funcionar, por esto se pueden conectar varios periféricos sin hacer uso de una fuente de alimentación externa para su funcionamiento, para esto existen concentradores también llamados USB Hubs.

También hay varios dispositivos que consumen una energía mayor que la que proporciona un conector USB y requieren de una fuente de alimentación propia, para

⁷ http://es.wikipedia.org/wiki/Teclado_Braille, Agosto 2013.

esto existen Hubs USB con fuente de alimentación que pueden proporcionar corriente eléctrica a los dispositivos sin quitar corriente al resto de los dispositivos.⁸

A continuación se muestra los tipos de conectores USB que existen (ver Fig. 2.3) y la descripción de cada uno de sus pines (ver Tabla 2.1)

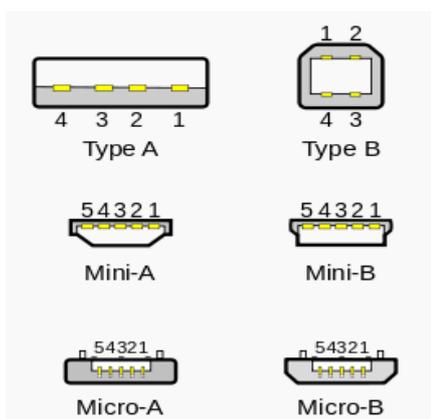


Fig. 2.3. Tipos de conectores USB⁹

Pin	Nombre	Color	Descripción
1	VCC	Rojo	+5 V
2	D-	Blanco	Data -
3	D+	Verde	Data +
			Permite la distinción de Micro-A y Micro-B
4	ID	Ninguno	Tipo A: conectado a masa Tipo B: no conectado
5	GND	Negro	Masa y retorno o negativo

Tabla 2.1. Descripción de pines.¹⁰

⁸ http://es.wikipedia.org/wiki/Universal_Serial_Bus, Agosto 2013.

⁹ http://commons.wikimedia.org/wiki/File:Types-usb_new.svg, Agosto 2013.

¹⁰ http://es.wikipedia.org/wiki/Universal_Serial_Bus, Agosto 2013.

CAPÍTULO 3

ESTUDIO, DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO DE SISTEMA DE ESCRITURA BRAILLE PARA ENVIAR SMS VÍA TELÉFONO CELULAR.

Introducción

En este capítulo se explica el estudio, el diseño del sistema mediante diagramas de bloques, así como el diseño de software planteado mediante bloques de programación realizados en AppInventor, lo que permite la implementación y el montaje del sistema completo, con varias explicaciones necesarias.

3.1. Estudio de las características de los dispositivos que serán utilizados para la correcta funcionalidad del sistema al momento de enviar los mensajes de texto.

Para que el sistema propuesto trabaje correctamente, se necesita cumplir con los siguientes requerimientos:

- Se necesita de un teclado Braille que sea compatible con conexión USB 2.0.
- Se necesita de una interfaz que permita conectar el teclado Braille al teléfono celular.
- Se requiere de una aplicación para interactuar entre el teléfono y el teclado y permitir el envío de mensajes de texto.
- Se requiere de un teléfono celular que pueda interconectarse con el teclado Braille.

3.1.1. Teclado Braille Portset

Este teclado es fabricado por Portset en Inglaterra, es un teclado Braille electrónico de 6 puntos y tiene varias teclas auxiliares como Ctrl, Alt, Backspace, Enter, Barra espaciadora y cursores de selección que permite ingresar simbología adicional presionando conjuntamente las teclas de escritura, posee una conexión USB que no necesita un controlador adicional para que sea detectado por una PC independientemente de su sistema operativo. Sólo se requiere de un puerto USB 2.0 para su conexión.

Permite a personas con deficiencia visual la escritura Braille mediante combinaciones de teclas para formar el caracter requerido por el usuario (Fig.3.1).



Fig. 3.1. Teclado Braille Portset.

3.1.1.1. Características técnicas teclado Braille Portset

Conector	
Conector	USB 2.0
Tipo	Macho
Características Físicas	
Longitud del Cable USB	2,5 m
Ancho	220 mm
Profundidad	111 mm
Altura Delantera	20 mm
Altura Trasera	40 mm

Tabla 3.1. Características técnicas teclado Braille Portset.¹¹

3.1.2. USB (On-The-Go)

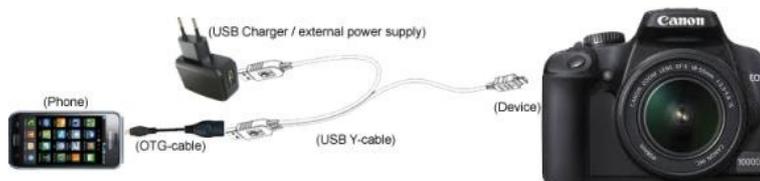
Conocido también como USB OTG (ver Fig. 3.2), permite a dispositivos electrónicos como teléfonos celular, tabletas, cámaras digitales, entre otros actuar como servidores, permitiendo así interactuar con dispositivos como teclados, ratones, memorias USB y otros (ver Fig. 3.3).



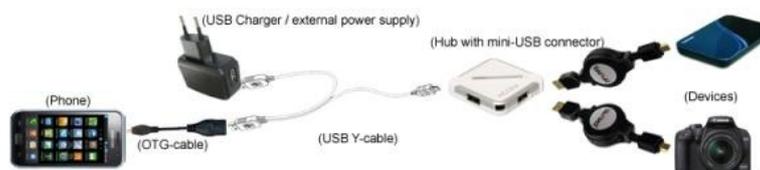
Fig. 3.2. USB OTG.

¹¹ <http://www.portset.co.uk/braille-keyboards/>, Agosto 2013.

How to connect to devices which have mini-USB connectors (external hard drives, DSLRs, Windows Mobile 6 phones, etc.):



How to connect through a hub which has a mini-USB connector:



How to connect through a hub which has an external power connector:



Fig. 3.3. Conexión USB OTG.

3.1.2.1. Características técnicas USB OTG

Conectores	
Conector A	1 - Micro USB-B
Conector B	1 – USB-A
Tipo de Conector A	Macho
Tipo de Conector B	Hembra
Número de Pines Conector A	5 Pines
Número de Pines Conector B	4 Pines
Características Físicas	
Longitud del Cable	127 mm
Peso del Cable	6.6 gr
Color	Negro
Calibre del Conductor	28 AWG
Revestimiento del Conductor	Níquel

Tabla 3.2. Características técnicas USB OTG.¹²

¹² <http://mx.startech.com/Cables-Adaptadores/USB-2.0/Adaptadores-USB/Adaptador-12cm-Micro-USB-a-USB-OTG-Host-macho-a-hembra~UUSBOTG>, Agosto 2013.

3.1.3. Software de programación App Inventor

Es una aplicación desarrollada en su inicio por Google, hoy en día esta plataforma es administrada por el MIT (Massachusetts Institute of Technology), esta herramienta permite realizar programas en Android con un conjunto de herramientas básicas y con un conocimiento de programación básico, donde el usuario debe ir enlazando bloques de programación para completar la programación, aunque es una herramienta muy simple cubre un gran número de necesidades requeridas que se pueden llegar a cumplir con este software y esta disponible tanto para Windows, Linux y Mac OS X.

3.1.3.1. Características de App Inventor

Para la edición visual de bloques utiliza una librería llamada Open Blocks de Java, dichas librerías son distribuidas por el MIT, bajo licencia libre (MIT License).

El compilador que usa para traducir la programación visual por bloques para la aplicación final de Android es KAWA que no es nada mas que un lenguaje de programación que forma parte y es distribuido por el sistema operativo GNU (GNU is Not Unix) por parte del movimiento y comunidad de Software y Conocimiento Libres.¹³

¹³ http://es.wikipedia.org/wiki/App_Inventor, Agosto 2013.

3.1.4. Tabla comparativa para selección de Smartphone

	Sony Xperia Go ¹⁴	Samsung Galaxy S3 ¹⁵	Samsung Galaxy S4 ¹⁶
Sistema Operativo	Google Android 4.0 (Ice Cream Sandwich) Actualizable	Google Android 4.0.4 (Ice Cream Sandwich) Actualizable	Google Android 4.2.2 (Jelly Bean) Actualizable
Procesador	Dual-Core 1 GHz Cortex-A9	Quad-Core 1.4 GHz Cortex-A9	Quad-Core 1.6 GHz Cortex-A15 & Quad-core 1.2 GHz Cortex-A7
RAM	512 MB	1 GB	2 GB
USB	microUSB v2.0 USB On-the-go	microUSB v2.0 USB On-the-go	microUSB v2.0 USB On-the-go
Existe en el País	Sí	Sí	Sí
VALOR (\$)	300	500	800

Tabla 3.3. Tabla comparativa selección Smartphone.

3.1.4.1. SONY Xperia Go

Smartphone fabricado por SONY en Julio de 2012 (Fig. 3.4).



Fig. 3.4. SONY Xperia Go

¹⁴ http://www.gsmarena.com/sony_xperia_go-4782.php, Junio 2013.

¹⁵ http://www.gsmarena.com/samsung_i9300_galaxy_s_iii-4238.php, Junio 2013.

¹⁶ http://www.gsmarena.com/samsung_i9500_galaxy_s4-5125.php, Junio 2013.

A continuación se muestran las características técnicas y las especificaciones generales (ver Tabla. 3.4) del Smartphone:

- Compatibilidad con USB 2.0 de alta velocidad, Micro USB y On-The-Go.
- Compatible con redes UMTS HSPA 850, 1900, 2100.
- Memoria interna del teléfono: 8 GB (hasta 4 GB de memoria disponible para el usuario).
- Memoria flash: 8 GB de eMMC.
- Ranura de expansión: microSD™, hasta 32 GB.
- Funciones Wi-Fi y de punto de conexión Wi-Fi.
- Sincronización a través de Exchange ActiveSync y Google Sync
- GPS
- Navegador web WebKit con desplazamiento y zoom.
- Tecnología Bluetooth.
- Vinculación USB nativa.
- Eliminación de ruido.
- Reconocimiento de escritura manual.
- Widget de escribir y enviar.

Especificaciones Generales	
Peso	110 gr
Dimensiones	111 x 60.3 x 9.8 mm
Tiempo de Conversación (máx.)	6 horas y 30 minutos
Tiempo en espera (máx.)	520 horas
Tiempo en reproducción de música	45 horas
Tiempo en reproducción de video	6 horas
Pantalla	Pantalla TFT táctil de 3,5 pulgadas resistente a arañazos 16 millones de colores y 480 x 320 píxeles
Sistema Operativo	Google Android 4.0 (Ice Cream Sandwich)
Procesador	Cortex A9 de doble núcleo NovaThor™ U8500 de 1 GHz
RAM	512 MB
Cámara	Cámara de 5 megapíxeles con función de enfoque automático Zoom digital de 16 aumentos y flash LED
Resistencia	Resistencia al polvo y agua

Tabla 3.4. Especificaciones generales Xperia Go.¹⁷

3.2. Diseño del prototipo de sistema de escritura Braille para enviar SMS vía celular.

La Fig. 3.5. muestra el diagrama de bloques del sistema propuesto.

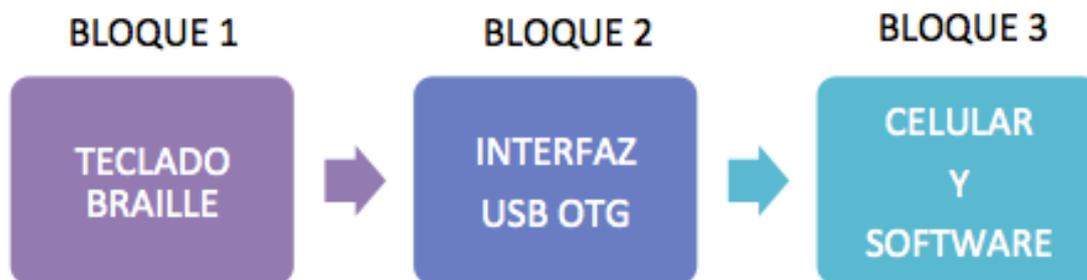


Fig. 3.5. Diagrama de bloques.

¹⁷ <http://www.sonymobile.com/es/products/phones/xperia-go/specifications/#yellow>, Agosto 2013.

A continuación se describe el funcionamiento del sistema bloque por bloque según el diagrama de bloques de la Fig. 3.5.

