



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA INDUSTRIAL

DISEÑO Y VALIDACIÓN DE UN MÉTODO EN COMPUTACIÓN UBICUA QUE MEJORE
EL FUNCIONAMIENTO DE LA PLATAFORMA LAZARILLO EXTENDIENDO SU USO A
ESPACIOS CERRADOS

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL

JONATHAN ALONSO TAIVO RODRÍGUEZ

PROFESOR GUÍA:
ÁNGEL JIMÉNEZ MOLINA

MIEMBROS DE LA COMISIÓN:
EDGARDO SANTIBÁÑEZ VIANI
IGNACIO CALISTO LEIVA

SANTIAGO DE CHILE
2017

RESUMEN DE LA MEMORIA PARA OPTAR AL
TÍTULO DE: Ingeniero Civil Industrial
POR: Jonathan Alonso Taivo Rodríguez
FECHA: 12/06/2017
PROFESOR GUIA: Ángel Jiménez Molina

DISEÑO Y VALIDACIÓN DE UN MÉTODO EN COMPUTACIÓN UBICUA QUE MEJORE EL FUNCIONAMIENTO DE LA PLATAFORMA LAZARILLO EXTENDIENDO SU USO A ESPACIOS CERRADOS

El presente trabajo se enmarca en el proceso de mejora de la plataforma Lazarillo, aplicación guía para personas en situación de discapacidad visual¹ que a través de mensajes de voz, informa al usuario acerca de su entorno, dándole a conocer los servicios que se encuentran a su alrededor, como calles por las cuales se desplaza, cruces, paraderos, entre otros elementos relevantes. Su funcionamiento se acota a espacios abiertos, haciendo uso del GPS y distintas fuentes de datos y, a través de esta investigación e implementación realizadas, se extiende su uso a espacios cerrados, haciendo uso de dispositivos Bluetooth de bajo consumo conocidos comúnmente como beacons.

Luego del estudio de diversas tecnologías asociadas a los beacons, se concluye que la apropiada para este proyecto es el uso del protocolo Eddystone, desarrollado por Google, lanzado el 2015 y de gran popularidad en la actualidad. Dentro de sus ventajas destaca ser de código abierto, permitiendo a la comunidad poder realizar aportes, además cuenta con una plataforma en línea que permite asociar attachments o mensajes a cada dispositivo, incentivando el desarrollo de aplicaciones de terceros, compatibilidad con cualquier dispositivo BLE, flexibilidad y API's de alto nivel que permiten la conexión de forma fácil, manejando internamente los problemas de conexión.

Para la generación de mensajes auditivos se toma como referencia un estándar desarrollado por Wayfindr, quienes pretenden generar las bases de la navegación indoor para PeSD visual, contando con el apoyo de grandes instituciones como Google y otras relacionadas. La metodología utilizada para la captura de la información contextual se basa en la generación de rutas dirigidas que cubran todos los elementos de navegación que se requiera comunicar, sin la necesidad de ingresar información estructural del espacio. Con sólo algunos parámetros es posible construir mensajes contextuales de gran precisión. La metodología permite establecer asociaciones beacon - beacon o punto georreferenciado - beacon, con el potencial de generar 3 tipos de mensajes: posicionamiento del elemento con respecto a la ruta en sentido inicial, posicionamiento con respecto a la ruta en sentido inverso y un mensaje genérico en caso de no tener información del sentido del recorrido.

La arquitectura de datos que hace posible el funcionamiento propuesto se configura en torno a árboles JSON que permiten estructurar fácilmente la lógica de nodos padres e hijos, representando así lugares, rutas asociadas a lugares y beacons asociados a rutas y, en algunos casos, directamente a lugares. La implementación se lleva a cabo utilizando una base de datos en tiempo real llamada Firebase la que cuenta con API's de conexión multiplataforma que facilitan el desarrollo tanto para dispositivos móviles como para plataformas web.

¹en adelante PeSD visual

Tabla de Contenido

1. Introducción	1
1.1. Descripción	2
1.2. Justificación	2
1.2.1. Relevancia del trabajo a realizar	3
1.2.2. Viabilidad de implementación	3
1.2.3. Apoyos Institucionales	4
1.3. Objetivos	4
1.3.1. Objetivo General	4
1.3.2. Objetivos Específicos	4
1.4. Metodología	5
1.5. Alcances	6
1.5.1. Output de la aplicación	6
1.5.2. Entorno para desarrollo de pruebas piloto	6
1.6. Resultados Esperados	7
1.7. Plan de Trabajo	7
2. Marco Teórico	10
2.1. Tecnologías de uso común en navegación indoor	10
2.1.1. Wifi	10
2.1.2. Global Positioning System (GPS)	11
2.1.3. Near Field Communication (NFC)	11
2.1.4. Bluetooth	12
2.2. Métodos de posicionamiento	13
2.2.1. Triangulación [1]	13
2.2.2. Lateración [2]	13
2.2.3. Posicionamiento en base a celdas [3]	13
2.2.4. Fingerprinting [4]	14
2.2.5. Pedestrian Dead Reckoning	14
2.3. Tecnologías asistivas	14
2.3.1. Algoritmos y Tecnologías de Ruteo de PeSD Visual	14
2.3.2. Modelo de Grafos para Estructuras de Edificaciones	15
2.3.3. Semántica de Accesibilidad	16
2.3.4. Clasificación de mensajes para la semántica de accesibilidad de espacios interiores	16
2.3.5. Sistemas de Computación Ubicua para Asistir a PeSD Visual	16
2.3.6. Uso de Tecnologías Asistivas	18

2.4.	Soluciones existentes para ayuda desplazamiento PeSD visual	18
2.4.1.	Asistencia a través de perros guía o lazarillos	18
2.4.2.	Asistencia humana	19
2.4.3.	Asistencia a través de aplicaciones móviles	19
3.	Estado actual de Lazarillo y relevancia del trabajo	20
3.1.	Acerca de Lazarillo	20
3.1.1.	Misión	21
3.1.2.	Visión	21
3.2.	Estado actual	21
3.2.1.	Modelo de negocio	21
3.2.2.	Análisis estratégico	22
3.2.3.	Análisis del microentorno	23
3.3.	Consecuencias del trabajo	24
3.3.1.	Extensión del modelo de negocio	24
4.	Servicio de localización	26
4.1.	Protocolos de beacons	26
4.2.	APIs de conexión	27
4.3.	Eddystone e iniciativas abiertas	27
4.4.	Hardware a utilizar	28
4.4.1.	Especificaciones técnicas Onyx Beacon Enterprise	28
4.4.2.	Especificaciones técnicas Kontakt Beacon Pro	29
5.	Algoritmo de flujos de información	30
5.1.	Funcionamiento actual de LazarilloApp	30
5.2.	Exploración en base a rutas predefinidas	32
5.2.1.	Modo Exploración y Modo Ruteo	34
5.2.2.	Primera iteración: enfoque rutas dirigidas	36
5.2.3.	Segunda Iteración: enfoque tramos dirigidos	37
5.2.4.	Tercera Iteración: tramos dirigidos y Puntos de Referencia	38
5.3.	Diagramas de flujo de información	39
6.	Procedimiento para levantamiento de información	44
6.1.	Base de datos	44
6.2.	Arquitectura de datos	45
6.2.1.	Estructura base	45
6.2.2.	Elementos secundarios	46
6.3.	Flujo levantamiento de información	54
6.4.	Portal Lazarillo	54
7.	Desarrollo de prototipo funcional	60
7.1.	Vistas prototipo funcional e interacciones	60
7.1.1.	Características del Prototipo para Pruebas	60
8.	Evaluación de usabilidad, impacto y resultados	63
8.1.	Evaluación de usabilidad LazarilloApp	63
8.1.1.	Resultados	64

8.2. Preexperimento 1	67
8.2.1. Correcciones	69
8.3. Preexperimento 2	70
8.4. Experimento	72
8.4.1. Instrumentación	72
8.4.2. Grupo experimental	72
8.4.3. Diseño del experimento	73
8.4.4. Métricas	76
8.5. Análisis y resultados	77
9. Conclusiones y trabajo futuro	81
9.1. Conclusiones del trabajo	81
9.2. Trabajo futuro	82
Bibliografía	83
Anexos y apéndices	87
A. Pauta Evaluación de Usabilidad	88
B. Tipos de mensajes emitidos	89

Índice de tablas

4.1. Tabla comparativa iBeacon y Eddystone	27
6.1. lzBeacons	52
6.2. routes	52
6.3. geoLzPlace	52
6.4. lzPlaces	53
6.5. Usuarios	53
6.6. Lugares	53
8.1. Dimensiones SERVQUAL, encuesta usabilidad LazarilloApp	64
8.2. Tabla satisfacción de usuarios	67
8.3. Perfiles usuarios de prueba, preexperimento	68
8.4. Problemas detectados durante preexperimento	68
8.5. Sugerencias propuestas por usuarios de prueba durante preexperimento	69
8.6. Perfiles usuario de prueba, preexperimento 2	72
8.7. Perfiles usuarios de prueba, experimento	73
8.8. Variables cuantificables en experimento	77
8.9. Tabla comparativa modelo planteado versus modelo mensajes genéricos	79
B.1. Mensajes contabilizados por tipo	89

Índice de ilustraciones

1.1. Metodología	5
1.2. Carta Gantt	9
3.1. Social Lean Canvas Lazarillo	22
3.2. Nuevo Modelo de Negocios Lazarillo	25
4.1. Beacon Onyx modelo Enterprise	29
4.2. Beacon Kontakt	29
5.1. Vista principal LazarilloApp, Modo Exploración	32
5.2. Vista de lugar seleccionado por usuario	33
5.3. Funcionamiento Lazarillo	34
5.4. Layout Hall y rutas	37
5.5. Generación de avisos modo exploración	40
5.6. Construcción aviso auditivo	40
5.7. Procedimiento donde estoy	43
6.1. JSON ejemplo Instituciones	45
6.2. Esquemas de ejemplo	46
6.3. Esquemas de ejemplo	47
6.4. JSON ejemplo Arcos	48
6.5. Esquemas de ejemplo	50
6.6. JSON ejemplo Categorías	51
6.7. Flujo creación de tramos dirigidos	54
6.8. Flujo creación de Puntos de Referencia	55
6.9. Flujo creación de sucursales	55
6.10. Vista Portal Lazarillo: Nueva institución	56
6.11. Vista Portal Lazarillo: Nuevo evento	57
6.12. Vista Portal Lazarillo: Nueva sucursal	57
6.13. Vista Portal Lazarillo: Adjuntar plano en sucursal	58
6.14. Vista Portal Lazarillo: Nuevo Punto de Referencia	58
6.15. Vista Portal Lazarillo: Nuevo tramo dirigido	59
6.16. Vista Portal Lazarillo: Red de tramos dirigidos	59
7.1. Vista principal Prototipo para Pruebas	61
8.1. Distribución etaria	65
8.2. Distribución países	66

8.3. Idioma	66
8.4. Ejemplo flujos de activación de beacons teóricos vs en terreno	70
8.5. Segmentos dirigidos	70
8.6. Algoritmo de construcción de mensajes contextuales de posición relativa mejorado	71
8.7. Output mensajes según segmento recuperado con algoritmo mejorado	71
8.8. Plano recorrido Explanada	74
8.9. Plano recorrido Planta Baja	75
8.10. Proporción mensajes emitidos	78
8.11. Resultados test de usabilidad	79
A.1. Pauta de evaluación de usabilidad basada en pauta de profesor Jaime Sanchez . . .	88

Capítulo 1

Introducción

En Chile, según el último informe de discapacidad realizada por el INE el pasado 2015, un 20% de la población mayor de edad se encuentra en alguna situación de discapacidad. De este porcentaje, cerca de un 72,7% correspondiente a 2.606.914 personas, tiene alguna enfermedad relacionada al ojo, donde se incluye todo el rango de personas en situación de discapacidad visual, en adelante PeSD visual, desde baja visión a ceguera total. En el mundo la situación es similar, contabilizándose un número cercano a los 285.000.000 de PeSD visual.

La ley 20.422 acerca de los Derechos de las Personas con Discapacidad indica en su artículo 9: “las personas en situación de discapacidad pueden vivir en forma independiente y participar plenamente en todos los aspectos de la vida”. De acuerdo a esta normativa, el Estado de Chile debe asegurar la total inclusión en la sociedad de las personas con algún tipo de discapacidad, en particular de las personas en situación de discapacidad de visual. Sumado a lo anterior, el desarrollo de lectores de pantalla cada vez más intuitivos ha permitido acortar la brecha tecnológica de PeSD visual, abriéndose así un mundo de oportunidades antes limitada a la interacción con material desarrollado en braille, cuyo catálogo se encuentra restringido y que no actualiza con la velocidad que tiene, por ejemplo, internet. En esta línea, se han desarrollado en Chile iniciativas que han intentado sin éxito generar inclusión para PeSD visual en la sociedad. Un ejemplo reciente de éstas es el proyecto de placas NFC desarrollado por el Metro de Santiago, cuya implementación ha recibido críticas dentro de sus mismos usuarios. La idea de ofrecer información del entorno al usuario para poder desplazarse a través de la estación tiene una limitancia fundamental en la tecnología utilizada, ya que obliga al usuario a acercar su dispositivo móvil a centímetros de cada una de las placas, lo que implica que el usuario debe ser capaz de reconocer la ubicación - y previamente la existencia - de la primera placa de la cadena. Esto fue detectado por René Espinoza, uno de los desarrolladores de esta iniciativa, cuyos feedbacks le permitieron pivotar la idea inicial en una más robusta.

Así nace LazarilloApp, una aplicación móvil destinada a orientar y vincular a personas ciegas y con baja visión con los servicios de su ciudad. La aplicación por medio de mensajes de voz y un algoritmo inteligente de priorización informa de paraderos de bus, cafeterías, bancos, restaurantes, intersecciones de calles, la calle por la que se encuentra caminando y más, mejorando así su capacidad de desplazarse con mayor seguridad, disminuyendo tiempos de traslado y entregando autonomía e independencia.

LazarilloApp constituye un producto innovador en el mercado de aplicaciones para personas con discapacidad que funciona a nivel mundial. Ha sido desarrollado en conjunto con los usuarios, con procesos de iteración constantes y con la presencia en el equipo de desarrollo de una persona con discapacidad visual lo que ha permitido, junto con el constante feedback de los usuarios, desarrollar una interfaz altamente accesible a las personas con ceguera y baja visión, quienes pueden explorar la aplicación por medio de los lectores de pantalla disponibles en Android y iOS, Talkback y Voiceover.

LazarilloApp fue lanzada al mercado el 9 de septiembre de 2016 para celulares Android y iPhone, en sus respectivas tiendas de aplicaciones, quedando disponible para su descarga gratuita a nivel mundial en español e inglés. Actualmente Lazarillo cuenta con cerca de 3500 usuarios activos mensuales generados de forma orgánica y distribuidos principalmente en Chile, Argentina, México, Colombia y España.

1.1. Descripción

El trabajo a desarrollar consiste en la continuación del trabajo de título desarrollado por René Espinoza [5], en el cual se concluye que el mejor método de posicionamiento indoor para el proyecto corresponde al posicionamiento en base a celdas utilizando dispositivos emisores Bluetooth. De esta forma se espera mejorar el funcionamiento de Lazarillo, extendiendo su uso a espacios cerrados, dando continuidad al trabajo desarrollado en espacios abiertos.

1.2. Justificación

LazarilloApp cuenta actualmente con diversas funcionalidades que le permiten al usuario desenvolverse en la ciudad de forma independiente, la más importante de ellas es el Modo Exploración¹, funcionalidad base del sistema que consiste en la descripción del contexto en la medida que el usuario se desplaza de forma libre por el entorno la que, por su importancia, es activada inmediatamente luego de iniciada la aplicación. El objetivo principal de este modo es suplir, en parte, la carencia de visión, permitiéndole al usuario reconocer su entorno cercano, ayudando la construcción de mapas mentales y su posterior recordación. Existen dos limitantes importantes en la solución actual:

1. El algoritmo encargado de extraer la información y comunicarla al usuario cumple el objetivo primario de dar a conocer los elementos cercanos y funciona adecuadamente en condiciones estándar², pero posee errores tales como:
 - a) Todos los lugares tienen igual prioridad entre ellos, salvo por su distancia relativa al usuario. De esta forma los elementos más cercanos son nombrados en primera instancia lo que, en escenarios de alta densidad de lugares, provoca que el sistema deje de informar acerca de posibles lugares de mayor relevancia para la correcta descripción del entorno.

¹Más información en sección 5.2.1

²Densidad de lugares mínima y desplazamiento constante del usuario

b) Dada una posición fija, el algoritmo entregará los 10 lugares más cercanos – de existir – ordenados por su distancia relativa al usuario con un tope de metros, con un tiempo entre notificaciones fijo. Los lugares notificados son guardados en una lista. Al haber un desplazamiento del usuario, la posición se actualiza y el proceso se inicia con esta nueva posición como input. Un grave problema que tiene este método es que, dada una posición, al haber periodos de detención y zonas de baja densidad de lugares, el algoritmo tiende a notificar de lugares alejados que pasan a la lista de enunciados y no son notificados cuando el usuario efectivamente está cercano a ellos.

c) El algoritmo no considera preferencias del usuario.

2. La funcionalidad está acotada sólo a espacios abiertos a través del uso de diversas fuentes de información tales como Google Maps, OpenStreetMap, Foursquare, entre otros.

Dado lo anterior se propone un nuevo sistema de exploración para uso en espacios interiores, en base a un sistema de priorización que permita la notificación elementos de mayor relevancia para la descripción del entorno. El algoritmo debe ser capaz de funcionar sólo con el uso de un dispositivo móvil y dispositivos de emisión Bluetooth. Además, debe ser fácilmente extensible a entornos abiertos.

1.2.1. Relevancia del trabajo a realizar

La ley 20.422 acerca de los Derechos de las Personas con Discapacidad indica en su artículo 9: “las personas en situación de discapacidad pueden vivir en forma independiente y participar plenamente en todos los aspectos de la vida”. De acuerdo a esta normativa, el Estado de Chile debe asegurar la total inclusión en la sociedad de las personas con algún tipo de discapacidad, en particular de las PeSD visual por lo que el accesibilizar nuevos espacios tiene un gran impacto y relevancia para nuestra realidad nacional, sobre todo por no existir iniciativas que realicen este trabajo³. En este sentido presente trabajo tiene asociado un beneficio social de incalculable valor ya que permite generar independencia en PeSD visual.

1.2.2. Viabilidad de implementación

Los fondos actuales⁴ con los que cuenta el proyecto permiten el desarrollo del presente trabajo del estudiante. Adicionalmente, existe un proyecto de habilitación de espacios interiores para el Museo de la Memoria y los Derechos Humanos a desarrollar por Lazarillo en los próximos meses que le permitiría al estudiante realizar un piloto en condiciones de uso reales.

³Un primer acercamiento fue realizado por Metro de Santiago a través de sus placas con tecnología NFC. El problema evidente de esta iniciativa tiene que ver con la dificultad para la PeSD visual en encontrar la primera placa e iniciar el recorrido

⁴Para detalles, ver sección Apoyos Institucionales

1.2.3. Apoyos Institucionales

Actualmente se hace uso de un fondo SSAF de innovación de Corfo y el apoyo económico del Instituto de Sistemas Complejos de Ingeniería. En paralelo a lo anterior, Lazarillo ha contado con el apoyo de otros beneficios como un SSAF social de Corfo, un fondo Inspira Senadis entre otros. Los apoyos antes descritos han ido en directo beneficio de la investigación a desarrollar por el alumno.

Por otro lado, el proyecto de accesibilizar el Museo de la Memoria y los Derechos Humanos otorga al estudiante la posibilidad de realizar pruebas piloto en un espacio de uso público en condiciones de uso real, lo que puede entregar información valiosa de beneficio adicional para la investigación. Si bien lo anterior es importante para el desarrollo de la presente investigación ya que provee una infraestructura robusta para el funcionamiento indoor de Lazarillo, se considera una segunda alternativa la implementación de un piloto a menor escala en la facultad con fondos propios (Lazarillo SpA) en los espacios del OpenLab.

1.3. Objetivos

Este trabajo de título tiene un objetivo general y cinco objetivos específicos.

1.3.1. Objetivo General

Diseñar y validar un método en computación ubicua que mejore el funcionamiento de la plataforma Lazarillo, extendiendo su alcance a espacios cerrados.

1.3.2. Objetivos Específicos

1. Designar servicio de localización en base a método de posicionamiento por celdas impuesto por Lazarillo.
2. Diseñar procedimiento para el levantamiento de información en espacios cerrados.
3. Desarrollar algoritmo que genere flujos de información contextual de localización.
4. Desarrollar prototipo funcional en forma de aplicación móvil para sistema operativo Android.
5. Evaluar usabilidad e impacto del prototipo desarrollado en base a método propuesto con PeSD visual.

1.4. Metodología

La metodología a utilizar para el desarrollo del trabajo se muestra a continuación, en el esquema de la figura 1.1:



Figura 1.1: Metodología

Para el desarrollo del prototipo funcional existen 3 requisitos:

- Designar el servicio de localización en base a requerimientos de Lazarillo: Un input importante extraído del trabajo realizado por René Espinoza, corresponde al uso del método de posicionamiento por celdas y el uso de dispositivos Bluetooth para la estimación de localización en espacios cerrados. Como primera parte del trabajo se debe seleccionar algún servicio de posicionamiento que funcione utilizando este método. Se debe seleccionar el servicio que presente un equilibrio entre la facilidad de implementación como un nivel de confianza acorde en sus pronósticos. Una vez designado el servicio de localización se procede a su implementación.
- Diseñar procedimiento para levantamiento de información de espacios cerrados: el procedimiento debe permitir realizar este trabajo en diversos escenarios tales como centros comerciales, edificios públicos, oficinas, entre otros. Además debe permitir una fácil instalación por parte del usuario.
- Desarrollar un algoritmo que controle los flujos de información a partir de una estructura de datos e información adquirida a través de los sensores del dispositivo móvil.

Tanto la designación del servicio de localización, el diseño del procedimiento para el levantamiento de información y el desarrollo del algoritmo de flujos de información se deben desarrollar

en paralelo ya que todos dependen de los demás. En particular, es necesario realizar implementaciones de testing para comprobar el funcionamiento de las propuestas lo que genera en cada iteración un prototipo funcional. Este proceso se enmarca en la metodología denominada como Design Thinking y permite ir mejorando los pasos anteriores en base al resultado obtenido.

Luego de obtenido el prototipo funcional definitivo, se procede a la realización de las diversas pruebas de usabilidad e impacto para evaluar cualitativa y cuantitativamente la solución propuesta.

1.5. Alcances

A partir del análisis previo realizado por el alumno, se describen a continuación los alcances del estudio en dos ámbitos: output de la aplicación y entorno para desarrollo de pruebas piloto.

1.5.1. Output de la aplicación

Comunicación del entorno

Metodología encargada de la generación de los flujos de información entregados al usuario de la forma más humanizada posible. Sus inputs son la ruta, lectura de los beacons y, de ser necesaria, información de contexto tales como sentido, aceleración, sonido, etc. capturados a través de los distintos sensores del Smartphone.

Siendo una extensión del Modo Exploración de Lazarillo, la aplicación debe entregarle al usuario mensajes contextuales del entorno en la medida que éste se desenvuelve por el espacio. Dependiendo de la situación particular del espacio, dispositivo utilizado, tiempo de uso, velocidad del usuario, entre otras variables, el algoritmo encargado de generar los mensajes auditivos debe otorgar la mayor precisión posible de acuerdo a las variables antes descritas y posición anterior.

1.5.2. Entorno para desarrollo de pruebas piloto

Para el desarrollo de las pruebas pilotos se debe contar con un espacio interior acotado tal que cuente con un set mínimo de elementos de diverso tipo tales como pasillos, escaleras o ascensores, puertas, oficinas o tiendas, entre otros, tal que sea posible la generación de contenido. Para lo anterior se proponen dos entornos de pruebas:

1. En el contexto del proyecto de accesibilizar el Museo de la Memoria y los Derechos Humanos para PeSD visual, a desarrollar por Lazarillo en los próximos meses, se considera este como un lugar apto para el desarrollo de las pruebas toda vez que el mismo será abierto para este segmento de público.
2. Para la realización de pruebas preliminares a desarrollarse principalmente por el estudiante, se propone el OpenLab, perteneciente a la FCFM, para su habilitación para PeSD visual.

1.6. Resultados Esperados

Objetivo específico	Resultado
OE1. Designar servicio de localización en base método de posicionamiento por celdas impuesto por Lazarillo	R11. Tabla comparativa de servicios de localización indoor en base a método de posicionamiento por celdas a partir de análisis de Estado del Arte
OE2. Diseñar procedimiento para el levantamiento de información en espacios cerrados	R21. Modelo espacial en base a estructura jerárquica de grafos R22. Arquitectura de datos R23. Clasificación de mensajes
OE3. Desarrollar algoritmo que genere flujos de información contextual de localización	R31. Algoritmo de streaming de datos
OE4. Desarrollar prototipo funcional en forma de aplicación móvil para sistema operativo Android	R41. Interfaz de usuario compatible con PeSD visual R42. Arquitectura de datos montada e implementada R43. Módulo de lectura de Beacons capturada en tiempo real R44. Semánticas de accesibilidad en la aplicación
OE5. Evaluar usabilidad e impacto del prototipo desarrollado en base a método propuesto con PeSD visual	R51. Métricas evaluación de impacto R52. Métricas evaluación de usabilidad R53. Pruebas con PeSD visual, captura de información y análisis

1.7. Plan de Trabajo

El trabajo realizado se puede apreciar a continuación, a través de los resultados esperados y sus respectivas actividades asociadas. Los tiempos designados por tarea están indicados en la figura 1.2.

