



TESINA DE LICENCIATURA

Título: Calzado Háptico: Navegabilidad Asistida para Personas con Disminución Visual

Autores: Berretti Fernando

Director: Lic. Claudia Queiruga - Lic. Javier Díaz

Carrera: Licenciatura en Informática (Plan 1990)

Resumen

Las personas con capacidades disminuidas, como los no videntes, forman fuertes lazos de dependencia hacia otras personas, que cumplen la función de suplir la capacidad mermada o perdida en su totalidad. Se forman vínculos de dependencia en ambas direcciones: el no vidente requiere de manera obligatoria la presencia de un ayudante en su desplazamiento y el ayudante debe condicionar sus tiempos y rutinas diarias para cumplir dicha función, deteriorando la calidad de vida de ambos.

El surgimiento de plataformas de prototipado de hardware libre, hacen posible el diseño de soluciones electrónicas de bajo costo y de fácil acceso. Al mismo tiempo, la proliferación de teléfonos inteligentes equipados con sensores que permiten geo-localización, detección del campo magnético, acelerómetros, etc. disponen las condiciones para realizar tareas de innovación.

El presente trabajo de grado busca, hacer uso de los avances mencionados para desarrollar una solución tecnológica integral de bajo costo, en forma de calzado háptico, que implemente navegabilidad asistida en tiempo real, partiendo desde una dirección origen hacia una destino, así como también la detección de obstáculos físicos durante el recorrido.

Palabras Claves

Calzado háptico, Discapacidad visual, Navegación, Dispositivos móviles, Accesibilidad, Open Source Hardware, Open Source Software

Trabajos Realizados

- Análisis de tecnologías de prototipado de hardware
- Evaluación de los servicios intermedios involucrados en el proceso de navegación en tiempo real.
- Construcción de hardware de propósito específico
- Experimentación con interfaces hápticas de feedback vibratorio.
- Desarrollo de aplicación móvil
- Pruebas de usabilidad real del calzado háptico

Conclusiones

El prototipo de calzado háptico desarrollado abre una primera aproximación al perfeccionamiento de este tipo de aplicación de la tecnología para el mejoramiento de la calidad de vida de las personas no videntes, sirviendo de base para desarrollos futuros.

El prototipo desarrollado aplica y contribuye a iniciativas de Open Source Hardware y de Open Source Software en la creación y mejoramiento de productos tecnológicos.

Trabajos Futuros

- Perfeccionamiento de los componentes de hardware que componen el calzado
- Reemplazo de uso de servicios Web por implementaciones locales.
- Detección de obstáculos laterales.
- Extensión del desarrollo para uso en ambos pies mediante la utilización de dos calzados sincronizados



UNIVERSIDAD NACIONAL DE LA PLATA

TESIS DE GRADO

Calzado Háptico: Navegabilidad Asistida para Personas con Disminución Visual

Autor:

Fernando BERRETTI

Directores:

Lic. Claudia QUEIRUGA

Lic. Javier DIAZ

Licenciatura en Informática

Diciembre 2014

“En la utopía de ayer se incubó la realidad de hoy, así como en la utopía de mañana palpitarán nuevas realidades...”

José Ingenieros

UNIVERSIDAD NACIONAL DE LA PLATA

Resumen

Facultad de Informática

Licenciatura en Informática

Calzado Háptico: Navegabilidad Asistida para Personas con Disminución Visual

por Fernando BERRETTI

Las personas con capacidades disminuidas, como los no videntes, forman fuertes lazos de dependencia hacia otras personas, que cumplen la función de suplir la capacidad mermada o perdida en su totalidad. Se forman vínculos de dependencia en ambas direcciones: el no vidente requiere de manera obligatoria la presencia de un ayudante en su desplazamiento y el ayudante debe condicionar sus tiempos y rutinas diarias para cumplir dicha función, deteriorando la calidad de vida de ambos.

El surgimiento de plataformas de prototipado de hardware libre hace posible el diseño de soluciones electrónicas de bajo costo y de fácil acceso. Al mismo tiempo, la proliferación de teléfonos inteligentes equipados con sensores que permiten geo-localización, detección del campo magnético, acelerómetros, etc. dispone las condiciones para realizar tareas de innovación.

El presente trabajo de grado busca hacer uso de los avances mencionados para desarrollar una solución tecnológica integral de bajo costo, en forma de calzado háptico, que implemente navegabilidad asistida en tiempo real, partiendo desde una dirección origen hacia una destino, así como también la detección de obstáculos físicos durante el recorrido.

Agradecimientos

Es una excelente oportunidad para agradecer a mis padres por el apoyo permanente que me brindaron durante toda mi formación universitaria, siendo al mismo tiempo ejemplos de moral y ética en mi vida; a Claudia Queiruga y Javier Diaz por su amabilidad como directores; y especialmente a Sol y Felipe, mis dos amores.

Índice de Contenidos

Resumen	ii
Agradecimientos	iii
Índice de Contenidos	iv
Índice de Figuras	vii
Índice de Tablas	viii
Abreviaturas	ix
1 Introducción	1
1.1 Motivación de la tesis	1
1.2 Objetivo general y desarrollo de la tesis	2
1.3 Trabajo en el cual se inspira la tesis	2
1.4 Introducción a interfaces hápticas	3
1.5 Declaración de intención sobre Open Source Hardware	3
1.6 Estructura del informe	4
2 Open Source Hardware	5
2.1 Marco teórico, conceptos relacionados y OSH	5
2.2 Licencias abiertas (OSH)	8
2.3 Patentes de Hardware	10
2.4 Comparación entre patentes exclusivas y Open Source Hardware	12
2.5 Herramientas OSS usadas en OSH	13
2.6 Elección de plataforma de prototipado para el Calzado Háptico	15
2.6.1 Comparación entre plataformas	16
2.6.2 Plataforma elegida	19
3 Servicios	20
3.1 Cartografía - Mapas	20
3.1.1 Definición	20
3.1.2 Creación de cartografías	21
3.1.3 Derechos de autores y licencias	22
3.1.4 Proveedor elegido para Calzado Háptico	23
3.2 Geocoding	23

3.2.1	Definición, servicios disponibles y resumen de características	23
3.2.2	Licencias/ Términos de uso	23
3.2.3	Proveedor elegido para Calzado Háptico	24
3.3	Routing	25
3.3.1	Definición, servicios disponibles y resumen de características	25
3.3.2	Licencias/Términos de uso	26
3.3.3	Proveedor elegido para Calzado Háptico	26
4	Feedback Háptico	27
4.1	Definición - Esquema a nivel sistema	27
4.2	Sistema háptico humano - Fisiología	29
4.3	Aplicabilidad al Calzado Háptico	33
5	Módulos de Software en el Calzado Háptico	36
5.1	Módulo de reconocimiento de voz	36
5.2	Módulo de síntesis de voz a partir de texto (sustituido por TalkBack)	37
5.3	Módulo de codificación geográfica	38
5.4	Módulo de Ruteo	40
5.5	Módulo de optimización de ruta para navegabilidad en tiempo real	41
5.6	Módulo de navegabilidad en tiempo real	42
6	Módulos de Hardware del Calzado Háptico	47
6.1	Módulo Microcontrolador	47
6.1.1	Arduino Uno R3	47
6.1.2	Arduino Pro Mini	48
6.2	Módulo de feedback háptico	49
6.2.1	Micro-motores vibratorios	49
6.2.2	Módulo controlador de micro-motores vibratorios	49
6.3	Módulo de detección de obstáculos	51
6.4	Módulo de comunicación inalámbrica	53
6.5	Módulo regulador de tensión	53
6.6	Módulo de energía	53
6.7	Interconexiones entre módulos	54
7	Pruebas de usabilidad	55
7.1	Navegación	55
7.1.1	Variables de entrada de información	55
7.1.2	Descripción del contexto.	55
7.1.3	Características a estudiar	56
7.1.4	Resultados individuales de ambos contextos.	56
7.1.5	Interpretación de resultados	61
7.1.6	Conclusiones	61
7.2	Prueba de obstáculos	62
7.2.1	Características a estudiar y resultados	62
7.2.2	Conclusiones	63
7.3	Prueba de usabilidad con discapacitado visual	64
7.4	Conclusiones sobre mejoras posibles	66

8 Descripción de la aplicación	67
9 Conclusiones	69
A Apéndice de figuras	71
Referencias Bibliográficas	74

Lista de Figuras

2.1	Microcontrolador Teensy	16
2.2	Microcontrolador Wiring	17
2.3	Microcontrolador Pinguino	17
2.4	Microcontrolador Arduino	18
4.1	Esquema háptico a nivel sistema	27
4.2	Superposición de campo receptor	29
4.3	Umbral de distancia(milímetros) para diferenciar dos puntos	30
4.4	Relación entre estímulo y respuesta (Gardner,Martin,Jessell,2012)[7]	31
4.5	Esquema de diseño del calzado háptico	33
4.6	Plantilla con motores vibratorios	34
4.7	Esquema háptico aplicado al calzado	35
5.1	Fragmentación de versiones OS Android	38
5.2	Ruta con tramos de 100 metros	41
5.3	Ruta con tramos de 25 metros	42
5.4	Cálculo de puntos guía	43
5.5	Cálculo de ángulos límite	44
5.6	Alineación correcta dentro de límites	44
5.7	Límite derecho sobrepasado	45
5.8	Límite izquierdo sobrepasado	45
5.9	Ángulos límite ampliados dinámicamente	46
6.1	Micromotor vibratorio	49
6.2	Protección contra picos de voltaje	50
6.3	Circuito impreso controlador de motores	50
6.4	Protocolo de funcionamiento(Sensor de distancia)	52
6.5	Limitaciones del sensor de distancia	52
6.6	Esquema de interconexiones	54
8.1	Menú principal de la aplicación	67
8.2	Recorrido visual en tiempo real	68

Lista de Tablas

2.1	Comparación entre licencias abiertas (InMojo,2014)[15]	9
2.2	Comparación entre licencias abiertas de hardware	10
2.3	Comparación entre patentes exclusivas vs Open Source Hardware	12
2.4	Comparación entre los principales EDA open source	14
2.5	Comparación entre modelos de Teensy (Teensy,2014)[34]	16
2.6	Comparación entre modelos de Wiring (Wiring Inc,2014)[37]	17
2.7	Especificación microcontrolador Pinguino (Pinguino,2014)[27]	18
2.8	Comparación entre modelos de Arduino (Arduino,2014)[4]	18
3.1	Comparación entre servicios de cartografías	21
3.2	Derechos de autor sobre los datos(mapas)	22
3.3	Comparación entre proveedores de servicio(Geocoding)	24
3.4	Comparación entre proveedores de servicio(Ruteo)	25
5.1	Métodos destacados(Reconocimiento de Voz)	36
5.2	Métodos destacados(Síntesis de voz)	37
5.3	Métodos destacados(Codificación Geográfica)	39
5.4	Especificación de parámetros para ruteo	40
5.5	Métodos destacados(Ruteo)	41
5.6	Métodos destacados(Optimización de ruta)	42
5.7	Métodos destacados(Optimización con puntos guía)	46
7.1	Resultados de usabilidad (Contexto controlado)	58
7.2	Resultados de usabilidad(Contexto libre)	60
7.3	Materiales utilizados(Prueba en sensor de obstáculos)	63
7.4	Prueba de distancia a objetos(Sensor de obstáculos)	63

Abreviaturas

CAD	C omputer- A ided D esign
HDL	H ardware D escription L anguage
ISP	I n- S ystem P rogrammer
OSH	O pen S ource H ardware
OSM	O pen S treet M ap
OSS	O pen S ource S oftware
OSWHA	O pen S ource H ardware A ssociation
PCB	P rinted C ircuit B oard
PWM	P ulse W ith M odulation
RTOS	R eal T ime O peration S ystem
TTS	T ext T o S peech
YQL	Y ahoo Q uery L anguage

Capítulo 1

Introducción

1.1 Motivación de la tesis

Las personas con capacidades disminuidas, como los no videntes, forman fuertes lazos de dependencia hacia otras personas, que cumplen la función de suplir la capacidad mermada o perdida en su totalidad. Generalmente el que se ocupa de contener a la persona con discapacidad es el núcleo familiar, generando una dependencia en ambas direcciones: el no vidente requiere de manera obligatoria la presencia de un ayudante en su desplazamiento y el ayudante debe condicionar sus tiempos y rutinas diarias para cumplir dicha función, deteriorando la calidad de vida de ambos.

En Latinoamérica el crecimiento del segmento móvil es notorio: existen 328 millones de usuarios individuales de telefonía móvil ([GSM Association,2013](#))[11], de los cuales Argentina cuenta con 30 millones únicos y 60 millones de líneas activas ([INDEC,2013](#))[14], siendo alrededor de 16 millones de usuarios de smartphones ([Google Inc,2013](#))[8] que representan el 30 % de la población. Cada smartphone posee los componentes técnicos para realizar tareas de cómputo de mediana complejidad, así como también sensores que permiten geo-localización, detección del campo magnético,acelerómetros, etc.

Al mismo tiempo, el surgimiento de plataformas para prototipado de hardware libre, como Arduino, hace posible el diseño de soluciones electrónicas de bajo costo y de fácil acceso. La vida de las personas con discapacidades visuales no ha cambiado sustancialmente, pese a los avances de la tecnología([Anirudh Sharma,2012](#))[3]. En Argentina existen 3.272.945 de discapacitados visuales ([INDEC,2010](#))[13] en condiciones de poder mejorar.

El presente trabajo busca hacer uso de los avances mencionados para lograr, por un lado, eliminar la necesidad del acompañante o ayudante al momento de desplazarse caminando

por la vía pública, permitiendo al no vidente moverse de forma autónoma y, por otro lado, también persigue suprimir el uso del bastón blanco como método para detectar obstáculos frontales, liberando las manos para otra actividad, ya que es sabido que los no videntes tienen sobredesarrollados los demás sentidos, como es el caso del tacto.

1.2 Objetivo general y desarrollo de la tesis

El objetivo es desarrollar una solución tecnológica integral de bajo costo en forma de calzado háptico, que implemente navegabilidad asistida en modo turn-by-turn, partiendo desde una dirección origen hacia una destino, y que se complemente con la detección de obstáculos físicos durante el recorrido.

Para lograr el objetivo mencionado fue necesario realizar las siguientes actividades:

- Analizar las tecnologías de prototipado de hardware existentes, con sus respectivas comparaciones a nivel disponibilidad, costo y especificaciones técnicas.
- Evaluar los servicios intermedios involucrados en el proceso de geocoding, routing y cartografías.
- Experimentar con feedback vibratorios para determinar la mejor ubicación según la sensibilidad del sistema del tacto en diferentes partes del cuerpo.
- Abordar conceptos teóricos y prácticos de electrónica para construir hardware de propósito específico que cubra las funciones necesarias para sensor datos del entorno y actuar en función.
- Estudiar la plataforma Android, su manejo de concurrencia y sensores integrados en los dispositivos móviles para desarrollar una aplicación móvil que coordine la navegabilidad.
- Realizar pruebas de usabilidad en contextos controlados y sus diferencias con contextos de uso real.

1.3 Trabajo en el cual se inspira la tesis

El trabajo de grado está inspirado en “Le Chal” ([Vantika Dixit,2011](#))[35] que en hindú significa “llévame allí”, un trabajo llevado a cabo por Sharma y destacado con el premio a la innovación del año en MIT TR35 (innovadores menores a 35 años) y posteriormente publicado en la edición India de MIT Technology Review. Sharma hizo pública la

descripción general de la idea, sin publicar esquemas de diseños y documentación de consulta, por tal motivo no cumple con los requerimientos para ser considerado open source hardware. Se tomó la idea general de este proyecto para desarrollar el trabajo de grado, quedando abierta la posibilidad de realizar modificaciones y adaptaciones.

1.4 Introducción a interfaces hápticas

El término “háptico” carece de definición formal para la Real Academia Española, proviene del griego hápto, que significa “relativo al tacto”.

Los dispositivos hápticos permiten al usuario interactuar con objetos del entorno (real o virtual) a través del sentido del tacto. Una tecnología háptica implica la exploración del entorno por medio de sensores, los cuales adquieren información característica del medio, siendo procesada por el dispositivo para la posterior emisión de una respuesta (feedback) en forma de sensación háptica ([Precision Microdrivers,2012](#))[28].

Históricamente la manera en que un dispositivo indica información de estado al usuario es por medio de señales audibles o alertas visuales. Un ejemplo simple en la industria ha sido el reemplazo de los botones tradicionales por interfaces planas, sensibles a la presión del dedo, permitiendo diseños más elegantes, pero causando la desaparición de la sensación mecánica-física de oprimir un botón ([Precision Microdrivers,2012](#))[29]. Como sistema de compensación se hace necesario el uso de mecanismos de feedback háptico, que mejoran la experiencia del usuario, disminuyendo los errores producidos por falta de respuesta, muy comunes en los dispositivos móviles ([Immersion,2010](#))[12].

Dentro del alcance de este trabajo de grado, se usará feedback háptico como método de comunicación sensorial, mediante vibraciones localizadas en la zona del pie se logra la exploración física -no visual- del entorno y la detección de obstáculos frontales.

1.5 Declaración de intención sobre Open Source Hardware

Este trabajo de grado tiene la intención de ser open source hardware, mediante la publicación de los esquemas de diseño necesarios para la construcción del calzado, así como también del software involucrado y de su documentación; permitiendo la construcción, modificación y/o mejora del prototipo propuesto. Está basado en proyectos abiertos, como el caso de Arduino, de los cuales están disponibles sus esquemas. Todo el software utilizado es open source.

1.6 Estructura del informe

Este informe se divide en 9 capítulos: **Introducción, Open Source Hardware, Análisis de Servicios, Feedback Háptico, Módulos de software, Módulos de hardware, Pruebas de usabilidad, Descripción de la aplicación. Conclusiones.**

Introducción. Se mencionan las razones que llevaron a realizar este proyecto de tesis, los objetivos y desarrollos; una breve introducción a las tecnologías hápticas; así como también se manifiesta la intención de hacer open source hardware el proyecto.

Open Source hardware. Se realiza un recorrido por los conceptos necesarios para comprender la importancia que tienen las licencias abiertas en un desarrollo de hardware y su implicancia en el cambio de paradigma en cuanto a desarrollo de proyectos en la actualidad. Se describen los requerimientos necesarios para calificar un proyecto como open source hardware, en contraposición a una solicitud de patentes. Dentro de este capítulo, se fundamenta la elección de la plataforma a utilizar en el calzado háptico.

Análisis de Servicios. Se describen los servicios necesarios para implementar navegabilidad, sus diferentes proveedores y limitaciones en términos de uso (licencias). Dentro de este capítulo se fundamentan cuáles fueron los proveedores elegidos a utilizar en el calzado háptico.

Feedback Háptico. Se introducen conceptos relacionados sobre tecnologías hápticas, se aborda de forma reducida el sistema háptico humano desde el punto de vista fisiológico y se muestra su aplicabilidad al calzado háptico.

Módulos de Software. Se realiza una explicación funcional detallada, desacoplada en módulos, sobre todas las entidades implementadas a nivel software en el calzado háptico, fundamentando su diseño e implementación.

Módulos de Hardware. Se realiza una explicación detallada, desacoplada en módulos, sobre todas las entidades implementadas a nivel hardware en el calzado háptico, fundamentando su diseño e implementación. Se incluyen los esquemas de cada módulo, así como su interconexión.

Pruebas de usabilidad. Se realizan pruebas en entorno controlado y en entorno real, comparando los resultados y mostrando las conclusiones derivadas de éstos y posibles mejoras.

Descripción de la aplicación. Se provee una descripción de la aplicación móvil y de sus funciones.

Conclusiones. Se mencionan las conclusiones del trabajo.

Capítulo 2

Open Source Hardware

2.1 Marco teórico, conceptos relacionados y OSH

Se puede definir a la propiedad intelectual como el conjunto de derechos que tienen los autores sobre sus creaciones originales. Una acepción más amplia, dominante en el Derecho Internacional, la define en los siguientes términos:

“La propiedad intelectual tiene que ver con las creaciones de la mente, las invenciones, las obras literarias y artísticas, los símbolos, los nombres, las imágenes, los dibujos y modelos utilizados en el comercio” (WIPO,2014)[36]

Se divide en dos categorías: la propiedad industrial, que incluye las patentes, las invenciones, modelos industriales, etc., y el derecho de autor, que abarca las obras literarias, artísticas y científicas, entre otras.

El software es considerado como obra literaria, científica o artística; por tal motivo es tratado como propiedad intelectual, en la categoría de derecho de autor. El derecho de autor es uno de los principales derechos de la propiedad intelectual.

Está reconocido como uno de los derechos fundamentales en la Declaración Universal de los Derechos Humanos y en numerosos tratados. En la normativa internacional no se exige el registro de una obra intelectual a los efectos de su reconocimiento como tal. Desde el momento de su creación, los derechos de autor son adquiridos de forma automática.

Es bueno aclarar que en la legislación argentina el mencionado registro es condición necesaria para el pleno ejercicio de los derechos que reconoce; teniendo un rol clave

ante un litigio legal (INPI,2014)[16]. Los derechos de autores tienen un carácter moral y patrimonial: en relación a su carácter moral se encuentran la paternidad, integridad y divulgación, englobados en la protección de autoría de la obra. No se pueden vender ni transferir, independientemente de la licencia que tenga la obra, deben ser siempre reconocidos por su carácter moral.

En el caso de los derechos de carácter patrimonial se encuentran aquellos vinculados con la reproducción, distribución y transformación, englobados en todo lo que se encuentre relacionado a la explotación de la obra, retribución de uso, reproducción, difusión, etc.. El concepto de Copyright se encuentra ligado casi exclusivamente con el carácter patrimonial de los derechos de autor, los cuales pueden ser transferibles, contrariamente a los de carácter moral.

El mecanismo para transferir los derechos de autor vinculados con la explotación de una obra (patrimoniales) se lleva a cabo mediante la concesión de licencias (sección 2.2); significando esto que el autor puede determinar a qué derechos renunciar o qué derechos ceder, así como también restricciones o permisos sobre su explotación. Si en una obra no se especifica una licencia, ésta tendrá copyright implícitamente, quedando todos los derechos reservados al autor .

Ante la necesidad de encontrar mecanismos intermedios que permitan compartir obras, surgieron licencias menos restrictivas, como es el caso de Creative Commons (CC,2014)[5], pasando de “Todos los derechos reservados“ a “algunos derechos reservados”. Sin embargo, existen licencias más específicas para las obras de software. Como contra-cara al copyright, surgieron desde el ambiente del software licencias de tipo copyleft (sección 1.1.3), en las cuales se eliminan las restricciones de modificación y publicación, con la condición de que todas las obras derivadas de ella mantengan la misma licencia que la original, dándole un comportamiento vírico y asegurando su supervivencia.

Se puede definir al Open Source Hardware en los siguientes términos:

“Hardware cuyo diseño es expuesto de forma pública, para que cualquier persona pueda estudiar, modificar, construir y vender el diseño o el hardware construido a partir de éste”(OSWHA,2014)[25]

Las fuentes del hardware y la forma de diseño con la que fue hecho, están disponibles en un formato a elección, con la posibilidad de hacerle modificaciones. Es importante aclarar que el hardware open source se refiere específicamente al hecho de compartir archivos de diseño digitales de objetos físicos.