3.2.1. Bloque 1

Este bloque consta de un teclado especial de marca Portset, es un teclado que tiene 6 botones para realizar la escritura en Braille, además tiene varias teclas al igual que teclas convencionales como: Control, Alt, Enter, Backspace, Barra Espaciadora, y teclas de movimiento del cursor. Este teclado se conecta vía USB y la finalidad de este bloque es permitir escribir e interactuar con el Bloque 3 (Fig. 3.6).



Fig. 3.6. Teclado Braille Portset.

3.2.2. Bloque 2

Este bloque consta de la interfaz que permite conectar el Bloque 1 con el Bloque 3, es un cable USB OTG (On-The-Go) (ver Fig. 3.7), que va permitir que el teléfono celular alimente al teclado y pueda funcionar sin necesidad de una fuente externa. Consta en un extremo de un conector Micro USB Macho (ver Fig. 3.9), donde éste se conecta en el puerto Micro USB Hembra del

teléfono celular, en el otro extremo posee un conector USB Hembra (ver Fig. 3.8) que va ser conectado el teclado Braille.



Fig. 3.7. Cable USB On The Go.



Fig. 3.8. Conector USB Hembra.



Fig. 3.9. Conector micro USB.

3.2.3. Bloque 3

Este bloque consta del teléfono celular Sony Xperia Go Advanced (ver Fig. 3.10) que soporta la tecnología USB OTG, posee un Sistema Operativo Android 4.0.1. Este teléfono permite la conexión de diferentes dispositivos USB e interactuar con ellos, además este bloque consta del software diseñado que permite interactuar con el teclado Braille permitiendo almacenar

y seleccionar contactos para poder enviar mensajes de texto o realizar una llamada.



Fig. 3.10. Celular Sony Xperia Go Advanced.

3.2.4. Diseño De Software

El diseño de software se muestra mediante el siguiente diagrama de flujo (ver Fig. 3.11), donde se explica detalladamente el funcionamiento que tendrá el programa, en base a este diagrama se procederá a crear la aplicación que será ejecutada y los comandos serán entendidos por el Sistema Operativo Android.

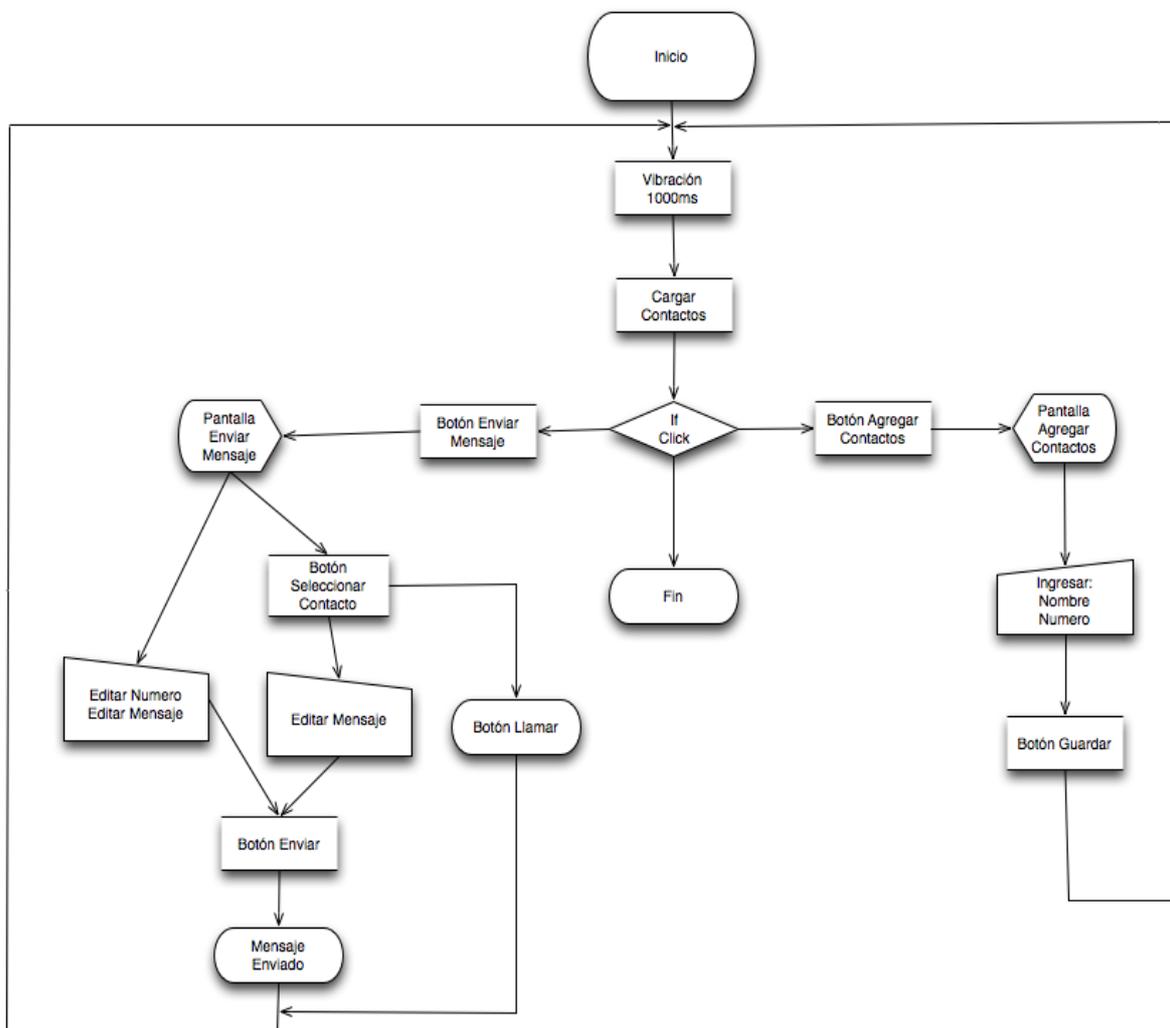


Fig. 3.11. Diagrama de flujo del software.

3.2.5. Descripción Del Software De Programación

Para la programación de la aplicación en Android se utilizará el Software App Inventor administrado por el MIT (Massachusetts Institute of Technology), a continuación se detallan los pasos de la instalación y los elementos del software para proceder con la programación.

3.2.5.1. Pasos para la instalación del software App Inventor

Primer Paso: Abrir página web de App Inventor

Para abrir la aplicación se necesita una conexión a internet y acceder mediante el navegador a la página

“<http://appinventor.mit.edu/explore/>” (Fig. 3.12).

The screenshot shows the MIT App Inventor website homepage. At the top is a navigation menu with tabs: Home, News+Events, Tutorials, Forums, Resources, Support, Stories, Setup, Teach, and Invent. Below the menu is the MIT App Inventor logo. On the right side, there is a large orange button that says "Invent YOUR OWN APPS NOW". Below this button, it says "All you need to get started with MIT App Inventor is a Google account to log in." The main content area is divided into several sections: "Why App Inventor?" which includes a video player and text explaining the process; "Success with MIT App Inventor" which features two articles with images and titles: "Bunny Bolt: Location-sensing outdoor fitness game" and "Plant Data Collector: a field biologist's personal assistant"; and "From the Blog" which lists recent blog posts with links. At the bottom right, there is a search bar and a section titled "App Inventor Code is Open Source" with a link to the open source code.

Fig. 3.12. Pantalla de inicio de App Inventor.

Segundo Paso: Instalar App Inventor

Para que la aplicación pueda correr adecuadamente debe ser instalada, para eso se debe dar clic en la pestaña Setup (Fig. 3.13).

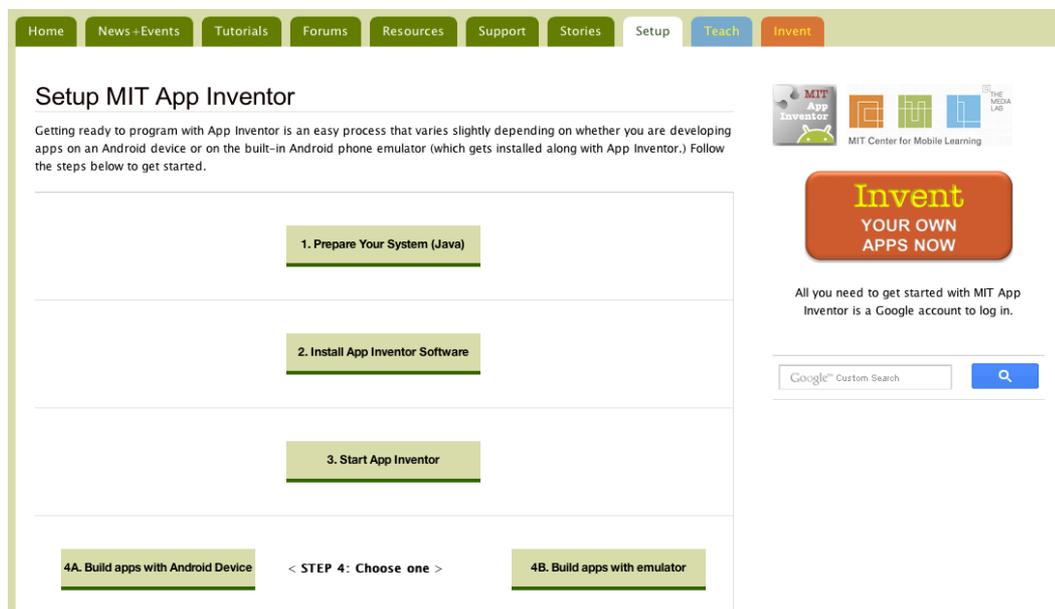


Fig. 3.13. Pantalla Setup.

Tercer Paso: Instalar Java

Para la instalación de Java y se debe dar clic en la opción Prepare Your System (Java), y se procede a dar clic en la opción Java Installation (Fig. 3.14), se abrirá la página de JAVA se selecciona el sistema operativo instalado y se siguen los pasos para la instalación (Fig. 3.15).

1. Prepare Your System (Java)

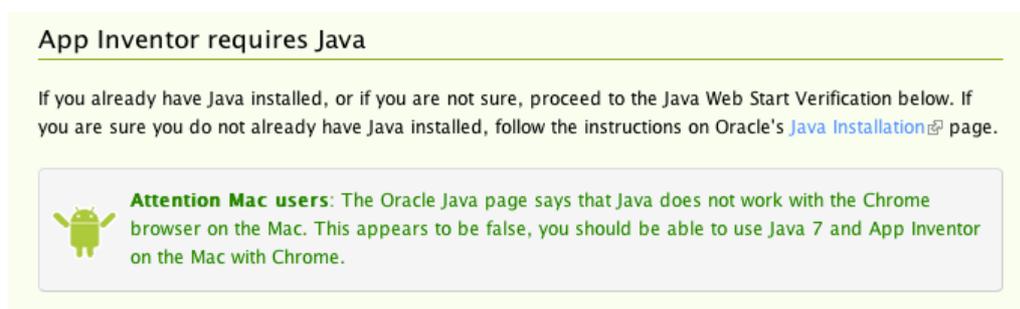


Fig. 3.14. Pantalla App Inventor para instalación Java.

The screenshot shows the Java website's help resources page. At the top, there is a red header with the Java logo on the left, a search bar on the right, and the words "Download" and "Help" in the center. Below the header, there is a "HELP RESOURCES" section on the left with a list of links: "Installing Java", "Disable Java", "Using Java", "General Questions", "Mobile Java", "Security", and "Support Options". To the right of this section is the main content area titled "How do I install Java ?" with a "Printable Version" link. The main content area contains the text "Choose the Operating System for instructions to install Java:" followed by a list of links: "Windows", "Mac OS X", "Linux", and "Solaris".

Fig. 3.15. Pantalla instalación Java.

Cuarto Paso: Instalar App Inventor Software

Una vez instalado Java, en la pestaña de Setup se da clic en la opción Install App Inventor Software y se selecciona el sistema operativo y se siguen los pasos para la instalación. (Fig. 3.16).