R1 Elaboración de informe EA

A1 Estudio uso de tecnologías asistivas

A2 Estudio HCI para PeSD visual

A3 Estudio de servicios de localización en base a posicionamiento por celdas

R2 Diseño de modelo espacial en base a estructura jerárquica de grafos

A4 Definir nodos

A5 Definir arcos

R3 Definición de arquitectura de datos

- A6 Definir propiedades
- A7 Definir acciones
- R4 Clasificación de mensajes
 - A8 Generar mensaje auditivo de elementos de navegación
 - A9 Generar mensaje auditivo contextual para los elementos de navegación
- R5 Método de estimación de prioridad de elementos en mapa
 - A10 Definición de características relevantes
 - A11 Asignación de pesos relativos
- R6 Desarrollo de algoritmo de streaming de datos
 - A12 Creación de pseudo código
- R7 Diseño e implementación de interfaz de usuario compatible con PeSD visual
 - A13 Diseño de mockups
 - A14 Desarrollo de layout en Android Studio
- R8 Implementación y carga de datos según arquitectura planteada
 - A15 Desarrollo de clases
 - A16 Carga de datos
- R9 Implementación de módulo de lectura de Beacons en tiempo real
 - A17 Análisis de documentación de API
 - A18 Implementación de módulo
- R10 Implementación de semánticas de accesibilidad en la aplicación
 - A19 Generación de cadenas de texto en algoritmo de generación de flujos de información
 - A20 Carga de datos
- R11 Definición de métricas de evaluación de impacto
 - A21 Definir KPI's de impacto de solución
- R12 Definición de métricas de evaluación de usabilidad
 - A22 Definir KPI's de evaluación de usabilidad
- R13 Elaboración e implementación de pruebas con PeSD
 - A23 Definir KPI's de evaluación de usabilidad
 - A24 Implementación de la prueba
 - A25 Análisis de resultados

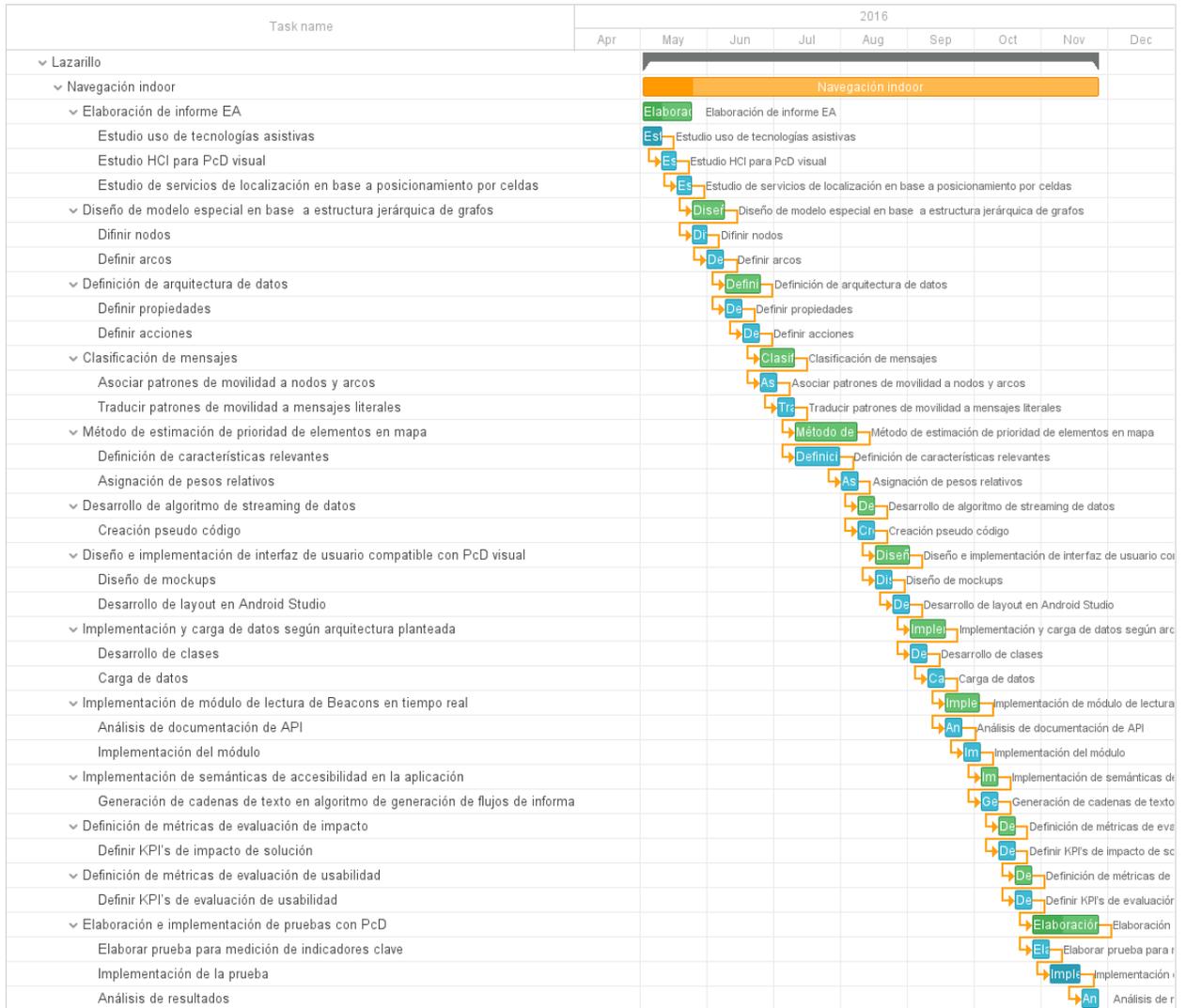


Figura 1.2: Carta Gantt

Capítulo 2

Marco Teórico

El desarrollo actual de técnicas y tecnologías que permitan estimar la localización de un individuo otorga un amplio abanico de posibilidades, sin embargo, cuando estas se deben restringir a la estimación de la localización en espacios cerrados y el uso de un dispositivo móvil, las posibles alternativas se limitan notablemente. A continuación se describen posibles tecnologías y técnicas existentes para el objetivo antes descrito.

2.1. Tecnologías de uso común en navegación indoor

2.1.1. Wifi

Esta no es una tecnología destinada para usarse directamente para estimar localización, sin embargo, ha sido y es ampliamente utilizada para ello en ambientes cerrados, por medio de la intensidad de la señal de un punto Wifi y su relación con otros puntos de acceso disponibles.

En [6] se implementa un sistema de localización para un Smartphone con sistema operativo Android, basado en el parámetro RSS que se recibe de los puntos de acceso Wifi. Se presenta un algoritmo que determina con bajo error la sección en que se encuentra la persona que posee el Smartphone en un área de 6m por 12m, constituida por 8 secciones. No obstante la exactitud, este algoritmo es limitado por cuanto necesita ex-ante una base de datos con los RSS promedio de cada sección, lo que haría impracticable su uso en edificaciones de gran tamaño, como un hospital, edificio educacional, centro comercial, entre otros.

En [7] se presenta una tesis de magister de la Universidad de Chile, guiada por uno de los investigadores de esta propuesta, en que se implementa un sistema de localización y orientación para PeSD visual (PYOM), en base a la intensidad de señal de puntos Wifi sobre un dispositivo móvil. Esta aplicación permite al usuario solicitar on-demand la localización. Al igual que en [6] también utiliza una base de datos de intensidad de señal previamente verificada. Se logra una buena aceptación de las PeSD visual respecto a su usabilidad, sin embargo, se requiere calibración y al menos cuatro puntos de acceso Wifi por habitación.

2.1.2. Global Positioning System (GPS)

Si bien es una tecnología que se utiliza para localización en ambientes abiertos, su mejoramiento ha permitido que actualmente en algunas zonas del interior de una edificación pueda detectarse sus señales. Es un sistema creado y mantenido por el gobierno de Estados Unidos, separado en tres segmentos: espacial, de control y de usuario. El segmento espacial corresponde a los satélites que orbitan la tierra a 20.220 km de altura; el segmento de control se encarga de mantener disponibles 24 satélites desde cualquier parte del mundo al menos el 95 % del tiempo [8] y el tercero corresponde al usuario que debe poseer un receptor de GPS al cual los satélites envíen sus señales y timestamps. Con estas marcas de tiempo y considerando que la señal viaja a la velocidad de la luz, se estima la posición del receptor. Todos los receptores GPS, a excepción de los militares, son GPS SPS, correspondientes al Servicio de Posicionamiento Estándar, el cual puede tener una precisión de 3 metros [9]. Sin embargo, esto es bastante inusual en ciudades, debido a los edificios, árboles y otros tipos de interferencia.

Como se dijo, esta tecnología es la más usada en ubicación en espacios abiertos y su potencial se explota cada día más debido que está incorporada en la mayoría de los Smartphones. Por lo general el error de localización en estos dispositivos móviles es peor que en los receptores especializados, teniendo a lo menos un error de alrededor 5 metros [10]. Las aplicaciones que guían a PeSD visual al aire libre requieren alta precisión para poder especificarles cuando realizar giros; sin embargo, por lo general las veredas o caminos de peatones se encuentran rodeados de edificios o árboles, lo que genera pérdida de precisión.

No obstante, con ciertas excepciones como se indicó anteriormente, los dispositivos GPS no pueden rastrear la información de localización en espacios cerrados o en áreas subterráneas.

2.1.3. Near Field Communication (NFC)

Near Field Communication o Comunicación de Campo Cercano, más conocido como NFC, es una tecnología de comunicación inalámbrica de corto alcance y alta frecuencia que permite un intercambio de datos de forma simple y segura entre dispositivos electrónicos. Esta tecnología cubre protocolos de comunicación y transmisión de datos basados en los estándares de RFID.

Se tienen dos modos de funcionamiento para la comunicación entre dispositivos que poseen la tecnología NFC:

- **Modo Activo:** Ambos dispositivos generan su propio campo electromagnético, que utilizarán para transmitir sus datos.
- **Modo Pasivo:** Sólo un dispositivo genera el campo electromagnético y el otro se aprovecha de la modulación de la carga para poder transferir los datos. El iniciador de la comunicación es el encargado de generar el campo electromagnético. Este modo es el utilizado para la comunicación entre tarjetas NFC y Smartphones con NFC.

En [11] se desarrolla un sistema de navegación para espacios cerrados en base a tags NFC, es decir dispositivos en modo pasivo que pueden ser tarjetas o stickers, los que se despliegan por múltiples zonas dentro de un edificio u otro ambiente cerrado y son leídos a través de un dispositivo

móvil que actúa como componente activo. Estos tags guardan internamente una URL o dirección a un servidor donde el dispositivo móvil descarga el mapa del lugar. Cada tag representa un punto en este mapa. Una vez que este se ha descargado, el usuario a través de su dispositivo móvil puede pedirle al sistema que lo guíe a un determinado lugar dentro de éste, y como el sistema tiene el último tag leído, sabe la posición de la persona y a la que desea llegar, por lo que calcula la ruta óptima entre ambos puntos usando el algoritmo de ruta más corta Dijkstra [12]. La lectura de otros tags, hace que el sistema se recalibre en caso que el usuario quiera otra ruta o no haya seguido instrucciones.

En [13] se desarrolla también un sistema de navegación para espacios cerrados, aunque este se enfoca a PeSD visual. Lo que tiene en común con el sistema anterior es que utiliza tags NFC y que el sistema da instrucciones a la persona también a través de un dispositivo móvil, que al pasarlo cerca de un tag NFC activa la aplicación. Sin embargo, este sistema no requiere nunca una conexión a un servidor.

Los tags se localizan en las manillas de las puertas, y por cada una de estas se instalan dos tags, uno en su parte superior, que tiene información sobre la localización de tags cercanos y otro en la parte inferior, que tiene información auditiva sobre la determinada pieza. La información de localización se describe en un código que la aplicación interpreta, para indicársela posteriormente al usuario. Por ejemplo “F5.5 Ur L R5” significa, “Sigue derecho en línea recta por 5.5 metros, dobla 90° a la izquierda, la pieza deseada es la quinta en el lado derecho del corredor”. Se hicieron pruebas con 8 personas ciegas donde tenían que localizar una determinada pieza en un hospital partiendo de la entrada del hospital, el cual tenía tags en su manilla. Todos los participantes lograron el objetivo en un tiempo promedio de 136 segundos. El mayor problema de este tipo de sistemas es que la forma de ubicación implica que la persona debe encontrar por sus medios el primer tag, que puede emitir información a una distancia no mayor de 20 centímetros.

2.1.4. Bluetooth

Esta tecnología fue diseñada para generar un estándar de comunicación inalámbrica de corto alcance. Mediante su desarrollo se han creado versiones Bluetooth de cada vez más bajo consumo, factor que ha motivado su uso como la tecnología indicada para realizar posicionamiento en espacios cerrados [10]. Su versión de más bajo consumo es el correspondiente a Bluetooth 4.0 o más conocido como Bluetooth Low Energy (BLE) y su versión más actual a la fecha corresponde al Bluetooth 4.1, anunciado el 4 de diciembre de 2013, el cual, se cree, dará el salto como tecnología habilitante para la Internet de las cosas [14].

La localización a través de Beacons Bluetooth en ambientes cerrados se puede realizar a través de métodos basados en el indicador de intensidad de señal recibida, o más conocido como RSSI (Received Signal Strength Indication) por sus siglas en inglés. En [15] se utiliza este indicador para estimar la posición de un Smartphone con Android a través del método de triangulación, en una pieza cerrada. El aporte es el método en que se realiza la calibración de los parámetros de la ecuación de propagación de la señal, con la que se determina la distancia, entre el emisor y receptor. Estos parámetros varían respecto a condiciones del ambiente, por lo que su recalibración mejora las estimaciones, aunque, por lo general, esta acción toma tiempo y requiere intervención de terceros.

Otro método alternativo utiliza los RSSI recibidos y enviados entre los Beacons, luego, mediante Least Square Error se definen los nuevos parámetros para la ecuación de propagación, de manera que el sistema se auto calibra sin intervención o trabajo extra de terceros.

En [16] se crea un sistema que se le llama Bluepass, el que también entrega la ubicación de un dispositivo móvil en un mapa y de forma visual a través de un área generada por cuadrados donde se estima dónde la persona se encuentra con el dispositivo. Además, se utiliza un modelo de propagación de la señal generado por la Unión Internacional de Telecomunicaciones (ITU).

A continuación se describe el estado del arte de métodos de posicionamiento en base a señales electromagnéticas, aplicables a la arquitectura establecida previamente, y de mayor uso en la actualidad:

2.2. Métodos de posicionamiento

2.2.1. Triangulación [1]

En el método de Triangulación se usan dos Beacons emisores y se miden los dos ángulos que se generan entre la línea de vista de cada uno de estos y el objeto que se desea detectar, además se debe saber la distancia y ángulos entre los Beacons. De esta manera, usando las propiedades de los triángulos, se puede obtener la distancia al objeto. Este tipo de localización no es adecuada a usar con Bluetooth, ya que no existe un método directo para obtener los ángulos de recepción de la señal.

2.2.2. Lateración [2]

La Lateración es muchas veces confundida con triangulación, sin embargo es un método en que se necesita medir la distancia a tres Beacons, los que deben estar ubicados en tres puntos no colineales. Con estas se generan tres círculos que luego se interceptan para obtener la posición.

Se puede utilizar la intensidad de señal recibida o RSSI para obtener una aproximación de la distancia entre el objeto y un Beacon. Esto debido a que la atenuación de la señal se puede modelar en función de la distancia; sin embargo, este modelo también depende mucho de los parámetros ambientales y de la antena del dispositivo receptor. Por ejemplo, para dos dispositivos que se encuentren a la misma distancia de un Beacon pero con distintas antenas, la intensidad de señal recibida será distinta.

2.2.3. Posicionamiento en base a celdas [3]

El posicionamiento en base a celdas determina la posición de un objeto de acuerdo a los Beacons que puede “ver”, considerando que cada Beacon puede propagar su señal en un rango lo

suficientemente acotado.

2.2.4. Fingerprinting [4]

El método estadístico de fingerprinting consiste en subdividir zonas en base a RSSI obtenidos en cada una. Para esto se deben hacer mediciones previas en todas las zonas que se quiera subdividir. Por ejemplo, en el caso de un Beacon emitiendo, se podría intentar realizar la subdivisión de su radio de emisión, como se muestra en la Figura 2.3. Como se puede ver en este caso un radio de emisión se divide en 3 zonas. Para poder realizar esta división se debe implementar un clasificador que en base a RSSI medidos pueda indicar en qué zona se encuentra el aparato; sin embargo, el funcionamiento del clasificador dependerá de la representatividad de los datos anteriormente medidos, lo que implica mucho tiempo, ya que para cada lugar nuevo en que se quiera instalar un Beacon se deberían hacer múltiples mediciones con distintos aparatos y con los posibles obstáculos que pudieran aparecer en el ambiente. De todos los métodos descritos el más estable y factible de realizar en base a la tecnología Bluetooth 4.0 para esta propuesta, es el método en base a celdas, debido a que el hardware que se utilizara para los Beacons, permite setear la intensidad con la que éstos emiten, acotando su radio de propagación. Además puede funcionar inclusive aunque los Beacons estén instalados de forma colineal.

2.2.5. Pedestrian Dead Reckoning

Esta tecnología permite la estimación del desplazamiento de una persona en base a las medidas obtenidas del acelerómetro y giroscopio, los que permiten determinar la cantidad de pasos de la persona y trazos de desplazamiento en función de la posición anterior. Este es un acercamiento distinto para la ubicación en espacios cerrados, ya que no necesita necesariamente de Beacons o módulos Wifi desplegados por el interior. Sin embargo, son recomendados para un calibramiento progresivo del algoritmo ya que intrínsecamente para cada estimación realizada el error crece de manera exponencial. Por otro lado, para ambientes abiertos es un buen complemento al GPS para mejorar su estimación de la posición [17].

2.3. Tecnologías asistivas

2.3.1. Algoritmos y Tecnologías de Ruteo de PeSD Visual

En esta propuesta, la creación de nuevos algoritmos y tecnologías para el ruteo de PeSD visual se centra en la dificultad de desplazamiento de este tipo de personas en zonas específicas de una edificación. En efecto, para el caso general de personas en situación de discapacidad en eventos de evacuación, diversos estudios [17, 18, 19, 20, 21, 22] hacen evidente los siguientes puntos: (1) dichas personas requieren tiempos de evacuación más largos que el promedio en todo tipo de superficies [18]; (2) se desplazan más lentamente que personas que no poseen esta condición [19]; (3) el ancho de las puertas es el factor más influyente en producir congestión y elevar los tiempos

de evacuación [20]. Por consiguiente, estos elementos nos permiten descartar algoritmos que no consideren estas restricciones o que lo realicen de forma parcial, como el caso de camino mínimo (Dijkstra).

En primera instancia se estudia el acercamiento que propone [21], consistente en un modelo que genera índices de accesibilidad para cada uno de los nodos de una estructura. Estos nodos representan niveles promedio de dificultad de acceso a cada una de dichas zonas desde todas las demás ubicaciones del edificio, considerando los efectos de distancia, tiempo y calidad de las rutas [26, 27, 28]. De esta manera, es posible calcular de manera agregada, por ejemplo, el índice de accesibilidad general de una estructura. Para el caso de una evacuación, [22] propone el término “egresabilidad”, el cual determina el esfuerzo necesario para salir del edificio desde alguna ubicación dentro de él. Para el cálculo de los índices de accesibilidad descritos anteriormente, [21] propone la creación de un modelo de grafos. Dado que el objetivo principal del cálculo de dichos índices es la creación de rutas, se propone la asignación de éstos a cada una de las aristas del grafo. Estas aristas corresponden a pasillos, escaleras, ascensores, puertas y rampas. Por otro lado, los nodos del grafo representan intersecciones de aristas y/o los términos de ellas. En el caso de una oficina, por ejemplo, se representa por un nodo dado que corresponde al término de la arista asignada a la puerta de dicha oficina.

Una vez construido el grafo de la estructura, se procede al cálculo de los índices de accesibilidad para cada arista. En este paso se definen diversas métricas por tipos de arista con valores fluctuantes entre 0 y 1, donde valores cercanos a 0 representan gran accesibilidad y valores sobre 1 representan lugares completamente inaccesibles. Por ejemplo, para el caso de los pasillos se considera su largo en comparación con otros pasillos del edificio, ancho, pendiente, cambios verticales y otros; en escaleras, si éstas poseen pasamanos, ancho, tipo de peldaños y otros; en puertas: el ancho, distancia de manilla, altura, tipo de puerta (giratoria, torniquete, barrera y todo tipo de entradas) y otros; en rampas: ancho, pendiente transversal y longitudinal, pasamanos entre otros elementos. Cada uno de los diversos elementos descritos anteriormente se recopila a partir de diferentes estudios y pruebas de laboratorio y se ponderan en una fórmula según su relevancia específica.

Para esta propuesta se requiere adaptar este enfoque a las necesidades propias de PeSD visual. En esta línea se propone estudiar las consideraciones expuestas por [23] con respecto al desarrollo de un algoritmo de ruteo considerando como foco: (1) evitar obstáculos para aumentar la seguridad del usuario; (2) preferencia en el seguimiento de puntos de referencia como ayuda para orientación; (3) disminución de, por ejemplo, paredes. No obstante lo anterior, el trabajo de estos autores es limitado por lo siguiente: (1) no se consideran todos los criterios requeridos para el cálculo de accesibilidad de la ruta, como en el caso de Hashemi y Karimi; (2) se requiere de un mapa preciso en 2D con la distribución completa de objetos a evitar, tanto estructurales como elementos de consideración que afecten la movilidad de dichas personas; (3) la clasificación de objetos como referenciales es arbitrario; (4) el algoritmo se hace cargo de enrutamiento en un nodo específico (por ejemplo una oficina, hall, dormitorio entre otras locaciones del edificio).

2.3.2. Modelo de Grafos para Estructuras de Edificaciones

Tanto los algoritmos de ruteo como la semántica de accesibilidad planteados de aquí en adelante, requieren de un método funcional de abstracción de estructuras, es decir, una representación

simplificada de la edificación. Para ambos casos esta propuesta recomienda el uso de un modelo de grafos como punto de partida, el cual permite tanto el manejo de la granularidad de la solución (se puede controlar el número de niveles de profundidad del grafo), como la definición de características para el conjunto de nodos y aristas del grafo, lo cual permitirá generar la semántica relacionada a la navegación por los espacios de la edificación. Con semántica nos referimos por ejemplo a la definición de mensajes de movilidad desde un nodo a otro, como “girar 90 grados y avanzar pegado a la muralla por cinco metros”, etc. A continuación, se describe una herramienta estado del arte de modelación de información arquitectónica que, con seguridad, a mediano plazo se establecerá como estándar del mercado.

2.3.3. Semántica de Accesibilidad

La generación de semánticas de accesibilidad proporcionará una comunicación efectiva entre el sistema ubicuo de esta propuesta y la PeSD visual, asemejándose al lenguaje humano y dando énfasis en la descripción del contexto por sobre la mera descripción de localización.

Con el propósito de definir rutas con una semántica de accesibilidad de alta precisión, en esta propuesta se adaptarán las recomendaciones sugeridas en [24], en que se propone un modelo consistente en: Un modelo espacial base con estructura jerárquica de grafos: Un edificio está compuesto de pisos; cada piso está compuesto de oficinas, escaleras y otros elementos; cada oficina a su vez está compuesto de diverso mobiliario. Se realizan tantos niveles como nivel de granularidad se estime necesario.

2.3.4. Clasificación de mensajes para la semántica de accesibilidad de espacios interiores

Para la representación de espacios indoor este enfoque se basa en el diseño jerárquico basado en grafos, cuyos nodos corresponden a ubicaciones (lugar, entrada, punto de interés) y las aristas a conexiones (la existencia de una arista particular depende de si existe la conexión entre el par de nodos en la vida real). Este diseño es compatible tanto con el modelamiento de la estructura del edificio como con la estrategia de navegación. Además, es una abstracción del fenómeno real fácil de entender por gente no instruida en el tema, por lo que su construcción no requiere de habilidades especiales.

2.3.5. Sistemas de Computación Ubicua para Asistir a PeSD Visual

El término computación ubicua nace de Mark Weiser en 1988, mientras trabajó como jefe de tecnologías en el centro de investigación Xerox en Palo Alto, y es considerada en el medio como la tercera era en el mundo de la computación. El Director de este proyecto enfoca su investigación en esta área. Este concepto consiste en que los computadores existen de manera ubicua e invisible, permitiendo entregar información y una serie de servicios computacionales en todo tiempo y todo lugar, de manera que las personas interactúen y los usen de manera transparente y no invasiva [25].

En [26] se presenta un sistema de computación ubicua para PeSD visual. Se propone el concepto de Chatty Environment, donde la idea detrás de esto es entregar la información visual que se encuentra alrededor, a través de otro medio. Este se define en base a discusiones con potenciales usuarios, que establecen dos grandes necesidades cuando se encuentran en ambientes nuevos. La primera, es la escasa percepción de su entorno, y la segunda, la necesidad de una navegación asistida a puntos de interés. Chatty Environment se enfoca en resolver ésta primera necesidad, dejando de lado la navegación asistida. En este trabajo se ejemplifica con un prototipo del sistema para asistir en la compra de productos en un supermercado, mediante la tecnología RFID, donde mediante un dispositivo móvil con lector RFID, se determina el ID (número identificador) de los productos, los que a su vez están asociados a un archivo XML en un servidor (local o en la nube), y posteriormente el dispositivo entrega de forma audible la información sobre los productos al usuario.

Una mejora del sistema Chatty Environment se presenta en [27], donde se especifica la interfaz entre el usuario y las etiquetas RFID, como también módulos Bluetooth. El dispositivo al acercarse al radio de transmisión de los módulos recibe un ID con el cual entrega la información al usuario. Además, este dispositivo permite leer la etiqueta RFID de un producto cuando el usuario lo acerca al lector RFID. De esta forma, la metáfora es un dispositivo que permite escuchar y sentir los objetos. En [28] se implementa un sistema de computación ubicua para ayudar a PeSD visual en el transporte público, específicamente en una parada de buses. Para lograr esta funcionalidad se creó la aplicación Ubibus, que permite a la PeSD visual pedir por adelantado el bus que desea tomar, notificación que se envía al conductor del correspondiente bus cuando se encuentra cerca del paradero. En este sistema la PeSD visual utiliza un dispositivo móvil con Bluetooth.

En [29] se desarrolla el sistema Talking-Points, el que tiene por objetivo permitir a las PeSD visual estar conscientes de las cosas que los rodean. Su implementación consiste en la instalación de un conjunto de Bluetooth Beacons en puntos de interés, que pueden ser entradas de tiendas o lugares informativos con respecto al entorno. Se desarrolla una aplicación para smartphone que es capaz de detectar estos beacons y recibir información de éstos. Cada uno tiene asociada información guardada en una base de datos, de manera que de pasar un smartphone lo suficientemente cerca, la recibirá. En [30] se implementa un sistema auxiliar perceptivo para PeSD visual, denominado Blueview. Este apunta a entregar una mayor percepción de puntos de interés que se encuentren en las cercanías de los usuarios y ayuda a localizarlos mediante señales auditivas. El sistema se conforma principalmente por dos elementos, un dispositivo móvil con Bluetooth y un Beacon Point (BP), que corresponde a un dispositivo con conectividad Bluetooth que además posee un parlante. De esta manera, los usuarios a través del dispositivo móvil pueden detectar los BP que se encuentren en su alrededor y directamente seleccionar alguno y activar su parlante para localizar dónde se encuentra instalado. El sistema fue probado con seis PeSD visual en un escenario de cuatro puertas con BP, ubicadas en un hall Universitario. Se realizaron los siguientes experimentos: los usuarios debían localizar una puerta en específico usando el sistema, considerando un usuario y multiusuario. En este último variando en dos factores, mismo BP con distinto audio o no, distinto BP con mismo audio o no. En todas las pruebas los usuarios fueron capaces de localizar las puertas, variando solamente el tiempo y número de veces en que se activó el parlante en el BP, obteniéndose los mejores resultados con el formato en que cada BP tenga un audio distinto y el peor en que todos los BP tengan el mismo audio. Sin embargo, la diferencia promedio en tiempo y número de activaciones fue de diez segundos y dos activaciones, respectivamente.

2.3.6. Uso de Tecnologías Asistivas

El desarrollo de las TICs ha permitido el avance de las tecnologías asistivas móviles, facilitando la independencia y seguridad, mejorando la calidad de vida de las PeSD visual, entregando soluciones que favorecen los procesos de inclusión y optimizando los desempeños de las personas en sus actividades de la vida diaria.