Como lo indica la definición, se deben compartir los archivos necesarios para construir y/o modificar el hardware, no siendo aceptadas versiones intermedias u ofuscadas.

Para el diseño mecánico, esto implica los archivos CAD. Para circuitos, los esquemas y los layouts de las placas.

Desafortunadamente, los archivos de diseño habitualmente están realizados en formatos propietarios a partir de herramientas de software propietario. En este caso es de igual ayuda ofrecerlos en formatos alternativos o intermedios que permitan verlos o modificarlos con herramientas de software comunes o gratuitas. Por ejemplo, para los esquemas de circuitos, en formato PDF, Gerbers para los layouts y STL para objetos industriales. Nótese que de todas formas esto no es un sustituto que evita publicar los archivos originales: debe tan sólo ser tomado como una ayuda.

Dar una licencia o licenciar una obra es una forma de darle a otras personas derechos que de otra manera no tendría: permitiéndoles usar derechos de propiedad intelectual a cambio de establecer ciertas restricciones o formas en que se ejercen esos derechos. Por ejemplo, estableciendo una licencia de tipo open source sobre una obra con derechos de autor, se le permite a las personas hacer copias, un derecho que de no ser así, no tendrían; a cambio se les puede pedir que los trabajos derivados de éste sean publicados bajo el mismo tipo de licencia (conocido como copyleft).

Los conceptos mencionados anteriormente son aplicados de manera correcta para código fuente, documentación y para otros trabajos donde los derechos de autor pueden ser aplicados. En el caso del hardware esto no es posible, dado que los derechos de autor, en especial los patrimoniales (copyright) no aplican al hardware de la manera que lo hace al software.

El hardware físico no es una obra literaria, científica o artística, y por tal motivo no se enmarca en los derechos de autor. Dentro de la propiedad intelectual, son encuadrados en propiedad industrial, aplicando al concepto de patentado.

Aplicar a un objeto físico una patente es un proceso costoso y lento, a diferencia de la manera automática en la que se obtienen los derechos de autor en el software.

Las expresiones literarias en archivos de diseño de objetos físicos son cubiertas por el derecho de autor; pero el hardware físico sin una patente no está protegido por los derechos de propiedad intelectual. Una persona no necesita pedir autorización al creador de la obra para copiar, modificar, clonar o construir hardware; quedando éste desprotegido de sus derechos morales de autoría, entre otros.

Si bien los archivos de diseño están protegidos por los derechos de autor, hay muchas formas diferentes de representación de una misma idea. Una idea puede ser representada por esquemas de circuitos totalmente diferentes; creando un conflicto en la protección de la obra; no pudiendo evitar la construcción de un hardware con la misma idea y diferentes representaciones literarias con adaptaciones mínimas en su forma escrita. También es

el caso del uso de ingeniería inversa directa sobre el hardware para la construcción de nuevos archivos de diseño con sutiles modificaciones, sin copiar los originales, logrando que no pudiera ser tomado como una infracción a los derechos de autor.

Aun así, ante la imposibilidad de licenciar los objetos físicos, es interesante aplicar una licencia al diseño del hardware para extender los derechos que naturalmente tiene el autor sobre éstos hacia otras personas que de otra forma no podrían usarlos, expresando los términos y la forma en que pueden ser usados los diseños de su obra.

A modo resumen, un producto OSH debe cumplir ([OSWHA,2014](#))[26] con las siguientes buenas costumbres:

- El hardware debe cumplir con la definición de OSH provista por la OSHWA.
- Permitir a cualquiera estudiar, modificar, distribuir, construir y vender el hardware. Si es usada una licencia CC(Creative Commons) para los archivos de documentación, tener en cuenta usar las opciones compatibles con la definición de abierto. Las licencias CC no-comerciales y de no derivación no son open source.
- Incluir el logo de la OSHWA para que el producto sea rápidamente identificable con OSH.
- Los archivos de diseño y documentación deben estar disponibles públicamente, libres de cargo y en un formato abierto.
- En caso de que no todas las partes del proyecto sean abiertas, se debe especificar claramente cuáles no lo son.

Aún está en discusión el uso de los términos “open hardware” u “open source hardware”. El término “source” tiene su correspondencia directa con código fuente programático, líneas de código, el cual encaja correctamente en diseños de hardware descritos mediante lenguajes como Verilog (HDL), en donde la descripción se realiza por medio de código fuente. En el caso de esquemas (CAD) realizados con software de diseño gráfico asistido por computadora, éste no tiene una correspondencia directa ([Jeffrey M. Osier-Mixon,2010](#))[18].

2.2 Licencias abiertas (OSH)

El copyright protege “la forma de la expresión” más que la invención en sus partes tangibles en el mundo físico. Como ya se ha mencionado, los derechos de autor sobre un diseño

se otorgan automáticamente, así como también los documentos que describen un microprocesador, pero no al microprocesador en sí mismo, en su parte tangible. Hay muchas licencias abiertas que se pueden aplicar al hardware (sólo diseño y documentación) para extender los derechos naturales derivados de los derechos de autor hacia otras personas. Siempre teniendo en cuenta que éstas están basadas en las licencias abiertas de software, que en su formulación están derivadas de los derechos de autor (parte patrimonial - copyright) por calificar como obra artística. La Tabla 2.1 muestra diferentes licencias que pueden ser aplicadas a diseños y documentos en general.

Licencia	MIT	Simplified BSD	Modified BSD	CC BY	CC BY-SA 3.0	GPL/LGPL
Referencia al creador	Opcional	Opcional	No permitido	Requerido	Requerido	No permitido
Obras derivadas deben mantenerse open source	No	No	No	No	Si	Si
Obras derivadas pueden tener un tipo de licencia diferente	Si Mantener derechos de autor y permisos	Si Mantener derechos de autor, condiciones y disclaimer	Si Mantener derechos de autor, condiciones y disclaimer	Si	Si	No
Obras derivadas pueden ser explotadas económicamente	Si	Si	Si	Si	Si	Si

TABLA 2.1: Comparación entre licencias abiertas ([InMojo,2014](#))[15]

No es recomendable usar licencias CC de atribución no-comercial o CC de atribución no derivable, GPL o LGPL, ya que sus restricciones hacen muy difícil que trabajos derivados puedan volver a insertarse en la comunidad con la misma libertad anterior.

Existen licencias específicas para OSH como son el caso de TAPR Open Hardware License (OHL) ([TAPR,2007](#))[33] y CERN Open Hardware Licence ([CERN,2013](#))[6].

La Tabla 2.2 muestra las licencias que pueden ser aplicadas específicamente a hardware.

Según el tipo de licencia, si una persona intenta solicitar una patente sobre la obra, se le son revocados los derechos que había obtenido por medio de la licencia.

Licencia	CERN Open Hardware	TAPR Open Hardware License (OHL)
Referencia al creador	Opcional	Opcional
Obras derivadas deben mantenerse open source	Si	Si
Obras derivadas pueden tener un tipo de licencia diferente	No	No
Obras derivadas pueden ser explotadas económicamente	Si	Si

TABLA 2.2: Comparación entre licencias abiertas de hardware

2.3 Patentes de Hardware

El diseño de hardware puede estar patentado en su forma física y tangible mediante el uso de patentes. Una patente de invención es un derecho exclusivo que el estado otorga al inventor, a cambio de que éste brinde a la sociedad el fruto de su investigación.

La solicitud de patente se publica a los 18 meses, dejando de ser secreta para pasar al estado de público conocimiento.

Las patentes, en su naturaleza, permiten al autor tener el derecho de excluir a otras personas de construir, usar o vender su invención (su parte tangible); ésto es lo contrario a la filosofía del OSH. Las compañías tradicionales de fabricación de hardware realizan contratos licenciados sobre sus diseños de patentes; por ejemplo AMD puede crear microprocesadores basados en una licencia otorgada por Intel, el autor de la invención.

Los tiempos para conseguir una patente son extensos dependiendo del país: en Argentina el proceso completo puede durar un promedio de 5 años.

Todos los titulares de patentes deben, a cambio de la protección de la patente, publicar información sobre su invención, a fin de enriquecer el cuerpo total de conocimiento técnico del mundo. Este creciente volumen de conocimiento público promueve una mayor creatividad e innovación en otras personas. Este cuerpo de conocimiento existente se llama “estado de la técnica”.

Una invención debe, por lo general, satisfacer ciertas condiciones para ser protegida por una patente: la ley de patentes en vigencia establece en su art. 4 los tres requisitos básicos que debe reunir una invención para ser patentable: novedad absoluta, actividad inventiva y aplicación industrial (INPI,2014)[16].

Novedad absoluta: El objeto de invención a patentar no debe estar comprendido dentro del estado de la técnica conocido, entendiéndose por “estado de la técnica” al conjunto de conocimientos técnicos que se han hecho públicos tanto en el país como en el resto del mundo antes de la fecha de presentación de la solicitud en cuestión.

Actividad inventiva: Considerada válida cuando el proceso creativo o sus resultados no se deduzcan del estado de la técnica en forma evidente para una persona normalmente versada en la materia técnica correspondiente. Ésto significa que si la invención es una combinación de elementos conocidos con resultado predecible, la misma carece de actividad inventiva puesto que el resultado final al que se accede partiendo de lo conocido es obvio para cualquier persona del oficio de nivel medio.

Aplicación industria: Válida cuando el objeto de la invención conduce a la obtención de un resultado o de un producto industrial, entendiéndose como industria todo aquello susceptible de ser fabricado o aplicado en serie o a escala industrial, en contraposición a una obra de arte por ejemplo, que es única e irrepetible.

Se puede hacer búsquedas de antecedente a través del Departamento de Información Tecnológica. Dicha dependencia realiza búsqueda de antecedentes para particulares y empresas que deseen saber si el desarrollo que están llevando a cabo ya existe en el país o en el mundo. Para ello, deberán presentar una descripción del objeto de invención y abonar el arancel correspondiente, según la base de datos en la que se requiera la búsqueda.

El tiempo que demandan estas búsquedas externas es de aproximadamente 2 meses y el hecho que no surjan antecedentes no implica que la solicitud a presentar sea novedosa. Una vez concedida la patente, deben pagarse anualidades para su mantenimiento, si éstas no son pagadas opera la caducidad del derecho, es decir, caduca la patente. La protección otorgada no es mundial sino de alcance nacional, aunque se puede hacer uso del convenio de París, para presentar la solicitud de patente en los países que estén incluidos en el tratado, y obtener prioridad; esto significa que cuando se evalúe la patente, se tendrá en cuenta la fecha de la presentación original, siempre y cuando no exceda 1 año a partir de la presentación en el país origen.

2.4 Comparación entre patentes exclusivas y Open Source Hardware

En la tabla 2.3 se observan las variaciones que se pueden obtener sobre un mismo desarrollo según la elección del tipo de protección sobre la propiedad intelectual.

	Patentes exclusivas	Open Source Hardware
Uso Comercial	El inventor se beneficia con la licencias de explotación que decida otorgar a terceras personas.	Beneficio de poder tomar un diseño existente sin ningún costo monetario y venderlo. Fabricar un producto a partir de diseños libres, recibiendo una remuneración monetaria por su venta.
Plagio	Evita el plagio del hardware físico y tangible. Otorga protección completa al autor.	Se protegen los derechos de autor sobre los diseños, pero no se puede evitar el plagio mediante ingeniería inversa en el hardware físico y la posterior creación de diseños sutilmente modificados.
Información Pública	Luego de pasado el período de aceptación, la información se hace pública.	Toda la información para la construcción o modificación es detalladamente documentada de forma pública, en un formato libre.
Monopolio	Obstáculo para la libre competencia, a mayor poder económico, más posibilidades de desarrollo de investigación, lo que lleva a que las empresas grandes retengan las patentes de miles de productos; impidiendo su uso para otras personas.	Fomenta la libre competencia y el desarrollo para todas las personas, brindando un marco equitativo para todos.
Acceso a la tecnología	Dificulta el acceso a la tecnología a los países empobrecidos que no pueden pagar el proceso de patentado o su mantenimiento anual arancelado.	Fomenta la posibilidad de que cualquier persona pueda disponer del acceso a la tecnología, mediante su propia fabricación personal, con costos reducidos.
Incentivo orientado a la investigación	Desincentiva la investigación al establecer un período de utilización exclusiva de una tecnología sin necesidad de mejorarla.	Incentivo permanente hacia la investigación, potenciado por la formación de comunidades, aprovechando la sinergia de trabajo en equipo.

TABLA 2.3: Comparación entre patentes exclusivas vs Open Source Hardware

Un proyecto que se pretende patentar se debe desarrollar en secreto para evitar que sea robado antes de presentar la solicitud de patente. Todo el financiamiento del desarrollo del proyecto es anterior a la aceptación de la patente, lo que es anterior a la posibilidad de comercializarlo, por lo que se necesita mucha inversión previa antes de probar si el producto funciona en el mercado. En el caso del OSH, el costo es mucho menor en

su desarrollo, ya que se puede basar en diseños anteriores, así como también realizar la comercialización inmediata al prototipo desarrollado. El perfeccionamiento del producto se realiza al mismo tiempo de su comercialización, con la ventaja de ser depurado por la propia comunidad. Se requiere invertir tiempo y dinero en la búsqueda de infracciones a los derechos de copia, ingresando en problemas legales que demoran años en resolverse.

En el OSH los mejores métodos de publicitar el producto provienen de los mismos usuarios de la comunidad, quienes suelen ser defensores comprometidos. Como ejemplo de explotación económica en el ambiente de OSH se destaca la empresa Adafruit, la cual genera 15 millones de dólares por año en ingresos por distribuir productos de electrónica open source.

2.5 Herramientas OSS usadas en OSH

Es aconsejable utilizar herramientas open source para el diseño de los diagramas de circuitos, layouts de PCB, y para la generación de documentación en general.

Mediante el uso de open source software en la publicación del proyecto se garantiza que la comunidad pueda visualizar el contenido y modificarlo fácilmente sin las restricciones que se presentan en los formatos privativos, además de ser necesario el pago de un arancel para su uso(OSWHA,2014)[24].

- Dibujos 2D o archivos de diseño asistidos por computadora (CAD) son generalmente usados para describir cortes para impresoras láser de dos dimensiones u otros tipos de trabajos vectoriales.

Herramientas: Inkscape en formato .svg

- Archivos de diseño en 3 dimensiones pueden ser construidos en impresoras 3D.

Herramientas:

<http://sourceforge.net/projects/free-cad/>

<http://www.openscad.org/>

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

<http://meshlab.sourceforge.net/>

- Archivos de diseño de placas, esquemas y archivos de layout de PCB.

Herramientas: KiCad, gEDA. Ver Tabla 2.4.

Aplicación	Esquemas	Simulación	Edición de PCB	Formatos (importar)	Formatos (exportar)
Fritzing	Si + placa de prototipado	No	Si	gEDA symbols, KiCad symbols, SVG	Gerber, DIY etching, BOM, SVG, PDF, EPS
gEDA	Si	Si	Si	Varios	Gerber/drill
KiCad	Si	No	Si	TinyCAD net lists, OrCAD EDIF	PDF, Gerber/drill, netlist, VRML2, IDFv3

TABLA 2.4: Comparación entre los principales EDA open source

- Archivos de diseños auxiliares, que ayudan al usuario a comprender el proyecto sin la necesidad de ser modificados en formatos que no son los originales, pudiendo ser visualizados sin la necesidad de utilizar herramientas propietarias, aunque estén en formatos generados por software propietario.

Formatos:

PDF para esquemas y documentación. STL para diseños 3D

Fotografías : JPEG,PNG,BMP (preferentemente vectoriales)

- Lista de materiales, con sus respectivos números de parte, proveedores y una corta descripción de cada componente son de utilidad, pudiendo ser generados en formato CSV, XLS

Referencias : OpenOffice - Google Docs

- Hosting de los archivos de diseño, así como la documentación en forma de wiki y el código fuente de firmware del desarrollo. La mayoría de estos sitios permiten control de versiones, y facilitan la publicación del material y sus actualizaciones en nuevas versiones. Incluyen issue trackers, para poder tener seguimiento sobre los bugs encontrados por la comunidad.

Referencia: Gitoruis - <https://gitorious.org/>

- Compartir modelos 3d en comunidad :

Referencia: Thingiverse - <http://www.thingiverse.com/>

Como solución integradora en la web, se puede optar por desarrollar el proyecto en un CAD online, que integra todas las funciones de las herramientas mencionadas anteriormente y optimiza el flujo de desarrollo, agregando capacidad colaborativa entre personas.

Referencia: Upverter - <http://upverter.com/>

2.6 Elección de plataforma de prototipado para el Calzado Háptico

En la actualidad existen plataformas de desarrollo de hardware que permiten realizar el prototipado de productos hasta alcanzar su versión de producción. Las plataformas presentan características similares en cuanto a su arquitectura básica y funcionamiento. Sus componentes están montados sobre una placa de circuito impreso, que incluye como componente central un microcontrolador. El microcontrolador puede ser programado para controlar sensores y actuadores.

Los sensores son dispositivos que permiten a la placa adquirir información del entorno que la rodea (temperatura, nivel de luz, distancia a un objeto, etc.). Los actuadores permiten crear cambios en el mundo físico como respuesta a la información sensada.

Estas placas lograron disminuir las dificultades derivadas de los conceptos de electrónica por medio de un entorno de programación, ya que la misma placa puede realizar tareas diferentes, sin necesidad de modificarse su arquitectura interna.

Para agregar sensores y actuadores, el hardware provee conectores llamados pines. Los pines pueden ser configurados como entrada o salida, dependiendo de lo que se quiere lograr, si es una lectura por medio de un sensor, o un efecto en el mundo real por medio de un actuador. La plataforma se conecta a una computadora por medio de un puerto USB, siendo reconocida como un puerto serie, por el cual es programada. Se utiliza un entorno de programación (IDE), para compilar y enviar el software a la placa.

Composición típica de una placa:

Pines digitales de entrada/salida. Pueden ser configurados como entrada o salida por medio de software (IDE). Si es configurado como entrada puede leer un valor high (encendido) y low (apagado) proveniente de un sensor. Cuando es configurado como salida puede encender y apagar actuadores, como motores, luces, generar pulsos, etc.

Pines de entrada analógicos. Son capaces de leer un determinado voltaje, que es traducido en números entre 0 y 1023. Son usados para medir cantidades continuas, como intensidad lumínica, temperatura, proximidad.

Pines de salida PWM (analógicos). Son pines que permiten valores de pulso con ancho modulado, consisten en cambiar de estado de high a low cientos de veces por segundo, simulando el comportamiento de una salida analógica; pudiendo variar la intensidad de una luz, o la velocidad de un motor.

Puertos Seriales. Dos puertos seriales, son usados para programar la placa cuando se la conecta a la computadora, los mismos son disponibles a través del USB: también para tareas de comunicación entre dispositivos.

Pines con funciones especiales. Los pines ISP y TWI (Two Wire Interface) conocidos como interfaz i2c. Permiten conectar muchos sensores/actuadores en forma de red, usando sólo 2 cables. También es posible usarlos para armar una red de placas microcontroladoras

Pines de interrupción externos. Permiten generar o atender una interrupción externa recibida/enviada a un dispositivo.

Fuente de energía. El voltaje necesario para el funcionamiento depende del modelo del microcontrolador y de su frecuencia (Mhz). Según el modelo, tienen un regulador interno que permite voltajes de 5v a 20v. También pueden ser alimentados de forma directa en el pin Raw. Versiones sin conector USB necesitan un adaptador serial-USB, para ser programados.

2.6.1 Comparación entre plataformas

Teensy

Es un sistema de desarrollo microcontrolado basado en USB. Sus dimensiones son reducidas. No es necesario un dispositivo programador adicional para cargar software.



FIGURA 2.1: Microcontrolador Teensy

Especificación de los distintos modelos :

Especificación	Teensy 2.0	Teensy ++ 2.0	Teensy 3.0	Teensy 3.1
Procesador	ATMEGA32U4 8 bit AVR 16 MHz	AT90USB1286 8 bit AVR 16 MHz	MK20DX128 32 bit ARM Cortex-M4 48 MHz	MK20DX256 32 bit ARM Cortex-M4 72 MHz
Memoria Flash	32256	130048	131072	262144
Memoria RAM	2560	8192	16384	65536
EEPROM	1024	4096	2048	2048
I/O	25 a 5 Volt	46 a 5 Volt	34 a 3.3 Volt	34 a 3.3V tolerable 5V
Entradas analógicas	12	8	14	21
PWM	7	9	10	12
UART,I2C,SPI	1,1,1	1,1,1	3,1,1	3,2,1

TABLA 2.5: Comparación entre modelos de Teensy ([Teensy,2014](#))[34]

En su última versión 3.1 tiene como característica destacada que tolera 5 volts en sus pines I/O, brindando compatibilidad a dispositivos que tienen sus líneas de control en 5v . Costo : Teensy 3.1 : US\$ 19.80 - Teensy 2.0: US\$ 16.00

Wiring

Entorno de desarrollo (IDE) open source (basado en Processing), cross-plataform, permite compilar código para cualquier hardware basado en procesadores AVR atmega, AVR Xmega, Avr Tiny, etc. Amplio desarrollo en comunidad y documentación muy completa. Todos los esquemas de diseño disponibles. + 100 librerías de software. Entorno sobre el cual se desarrolla el IDE de Arduino.

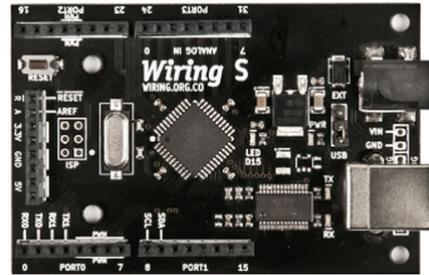


FIGURA 2.2: Microcontrolador Wiring

Especificación de modelos :

Especificación	Wiring hardware V1: atmega128 / Wiring hardware V1.1 Sparkfun atmega1281 / atmega2561	Wiring S (atmega644p)
I/O Pines	54	32
Memoria RAM	128K / 128K / 256K	64K
Entradas analógicas	8	8
Interrupciones Externas	8	3
Puertos Seriales	2	2
USB	Si	Si
Fuente de Energía	7-12V o 5v USB	7-12V o 5v USB
PWM	6	6
Lenguaje de programación	C++ con Wiring Framework	C++ con Wiring Framework

TABLA 2.6: Comparación entre modelos de Wiring ([Wiring Inc,2014](#))[37]

Costo: Wiring S - US\$ 26.99

Pinguino

Placa basada en los microcontroladores @MicroChip. Incluye su propio entorno de desarrollo (IDE) open source en Python. Difiere de Arduino (Atmega) en que posee usb integrado, sin necesidad de chip conversor de usb a serial. Su comunidad es reducida. Las librerías no son compatibles con Arduino. Limitados shields para expansión de funcionalidades.

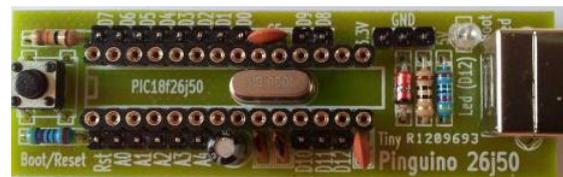


FIGURA 2.3: Microcontrolador Pinguino

Especificación	Microchip PIC18F26J50 48Mhz
Memoria Flash	64000
Memoria RAM	3800
I/O	17
Entradas analógicas	5
PWM	2
UART,I2C,SPI	2
voltaje de operación	2.0 - 3.6V, 5.5V tolerante

TABLA 2.7: Especificación microcontrolador Pinguino (Pinguino,2014)[27]

Modelos: PIC18Fx550, PIC18Fx5K50, PIC18Fx7J53 PIC32MX.