The screenshot shows the MIT App Inventor Setup page. At the top, there is a navigation bar with tabs: "Home", "News + Events", "Tutorials", "Forums", "Resources", "Support", "Stories", "Setup", "Teach", and "Invent". The "Setup" tab is selected. Below the navigation bar, the page title is "Setup" and the main heading is "2. Install App Inventor Software". Below this, it says "Part 2 of 4 Setup Instructions". The main content area is titled "Install the App Inventor Setup Software" and contains the text: "Before you can use App Inventor, you need to install some software on your computer. The software you need is provided in a package called *App Inventor Setup*. Follow the instructions for your operating system to do the installation, and then come back to this page to move on to Step 3 (starting App Inventor) and Step 4 (setting up your device or emulator)." Below this text are three links: "Instructions for Mac OS X", "Instructions for GNU/Linux", and "Instructions for Windows". On the right side of the page, there is a logo for "MIT App Inventor" and "MIT Center for Mobile Learning". Below the logo is a large orange button that says "Invent YOUR OWN APPS NOW". Below the button, it says "All you need to get started with MIT App Inventor is a Google account to log in." At the bottom right, there is a search bar with the text "Google Custom Search" and a search icon.

Fig. 3.16. Pantalla App Inventor y selección sistema operativo.

Quinto Paso: Abrir App Inventor

Una vez realizados los pasos anteriores en el explorador se ingresa a la página <http://beta.appinventor.mit.edu>. Para ingresar se necesita una cuenta de Gmail (Fig. 3.17).

Cuentas

MIT AppInventor Experimental utiliza las Cuentas de Google para el inicio de sesión.

Google no se hace responsable de los contenidos de MIT AppInventor Experimental ni está asociado con sus propietarios. Si inicias sesión, Google compartirá tu dirección de correo electrónico con MIT AppInventor Experimental, pero no compartirá tu contraseña ni ningún otro dato personal.

MIT AppInventor Experimental puede utilizar tu dirección de correo electrónico para adaptar su sitio web a tus preferencias.

Iniciar sesión Google

Dirección de correo electrónico

Contraseña

No cerrar sesión

[¿No puedes acceder a tu cuenta?](#)

Fig. 3.17. Pantalla Gmail e ingreso a App Inventor.

Sexto Paso: Inicio de App Inventor

Para poder usar el software cuando se requiera lo único que se debe hacer es ingresar a la página web “ <http://beta.appinventor.mit.edu> ”, y se inicia sesión como se indicó en el quinto paso y se verá una pantalla como la de la Fig. 3.18.

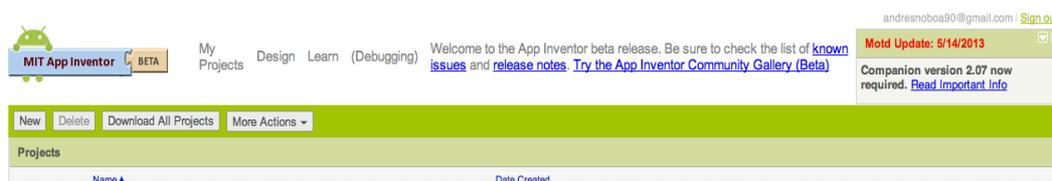


Fig. 3.18. Pantalla App Inventor.

3.2.5.2. Descripción de los elementos del software App Inventor.

Se detalla las partes que contiene el software, para iniciar se da clic en la opción New como se muestra en la Fig. 3.18, se ingresa el nombre del proyecto y se presiona Ok (Fig. 3.19).

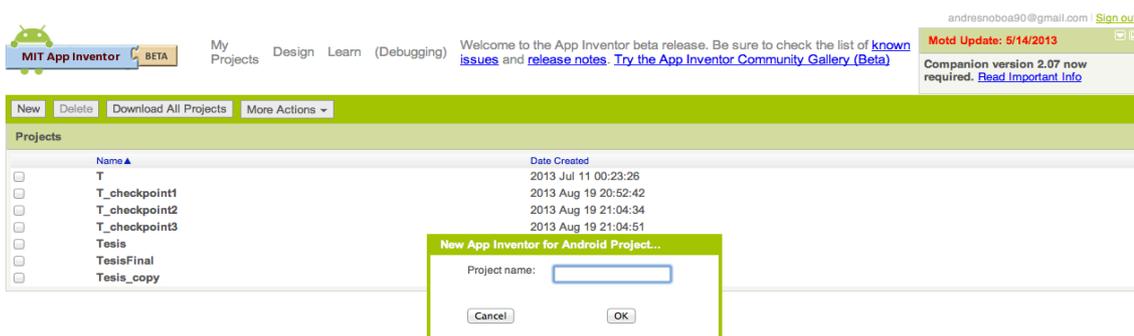


Fig. 3.19. Pantalla crear proyecto.

Una vez creado el proyecto se abrirá la pagina para el desarrollo de la interfaz gráfica, donde está dividido por bloques para su pronta explicación (Fig. 3.20).



Fig. 3.20. Pantalla de diseño.

Parte 1: Paleta

En este bloque se encuentran elementos que se pueden poner en la aplicación como botones de acción, cuadros para ingresar texto, labels, etc, son elementos gráficos que pueden ser visibles o no visibles en la aplicación dependiendo del que sea utilizado para poder realizar la programación (Fig. 3.20).

Parte 2: Visor

Aquí es donde se van a mostrar los elementos que sean arrastrados de la Parte 1 si estos son visibles y si no lo son se mostrarán en la parte de abajo (Fig. 3.20).

Parte 3: Lista de Componentes

Aquí se muestra la lista de los elementos que están ya en el visor de la Parte 2, desde aquí se puede cambiar el nombre o borrar los elementos que se requieran (Fig. 3.20).

Parte 4: Propiedades de los Componentes

Cuando se selecciona un elemento en el visor en este bloque se van a mostrar las propiedades de ese elemento que permitirá cambiar el tipo fuente, color y varias propiedades dependiendo del elemento seleccionado (Fig. 3.20).

Parte 5: Archivos Multimedia

En este bloque se muestran diferentes archivos multimedia como imágenes, sonidos que están disponibles para poder usarlos en la aplicación, estos elementos deben ser previamente cargados por el usuario únicamente dando clic en Upload New y seleccionando el archivo a cargar (Fig. 3.20).

3.2.5.3. Editor de bloques de programación

Para abrir el editor se procede a dar clic en la opción Open the Blocks Editor se descargará un archivo con extensión .jnlp, se lo abre y se dan los permisos que indique Java para abrir el editor y se muestra una ventana como de la Fig. 3.21.

El editor de bloques permite interactuar con los elementos seleccionados en la pantalla de diseño, únicamente arrastrando los bloques del menú del lado izquierdo hacia la pantalla de edición.

El menú está constituido por 3 secciones que se describen a continuación:

1. Built-In

Contiene los bloques de programación divididos en diferentes partes como Definition, Text, Lists, Math, Logic, Control y Colors, cada uno contiene bloques básicos para la programación.

2. My Blocks

Contiene la lista de los elementos que se seleccionaron en el visor, aquí se seleccionan dichos elementos para realizar la programación.

3. Advanced

En esta sección se puede acceder en forma global en elementos que son comunes, cambia algunas propiedades simultáneamente del elemento seleccionado.



Fig. 3.21. Pantalla editor bloques.

3.3. Montaje

A continuación se muestran las pruebas iniciales tanto del funcionamiento de los equipos y de la aplicación con la explicación de los bloques de programación, esto seguido por las pruebas finales del sistema para comprobar el funcionamiento correcto.

3.3.1. Bloques de programación

Se muestran los bloques de programación que se utilizaron para satisfacer el diseño requerido en la aplicación y su descripción.

3.3.1.1. Bloques de inicialización

Se define la variable Contactos para que pueda almacenar una lista.

Con el bloque Screen1.initialize se inicializa una vibración de 1000 ms para indicar que la aplicación se ha abierto (Fig. 3.22).

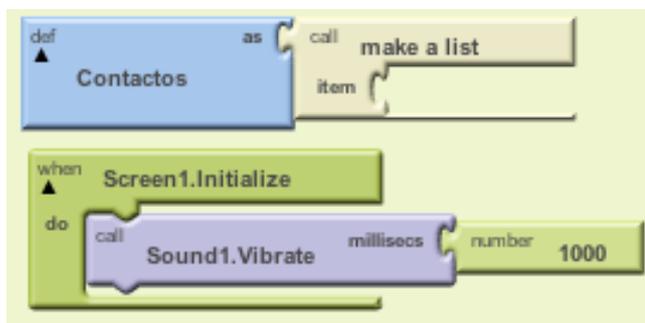


Fig. 3.22. Bloques de inicialización.

3.3.1.2. Bloque mostrar Pantalla Enviar SMS y seleccionar contactos

El bloque *ButtonEnviarSMSTinicio.Click* al momento de hacer clic sobre el botón Enviar SMS se muestra la tabla Ver Contactos que permite seleccionar los contactos almacenados, editar el número y mensaje para enviar (Fig. 3.23).

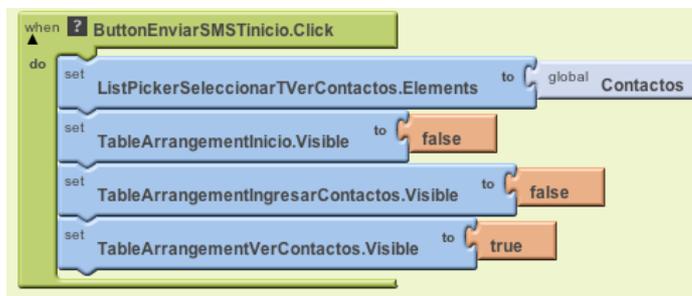


Fig. 3.23. Bloque mostrar tabla ver contactos.

3.3.1.3. Bloque mostrar pantalla ingresar y guardar contactos

En el bloque *ButtonAgregarContactosTinicio.Click* al momento de hacer clic sobre el botón *Agregar Contactos* se muestra la tabla Ingresar Contactos (Fig. 3.24).

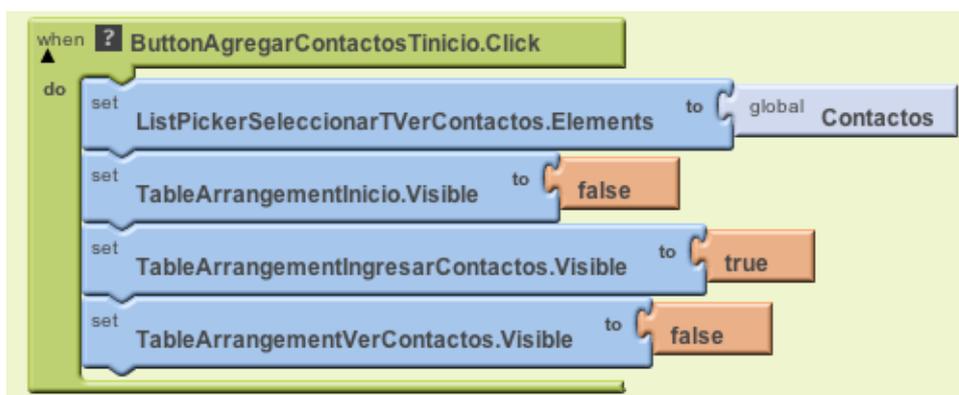


Fig. 3.24. Bloque mostrar tabla ingresar contactos.

3.3.1.4. Bloque regresar a pantalla de inicio

En el bloque `ButtonRegresarTVerContactos.Click` al momento que sea pulsado el botón Regresar se mostrará la tabla de inicio y borrará el contenido almacenado en `TextBoxNumeroTVerContactos.Text` y en `TextBoxEditarSMSTverContactos.Text`, además el botón `ListPickerSeleccionarTVerContactos.Text` se asignará el nombre “Seleccionar” (Fig. 3.25).

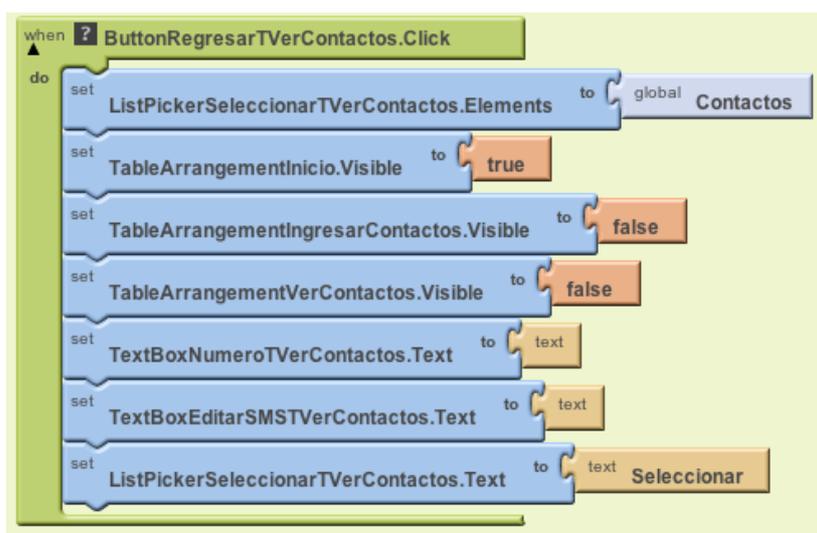


Fig. 3.25. Bloque botón regresar.