Desafortunadamente la pérdida de visión conlleva un impedimento en el acceso a la información, influyendo negativamente en la realización de actividades cotidianas. Las tecnologías táctiles/auditivas han resultado ser alternativas efectivas a la presentación visual tradicional de información [41,42]. Los beneficios de las tecnologías asistivas están bien documentados [43, 44, 45]. Las tecnologías asistivas móviles poseen la principal ventaja de tener el potencial de entregar información respecto a la localización gracias a la portabilidad. Los teléfonos móviles, particularmente los smartphones, tienen el atributo de ser usados masivamente, por lo que las personas que los utilizan como tecnologías asistivas sienten menos estigmatización, además son de fácil uso, por lo que no requieren una adaptación extra para poder utilizarlos [31].

Los smartphones utilizan una interfaz táctil en una pantalla lisa. Ante esto se puede pensar que las PeSD visual no podrían discriminar lo que se está seleccionando mediante sólo el tacto. Sin embargo, los sistemas operativos como Android e iOS han desarrollado aplicaciones que permiten a PeSD visual utilizar estos dispositivos para llamar, chatear y usar todo tipo de aplicaciones. Estas aplicaciones se denominan Talkback y Voiceover, de Android e iOS respectivamente. Talkback y Voiceover entregan mensajes por voz para describir el resultado de acciones que el usuario realiza, como abrir una aplicación. También, describe elementos que el usuario puede tocar, leer y recibir notificaciones o realizar lecturas acordes a acciones táctiles. El usuario puede utilizar gestos gráficos para desplazarse por las aplicaciones o activar acciones.

Para explicar estos gestos se utiliza el término “flick”. Hacer un “flick” es como se denomina la acción de hacer un deslizamiento rápido en una dirección por la pantalla con uno o varios dedos. Así una forma de realizar una exploración de lo que hay en pantalla es por medio de flicks a la derecha y flicks a la izquierda.

2.4. Soluciones existentes para ayuda desplazamiento PeSD visual

2.4.1. Asistencia a través de perros guía o lazarillos

Un perro guía, perro lazarillo o perro de asistencia es un perro adiestrado para guiar a aquellas personas ciegas o con deficiencia visual grave o para ayudarlas en los trabajos del hogar. El perro guía es uno de los tipos de perros de asistencia que existen, posiblemente el más antiguo y más conocido. Es el único reconocido legalmente hoy en día, hecho que le permite unos derechos y privilegios no reconocidos entre el resto de los perros como, por ejemplo, la aceptación obligatoria en el transporte público.

Durante el control de los deficientes visuales el perro debe tener capacidad para percatarse de peligros eventuales para el dueño debido a las barreras arquitectónicas, requiriéndose una capacidad bastante alta de inteligencia y nivel de entrenamiento avanzado del animal.

A pesar de que los perros pueden ser entrenados para evitar varios obstáculos, no son capaces de distinguir colores como el verde y el rojo, de forma que no pueden interpretar un semáforo.

2.4.2. Asistencia humana

Consiste en la ayuda de un tercero para asistir a una PeSD visual en su desplazamiento por la ciudad. Evidentemente esta solución es la que entrega mayor precisión, sin embargo, es una solución costosa tanto en términos monetarios como en términos de independencia.

2.4.3. Asistencia a través de aplicaciones móviles

Esta es una de las soluciones más comunes en la actualidad, apoyado principalmente por los avances tecnológicos que han permitido tener dispositivos móviles cada vez más potentes, económicos y al alcance de todos, sumado con el desarrollo de herramientas de accesibilidad, como lectores de pantalla, cada vez más intuitivos y de mejor respuesta. A continuación alguna de las aplicaciones de navegación más utilizadas por PeSD visual:

- Google Maps
- Blindsquare
- Lazzus

Capítulo 3

Estado actual de Lazarillo y relevancia del trabajo

Para comprender la relevancia del trabajo realizado, a continuación se muestra un resumen del estado actual de Lazarillo que permitirá al lector contrastar con un estado de Lazarillo que considere la implementación de la solución propuesta en el presente trabajo.

3.1. Acerca de Lazarillo

Lazarillo surge a partir de la necesidad de generar una solución inclusiva para que personas ciegas o de baja visión pudieran movilizarse a través de la ciudad en forma independiente, sin necesidad de acudir a terceros constantemente.

René Espinoza, actual Director Ejecutivo de Lazarillo, notó que era necesaria la creación de una herramienta efectiva para la sociedad, que brindara autonomía a las personas con discapacidad visual y permitiera también a las instituciones poder hacer uso de ella, volviendo inclusivos sus espacios. A través de sus experiencias trabajando en inclusión, René conoció a Miguel González, quien se dedicaba a adaptar puestos de trabajo para personas con discapacidad. Miguel es ciego de nacimiento y actualmente se desempeña como Director de Experiencia de Usuario en Lazarillo.

LazarilloApp es una aplicación guía desarrollada por Lazarillo, que por medio de mensajes de voz permite conocer la ubicación actual y servicios cercanos, informando por ejemplo de paraderos de buses, cafeterías, bancos, restaurantes, intersecciones de calles y más. También permite buscar destinos específicos y obtener indicaciones de cómo llegar a éstos por medio de distintos tipos de transporte. Lazarillo busca vincular a las personas con el lugar donde viven ayudándoles a mejorar su autonomía e independencia.

LazarilloApp actualmente funciona en todo el mundo y está disponible en 5 idiomas, utilizando las principales bases de datos internacionales para alimentar sus mapas, tales como Google Maps, OpenStreetMap y Foursquare, sumado a bases de datos propias generadas en conjunto con colaboradores o con instituciones y empresas que buscan hacer más accesibles sus instalaciones. Actual-

mente la aplicación se encuentra disponible en Inglés y Español, mientras tanto, se está trabajando en la traducción al resto de los idiomas.

Más allá de LazarilloApp, Lazarillo pretende ser la principal plataforma de información para personas con discapacidad visual, conectando a los usuarios con las instituciones y organizaciones relacionadas a la temática. Esto mediante el Portal Lazarillo que permite que se promuevan los servicios que entregan dichas instituciones, en conjunto con eventos, talleres y ofertas de trabajo. A futuro se busca que los usuarios y colaboradores puedan generar contenido para enriquecer y mejorar los mapas.

3.1.1. Misión

Contribuir a la calidad de vida y autonomía de las personas con ceguera y baja visión, generando herramientas que potencien la relación entre las personas e instituciones, haciendo al mundo un lugar más accesible y conectado.

3.1.2. Visión

Lazarillo busca ser la principal plataforma en torno a la discapacidad reconocida a nivel mundial.

3.2. Estado actual

3.2.1. Modelo de negocio

Hasta la fecha, Lazarillo se ha sustentado a partir de diversos fondos, concursos y apoyos¹ que le han permitido desarrollar LazarilloApp, presentando un crecimiento orgánico constante y contando con una comunidad de usuarios fieles a la aplicación, los que actualmente² alcanzan más de 4.300 usuarios y cerca de 18.000 descargas.

Se ha definido un modelo Freemium³, consistente en una aplicación con funciones basales que permiten el uso cotidiano de forma gratuita y, por otro lado, una serie de funciones Premium, solicitadas por los mismos usuarios a través del tiempo y uso, que mejoran la experiencia de uso de LazarilloApp. Este paquete de funciones tiene un costo mensual asociado cercano a los 5 dólares y ha sido obtenido a través de diversas encuestas y del trabajo personalizado del equipo. Se han definido 3 meses de desarrollo para dichas funcionalidades, luego de los cuales se ha acordado un tiempo de 1 a 2 meses de uso gratuito de éstas para incentivar su uso y dar a conocerlas a modo de enganche.

¹Entre los que destacan los fondos SSAF-s y SSAF-i de Corfo, apoyos del ISCI y concurso Inspira Senadis

²Hasta junio 2017

³Este modelo ha cambiado a partir de lo desarrollado en el presente trabajo. En la sección 3.2.2 se explica el nuevo modelo

Paralelamente a LazarilloApp se ofrece a la comunidad el Portal Lazarillo, un sitio que permite acercar a los diversos actores en un sólo lugar. Las instituciones y empresa privada pueden promover sus actividades, como talleres, eventos, ofertas de trabajo, entre otras, las cuales a su vez son informadas a los usuarios a través de LazarilloApp de acuerdo a sus intereses y contexto. Esto a su vez mejora la experiencia de los usuarios, genera comunidad y, a largo plazo, afianza la fidelización con el usuario y clientes.

Más información acerca del modelo se muestra de forma gráfica a través del Social Lean Canvas de Lazarillo de la figura 3.1.

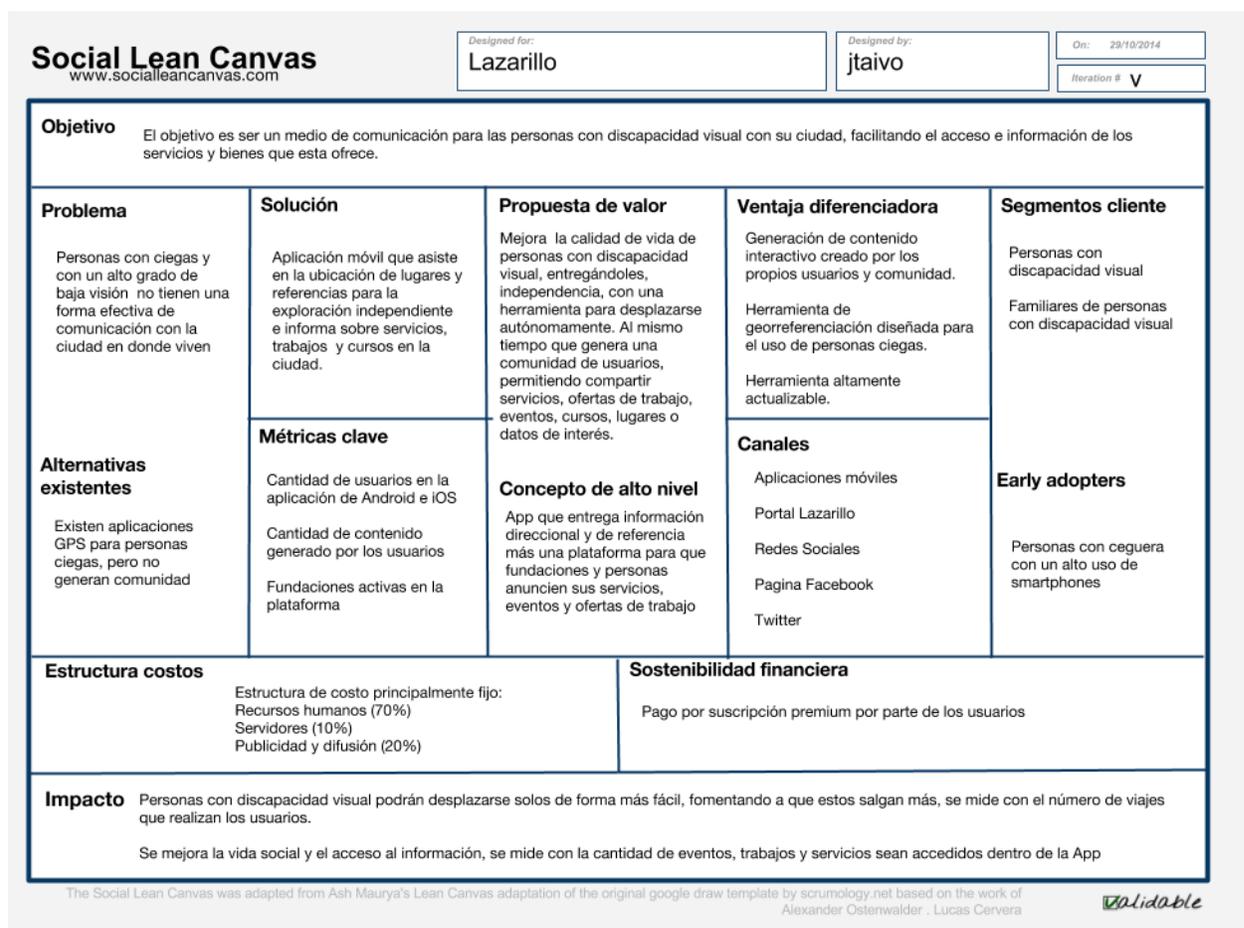


Figura 3.1: Social Lean Canvas Lazarillo

3.2.2. Análisis estratégico

Fortalezas

- Lazarillo cuenta con capital humano y tecnología de alto nivel, que permite estar en renovación constante, siendo lo suficientemente flexible para adaptarse a los requerimientos de sus usuarios y los cambios propuestos en el mercado.
- Lazarillo posee una comunidad fiel y cercana, que les permite tener una retroalimentación constante, para ir corrigiendo y agregando nuevas funciones al sistema.

- Instituciones especializadas en el área como Senadis, Cetram, OculabUc, entre otras, confían en Lazarillo y promueven su uso tanto a potenciales usuarios de LazarilloApp, como a instituciones que puedan hacer uso del Portal Lazarillo, lo que le ha permitido a Lazarillo generar alianzas y proyectos.

Oportunidades

Actualmente las PeSD visual tienen a su disposición diversas aplicaciones que les son de utilidad en distintos aspectos de la vida cotidiana, como de ayuda para el desplazamiento, otras para la utilización del transporte público o para solicitar un móvil, entre otras. Lazarillo tiene una gran oportunidad de crecimiento al integrar diversos servicios y funcionalidades en una sola aplicación que se transforme en el "top of mind"⁴ de los usuarios. Para lograr lo anterior, Lazarillo debe apuntar sus esfuerzos en aplicar innovación de tipo incremental, es decir, crear valor sobre un producto ya existente, pero añadiéndole nuevas mejoras.

Debilidades

Si bien el modelo antes descrito es escalable y aplicable a escala mundial, una de las debilidades que presenta es que este es solo sustentable a largo plazo, una vez alcanzada una masa crítica considerable de usuarios. A diferencia de otras aplicaciones que apuntan a públicos más masivos, Lazarillo ha presentado un crecimiento orgánico constante, en desmedro del uso de publicidad a través de canales tradicionales de comunicación, que han demostrado no ser efectivos en este mercado, lo que ha dificultado el crecimiento explosivo de usuarios por medio de inversión publicitaria.

Amenazas

Dada la dificultad para generar usuarios en este mercado, la posibilidad de que nuevos competidores prosperen es un escenario poco factible, sin embargo, la posibilidad de que competidores actuales comiencen a innovar y ofrezcan un mejor servicio partiendo desde su base de usuarios es una amenaza real.

3.2.3. Análisis del microentorno

Amenaza de nuevos entrantes

Las barreras de entrada existentes para LazarilloApp son altas debido a que existe un alto grado de fidelización con los usuarios. La oferta de aplicaciones en este mercado es pobre y, en muchos casos, de baja calidad o con una tasa de actualización e innovación deficiente. A través de la aplicación, el usuario puede generar su propio contenido, como lugares marcados en favoritos

⁴Primer servicio que surge en la mente del consumidor al pensar en una industria en específico

o configuraciones personalizadas para cada persona, que le hacen difícil la decisión a un usuario para migrar a probar suerte con otras aplicaciones. Adicionalmente a lo anterior, si bien existen varias aplicaciones que ayudan a PeSD visual en su desplazamiento, el funcionamiento entre ellas es diferente y su uso continuo va generando costumbre y, a largo plazo, dependencia, sobre todo en personas mayores que no están dispuestas a aprender nuevas tecnologías.

Amenaza de productos sustitutos

Actualmente la asistencia humana es el único servicio sustituto para LazarilloApp, dado que cumple con funciones y cualidades similares a la aplicación, entregando precisión perfecta pero restándole libertad a la PeSD visual y muy fuera del alcance monetario de la mayoría de la población de PeSD visual -según la OMS cerca de un 90% de la población de PeSD visual se concentra en países de bajos ingresos [?]-.

Por otro lado existen productos complementarios, como lo son los perros guías, las guías podotáctiles, los mapas táctiles y guías braille que ofrecen una ayuda adicional, que les permiten a las PeSD visual poder desplazarse con más facilidad mejorando la experiencia.

3.3. Consecuencias del trabajo

3.3.1. Extensión del modelo de negocio

De acuerdo al análisis anterior, la habilitación de espacios interiores se presenta como una oportunidad para Lazarillo de mejorar y ampliar su servicio, generando mayores barreras de entrada y aumentando la fidelización con los usuarios a través de innovación incremental. Pero tal vez uno de los mayores beneficios de contar con este nuevo servicio es la posibilidad de generar una extensión al modelo de negocios antes planteado, basado solo en un modelo Freemium a uno que incluye servicios prestados a instituciones.

A partir del presente trabajo y del interés generado en este desarrollo de empresas que deseaban contar con él, es que se itera sobre un nuevo modelo de negocio dejando atrás el modelo basado en una aplicación freemium a uno como el mostrado en la figura 3.2, basado por una parte en una aplicación -LazarilloApp- de uso gratuito para los usuarios y, por otro lado, servicios de accesibilidad ofrecidos a empresas e instituciones públicas de atención masiva de clientes a través de 3 modalidades explicadas a continuación:

Lazarillo For Business

Genera información de calidad y accesible sobre comercios, bancos y lugares que atienden público, permitiendo a este tipo de lugares no solo ser reconocidos en los mapas de LazarilloApp, sino también entregar de forma accesible toda la información relevante para que las personas con discapacidad visual puedan interactuar con sus sucursales y acceder a sus productos y servicios.



Figura 3.2: Nuevo Modelo de Negocios Lazarillo

Lazarillo Outdoor

Permite el desplazamiento autónomo en espacios abiertos de gran extensión, mucha información y puntos de interés, como parques o zoológicos y eventos, transformándose en un sistema de posiciones interconectados que amplía las posibilidades de LazarilloApp al permitir al lugar guiar al usuario por este con información interrelacionada.

Lazarillo Indoor

Entrega autonomía en el desplazamiento dentro de edificios, donde el GPS no funciona, gracias a la instalación de dispositivos electrónicos, entregando información de orientación y contextual. Este servicio está orientado a empresas e instituciones con instalaciones que requieran el desplazamiento de las personas en su interior, permitiendo accesibilizar sus espacios tanto en desplazamiento como en acceso a información para personas con discapacidad visual.

Desde la fecha en que se comenzó el presente trabajo se han realizado 2 ventas en blanco de habilitación indoor a instituciones bancarias, junto con proyectos conjuntos como el desarrollado junto con SENADIS y USACH para el Museo de la Memoria. Todos los cuales han surgido del interés propio de dichas instituciones, bajo recomendación de SENADIS. Por otro lado, se han generado ventas y alianzas importantes a partir del desarrollo de esta investigación, no directamente relacionadas con el uso de beacons sino de geopuntos explicados en la sección 5.2.4, entre las que destaca la realizada con Lollapalooza, evento realizado en las fechas 1 y 2 de abril del 2017.

De esta forma, el nuevo modelo ha quedado validado comercialmente en el segmento de empresas. El siguiente paso es lograr simplificar el proceso de habilitación indoor dentro de la plataforma, de manera que pueda ser ofrecido directamente a familiares y amigos de PeSD visual que quieran realizar habilitaciones en sus hogares a través de paquetes de auto instalación. El beneficio de esto es que es un modelo altamente escalable, con potencial de venta alrededor de todo el mundo.

Capítulo 4

Servicio de localización

Se ha definido previamente el uso de dispositivos emisores Bluetooth de bajo consumo, conocidos como beacons, como instrumento que le permita al usuario, a través de su teléfono móvil, el reconocimiento de los diversos elementos en un espacio cerrado.

Lo que resta por definir a continuación es que protocolos y técnicas son los más adecuados para el propósito de elaborar un sistema de navegación indoor.

4.1. Protocolos de beacons

En la actualidad existen tres protocolos para beacons, cada uno de los cuales cuenta con diversas características que los diferencian y que se describen a continuación.

- **iBeacon:** Es un protocolo creado por Apple, que se introdujo por primera vez en la Worldwide Developers Conference 2013. Apple fue la primera empresa que hizo conocida esta tecnología a nivel mundial, pero la tecnología (BLE) fue creada por Nokia.
iBeacon utiliza BLE para transmitir un identificador único universal (UUID) que es recogido por una aplicación o sistema operativo compatible con el protocolo. El identificador más otros bytes enviado se pueden usar para identificar la posición física del dispositivo, o lanzar acciones basadas en la localización como notificación push, etc.
- **AltBeacon:** Protocolo desarrollado por Radius Networks. La especificación de Altbeacon define el formato de los mensajes broadcast por proximidad de los beacons. Es gratuito es open-spec. Viene a ser una respuesta al protocolo cerrado iBeacon propiedad de Apple, cubriendo las mismas funcionalidades iBeacon puede ofrecer, pero no tiene un amplio apoyo aún.
- **Eddystone:** Es el proyecto de código abierto de Google para beacons. Google con esta tecnología pretende fomentar el internet de las cosas. Similar al protocolo de iBeacon pero open source. iBeacon está soportado oficialmente por los dispositivos iOS solamente, Eddystone tiene soporte oficial para iOS y Android.

Está diseñado para soportar múltiples tipos de paquetes de datos, a partir de Eddystone-

UID y Eddystone-URL. Hay un tercer tipo de paquete de telemetría Eddystone-TLM. Este paquete se emite junto con Eddystone-UID o Eddystone-URL y contiene el estado de salud del beacon como por ejemplo, la duración de la batería.

Eddystone se basa en un método único en este momento: Eddystone Discovery que es similar a iBeacon Ranging. Proporciona estimaciones de proximidad y solo funciona cuando está activa.

4.2. APIs de conexión

Tanto AltBeacon, iBeacon, Eddystone y los diversos proveedores de beacons como Onyx Beacon, Estimote, Kontakt, entre otros, desarrollan sus propias API's con diversas características. En el caso de estos últimos, dichas API's permiten inicializar y configurar los beacons¹, en su mayoría permitiendo emisión utilizando Eddystone o iBeacon son sólo compatibles con su propio hardware, lo que implicaría que todo desarrollo de soluciones que utilicen sólo alguna de ellas generaría una dependencia poco deseable con el proveedor. De ser estrictamente necesario se recomienda contar con implementaciones de dichas API's para los proveedores más importantes, pero evidentemente esta alternativa implica más tiempo de desarrollo. Por suerte, para el caso particular que se desea realizar, la mayoría de las funcionalidades de utilidad están cubiertas por las API's de AltBeacon, iBeacon y Eddystone, las que permiten trabajar con un espectro más amplio de hardware.

En la tabla 4.1 se resume una comparativa entre los principales protocolos utilizados en la actualidad.

Compatibilidad	Android y nativo en iOS	Cross-Plataform
Propiedad	Software propietario	Open Source
Facilidad de uso	Fácil de implementar	Flexible pero más complejo
Paquetes transmitidos	Un paquete que incluye: id, UUID, major y minor	3 paquetes: URL, TLM y EID
Comunidad	Estable	En crecimiento

Tabla 4.1: Tabla comparativa iBeacon y Eddystone

4.3. Eddystone e iniciativas abiertas

El caso de la ciudad de Amsterdam² es un ejemplo de lo que se vislumbra en corto plazo en el contexto de IoT y Smart Cities. Se está desarrollando un plan de iluminar la ciudad con beacons distribuidos en diversas partes del mapa, con información acerca de servicios de utilidad para las personas. Actualmente existen 452 beacons instalados y funcionando en la ciudad. Lo relevante de la iniciativa es que dichos dispositivos se encuentran configurados con protocolo Eddystone lo que

¹ en donde se incluye la lectura de datos telemétricos como estado de la batería, temperatura entre otros, dependiendo del proveedor

² <http://iotlivinglab.com/projects/amsterdam-beacon-mile/>

le permitirá a los desarrolladores que así lo deseen hacer uso de estos, utilizando la información ya guardada en un set de attachments asociados previamente o generando información de conveniencia, agregando los attachments necesarios de acuerdo a la aplicación que se esté construyendo. Para Lazarillo este tipo de iniciativas le permiten habilitar espacios abiertos con un mayor grado de precisión, mejorando notablemente el desempeño de la aplicación y, por ende, una mejora sustancial al servicio.

Otro ejemplo de lo que se puede lograr realizar con el protocolo Eddystone es el proyecto de balizas Waze³, cuyo objetivo es permitir a diversos asociados la instalación de beacons en espacios subterráneos de desplazamiento vehicular, presentándose como una solución escalable y asequible para habilitar una perfecta navegación subterránea tanto para usuarios de Waze como para desarrolladores que quieran hacer uso de los dispositivos para la creación de aplicaciones vinculados a ellos.

Como estas, existen otras tantas iniciativas similares que dejan a Eddystone como el protocolo por excelencia para este tipo de desarrollo. El modelo de negocios de Lazarillo no se sustenta en la comercialización de beacons, por lo que si un tercero realiza la instalación de estos dispositivos en un espacio con otros fines, por ejemplo de marketing, Lazarillo podría hacer uso de ellos en el caso que estén configurados para emitir en Eddystone y cobrar el servicio asociado, por lo que su uso tiene una ganancia implícita asociada para todos los participantes.

4.4. Hardware a utilizar

De acuerdo a lo anterior, los beacon a utilizar deben ser compatibles con el protocolo Eddystone de Google y cumplir con requerimientos mínimos que permitan su utilización tanto en espacios abiertos como cerrados. En este sentido, se privilegiarán aquellos dispositivos que sean aptos para su uso a la intemperie -soportando posibilidad de lluvias, vientos intensos y el desgaste natural del sol- y, por otro lado, dispositivos que no rompan la estética de espacios interiores y permitan camuflarse en el entorno. Deben ser autónomos, sin la necesidad de una conexión a internet ni de fuentes de energía -o al menos que sea de uso opcional, para proyectos que así lo requieran- salvo el uso de una batería interna que les permita permanecer emitiendo señal durante años sin realizar reposiciones continuas.

4.4.1. Especificaciones técnicas Onyx Beacon Enterprise

Protocolo: Eddystone, Bluetooth 4.0 Seguridad: Security (128 bit AES) Fuente de poder: Batería AA Resistencia al agua: IP65 Dimensiones: Diámetro 134mm Ancho: 25mm Instalación: Cinta 3M, tornillos Certificaciones: FCC, CE, RoHS,IP65 Color: Blanco

Este dispositivo ha sido seleccionado para el uso en espacios abiertos ya que presenta una carcasa dura y resistente al agua, además del uso de baterías AA de fácil reemplazo.

³<https://www.waze.com/es/beacons>



Figura 4.1: Beacon Onyx modelo Enterprise

4.4.2. Especificaciones técnicas Kontakt Beacon Pro

Protocolo: Eddystone, Bluetooth 4.0 Fuente de poder: 3 baterías CR2477 de 1000mAh Dimensiones: Alto 69mm, Ancho 69mm, Grueso 21mm Color: Blanco

Este dispositivo cuenta con un diseño atractivo y tamaño pequeño especialmente diseñado para espacios interiores. Posee una serie de sensores adicionales al Onyx que permitirían desarrollar nuevas funcionalidades a futuro.