Costo: US\$ 16

Arduino

Placa basada en el microcontrolador Atmega. Es una placa líder en el área, cumpliendo todos los requerimientos para ser open source hardware y open source software. Documentación disponible. Su comunidad de desarrollo es la más grande del mundo.



FIGURA 2.4: Microcontrolador Arduino

Especificación de modelos :

Especificación	Arduino uno (ATmega328)	Arduino Mega (ATmega2560)	Arduino Pro Mini (Atmega168)
Velocidad de CPU	16 Mhz	16 Mhz	8 MHz (3.3V modelo) / 16 MHz (5V modelo)
Memoria Flash	32000	256000	16 KB
Memoria RAM	2048	4000	1024
I/O	14	54	14
Entradas analógicas	6	16	8
PWM	6	15	6
UART	1	4	1
Voltaje de operación	5 V por usb /7-12 V jack	5 V /7-12 V	3.3V modelo/ 5V modelo

TABLA 2.8: Comparación entre modelos de Arduino (Arduino,2014)[4]

Costo: Arduino Pro Mini: US\$ 9.95 (Sparkfun)-Arduino Uno R3: US\$ 24.95 (Sparkfun)

2.6.2 Plataforma elegida

El proyecto se desarrolló usando la plataforma Arduino, por sus ventajas de disponibilidad a nivel nacional, entre otras características mencionadas a continuación. En su declaración de principios promueve el uso de hardware y software abierto, cumpliendo con todos los requerimientos necesarios. Es la plataforma con más cantidad de dispositivos de expansión compatible con su tecnología. La comunidad es la más grande del mundo, realizando mejoras permanentes tanto a nivel de nuevas placas, como del entorno de programación. Disponibilidad de los esquemas de diseño y documentación detallada de consulta.

En comparación con las demás plataformas, es la que más se ajusta a las necesidades del desarrollo propuesto en costos, disponibilidad de componentes, y documentación. Se eligió el modelo “Pro Mini” de Arduino, por razones de espacio y presentaciones (detalle sección 2.1). Otras plataformas poseen procesadores de mayor potencia y mayor número de entradas/salidas, condiciones que no son necesarias para este proyecto. Elegir una plataforma con mayores presentaciones repercute en los costos finales; así como también en el consumo energético.

Plataformas menos difundidas como Pingüino, son interesantes a nivel técnico y en costos de construcción: se descartó para el proyecto por no estar lo suficientemente madura a comparación de Arduino y por su falta de compatibilidad directa en librerías desarrolladas por terceros, no siendo posible el re-uso de lo generado por la comunidad Arduino. La plataforma Wiring posee una comunidad amplia respetable y ha participado en la mejora de Arduino con sus contribuciones, es abierta y bien documentada; Arduino utiliza el entorno de desarrollo creado por Wiring; no fue tomada en cuenta por la dificultad de disponer de esta tecnología a nivel nacional. Teensy es una excelente opción por el tamaño de la placa; presenta una comunidad con participación activa, pero la disponibilidad a nivel nacional es pobre.

Capítulo 3

Servicios

En este capítulo se realiza un recorrido por los servicios necesarios para implementar navegabilidad asistida en tiempo real (turn-by-turn) en el calzado háptico. Para cada servicio se describen diferentes proveedores, sus características, términos de uso y la plataforma elegida para el proyecto.

3.1 Cartografía - Mapas

3.1.1 Definición

En una primera aproximación parece no ser necesario un servicio de cartografías, debido a la imposibilidad del uso visual para el no vidente; la información contenida en los mapas, es utilizada en la fase de desarrollo de la aplicación, con fines de diagnóstico.

También la aplicación cuenta con funcionalidad para que un tercero pueda definir un recorrido a realizar por el no vidente, mediante el uso visual de las cartografías. Si la decisión de diseño fuera la no utilización de servicios web externos (geocoding,routing), sería necesario obtener la información de los mapas, y a partir de ella implementar el resto de las funcionalidades.

Se puede definir a una cartografía como una representación gráfica de un territorio o parte de éste, sobre una superficie plana (bidimensional). Un mapa con propiedades métricas significa que es posible tomar medidas de distancias, ángulos o superficies sobre él y obtener un resultado que se puede relacionar con las mismas medidas realizadas en el mundo real ([RAE,2014](#))[31].

A partir de las cartografías se han construido bases de datos, que permiten asignar atributos característicos a la información geográfica. De esta manera, dada una posición

en el mapa, es posible obtener información relacionada. El uso de tags para asignarle semántica a los puntos georeferenciados permite hacer cálculo de rutas óptimas entre dos o más ubicaciones. La Tabla 3.1 muestra las comparaciones entre los servicios de cartografía.

	Google Maps	Yahoo! Maps	Bing Maps	OpenStreetMap	Nokia Here
Cobertura	Mundial	Mundial	Mundial	Mundial	Mundial
Tipos de mapa	Mapa, Satelital, Híbrida, Callejera, Tráfico, 3D	Mapa, Satelital, Híbrida, Callejera, Tráfico(US))	Mapa, Satelital, Híbrida, Ojo de pájaro, Tráfico, 3D, Street, London Street Map, Ordnance Survey Map, Venue Maps	Mapa, Transporte, Mapa Ciclistico, Variados generados por la comunidad	Mapa, Satelital, Terrestre, 3D, Tráfico, Transporte Publico, Temperaturas, Map Creator para la comunidad
Zoom	19	19	23	19	18

TABLA 3.1: Comparación entre servicios de cartografías

3.1.2 Creación de cartografías

Todos los servicios mencionados en la [sección 3.1.1](#) utilizan recursos privados para el relevamiento de los datos geográficos necesarios para la construcción de sus mapas. En la mayoría de los casos usan licencias otorgadas por empresas especializadas en relevar información geográfica.

El servicio que se distingue del resto es OpenStreetMap, cuyos mapas están construidos por la comunidad, utilizando gps personales para el trazado de mapas o el completado de información característica de forma manual por cada miembro. También hay contribuciones de imágenes aéreas y satelitales de carácter público, provistas por organismos del estado de cada país.

En empresas privadas, como Here o Google Maps, se necesita una importante inversión en recursos tecnológicos y humanos: más de 6.000 personas trabajan en HERE en 53 países ([Javier Pastor, 2014](#))^[17]. Proveen variados tipos de mapas, el relevamiento es más complejo, se realiza por medio de autos con equipo especial para obtener mapas de vista callejera, o modelos en 3d.

OpenStreetMap mediante la sinergia de su comunidad, provee información actualizada a mayor velocidad de lo que lo puede hacer una privada, ya que los procesos de integración y verificación de información son mucho más lentos.

3.1.3 Derechos de autores y licencias

Todos los servicios, a excepción de OpenStreetMap se reservan los derechos de autor sobre sus mapas.

Servicio	Google Maps	Yahoo! Maps	Bing Maps	OpenStreetMap	Nokia Here
Derechos de Autor	Google Inc + licencias de proveedores	Nokia + licencias de proveedores	Microsoft + Nokia + licencias de proveedores	1.840.000 usuarios de los cuales 22.600 editores activos	Nokia + licencias de proveedores

TABLA 3.2: Derechos de autor sobre los datos(mapas)

OpenStreetMap hace uso de una licencia Open Data; más específicamente Open Data Commons Open Database License(ODbL). La licencia define los siguientes derechos:

“Se puede copiar, distribuir, transmitir y adaptar nuestros mapas e información libremente siempre y cuando se dé reconocimiento a OpenStreetMap y sus colaboradores. Si se altera o genera contenido sobre nuestros mapas e información, se debe distribuir los cambios bajo la misma licencia” ([OpenStreetMap,2014](#))[22]

Esta licencia deja en claro que se puede usar los mapas para el propósito que se desee, en contraposición con los demás servicios propietarios.

Términos de uso de Google:

Restrictions against Data Export or Copying.

(a) **No Unauthorized Copying, Modification, Creation of Derivative Works, or Display of the Content.** You must not copy, translate, modify, or create a derivative work (including creating or contributing to a database) of, or publicly display any Content or any part ...; For example, the following are prohibited: (i) creating server-side modification of map tiles; (ii) stitching multiple static map images together to display a map that is larger than permitted ...; (iv) exporting, writing, or saving the Content to a third party’s location-based platform or service. (b) **No Pre-Fetching, Caching, or Storage of Content.** You must not pre-fetch, cache, or store any Content, except that you may store: (i) limited amounts of Content for the purpose of improving the performance of your Maps API Implementation if you do so temporarily (and in no event for more than 30 calendar days). ([Google Inc,2014](#))[9]

Las restricciones en los términos de uso de las demás plataformas son muy similares a la citada.

3.1.4 Proveedor elegido para Calzado Háptico

Se eligió como proveedor del servicio OpenStreetMap, tomando como punto principal su licencia abierta. La información está disponible para cualquier fin, permitiendo realizar copias digitales o impresiones en mapas de papel. La actualización de nuevos datos es inmediata, con tiempos menores de validación, en contraposición con servicios pagos donde el mismo proceso puede demorar meses.

La cartografía es más precisa, con corrección de errores provistos por la comunidad, por medio de usuarios que residen en la región a modificar, obteniendo información de primera mano, a diferencia de lo provisto por empresas propietarias que, dada la propia limitación de recursos, no pueden abarcar todas las zonas con el conocimiento específico que sólo se puede dar viviendo en la región.

La posibilidad de descargar versiones vectoriales para uso fuera de línea abre la posibilidad al desarrollo de aplicaciones que de otra forma no serían posibles.

3.2 Geocoding

3.2.1 Definición, servicios disponibles y resumen de características

Un servicio de geocoding directo se encarga de traducir información geográfica expresada en forma de texto, generalmente una dirección de calle, hacia coordenadas geográficas en formato de latitud, longitud ([Android Inc, 2014](#)) [2]. Los servicios de geocoding son variados, su dificultad radica en la capacidad de cobertura mundial y en su precisión de resultados. Es difícil reunir dichas condiciones en un servicio abierto y sin restricciones. Generalmente los servicios pagos en su variante gratuita presentan limitaciones de uso.

3.2.2 Licencias/ Términos de uso

Here y Google en sus términos de uso, prohíben la utilización independiente del servicio de geocoding. Los derechos de uso están otorgados sólo cuando se utilizan sobre sus propios mapas.

	Here	Google	Yahoo	Opencagedata	Nominatim
Prueba de precisión (La Plata)	Excelente	Excelente	Excelente	Media	Media
Método de consulta	Geocoder API, Enterprise JavaScript API	API JavaScript de Google Maps	YQL (consulta web en formato sql-like)	API web	Api web + motor local en servidor propio
Descomposición de dirección en partes	Permitido	Permitido	Permitido	No permitido (interpretado server-side en única línea)	Modo experimental
Restricciones de uso en mapa	Solo con sus mapas	Solo con sus mapas	Libre a través de tablas YQL servicio placefinder	Libre en su etapa beta	Libre

TABLA 3.3: Comparación entre proveedores de servicio(Geocoding)

Los proveedores mas destacados en servicio de geocoding se describen en la Tabla 3.3.

Licencia de uso de Google:

No Use of Content without a Google Map. You must not use or display the Content without a corresponding Google map, unless you are explicitly permitted to do so in the Maps APIs Documentation. In any event, you must not use or display the Content on or in conjunction with a non-Google map. For example, you must not use geocodes obtained through the Service in conjunction with a non-Google map. (Google Inc,2014)[9]

Yahoo Geo permite el uso independiente de su servicio de geocoding, que es parte de BOSS Geo Services, mediante el lenguaje de consultas YQL.(Yahoo,2014)[38]

Opencagedata es open source, ya que utiliza servicios abiertos en su implementación, es una API de agregación. Nominatim se puede descargar para montar un servidor de búsqueda propio, la api web provista por OpenStreetMap tiene restricciones de uso, ya que se encuentra instalada en servidores donados y no desean sobrecargarlos con tráfico.

3.2.3 Proveedor elegido para Calzado Háptico

Se eligió BOSS Geo Services de Yahoo, mediante YQL (Yahoo Query Language), por ser la más completa en prestaciones. En la ciudad de La Plata la calidad de los resultados es excelente, sumado a la posibilidad de formar direcciones de calles por construcción de componentes (sección 5.3), mejorando el porcentaje de aciertos. No es necesario utilizar

los mapas de Yahoo, se utiliza de forma independiente. La manera en que se expresan los requerimientos en yahoo query language, agrega legibilidad a la construcción. Opencagedata también es una opción a tener en cuenta, al ser una API de agregación, en su interior utiliza servicios de yahoo, pero fue descartada por estar en una fase beta de desarrollo(OpenCage,2014)[21].

3.3 Routing

3.3.1 Definición, servicios disponibles y resumen de características

El Servicio de routing, permite calcular una ruta desde una ubicación origen hacia una ubicación destino. Los algoritmos de ruteo utilizan la información contenida en los mapas para poder determinar la ruta óptima. Es importante mencionar que sólo se obtienen los puntos que forman el recorrido, sobre el cual es necesario implementar un algoritmo que realice la navegación en tiempo real; sin un algoritmo de navegación online la ruta sólo sirve a modo de itinerario de recorrido. La Tabla 3.3.1 describe los servicios de ruteo que pueden ser basados en web o fuera de línea.

	Recorrido a pie	Actualización de rutas	Cobertura	Uso del servicio independiente de sus mapas	Fuera de línea
Google Maps	Si	Regular	Mundial	No	No
Yahoo maps	Si	Regular	Mundial	No	No
Here(Nokia)	Si	Regular	Mundial	No	No
GraphHopper	Si	Dependiente del usuario y sus mapas	Dependiente del mapa utilizado	Si	Si
MapQuest	Si	Diaria	Mundial	Si	No
OpenRoute	Si	Diaria	Europa	Si	No
YOURS	Si	Pobre	Mundial	Si	No

TABLA 3.4: Comparación entre proveedores de servicio(Ruteo)

Los servicios analizados soportan recorridos a pie; es decir, sin tener en cuenta el sentido de la mano en la calle, ignorando los tags de semántica asociados a las restricciones en cuanto a giros permitidos. Su cobertura varía en forma directa según el alcance y la precisión de los mapas en los que están basados; por tal motivo su precisión es proporcional a los atributos analizados en [sección 3.1.1](#). Google, Yahoo y Here usan sus API online de mapas para brindar este servicio. GraphHopper es un motor de routing

fuera de línea, eficiente y de gran velocidad, basado en java. Funciona sobre desktop, en modo servidor y también se puede utilizar en Android. Usa algoritmos Dijkstra, A* con optimizaciones especiales ([Graphhopper,2014](#))[10]. Es necesario generar un mapa de rutas, derivado del mapa original de OpenStreetMap.

3.3.2 Licencias/Términos de uso

Como mencionamos anteriormente, los algoritmos de navegación utilizan la información contenida en los mapas para poder determinar la ruta óptima. Es éste el motivo por el cual mapas con licencias cerradas no pueden ser utilizados para dicho fin, siendo el caso de Google, Here(Nokia), Yahoo maps, Bing maps (Microsoft).

En sus licencias/términos se hace explícito que los servicios de routing (también válido para geocoding) se pueden utilizar únicamente para visualización en sus propios mapas, generando una dependencia entre el servicio web provisto y la visualización de la información, no pudiendo ser usada sin fines de visualización y de ser visualizada debe mostrarse en sus mapas.

Como punto de partida se necesita un servicio de routing que se apoye en una cartografía libre, como es el caso de Mapquest, en su versión de Open Data, que utiliza los mapas de OpenStreetMap en la implementación de su API web de routing. ([Mapquest,2014](#))[20] El motor de GraphHopper utiliza licencia abierta Apache. Open route Service licencia BSD. YOURS utiliza licencia Creative Commons Attribution-ShareAlike 2.0 license.

3.3.3 Proveedor elegido para Calzado Háptico

El proveedor de servicio de routing elegido es Mapquest, que por estar basado en OpenStreetMap es totalmente libre de uso ([Mapquest,2011](#))[19]. También se consideró la posibilidad de utilizar el motor fuera de línea GraphHopper, evitando la necesidad de tener conexión a internet, su desventaja es que requiere los mapas de OpenStreetMap actualizados constantemente, descargándolos al dispositivo, para luego derivar un archivo de rutas, con el consiguiente consumo de cómputo y tiempo en ser generador.

La ventaja de usar el servicio web de Mapquest es que las actualizaciones son transparentes, siendo automáticas del lado del servidor al actualizar sus mapas de OSM. Google, Yahoo y Here, fueron descartados por sus políticas restrictivas de uso y los restantes por falta de cobertura o precisión.

Capítulo 4

Feedback Háptico

4.1 Definición - Esquema a nivel sistema

La representación abstracta a nivel sistema de cómo funciona una tecnología háptica se puede ver en la Figura 4.1.

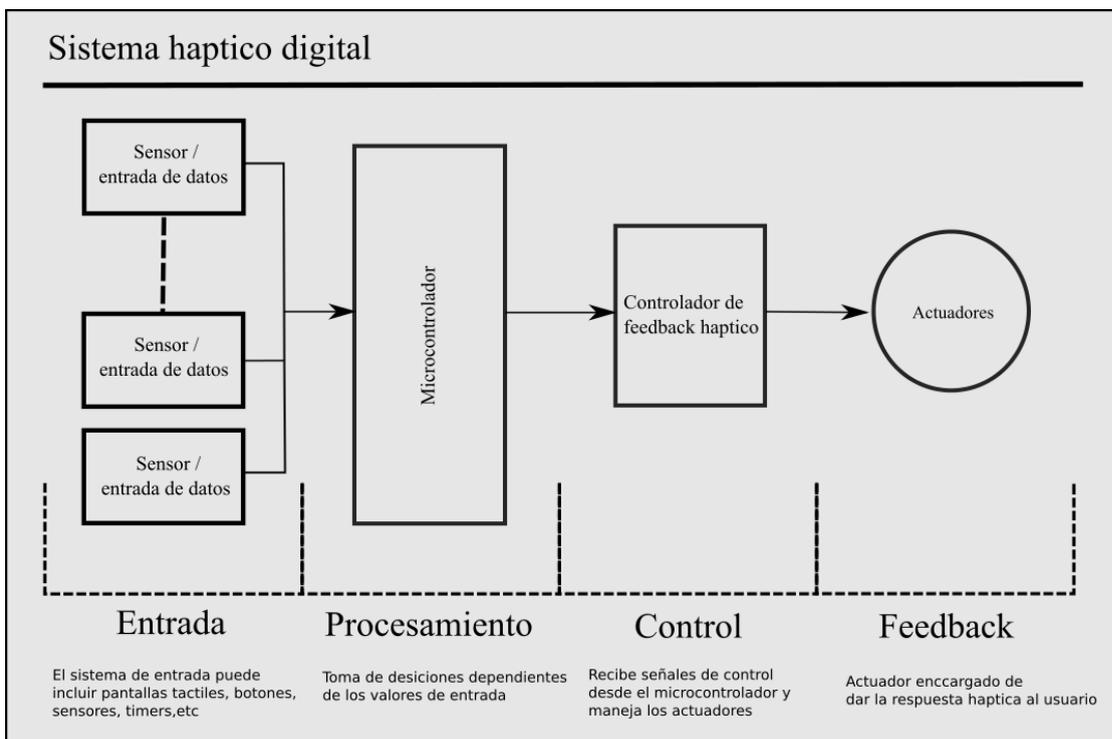


FIGURA 4.1: Esquema háptico a nivel sistema

Este trabajo de grado se ocupa del uso de tecnologías hápticas especializadas en feedback sensorial de tipo vibratorio, no incluyendo tecnologías hápticas de otras áreas, como puede ser el caso de feedbacks de fuerza, etc..

Entradas de datos. Para poder transmitir información de estado al usuario, primero se debe definir cuáles van a ser los eventos o datos de entrada del mundo real que van a ser tenidos en cuenta para posteriormente ser representados en forma de feedback háptico. Esta información es sensada a través de instrumentación específica y convertida en digital, quedando disponible para su inmediato procesamiento.

Microcontrolador. La lógica de control sobre la información de entrada es procesada por un microcontrolador, encargado de procesar los datos y activar las líneas de control según corresponda. El actuador no suele estar conectado de forma directa al microcontrolador por cuestiones de incompatibilidad de recursos electrónicos, por tal motivo se necesita un controlador que maneje los actuadores. Los microcontroladores sólo manejan líneas de control, señales, delegando la implementación del consumo y protección de circuitos a un controlador específico del actuador. Mediante líneas de control especiales, llamadas PWM, se puede variar la intensidad de operación sobre el actuador. En el caso de un motor, definir sus revoluciones por minuto, variando su voltaje.

Controlador de actuadores. El circuito controlador de actuadores se encarga de manejar la corriente suficiente para la activación del actuador, en el caso de un motor, el consumo mayor se da en el arranque. El microcontrolador no puede suministrar este voltaje y sólo se encarga de activar/desactivar las líneas de control lógico, para encender/apagar o variar la intensidad de funcionamiento del motor.

Actuador háptico. El actuador háptico es un micro motor vibratorio que, según sus características técnicas, logra patrones de vibración que son captadas por el usuario final del sistema. Como se mencionó en la introducción ([capítulo 1](#)), las tecnologías hápticas permiten, entre otras posibilidades, dar información de estados y sus cambios, especialmente útil en situaciones donde la sensación real que acompaña al cambio de estado fue suprimida. Como mecanismo de compensación se hace necesario desarrollar tecnologías de feedback táctil que relacionan objetos y sus propiedades; con un estímulo sensorial al usuario.

En el caso de este trabajo se pueden relacionar varios eventos del mundo real, como son la distancia a un objeto en relación al usuario (obstáculos), la orientación cardinal y la ubicación geográfica como características propias de la persona con su correspondiente feedback sensorial táctil por medio de actuadores (micromotores vibratorios) según corresponda (por ej. para indicarle a la persona que debe retroceder porque adelante hay un obstáculo)

4.2 Sistema háptico humano - Fisiología

El sistema somatosensorial se encarga de la recepción y procesamiento de información acerca de tacto, presión, dolor y temperatura. Los receptores implicados en la transducción de estas sensaciones son mecanorreceptores, estimulados por el desplazamiento mecánico de algún tejido del organismo; termorreceptores, que detectan calor y frío y nociceptores que se activan por cualquier factor que dañe los tejidos localizados por la superficie de todo el cuerpo.

El tacto está considerado uno de los cinco sentidos, sin embargo no debe ser tomado como una unidad, sino como la combinación de varios mecanismos. Es por esta razón que el termino “tacto” es reemplazado por “sentidos somáticos”, dando idea de la variedad en sus mecanismos y formas.