3.3.1.5. Bloque ver contactos y editar mensaje

En el bloque *ButtonVerContactosTIngresarContactos.Click* al ser seleccionado el botón Ver Contactos se muestra la tabla para seleccionar contactos y editar mensajes (Fig. 3.26).

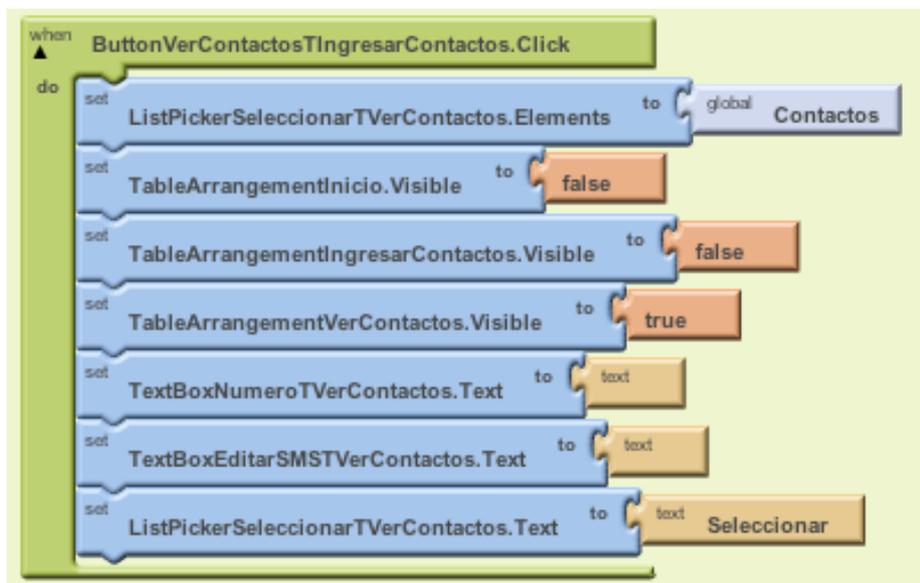


Fig. 3.26. Bloque botón ver contactos y editar SMS.

3.3.1.6. Bloque cancelar y mostrar pantalla de inicio

El bloque *ButtonCancelarTIngresarContactos.Click* al ser seleccionado el botón *Cancelar* se mostrará la tabla de Inicio (Fig. 3.27).

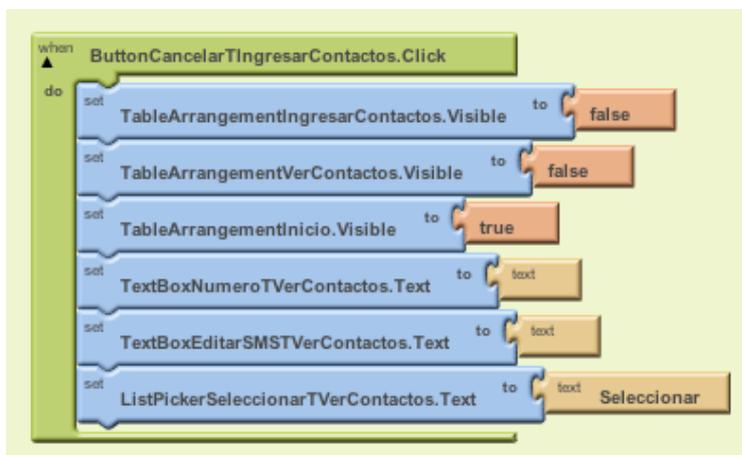


Fig. 3.27. Bloque botón cancelar.

3.3.1.7. Bloque ingreso de contactos

En el bloque *ButtonGuardarTIngresarContactos.Click* cuando sea seleccionado el botón *Guardar* procederá a ingresar el contacto en la base de datos, para ésto se requiere ingresar el nombre y número de contacto para que sean almacenados en una base de datos. El bloque *TinyDB1.StoreValue* permite almacenar los datos ingresados en *TextBoxNombreTIngresarContactos.Text*, donde se ingresa el nombre del contacto y en el bloque *TextBoxNumeroTIngresarContactos.Text*, se ingresa el número.

El bloque de control *if else* permite validar si un nombre ya existe en la lista, en el caso de que exista en la lista, la aplicación mostrará un mensaje que dirá “*NOMBRE YA EXISTE*”, y si no es así se procederá a guardar el contacto en la lista y se indicará que el contacto ha sido guardado, una vez los datos almacenados se procede a borrar el contenido de los cuadros de texto del nombre y número ingresados manualmente (Fig. 3.28).

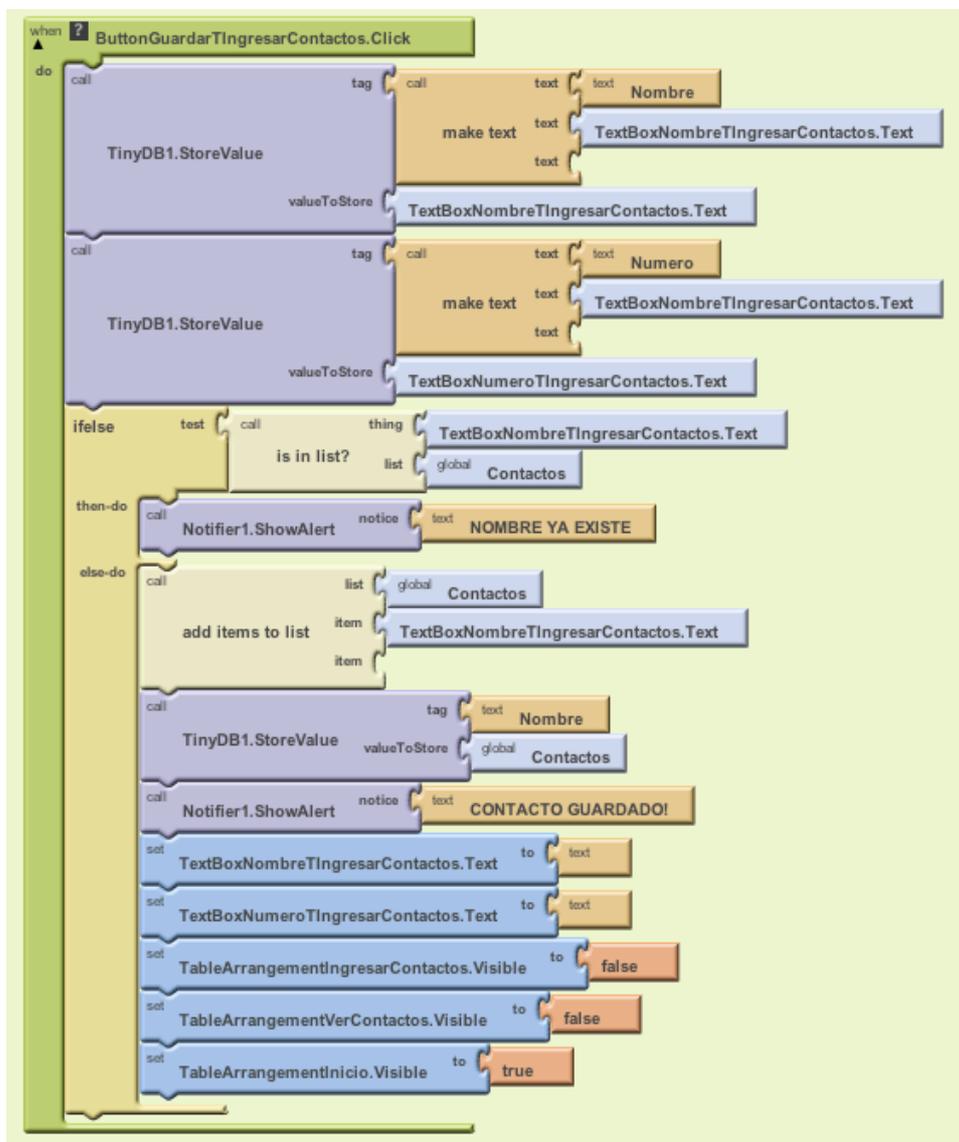


Fig. 3.28. Bloque botón guardar contactos.

3.3.1.8. Bloque de selección de contactos en la lista

El bloque *ListPickerSeleccionarTverContctos.AfterPicking* permite seleccionar un contacto de la lista una vez que se de clic en el botón Seleccionar, una vez que el contacto sea seleccionado, el nombre del contacto se le asignará al texto del botón *ListPickerSeleccionarTVerContactos.Text* y el número de teléfono se asignará al texto del bloque *TextBoxTVerContactos.Text*, esta parte es

llamada por el bloque *TinyDb1.GetValue* que es donde se almacena la base de datos (Fig. 3.29).

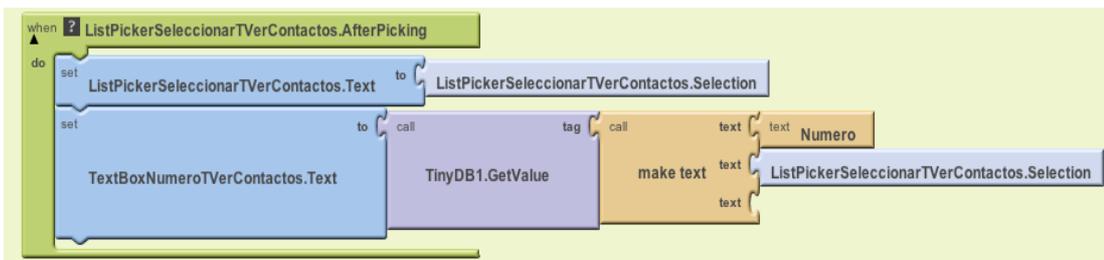


Fig. 3.29. Bloque selección de contacto.

3.3.1.9. Bloque enviar mensaje

En el bloque *ButtonEnviarTVerContactos.Click* cuando sea pulsado el botón *Enviar* se enviará el mensaje al número ingresado en *TextBoxNumeroTVerContactos.Text* y el mensaje editado en *TextBoxEditarSMSTVerContactos.Text*, y se procederá a enviar el mensaje, en caso que el número no haya sido ingresado se mostrará un mensaje de “INGRESE NÚMERO”, una vez enviado el mensaje se mostrará la tabla de Inicio (Fig. 3.30).