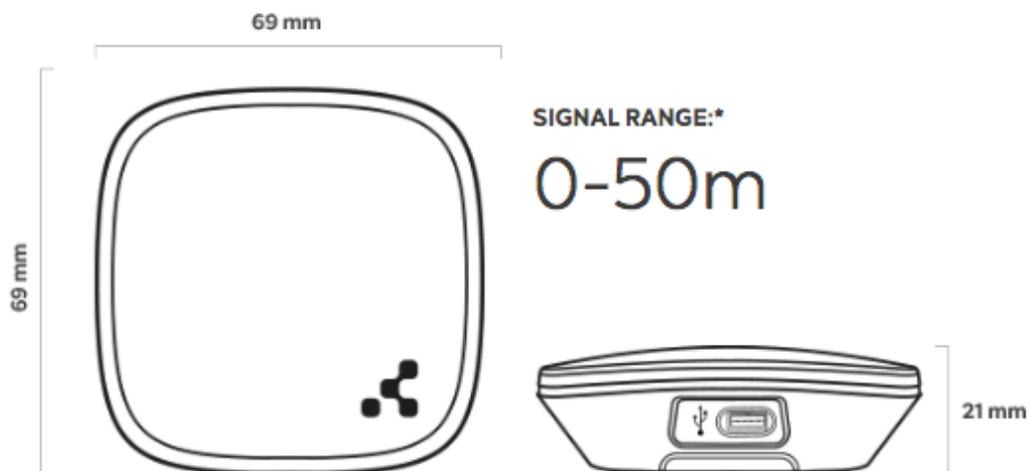


Figura 4.2: Beacon Kontakt

Capítulo 5

Algoritmo de flujos de información

Tal como se ha indicado anteriormente, el sistema de navegación indoor debe cumplir con ciertos requisitos definidos por Lazarillo de acuerdo a su modelo de negocios:

- El sistema debe dar la flexibilidad necesaria tal que pueda ser comercializado a través de kits de instalación, en la cual el cliente tenga la posibilidad de instalar y configurar los dispositivos en el espacio definido.
- Debe adaptarse a distintos entornos tales como oficinas, locales comerciales, hogares, entre otros ofreciendo un funcionamiento similar en todos los casos.
- Debe estar preparado para funcionar tanto en espacios de pequeño tamaño¹ como en espacios amplios².
- La forma en que la información es suministrada debe ser coherente con el funcionamiento actual de Lazarillo en espacios abiertos, de modo que para el usuario la transición entre ambas modalidades de funcionamiento se realice de forma transparente.

5.1. Funcionamiento actual de LazarilloApp

LazarilloApp es una aplicación móvil de orientación inteligente para personas en situación de discapacidad visual, disponible para celulares con sistema operativo Android (4.3 en adelante) e iOS (9.0 en adelante). Actualmente hace uso de 3 diferentes fuentes de datos: Google Maps, OpenStreetMap y Foursquare, las cuales son accedidas haciendo uso de datos móviles y/o WiFi para obtener información acerca de los lugares cercanos de acuerdo la posición registrada por el GPS del dispositivo móvil y con una API propia para consultas más complejas. Cuenta además de una interfaz gráfica accesible por los lectores de pantalla de los celulares (Talkback y Voiceover) e incorpora un serie de funcionalidades de carácter gratuito que se listan a continuación.

- Modo Exploración: Por medio de mensajes de voz el usuario será notificado del entorno que se encuentre a su alrededor, constituyendo esto, calles por la que transita, intersecciones que

¹De aproximadamente 3 beacons

²Más de 20 beacons

cruza y distintos servicios públicos y privados. Por ejemplo: Paraderos de buses, Entradas de estaciones de metro, Sucursales de Banco, Cajeros, Tiendas, Cafeterías, Restaurantes, Estaciones de Policía o Bomberos, Farmacias, Centros de Salud, Centros Educativos y Edificios Públicos. A esto se le suma los lugares favoritos del usuario que el haya creado y cualquier tipo de información localizada que el Algoritmo de Lazarillo estime útil para el usuario.

- **Búsqueda por categorías:** Por medio de botones accesibles, el usuario puede realizar búsqueda de servicios divididos por categorías en sus alrededores. Obteniendo una lista ordenada por prioridad considerando cercanía y accesibilidad. Entre las búsquedas que puede realizar son: Transporte, Bancos y cajeros, Salud, Comida, Tiendas, Arte y Entretenimiento, Edificios públicos, Pubs y discoteques. A estas categorías es posible que se agreguen más o menos en un futuro.
- **Favoritos:** Los usuarios pueden agregar los lugares que Lazarillo enuncia o encuentra a su lista de favoritas para acceso directo más rápido. Además el usuario puede crear sus propios lugares favoritos en posiciones GPS que este quiera marcar, entregando un nombre, categoría y descripción.
- **Medios de enrutamiento:** Cuando un usuario selecciona un lugar, tiene la posibilidad de pedir que se le den indicaciones para llegar a este, caminando, en transporte público, en auto y en algunos lugares en Uber. Cuando el usuario selecciona el medio, se abre la aplicación correspondiente que junto a Lazarillo le darán indicaciones al usuario para llegar al destino.
- **Configuraciones de Audio:** Los usuarios pueden modificar las voces e idiomas con la que Lazarillo habla.

Este sistema permite mejorar la independencia, otorgar accesibilidad a las personas con discapacidad visual en espacios abiertos, además de colaborar con la inclusión.

Actualmente se está trabajando para el desarrollo de Lazarillo Premium que consiste en una serie de funcionalidades extras que se agregaran a la aplicación y se podrá acceder a estas bajo una suscripción. Algunas de las funciones consideradas en esta versión son:

- **Modo Ruteo:** Guía para llegar de un punto específico a otro sin abrir aplicaciones extras a Lazarillo.
- **Comandos de voz:** Entregar instrucciones por voz a la aplicación, a través del micrófono o manos libres, sin tocar el Smartphone.
- **Navegación Offline:** Opción para descargar en el Smartphone mapas de Lazarillo de zonas específicas, para poder utilizar la aplicación sin conexión a internet, en esa zona.
- **Filtros avanzados** para configurar el tipo de lugares o información que se entrega y prioriza, al utilizar Lazarillo en modo exploración.

Lazarillo tiene un funcionamiento complejo que se puede apreciar en la figura 5.3. Sin ahondar en los detalles del funcionamiento actual, es importante recordar que todo el desarrollo del funcionamiento en espacios interiores debe ser compatible con el funcionamiento en espacios abiertos, de modo que convivan indistintamente y cuyos mensajes generados en ambos entornos sean tratados por el enunciador de manera similar. Por otro lado, el comportamiento desde el punto del usuario debe ser similar haciendo transiciones de uno a otro entorno de la forma más transparente posible.

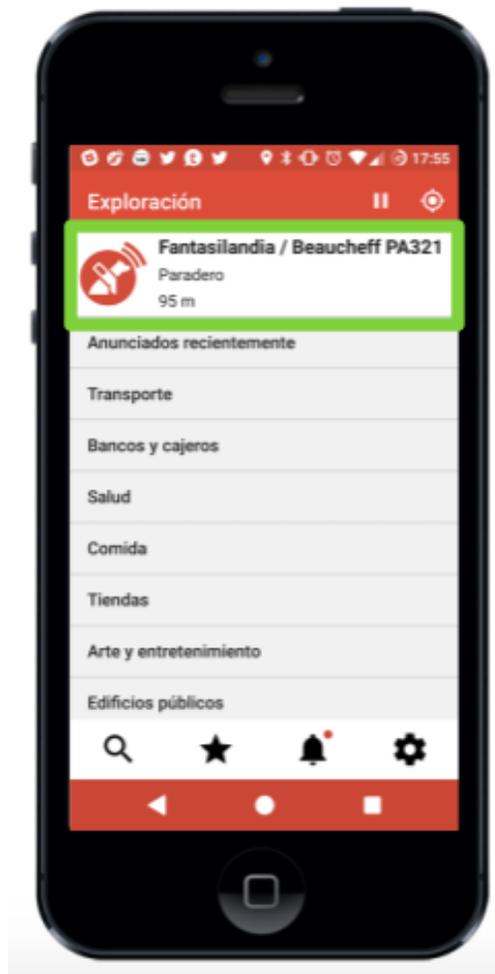


Figura 5.1: Vista principal LazarilloApp, Modo Exploración

5.2. Exploración en base a rutas predefinidas

Un primer enfoque para habilitar un espacio interior consiste en lograr dividir la superficie total en diversos subespacios identificables entre sí o celdas. Esta técnica se conoce como método de posicionamiento por celdas y consiste en la intersección de emisiones producidas por un cierto número de beacons distribuidos por el lugar. El conjunto de beacons activos en los diversos puntos del espacio va configurando cada una de las celdas, las que al haber sido previamente asociadas a una posición en el lugar, permiten determinar la posición del usuario/dispositivo móvil en el espacio al asignarle alguna de las celdas. El tamaño de dicha celda puede variar en función de la distribución y densidad de beacons en el espacio. Otros factores como el fabricante, tipo de dispositivo móvil utilizado y otros pueden afectar en la estabilidad de la señal. De esta forma, teniendo la posición relativa de los elementos en el espacio, se pueden generar mensajes contextuales propicios para la navegación indoor. Adicionalmente, al disponerse información acerca de la celda activa en función del tiempo, es posible también conocer los cambios de celdas activas y, por ende, la dirección de desplazamiento del individuo lo que es de utilidad para corregir desplazamientos incorrectos con relativa precisión.

Dado el funcionamiento en base a señales electromagnéticas de los beacons, su área de cobertura



Figura 5.2: Vista de lugar seleccionado por usuario

no posee una naturaleza estática ya que diversos elementos del ambiente afectan su funcionamiento ideal, como la estructura misma del espacio, el hardware tanto del dispositivo móvil como del beacon, la altura a la que se lleve el dispositivo móvil y otras causas menos evidentes como, por ejemplo, si el dispositivo móvil es portado dentro de un bolsillo o en el exterior, ya que el mismo cuerpo humano afecta las ondas al estar compuesto de agua. Por esta razón, para obtener un mejor resultado utilizando el método de posicionamiento por celdas, se sugiere emitir a distancias más acotadas contando adicionalmente con un mayor número de beacons en un espacio a habilitar, incumpléndose así la condición de escalabilidad.

Un enfoque compatible con las condiciones fijadas al comienzo del capítulo es uno que llamaremos de exploración por rutas predefinidas. La idea básica es similar a la utilizada por la plataforma Indoor Atlas³ en la cual se definen previamente rutas de tránsito en el espacio, las que posteriormente son registradas con un dispositivo móvil utilizando diversas técnicas de IPS como análisis de ondas magnéticas, ondas de radio, señales de audio y otras señales capturadas por el sensor del dispositivo móvil, configurándose así una “huella digital” de cada uno de los trazos de la ruta. Posteriormente, si algún dispositivo detecta un comportamiento similar a alguna de las “huellas digitales” anteriormente registradas, se asume con un cierto margen de error que el dispositivo se

³exploración y navegación en espacios cerrados utilizando indoor positioning systems (IPS)

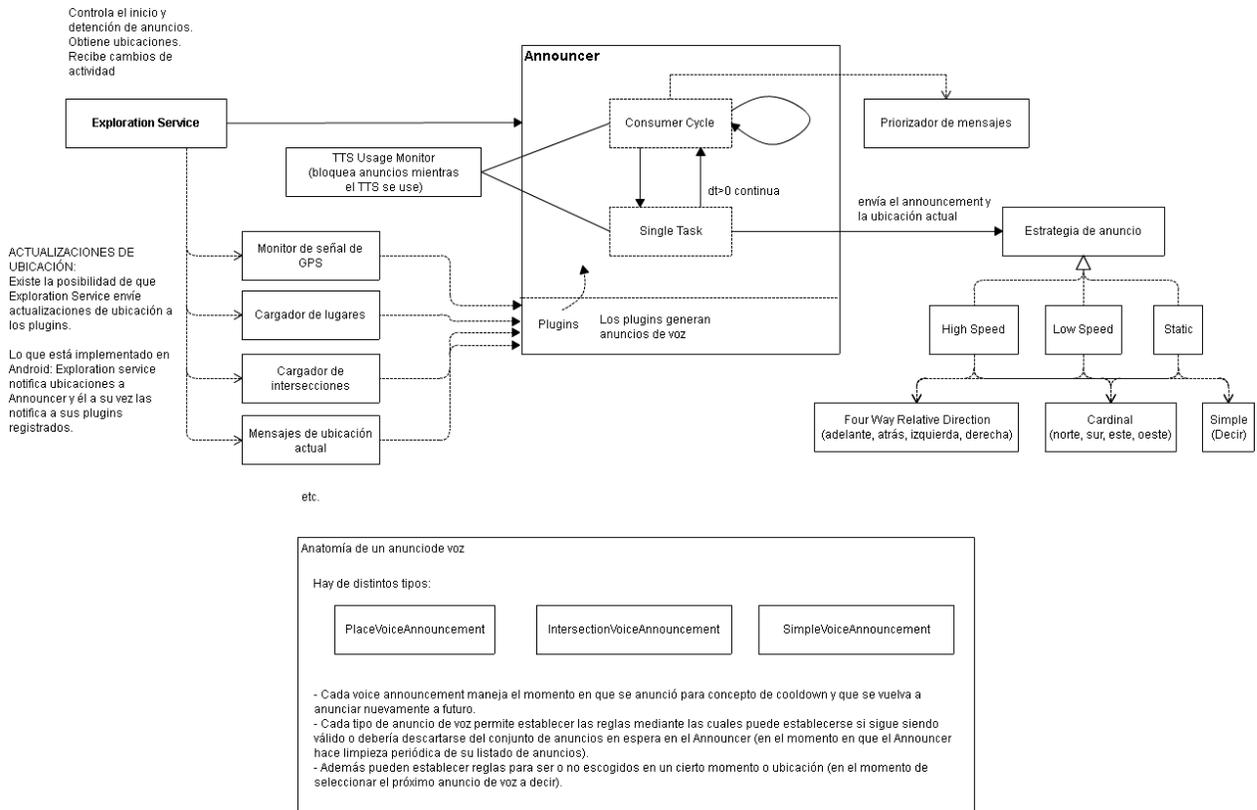


Figura 5.3: Funcionamiento Lazarillo

encuentra en un tramo conocido. La solución propuesta en el presente trabajo es diametralmente más sencilla en su implementación ya que no requiere de una grabación previa, pero requiere la instalación de beacons en diversos puntos de interés dentro del espacio emitiendo a corta distancia para dar precisión. El funcionamiento detallado se explica a continuación a través de un ejemplo.

5.2.1. Modo Exploración y Modo Ruteo

Antes de comenzar es importante realizar la diferenciación entre el Modo Exploración -que es el objetivo principal del trabajo- y el Modo Ruteo, ambos modos utilizados en Lazarillo.

Modo exploración

El Modo Exploración corresponde a un modo de desplazamiento libre, en la cual la PeSD visual no indica con antelación a Lazarillo la ruta que desea recorrer y, por consiguiente, ésta debe informar acerca de los elementos relevantes cercanos para ayudar a construir una “imagen” del entorno. En espacios abiertos, este modo tiene varios desafíos, como por ejemplo: determinar que elementos del entorno son relevantes dada la gran cantidad de potenciales elementos a comunicar, extraídos de las diversas fuentes de datos; los distintos submodos de funcionamiento de acuerdo al contexto -si la persona camina o si la persona se desplaza a través de transporte público por ejemplo-; comunicar información contextual en condiciones de muy baja precisión, entre otros. En

espacios cerrados, el desafío es diametralmente opuesto: se requiere generar métodos y técnicas que permitan entregar información contextual en condiciones haciendo uso de la poca información disponible.

Modo Ruteo

El Modo Ruteo puede tender a confundirse con el enfoque de rutas dirigidas planteado en la presente sección. El primero corresponde a una planificación realizada por la PeSD visual, en la cual ella indica a Lazarillo que desea realizar un desplazamiento desde un punto A (donde se encuentra actualmente posicionada) hasta un punto B. En este contexto, el Modo Ruteo le dará las indicaciones necesarias a la PeSD visual para irse desplazando en cada tramo de la ruta hasta llegar al punto B. Este modo es independiente del Modo Exploración, dejando la opción al usuario de escoger si desea realizar el recorrido utilizando solo el Modo Ruteo o en conjunto con el Modo Exploración. En base a feedbacks recibidos por los propios usuarios, es posible indicar que se prefiere esta última opción.

Hasta el momento del presente trabajo, Lazarillo no cuenta con un Modo Ruteo nativo, por lo cual se utilizan servicios externos para que realicen esta tarea como Google Maps y Waze, vinculados a través de la misma aplicación. Actualmente se considera agregar este modo de forma nativa, a partir de la demanda de los propios usuarios como una funcionalidad incluida dentro un paquete con otras tantas, a ser ofrecido en modalidad Freemium, explicada en la sección 3.2.1.

El Modo Exploración planteado en el presente trabajo considera la creación de arcos para la generación de mensajes. Estos arcos poseen nodos que representan Puntos de Referencia asociados o no a beacons, que indican el sentido diferenciando el nodo de partida y término e indican el lugar padre. El conjunto de todos los arcos dirigidos, explicados en la sección 5.2.3 de un lugar en particular, configuran la malla total de rutas posibles. Tomando las latitudes y longitudes de cada uno de los nodos de esta malla y transformándolos a coordenadas x e y, es posible tanto las distancias relativas entre nodos y los ángulos formados entre arcos utilizando la fórmula:

$$\cos\alpha = \frac{\bar{a} \cdot \bar{b}}{|\bar{a}| \cdot |\bar{b}|} \quad (5.1)$$

Luego, utilizando mensajes contextuales asociados a ángulos y distancias se pueden construir rutas de forma dinámica.

La construcción de la ruta particular se realiza utilizando cualquier algoritmo que permita la generación de rutas mínimas como Dijkstra, dado que se asume que todas los arcos construídos son accesibles. En ese caso se asignan los mismos pesos al grafo, no obstante, se pueden asignar distintos valores de accesibilidad por arco lo que podría generar rutas más seguras.

5.2.2. Primera iteración: enfoque rutas dirigidas

Supóngase que se desea habilitar un hall de cierto edificio, cuyo layout es el presentado en la figura 5.4. En primera instancia se define una ruta dirigida que cubra todos los elementos del espacio que se quieran comunicar. En este caso, se han utilizado 7 beacons que representan la entrada principal, la salida, baño de mujer, baño de hombre, sala 2, sala 1 y oficina 1 respectivamente. El objetivo principal del método planteado requiere guardar 3 elementos relevantes:

- La posición relativa de los elementos a comunicar, representados por los respectivos beacons, en relación a la (las) ruta(s) en el sentido fijado y en el sentido inverso. Esta información permitirá, como será explicado en el punto 5.3, entregar información contextual de dicho punto de interés. Finalmente, los mensajes contextuales generados son de tres tipos: descripción del elemento más información contextual a favor de la ruta, descripción del elemento más información contextual en dirección inversa de la ruta y, en el caso que no sea posible determinar el sentido recorrido, un mensaje genérico sólo con la descripción del elemento.
- Se requiere conocer el sentido en el cual se recorre la ruta definida, para lo cual se debe conocer el orden de los beacons en la ruta, en particular identificar el beacon anteriormente visitado para contrastarlo con el actual activo.
- Información acerca del punto de interés que se desea comunicar, de modo que se pueda construir un mensaje auditivo que lo describa de forma automática de acuerdo a su tipo y relevancia.

Dados todos los elementos antes mencionados permiten generar las siguientes posibles interacciones:

- Un individuo entra al hall. El sistema tiene registrado un beacon anteriormente leído que no pertenece al hall y la lectura actual registra al beacon de la entrada principal. Dado lo anterior, el sistema puede reconocer que el individuo se está acercando a la entrada principal desde afuera, generándose así un potencial mensaje del tipo “puerta corrediza hall principal, en frente tuyo”. El mensaje contiene una parte que describe al punto de interés, en este caso una puerta del hall que es de tipo corrediza y, por otro lado, información contextual que le indica a la persona que el elemento se encuentra en frente suyo. Dependiendo del comportamiento de los beacons, la potencia a la cual fueron configurados, el comportamiento de los dispositivos móviles en modo lectura considerando diversas situaciones, se podría extender el mensaje contextual otorgando mayor información, como por ejemplo “puerta corrediza hall principal a 2.5 metros, en frente tuyo”. El detalle acerca de la construcción misma del mensaje se explica en la sección 5.3.
- El individuo ya se encontraba recorriendo el hall. Para este ejemplo la persona viene caminando desde la ubicación de la oficina 1, por lo que ese beacon queda registrado como beacon anterior mientras que la lectura actual del dispositivo lee al beacon correspondiente al de la entrada principal. A diferencia del caso anterior, dado que el beacon anterior es un punto de la ruta, se conoce el sentido la posición relativa de la entrada con respecto a la ruta por lo que se genera un mensaje tipo “puerta corrediza hall principal, a tu derecha”.
- En los casos en que no existe registro acerca de un beacon anteriormente leído (cuando se prende la aplicación por primera vez) o el trazo entre el beacon anteriormente leído y el actualmente leído no pertenece a ninguna ruta predefinida, el mensaje se construirá sin incluir

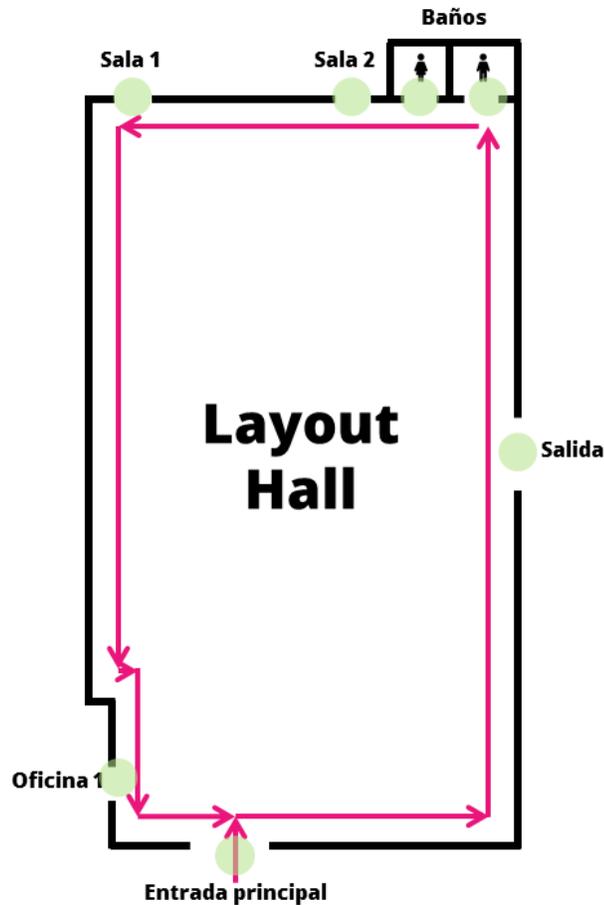


Figura 5.4: Layout Hall y rutas

información contextual del punto de interés. Por ejemplo, si la persona prende el dispositivo móvil dentro del hall y se realiza la primera lectura en la entrada principal, en este caso el mensaje tipo será “puerta corrediza hall principal”. Eventualmente, tal como se indicó en el primer punto, dicho mensaje puede ir acompañado con la distancia aproximada a dicho elemento. Dadas las condiciones no se puede añadir más precisión al mensaje.

El layout antes mencionado tiene una distribución simple que permite la generación de una ruta, sin embargo, en algunas situaciones en casos con distribuciones más complejas, es posible que sea necesaria la generación de 2 o más rutas para un mismo espacio. Para estos casos, el funcionamiento del método es el mismo, con la precaución de guardar los valores de posición relativa del beacon con respecto a la ruta, posición relativa del beacon con respecto a la ruta inversa y beacon previamente recorrido, cada uno en función de la ruta a la cual pertenecen.

5.2.3. Segunda Iteración: enfoque tramos dirigidos

Conceptualmente, la idea de las rutas predefinidas va en línea con el desarrollo de proyectos de habilitación indoor de mayor envergadura, dado que en estos se debe considerar la elaboración de planos de desplazamiento accesible para PeSD visual, con recorridos que otorguen seguridad para

las personas que los recorren y que, en muchos casos, además se complementan con otros tipos de ayudas como guías podotáctiles, barandas en escaleras, entre otros elementos. En el caso de recorridos guiados, como el proyecto que se desarrolla para el Museo de la Memoria, en donde la persona debe desplazarse por una ruta cerrada y sin ramificaciones - puede ser de ayuda imaginar una pista de carreras por ejemplo-, este enfoque es fácilmente implementable ya que solo consiste en una ruta, claramente definida. No obstante lo anterior, en la mayoría de los casos de habilitación indoor, lo que se desea es el libre desplazamiento de las PeSD visual en áreas acotadas del espacio, por lo que se debiesen generar una serie de posibles rutas que se cruzan entre ellas lo que hace dificultoso su identificación.

En los casos de mayor complejidad antes descritos, como puede ser la habilitación de un centro comercial, un supermercado y en general espacios con ramificaciones de rutas que hacen compleja la implementación utilizando el enfoque anterior, se considera una modificación del mismo que llamaremos enfoque de “tramos dirigidos”. Básicamente consiste en seleccionar pares de coordenadas dentro del espacio entre las cuales existe una ruta accesible para la PeSD visual. Las instrucciones para ir de una coordenada a otra no son necesariamente inversas cuando estas son recorridas en sentido inverso, razón por la cual se han definido como tramos “dirigidos”. Un espacio a habilitar va a tener entonces un set de tramos dirigidos que configurarán una red de coordenadas conectadas entre sí y permitirán a la PeSD visual desplazarse libremente por estas áreas habilitadas.

Adicionalmente se pueden generar rutas óptimas -para ser utilizadas en el Modo Ruteo- sobre estas redes que permitirían ir de un punto A a un punto B cualesquiera. Dado que cada tramo está preparado para el desplazamiento de PeSD visual, la generación de rutas optimizadas por algún parámetro, como de distancia por ejemplo, es válido. Eventualmente se podrían agregar sobre este mismo modelo, parámetros que indiquen el grado de dificultad que tiene el desplazamiento para PeSD visual por ciertos tramos previamente seleccionados, como por ejemplo, privilegiar el uso de ascensores por sobre el uso de escaleras. Así los tramos que conectan a escaleras tendrán un ponderador más bajo y, por consiguiente, las rutas generadas utilizando algún método optimizador, tenderán a no considerarlas.

5.2.4. Tercera Iteración: tramos dirigidos y Puntos de Referencia

Realizando pruebas preliminares en el OpenLab, se observa una situación no prevista. En este caso, el laboratorio ha sido habilitado a través de beacons pero no el edificio que lo contiene. En este contexto, el primer punto de interés al cual se enfrenta el usuario nunca tendrá un mensaje contextual, dado que no hay beacons anteriores en el camino que permitan identificar el sentido en el cual la persona se esta enfrentando al punto de interés en cuestión, en este caso la puerta de entrada. Es decir, para el sistema es lo mismo que la persona esté entrando o saliendo del laboratorio, en lo casos en que el usuario haya prendido el dispositivo dentro de éste y no haya registrado el paso en otro punto de interés. Para subsanar este problema se ha agregado un nuevo elemento al sistema: los “Puntos de Referencia”. Un “Punto de Referencia” es un elemento dentro del entorno que posee latitud y longitud, lo cual es válido tanto para espacios abiertos como cerrados. Un “Punto de Referencia” representa entonces cualquier punto en el espacio y puede, o no, tener un beacon asociado. Luego todas las interacciones entre beacons descritas en puntos anteriores se realizaran entre “Puntos de Referencia”, de manera que puedan existir tramos dirigidos de tres tipos: Punto

de Referencia - beacon, Punto de Referencia - Punto de Referencia, beacon - beacon.