Mecanorreceptores. El sistema reacciona a los estímulos usando diferentes receptores. Para el área de estudio de esta tesis, se abordan particularmente los mecanorreceptores, los cuales son neuronas secundarias que responden a estímulos mecánicos disparando potenciales de acción. Su función principal es reaccionar ante la presión mecánica o las distorsiones. Los **corpúsculos de Pacini** y los de **Meissner**, presentes en la piel, son los más importantes en la detección de las vibraciones. La sensación táctil está determinada por la cercanía de los mecanorreceptores entre sí y el tamaño del campo receptivo. (Roger W. Cholewiak, 2013)[32]

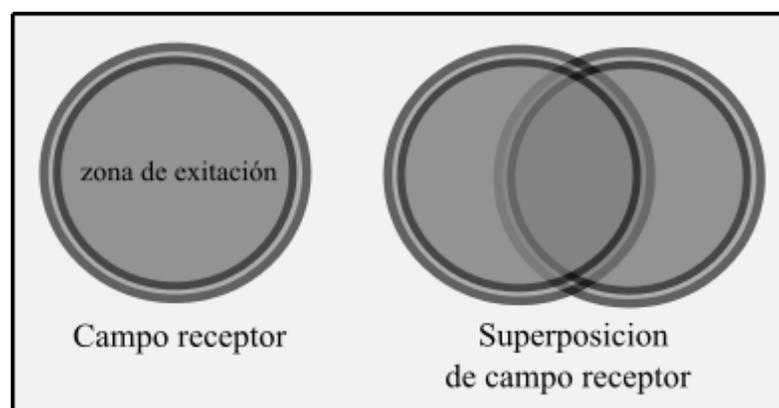


FIGURA 4.2: Superposición de campo receptor

El umbral mínimo entre dos puntos está definido según el tamaño de los campos receptivos y su solapamiento. Porciones adyacentes de superficie de piel están representadas por porciones adyacentes de la corteza cerebral, formando un mapa somatosensorial. Cuanto mayor es el área de representación en el cerebro, mayor es la precisión y el detalle de la percepción.

La transmisión de información desde los receptores viaja por vías nerviosas hacia la médula espinal y finalmente a la corteza cerebral, donde se encuentran áreas especialmente atribuidas a la superficie del cuerpo que ha recibido el estímulo.

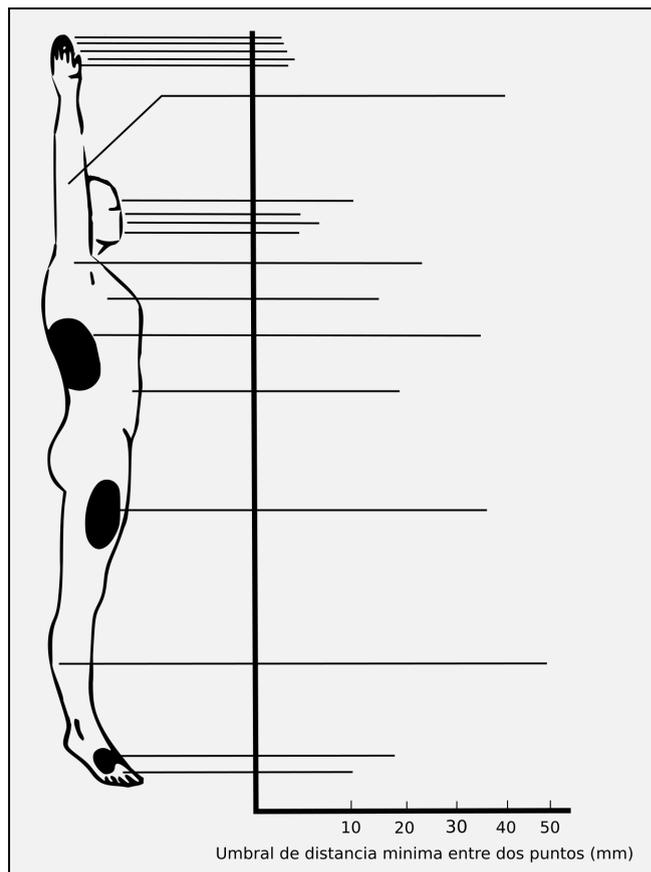


FIGURA 4.3: Umbral de distancia(milímetros) para diferenciar dos puntos

(Gardner,Martin,Jessell,2012)[7]

varían de un lugar a otro en el cuerpo de una persona, entre personas, y a través de todas las edades, desde los bebés hasta los ancianos. Las manos callosas de un obrero son diferentes a las de un trabajador de oficina. Las lesiones producen tejido cicatricial. Todos estos factores implican que existe una posibilidad de variación en cómo se transmiten las señales mecánicas desde la superficie de la piel a los receptores más profundamente alojados. Los receptores de los discos de Merkel responden a frecuencias extremadamente bajas, entre 5 y 15 Hz. Los corpúsculos de Meissner son más sensitivos a estimulación mediana, entre 20 y 50 Hz. Los corpúsculos de Pacinian tienen los umbrales más bajos para la reacción a estímulos (muy sensibles) para altas frecuencias que van desde los 60 a 400 Hz. Se puede afirmar que somos especialmente sensibles a vibraciones en el rango de 200 a 250 Hz.

El concepto de magnificación cortical se puede definir como una desproporción entre las zonas táctiles representadas en el cortex y su superficie de la piel. Hay zonas de mayor tamaño de representación en relación a otras, y a su superficie en la piel. Las zonas de mayor amplificación cortical son labios, nariz, lengua y dedos de la mano.

Vibraciones. Dado que los sucesos táctiles que entran en contacto con la piel tienen que pasar a través de un medio viscoelástico de varias capas (formado por la epidermis, la dermis, los músculos, la grasa, entre otros) para llegar hasta los receptores, la definición de estímulo es una particularmente difícil. Las características de la piel en sí

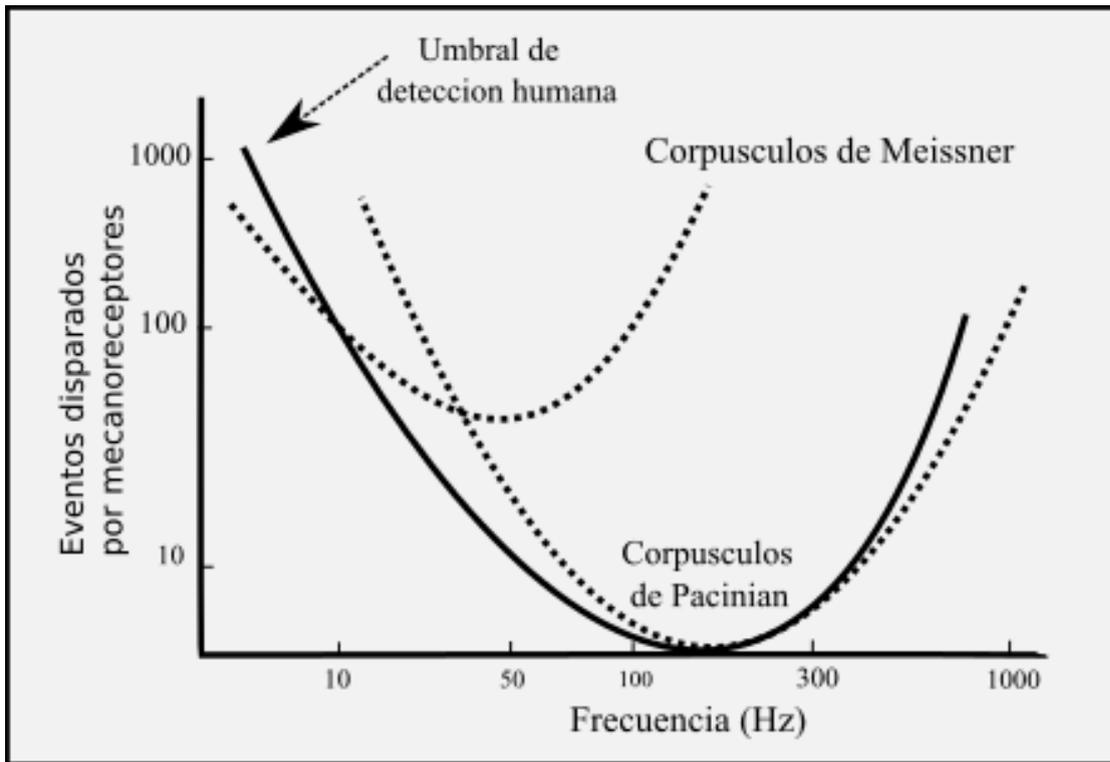


FIGURA 4.4: Relación entre estímulo y respuesta ([Gardner,Martin,Jessell,2012](#))[7]

A medida que envejecemos, la sensibilidad global disminuye de tal modo que hacia los 70 años de edad la sensibilidad de alta frecuencia podría haberse deteriorado hasta en un factor de treinta. La sensibilidad en los labios o los dedos a la vibración puede ser de 20 a 30 veces mejor que en lugares del cuerpo menos sensibles como el muslo o la pantorrilla de la pierna. La explicación de estas diferencias entre distintos lugares del cuerpo se debe a la variabilidad en la densidad de los receptores, a los tipos de receptores presentes, y a las correlativas representaciones corticales de éstos.

Electrofisiológicamente, el sistema nervioso responde a cambios en los estímulos mecánicos de formas complejas. Si es sólo un toque (una presión en la piel que dure varios cientos de milisegundos o más), la mayoría de los receptores dan una respuesta altamente explosiva en el inicio. Dependiendo de cual, un tipo de receptores se calma en decenas de milisegundos, mientras que otro tipo se mantiene disparando impulsos, aunque a una tasa más baja, hasta el final del contacto. Y, para ambos tipos, cuanto más fuerte sea la presión, mayor será el número de respuestas neurales. Si el estímulo es una vibración ambos tipos de receptores responden en las intensidades de su umbral de estímulo.

A medida que el estímulo se hace más fuerte, se produce un fenómeno llamado arrastre. En él aparecen las respuestas con los ciclos (con cada “presión” en la piel), de modo que

hay una respuesta a cada ciclo (a un estímulo de 250 Hz le corresponden 250 respuestas por segundo). A medida que la intensidad del estímulo continúa aumentando, se producen respuestas adicionales, pero siguen estando vinculadas a cada ciclo, de modo que se pueden registrar dobletes o tripletes en las respuestas. Los diferentes tipos de receptores responden mejor a lo largo de los intervalos más pequeños dentro del rango que se extiende entre los 20 y los 350 Hz de la sensibilidad humana vibrotáctil, y requieren diferentes desplazamientos mínimos para evocar una respuesta neural (deben sobrepasar el umbral de intensidad), actuando conjuntamente en concierto para cubrir el rango descrito. (Roger W. Cholewiak,2013)[32]

Fenómeno de adaptación. Las cosas que no cambian, por lo general, no son muy importantes para nuestra supervivencia, por lo tanto no son interesantes para el sistema nervioso. En consecuencia, si un estímulo visual, olfativo, o incluso táctil, no cambia durante un período de tiempo tan corto como un segundo o dos, comienza a producirse un fenómeno conocido como adaptación. La intensidad del estímulo que puede excitar un canal o un receptor adaptado (su umbral) se eleva, a veces hasta el punto de que el canal es completamente insensible a cualquier nivel normal del estímulo. Un ejemplo de esto en la vida cotidiana es la adaptación de los receptores del ojo. Los niveles de luz durante el día son tan fuertes que los bastones se adaptan, lo que lleva a nuestra falta de sensibilidad cuando entramos en una sala oscura.

El lento retorno de los bastones a su sensibilidad, aproximadamente en 20 minutos, se conoce como recuperación de la adaptación. Del mismo modo, la ropa que llevamos puesta produce una presión constante sobre el cuerpo, que no es apreciada luego de su colocación, hasta que nos movemos, y entonces la interfaz entre la piel y la tela cambia y sentimos que algo sucede. La adaptación puede darse incluso con niveles estables de vibración. Esto es debido a que los canales del receptor a menudo manifiestan su independencia de canal: es posible, por ejemplo, adaptar los corpúsculos de Pacini, en la yema del dedo, sin afectar a los receptores de otros canales.

Hay otro fenómeno asociado, llamado habituación. Éste es de naturaleza cognitiva, y en el cual llegamos a no prestar atención a un estímulo repetitivo regular. Un ejemplo común es nuestra habituación al tic-tac de un reloj. Pero si se atrae nuestra atención sobre él, el sonido es obvio. Sin embargo, ninguna cantidad de atención puede “despertar” un canal sensorial adaptado como en el caso de la luz o de una adaptación de vibración regular.

4.3 Aplicabilidad al Calzado Háptico

En la Figura 4.5 se muestra a modo ilustrativo un esquema del diseño del calzado háptico con sus correspondientes descripciones generales. Los detalles de implementación en cuanto a software y hardware, se mencionan en [capítulo 7](#) y [capítulo 8](#), respectivamente.

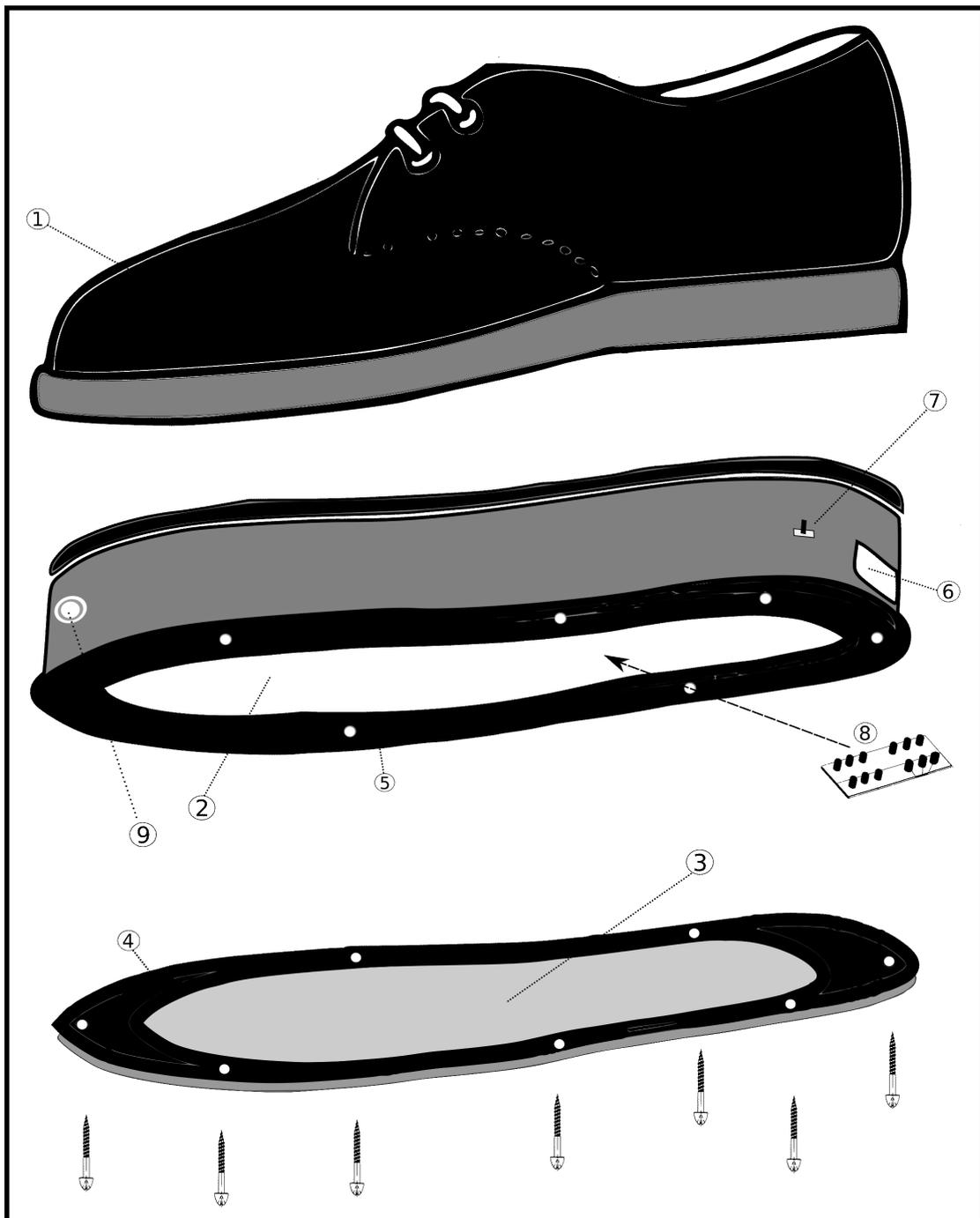


FIGURA 4.5: Esquema de diseño del calzado háptico

1. Vista exterior del calzado
2. Zona hueca de protección (2 cm) donde se ubican todos los módulos de hardware.
3. Suela que cumple la función de tapa inferior, la cual está en contacto permanente con la superficie del suelo. Su montaje es por medio de tornillos medianos.
4. Borde de apoyo perimetral de 1.5 cm de espesor, aumentado la zona hueca mencionada en el punto 2.
5. Borde de apoyo perimetral de 1.5 cm de espesor, que se encuentra en contacto con el borde descrito en el punto 4, juntos logran la cavidad.
6. Puerto para cargar las baterías.
7. Interruptor de encendido/apagado/carga.
8. Módulos de Hardware.
9. Sensor de obstáculos.

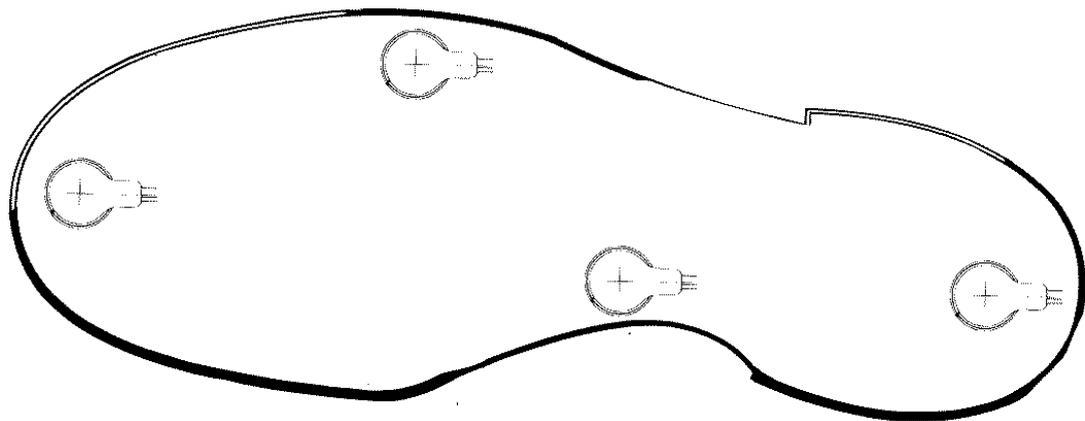


FIGURA 4.6: Plantilla con motores vibratorios

En el interior del zapato se encuentra una plantilla que contiene 4 micromotores vibratorios encargados del feedback háptico (detalles en los capítulos siguientes).

Como producto de la información estudiada ([sección 4.2](#)) y a las pruebas de usabilidad ([capítulo 7](#)) se decide trasladar los actuadores de la planta del pie hacia el tobillo.

Las razones de este cambio son la poca separación existente entre el motor que indica girar a la derecha y el motor que indica girar a la izquierda, haciendo difícil la tarea

de diferenciar donde se originan las vibraciones. También en el momento de la implementación se comprobó que la disposición de los motores en una plantilla requiere de mucha precisión, debido a que éstos deben ser protegidos para que el peso del cuerpo no los dañe. Se diseñó una plantilla sólida y se ubicaron en huecos de 4 mm, quedando los motores a nivel (subrelieve), pero manteniendo un mínimo contacto con la planta del pie. Se comprobó que el peso del cuerpo detiene la vibración de los motores y no se aprecian las vibraciones. Se probó protegiéndolos completamente del pie, sin contacto directo, pero las vibraciones se transmitían hacia toda la plantilla. Al trasladar el sistema al tobillo, mediante la construcción de una tobillera, la sensibilidad mejoró al ser un área de mayor distancia, y al no recaer el peso del cuerpo en los motores la vibración se mantiene fuerte. Los detalles técnicos referidos al hardware desarrollado se explican en la sección 6 (Módulos de Hardware).

En la Figura 4.7 se observa el esquema utilizado en el calzado háptico:

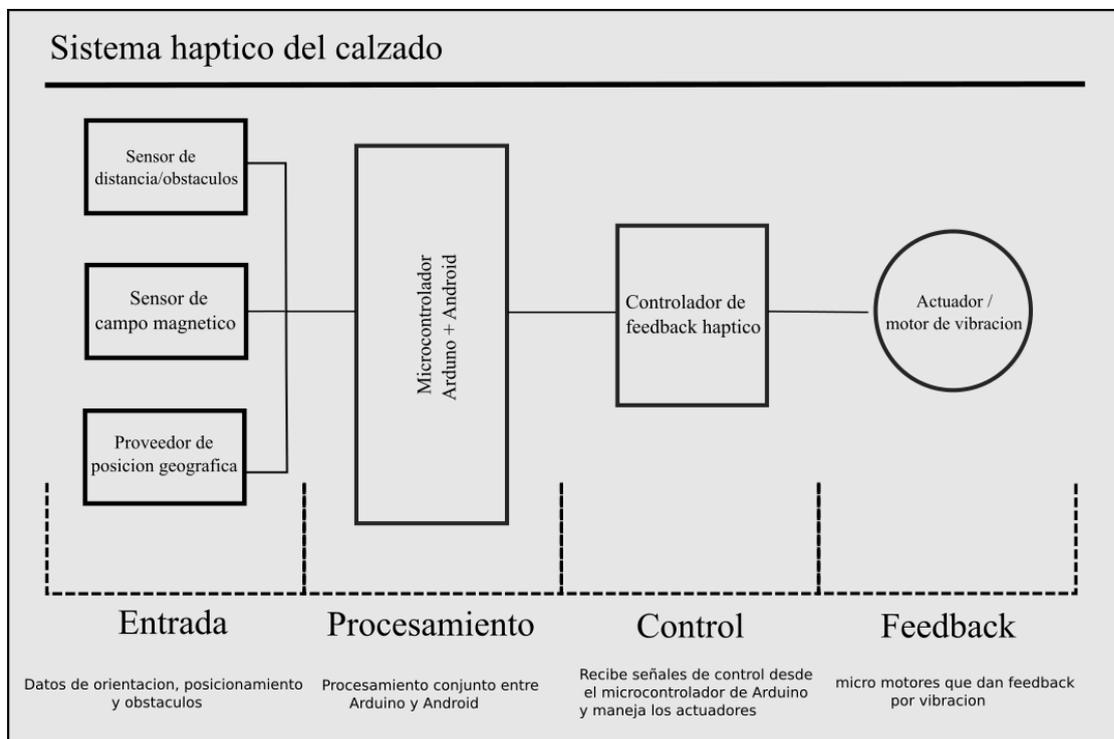


FIGURA 4.7: Esquema háptico aplicado al calzado

Capítulo 5

Módulos de Software en el Calzado Háptico

5.1 Módulo de reconocimiento de voz

Android provee, a partir de SDK nivel 8, mecanismos de acceso al servicio de reconocimiento de voz: en la mayoría de sus versiones el sistema operativo trae instalado Google Voice Search, el cual implementa la funcionalidad. En su implementación, la clase `SpeechRecognizer` envía el audio grabado a un servidor remoto de Google para realizar la tarea de reconocimiento de voz, devolviendo el resultado en forma de texto. Tiene como requerimiento la necesidad de una conexión a internet, sin embargo para algunos dispositivos en particular se permite que este proceso se realice de manera offline.