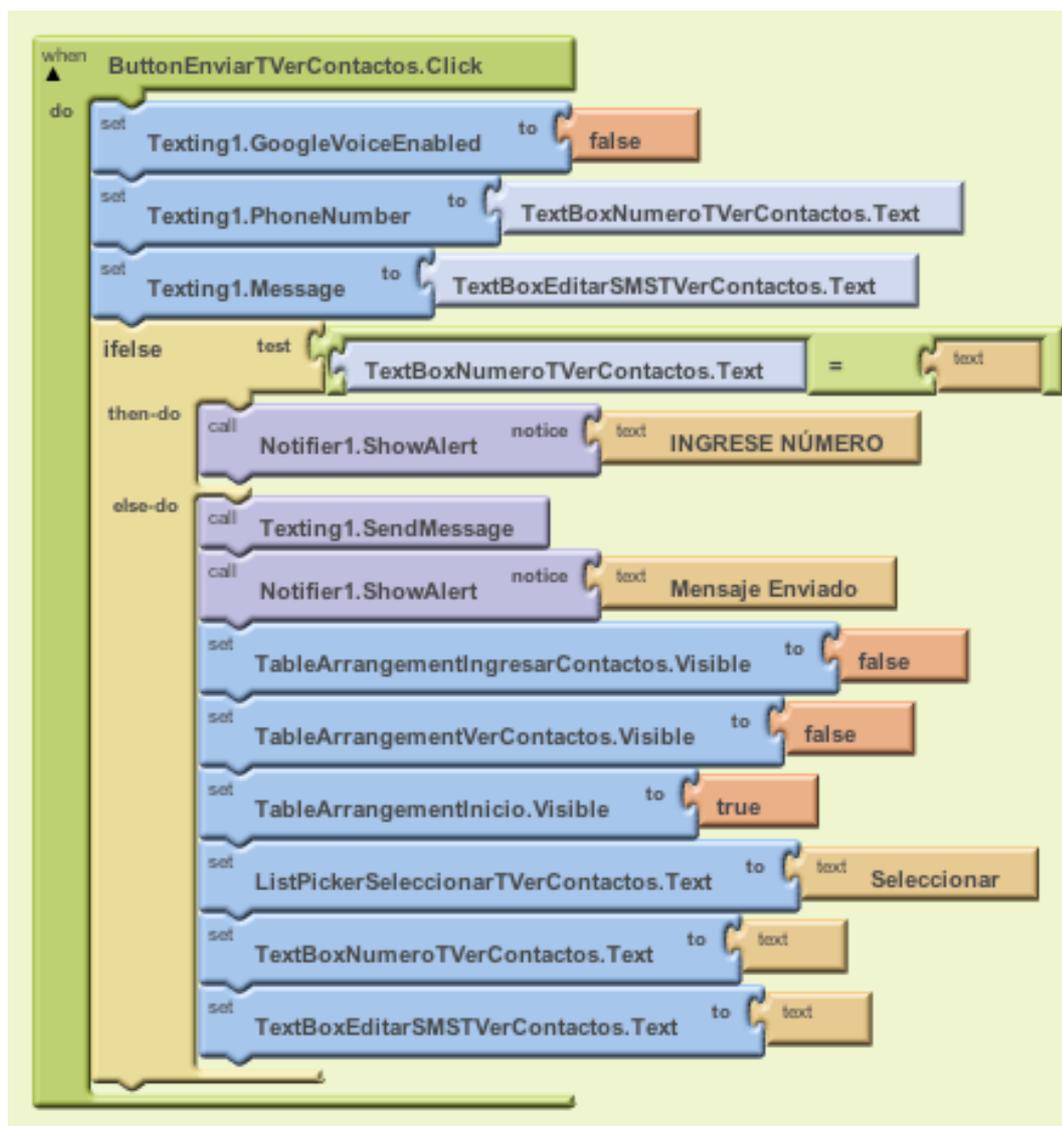


Fig. 3.30. Bloque botón enviar SMS.

3.3.1.10. Bloque realizar llamada

En el bloque *ButtonLlamarTVerContactos.Click*, cuando se seleccione el botón *Llamar* se hará la llamada al número ingresado en *TextBoxNumeroTVerContactos.Text*, si el número no ha sido ingresado

se mostrará un mensaje de “INGRESE NÚMERO”, y después de realizada la llamada se mostrará la tabla de Inicio (Fig. 3.31).

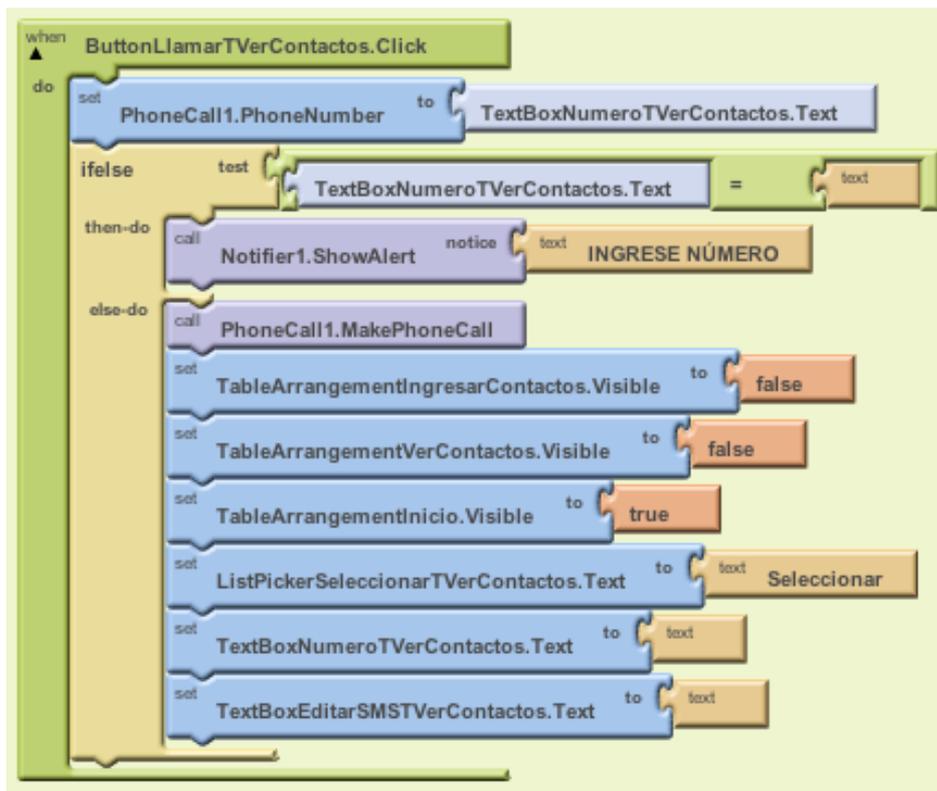


Fig. 3.31. Bloque botón llamar.

3.3.2. Pruebas Iniciales

3.3.2.1. Pruebas iniciales de la aplicación

Una vez desarrollada la aplicación se empezará a realizar las pruebas respectivas de la aplicación y verificar el correcto funcionamiento en cada una de las acciones que tiene la aplicación, como se describe a continuación.

1. La aplicación se instala correctamente en el teléfono celular y se muestra el ícono respectivo de la aplicación en la lista de aplicaciones (Fig. 3.32).

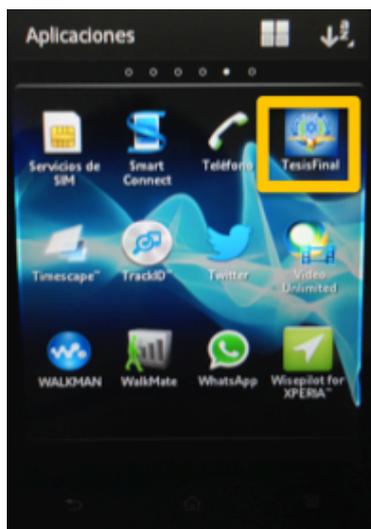


Fig. 3.32. Ícono de la aplicación.

2. Al dar un clic en la aplicación se abre y muestra la pantalla de inicio y seguido de una vibración de 1000 ms (Fig. 3.33).



Fig. 3.33. Pantalla de inicio de la aplicación.

3. A continuación se selecciona la opción *Agregar Contactos* y la aplicación muestra la pantalla con los cuadros de edición de texto para editar número y nombre del contacto a ingresar , además se muestra el botón para almacenar

dicho contacto y otras opciones adicionales como se muestra en la Fig. 3.34.



Fig. 3.34. Pantalla agregar contactos.

4. Ahora se procede a agregar un contacto como se indica en la Fig. 3.35 y se verifica en primer caso que el mensaje de advertencia “*Contacto Guardado*” se mostró y que el contacto fue agregado satisfactoriamente. Enseguida muestra la pantalla de inicio (Fig. 3.36).

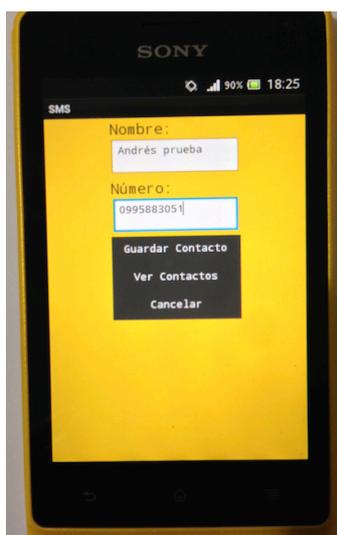


Fig. 3.35. Contacto a agregar.



Fig. 3.36. Mensaje de advertencia.

5. Para enviar un mensaje y verificar que el contacto se haya almacenado correctamente se selecciona la opción “*Enviar Mensaje*” que además de permitir seleccionar los contactos almacenados, permite editar el número y el mensaje a enviar como se muestra en la Fig. 3.37.

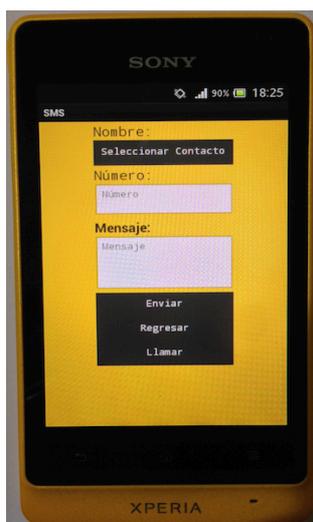


Fig. 3.37. Pantalla seleccionar contactos.

6. Para proceder a enviar un mensaje se selecciona la opción “*Seleccionar*” y se despliega una lista con los contactos almacenados (Fig. 3.38).

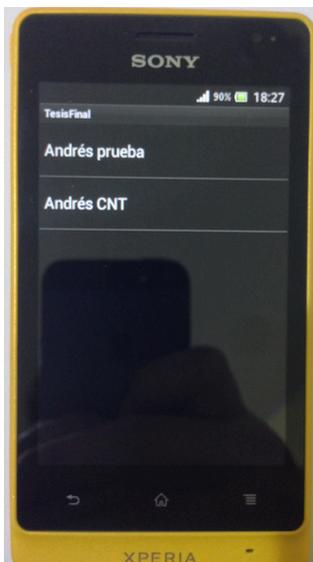


Fig. 3.38. Lista de contactos.

7. Se selecciona el contacto y a continuación se abre la pantalla de la Fig. 3.36, pero con un cambio en el cuadro de texto del botón Seleccionar, aquí se muestra el nombre del contacto seleccionado y en el cuadro de edición de número indica el número del contacto antes seleccionado (Fig. 3.39).



Fig. 3.39. Pantalla contacto seleccionado.

8. Para enviar el mensaje se edita el cuadro de texto del mensaje y se procede a seleccionar el botón *Enviar*, una vez enviado el mensaje se verificó que el mensaje de advertencia "*Mensaje Enviado*" aparece en pantalla (Fig. 3.40).



Fig. 3.40. Advertencia de mensaje enviado.

9. Se verificó que los Botones Cancelar en la Fig. 3.34, Regresar y Llamar de la Fig. 3.37 funcionan correctamente en cada caso redirigiendo a la pantalla de inicio de la aplicación como se muestra en la Fig. 3.33 y el botón Llamar realiza la llamada correspondiente al contacto seleccionado.

10. Al momento de salir de la aplicación y volverla a correr se detectó un error donde los contactos no se muestran en la lista de contactos, para solucionar este error se agregó una subrutina en el bloque de programación para que los contactos se visualicen correctamente.

A continuación se muestra en la Fig. 3.41 la subrutina programada que será llamada en el bloque de inicialización (ver Fig. 3.22), y se obtendrá el bloque final de programación *Screen1.Initialize* como se muestra en la Fig. 3.42.

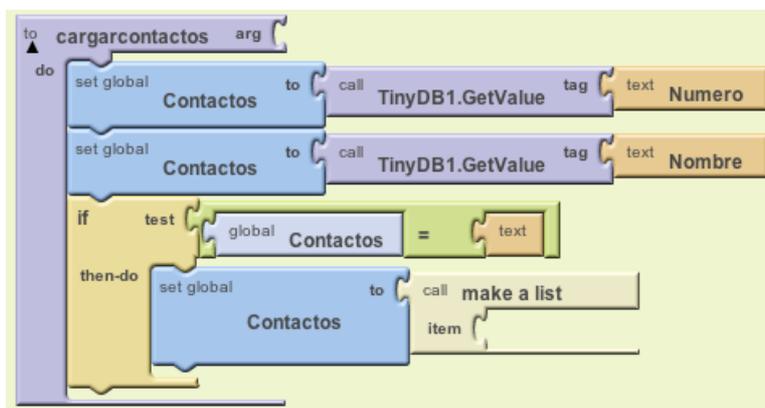


Fig. 3.41. Subrutina carga de contactos.

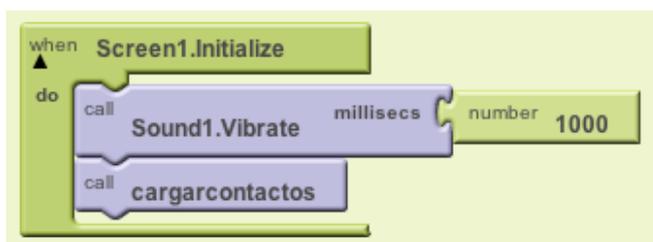


Fig. 3.42. Bloque final de inicialización.