De esta forma, retomando el ejemplo anterior, para generar un mensaje contextual para la puerta de entrada del OpenLab bastaría con incluir un “Punto de Referencia” de forma estratégica la entrada del edificio, en el exterior y vincularla en un mismo tramo con la entrada. De esta forma dado que la persona necesariamente debe pasar por este punto, al momento de acercarse a la puerta de entrada el sistema ya posee información del “Punto de Referencia” anteriormente activado y por tanto puede generar el mensaje contextual respectivo.

5.3. Diagramas de flujo de información

Utilizando Nearby Message API de Google, esta genera 4 posibles callbacks para el manejo de attachments en modalidad BLE⁴, 2 de los cuales (onDistanceChanged y onBleSignalChanged) se encuentran disponibles a partir de las últimas versiones como Advanced Features.

- onFound: es ejecutado al ser encontrado un attachment en la zona
- onLost: es ejecutado cuando un attachment se pierde de la zona
- onDistanceChanged: se ejecuta cuando un attachment sufre un cambio de posición, recuperando su distancia
- onBleSignalChanged: se ejecuta cuando un attachment sufre un cambio de señal, la cual es recuperada

Estas últimas funciones avanzadas están actualmente en desarrollo y aún no entregan resultados confiables, por lo que se decide en no darles uso hasta una próxima actualización. De esta forma el siguiente esquema mostrado en la figura 5.5 el flujo comienza tanto al recibir o al perder un attachment, que para este caso representa el id del beacon dentro o fuera de la zona de detección.

Al ser encontrado un attachment de tipo id se procede a verificar si este se encuentra previamente registrado en el arreglo de beacons en zona. De ser la primera vez que este beacon es leído durante la sesión, este es agregado al arreglo para pasar inmediatamente al proceso de construcción del mensaje a ser reproducido por el servicio TTS de Lazarillo, por otro lado, si el beacon ya fue leído previamente en la sesión este se va a encontrar registrado en el arreglo y, por tanto, se verifica si cumple con la condición de los 20 segundos desde que se registró su pérdida de señal. Si han pasado más de 20 segundos se procede a la construcción del mensaje, en caso contrario la proceso termina previamente limpiando el registro de pérdida de señal (que en términos prácticos reasigna el valor de la fecha a un valor excesivamente alto). Luego de haberse elaborado el mensaje auditivo, se guarda el valor del actual beacon en zona en la variable last-id, cuya utilidad será explicada más adelante durante el detalle del proceso de construcción del mensaje. El paso final corresponde a encolar el mensaje en Lazarillo y la posterior limpieza del registro de pérdida de señal.

En el caso de recibirse la pérdida de algún attachment, dicho id es registrado junto al momento preciso en el arreglo de beacons en zona.

⁴La API es capaz de funcionar no sólo entre dispositivo móvil-beacon sino da la posibilidad de conexión entre 2 dispositivos móviles entre sí

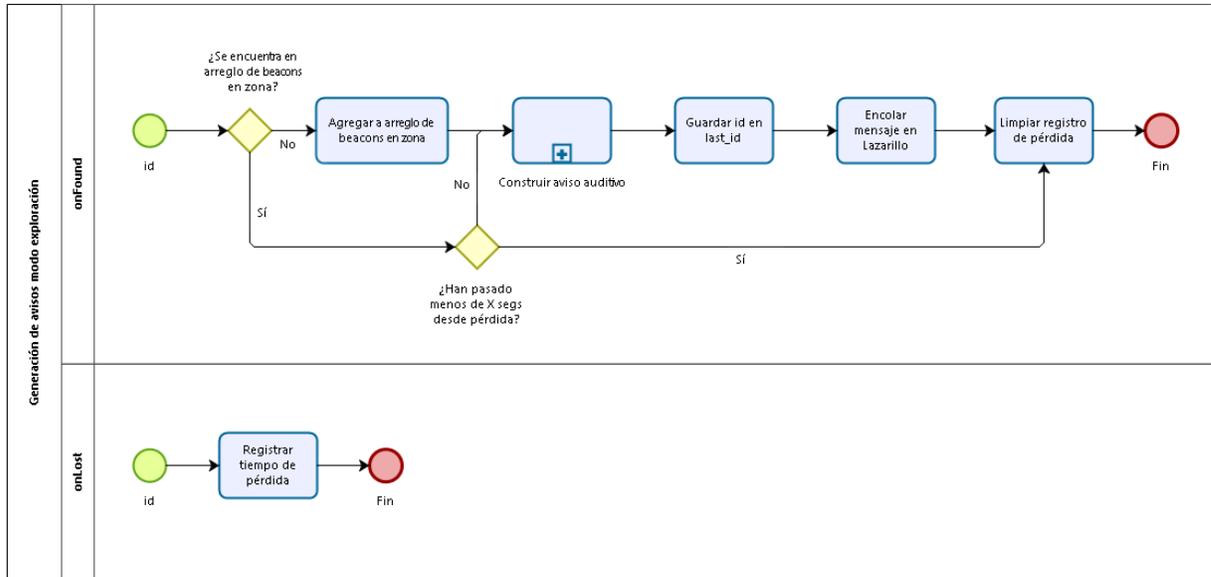


Figura 5.5: Generación de avisos modo exploración

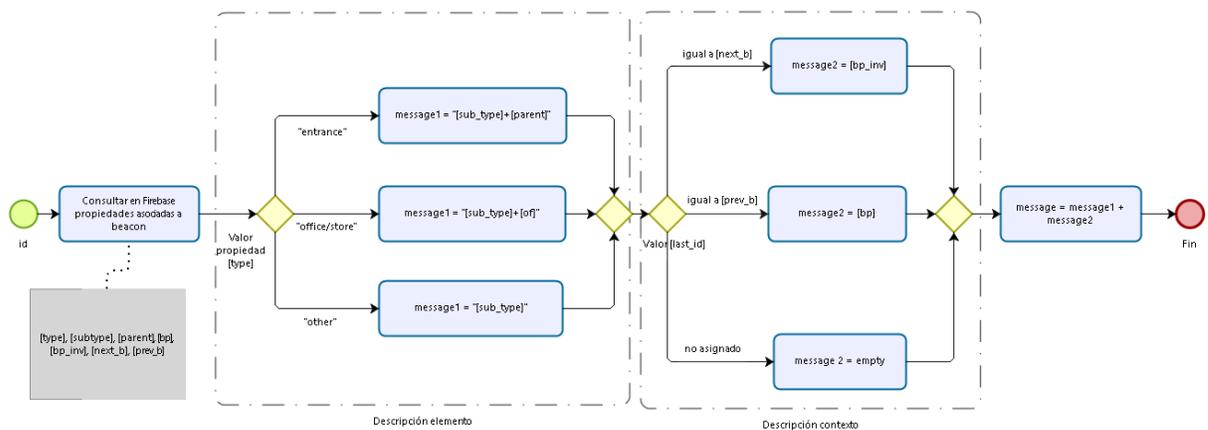


Figura 5.6: Construcción aviso auditivo

A continuación se muestra el código final generado en base al flujo anterior y a modificaciones producto de las pruebas realizadas durante el preexperimento, documentadas en la sección 8.2. Se agrega la distancia al mensaje contextual y la posición relativa del mensaje contextual va a depender de los arcos generados y sus relaciones.

```

private void buildMessage(String beaconId){
    String beaconMessage;
    String type = lzPoints.get(beaconId).getType();
    String subtype = lzPoints.get(beaconId).getSubtype();
    String name = lzPoints.get(beaconId).getName();
    boolean announceableMessage = true;
  
```

```

switch (type) {
    case "route_alerts":
        beaconMessage = "Atencion, " + subtype + " " + name;
        break;
    case "confirmation":
        beaconMessage = sharedPref.getString("general_category/
confirmation/" + subtype, null);
        break;
    default:
        beaconMessage = subtype + " " + name;
}
/*
Contextual message: distance
*/
double metersDistance = lzPoints.get(beaconId).getBeaconMetersDistance();
if(metersDistance < 1 && !type.equals("confirmation")){
    beaconMessage += " a menos de un metro";
} else {
    beaconMessage += " a aproximadamente " + Math.round(metersDistance)
        + " metros";
}
/*
Contextual message: position
*/
if(!type.equals("confirmation")){
    if(segmentExists(lzArcs, prevBeaconId, beaconId)
    && !segmentExists(lzArcs, prevPrevBeaconId, beaconId)){
        beaconMessage += contextMessage(lzArcs, prevBeaconId, beaconId);
    } else if (!segmentExists(lzArcs, prevBeaconId, beaconId)
    && segmentExists(lzArcs, prevPrevBeaconId, beaconId)){
        beaconMessage += contextMessage(lzArcs, prevPrevBeaconId, beaconId);
    } else if (!segmentExists(lzArcs, prevBeaconId, beaconId)
    && !segmentExists(lzArcs, prevPrevBeaconId, beaconId)){
        String predictedPrevNode =
        findPrevNode(lzArcs, prevBeaconId, beaconId);
        if(!predictedPrevNode.equals("") ||
        prevBeaconId.equals("")){
            beaconMessage += contextMessage(lzArcs,
            predictedPrevNode, beaconId);
        } else {
            announceableMessage = false;
        }
    }
}
}
if(announceableMessage){
    mp = MediaPlayer.create(this, R.raw.message);
    mp.start();
    speakMe(beaconMessage);
    Log.i(TAG, beaconMessage + ": " + beaconId);
}

```

```
}  
prevPrevBeaconId = prevBeaconId;  
prevBeaconId = beaconId;  
}
```

La estructura de los mensajes auditivos a generarse se basan en un estándar abierto desarrollado por Wayfinr⁵. Wayfindr es una organización financiada y difundida por la Google Foundation, cuyo objetivo es generar las bases de la navegación indoor para PeSD visual. El presente trabajo se ha desarrollado tomando en consideración dichos alineamientos planteados en el Working Draft 1.0 liberado el 12 de Mayo del 2016, siendo la última versión liberada hasta la fecha⁶. La detalle de la construcción del aviso auditivo se puede apreciar en 5.6. Básicamente, cada uno de los beacons dispuestos en el espacio, representan elementos de exploración que, en una primera instancia, se han dividido en 3 diferentes tipos: 'entrance', que son elementos que representan entradas (cuyo subtipo puede ser una puerta con manilla, puerta eléctrica, puerta giratoria, entre otras); 'subplace', que representan subespacios dentro de un entorno, que dependiendo del caso pueden corresponder a oficinas, locales, salas, estaciones de trabajo, entre otras; 'other' en el caso de todos los demás tipos que no corresponden a ninguno de los anteriormente mencionados.

La anterior separación tiene como objetivo generar distintos tipos de mensaje por cada grupo de elementos. Así, por ejemplo, en el caso de enunciarse una entrada, esta tendrá una estructura [sub_type] + [parent] que en una situación hipotética puede ser representada como 'puerta eléctrica OpenLab'. En el caso de un subespacio la estructura es del tipo [sub_type] + [name] en otra situación hipotética puede ser leída como 'Oficina Director OpenLab'.

La segunda parte del flujo corresponde a la construcción de la información contextual, la que entrega 3 tipos de mensajes: el primero, en el caso que se detecte que la ruta se está recorriendo en sentido inverso, se recupera la posición del elemento a ser comunicado representado por el beacon, guardado previamente como bp_inv o posición inversa del beacon; si la ruta esta siendo recorrida en el sentido correcto, que ocurre cuando el beacon anteriormente recorrido corresponde con el beacon anterior al actual, se recupera el valor bp o posición del beacon en la dirección natural; en el caso en que no se logre determinar el sentido en que se está recorriendo la ruta, entonces el mensaje auditivo no contará con información contextual adicional.

Finalmente, para el caso en que el usuario requiera conocer el lugar en que se encuentra, haciendo uso del botón '¿dónde estoy?', presente actualmente en Lazarillo para espacios abiertos, pero aplicado ahora para espacios cerrados, lo que se requiere es conocer el valor del parent o lugar padre al cual pertenece el beacon en zona. Si existen varios beacons en zona, bastará con recuperar el parent del primero, ya que todos compartirán el mismo valor. El procedimiento se puede apreciar en la figura 5.7.

⁵<https://www.wayfindr.net/>

⁶12 de junio de 2017

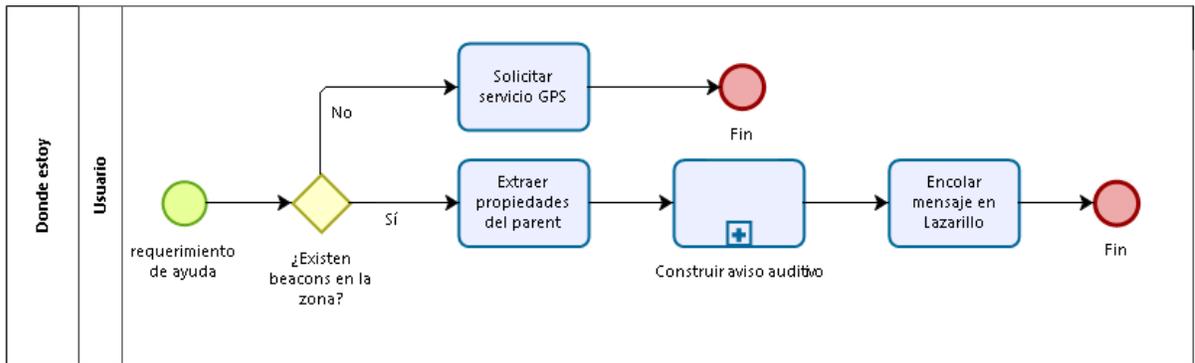


Figura 5.7: Procedimiento donde estoy

Capítulo 6

Procedimiento para levantamiento de información

La arquitectura de datos que hace posible el funcionamiento propuesto en la sección 5, se configura en torno a árboles JSON que permiten estructurar fácilmente la lógica de nodos padres e hijos, representando así lugares, rutas asociadas a lugares y beacons asociados a rutas y, en algunos casos, directamente a lugares.

6.1. Base de datos

Las bases de datos relacionales (BDR) tienen un uso extendido y transversal en el mundo del desarrollo de software, entre las razones que justifican este hecho destacan: el largo tiempo que llevan en el mercado, bajo costo de servicios y mayor soporte, aunque principalmente se debe a que los desarrolladores están adaptados a su uso. No obstante lo anterior, en el caso particular del presente desarrollo, la estructura de árboles es la que mejor se adapta a las necesidades, además de su gran flexibilidad y posibilidad de escalamiento.

Para llevar a cabo la implementación se utiliza una base de datos en tiempo real llamada Firebase, desarrollada por Google la que cuenta con API's de conexión multiplataforma que facilitan el desarrollo tanto para dispositivos móviles como para plataformas web, además es tiene un manejo interno de conexiones que permite trabajar indistintamente tanto online como offline, en este último caso deja las operaciones en cola hasta que la conexión se restablece.

6.2. Arquitectura de datos

6.2.1. Estructura base

A continuación se muestra la estructura de la base de datos propuesta, la que es compatible con la estructura de LazarilloApp y extensible para la plataforma social. Se definen como elemento principal las instituciones (IzInstitutions), que representan una organización que desee realizar una habilitación indoor en alguno de sus espacios. Inicialmente se definen para este propósito eventos (IzEvents) y sucursales (IzOffices), donde los primeros se refieren a actividades realizadas durante un periodo acotado de tiempo y los segundos a la o las representaciones físicas de dichas instituciones. Tanto los eventos como las sucursales son nodos hijos de las instituciones en esta estructura. En un tercer nivel jerárquico se definen los puntos de referencia (IzPoints), que representan coordenadas relevantes tanto para los eventos como para las sucursales y que, como se explicó en la sección 5.2.4, pueden estar o no asociados a beacons. Al igual que en base de datos relacionales, los elementos individuales pueden ser reconocidos a través de sus respectivos identificadores o llaves auto generadas por Firebase, de modo que sean únicos. En la figura 6.1 se muestra un esquema tipo para instituciones. A simple vista se pueden apreciar las diferentes llaves autogeneradas que representan las instituciones guardadas en la base de datos. Se pueden observar también las distintas propiedades de la institución, en este caso particular de la institución “Museo de la Memoria” que se despliegan de la forma clave-valor.

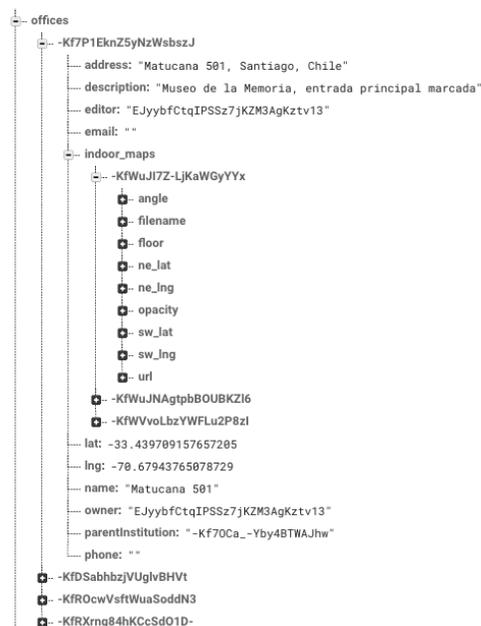
```
institutions
├── -Kf7h9S2CAma3EQWq1lx
├── -Kf70Ca_-Yby4BTWAJhw
│   ├── category: "museum"
│   ├── description: "Proyecto de habilitación del Museo de la Memoria"
│   ├── editor: "EJyybfCtqIPSSz7jKZM3AgKztv13"
│   ├── email: "contacto@lazarillo.cl"
│   ├── facebook: ""
│   ├── isVerified: true
│   ├── name: "Museo de la Memoria"
│   ├── owner: "EJyybfCtqIPSSz7jKZM3AgKztv13"
│   ├── phone: ""
│   ├── twitter: ""
│   └── web: "www.museodelamemoria.cl/"
├── -KfDphMf2nl8MIH9dEVj
├── -KfDRdbAvOqKg_LWeQHB
└── -Kfls50X93a_qvsDa-Jk
```

Figura 6.1: JSON ejemplo Instituciones

En la figura 6.2 se muestran estructuras de ejemplo para eventos y para sucursales. En ambos casos se pueden observar las diferentes llaves autogeneradas que representan una instancia y, por otro lado, las propiedades que caracterizan a dichos eventos y sucursales.



(a) JSON ejemplo eventos



(b) JSON ejemplo sucursales

Figura 6.2: Esquemas de ejemplo

Los 4 elementos antes descritos más sus respectivas propiedades configuran la estructura básica en la cual el presente trabajo se construye y permiten desarrollar los siguientes elementos más complejos. A continuación se explican cada uno de ellos:

6.2.2. Elementos secundarios

Puntos de Referencia

LazarilloApp basa su funcionamiento en el posicionamiento de los elementos en el espacio, razón por la cual cada uno de los elementos creados o recopilados desde sus plataformas o de terceros deben contar con sus respectivas coordenadas espaciales. En particular, cada uno de los 4 nodos principales nombrados anteriormente poseen dentro de sus propiedades su posición GPS que los identifica y cuya precisión aumenta en la medida que baja el nivel jerárquico: la posición de una institución representa a groso modo la región geográfica que abarca, generalmente un país; las coordenadas de una sucursal o de un evento representan una coordenada GPS central que permita identificar fácilmente que lugares se encuentran habilitados alrededor; las coordenadas de los puntos de referencia son más precisas, ya que representan elementos a un nivel más específico, como un una escalera, un ascensor, un baño en espacios cerrados o una barra de bebidas, un stand, un carrito de comida en un evento en un espacio abierto.

Si bien la base de datos jerárquica utilizada hace más fácil y óptimo el manejo de datos anidados que se utilizan en este desarrollo, esta tiene algunas limitaciones que pueden ser subsanadas utilizando algunas herramientas adicionales. Por ejemplo, realizar una consulta sobre todos elementos cercanos a una coordenada GPS, no es posible directamente utilizando la API de Firebase,

ya que esta permite realizar consultas del tipo “mayor que”, “menor que” e “igual que” respecto a una propiedad numérica y solo “igual que” para propiedades alfanuméricas. Lo anterior aplicable a una propiedad por vez por lo que la consulta antes descrita no está permitida ya que implicaría dos consultas simultáneas sobre latitud y longitud. Por esta razón se utiliza una librería de código abierto llamada "GeoFire for Android/Java" permite guardar y consultar identificadores basado en su ubicación geográfica. De esta, cada vez que se guardan las propiedades de latitud y longitud para instituciones, eventos y sucursales, en simultáneo GeoFire genera una estructura propia que permite guardar apropiadamente el identificador o llave del elemento a ser consultado (institución, evento o sucursal).

De esta forma la aplicación puede realizar consultas geolocalizadas en torno a la posición actual de la persona para precargar de forma local los GeoPuntos cercanos asociados a distintos Puntos de Referencia. Dado que es muy costoso en términos de tiempo y procesamiento realizar la consulta sobre todos los GeoPuntos de la base de datos -ya que pueden existir lugares como el Museo de la Memoria, el cual posee más de 60 GeoPuntos-, se realizan dos consultas por separado buscando eventos y sucursales cercanas utilizando GeoFire, una vez encontrados dichos eventos y sucursales se recuperan sus GeoPuntos asociados y se guarda en caché para que se encuentren disponibles al momento de recibir un attachment de un beacon que corresponda a alguno de estos GeoPuntos.

En la figura 6.3a se observa un GeoPunto asociado a la llave “-Kf8Gq3--1fGwtYX61FA” asociada al punto de referencia de la figura 6.3b, con las propiedades que ahí se despliegan.



Figura 6.3: Esquemas de ejemplo

Arcos

Una vez dispuestos todos los Puntos de Referencia de un evento o sucursal, el siguiente paso corresponde a la generación de tramos dirigidos, concepto explicado en la sección 5.2.3. Lo que se requiere es agregar una interacción entre dos Puntos de Referencia, que permita la construcción de mensajes contextuales que indiquen la posición relativa del elemento con respecto al tramo recorrido por la PeSD visual y, por otra parte, de la posibilidad de construir rutas óptimas en espacios cerrados. En la figura 6.4 se observa el arco “-Kd1K1L4GuOnnadK-EXH” con las propiedades: bp, que representa la posición del elemento a enunciar, en ángulo, con respecto al segmento dirigido;

from y to, que relacionan el segmento con los puntos de referencia o IzPoints inicial y terminal; inverseOf, que representa la llave del segmento dirigido opuesto.



Figura 6.4: JSON ejemplo Arcos

Contexto

La estructura de contexto en términos simples es un diccionario que permite a la aplicación traducir elementos y características del entorno en el idioma de preferencia, siempre y cuando éste se encuentre disponible en el diccionario de contexto, en caso contrario la traducción se realizará al idioma definido por defecto, que en este caso es el inglés.