La funcionalidad se implementó haciendo uso de la clase `android.speech.SpeechRecognizer`

Tipo de retorno	Signatura de método
<code>SpeechRecognizer</code>	<code>createSpeechRecognizer(Context context)</code> Creación de un <code>SpeechRecognizer</code>
<code>void</code>	<code>setRecognitionListener(RecognitionListener listener)</code> Asignación de listener que recibe callbacks
<code>void</code>	<code>startListening(Intent recognizerIntent)</code> Inicia la escucha de voz
<code>abstract void</code>	<code>onResults(Bundle results)</code> callback para cuando el resultado está listo

TABLA 5.1: Métodos destacados(Reconocimiento de Voz)

La clase `android.speech.RecognitionListener` es usada para recibir notificaciones de los eventos ocurridos como respuesta al reconocimiento. El servicio es muy rápido y preciso, mostrando un gran avance en el área en comparación a sistemas de años atrás.

5.2 Módulo de síntesis de voz a partir de texto (sustituido por TalkBack)

Android provee, a partir de la SDK nivel 4, mecanismos para soportar TTS (Text-to-Speech), lo que permite crear una voz humana sintetizada a partir de texto. Se necesita tener instalado un motor de síntesis de voz, que en este caso viene integrado con el sistema operativo; si bien también existe la posibilidad de instalar otros. Antes de crear una actividad que usa TTS es necesario chequear qué recursos están instalados, y proveer un método para descargar los recursos faltantes. Algunos dispositivos tienen poco espacio de almacenamiento, por lo cual es probable que no se encuentren los paquetes de idiomas necesarios. La clase `android.speech.tts.TextToSpeech` brinda soporte a TTS, usando un listener de inicialización para saber cuándo está listo el servicio para comenzar a recibir texto. La Tabla 5.2 describe los métodos destacados en la clase `android.speech.tts.TextToSpeech`.

Tipo de retorno	Signatura de método
<code>TextToSpeech</code>	<code>TextToSpeech (Context context, TextToSpeech.OnInitListener listener)</code> Constructor para la clase <code>TextToSpeech</code> , usando el motor TTS por defecto
<code>void</code>	<code>setLanguage(Locale loc)</code> Configuración del lenguaje
<code>void</code>	<code>setPitch(float pitch)</code> Configuración de la velocidad del habla.
<code>void</code>	<code>speak(String text, int queueMode, HashMap[String, String] params)</code> Método asincrónico que pronuncia el texto

TABLA 5.2: Métodos destacados(Síntesis de voz)

La aplicación se empezó a desarrollar para Android 2.3.6 API level 10, en donde el soporte de accesibilidad no contaba con la función de exploración por tacto, haciendo necesario implementar dentro de la aplicación dicha funcionalidad.

A partir de Android 4.1.2 API nivel 16, el soporte de accesibilidad mejoró, brindando de forma nativa exploración por tacto (Talkback) para todo el sistema operativo, dejando sin sentido la implementación propia del servicio, la cual, además, entraba en conflicto con funciones de Talkback nativo.

Al iniciar Talkback la aplicación hereda funcionalidades de lector de pantalla, devolviendo al usuario un feedback hablado según el contexto. Mediante este mecanismo es posible recibir notificaciones habladas al interactuar con ventanas que están activas en ese momento y producen eventos visuales. La exploración por tacto consiste en explorar con los dedos la pantalla, obteniendo feedback hablado correspondiente al elemento enfocado. Mediante un doble tap se ingresa al ítem explotado.

Durante el desarrollo de la funcionalidad del zapato háptico, se hizo la transición hacia Android 4.1.2 api nivel 16, dado el porcentaje que ocupa dentro de la fragmentación de las diferentes versiones de Android disponibles a la fecha. De la Figura 5.1 se desprende que el 50 % de los dispositivos Android activos a la fecha de acuerdo a datos recolectados por Google Play al 1/11/2014, son Android Jelly Bean. (Android,2014)[1]

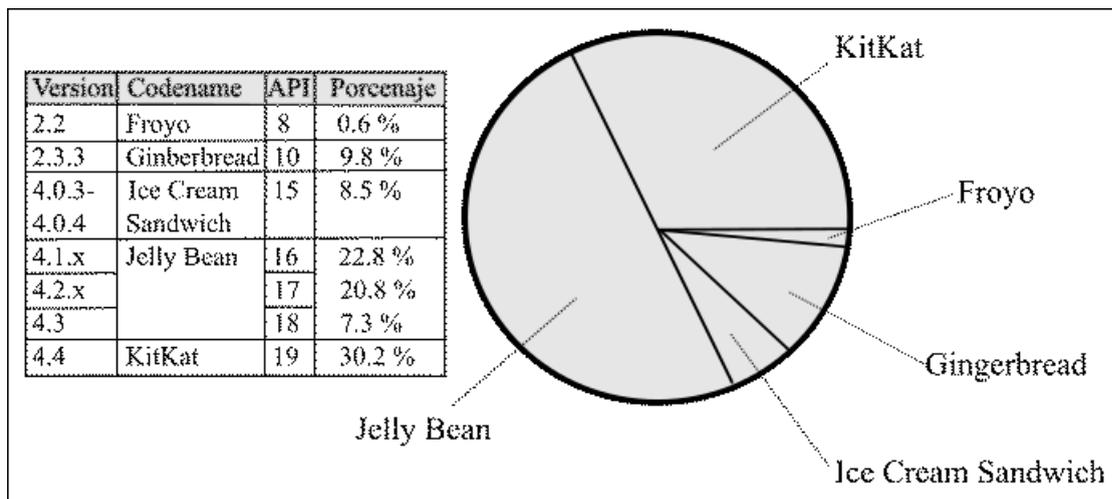


FIGURA 5.1: Fragmentación de versiones OS Android

5.3 Módulo de codificación geográfica

Como se describió en la [sección 3.2](#), la codificación geográfica es el proceso de transformar direcciones de calles u otra descripción textual de una ubicación en coordenadas geográficas expresadas en una latitud y una longitud.

Yahoo, mediante BOSS Geo Services y utilizando YQL (Yahoo Query Language), permite el acceso a su geocoder. Hay tres formas de especificación de direcciones: free-form en una línea, multilínea y fully-parsed. Free-form delega la interpretación al servicio server-side, produciendo errores de semántica.

Luego de probar las tres formas, se optó por el uso de fully-parsed, pudiendo describir la dirección en base a su construcción, siendo ésta: nombre de calle, altura, intersección con calle perpendicular, ciudad y país. (Yahoo,2014)[39]

Dirección: Calle 6 y Calle 59, La Plata, Argentina

Atributos : City: La Plata - Country: Argentina - Street: Calle 6 - xStreet: Calle 59

Construcción en YQL del requerimiento:

```
Select * from geo.placefinder where city="La Plata" and
country="Argentina" and street="calle 6" and xstreet="Calle 59"
```

Acceso a la tabla tabla geo.placefinder de YQL Open Data, mediante consola web.

Requerimiento web:

```
https://developer.yahoo.com/yql/console/#h=select**from+
geo.placefinder+where+city%3D%22La+Plata%22+and+country
%3D%22Argentina%22+and+street%3D%22calle+6%22+and+
xstreet%3D%22Calle+59%22
```

Fragmento del resultado en JSON:

```
<Result>
  <quality>82</quality>
  <latitude>-34.919041</latitude>
  <longitude>-57.941688</longitude>
  <street>Calle 6</street>
  <xstreet>Calle 59</xstreet>
</Result>
```

La clase `com.calzadoHaptico.geocoding.Geocoding`, implementa toda la funcionalidad, soportando el ingreso de direcciones en formato de intersección (Calle 6 y Calle 60) y formato de altura (Calle 6 número 1500).

Dada la naturaleza asincrónica en la implementación de los servicios, también se provee la clase `com.calzadoHaptico.geocoding.ListenerGeocoder` resumida en la tabla 5.3.

Tipo de retorno	Signatura de método
Geocoder	GeocoderYQL (ListenerGeocoder listenerGeocoder)
android.location.Address	request (String locationAddress)

TABLA 5.3: Métodos destacados(Codificación Geográfica)

5.4 Módulo de Ruteo

El servicio de routing de la plataforma MapQuest permite obtener una ruta entre dos o más puntos, generando un resultado en formato JSON. Se usó la clase MapQuestRoadManager de la librería Osmdroid(Osmdroid,2014)[23] para la construcción de la url (parámetros origen, destino, etc.) y el retorno de los resultados (ruta).

Construcción del requerimiento:

```
http://open.mapquestapi.com/guidance/v1/route?key=key&from=origen
&to=destino&routeType=pedestrian
```

Especificación de parámetros:

Parámetros	Descripción
from Location(origen)	Ubicación del origen de la ruta en formato (latitud,longitud)
to Location(destino)	Ubicación del destino de la ruta en formato (latitud,longitud)
routeType String	<p>Especifica el tipo de ruta a calcular Valores permitidos:</p> <ul style="list-style-type: none"> • fastest - Ruta de manejo en auto más rápida teniendo en cuenta el tiempo • shortest - Ruta más corta manejando un auto. • pedestrian - Ruta más corta en recorrido a pie. • bicycle - Ruta mas corta en bicicleta. • multimodal - Combinación de recorrido a pie y si está disponible transporte público

TABLA 5.4: Especificación de parámetros para ruteo

Resultados:

```
<shapePoints>
    <lat>40.2562</lat>
    <lng>-76.5095</lng>
    <lat>40.2561</lat>
    <lng>-76.5098</lng>
    ...
</shapePoints>
```

shapePoints. Conjunto de puntos (latitud,longitud) de la ruta calculada en orden de recorrido. Para acceder a un punto específico se usa $i/2$.

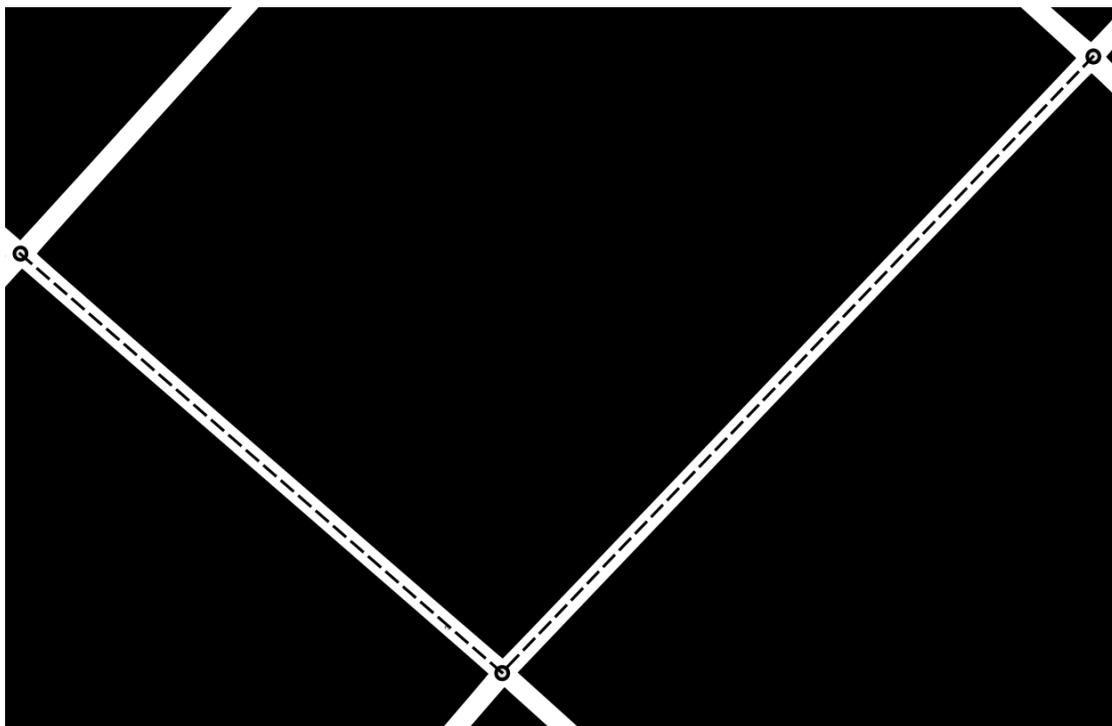


FIGURA 5.2: Ruta con tramos de 100 metros

Se implementó la clase `com.calzadoHaptico.routing.ServicioDeRuta` que realiza todas las tareas relacionadas con resolver la ruta. Se utilizaron mecanismos de concurrencia para no bloquear la ejecución de otras tareas durante la espera del resultado.

Tipo de retorno	Signatura de método
void	<code>calcularRuta (ArrayList[GeoPoint] puntos)</code>
void	<code>setmRoad (Road result) de InterfaceServicioRutaListener</code>

TABLA 5.5: Métodos destacados(Ruteo)

5.5 Módulo de optimización de ruta para navegabilidad en tiempo real

Se toma la ruta calculada en [sección 5.4](#) y se le realiza una optimización, asegurando que no existan tramos mayores a una determinada distancia. Se tomó como valor 50 metros: en el caso de que algún tramo sobrepase los 50 metros, se divide su longitud a la mitad. Este ajuste permite que el sistema de navegabilidad explicado en la [sección 5.6](#) sea más preciso, teniendo mayor cantidad de puntos de control y a menor distancia.

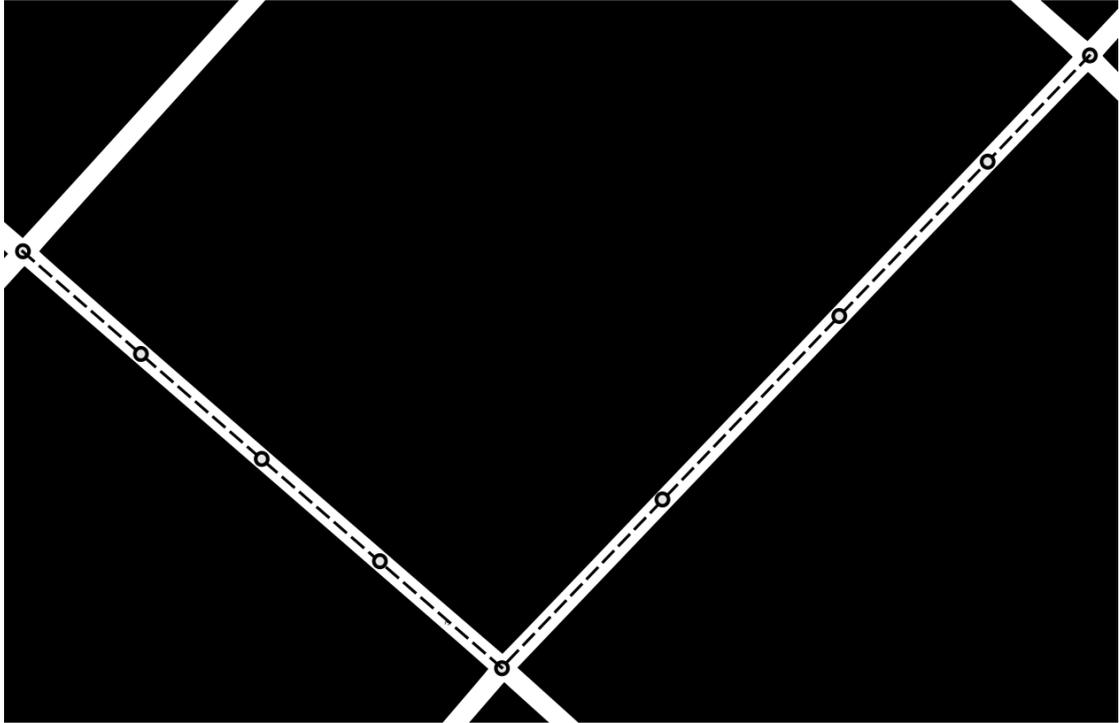


FIGURA 5.3: Ruta con tramos de 25 metros

Tipo de retorno	Signatura de método
Road	optimizarRuta(Road ruta)

TABLA 5.6: Métodos destacados(Optimización de ruta)

5.6 Módulo de navegabilidad en tiempo real

La implementación del algoritmo de navegabilidad consta de los siguiente pasos:

1. Se toma la ruta obtenida en la [sección 5.5](#) y para cada nodo de la ruta se calculan dos puntos guía auxiliares.

Siendo:

P0 = ubicación coordenada de la persona.

P1 = ubicación del siguiente nodo en ruta.

P2 = ubicación de punto guía auxiliar.

P3 = ubicación de punto guía auxiliar.

T1 = Recta que pasa por los puntos P0 y P1.

Ubicación de P2 = + 90 grados en relación a P1.

Ubicación de P3 = - 90 grados en relación a P1.

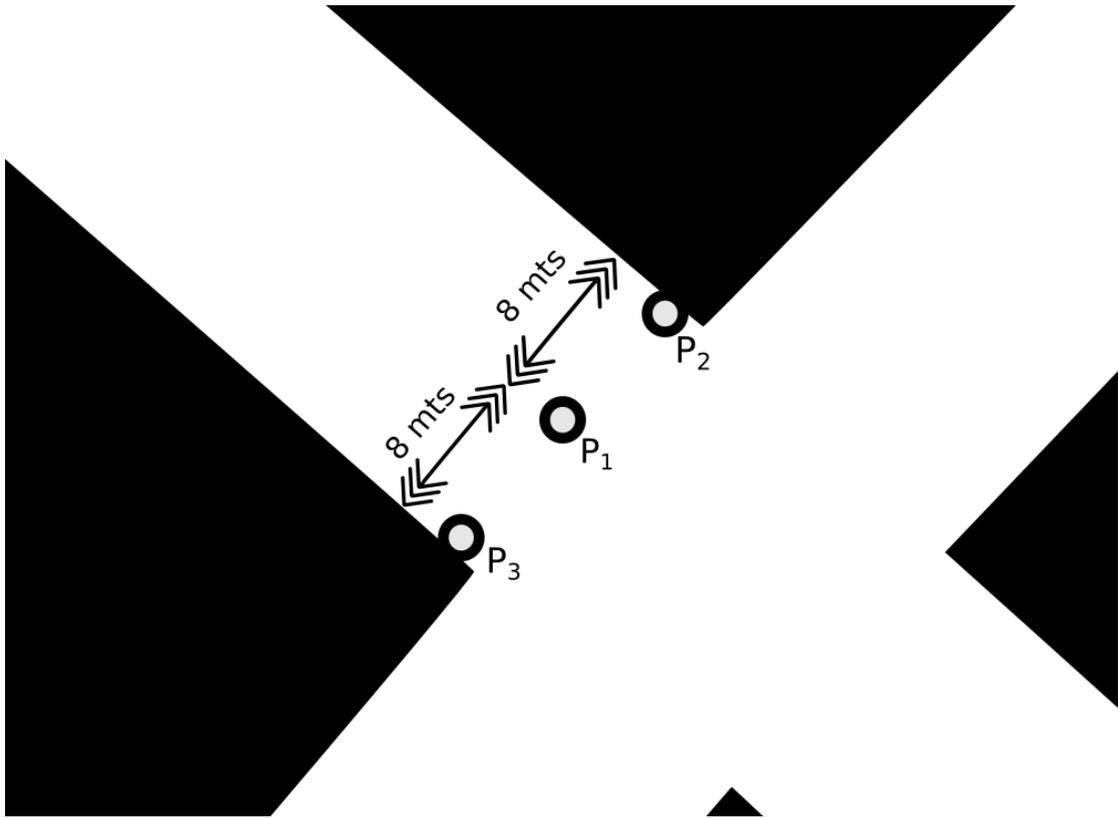


FIGURA 5.4: Cálculo de puntos guía

2. Se calcula el Bearing desde P0 hacia los puntos auxiliares.
 - B2 = Bearing desde P0 hacia P2
 - B3 = Bearing desde P0 hacia P3
 B2 y B3 son considerados los ángulos límites de alineación, izquierdo y derecho, respectivamente.
3. Se calcula el bearing de la persona, en función del norte magnético y se convierte a coordenadas respecto al norte geográfico. Ver Figura 5.5.
 - B0 = Bearing de la persona.
4. Si B0 se encuentra entre B2 y B3 (ángulos límites), implica que está alineado en su camino, en caso contrario está fuera del camino y debe girar. Ver Figura 5.6.
5. Si ha tocado o sobrepasado el límite derecho, debe girar hacia la izquierda. Si ha tocado o sobrepasado el límite izquierdo, debe girar hacia la derecha. Ver Figura 5.7 y Figura 5.8.

Vale aclarar que se hacen conversiones especiales y se toma en cuenta casos particulares relacionados con la matemática de los ángulos: si la orientación de la persona está más cercana a un límite que al otro, se la hace girar en el sentido más cercano y se evita que rote sobre su eje en un ángulo mayor.