3.3.2.2. Pruebas iniciales del teclado braille

En esta sección se indican las pruebas realizadas para verificar que el teclado ha sido detectado por el teléfono celular y funciona correctamente.

1. Se conecta el cable USB OTG al teclado Braille (Fig. 3.43)



Fig. 3.43. Conexión cable OTG.

2. El otro extremo del cable USB OTG se conecta al puerto mini USB del teléfono celular (Fig. 3.43).



Fig. 3.44. Cable OTG conectado al teléfono celular.

3. Se verifica que el teclado fue reconocido y funciona correctamente como se indica en la Fig. 3.45, Fig. 3.46 y Fig. 3.47.



Fig. 3.45. Teclado USB reconocido.



Fig. 3.46. interactuando con el teclado Braille.

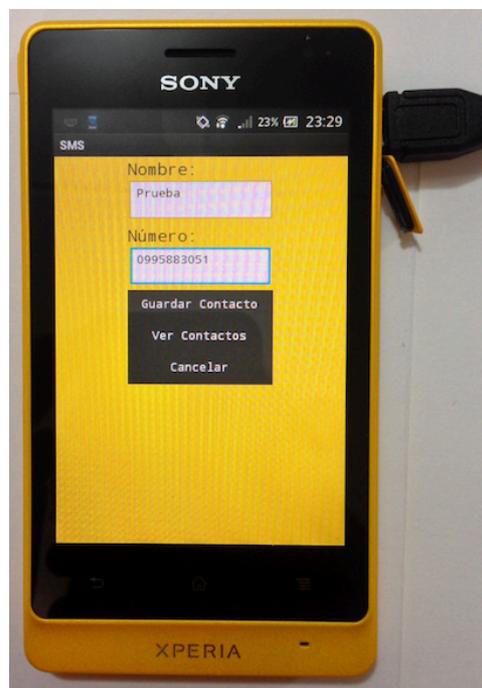


Fig. 3.47. Teclado USB funcionando con la aplicación.

3.3.3 Pruebas de Verificación

Estas pruebas se realizaron con el sistema completamente enlazado y con la aplicación ejecutada en el teléfono celular, se verificó que el teclado Braille ejecuta cada una de la acciones que presenta la aplicación, esto con la ayuda de un Check List de verificación del sistema.

Todas las pruebas realizadas fueron validadas y verificadas por el Ing. Mauricio Alminati como tutor de esta tesis de grado.

1. Una vez conectado el sistema se verificó que con los botones de movimiento de cursor y con el botón Enter del teclado Braille se seleccionó la opción *Agregar Contactos* y se abrió la pantalla correspondiente (Fig. 3.48).



Fig. 3.48. Pantalla agregar contactos.

2. A continuación se procede a ingresar el nombre y número con el teclado Braille verificando las combinaciones correspondientes dadas por el fabricante para cada

letra y número como se muestra en la Fig. 3.49 y a continuación se selecciona la opción *Guardar Contacto* y apareció el Mensaje de “*Contacto Guardado*” y regresó a la pantalla de inicio como se indica en la Fig. 3.50.

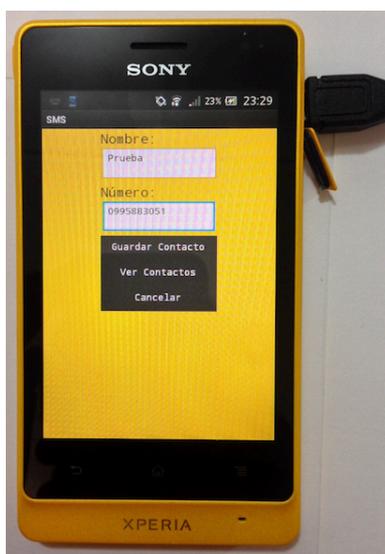


Fig. 3.49. Ingreso de datos de contacto.



Fig. 3.50. Contacto guardado.

3. Para verificar que los contactos se están cargando correctamente se procede a salir de la aplicación presionando el botón Home del celular, a continuación se

ejecuta la aplicación y se debe ingresar en la opción *Enviar Mensaje* y en la opción *Seleccionar* y se verifica que los contactos aparecen en la lista, en el diseño inicial se presentó un problema donde los contactos no se mostraban luego de salir de la aplicación y ahora se muestra que este problema ya no se presenta (Fig. 3.51).



Fig. 3.51. Lista de contactos.

4. A continuación para enviar el mensaje de texto seleccionamos un contacto de la lista de contactos y aparecerán los datos como se muestra en la Fig. 3.52, se edita el mensaje y se selecciona la opción *Enviar* para enviar el mensaje y se muestra el la advertencia de *"Mensaje Enviado"* como se indica en la Fig. 3.53.

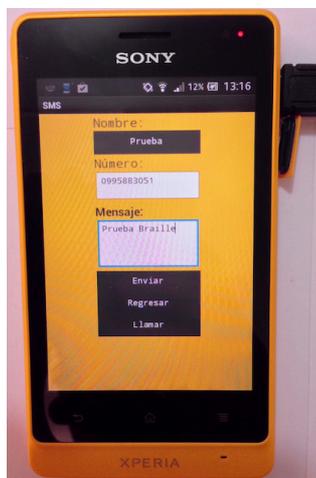


Fig. 3.52. Contacto seleccionado.



Fig. 3.53. Advertencia de mensaje enviado.

3.4. Implementación del prototipo de sistema de escritura Braille para enviar SMS por medio de teléfono celular, de acuerdo a las características del proyecto.

Se indican los procedimientos para la instalación de la aplicación desarrollada y las utilidades que se instalaron para que el sistema tenga un desempeño adecuado y esté puesto a punto para su funcionamiento.

3.4.1. Procedimiento para descargar la aplicación

A continuación se explica el procedimiento para realizar la descarga e instalación de la aplicación desarrollada.

1. Descargar el instalador **TesisFinal.apk** de la siguiente dirección WEB <http://sdrv.ms/1aaHGrG> de SkyDrive (Fig. 3.54), este servicio es proporcionado por Microsoft para compartir archivos.

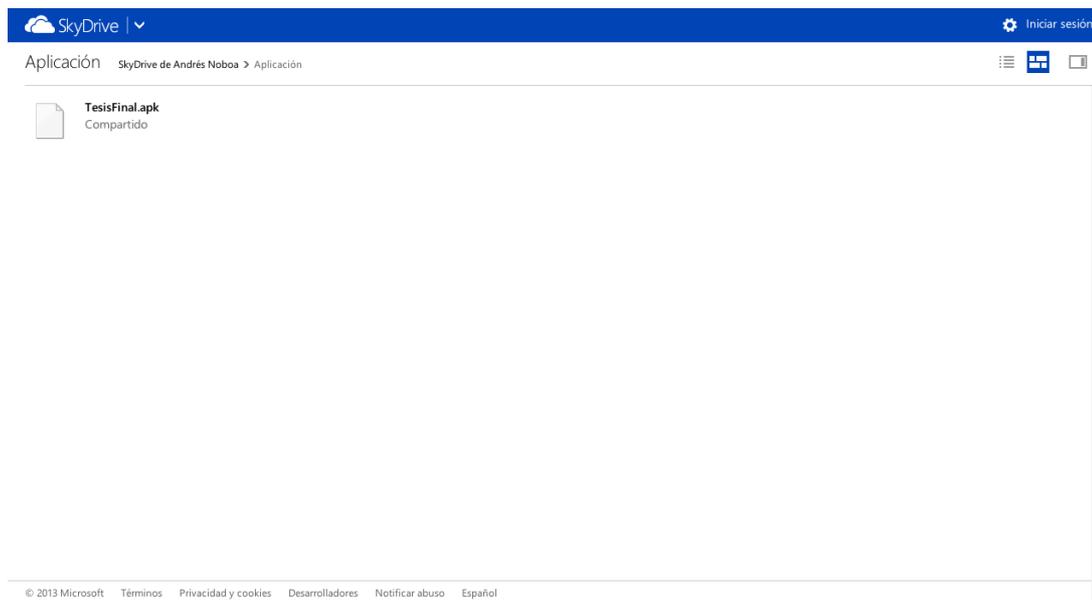


Fig. 3.54. Página SkyDrive.

2. Una vez descargado el instalador que se muestra en la Fig. 3.55, éste se deberá transferir al teléfono celular mediante un cable USB conectado a la PC o vía Bluetooth.

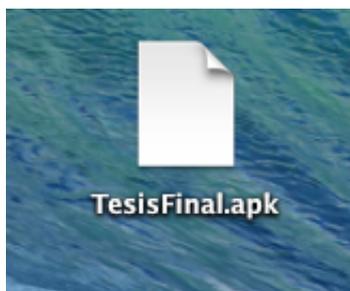


Fig. 3.55. Instalador TesisFinal.apk.

3.4.2. Pasos para la instalación de la aplicación

Para proceder a la instalación de la aplicación se realizan los siguientes pasos:

1. Se debe ir a la carpeta dónde se guardó el archivo **TesisFinal.apk** en el teléfono celular y pulsar sobre el archivo antes mencionado (Fig. 3.56).

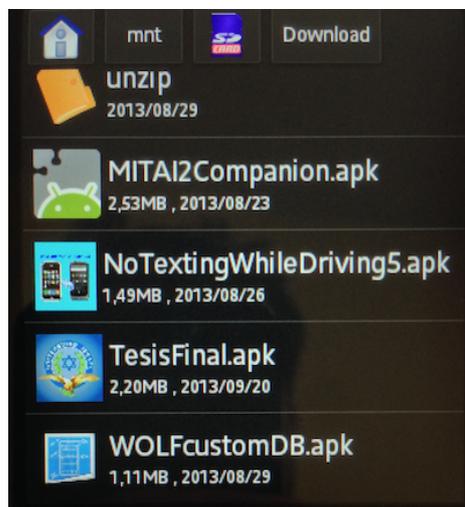


Fig. 3.56. Carpeta de destino.

2. A continuación se muestra una ventana (ver Fig. 3.57) y se procede a seleccionar la opción *Verificar e Instalar*.

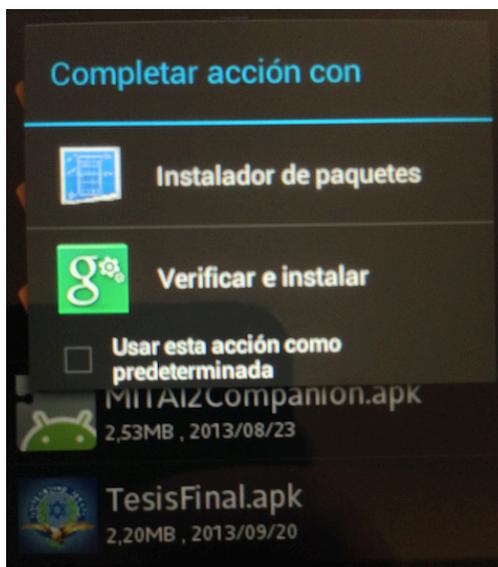


Fig. 3.57. Ventana de verificación.

3. En la Fig. 3.58 se muestra una ventana donde se selecciona *Instalar*, la pantalla mostrará que la aplicación ha sido instalada y se selecciona la opción *Hecho* (ver Fig. 3.59).



Fig. 3.58. Ventana de instalación.

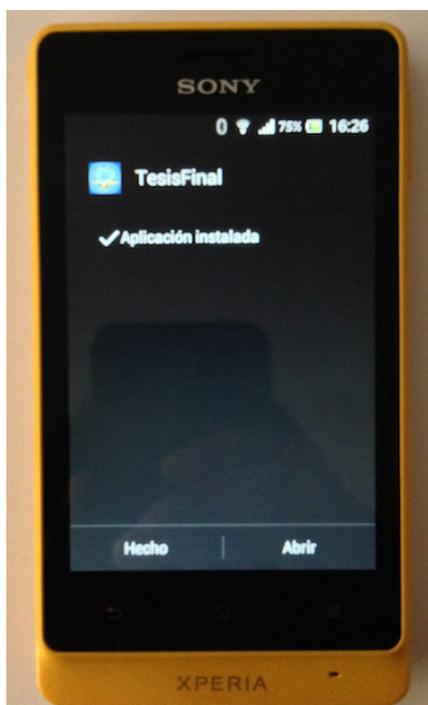


Fig. 3.59. Ventana de finalización de instalación.