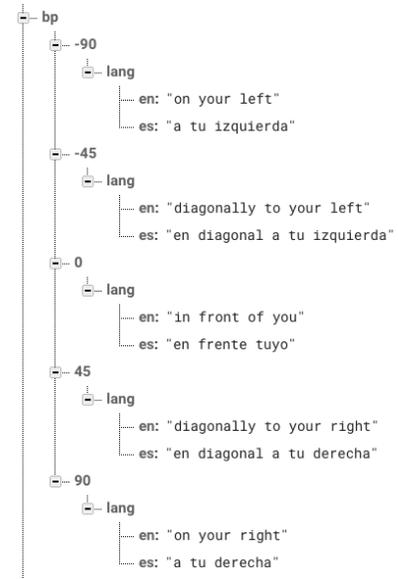
El diccionario y la construcción misma de los mensajes, explicada en profundidad en la sección 5.3, se basa en el estándar abierto de audio localización desarrollado por Wayfindr y apoyado por Google, en su versión “Candidate Recommendation 1.0”, publicada el 15 de Diciembre del

2016. Este estándar provee recomendaciones para mejores prácticas para el desarrollo de aplicaciones destinadas a PeSD visual, directrices, consideraciones y sugerencias para investigaciones futuras en el área. En particular, para el desarrollo de este trabajo, se han tomado en consideración fundamentalmente los siguientes puntos:

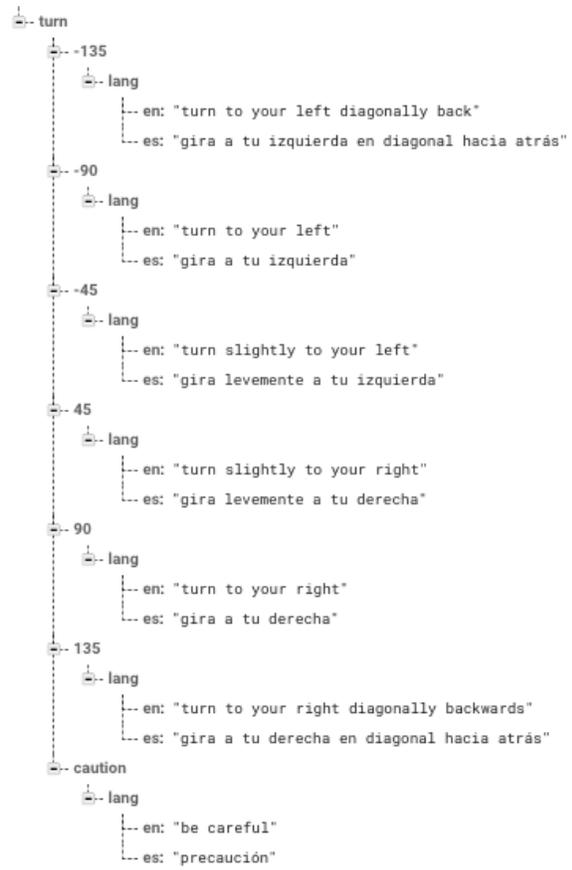
- Para la construcción de rutas: la ruta más segura, no la más rápida o más corta
- Utilización de beacons BLE como tecnología de posicionamiento
- Elementos relevantes del entorno
- Estructura de mensajes auditivos

La estructura posee 3 nodos principales que se explican a continuación:

- bp (Beacon position): este diccionario le permite a la aplicación asignarle un mensaje contextual de dirección al elemento enunciado de acuerdo al ángulo especificado en la sección 6.2.2. La estructura se observa en la figura 6.5a.
- turn: actualmente este diccionario, de la figura 6.5b, no está en uso ya que fue construido para la generación de rutas. Corresponde a mensajes imperativos que le indican a la PeSD visual modificar su desplazamiento actual.
- general_category: es el diccionario de elementos cercanos. Los elementos han sido agrupados en categorías generales tales como entradas, elementos del entorno, puntos de interés y alertas en rutas. Estas categorías generales no tienen ningún efecto sobre los mensajes auditivos generados por la aplicación, sino más bien es un ordenamiento de las categorías al momento de desplegarlas en el Portal Lazarillo debido a su cantidad. Las categorías generales y categorías creadas hasta la fecha se aprecian en la figura 6.6



(a) JSON ejemplo Beacon position



(b) JSON ejemplo Instrucciones en ruta

Figura 6.5: Esquemas de ejemplo

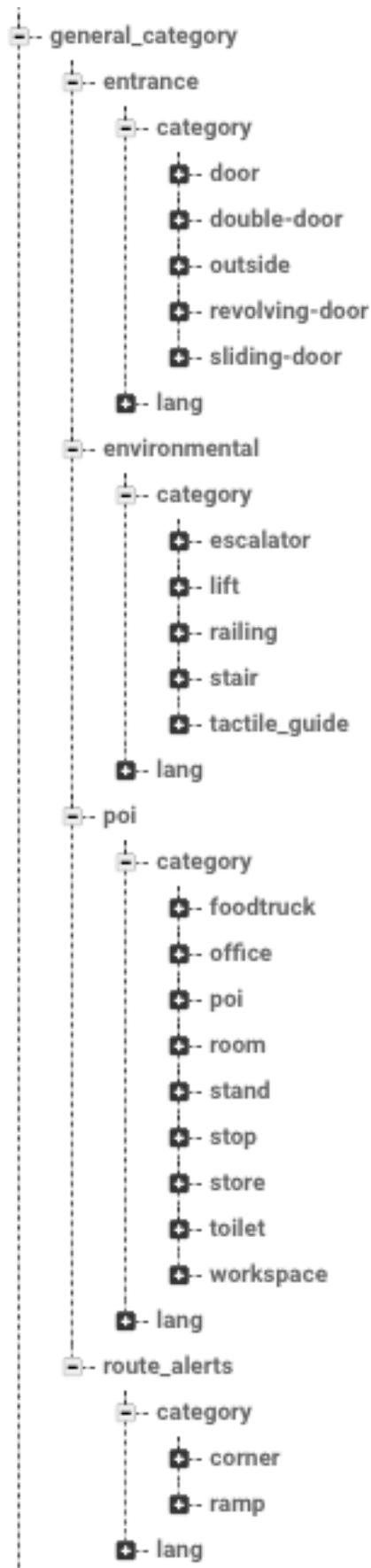


Figura 6.6: JSON ejemplo Categorías

<ruta>	/lzBeacons
id	Identificador otorgado por el servicio que proveyó el lugar.
type	{entrance, subplace, pathways }
subtype	{ entrance: {door, electric-door, revolving-door, sliding-door, pushing-door}, subplace: {office, shop}, pathways: {corridor, ramp, tunnel, subway, pavement, crossing, intersection, escalator, stair, lift}, other: {open} }
route/id	identificador de la ruta
route/id/prev	Identificador del anterior lzBeacon en ruta
route/id/next	Identificador del siguiente lzBeacon en ruta
route/id/bp	Valor {90(right), -90(left), 0(front)} posición de beacon con respecto a la ruta
route/id/bp_inv	Valor {90(right), -90(left), 0(front)} posición de beacon con respecto a la ruta en sentido inverso
parent/lzPlace	Identificador del lzPlace al cual pertenece
name/lang	Etiqueta complementaria a type y subtype
properties/prop(i)/name/lang	Nombre de la propiedad a agregar al elemento
properties/prop(i)/value/lang	Valor de la propiedad a agregar al elemento
lat	Latitud del beacon
lon	Longitud del beacon

Tabla 6.1: lzBeacons

<ruta>	/routes
id	Identificador otorgado por el servicio que proveyó el lugar
lzBeacons/id	Identificadores de todos los beacons de la ruta
waypoints/point(i)/lat	Latitud del i-ésimo waypoint
waypoints/point(i)/lon	Longitud del i-ésimo waypoint
waypoints/point(i)/floor	Piso del i-ésimo waypoint
parent/lzPlace	Identificador del lzPlace al que pertenece la ruta

Tabla 6.2: routes

<ruta>	/geoLzPlaces
key	Identificador de lugar creado a través de plataforma Lazari- llo

Tabla 6.3: geoLzPlace

<ruta>	/IzPlaces
id	Identificador del lugar correspondiente con la key del geoLzPlace
creator	Referencia a la institución o user. Ej: '/users/<uid>' ó 'institutions/<iid>'
name/lang	Nombre de IzPlace en el idioma correspondiente
address	Dirección del IzPlace
lat	Latitud del IzPlace
lon	Longitud del IzPlace
geoLzPlaces/id	Identificador de referencia al geoLzPlace
routes/id	Identificadores de las rutas asociadas al IzPlace
description/lang	Descripción del IzPlace en el idioma correspondiente
phone	Teléfono de contacto del IzPlace
web	Página web de contacto del IzPlace
email	Correo de contacto del IzPlace
lzBeacons/id	Identificadores de los beacons asociados al IzPlace sin ruta (opcional)

Tabla 6.4: IzPlaces

<ruta>	/users
id	Identificador del usuario

Tabla 6.5: Usuarios

<ruta>	/places
id	Identificador del lugar que debe coincidir con geoPlace
source	Identificador de la fuente de
name	Nombre asociado al place
category	Categoría según Lazarillo. Puede ser Transport, Bank, Health, Food, Store, Art, Building, Education, Pub, Other.
specific_category	Clasificación más fina del lugar que por lo general, es la otorgada por el source. Si hay más de una desde el source, se usa la principal y que más lo identifica.
lat	Latitud
lon	Longitud
address	Dirección entregada por el source
properties	Propiedades adicionales

Tabla 6.6: Lugares

6.3. Flujo levantamiento de información

Para que el sistema de navegación en espacios cerrados pueda generar los mensajes auditivos correspondientes al contexto, es necesario que los tramos dirigidos estén contruidos. En la figura 6.7 se observa que una condición inicial es la existencia de al menos 2 Puntos de Referencia en alguna sucursal o evento, de existir se procede a crear y guardar la posición relativa del elemento a enunciar.

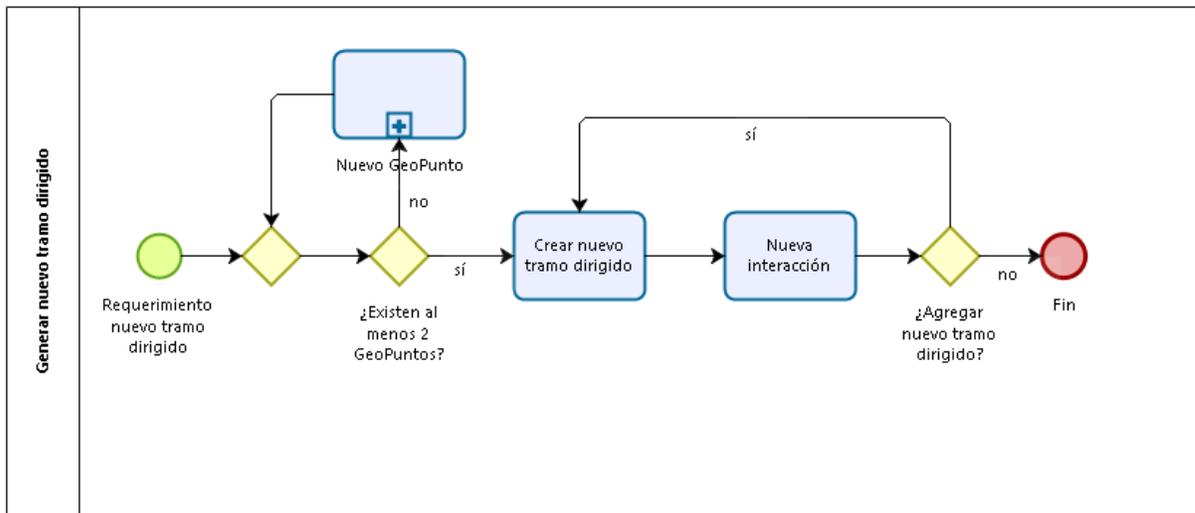


Figura 6.7: Flujo creación de tramos dirigidos

De no existir al menos dos Puntos de Referencia, se procede a la creación de ellos, tal como se ve en la figura 6.8, que representa el macroproceso de nuevo GeoPunto. En este flujo se observa que la existencia de una sucursal es una condición necesaria. En estricto rigor, esta condición es aplicable también para los eventos por lo que bastará que exista alguno de estos dos para que se puedan asignar Puntos de Referencia a ellos. Otro punto importante a destacar es que los Puntos de Referencia pueden o no estar vinculados a un beacon. De no estar vinculados pueden ser capturados a través del GPS del dispositivo móvil o simplemente pasar a ser elementos referenciales de la ruta.

Finalmente, de no existir ninguna sucursal (o evento, ya que es análogo), el procedimiento para la creación de una nueva sucursal es el que se muestra en la figura 6.9.

6.4. Portal Lazarillo

Toda la información contenida en el arquitectura de datos mostrada anteriormente debe ser proveída para el correcto funcionamiento de la aplicación. A continuación se propone el Portal Lazarillo, cuya finalidad es permitir que se pueda realizar la carga de datos de una forma fácil, tanto para el mismo equipo de Lazarillo como para, eventualmente, un tercero. Todas las vistas, interacciones y lógica ha sido desarrollada en el contexto del presente trabajo, la construcción y puesta en marcha ha sido desarrollada por un miembro del equipo Lazarillo.

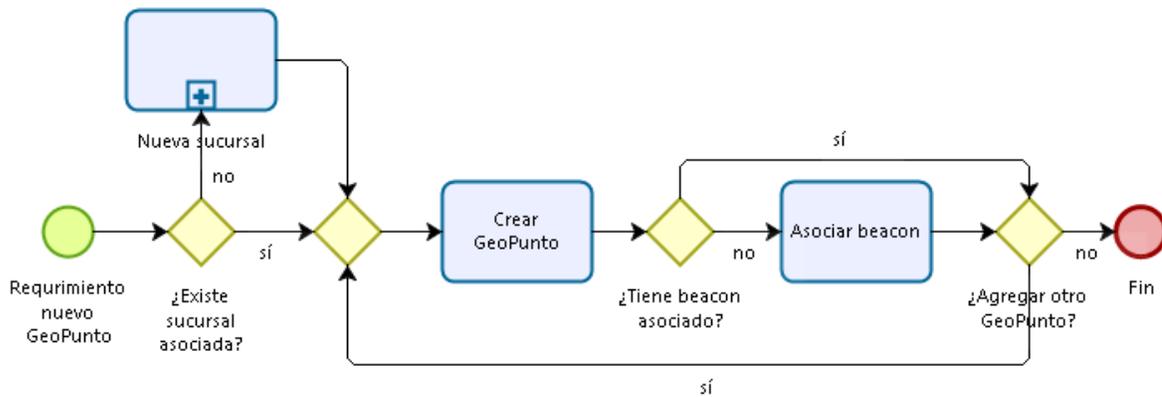


Figura 6.8: Flujo creación de Puntos de Referencia

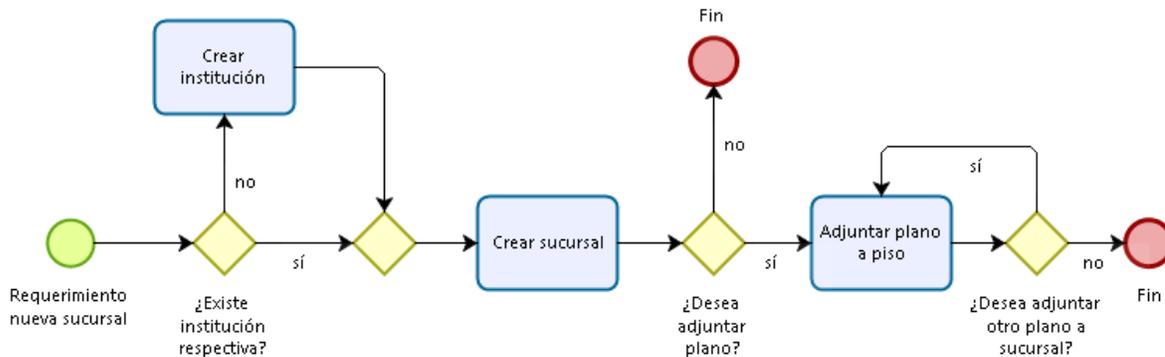


Figura 6.9: Flujo creación de sucursales

En la figura 6.10 se puede apreciar el formulario básico para la creación de una institución en el portal. Algunos de los campos son obligatorios y son indicados con asterisco. El campo “Categoría” es una etiqueta que se asigna para poder agrupar a las instituciones dentro de la plataforma.

Luego de creada la institución se pueden crear tanto eventos o sucursales asociadas a ella. Para generar un nuevo evento se deben llenar los campos mostrados en la figura 6.11. Se debe incluir una dirección, la que posicionará un pin sobre el plano. El usuario puede ajustar el pin sobre el punto que encuentre más representativo del lugar donde se realizará el evento, posiblemente la entrada principal de este. Si bien existe la posibilidad de agregar Puntos de Interés asociados a este evento, el pin generado sobre el evento permite a la aplicación realizar pre carga de eventos cercanos a la posición de la PeSD visual y luego, si corresponde, la carga de los Puntos de Interés asociados al evento, evitando así realizar consultas geolocalizadas directamente sobre los Puntos de Interés, lo que sería costoso.

De forma similar se generan las sucursales, tal como se muestra en la figura 6.12. A diferencia de los eventos, las sucursales no poseen fecha de inicio y término. Para ambos elementos existe la posibilidad de adjuntar un plano, el cual se asocia a un piso -si corresponde- y se modifica en ancho, alto y ángulo hasta conseguir el alineamiento indicado sobre el mapa. Este proceso es relevante ya

Edita tu institución Subtítulo

Nombre institución*	<input type="text" value="Museo de la Memoria"/>
Breve descripción*	<input type="text" value="Proyecto de habilitación del Museo de la Memoria"/>
Categoría*	<input type="text" value="Museo"/>
Email de contacto*	<input type="text" value="contacto@lazarillo.cl"/>
Sitio Web*	<input type="text" value="www.museodelamemoria.cl/"/>
Teléfono de contacto	<input type="text"/>
Facebook	<input type="text"/>
Twitter	<input type="text"/>

Figura 6.10: Vista Portal Lazarillo: Nueva institución

que permite la fijación de coordenadas GPS de los Puntos de Referencia de forma más intuitiva. En la figura 6.13 se muestra un plano nivel explanada correspondiente al Museo de la Memoria, el que ha sido ajustado para el calce con el mapa inferior. A parte de los controles mencionados anteriormente, se adiciona un control de transparencia que es de gran ayuda al momento de verificar si el plano ha sido posicionado exitosamente.

Los Puntos de Referencia se crean mediante el formulario de la figura 6.14 donde destacan los campos “nombre”, “categoría”, “posición” y “piso”. Los dos primeros sirven para identificar el punto durante la navegación, la posición para la construcción dinámica de rutas y el piso para precargar el mapa correcto para posicionar el pin.

Probablemente la parte más importante del proceso es la creación de un nuevo tramo dirigido, cuya vista se muestra en la figura 6.15. La correcta construcción de estos tramos permite generar una acertada interacción entre beacons durante la navegación. Básicamente se le indica a la aplicación que mensaje contextual construir al recorrer un arco cuyos nodos que representan Puntos

Edita tu evento

Nombre del evento* Lollapalooza 2017

Descripción* Lollapalooza Santiago 2017, Parque O'Higgins

Dirección* Parque O'Higgins, Santiago, Chile

Fecha Inicio* 01/04/2017 12:00

Fecha Término* 02/04/2017 23:59

Email de contacto* contacto@lotuspro.cl

Teléfono de contacto* +56 2 2789 0473

Publicar como*: Lotus Producciones

Precio Variable

Publicar Click para confirmar la publicación del evento

Guardar

Mejorar la referencia con un plano del edificio
Formatos válidos: .jpg, .png

Subir archivo

Figura 6.11: Vista Portal Lazarillo: Nuevo evento

Sucursales Agrega tu sucursal

Nombre sucursal*

Dirección* Busca un lugar o dirección válida

Breve descripción* Cuéntanos acerca brevemente acerca de tu empresa

Email de contacto nombre@miempresa.cl

Teléfono de contacto

Guardar

Mejorar la referencia con un plano del edificio
Formatos válidos: .jpg, .png

Subir archivo

Ver mi Institución

Figura 6.12: Vista Portal Lazarillo: Nueva sucursal

de Referencia dentro del espacio y que pueden estar asociados o no a un beacon, generándose así interacciones beacon - beacon, beacon - geopunto, geopunto - beacon y geopunto - geopunto. Se debe tener en consideración que los Puntos de Referencia al ser detectados solo a través del uso del GPS, son sólo referenciales en espacios cerrados ya que no serán activados durante la navegación pero servirán para la correcta construcción de rutas dinámicas.

Finalmente, un espacio ya habilitado contará con una red de tramos dirigidos con las distintas interacciones que se pretende generar en el espacio habilitado. En el caso de la habilitación realizada en el Museo de la Memoria, la red de tramos dirigidos es la que se aprecia en la figura 6.16.

Sucursales [edita tu sucursal](#) [Ver mi institución](#)

Nombre sucursal*

Dirección*

Breve descripción*

Email de contacto

Teléfono de contacto

[Guardar](#)

Mejorar la referencia con un plano del edificio [Subir archivo](#) [Eliminar](#)

Formatos válidos: **.jpg, .png**

Angulo de rotación (grados)

Transparencia

Piso (nivel)

[Agregar otro nivel](#)

Figura 6.13: Vista Portal Lazarillo: Adjuntar plano en sucursal

Crea tu Punto de Referencia Permite georeferenciar mejor tu lugar

Nombre*

Categoría General*

Categoría*

Posición*

Piso*

Radio de Emisión (metros)*

Propiedades

[+ Agregar Propiedad](#)

Usar Beacons (Opcional): **SI**

Eddystone name*

Eddystone instance*

Estabilidad*

Descripción

[Guardar](#)

Figura 6.14: Vista Portal Lazarillo: Nuevo Punto de Referencia

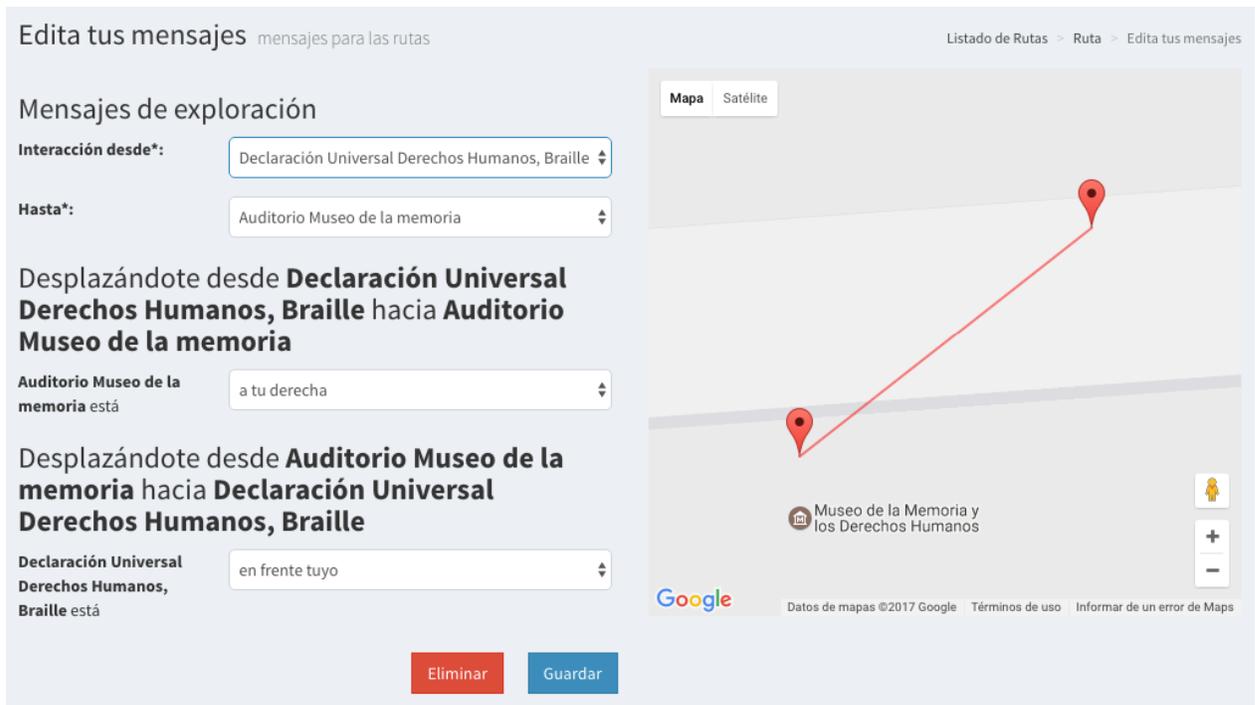


Figura 6.15: Vista Portal Lazarillo: Nuevo tramo dirigido

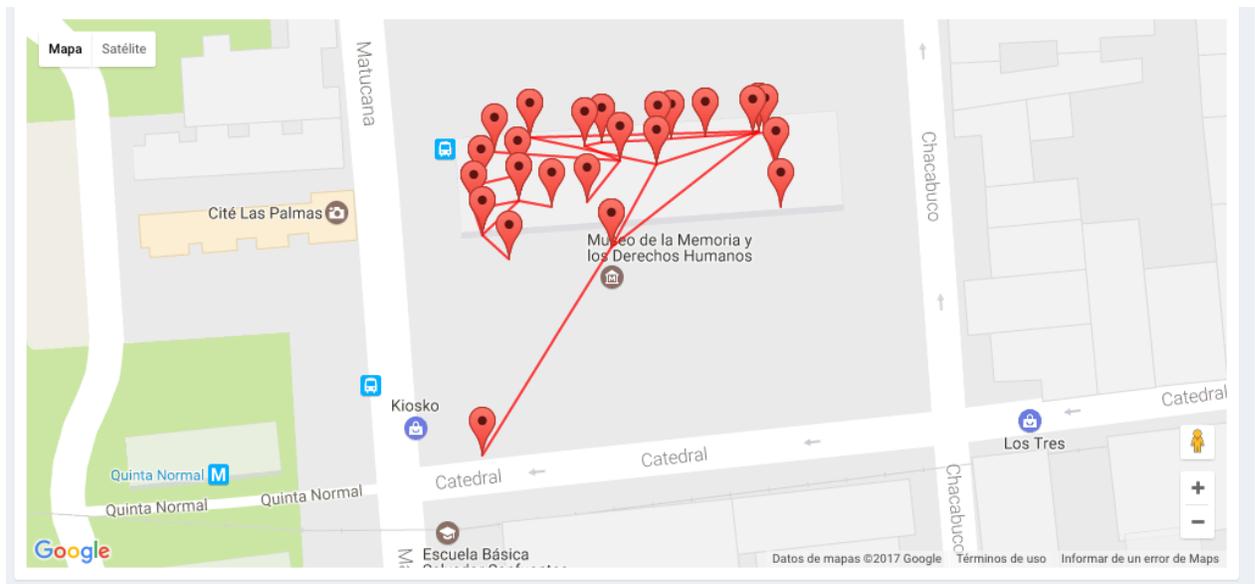


Figura 6.16: Vista Portal Lazarillo: Red de tramos dirigidos

Capítulo 7

Desarrollo de prototipo funcional

En esta sección se muestran las diferentes vistas propuestas para la implementación de la exploración indoor en Lazarillo, acorde con las líneas de diseño actualmente utilizadas en la navegación en espacios abiertos.

7.1. Vistas prototipo funcional e interacciones

Para las pruebas se ha desarrollado una prototipo de pruebas con dos objetivos, primero enfocado en el testeo del funcionamiento del sistema, estado de los beacons, sensibilidad del escaneo para calibración y, en general, la capacidad de mostrar en pantalla los distintos estados para corroborar que los algoritmos funcionan correctamente y/o realizar correcciones sobre la misma. El segundo objetivo para la realización de pruebas de usuario, cuyos detalles se muestran en la sección 8.4.3.

7.1.1. Características del Prototipo para Pruebas

El elemento más importante de esta aplicación es la consola, cuya función principal es de mostrar los datos crudos que se van capturando desde los beacons cercanos, como estimaciones de distancia, RSSI, identificador de instancia (paquete Eddystone UID) que permite reconocer de forma única a cada dispositivo e identificador del Punto de Referencia en el caso que exista un beacon asociado. La vista principal de esta aplicación se puede apreciar en la figura 7.1. No obstante lo anterior, la consola tiene la flexibilidad de modificar su contenido de acuerdo a lo que se esté realizando. Para las pruebas de usuario, la consola se utilizó para mostrar en pantalla el mensaje auditivo comunicado al usuario de forma de poder asistirlo ante algún problema presentado.

Uno de los problemas encontrados durante una de las pruebas fue que en ciertos lugares del museo la señal de internet móvil es muy baja e intermitente a ratos lo que produjo que Nearby API no lograra recuperar el identificador de algunos beacons durante el recorrido, haciendo que se perdiera el hilo del circuito. Se decide finalmente utilizar el SDK del fabricante (Kontakt) que permite la lectura de beacons de manera offline, la utilización de sensores adicionales de dichos

beacons y la lectura estándar de beacons de otro fabricante. Además la forma de implementarse es similar al Nearby API por lo que solo fueron necesarios algunos ajustes del código anterior.



Figura 7.1: Vista principal Prototipo para Pruebas

Otro problema presentado durante las pruebas iniciales fueron las inconsistencias en las estimaciones de distancia, principalmente entre dispositivos de distintos fabricantes y la utilización de distintas librerías. Se ha solucionado incluyendo radios de activación diferenciados para cada beacon, guardados como una propiedad de cada Punto de Referencia en la base de datos y la posibilidad de modificar el rango de emisión de los beacons dinámicamente desde el control "Rango" de la aplicación.

Se agrega un control para regular a velocidad del motor de voz luego de la realización del preexperimento 2, el cual se detalla en la sección 8.3.

Capítulo 8

Evaluación de usabilidad, impacto y resultados

8.1. Evaluación de usabilidad LazarilloApp

Como se ha explicado en secciones anteriores, el objetivo del trabajo es desarrollar una extensión del funcionamiento actual de Lazarillo para espacios interiores. En particular, se espera generar una experiencia de uso acorde con lo desarrollado hasta el día de hoy, de modo que para un usuario común de LazarilloApp la transición entre la exploración en espacios abiertos y cerrados se realice de la forma más transparente posible. Un potencial problema que puede llevar el enfoque anterior es que, al no existir conocimiento empírico acerca del grado de satisfacción de los usuarios respecto a la usabilidad actual de LazarilloApp, se podrían heredar problemas de usabilidad en el presente desarrollo.