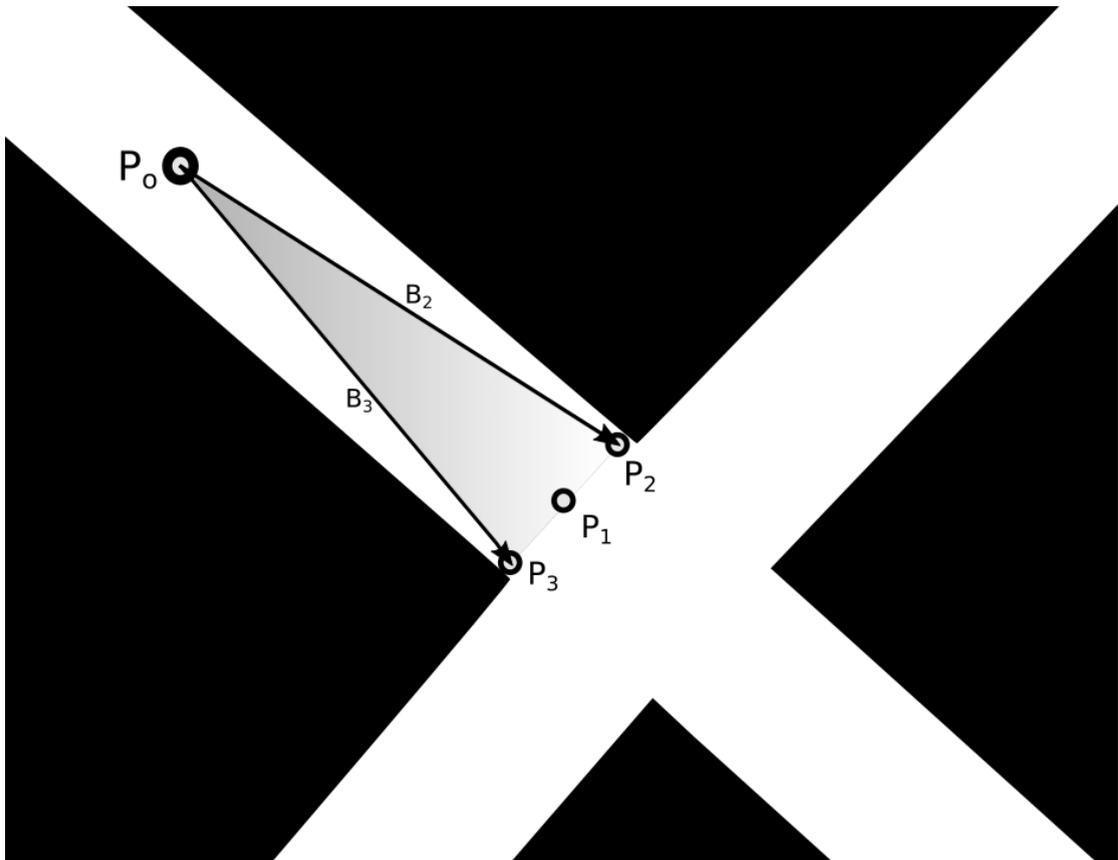


FIGURA 5.5: Cálculo de ángulos límite

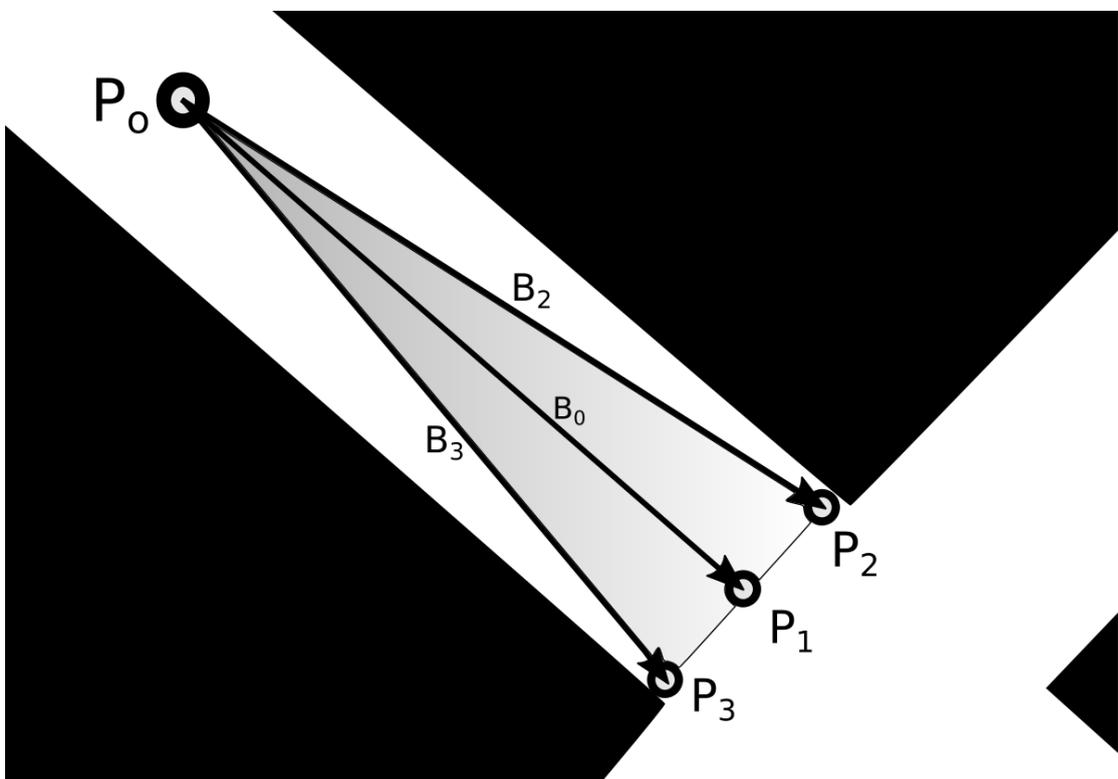


FIGURA 5.6: Alineación correcta dentro de límites

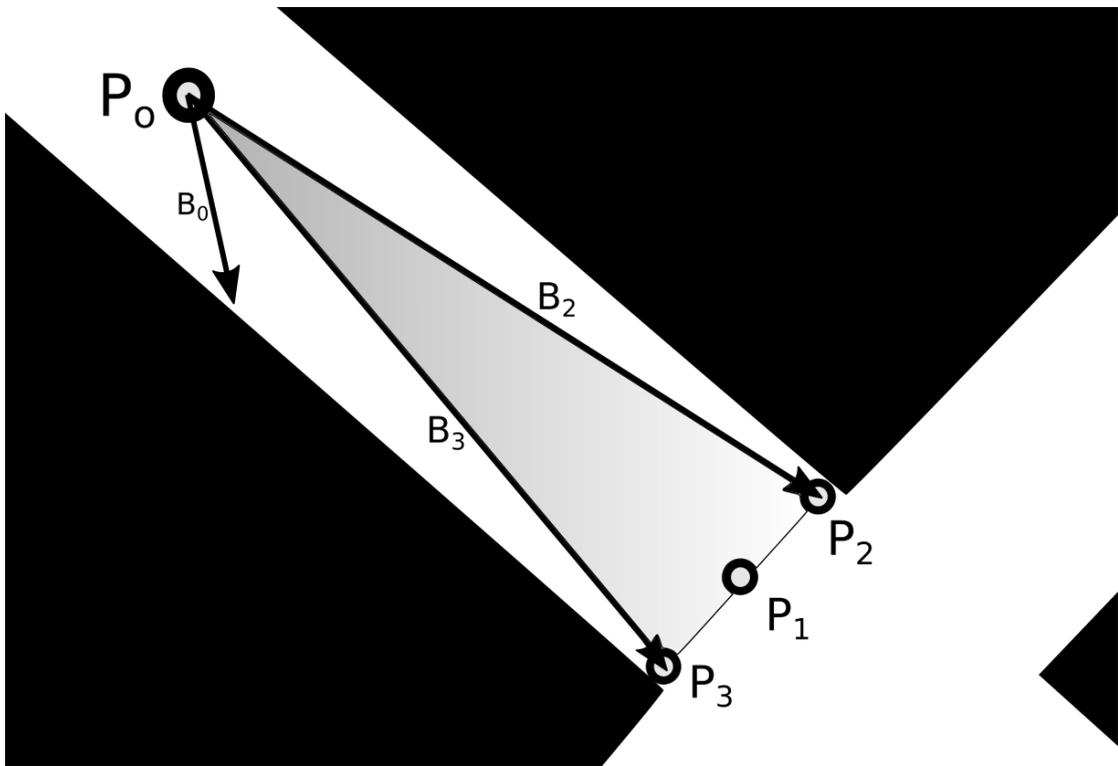


FIGURA 5.7: Límite derecho sobrepasado

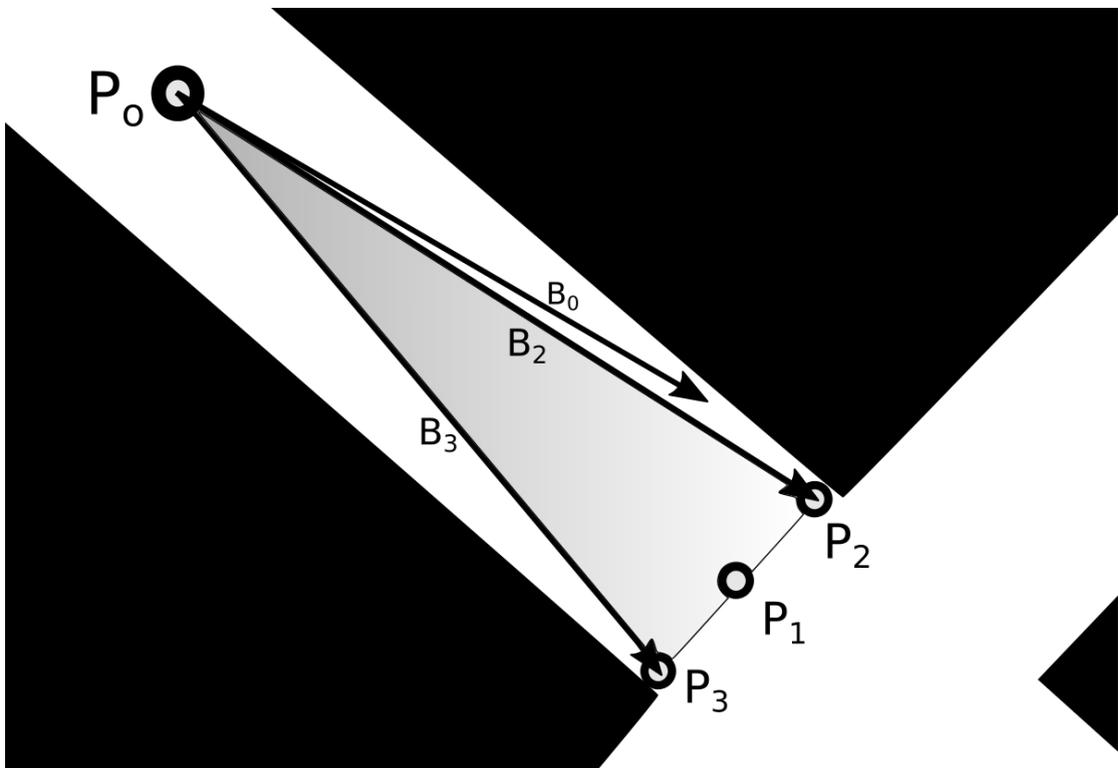


FIGURA 5.8: Límite izquierdo sobrepasado

6. Cuando la persona se encuentra a una distancia determinada del nodo, fijada en 8 metros, se actualiza al siguiente nodo.
7. Se repite todo el algoritmo con el nuevo nodo, hasta llegar al último nodo destino.

El algoritmo de navegación ajusta los límites óptimos permitidos, recalculando los ángulos dinámicamente, al orden de 2 veces por segundo. De esta manera, a medida que la persona está más cerca de un punto de chequeo, el ángulo permitido se va ampliando. Se puede observar en la figura Figura 5.9.

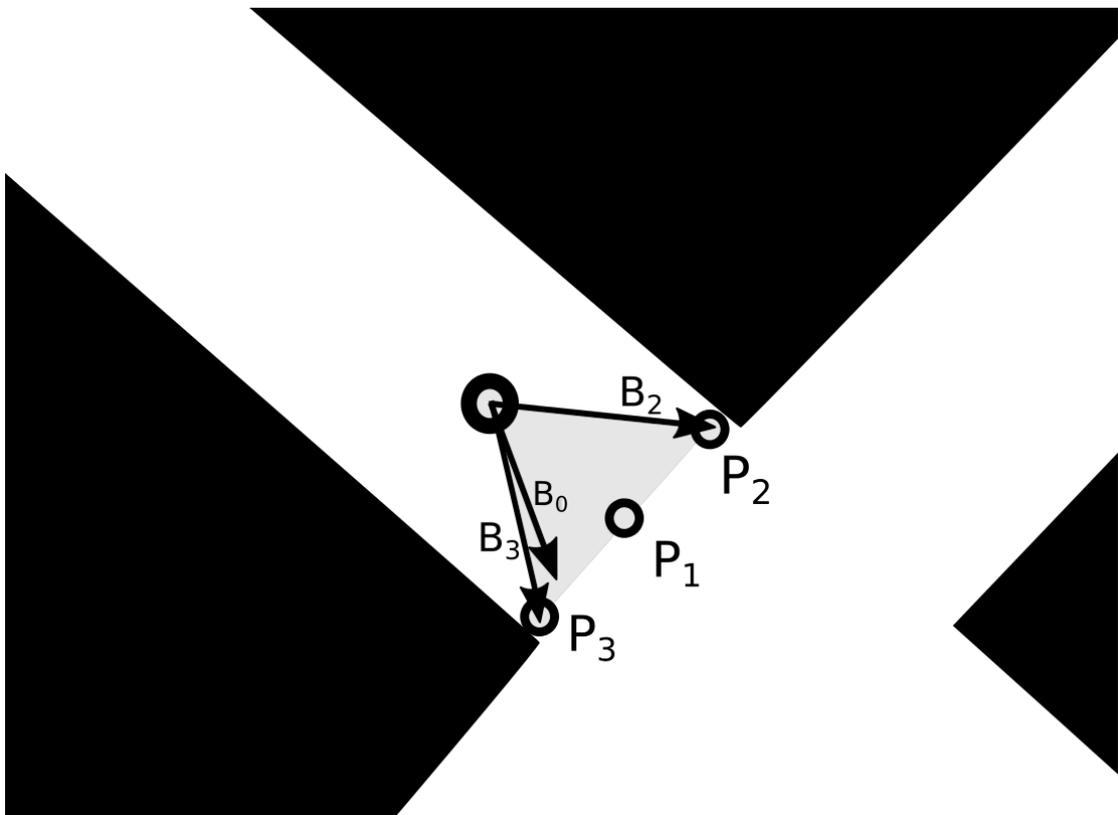


FIGURA 5.9: Ángulos límite ampliados dinámicamente

La clase `com.calzadoHaptico.navegabilidad.Navegacion` implementa toda la lógica mencionada, sus métodos destacados se observan en la Tabla 5.7.

Tipo de retorno	Signatura de método
Road	<code>optimizarConPuntosGuia(Road ruta) de routing.ServicioDeRuta</code>
void	<code>Navegacion(Calzado interfazCalzado, Activity context, MyRunnable run)</code>
void	<code>iniciarNavegacion()</code>
void	<code>detenerNavegacion()</code>

TABLA 5.7: Métodos destacados(Optimización con puntos guía)

Capítulo 6

Módulos de Hardware del Calzado Háptico

6.1 Módulo Microcontrolador

6.1.1 Arduino Uno R3

Al inicio del desarrollo se utilizó la placa microcontroladora Arduino Uno R3.

Ventajas:

- Programación directa desde una computadora, a través de su puerto usb.
- Pines hembra incorporados que facilitan las conexiones sin necesidad de soldar.
- Posibilidad de alimentarlo en su jack externo o por usb
- Posibilidad de reemplazar el chip microcontrolador atmega328p, gracias al zócalo para circuito integrado.

El software que se implementó en el microcontrolador cumple con:

- Recibir comandos a través del módulo bluetooth y realizar diferentes acciones según corresponda.
- Protocolo de sincronización de comandos con el calzado háptico, para evitar conflictos con las diferentes velocidades de procesamiento entre calzado háptico y el dispositivo móvil.

- Uso de librería RTOS ChibiOS/RT, para sistemas embebidos que requieren respuestas en tiempo real, agregando capacidad multithread al entorno, permitiendo una excelente ejecución de tareas.
- Uso de Timers no bloqueantes, mejorando la velocidad de respuesta y evitando problemas de sincronización
- Lógica de manejo de micromotores vibratorios, en conjunto con el módulo de detección de obstáculos.

Uso de pines en Arduino Uno:

- Los pines TX Y RX se usan para la comunicación con el módulo bluetooth.
- Los pines 3.3V y GND se usan para alimentar el módulo bluetooth.
- Los pines 10 y 11 transmiten Ting y Echo al módulo detector de obstáculos.
- Los pines 5v y GND se usan para alimentar el módulo detector de obstáculos.
- Los pines 9,6,5 y 3 habilitan o deshabilitan los actuadores (motores vibratorios).

6.1.2 Arduino Pro Mini

Como se mencionó en la [sección 6.1.1](#), se utilizó un Arduino Uno para desarrollar debido a sus facilidades; sin embargo, una vez finalizada la parte de programación del microcontrolador, se portó el prototipo hacia un Arduino Pro Mini (5v).

Por razones de espacio físico, es necesario reducir el tamaño del microcontrolador: Arduino Pro Mini cumple con el tamaño requerido, ambas placas funcionan con 5v, lo que lo hace compatible y minimiza las adaptaciones necesarias. La ganancia de espacio se debe a la ausencia de un puerto usb para poder programarlo, de un jack de alimentación externo y del atmega328 en su versión de montaje superficial (sin zócalo superficial para circuito integrado en la placa).

La cantidad de pines de entrada/salida son suficientes para el desarrollo. Al no contar con pines de salida de 3.3V, fue necesario obtener la alimentación para el módulo bluetooth de otra forma, detallada en la [sección 6.7](#) (interconexión de módulos).

Debido a la ausencia de un puerto usb, es necesario usar un placa FTDI para la programación del microcontrolador. También se puede utilizar otro arduino como programador ISP para cargar los programas.

6.2 Módulo de feedback háptico

6.2.1 Micro-motores vibratorios

Los motores vibratorios ofrecen un feedback háptico al usuario por medio de vibraciones. Las dimensiones de estos micromotores son de 3 milímetros de alto, lo que los hace indicados para lugares de dimensiones reducidas, como es el caso de este calzado. Su voltaje nominal es de 3V y alcanzan 9000 rpm, produciendo una vibración intensa.



FIGURA 6.1: Micromotor vibratorio

6.2.2 Módulo controlador de micro-motores vibratorios

Los motores vibratorios no deben ser conectados de forma directa a un microcontrolador por sus propiedades eléctricas. Cuando una carga inductiva (motor) es encendida y apagada repetidamente, se generan picos de voltaje (spikes) que pueden interferir con el correcto funcionamiento del microcontrolador, produciendo reinicios o dañándolo de manera irreversible. Generalmente los picos de voltaje pueden ser atenuados agregando un diodo de tipo “fly-back” en paralelo con el motor. El diodo 1N4444, al ser polarizado, permite el paso de electricidad en una única dirección, cumpliendo la función de protección. Un capacitor de cerámica de 0.1 uF ayuda a compensar la inestabilidad en el voltaje causada por el ruido, actuando como buffer temporal. No se debe alimentar a los motores con los pines de voltajes de la placa Arduino, ya que se supera el consumo (40 mA) que éstos pueden dar. Se utilizó una fuente de alimentación externa y un transistor NPN darlington. Para aislar completamente el controlador del resto del circuito se puede usar un opto-acoplador, evitando que los picos de interferencia transitorios puedan llegar al microcontrolador por medio de las líneas de control. ([Precision Microdrivers,2014](#))[30]

El controlador puede manejar 4 motores de corriente continua; el encendido y apagado de cada motor se hace por medio del transistor 2n2222. El microcontrolador es utilizado como línea de control para saturar la base del transistor y permitir el pasaje de corriente entre el colector y el emisor.

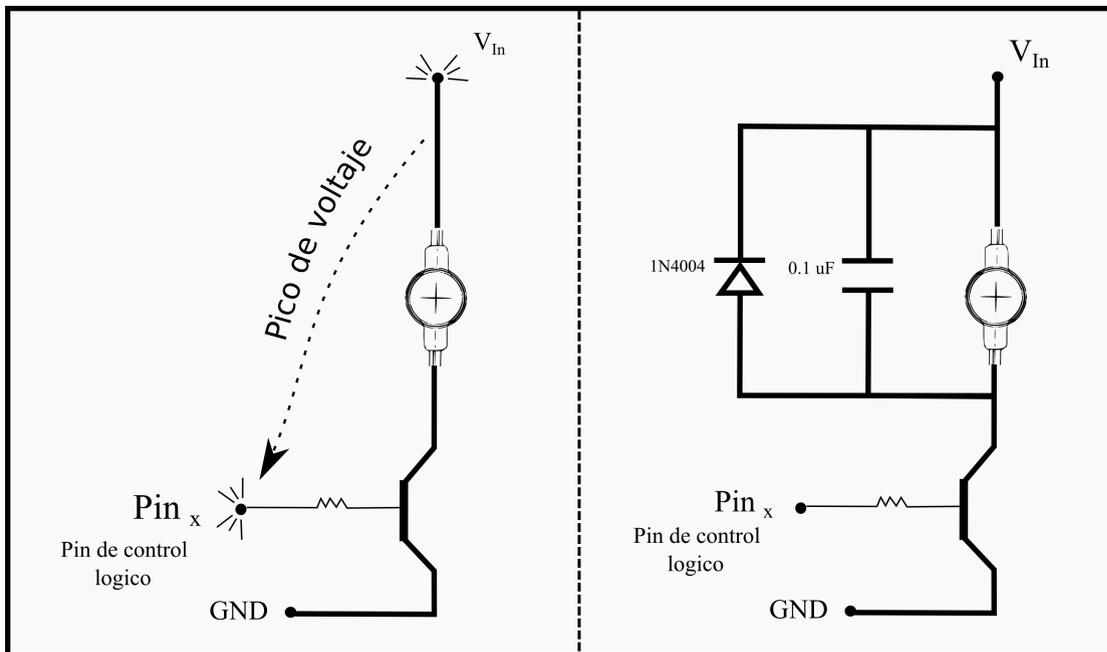


FIGURA 6.2: Protección contra picos de voltaje

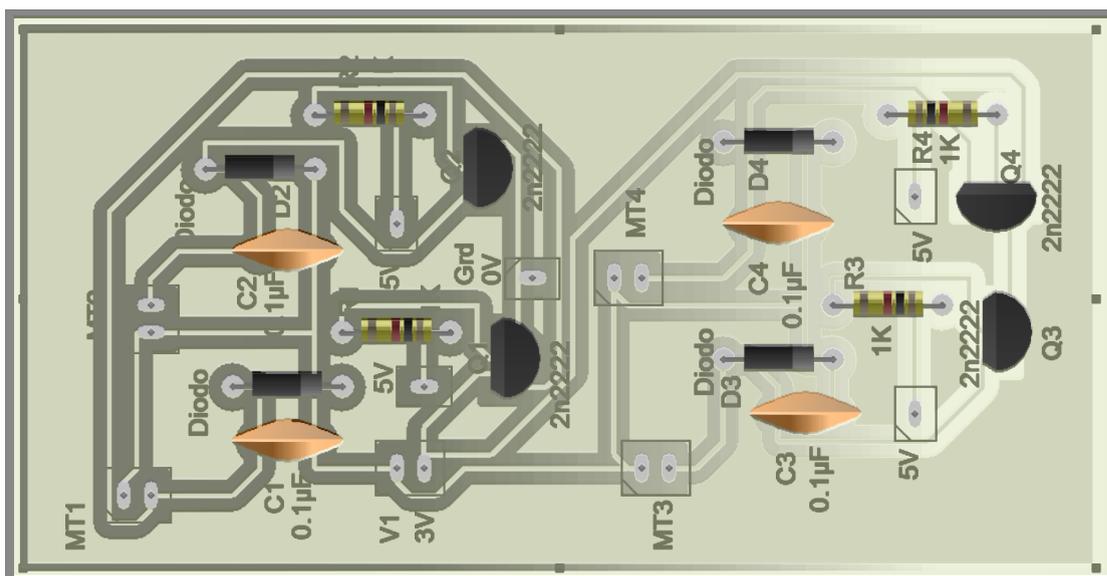


FIGURA 6.3: Circuito impreso controlador de motores

6.3 Módulo de detección de obstáculos

La detección de obstáculos se implementa mediante un sensor ultrasónico que permite medir distancias de hasta 5 metros con una resolución de 3 milímetros. No es necesario el contacto físico con el objeto a sensar y el consumo de energía es muy bajo. Dado que el usuario va a estar expuesto al aire libre, es importante que se detecten la mayor variedad de objetos posibles. Es deseable que la superficie del objeto tenga al menos 0.5 m² para un mejor rendimiento. En cuanto al color, este sensor tiene la ventaja de que su funcionamiento no se ve afectado por el color del objeto, a diferencia de lo que ocurre con los sensores infrarrojos.

El módulo está formado por un transductor que emite una ráfaga de 8 pulsos ultrasónicos, inaudible para el oído humano. Cuando el pulso rebota con un objeto, el sensor captura el eco producido por medio de un receptor y luego de ser procesado se obtiene la distancia al objeto. Dicho procesamiento se lleva a cabo en la placa arduino.

Fórmula para calcular la distancia:

$$d = 1/2v.t$$

d = distancia

v = velocidad del sonido (343 m/s)

t = tiempo

Se debe dividir por 2, la longitud del pulso incluye el tiempo de regreso.