3.4.3. Instalación de Talkback

1. Se procede a la instalación de la aplicación TalkBack, esta aplicación se la descarga desde la tienda Play Store de Android que viene ya instalada junto al sistema operativo de Android, este programa sirve para que el teléfono celular hable cuando una opción es seleccionada (Fig. 3.60).

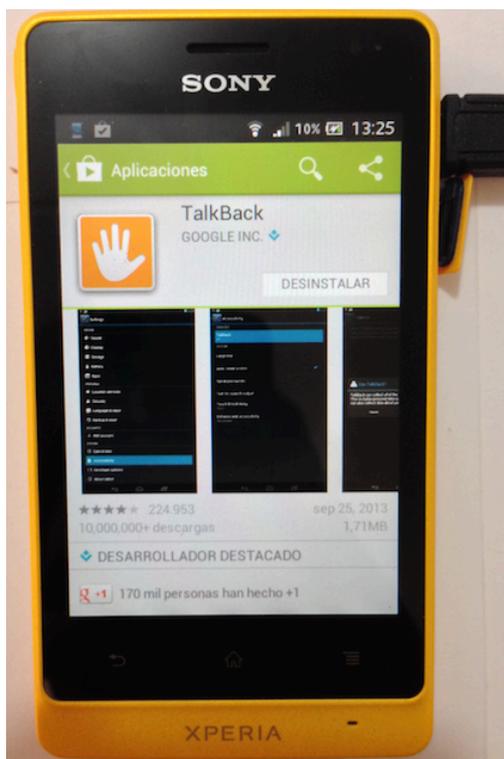


Fig. 3.60. Aplicación TalkBack en la Play Store.

2. Activar la aplicación TalkBack ingresando al menú de configuración del equipo, a continuación ingresar en la opción accesibilidad luego TalkBack y activarla (Fig. 3.61).

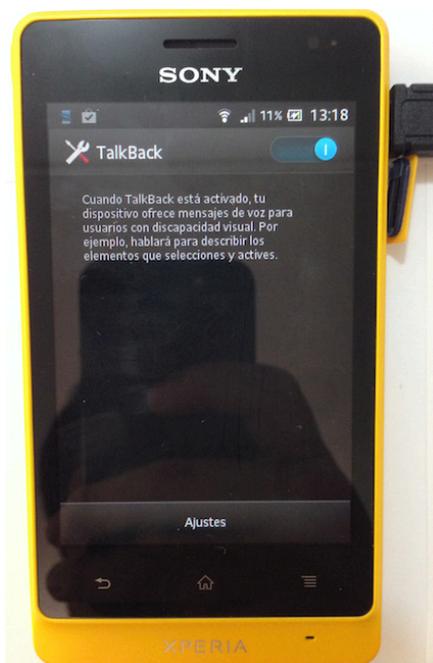


Fig. 3.61. Activación de TalkBack.

3.4.4. Configuración de Smart Connect

Para que la aplicación se ejecute automáticamente cuando el teclado Braille se haya conectado al teléfono celular se hizo uso de una utilidad llamada Smart Connect que viene por defecto en el teléfono celular con Sistema Operativo Android 4.0.1, esta utilidad permite crear una acción cuando un dispositivo externo es conectado al teléfono celular, en este caso se utilizó la acción que cuando el teclado Braille USB sea conectado se ejecute la aplicación TesisFinal (Fig. 3.62).

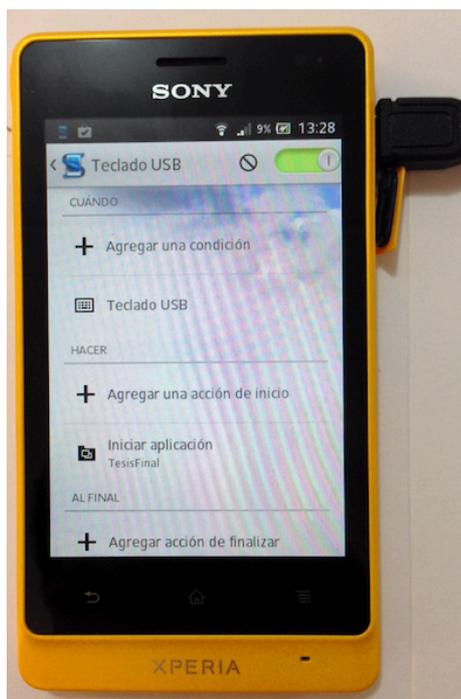


Fig. 3.62. Utilidad Smart Connect.

Con los pasos realizados anteriormente el sistema se configura totalmente y se lo ha puesto a punto para su funcionamiento, esto quiere decir que el proyecto esta completamente operativo con los requerimientos antes propuestos, su funcionamiento se validó con las pruebas realizadas en cada una de sus partes antes mencionadas.

3.5. Validación del funcionamiento del sistema de escritura Braille para enviar SMS vía teléfono celular.

3.5.1. Pruebas de confirmación de implementación del sistema

Una vez implementado el sistema prototipo, se procedió a realizar las pruebas de confirmación de operatividad y conectividad y cuyos resultados se presentan en la siguiente lista de chequeo:

SISTEMA DE ESCRITURA BRAILLE			
ACCIÓN	FUNCIONAMIENTO (%)		OBSERVACIONES
	SI	NO	
Instalación de la Aplicación en el Teléfono Celular	100		
Instalación de TalkBack en el Teléfono Celular	100		
Detección y Funcionamiento del Teclado Braille	100		
Ejecución de la Aplicación	100		La aplicación se demora en abrir aproximadamente 3 segundos después de conectado el teclado.
Funcionamiento de TalkBack	100		
Ingreso del Contacto dentro de la aplicación	100		
Advertencia de Contacto Guardado	100		
Contacto almacenado en la Aplicación	100		
Lista de Contactos Completa	100		
Selección de Contactos	100		
Edición del Mensaje de Texto	100		
Envío del Mensaje de Texto	100		Para el envío de mensajes de texto se debe tener activo un paquete de mensajes o disponer de saldo.
Advertencia de SMS Enviado	100		
Destinatario recibió el SMS	100		El envío de SMS se demora dependiendo del operador utilizado.
Realización de la Llamada	100		
Llamada en curso	100		

Tabla 3.5. Lista de chequeo de operatividad y conectividad.

3.5.2. Pruebas de Funcionamiento

Se realizaron una serie de pruebas con la ayuda de personas con deficiencia visual.

En la Tabla 3.6 se muestra las pruebas de manejo del sistema realizadas gracias a la colaboración de la Srta. María Priscila Garcés Guevara una persona con discapacidad visual y con conocimiento de escritura en teclado Braille.

PRUEBAS DE MANEJO DEL SISTEMA			
ACCIÓN	FUNCIONAMIENTO		OBSERVACIONES
	SI	NO	
El usuario conectó debidamente la interfaz entre el teclado Braille y el teléfono celular	✓		
El usuario seleccionó correctamente la opción "Enviar Mensajes" en la aplicación.	✓		
El usuario ingresó correctamente el número del destinatario 0997595188.	✓		
El usuario escribió correctamente el mensaje "Hola soy María Priscila".	✓		
El usuario seleccionó la opción "Enviar" en la aplicación.	✓		
El mensaje fue recibido por el destinatario.	✓		

Tabla 3.6. Pruebas de manejo del sistema.

A continuación en la Tabla 3.7 se muestran las pruebas de funcionamiento del sistema:

PRUEBAS DE FUNCIONAMIENTO							
Prueba	Acciones propuestas por el usuario		Número de destinatario que escribió el usuario	SMS a ser enviado que editó el usuario	SMS Enviado ó Llamada realizada	SMS Recibido, Llamada recibida ó Finalizar Llamada	Observación
1	Enviar SMS al Número	0997595188	097595188	Hola	No	-	El SMS no se envió porque el usuario escribió erróneamente el número.
	Escribir el siguiente Mensaje	Hola					
2	Enviar SMS al Número	0997595188	0997595188	Hola	Si	Hola	Se realizó la prueba 1 y escribió el número correcto.
	Escribir el siguiente Mensaje	Hola					
3	Enviar SMS al Número	0993888148	0993888148	HOLA	Si	HOLA	
	Escribir el siguiente Mensaje	HOLA					
4	Enviar SMS al Número	0983033179	0983033179	HOLA	No	-	El Usuario escribió erróneamente el mensaje por tanto el SMS no se envió.
	Escribir el siguiente Mensaje	HoLa					
5	Enviar SMS al Número	0983033179	0983033179	HoLa	Si	HoLa	Se volvió a realizar la prueba 4 y a escribir el texto del SMS.
	Escribir el siguiente Mensaje	HoLa					
6	Enviar SMS al Número	0996654607	0996654607	hoLA	Si	hoLA	
	Escribir el siguiente Mensaje	hoLA					
7	Enviar SMS al Número	0984194463	0984194463	Hola esta es una prueba	Si	Hola esta es una prueba	
	Escribir el siguiente Mensaje	Hola esta es una prueba					
8	Enviar SMS al Número	0969085350	0969085350	Hola esta es una prueba de un nuevo sistema	No	-	El usuario escribió erróneamente el mensaje por tanto el SMS no se envió.
	Escribir el siguiente Mensaje	Hola esta es una prueba de un nuevo sistema					
9	Enviar SMS al Número	0969085350	0969085350	Hola esta es una prueba de un nuevo sistema	Si	Hola esto es una prueba de un nuevo sistema	Se volvió a realizar la prueba 8 y a escribir el texto del SMS.
	Escribir el siguiente Mensaje	Hola esta es una prueba de un nuevo sistema					

10	Realizar llamada al número 0995575962	0995575962	-	Si	Llamada Recibida	
11	Finalizar llamada presionando ESC (Ver Anexo 1) en el teclado Braille 0995575962	0995575962	-	-	Llamada Finalizada	
12	Realizar llamada al número 0987254300	-	-	No	-	El usuario no ingresó ningún número.
13	Realizar llamada al número 0987254300	0987254300	-	Si	-	Se volvió a realizar la prueba 12 ingresando el número propuesto.
14	Finalizar llamada presionando ESC (Ver Anexo 1) en el teclado Braille 0987254300	0987254300	-	-	Llamada Finalizada	

Tabla 3.7. Pruebas de funcionamiento.

CAPÍTULO 4

RESULTADOS Y COSTOS DEL PROYECTO

Introducción

En este capítulo se muestra el análisis de las pruebas realizadas bajo los parámetros establecidos; además se presenta un análisis FODA del sistema implementado; y finalmente se presenta una lista de costos de los equipos utilizados en el proyecto.

4.1. Análisis de los Resultados

Prueba 1

El usuario propuso enviar un SMS al número **0997595188** y escribir el mensaje **Hola**, y en esta prueba se verificó que el número de celular propuesto del destinatario fue escrito erróneamente, por tanto el mensaje no se envió y se volvió a realizar la prueba (ver Tabla 3.7).

Prueba 2

Se volvió a realizar la Prueba 1 bajo la misma propuesta, donde el usuario escribió correctamente el número propuesto del destinatario, el mensaje se envió y fue recibido correctamente por el destinatario (ver Tabla 3.7).

Prueba 3

En esta prueba el usuario propuso enviar un SMS al número **0993888148** y escribir un mensaje de **HOLA**, se verificó que el número y mensaje propuesto se escribió correctamente, por lo tanto el mensaje fue enviado y recibido satisfactoriamente (ver Tabla 3.7).

Prueba 4

El usuario propuesto enviar un SMS al número **0983033179** y escribir un mensaje de **HoLa** y se verificó que el usuario escribió incorrectamente el mensaje, por esto el SMS no se envió (ver Tabla 3.7).

Prueba 5

Se volvió a realizar la prueba, se verificó que el número y el mensaje se escribió correctamente por lo tanto el SMS se envió correctamente (ver Tabla 3.7).

Para las pruebas 6, 7, 8, 9 se realizó el mismo procedimiento de verificación del número y mensaje ingresados y propuestos por el usuario (ver Tabla 3.7).

Prueba 10

El usuario propuso realizar una llamada al número **0995575962** y se verificó que el número fue ingresado correctamente, la llamada se conectó satisfactoriamente con el número del destinatario propuesto (ver Tabla 3.7).

Prueba 11

Para terminar la llamada realizada en la prueba 10 el usuario propuso cancelar la llamada presionando ESC (ver Anexo 1), el usuario presionó las teclas indicadas y la llamada finalizó (ver Tabla 3.7).

Prueba 12

Para esta prueba el usuario decidió llamar al número **0987254300** y se verificó que el usuario no ingresó ningún número por lo tanto la llamada no se realizó y la aplicación advirtió que se debe ingresar un número (ver Tabla 3.7).