Para prevenir lo anterior se ha llevado a cabo una encuesta de usabilidad de LazarilloApp, en el contexto de una encuesta de calidad de servicio, que le permitirá a Lazarillo corregir o potenciar distintas funcionalidades en base a la retroalimentación de los mismos usuarios. La metodología utilizada se basa en el modelo SERVQUAL o Service Quality Model, que define la calidad de servicio como la diferencia entre expectativas y percepciones de los clientes. Si las percepciones de los clientes superan las expectativas, implica una alta calidad del servicio y una alta satisfacción de las mismas. El instrumento utilizado para la realización de la encuesta fue auto aplicado, es decir de envío masivo en formato digital. La herramienta utilizada para la confección misma del formulario es Google Forms, principalmente por ser la única que cumple con los estándares de usabilidad para PeSD visual, siendo compatible con lectores de pantalla de dispositivos móviles y computadores de escritorio.

Se utiliza Servqual porque es un modelo comprobado, publicado por primera vez en el año 1988, y que ha experimentado numerosas mejoras y revisiones desde dicha fecha. Permite analizar aspectos cuantitativos y cualitativos de los clientes y conocer factores incontrolables e impredecibles de los clientes.

Respecto a la confección de la encuesta, se consideran 34 preguntas cerradas en escala Likert

con categorías específicas y 2 preguntas abiertas, que incluyen recopilación de información demográfica, nivel de satisfacción con la aplicación, nivel de satisfacción y expectativas con cada una de las funcionalidades.

Las dimensiones consideradas según la metodología SERVQUAL para la presente encuesta se observan en la figura 8.1

Dimensiones	Significado
Funciones	Las funciones que entrega la aplicación son las necesarias o no para que los usuarios puedan guiarse por la ciudad
Contenido	Si el usuario encuentra que la distribución o el formato en que se encuentra la información en la aplicación es necesaria para el correcto uso de esta. ¿Están todas las direcciones que los usuarios quieren?
Eficiencia	Si los tiempos en relación a la usabilidad de la aplicación son los adecuados para poder movilizarse dentro de la ciudad (aviso de calles próximas, búsqueda de dirección, entre otras)
Accesibilidad	La manera en que la aplicación se adapta a los usuarios en relación a la discapacidad visual, es decir, si los usuarios están conformes con la forma en que la aplicación les muestra todas las funciones que tiene para ellos

Tabla 8.1: Dimensiones SERVQUAL, encuesta usabilidad LazarilloApp

En un comienzo se realiza una muestra aleatoria de 1000 usuarios obteniéndose 53 respuestas. Dada la baja convocatoria se procede a la extensión de la muestra aleatoria a 7.431 usuarios, obteniéndose esta vez un total de 380 respuestas.

Inicialmente se considera la opción de enviar la encuesta de forma digital utilizando los correos, sin embargo al no existir doble confirmación de correo en el registro de la aplicación, muchos usuarios tienen mal registrados sus cuentas de correo, lo que se traduce en un alto rebote durante el envío. Por esta razón el envío del formulario se realiza a través de notificaciones push directamente a la aplicación, con un link que redirige a dicho formulario, en paralelo con el envío del correo, previo filtrado de las cuentas si es que ellas no tienen el formato correspondiente. El envío masivo se realiza utilizando el servicio de correo transaccional de Mailgun¹.

8.1.1. Resultados

A continuación se enuncian los resultados más relevantes encontrados.

Datos socio demográficos

- 1 de cada 4 usuarios son mujeres

¹<https://www.mailgun.com>

- El grupo etario más significativo es el que va en el rango de los 21 a 35 años. La distribución total se muestra en la figura 8.1
- El 54 % de los encuestados trabaja
- Un 50 % de los encuestados utiliza S.O. Android
- Un 75 % de los encuestados es extranjero. La distribución total de países se observa en la figura 8.2, sin embargo un 93 % es de habla hispana según figura 8.3
- El 42 % de los encuestados utiliza LazarilloApp para navegar por la ciudad caminando
- Un 79 % de los encuestados declara un grado de satisfacción buena o muy buena de experiencia con LazarilloApp

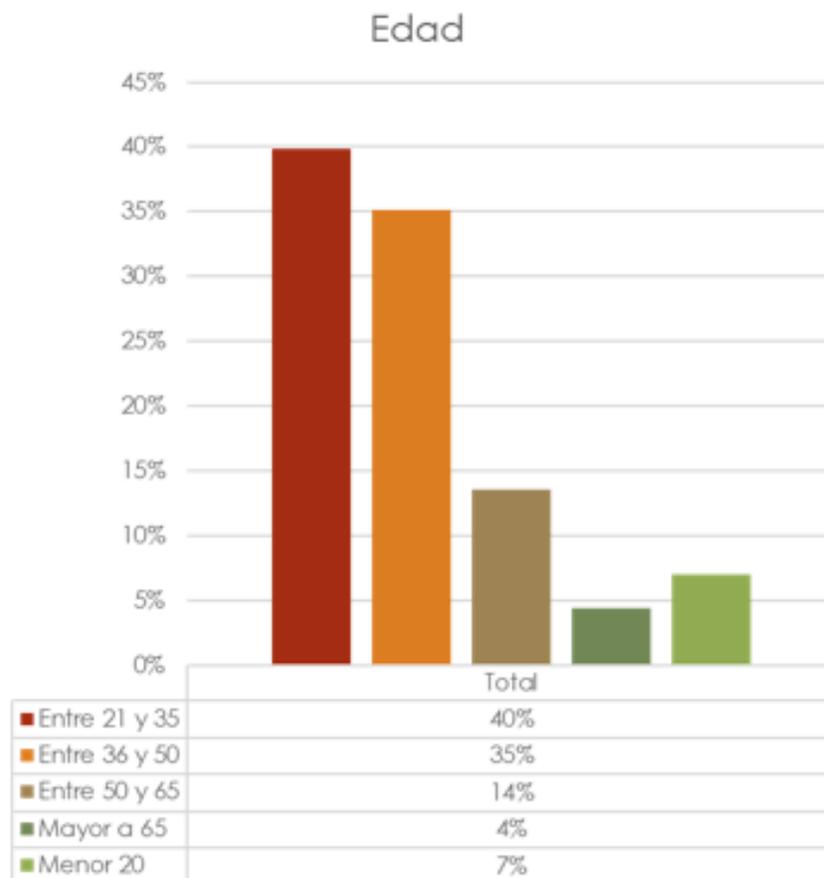


Figura 8.1: Distribución etaria

Cumplimiento de expectativas

En la tabla 8.2 se pueden observar distintos atributos y funciones con su respectivos índices de percepción, expectativas y cumplimiento de las mismas.

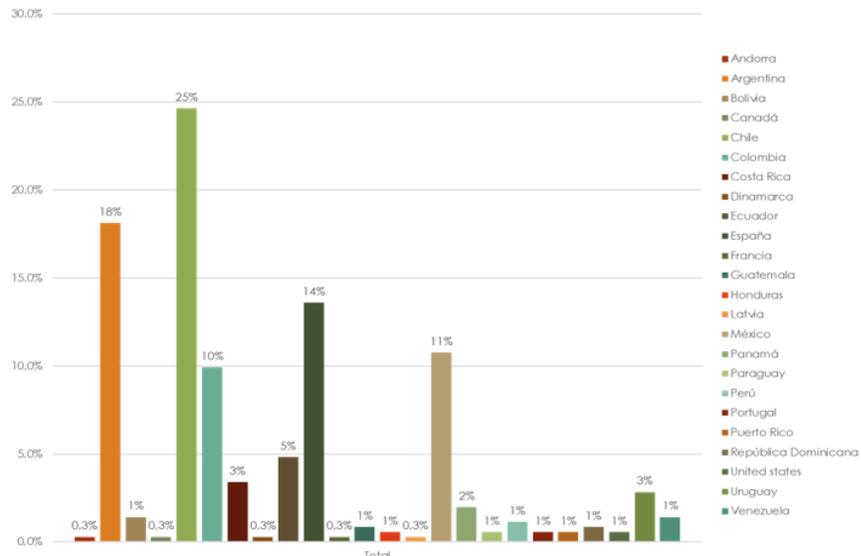


Figura 8.2: Distribución países

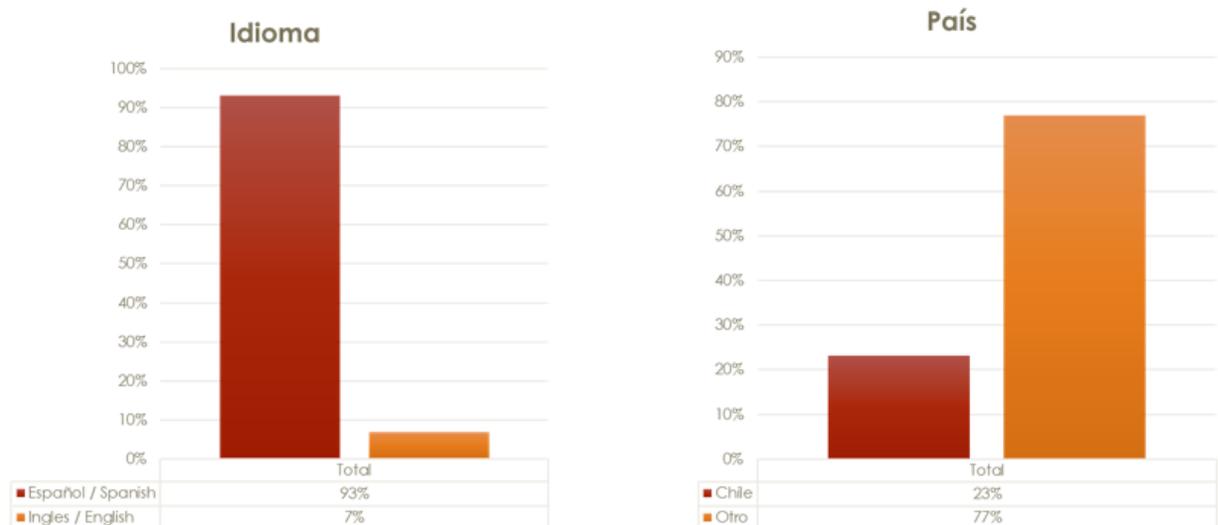


Figura 8.3: Idioma

Sugerencias de usuarios

- Mapas directamente de Lazarillo. No quieren usar aplicaciones externas.
- Mas precisión en las paradas de buses.
- Adaptar la LazarilloApp a cada país en relación al idioma.
- Datos móviles solo usarlos en caso de buscar una dirección, para no tener que ocupar siempre los datos móviles.
- Que LazarilloApp tenga un manual para orientarlos.
- Agregar más lugares como hospitales, comisarias, lugares históricos, estaciones de metro, centros comerciales, entre otros.
- Que sea más accesible para personas con baja visión (letras e iconos mas grandes).

Satisfacción de usuario	Percepción	Expectativas	Cumplimiento expectativas
Accesibilidad	4.3	4.9	88 %
Contenido	4.2	4.8	89 %
Eficiencia	4.1	4.9	84 %
Función Exploración	4.0	4.5	89 %
Función Búsqueda por Categorías	4.1	4.5	91 %
Función Agregar a Favoritos	4.2	4.4	96 %
Función Navegación	4.0	4.5	89 %

Tabla 8.2: Tabla satisfacción de usuarios

- Para las personas que usan LazarilloApp caminando, que le indique semáforos próximos, pasos de cebra, locales cerca, entre otros.

Conclusiones

- Tanto en el cruce de datos socio demográficos, como en el cruce de expectativas y percepción se logra con el cumplimiento de las expectativas.
- En relación a las dimensiones, la que logra un mayor grado de satisfacción en los usuarios es el contenido con un 89 %.
- La función con la que los usuarios se sienten mas satisfechos es con “Agregar a favoritos”, la cual logra un grado de satisfacción del 96 %
- Con el cumplimiento de las expectativas se logra una alta satisfacción por parte de los usuarios de LazarilloApp. Por consecuencia se concluye que si bien las funciones de Exploración y Navegación pueden ser mejoradas, los usuarios las evalúan con nota cercana al 90 % de calidad por lo que se mantiene la misma línea de usabilidad y aspecto para el desarrollo de las funcionalidades en espacios cerrados.

8.2. Preexperimento 1

Anterior a la realización del experimento se lleva a cabo un preexperimento, consistente en un recorrido por el circuito completo definido para el Museo de la Memoria junto con dos individuos de prueba. El objetivo principal es seleccionar un subcircuito para el experimento final que contenga diferentes elementos del entorno, para corregir posibles errores, adicionar sugerencias de mejora y, además, permita prever posibles complicaciones de la realización misma. En la tabla 8.3 se caracterizan los individuos de prueba.

A partir del recorrido de pruebas se logran identificar los problemas enunciados en la tabla 8.4, donde se desarrolla la descripción de los mismos y sus posibles causas.

Adicionalmente a los problemas antes mencionados detectados durante el recorrido de pruebas, los usuarios han indicado algunas sugerencias mostradas en la tabla 8.5.

Usuario	Edad	Sexo	Diagnóstico	Grado de visión	Periodo de diagnóstico	S.O. Smartphone	Periodo de uso
1	40	M	Catarata congénita	Ceguera	Toda la vida	iOS y Android	3 años y medio
2	35	F	Catarata congénita	Ceguera	Toda la vida	Android	2 años

Tabla 8.3: Perfiles usuarios de prueba, preexperimento

Problema	Descripción	Causa
Orden de activación erróneo de beacons	En algunas zonas del recorrido ciertos beacons se activan antes del orden lógico del recorrido, lo que se traduce en mensajes contextuales erróneos ya que el algoritmo entiende que el sentido del recorrido ha cambiado	Radios de activación mal configurados
Confusión mensajes de giro	Actualmente se tienen configurados mensajes del tipo “escalera a 3 metros, a tu derecha”, lo que se traduce en “continúa caminando en la dirección actual durante 3 metros y la escalera se encontrará a tu derecha”, se confirma que dicho mensaje se entiende de esta forma, debido al funcionamiento de Lazarillo, no obstante, si los mensajes en que se debe indicar a la persona que debe caminar hacia su derecha durante algunos metros se estaba indicando de la misma forma.	No se están diferenciando estos dos tipos de mensajes que requieren acciones completamente diferentes
Beacons que están a nivel de muro emiten a mayor potencia y se confunden entre ellos	Todos los beacons en el Museo de la Memoria, salvo en espacios exteriores, han sido instalados en los cielos de las diferentes exposiciones salvo siguiendo las recomendaciones de WayFindr y de los fabricantes, no obstante lo anterior, en algunas vitrinas se realizó la instalación bajo estas plataformas, paralelo al piso a 1 metro de distancia de éste	Estos puntos de interés se encuentran muy cercanos entre ellos. Los beacons instalados en esta posición emiten a mayor distancia utilizando los mismos filtros que los demás, posiblemente porque están a menor distancia de los dispositivos móviles de los usuarios, de forma tal que las señales se confunden entre las distintas obras.

Tabla 8.4: Problemas detectados durante preexperimento

Sugerencia	Descripción
Agregar mensaje de advertencia antes de llegar a la escalera	Ciertos elementos del entorno requieren una mayor atención por parte de los usuarios ya que pueden ser potencialmente peligrosos, como el caso de escaleras, plataformas y otros. Se sugiere agregar un mensaje de alerta antes de enfrentarse a dichos elementos.
Tipos de escalera	Ofrecer el detalle acerca de características especiales de algunos elementos del entorno que sean más complicados de recorrer. En el caso del Museo de la Memoria, existen escaleras de gran extensión con algunos descansos los que pueden hacer pensar a la PeSD visual que dicha escalera ha terminado, apareciendo de forma desprevénida la siguiente tanda de escalones.
Dos tipos de mensajes para beacons de mayor alcance	Agregar dos tipos de mensajes a los beacons de largo alcance, en particular los utilizados en espacios exteriores con radios de activación superiores a 6 metros, donde se entre un mensaje de larga distancia similar al utilizado en los anteriores pero un mensaje cuando la persona se encuentre cercana a un metro de distancia que indique que está ahí.

Tabla 8.5: Sugerencias propuestas por usuarios de prueba durante preexperimento

8.2.1. Correcciones

A partir de los problemas encontrados durante la realización del preexperimento, se realizan algunas correcciones que se detallan a continuación.

Cambio en el algoritmo de emisión de mensajes contextuales

El algoritmo utilizado durante el preexperimento para la construcción de mensajes contextuales era extremadamente simple, sin embargo, solo funcionaba en condiciones ideales. Para mostrar el problema existente, imaginar el siguiente flujo de activación de beacons en condiciones ideales o teóricas, dada la distribución de ellos en el espacio y las rutas esperables de movimiento de los individuos de prueba en el espacio versus el flujo de activación obtenida en terreno mostrada en la figura 8.4. En este caso de ejemplo, el beacon 4 se activa antes que el beacon 3 en un hecho fortuito -recordar que las áreas de cobertura de los beacons no son estáticas y, dependiendo de las condiciones del espacio, pueden expandirse o contraerse más allá del radio teórico-. Dado el orden de activación teórico, los segmentos dirigidos existentes para la construcción de mensajes en este escenario corresponderían a los pares mostrados en la figura 8.5

Según el algoritmo inicial, los mensajes a desplegarse según el flujo de activación en terreno mostrado como ejemplo en la figura 8.4, corresponderían a los de los segmentos: AB, No hay segmento asociado (mensaje genérico C), DC, No hay segmento asociado (mensaje genérico E). Se puede apreciar que el único mensaje contextual arrojado correctamente es el correspondiente al segmento AB, para otros dos no existe mensaje contextual y para DC el mensaje contextual se

Orden de activación teórico
A - B - C - D - E
Orden de activación en terreno
A - B - D - C - E

Figura 8.4: Ejemplo flujos de activación de beacons teóricos vs en terreno

AB, BC, CD, DE
Y en sentido inverso
ED, DC, CB, BA

Figura 8.5: Segmentos dirigidos

lanza pero en sentido inverso, es decir se lanza un mensaje posiblemente incorrecto que puede confundir notoriamente el desplazamiento de la PeSD visual. En la figura 8.6 se propone un algoritmo mejorado que logra recuperar mensajes perdidos a causa de malas lecturas. Esto soluciona tanto los problemas presentados por no lectura de un beacon particular y los problemas asociados a lecturas en desorden.

La función “construirSegmentoAnterior(beacon, prevBeacon)” del nuevo algoritmo recupera el arco anterior al beacon enunciado, es decir, si el beacon activado es el beacon D (y el anterior el B según el ejemplo de activación de la figura 8.4) se realiza el siguiente proceso: primero se obtiene el conjunto de los beacons que son nodos iniciales de los segmentos cuyo nodo final es el beacon D, es decir los beacons C (segmento CD) y E (segmento ED), según la figura 8.5; luego se obtiene el conjunto de beacons que son nodos finales de los segmentos cuyo nodo inicial es el beacon B, es decir los beacons C (segmento BC) y A (segmento BA). La intersección de ambos conjunto es el beacon C, recuperandose así el segmento CD.

Finalmente el algoritmo obtiene, para el mismo input de activación en terreno mostrado en la figura 8.4, el output mostrado en la figura 8.7. Se aprecia como ha recuperado dos mensajes que en el algoritmo anterior se generaban como mensajes genéricos y ha transformado el mensaje contextual que anteriormente se generaba erróneamente por otro genérico.

8.3. Preexperimento 2

El preexperimento 2 corresponde a un recorrido de pruebas a través del circuito obtenido en el Preexperimento 1 y utilizando las mejoras del algoritmo. El objetivo es familiarizarse con el procedimiento explicado en la sección 8.4.3 e identificar posibles fallas en la implementación de la prueba.

La prueba se realiza con un individuo caracterizado en la tabla 8.6.

Luego de realizado el recorrido, se identifican dos problemas:

- El primer beacon no esta siendo leído. Se logra deducir que dada su posición perpendicular al suelo, a diferencia de los beacons instalados en el interior del museo, produce el efecto de que al circular muy cercano a la pared, las ondas emitidas tienden a pasar por encima del

```

// Construccion de mensaje contextual: posicion relativa
/* Inicializar variables aux */
prevBeacon:= "";
prevPrevBeacon:= "";
lecturaBeacon(beacon){
    if(existeSegmento(prevBeacon, beacon) &&
        !existeSegmento(prevPrevBeacon, beacon)){
        emitirMensajeSegmento(prevBeacon, beacon);
    } else if(!existeSegmento(prevBeacon, beacon) &&
        existeSegmento(prevPrevBeacon, beacon)){
        emitirMensajeSegmento(prevPrevBeacon, beacon);
    } else if(!existeSegmento(prevBeacon, beacon) &&
        !existeSegmento(prevPrevBeacon, beacon)){
        construirSegmentoAnterior(prevBeacon, beacon);
        emitirMensajeSegmentoConstruido();
    } else {
        emitirMensajeGenerico(beacon);
    }
    prevPrevBeacon := prevBeacon;
    prevBeacon := beacon;
}

```

Figura 8.6: Algoritmo de construcción de mensajes contextuales de posición relativa mejorado

AB, CD, Mensaje genérico C, DE

Figura 8.7: Output mensajes según segmento recuperado con algoritmo mejorado

Usuario	Edad	Sexo	Diagnóstico	Grado de visión	Periodo de diagnóstico
1	61	M	Retinitis Pigmento- sa	Ceguera	Toda la vida

Tabla 8.6: Perfiles usuario de prueba, preexperimento 2

usuario evitando así su lectura. Se corrige esta situación aumentando la potencia de emisión del beacon.

- El usuario no se siente cómodo con la velocidad con el motor de voz funciona. Se corrige esta situación agregando una barra de velocidad que se configura en presencia del usuario.
- Durante la prueba se acrecientan problemas de conectividad producto de intermitencias en el servicio de internet móvil en el dispositivo. Por esta razón se reemplaza el uso del Nearby API por el SDK del fabricante (Kontakt), los detalles de este punto se pueden apreciar en la sección 7.1.1.

8.4. Experimento

8.4.1. Instrumentación

Para llevar a cabo la experiencia y recopilación de datos se han utilizado las siguientes herramientas y dispositivos:

- Aplicación Prototipo de Pruebas: utilizada tanto para el escaneo de beacons, generación de mensajes auditivos con instrucciones, sonidos y configuración inicial tal como se explica en la sección 7.1.1
- Alcatel Idol 3: dispositivo móvil de gama media utilizado para correr aplicación Prototipo Pruebas de Usuario, bajo Android 5.0.2. El dispositivo cuenta con las siguientes especificaciones relevantes: Procesador Qualcomm MSM8939 Snapdragon 615 octa-core, 2 GB RAM, 16 GB de almacenamiento interno y Bluetooth 4.0 (característica fundamental para la lectura de beacons).

8.4.2. Grupo experimental

En la tabla 8.7 muestran los perfiles de los usuarios de prueba para el experimento de usabilidad.

En el campo de UX y usabilidad existe un cierto consenso con respecto al número óptimo de participantes para la realización de los test de usabilidad. Nielsen y Cockton[32], Faulkner[33] y otros investigadores concuerdan en que 5 usuarios son suficientes, en promedio, para reconocer cerca de un 85% de los problemas de usabilidad del software en estudio, lo que justificaría la realización del presente estudio sobre un grupo de prueba de 6 usuarios.

El siguiente test de usabilidad para el usuario final se realiza en base a la metodología de estudio

Usuario	Edad	Sexo	Diagnóstico	Grado de visión	Periodo de diagnóstico
1	60	F	Glaucoma	Ceguera	Desde los 10 años
2	37	M	Queratocono	Baja visión	Desde los 25 años
3	48	F	Glaucoma vilateral	Ceguera	Toda la vida
4	26	F	Coloboma vilateral	Baja visión	Toda la vida
5	68	M	Retinopatía pigmentaria	Ceguera	Toda la vida
6	25	M	Retinopatía congénita	Baja visión	Desde los 11 años

Tabla 8.7: Perfiles usuarios de prueba, experimento

de caso, dada la dificultad para contar con un número elevado de individuos para las pruebas, involucrando un análisis para cada uno de los participantes del [34]. Dado lo anterior, la razón principal para la elección de esta metodología es que esta no requiere de un número mínimo de casos ni tampoco la elección aleatoria de individuos de prueba [35]. A través del estudio de caso se pretende contribuir a la construcción de conclusiones para el estudio completo.

8.4.3. Diseño del experimento

Consideraciones iniciales

Como fue explicado en la sección 5.2.1, el Modo Exploración planteado en el presente trabajo tiene como objetivo otorgar una forma que le permita a la PeSD visual, identificar elementos cercanos en la medida que se desplaza por espacios interiores, de manera similar a como lo haría una persona sin discapacidad visual reconociendo un lugar. Por esta razón, el usuario necesitará una guía que le permita desplazarse por el museo a falta de un sistema de ruteo implementado en la aplicación. Para este fin, usuario recibirá instrucciones de ruteo de un tercero, denominado comúnmente como experimento Mago de Oz, quien le indicará los giros adecuados de manera de que todos los usuarios realicen el mismo circuito, pero que lo recorran a su manera tanto en velocidad, desvíos, distancia relativa a las paredes, detenciones y otras variables.

Durante el recorrido el usuario recibirá mensajes auditivos que le indicarán distintos elementos al interior del museo que le permitirán reconocer los distintos espacios y generar recordación. En la actualidad, gran parte de estos puntos de interés marcados en el Prototipo de Pruebas son vitrinas y otros elementos no identificables por PeSD visual que no permiten su interacción por lo que un tercero será responsable de indicar el lugar preciso del elemento en cuestión. En un futuro cercano estos puntos marcados serán elementos interactivos que le permitirán a la persona identificarlos de manera fácil.

Implementación

El experimento comienza con una breve introducción al usuario de cerca de 15 minutos acerca del uso de la aplicación y del experimento mismo. Además se configura el motor de voz en la velocidad que le acomode al usuario a un volumen adecuado.

Luego se le solicita al usuario posicionarse en el punto A del recorrido, tal como se observa en las figuras 8.8 y 8.9. Una vez posicionado en el punto de partida se le hace entrega del dispositivo móvil con el Prototipo de Pruebas cargado y funcionando, el cual debe llevar durante todo el circuito en su mano, a la altura de su pecho. Antes de comenzar se le dan las instrucciones para recorrer el primer segmento informándole que, para cada mensaje recibido, una persona le indicará de forma precisa el lugar que el mensaje entrega de manera tal que el usuario pueda reconocer el error si es que existe.

El usuario en ningún momento debe interactuar con el dispositivo móvil, los mensajes se generan de forma automática en la medida que la persona se desplaza por el lugar.