Pines de conexión:

- VCC: Alimentación +5V (4.5V min – 5.5V max)
- TRIG: Trigger entrada (input) del sensor (TTL)
- ECHO: Echo salida (output) del Sensor (TTL)
- GND

Protocolo de funcionamiento:

1. Poner en high el pin Trigger (disparador) por al menos 10uS de tiempo.
2. El sensor envía una ráfaga de 8 pulsos de 40 KHz (ultrasonido) y coloca su salida Echo en high, inmediatamente se debe detectar este evento e iniciar un conteo de tiempo.
3. La salida Echo se mantendrá en high hasta recibir el eco reflejado por el obstáculo, el sensor pondrá el pin Echo en low, en este momento se debe detener el conteo de tiempo.
4. Se recomienda dar un tiempo de aproximadamente 50 ms de espera después de terminar la cuenta.

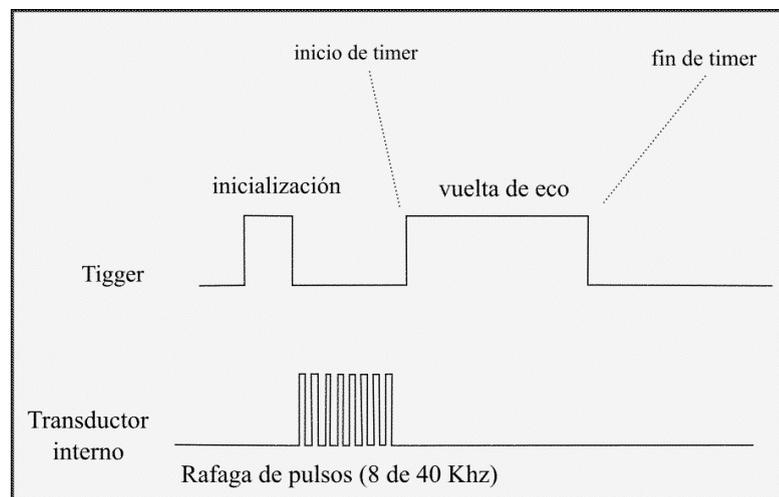


FIGURA 6.4: Protocolo de funcionamiento(Sensor de distancia)

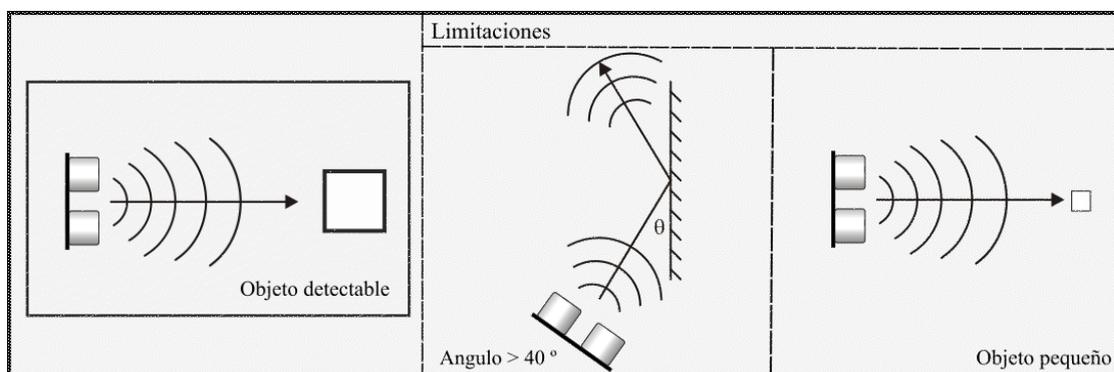


FIGURA 6.5: Limitaciones del sensor de distancia

6.4 Módulo de comunicación inalámbrica

Aprovechando la popularidad del protocolo bluetooth en los dispositivos móviles, se utiliza el módulo HC-06 para agregarle al microcontrolador capacidad de comunicación inalámbrica hasta una distancia de 10 metros. Los pines de datos lógicos TX y RX en este módulo trabajan con voltajes inferiores a los que entrega Arduino, por lo que fue necesario implementar un TTL de 5.5v a 3.3v por medio de un divisor de voltaje usando resistencias, método que sólo es recomendable para líneas de control lógico, evitando su uso en líneas con carga.

6.5 Módulo regulador de tensión

El módulo regulador de tensión, tiene la función de entregar voltaje regulado (3v) hacia el módulo controlador de motores, y una salida directa no regulada hacia la placa Arduino.

6.6 Módulo de energía

Puede ser alimentado por cualquier fuente de energía que esté entre 8v y 37v; el límite superior está dado por el regulador LM317T y el inferior por la placa arduino. En versiones tempranas de prototipado se utilizó un LM317T, debido a problemas de alta temperatura y la imposibilidad de agregar un disipador por las dimensiones de espacio físico en las que se trabaja, se reemplazó por una fuente switching step-down L2596 de alta eficiencia que se ajusta a las dimensiones y no presenta problemas térmicos para los 0.60 Watts que requiere el proyecto. Para el prototipo se utilizó alternativamente una batería 9v recargable (450 mha) , y se extrajo el conector hacia el exterior, para permitir la carga con cualquier cargador universal de baterías. También se desarrolló una versión que usa dos baterías Lipo 3.7V en serie, obteniendo 8.4 v como resultado. Para dicha opción, se cuenta con un cargador integrado con entrada mini-usb. Para ambas versiones, se usa un interruptor de 3 posiciones, definiendo el modo de operación: encendido, apagado, recarga de batería.

6.7 Interconexiones entre módulos

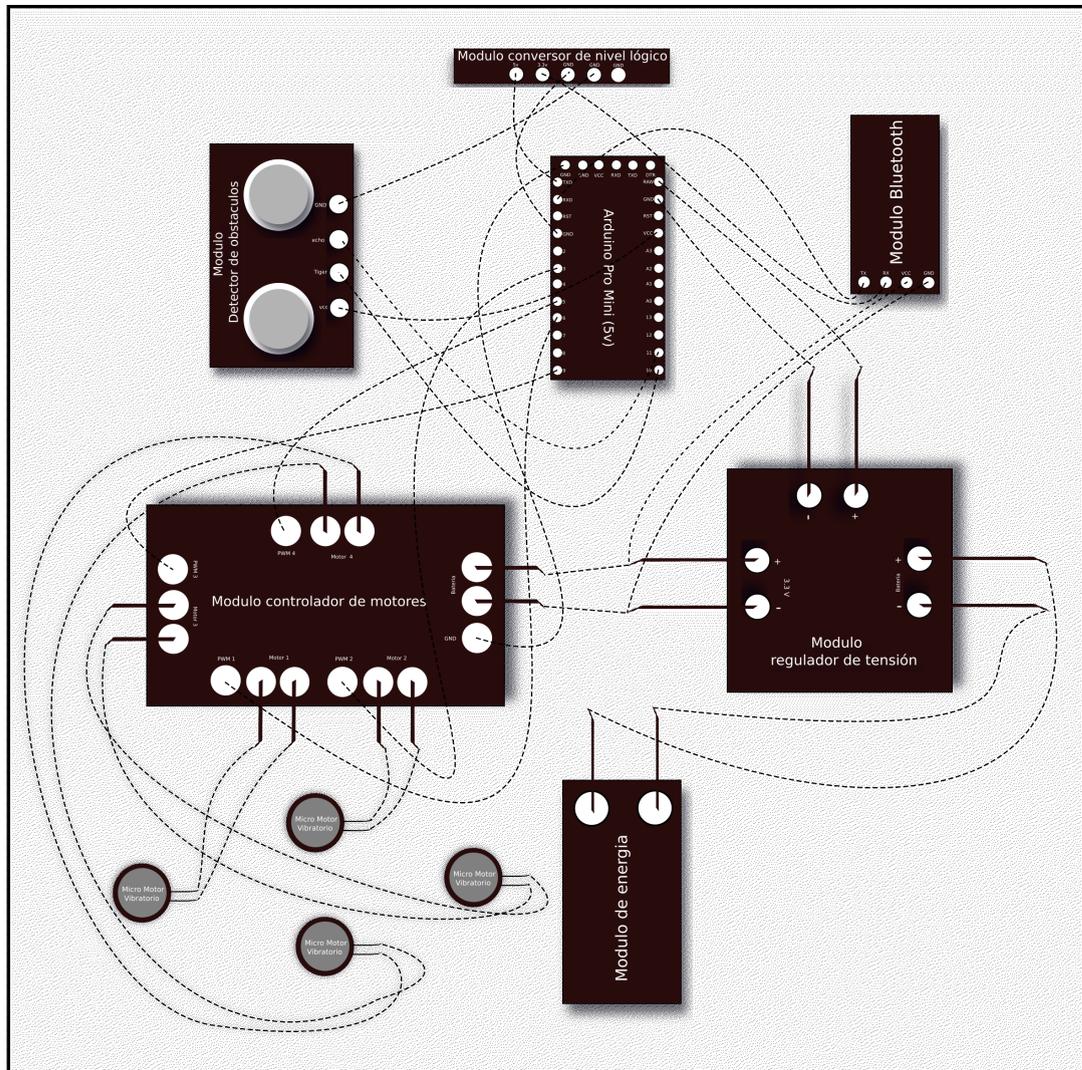


FIGURA 6.6: Esquema de interconexiones

Capítulo 7

Pruebas de usabilidad

7.1 Navegación

7.1.1 Variables de entrada de información

- Dirección Destino: coordenadas geográficas del destino final de la ruta.
- Localización de la persona(GPS) en formato (latitud,longitud).
- Orientación (Bearing) de la persona, expresada en grados con respecto al norte geográfico.

La localización y orientación varían a medida que se avanza en el recorrido.

7.1.2 Descripción del contexto.

Las pruebas se dividen en dos grupos; contexto controlado y contexto libre.

Contexto controlado:

Los valores de las variables de entrada son simulados por software, logrando independencia de los instrumentos de sensado, conforme a los siguientes términos.

La ruta es simulada por software, mediante un archivo .klm que contiene las coordenadas correspondientes del recorrido, un programa (Teleport, de uso gratuito) simula el recorrido leyendo las ubicaciones desde un archivo y las envía al dispositivo real haciendo uso de la función mock location de Android.

La red de datos móvil (EDGE o UTMS) será reemplazada por red WIFI.

El sensor de campo magnético (compass) es usado sin emular, rotando el dispositivo móvil según se necesite.

Contexto Libre:

No se realizan simulaciones, la adquisición de los datos es mediante los instrumentos de sensado de hardware.

Se utiliza la red móvil de datos y el GPS .

En la [sección 7.1.4](#) se instancian las variables mencionadas según la prueba contextual a la que pertenecen, junto con los valores obtenidos y los esperados.

En ambos contextos se agregan al test los ángulos límite derecho e izquierdo según la posición y el nodo actual.

7.1.3 Características a estudiar**Relativas a la navegabilidad :**

- Error en posición: diferencia entre la posición esperada y la obtenida
- Error en orientación: diferencia entre la orientación esperada y la obtenida
- Error en acción de navegabilidad: diferencia entre la acción esperada y la obtenida (Alineado - Girar derecha - Girar izquierda)
- Error en distancia al punto de giro (extras): giros adelantados o tardíos

Relativas al uso general:

- Error en sincronización con calzado (extras): falla de hardware
- Tiempo de respuesta del servicio (extras): tiempo de espera razonable excedido
- Cálculo de Geocoder (extras): Falta de precisión en la geocodificación.
- Cálculo de Ruta(extras): No se obtiene ruta o la ruta es defectuosa.

7.1.4 Resultados individuales de ambos contextos.**Resultados correspondientes al contexto controlado:**

Origen : Calle 57 número 885

Destino : Calle 10 y calle 58

Total recorrido: 4 cuadras

Cantidad de desplazamientos geográficos: 98

Cantidad de desplazamientos chequeados: 21 (más importantes)

Link con la ruta generada : <http://share.mapbbcode.org/vhcfk>

Descripción de la Tabla 7.1:

- La columna “Posición simulada” expresa la localización de la persona en formato(latitud,longitud). Datos ingresados externamente para simulación.
- La columna “Posición obtenida” expresa el valor reportado internamente por la aplicación, debe ser igual al ingresado en la columna de posición simulada; en caso contrario existe una falla en el algoritmo interno de navegación.
- La columna “Bearing simulado” indica la orientación en grados de la persona respecto al norte geográfico.
- La columna “Bearing obtenido” expresa el valor reportado internamente por la aplicación, debe ser igual al ingresado en la columna de bearing simulado; en caso contrario existe una falla en el algoritmo interno de navegación.
- La columnas “Límite derecho” y “Límite izquierdo” expresan los ángulos límite.
- Dentro de este intervalo debe encontrarse el valor de “Bearing obtenido” para permanecer alineado en el camino recorrido.
- La columna “Navegación esperada” es la acción que debería realizar la persona de acuerdo a los datos anteriores.
- La columna “Navegación obtenida” indica la recomendación efectiva que realiza la aplicación, debería coincidir con la “ Navegación esperada”; si no fuera el caso existe un error en el algoritmo de navegación interno.

Posición simulada	Posición obtenida	Bearing simulado	Bearing obtenido	Limite Izquierdo	Limite Derecho	Nav. esperada	Nav. obtenida
-57.95055, -34.92386	-57.95055, -34.92386	4	4	16	62	Girar Derecha	* Girar Derecha
-57.95055, -34.92386	-57.95055, -34.92386	35	35	16	62	Alineado	*Alineado
-57.95045, -34.92377	-57.95045, -34.92377	38	38	16	63	Alineado	Alineado
-57.95024, -34.92357	-57.95024, -34.92357	37	37	14	72	Alineado	*Alineado
-57.95024, -34.92357	-57.95024, -34.92357	79	79	14	72	Girar Izquierda	* Girar Izquierda
-57.95024, -34.92357	-57.95024, -34.92357	293	239	14	72	Girar Derecha	* Girar Derecha
-57.95001, -34.92337	-57.95001, -34.92337	36	36	24	57	Alineado	Alineado
-57.94987, -34.92324	-57.94987, -34.92324	30	30	19	67	Alineado	Alineado
-57.9498, -34.92316	-57.9498, -34.92316	40	40	93	150	Girar Derecha	Girar Derecha
-57.94975, -34.92319	-57.94975, -34.92319	124	124	90	145	Alineado	Alineado
-57.94937, -34.92347	-57.94937, -34.92347	109	109	97	154	Alineado	*Alineado
-57.94896, -34.92374	-57.94896, -34.92374	135	135	118	151	Alineado	Alineado
-57.94888, -34.92381	-57.94888, -34.92381	112	112	102	164	Alineado	Alineado
-57.94883, -34.92384	-57.94883, -34.92384	100	100	36	75	Girar Izquierda	Girar Izquierda
-57.94871, -34.92378	-57.94871, -34.92378	50	50	19	59	Alineado	Alineado
-57.94824, -34.92334	-57.94824, -34.92334	51	51	19	66	Alineado	Alineado
-57.94773, -34.92288	-57.94773, -34.92288	31	31	23	66	Alineado	Alineado
-57.94707, -34.92229	-57.94707, -34.92229	40	40	16	63	Alineado	Alineado
-57.94684, -34.92208	-57.94684, -34.92208	35	35	20	68	Alineado	Alineado

TABLA 7.1: Resultados de usabilidad (Contexto controlado)

Resultados correspondientes al contexto libre:**Descripción de la Tabla 7.2:**

- La columna “Posición obtenida” expresa el valor reportado internamente por la aplicación, obtenido desde el dispositivo GPS real.
- La columna “Inspección visual en el mapa” expresa la verificación visual de que la posición obtenida no tiene errores (en el modo visual de la aplicación).
- La columna “Bearing rea ” indica la orientación en grados de la persona respecto al norte geográfico, obtenido usando una brújula como referencia.
- La columna “Bearing obtenido” expresa el valor reportado internamente por la aplicación, obtenido desde la Brújula (sensor de campo magnético) integrada en el dispositivo móvil, su valor deberá ser lo mas cercano al de la columna “Bearing real”.
- La columnas “Límite derecho” y “Límite izquierdo” expresan los ángulos límite. Dentro de este intervalo debe encontrarse el valor de “Bearing obtenido” para permanecer alineado en el camino recorrido.
- La columna “Navegación esperada” es la acción que debería realizar la persona de acuerdo a los datos anteriores bearing real y localización.
- La columna “Navegación obtenida” indica la recomendación efectiva que realiza la aplicación, debería coincidir con la “ Navegación esperada”; si no fuera el caso algún valor en los datos obtenidos por los instrumentos de sensado no es el correcto.

Posición obtenida	Inspección visual en mapa	Bearing real	Bearing obtenido	Limite Izquierdo	Limite Derecho	Nav. esperada	Nav. obtenida
-57.95055, -34.92386	Correcto	10	13	20	50	Girar Derecha	*Girar Derecha
-57.95055, -34.92386	Correcto	25	29	17	62	Alineado	*Alineado
-57.95045, -34.92377	Correcto	40	37	16	63	Alineado	Alineado
-57.95024, -34.92357	Incorrecto desfasando a izquierdo	37	41	45	58	Alineado	*Girar Derecha
-57.95024, -34.92357	Incorrecto algo mejor que el anterior	42	45	30	50	Alineado	Alineado
-57.95024, -34.92357	Correcto	293	310	20	60	Girar Derecha	* Girar Derecha
-57.95001, -34.92337	Correcto	36	41	24	57	Alineado	Alineado
-57.94987, -34.92324	Pérdida de señal GPS	Pausado	Pausado	Pausado	Pausado	Pausado	Pausado
-57.9498, -34.92316	Correcto	40	45	80	140	Girar Derecha	Girar Derecha
-57.94975, -34.92319	Correcto	124	130	90	145	Alineado	Alineado
-57.94888, -34.92381	Correcto	145	147	102	164	Alineado	Alineado
-57.94883, -34.92384	Correcto	100	105	36	75	Girar Izquierda	Girar Izquierda
-57.94871, -34.92378	Correcto	50	42	19	59	Alineado	Alineado
-57.94824, -34.92334	Perdida Señal GPS	Pausado	Pausado	Pausado	Pausado	Pausado	Pausado
-57.94773, -34.92288	Incorrecto desplazado a izquierda	31	35	53	66	Alineado	Girar Derecha

TABLA 7.2: Resultados de usabilidad(Contexto libre)

7.1.5 Interpretación de resultados

Contexto controlado:

- No se observan errores de posición, de orientación o de acción de navegación.
- Mediante comprobación visual durante el recorrido no se observan errores en la distancia al punto de giro, ni giros adelantados ni tardíos.
- El tiempo de respuesta del servicio no presentó demoras, nunca se excedió el tiempo de espera.
- El geocoder tradujo la descripción textual de la calle destino hacia coordenadas geográficas con precisión, no se observan errores.
- No se observan errores en el cálculo de la ruta.

Contexto libre:

- Existen errores de posición, se encuentran diferencias entre la posición deseada en el mapa y la posición obtenida, generalmente desplazamientos hacia la izquierda.
- Hay errores de orientación, una desviación de +/- 6 grados promedio entre la orientación real y la obtenida con el sensor de campo magnético.
- Se observan errores en las acciones de navegación a tomar, producidos por los errores previos y relacionados con los instrumentos de sensado.
- Hay algunos errores de distancia al punto de giro (tardíos o adelantados), producidos por los errores de posicionamiento.
- El tiempo de respuesta del servicio fue normal, con intervalos en estado de pausa, debido a la pérdida de la señal del GPS.
- El geocoder tradujo la descripción textual de la calle destino hacia coordenadas geográficas con precisión, no se observan errores.
- No se observan errores en el cálculo de la ruta.

7.1.6 Conclusiones

De los resultados se desprende que el sistema de navegación presenta alta sensibilidad a la precisión de los instrumentos de sensado, no observándose ningún error en ninguna

categoría en el contexto simulado; en contraposición, en el contexto libre el número de errores es considerable, atribuible en su totalidad a la instrumentación y a los servicios de datos móviles y de posicionamiento global.

El sistema de posicionamiento (GPS) reporta posiciones imprecisas en numerosas ocasiones, posicionando a la persona hasta 6 metros desviado de su ubicación real; esto afecta el cálculo de los ángulos límite, achicando el margen de tolerancia y por ende la acción a tomar en la navegación. La diferencia de orientación entre el sensor y una brújula real es de 6 grados promedio, lo cual no afecta considerablemente al sistema de navegación.

El servicio de conexión de datos móviles presenta fallas en toda la ciudad de La Plata, perjudicando el proceso de Geocoding y el cálculo de la ruta en las zonas en que el servicio de internet es lento o inestable.

En la versión visual, dependiendo de la velocidad de la conexión a internet, puede existir una demora en la adquisición de los mapas, lo cual puede ser solucionado a futuro con la implementación de mapas fuera de línea (mapsforge).

Manejo de la interrupción de servicios en las pruebas:

Si se interrumpe la adquisición de la ubicación por problemas de la señal del GPS, el algoritmo de navegación se pone en modo pausado hasta que se logre una posición válida. Para el caso en que la ubicación obtenida tiene más de 10 segundos de antigüedad, la navegación se interrumpe hasta que se obtenga una posición actualizada que cumpla con dichas condiciones.

Si hay una falla en la conexión a internet, obtención de posición (actualizada o nula), o conexión con el calzado, la navegación ingresa en modo pausada o detenida según el caso, permitiéndole al no vidente advertir un estado de error y detener su marcha sin correr riesgo de avanzar sin sistema de guía. Para ayudar a manejar los problemas de disponibilidad de servicio, se permite recalcular la ruta en cualquier momento en base a la ubicación actual.

7.2 Prueba de obstáculos

7.2.1 Características a estudiar y resultados

- Distancia al objeto: distancia en cm al objeto.
- Tamaño del objeto: m² del objeto.
- Textura del objeto: tela, madera, papel, material sólido.

- Ángulo: Ángulo de desviación permitido en relación al centro del sensor.
- Velocidad de desplazamiento: lento, normal, rápido, muy rápido.

Materiales utilizados: Pruebas:

Material	Tamaño	Textura
Tapa de Madera 20	20 x 20 cm	Madera sólida de 4 mm
Hoja de Papel A4	21 x 30 cm	Papel fino
Hoja de Papel 5	5 x 5 cm	Papel fino
Hoja de Papel 6	6 x 6 cm	Papel fino
Hoja de Papel 7	7 x 7 cm	Papel fino
Aluminio	10 x 40 cm	Aluminio

TABLA 7.3: Materiales utilizados(Prueba en sensor de obstáculos)

Nombre	Distancia al objeto (grados de desviación máxima logrados)
Tapa de Madera 20	4 cm (5°) - 10 cm (7°) - 30 cm (15°) - 60 cm (20°)
Hoja de Papel A4	4 cm (5°) - 10 cm (7°) - 30 cm (15°) - 60 cm (20°)
Hoja de Papel 5	4 cm (5°) - 10 cm (7°) - 30 cm (15°) - No detectada en 60 cm
Hoja de Papel 6	4 cm (5°) - 10 cm (7°) - 30 cm (15°) - No detectada en 60 cm
Hoja de Papel 7	4 cm (5°) - 10 cm (7°) - 30 cm (15°) - 60 cm (20°)
Aluminio	4 cm (5°) - 10 cm (7°) - 30 cm (15°) - 60 cm (20°)

TABLA 7.4: Prueba de distancia a objetos(Sensor de obstáculos)

Los resultados obtenidos coinciden con las especificaciones del sensor. El objeto de prueba es detectado a una velocidad de desplazamiento normal. El tamaño del objeto no influye sobre los grados de desviación máxima logrados, que dependen de la distancia a la cual se encuentra el objeto y no de su tamaño.