Prueba 13

Se volvió a realizar la prueba 12 ahora el usuario ingresó el número y correctamente y se procedió a realizar la llamada satisfactoriamente (ver Tabla 3.7).

Prueba 14

Para terminar las pruebas propuestas el usuario decidió finalizar la llamada realizada en la Prueba 13 presionando ESC, y se verificó que la llamada finalizó correctamente y se mostró la pantalla de inicio de la aplicación (ver Tabla 3.7).

Con las pruebas realizadas anteriormente se verificó que el sistema funciona adecuadamente sin ningún problema en el prototipo, ya que los errores presentados durante las pruebas fueron por parte del usuario.

4.2. Matriz FODA

F	O
<ul style="list-style-type: none"> • El sistema es estable y no requiere de un mantenimiento constante. • La aplicación desarrollada es totalmente libre y no se requiere comprar ninguna licencia. • La manipulación del sistema es sencilla y amigable con el usuario. • Lo equipos utilizados son de excelente calidad y se garantiza el funcionamiento óptimo del sistema. 	<ul style="list-style-type: none"> • Con ayuda de la empresa pública o privada el prototipo puede llegar a ser producido en masa. • La calidad de vida de las individuos con deficiencia visual mejorará significativamente. • Con ayuda del gobierno se puede implementar el proyecto e incluir en un plan de ayuda social. • Con ayuda de la sociedad se podrá dar a conocer el sistema desarrollado.
D	A
<ul style="list-style-type: none"> • La tecnología utilizada quede obsoleta con el paso del tiempo. • La fácil manipulación del software libre provocaría cambios indebidos en la aplicación. • El usuario no tenga el conocimiento de escritura en un teclado Braille. • El usuario tenga otro tipo de capacidad especial y no pueda hacer uso del sistema. 	<ul style="list-style-type: none"> • El plagio del proyecto por terceras personas. • En el futuro se diseñen prototipos con mas funcionalidades. • El costo de otros prototipos desarrollados sean considerablemente mas bajo. • El sistema deje de funcionar debido a software mal intencionado instalado en el Smartphone.

Tabla 4.1. Matriz FODA.

4.3. Costos del Proyecto

En la Tabla 4.2. se muestran los costos de los equipos y elementos utilizados y así también la mano de obra requerida para la implementación.

Equipos	Costo
Teclado Braille Portset	400.00
Smartphone Sony Xperia Go	350.00
Cable USB OTG	3.00
Mano de Obra	300.00
Total:	\$ 1053.00

Tabla 4.2. Costos del proyecto.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

- Se aprovechó las prestaciones que brindan los teléfonos celulares fabricados en los últimos años para desarrollar una forma de comunicación fácil destinada a personas con deficiencia visual.
- Para lograr un adecuado desempeño del sistema se necesita que el Software del Smartphone tenga instalado la versión Android 4.0 o superior.
- Para el desarrollo de aplicaciones móviles no se requiere un software exclusivo, sólo se busca crear aplicaciones con facilidad y que sean eficientes al momento de su ejecución.
- El sistema diseñado permite que el prototipo trabaje eficientemente y que el usuario opere con facilidad el sistema.
- La interfaz USB On-The-Go es una gran opción para interactuar con diferentes periféricos y optimizar los recursos de los dispositivos compatibles con esta tecnología.
- Se desarrolló una aplicación para teléfonos móviles para que personas con deficiencia visual puedan hacer uso de la tecnología móvil.

- Para la instalación de las aplicaciones y puesta a punto del software requerido en el sistema, lo deberá hacer una persona sin ninguna discapacidad especial ya que se necesitan varios pasos para instalar los requerimientos.
- Tanto las opciones seleccionadas como las advertencias, se muestran y se verbalizan correctamente en la aplicación del sistema sin generar interferencias que distraigan al usuario.

5.2. Recomendaciones

- El uso de una interfaz inalámbrica se podría integrar al sistema para tener una mejor prestación del prototipo.
- Buscar software de programación que tenga mas utilidades disponibles para desarrollar aplicaciones con mas requerimientos.
- Se puede agregar nuevas funcionalidades a la aplicación para mejorar el uso del sistema.
- La aplicación móvil desarrollada puede ser subida en la tienda Play Store de Android para que en el futuro se pueda realizar actualizaciones.
- Se puede desarrollar y dar soporte a la aplicación en diferentes plataformas móviles como iOS, Windows Phone, Blackberry 10.

Bibliografía

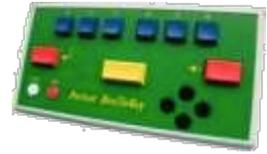
- Portset Systems UK. (2013). *Braille Keyboard*. Recuperado de <http://www.portset.co.uk/braille-keyboards/>
- CONADIS. (2012). *Antecedentes Históricos*. Recuperado de http://www.conadis.gob.ec/index.php?option=com_content&view=article&id=57%3Aantecedentes-historicos&catid=34%3Aentidad&Itemid=1
- ONCE. (2013). *Concepto de ceguera y Deficiencia visual*. Recuperado de <http://www.once.es/new/servicios-especializados-en-discapacidad-visual/discapacidad-visual-aspectos-generales/concepto-de-ceguera-y-deficiencia-visual>
- José Luis Cózar Mata. (2013). *Deficiencia visual: Intervención Psicopedagógica*. Recuperado de <http://www.psicopedagogia.com/articulos/?articulo=459>
- Nicolás Álvarez, Juan Monsalve. (2008). *Programación de dispositivos móviles*. Recuperado de <http://www2.elo.utfsm.cl/~iwg101/ClaseCelulares.pdf>
- MIT APP INVENTOR. (2013). *APP INVENTOR*. Recuperado de <http://appinventor.mit.edu/explore/>

- SONY. (2012). *XPERIA TM GO*. Recuperado de <http://www.sonymobile.com/es/products/phones/xperia-go/specifications/#yellow>
- Aitor Santana. (2013). *Todo sobre el USB OTG*. Recuperado de <http://www.elandroidelibre.com/2013/03/todo-sobre-el-usb-otg-que-es-como-se-usa-es-compatible-mi-smartphone.html>

ANEXOS

ANEXO 1

Códigos de escritura teclado Braille Portset



**THE PORTSET BRAILLE KEYBOARD Universal Computer Version (G1)
Version 4 (Grade 1)**

The Portset BrailleKey is designed as an alternative to the standard QWERTY keyboard, for the entry of information into a personal computer. This device does not replace the standard keyboard for control of the computer.

DESCRIPTION

The Portset Braille Keyboard is an oblong case, with the face sloping slightly from the back to the front.

Connection is via a USB leaded plug from the rear of the case. The USB plug is inserted into a USB port of the personal computer. There is no power cable, like the standard keyboard the power needed to operate the device is taken directly from the computer.

The keys are laid out in the familiar "Perkins" fashion. That is with the six Braille keys at the top, arranged in the sequence 3, 2, 1 and 4, 5, 6. Below these six keys are three other keys, these are from left to right:- Carriage return or "enter" key, space bar, and the back-space delete key. Below these keys on the left are the control keys Ctrl and Alt. On the right side is a cluster of four keys providing cursor up, down, left and right.

These keys are all round in shape and provide additional support for Window based programs.

OPERATION

Because several keys may need to be pressed to form a single character, a character is only sent to the computer when all keys are released.

Shifted (upper case and some other symbols) are obtained by holding down the "enter" key (bottom left, Dot 7) while pressing the keys to make up the standard character.

BRITISH COMPUTER BRAILLE

British Computer Braille is, in most respects, like grade 1 literary Braille. The main exception is in the representation of numbers. Because the computer recognises characters as a single key depression, the principal of double cell characters cannot be used, so the standard method of using a "number" sign before a string of letters cannot be used. Therefore to input numbers 1 to 9 you use the dot configuration for the letters A to I plus dot 6. For example the number 1 is dots 1 and 6, number 2 is dots 1, 2 and 6, etc. The only exception to this rule is zero. Because adding dot 6 to the letter J gives us W, this cannot be used. Therefore for zero all six dots are used.

CONTROL FUNCTIONS.

The Portset BrailleKey provides direct key control of the Ctrl and Alt keys found on a normal QWERTY keyboard. These keys operate on the key press or key down action and are single key functions.

CURSOR CONTROL.

BrailleKey has four cursor keys in a cluster. They perform the same functions as on a QWERTY keyboard and act on the key press action.

SPECIAL FUNCTIONS.

These are called CORDED COMMANDS, The following functions are available:

**Keys 5,6,SPACE toggle a SHIFT command. To turn on SHIFT enter the 3 keys 5,6,SPACE and to turn off SHIFT repeat the key combination. HOME = 1,6,SPACE;
END = 1,4,5,6,SPACE; TAB = 3,4,5,SPACE; ESC = 4,5,6,SPACE.**



THE PORTSET BRAILLE KEYBOARD.
UK International Set UK Keyboard Windows.

Char	Key/Dot	+Dot 7	
a	1	A	Alpha
b	1,2	B	Beta
c	1,4	C	Charlie
d	1,4,5	D	Delta
e	1,5	E	Echo
f	1,2,4	F	Foxtrot
g	1,2,4,5	G	Golf
h	1,2,5	H	Hotel
i	2,4	I	India
j	2,4,5	J	Juliet
k	1,3	K	Kilo
l	1,2,3	L	Lima
m	1,3,4	M	Mike
n	1,3,4,5	N	November
o	1,3,5	O	Oscar
p	1,2,3,4	P	Papa
q	1,2,3,4,5	Q	Quebec
r	1,2,3,5	R	Romeo
s	2,3,4	S	Sierra
t	2,3,4,5	T	Tango
u	1,3,6	U	Uniform
v	1,2,3,6	V	Victor
w	2,4,5,6	W	Whisky
x	1,3,4,6	X	Xray
y	1,3,4,5,6	Y	Yankee
z	1,3,5,6	Z	Zulu
1	1,6	!	
2	1,2,6	"	
3	1,4,6	£	
4	1,4,5,6	\$	
5	1,5,6	%	
6	1,2,4,6	^	
7	1,2,4,5,6	&	
8	1,2,5,6	*	
9	2,4,6	(
0	1,2,3,4,5,6)	

Char	Key/Dot	+Dot 7	
,	2	<	
;	2,3	:	
.	2,5,6	>	
:	2,5	:	colon
[1,2,3,5,6	{	
]	2,3,4,5,6	}	
&	1,2,3,4,6	&	ampersand
+	2,3,5	+	plus
-	3,6	_	
@	2,3,4,6	@	at
=	2,3,5,6	+	
<	2,3,6	<	
"	4	"	
?	2,6	?	question
'	3	@	
/	3,4	?	
)	3,4,5)	
!	3,4,5,6	!	exclamation
_	3,4,6	_	underscore
*	3,5	*	asterisk
>	3,5,6	>	
(4,5	(
\$	4,5,6	\$	
%	4,6	%	percent
	5		
#	5,6	~	
~	6	~	tilde

ANEXO 2

Hoja de trabajo para chequeo de operatividad y conectividad del sistema.

SISTEMA DE ESCRITURA BRAILLE

ACCIÓN	FUNCIONAMIENTO (%)		OBSERVACIONES
	SI	NO	
Instalación de la Aplicación en el Teléfono Celular			
Instalación de TalkBack en el Teléfono Celular			
Detección y Funcionamiento del Teclado Braille			
Ejecución de la Aplicación			
Funcionamiento de TalkBack			
Ingreso del Contacto dentro de la aplicación			
Advertencia de Contacto Guardado			
Contacto almacenado en la Aplicación			
Lista de Contactos Completa			
Selección de Contactos			
Edición del Mensaje de Texto			
Envío del Mensaje de Texto			
Advertencia de SMS Enviado			
Destinatario recibió el SMS			
Realización de la Llamada			
Llamada en curso			

ANEXO 3

Hoja de trabajo para pruebas de manejo del sistema

PRUEBAS DE MANEJO DEL SISTEMA			
ACCIÓN	FUNCIONAMIENTO		OBSERVACIONES
	SI	NO	
El usuario conectó debidamente la interfaz entre el teclado Braille y el teléfono celular			
El usuario seleccionó correctamente la opción "Enviar Mensajes" en la aplicación.			
El usuario ingresó correctamente el número del destinatario.			
El usuario escribió correctamente el mensaje.			
El usuario seleccionó la opción "Enviar" en la aplicación.			
El mensaje fue recibido por el destinatario.			