7.2.2 Conclusiones

La gama de materiales detectados es muy amplia, siendo un punto determinante en su usabilidad. La precisión en objetos menores a 50 cm² es muy baja. Los objetos de menor tamaño son detectados a una distancia máxima de 30 cm; a mayor distancia se necesitan objetos de mayor tamaño.

Si bien a mayor distancia el ángulo de medición es mayor, el tamaño del objeto afecta la intensidad de sonido que puede volver al sensor; si ésta no es suficiente no se detecta el objeto. Todas las pruebas se realizaron con objetos con inclinación de 0 ° (de frente al sensor): los objetos que se inclinaron más de 40° no son detectados, debido a que el sonido no rebota.

7.3 Prueba de usabilidad con discapacitado visual

Se realizó una prueba de usabilidad con Maximiliano Vazquez de la cual se desprende la transcripción de esta entrevista:

1. ¿Pensás que la tecnología permite mejorar la independencia de las personas no videntes?

R: Sí, sin duda. La tecnología sirve para mejorar la independencia: la tecnología GPS y la de internet facilitan trámites, distancias, moverse de lugar, un montón de cosas. No sólo en discapacidad visual sino en personas mayores y en otras discapacidades también.

2. Si bien es un prototipo, ¿te parece interesante la posibilidad de perfeccionar esta tecnología? ¿Qué mejoras se te ocurre como usuario que se le podrían hacer?

R: Sí, me parece interesante. La parte más importante es la de navegación para orientarse bien y poder alcanzar el destino al que queremos llegar. También está bueno lo de los obstáculos, que con la ayuda del bastón se puede complementar. Como mejora se me ocurre el tema tamaño, pero es obvio que con mejoras se puede llegar a reducir el espacio.

3. ¿Qué opinás sobre el tiempo que necesita una persona no vidente en poder adaptarse al uso de un calzado de este tipo?

R: No requiere mucho tiempo. Seguramente al que ya trabaja con tecnología le va a costar mucho menos que al que no sabe nada, que tendría que aprender a manejar un táctil para programar el GPS, pero es algo que no es difícil de aprender.

4. ¿Qué te parece el feedback háptico de tipo vibratorio? ¿Tiene ventajas o preferirías mensajes de guía emitidos por voz?

R: Con el audio uno tendría que tener el celular cerca de la oreja, en la mano o con auriculares, así que está bueno que sea vibratorio. El audio tiene algunas ventajas pero otras desventajas, como tener que estar con auriculares, y lo mismo con la vibración, con la que hay que llevar el zapato y estar equipado con la tobillera para que vibre.

5. ¿Qué te parece que el sentido auditivo quede libre para otras tareas?
R: Está bien que quede libre para otras tareas, también por el equipamiento, para no tener que andar con auriculares o cosas que pueden dificultar saber si viene un auto u otra cosa.
6. ¿Pensás que el perfeccionamiento de este tipo de tecnologías puede dar una solución definitiva al traslado independiente?
R: Parece difícil que sea definitivo. Siempre dependemos de alguien: para cruzar una calle, por ejemplo, todavía dependemos de una persona. Hasta ahora puede haber alternativas que no son definitivas, en una esquina puede haber un semáforo accesible, pero no va a estar en todas las esquinas, así que hasta ahora es difícil lograr una independencia total.
7. ¿Pensás que existen ventajas o desventajas en el uso de un calzado háptico en comparación a un perro guía?
R: Son alternativas, pero la ventaja es el costo. Un perro guía hay q conseguirlo. Tal vez es necesario irse a otra ciudad, a otro país. Y tiene mucho costo de viaje, de tiempo, y lo que cuesta también mantener un perro guía es diferente, el costo de vida de un perro guía no tiene comparación con una tecnología, pero son alternativas.
8. ¿Pensás que en un futuro el bastón blanco puede ser sustituido por completo?
R: Sí, hasta ahora no me imagino sin el bastón, como tampoco me imagino con un perro guía. Me acostumbré al bastón, pero puede llegar a estar sustituido. No me lo imagino, pero de acá a veinte años nunca se sabe.
9. No te imaginas con un perro guía o , sin el bastón ¿Con un calzado así te imaginás?
R: Sí, más que nada para orientarme porque a veces me cruzan mal las calles.
10. ¿Conocés o has escuchado hablar de algún otro tipo de tecnología que sirva como ayuda?
R: Conozco en la ciudad de Buenos Aires los semáforos accesibles, con sonido. Conozco solamente ese tipo de semáforo, pero sé que hay otro tipo de en Mendoza, en donde te dan un llavero que sirve como un sensor que cuando vas a cruzar te avisa el estado del semáforo.
11. ¿Acá en la plata no hay?
R: No, presenté un proyecto en la Legislatura pero no ha sido aprobado aun.
12. ¿Una reflexión final acerca del calzado?
R: Está bueno porque ayuda a la orientación y a la independencia de una persona con discapacidad. Especialmente en la navegación, en poder llegar a un destino.

Sobre todo cuando las calles son diagonales, que a veces se pierden, me dan una dirección que no es. También está bueno para buscar las paradas de los micros.

13. ¿Te resultó fácil ponerte el zapato?

R: Sí, fue bastante fácil. Pude llegar a darme cuenta bien en la prueba, incluso me cambiaba de posición para ver cómo andaba la orientación, y andaba bien.

7.4 Conclusiones sobre mejoras posibles

Para un óptimo funcionamiento del sistema es deseable sustituir el GPS convencional por un GPS profesional con compensaciones especiales y de alta precisión, así como la sustitución del sensor de distancia por uno profesional de mayor ángulo, y la contratación de un servicio con mejor cobertura de red de datos de alta velocidad. A modo de síntesis, se deberían reemplazar los instrumentos de medición, por unos de calidad profesional.

Capítulo 8

Descripción de la aplicación

En la Figura 8.1 se puede ver el menú principal de la aplicación.

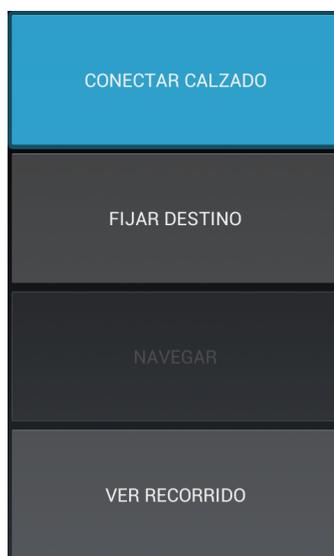


FIGURA 8.1: Menú principal de la aplicación

El Menú Principal provee cuatro funciones:

- **Conectar Calzado.** Realiza la conexión inalámbrica con el calzado que fue previamente activado desde el interruptor de encendido.
- **Fijar Destino.** Mediante reconocimiento de voz humana, se ingresa la dirección destino deseada. La dirección destino es pronunciada por el lector de pantalla para su confirmación.
- **Navegar.** Inicia el proceso de navegación. Este botón está activado sólo cuando la conexión con el calzado fue realizada con éxito y además se fijó una dirección destino. Ver Figura 8.2.



FIGURA 8.2: Recorrido visual en tiempo real

- **Ver Recorrido.** Mapa visual que contiene la ruta calculada e información en tiempo real del estado del recorrido. Utilizado por personas videntes con fines de diagnóstico y como posibilidad para acelerar la curva de aprendizaje.

Ver Figura 8.2.

Inmediatamente de iniciada la navegación, el usuario recibe el feedback háptico indicando la acción a realizar para mantenerse en la ruta hacia su destino.

El diseño de la interfaz es minimalista, para evitar confusiones en el uso. Todos los componentes visuales fueron descritos programáticamente de acuerdo con la especificación de accesibilidad propuesta por Android, asegurando la compatibilidad con Talkback.

Capítulo 9

Conclusiones

Es indiscutible el uso de la tecnología como herramienta para mejorar la calidad de vida de las personas en general. Desde una óptica simplista, se la puede reducir a un mecanismo para automatización de tareas repetitivas; función que cumple a la perfección.

El rol de las tecnologías de innovación, como el presente trabajo, va aún más allá de la automatización, permitiendo también sustituir el órgano encargado de recibir los estímulos del exterior; para luego sí realizar una tarea con dicha información.

Se pueden pensar los datos obtenidos del GPS, del sensor de campo magnético y del sensor de distancia, como una sustitución de la ubicación, orientación y distancia obtenida por el sentido de la vista. Esta información visual es procesada y redirigida hacia el sentido del tacto, evidenciándose el reemplazo de un sentido de percepción por otro.

Tomando como base las pruebas de usabilidad sobre este prototipo, las innovaciones en esta área pueden ser una ayuda real para personas con capacidades visuales disminuidas. Es importante destacar que el reemplazo del sentido de la vista por un instrumento de sensado electrónico requiere de una gran precisión en la obtención de los datos del mundo real, ya que la sincronización de éstos es en lo que se basa todo el sistema. A medida que los servicios e instrumentos se perfeccionen, mayor será la precisión de los resultados.

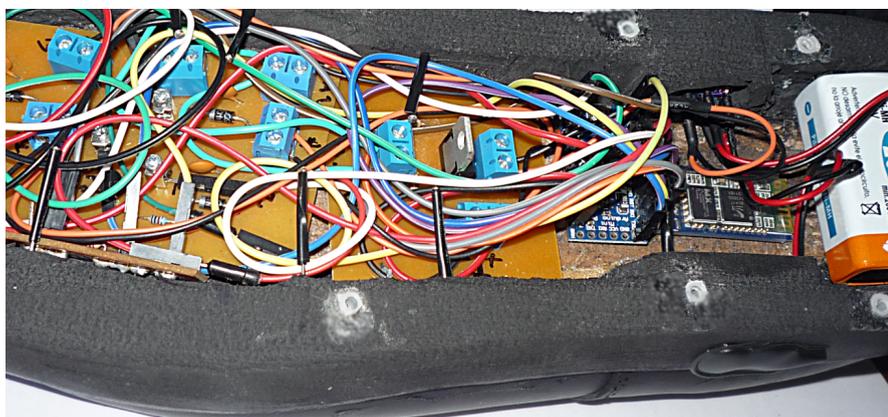
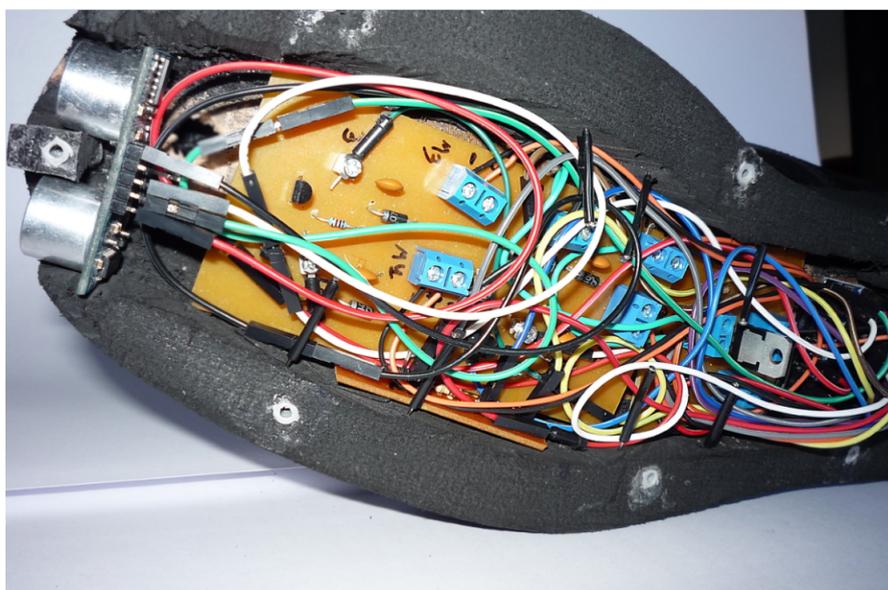
Todo el trabajo se logró desarrollar con un muy bajo costo económico, gracias a la existencia de plataformas de open source hardware y sus comunidades que mediante el intercambio de esquemas de diseño de componentes permiten crear módulos de hardware propios, abaratando los costos en un orden de 5 a 6 veces el valor de un producto comprado. La reusabilidad de código fuente, disponible mediante licencias abiertas sin restricciones en el uso de los datos, permite lograr mayor velocidad de desarrollo.

Como conclusión final, el prototipo de calzado háptico desarrollado abre una primera aproximación al perfeccionamiento de este tipo de aplicación de la tecnología para el

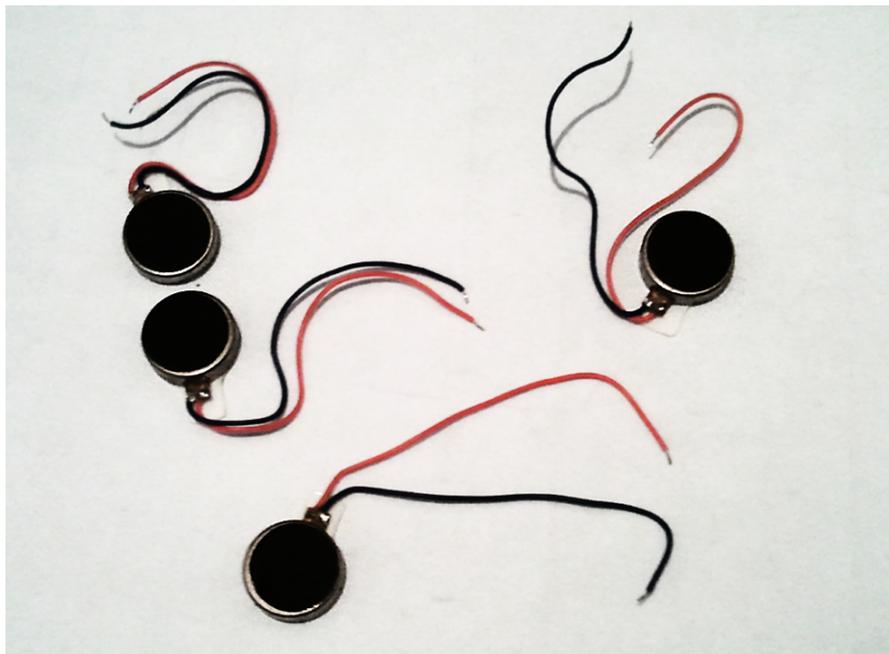
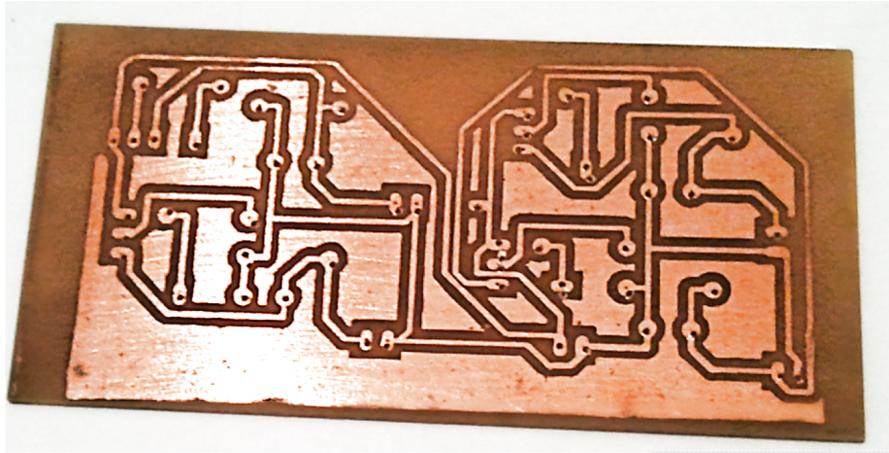
mejoramiento de la calidad de vida de las personas no videntes, sirviendo de base para desarrollos futuros; así como también aplica y contribuye a iniciativas de Open Source Hardware y de Open Source Software en la creación y mejoramiento de productos tecnológicos.

Apéndice A

Apéndice de figuras







Referencias Bibliográficas

- [1] Android. *Dashboards*, Octubre 2014. URL <http://developer.android.com/about/dashboards/index.html>.
- [2] Android Inc. *El API de codificación geográfica de Google*, Diciembre 2014. URL <https://developers.google.com/maps/documentation/geocoding/>.
- [3] Anirudh Sharma. *Haptic shoe for the visually impaired. MIT TR35 'Innovator of the year'*, 2012. URL <http://www2.technologyreview.com/tr35/profile.aspx?TRID=1258>.
- [4] Arduino. *Products and specifications*, Diciembre 2014. URL <http://arduino.cc/en/Main/Products>.
- [5] CC. *Documentation*. Creative Commons, Diciembre 2014. URL <https://wiki.creativecommons.org/Documentation>.
- [6] CERN. *CERN Open Hardware Licence v1.2*. The European Organization for Nuclear Research, Septiembre 2013. URL http://www.ohwr.org/attachments/2390/cern_ohl_v_1_2.pdf.
- [7] Gardner, Martin, Jessell. *Principles of Neural Science. The bodily Senses*, Octubre 2012. URL <http://www.amazon.com/Principles-Neural-Science-Edition-Kandel/dp/0071390111/>.
- [8] Google Inc. *Our Mobile Planet: Argentina*, 2013. URL <http://services.google.com/fh/files/misc/omp-2013-ar-local.pdf>.
- [9] Google Inc. *Google Maps/Google Earth APIs Terms of Service*, Noviembre 2014. URL <https://developers.google.com/maps/terms>.
- [10] Graphhopper. *MapQuest Open Guidance API Web Service*, Diciembre 2014. URL <https://graphhopper.com/#community>.
- [11] GSM Association. *Mobile Economy Latin America*, 2013. URL http://gsma.com/newsroom/wp-content/uploads/2013/12/GSMA_ME_LatAm_Report_2013.pdf.

- [12] Immersion. *The value of haptic*, 2010. URL http://www.immersion.com/docs/Value-of-Haptics_Jun10-v2.pdf.
- [13] INDEC. *Cuadro P10. Total del país. Población en viviendas particulares con dificultad o limitación permanente por tipo de dificultad o limitación permanente, según sexo y grupo de edad*. Instituto Nacional de Estadística y Censos, 2010. URL http://www.censo2010.indec.gov.ar/cuadrosDefinitivos/Total_pais/P10-Total_pais.xls.
- [14] INDEC. *Servicio telefónico básico: líneas instaladas, líneas en servicio, teléfonos públicos y llamadas nacionales urbanas; Servicio de telefonía celular móvil: teléfonos en servicio, llamadas y mensajes de texto SMS*. Instituto Nacional de Estadística y Censos, 2013. URL http://www.indec.gob.ar/desaweb/nivel4_default.asp?id_tema_1=3&id_tema_2=11&id_tema_3=53.
- [15] InMojo. *OSHW License Guide*, Diciembre 2014. URL <http://www.inmojo.com/licenses/>.
- [16] INPI. *Administración Nacional de Patentes-Preguntas Frecuentes*. Instituto nacional de propiedad industrial (Argentina), Diciembre 2014. URL <http://www.inpi.gov.ar/index.php?Id=107&criterio=2>.
- [17] Javier Pastor. *¿Quiénes son los dueños de los mapas en el mundo?*, Marzo 2014. URL <http://www.xataka.com/gadgets-y-coches/quienes-son-los-duenos-de-los-mapas-en-el-mundo>.
- [18] Jeffrey M. Osier-Mixon. *Open hardware: How and why it works-Applying open source concepts to physical objects*, Noviembre 2010. URL <http://www.ibm.com/developerworks/library/os-openhardware/os-openhardware-pdf.pdf>.
- [19] Mapquest. *Community Edition License Agreement*, Noviembre 2011. URL <http://developer.mapquest.com/web/info/terms-of-use>.
- [20] Mapquest. *Licensed Data vs. Open Data*, Noviembre 2014. URL <http://developer.mapquest.com/web/tools/getting-started/platform/licensed-vs-open>.
- [21] OpenCage. *The OpenCage Geocoder API*, Diciembre 2014. URL <http://geocoder.opencagedata.com/api.html>.
- [22] OpenStreetMap. *Derechos de autor y licencia*, Diciembre 2014. URL <http://www.openstreetmap.org/copyright>.
- [23] Osmroid. *Github project*, Diciembre 2014. URL <https://github.com/osmdroid/osmdroid>.

- [24] OSWHA. *Best Practices for Open-Source Hardware 1.0*, Diciembre 2014. URL <http://www.oshwa.org/sharing-best-practices/>.
- [25] OSWHA. *Definition*, Diciembre 2014. URL <http://www.oshwa.org/definition/>.
- [26] OSWHA. *May and Must*, Diciembre 2014. URL <http://www.oshwa.org/wp-content/uploads/2014/08/OSHW-May-and-Must.pdf>.
- [27] Pinguino. *Open Hardware Electronics Prototyping Platform*, Diciembre 2014. URL <http://www.pinguino.cc/>.
- [28] Precision Microdrives. *Adding and improving haptic feedback*, 2012. URL <http://www.precisionmicrodrives.com/haptics-haptic-feedback-vibration-alerting/haptic-feedback-in-detail/adding-and-improving-haptic-feedback>.
- [29] Precision Microdrives. *What is Haptic / Tactile Feedback?*, 2012. URL <http://www.precisionmicrodrives.com/haptics-haptic-feedback-vibration-alerting/haptic-feedback-in-detail/an-introduction-to-haptic-feedback>.
- [30] Precision Microdrives. *Discrete Driver Circuits for Vibration Motors*, Diciembre 2014. URL <http://www.precisionmicrodrives.com/application-notes-technical-guides/application-bulletins/ab-001-discrete-driver-circuits-for-vibration-motors>.
- [31] RAE. Real Academia Española, Agosto 2014. URL <http://lema.rae.es/drae/srv/search?id=1LHI73Zr8DXX20IBnlbt>.
- [32] Roger W. Cholewiak. *La Háptica y los sentidos del tacto*, Octubre 2013. URL <http://ilevolucionista.blogspot.com.ar/2013/10/la-haptica-y-los-sentidos-del-tacto.html>.
- [33] TAPR. *The TAPR Open Hardware License*, Mayo 2007. URL http://www.tapr.org/TAPR_Open_Hardware_License_v1.0.pdf.
- [34] Teensy. *Teensy 3.1 - Features*, Diciembre 2014. URL <https://www.pjrc.com/teensy/teensy31.html>.
- [35] Vantika Dixit. *Haptic Shoe For The Blind - Le Chal. MIT (Massachusetts Institute of Technology) India Edition- Magazine Technology Review*. Anirudh Sharma, Agosto 2011. URL <https://www.scribd.com/fullscreen/62713533>.
- [36] WIPO. Principios básicos de la propiedad industrial. World Intellectual Property Organization, Diciembre 2014. URL http://www.wipo.int/edocs/pubdocs/es/intproperty/895/wipo_pub_895.pdf.

-
- [37] Wiring Inc. *Wiring Hardware overview*, Dicembre 2014. URL <http://wiring.org.co/hardware/>.
- [38] Yahoo. *Free Non-Commercial YQL Table Usage*, Dicembre 2014. URL https://developer.yahoo.com/boss/geo/docs/free_YQL.html.
- [39] Yahoo. *PlaceFinder location parameters*, Dicembre 2014. URL <https://developer.yahoo.com/boss/geo/docs/location-parameters.html>.