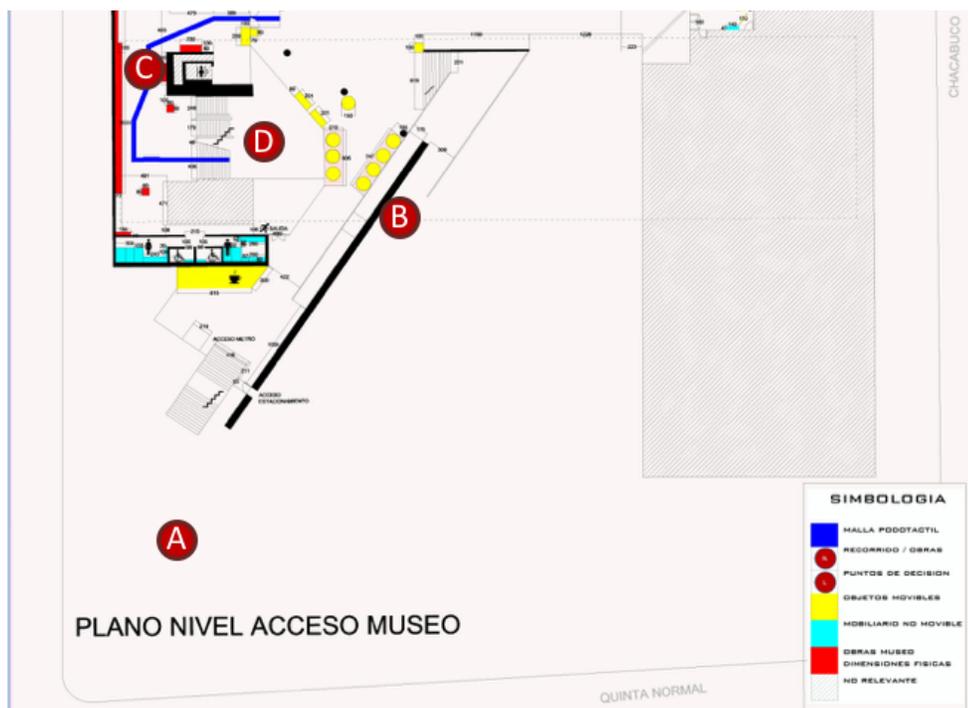


Figura 8.8: Plano recorrido Explanada

El usuario puede comenzar el recorrido cuando estime conveniente, momento en cual un tercero presionará el botón “START” lo que iniciará el escaneo de beacons. Una vez presionado, la aplicación suministrará los mensajes auditivos que le permitirán al usuario reconocer espacialmente los diferentes puntos de interés, donde el primer elemento de interacción correspondiente a la placa braille de Derechos Humanos. El circuito completo comprende los nodos A-B-C-D-E-H-I-Ñ-O-P-Q-S-R-T-Q-P-O-Ñ-I-H-E, incluyendo los puntos de interés que van desde el 0 al 15 sin contabilizar el punto 7 por problemas de emisión del beacon detectados durante las pruebas. El circuito puede ser pausado durante la realización de la prueba en el caso de existir problemas evidentes de movilidad, producto de aglomeraciones de los visitantes del museo. El botón “STOP” permitirá realizar

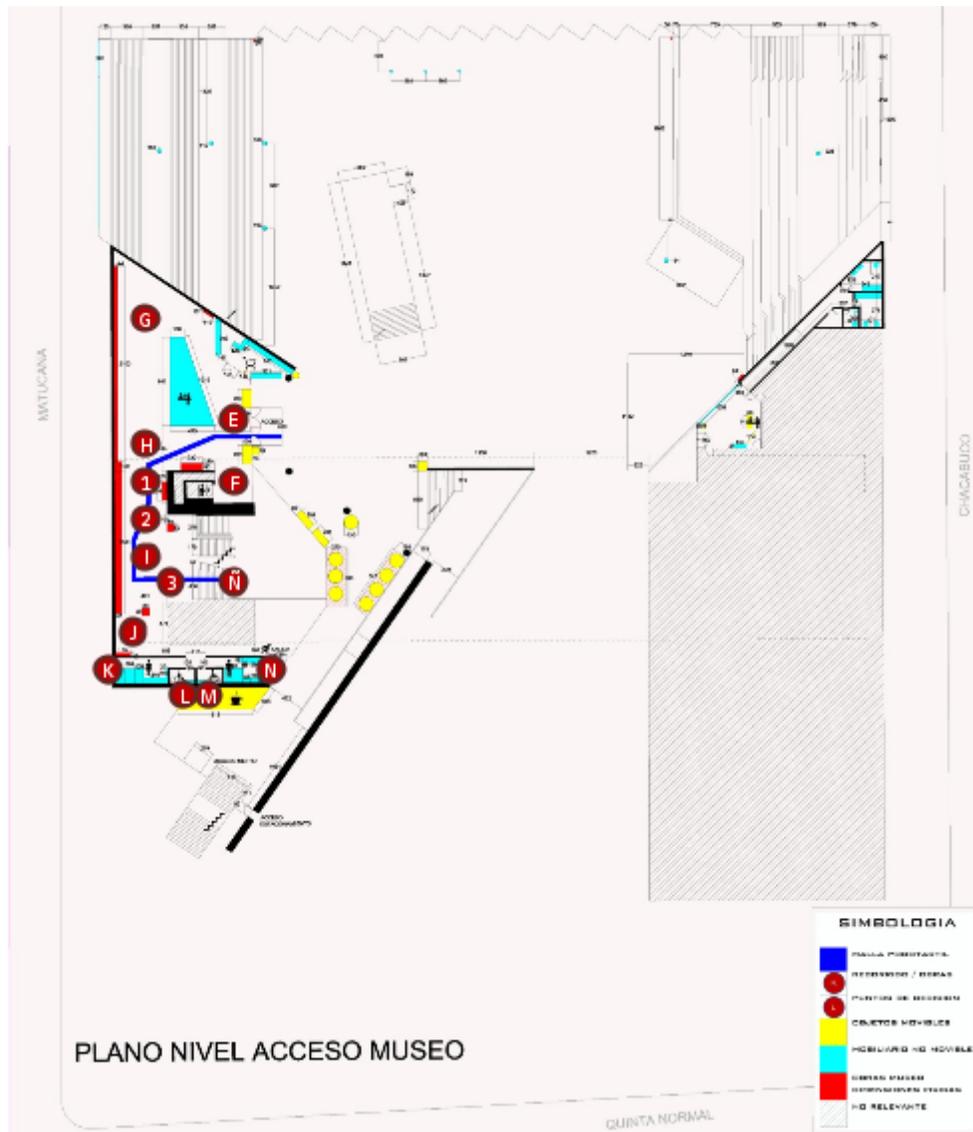


Figura 8.9: Plano recorrido Planta Baja

la pausa. El recorrido termina al volver a la puerta de entrada representada en el nodo E.

Todos los mensajes generados durante el recorrido son guardados en el dispositivo móvil una vez terminado el circuito y presionando el botón “SAVE”. Los mensajes luego son analizados de forma manual para la construcción de las métricas, explicadas en la sección 8.4.4.

Una vez terminado el recorrido se procede a la realización de la evaluación de usabilidad, basada en la pauta resumida usuario final de la “Evaluación de Usabilidad de Software Para Niños Ciegos” desarrollada por el profesor Jaime Sánchez y utilizada también en la Evaluación del sistema de posicionamiento en ambientes cerrados [7], consistente en 16 afirmaciones con 10 niveles de evaluación, donde el valor 1 corresponde a una afirmación que el usuario considera completamente alejada de la realidad y el valor 10 a una afirmación que el usuario comparte completamente. Las afirmaciones consideradas en este test son: “Me gusta la aplicación”, “La aplicación es útil”, “La aplicación me hace estar activo”, “Volvería a usar esta aplicación”, “Recomendaría esta aplicación

a otros”, “Aprendí con esta aplicación”, “Me sentí en control de la situación”, “La aplicación es interactiva”, “La aplicación es fácil de utilizar”, “La aplicación es motivadora”, “La aplicación se adapta a mi ritmo”, “La aplicación me permitió reconocer el espacio”, “Me gustan los sonidos de la aplicación”, “Los sonidos de la aplicación son claramente identificables”, “Los sonidos de la aplicación me transmiten información”.

Para terminar se realiza una serie de 5 preguntas abiertas basadas en la misma pauta de evaluación del profesor Jaime Sánchez, 2 de las cuáles han sido confeccionadas para este caso particular. Las preguntas son: “¿La aplicación es útil para ti?”, “¿Qué es lo que más te gustó de la aplicación?”, “¿Qué cree que le falta a la aplicación para mejorarla?”, “¿Entendiste los mensajes o te confundieron?”, “¿Te sentiste seguro durante el trayecto?”.

Escenario

Las pruebas han sido desarrolladas en un circuito acotado dentro del Museo de la Memoria. Actualmente el museo no cuenta con espacios totalmente accesibles, encontrándose vitrinas, bancas y otros elementos en los diversos espacios que pueden ser de especial dificultad para PeSD visual, razón por la cual el proyecto final considera también la instalación de guías podotáctiles. Todos los elementos antes mencionados podrían mermar el desempeño del presente proyecto por lo que se sugiere, para trabajos futuros, la realización del mismo experimento utilizando la guía podotáctil para medir el impacto que tienen otras herramientas complementarias en la calidad de la navegación.

8.4.4. Métricas

Durante el recorrido se guardan todos los mensajes auditivos en forma de texto para luego, de forma manual, realizar la construcción de las siguientes variables cuantificables:

- Tasa de mensajes con información contextual errónea: relación entre mensajes contextuales indicados de forma errónea (e.g. se indica un elemento hacia el costado opuesto) por sobre el total de mensajes generados.
- Tasa de mensajes acertados sin información contextual: relación de mensajes genéricos generados por falta de información por sobre el total de mensajes generados.
- Tasa de mensajes acertados con información contextual: relación entre mensajes contextuales acertados sobre el total de mensajes generados.
- Tasa de mensajes fuera de recorrido: relación entre el total de mensajes emitidos fuera de lugar (e.g. se indica el baño del segundo piso estando en el primero) por sobre el total de mensajes generados.
- Tasa de mensajes no emitidos: Relación entre el número de mensajes que no fueron emitidos y que debieron serlo sobre el total de mensajes que debieron ser emitidos.

8.5. Análisis y resultados

Luego de realizadas las pruebas con usuarios se analizan los registros guardados en la aplicación para cada uno de los usuarios. Las tasas obtenidas se muestran en la figura 8.8.

Métrica	Valor
Tasa de mensajes con información contextual errónea	0.065
Tasa de mensajes acertados sin información contextual	0.090
Tasa de mensajes acertados con información contextual	0.709
Tasa de mensajes fuera de recorrido	0.136
Tasa de mensajes no emitidos	0.070

Tabla 8.8: Variables cuantificables en experimento

Algunos alcances antes de realizar el análisis de los resultados:

- El total de mensajes emitidos no coinciden a pesar de realizarse los mismos recorridos. Existen varias causas para esta situación: recorrer el circuito a una velocidad más baja produce que ciertos elementos se repitan al cumplirse la condición de tiempo, la humedad del ambiente produce que la lectura se vea potenciada o mermada, la cercanía a las paredes con la que se desplaza el usuario, entre otras variables.
- Dado que las pruebas se realizaron en horario de atención, en algunos recorridos se tuvieron que efectuar pausas producto de la congestión en pasillos que no permitía la realización expedita de la prueba, principalmente por la visita de cursos completos de colegio. El reanudar el recorrido desde la aplicación, se borran las variables locales lo que hace que el primer elemento luego de la reanudación se entregue siempre sin la información contextual de posición, por lo que esa lectura no se considera como mensaje emitido. De retomarse el recorrido desde el mismo punto donde se realizó la pausa, ese último elemento no es contabilizado modificando el total de mensajes emitidos.
- Para cada mensaje emitido, a la persona se le indica donde se encuentra precisamente el elemento. Esto no está considerado en las métricas anteriores pero será capturado de forma indirecta en la evaluación de usabilidad, donde el usuario puede indicar su conformidad o disconformidad con el grado de precisión de los mensajes emitidos.

Para una evaluación correcta de los resultados anteriores, es importante entender el impacto que tiene cada una de las variables consideradas:

- Mensajes con información contextual errónea: es el mensaje más dañino ya que se entrega información errada, indicando un elemento en una posición equivocada lo que produce desconcierto en el usuario. Impacto: -3
- Mensajes acertados sin información contextual: Este tipo de mensajes se produce cuando el algoritmo por falta de información no puede identificar el sentido en que se recorre el circuito al momento de activar el beacon en cuestión. Este mensaje es acertado pero carece de precisión. Es preferible este tipo de mensajes frente a la no emisión del mismo. Impacto: 1
- Mensajes acertados con información contextual: Este es el tipo de mensaje que se espera obtener en la mayoría de los casos ya que entrega información genérica del elemento más

información contextual referente a su posición, que le permite al usuario identificarlo de forma óptima. Impacto: 3

- Mensajes fuera de recorrido: Son mensajes acerca de elementos que no se encuentran cercanos al punto del recorrido y no siguen un orden lógico. Mayoritariamente se entregan de forma genérica (sin información contextual de posición) y corresponden a elementos emitidos anteriormente por lo que el usuario logra identificarlos como elementos fuera de lugar. Son mensajes no deseables pero menos dañinos que los primeros. Impacto: -1
- Mensajes no emitidos: Mensajes que debieron ser emitidos pero que por distintas causas fueron omitidos. Impacto: 0

En la figura 8.10 se puede apreciar de forma gráfica la distribución de mensajes para el total de pruebas, considerando las primeras 4 variables. Un 70.9% de los mensajes emitidos corresponde a los mensajes de mayor precisión, mientras que los mensajes emitidos incorrectamente alcanzan solo un 6.5% de las emisiones totales. En términos generales, considerando tanto los mensajes acertados con y sin información contextual, se alcanza una cifra cercana al 80% de mensajes emitidos correctamente.

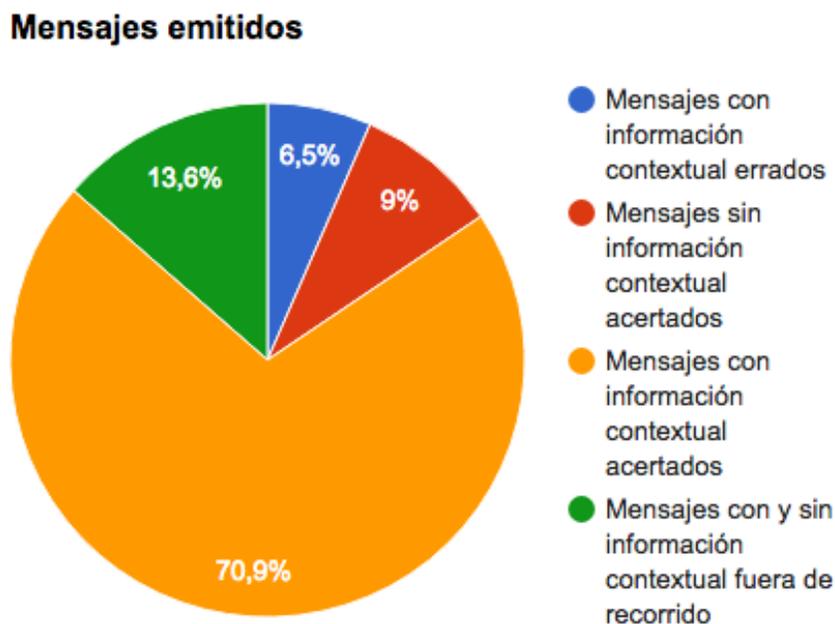


Figura 8.10: Proporción mensajes emitidos

En términos aún más estrictos, se procede a comparar el modelo utilizado con uno que solo sea capaz de generar mensajes genéricos sin información contextual de posición. Para este ejercicio se utilizarán las métricas de impacto definidas anteriormente para cada una de las variables. De esta forma se pueden comparar cuantitativamente como se observa en la figura 8.9

Se considera que dado solo un cambio de modelo, manteniéndose todo lo demás constante, los mensajes que no fueron emitidos en el primer modelo, tampoco serán emitidos en el segundo. En

Métrica	Frecuencia	Impacto	Modelo	Genérico
Mensajes con información contextual errónea	13	-3	-39	0
Mensajes acertados sin información contextual	18	1	18	199
Mensajes acertados con información contextual	141	3	423	0
Mensajes fuera de recorrido	27	-1	-27	0
Mensajes no emitidos	15	0	0	15
Total	214		375	199

Tabla 8.9: Tabla comparativa modelo planteado versus modelo mensajes genéricos

estas condiciones, el modelo planteado en el trabajo muestra un rendimiento superior al modelo de mensajes genéricos. A continuación se analizan los resultados obtenidos del test de usabilidad.

Para el análisis de la Pauta de Usabilidad de Usuario Final de Sánchez, las diferentes sentencias se agrupan en 3 categorías: Satisfacción, Control & Uso y Sonidos. Para cada una de las categorías nombradas se obtiene un promedio de todos los usuarios de la muestra. Los resultados se pueden observar en la figura 8.11, donde es posible verificar que para todas se ha obtenido un valor promedio superior a los 8 de 10 puntos, muy por sobre las expectativas iniciales: Satisfacción ha obtenido una calificación de 9.028; Control & Uso, 8.806; Sonidos, 8.778. Todos los valores con un error menor a los 1.6 puntos.

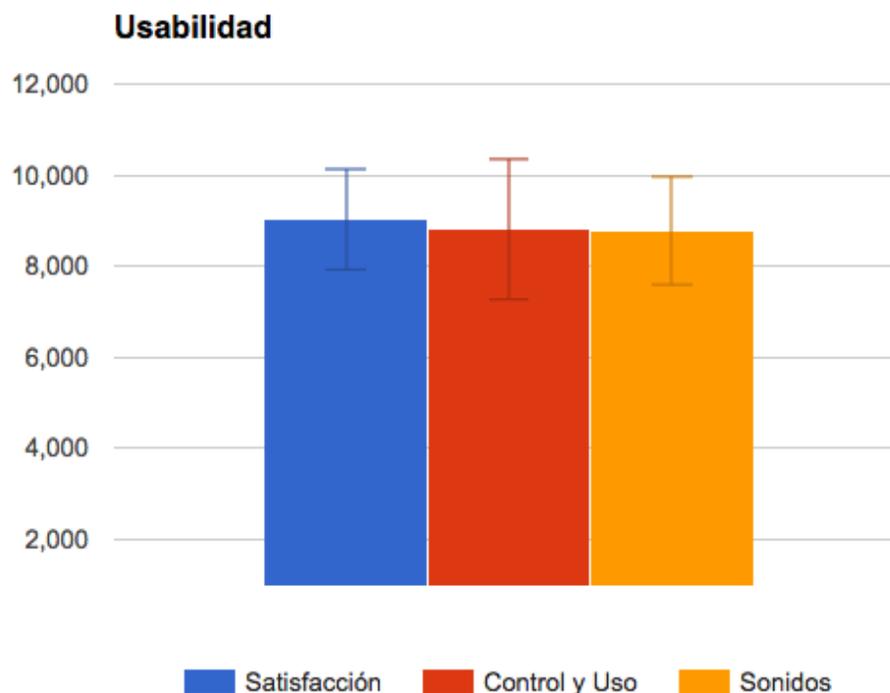


Figura 8.11: Resultados test de usabilidad

Por otra parte la sentencia “Los mensajes me ayudan a comprender mi entorno” ha sido incluida de forma adicional para verificar el impacto real del prototipo desarrollado, obteniéndose un valor promedio de 9.333 de los 10 totales y un error cercano a los 1.2 puntos.

Adicionalmente a lo anterior, frente a las preguntas abiertas realizadas en el test, se obtienen los siguientes resultados:

- Frente a la pregunta “¿La aplicación es útil para ti?” el 100 % de los usuarios contesta que sí. Uno de los encuestados asegura que considera la aplicación útil a pesar de que no es del todo certera con algunos mensajes.
- Lo que más gustó de la aplicación se relaciona con la forma en que se comunica el espacio. La pregunta “¿Qué es lo que más te gustó de la aplicación?” da cuenta de ello, donde todos los usuarios muestran conformidad con el tipo de información entregada y el nivel de detalle.
- No existe un consenso frente a la pregunta “¿Qué cree que le falta a la aplicación para mejorarla?”, algunos apuntan a ofrecer más información en el recorrido, lo cual no era parte de las pruebas pero sí parte del proyecto final a desarrollarse en el Museo de la Memoria y otros usuarios comentan sobre la inclusión de personalización del motor de voz, lo cual está considerado como parte de la implementación del prototipo dentro de LazarilloApp².
- En general los mensajes auditivos construidos fueron comprendidos. Algunos usuarios consideran que el mensaje contextual “a tu derecha” para indicar un elemento que se encuentra al costado derecho puede mejorarse utilizando “al costado derecho”, sin embargo comprenden el mensaje como está.

Un resultado sorprendente es que el usuario 4 declara tener un grado de visión bajo, pero ostensiblemente mejor que el de los demás usuarios que realizaron el test, tanto así que puede desplazarse sin el uso de bastón. Se esperaría entonces que el grado de satisfacción de la aplicación para este usuario fuera menor con respecto al resto, bajo el supuesto de que para dicho usuario la aplicación no es imprescindible. No obstante lo anterior, el usuario registra la segunda mejor puntuación para el ítem de satisfacción, contradiciendo la hipótesis inicial.

²LazarilloApp cuenta con la opción de personalizar motor de voz

Capítulo 9

Conclusiones y trabajo futuro

9.1. Conclusiones del trabajo

El desarrollo de aplicaciones para PeSD visual conlleva ciertas precauciones a tener presentes, primordialmente relacionadas con evitar la tendencia a formular hipótesis de comportamiento y validarlas en base a nuestra propia experiencia y comportamiento como persona sin situación de discapacidad visual (PsSD visual), lo cual en la mayoría de los casos conduce a error. Es en este punto donde las pruebas de usuario cobran real valor y ayudan a mejorar el producto final. Un claro ejemplo de esto ocurrió durante el desarrollo del segundo pre experimento donde, a pesar del sinnúmero de pruebas exitosas realizadas con anterioridad, no se logra realizar la lectura del primer beacon del recorrido, incluso repitiendo el recorrido ante la posible falla del smartphone. Finalmente se deduce que el problema de lectura es producido por el comportamiento mismo del usuario, quien tiene la tendencia natural a caminar muy cercano a la pared que utiliza como guía y que en este caso, al tener el beacon adosado, producía un efecto en que las ondas emitidas pasaban por encima del usuario haciendo al beacon invisible. Finalmente la solución consistió en aumentar la potencia del beacon en cuestión.

El catálogo de aplicaciones y, en general, de soluciones tecnológicas para PeSD visual es muy acotado. La razón principal es el “pequeño mercado” en relación al mercado de PsSD visual, por lo que se tiende a elegir esta última dado el N más grande de potenciales usuarios. Así, los desarrolladores que pretendan aventurarse en el mercado de aplicaciones para PeSD visual optarán por cobrar altos precios para hacer sustentable el negocio (lo que sucede en la mayoría de los casos) o generar modelos de negocio más ingeniosos que eviten cargarle el peso total a los mismos usuarios como es el caso de Lazarillo. Un aprendizaje importante obtenido a través del desarrollo de la presente investigación y del trabajo realizado en Lazarillo es que probablemente debido a lo anterior, la mayoría de los usuarios tiene muy bajas expectativas con respecto a desarrollos tecnológicos y en particular de aplicaciones. Esto se traduce en una tendencia a conformarse con muy poco en la mayoría de los casos, lo que puede llevar al desarrollador a dejarse influenciar y creer que el trabajo está terminado antes de tiempo.

La utilización de la metodología de trabajo Design Thinking ha permitido desarrollar el trabajo de una forma ágil y rápida, permitiendo adaptar el desarrollo de acuerdo a las necesidades,

descubrimientos o fallos capturados durante el proceso. Tal vez uno de los grandes desafíos del trabajo consistió en lograr sintetizar el gran número de técnicas y herramientas disponibles en la actualidad, con el propósito particular de habilitar espacios interiores orientado a un segmento con requerimientos específicos fuertes. Muchas de las opciones que parecían viables en el papel fueron quedando descartadas en el proceso gracias al prototipado-testeo en ciclos cortos.

9.2. Trabajo futuro

Se proponen algunas líneas de desarrollo que podrían potenciar aún más el trabajo desarrollado:

- En base a los resultados obtenidos en las pruebas con usuarios, se propone mejorar el Modo Exploración planteado, intentando evitar la generación de mensajes fuera de recorrido. Para esto se debe modificar el algoritmo de modo que se realicen comparaciones no solo con uno o dos lecturas previas de beacons, sino que con una visión mas general, lo que de forma indirecta reducirá también el número de mensajes con información contextual errada.
- El Modo Exploración en base a segmentos dirigidos sienta las bases para el ruteo en espacios cerrados. El desafío en este punto, más allá de la elección de un método de optimización coherente con el problema, es la construcción automatizada de los mensajes que permitan el desplazamiento entre puntos. El sistema debe ser capaz de generar el mensaje o set de mensajes para el desplazamiento entre nodos que reduzca al mínimo la manipulación humana de modo que genere mensajes consistentes con el funcionamiento en otros contextos similares y sean coherentes con la línea de mensajes de LazarilloApp.
- Si bien el Modo Exploración propuesto es de gran ayuda para PeSD visual que se encuentran en ambientes desconocidos, su complementación con algún método de ruteo que permita generar indicaciones similares a las utilizadas de forma manual en el prototipo funcional es esencial. Otro desafío importante es lograr que el ruteo generado en espacios abiertos - actualmente entregado por un servicio externo que funciona en paralelo con la aplicación- se integre con el ruteo en espacios cerrados y entregue resultados que sean fáciles de interpretar y que permitan a los usuarios realizar una planificación previa de su viaje.
- Complementar el uso de beacons con otras tecnologías de uso común como Wifi, acelerómetro, entre otras, que permitan mejorar la estimación de las coordenadas del usuario en el espacio y, por consiguiente, mejorar la precisión de los mensajes.

Bibliografía

- [1] R. I. Hartley and P. Sturm, 'Triangulation,' *Computer vision and image understanding*, vol. 68, no. 2, pp. 146-157, 1997.
- [2] J. Hightower and G. Borriello, 'Location sensing techniques,' *IEEE Computer*, vol. 34, no. 8, pp. 57-66, 2001.
- [3] R.-H. Jan, H.-C. Chu, and Y.-F. Lee, 'Improving the accuracy of cell-based positioning for wireless networks,' *Computer networks*, vol. 46, no. 6, pp. 817-827, 2004.
- [4] R. Mautz, 'Indoor positioning technologies,' Ph.D. dissertation, Habilitationsschrift ETH Zürich, 2012, 2012.
- [5] R. Espinoza, "Sistema de computación ubicua para relacionar a personas con discapacidad visual con su entorno," 2015.
- [6] S.-H. C. J.-Y. K. W. J. L. B.-J. Shin, K.-W. Lee and H. S. Kim, "Indoor wifi positioning system for android-based smartphone," pp. 319–320, 2010.
- [7] M. A. S. Correa, "Sistema de posición y orientación móvil para personas ciegas en ambientes cerrados," Master's thesis, Universidad de Chile, Santiago Chile, 2009.
- [8] U. S. Government, "Space segment," <http://www.gps.gov/systems/gps/space/>, fecha de consulta: julio 2014.
- [9] U. Government, "Gps accuracy," <http://www.gps.gov/systems/gps/performance/accuracy/>, fecha de consulta: julio 2014.
- [10] T. Perry, "Indoor navigation takes signals and sensors," 2012.
- [11] B. Ozdenizci, K. Ok, V. Coskun, and M. N. Aydin, "Development of an indoor navigation system using nfc technology," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1991, pp. 11–14.
- [12] D. B. Johnson, 'A note on dijkstra's shortest path algorithm' *J. ACM*, vol. 20, no. 3, pp. 385-388, Jul. 1973. [Online]. Available: <http://doi.acm.org/10.1145/321765.321768>.
- [13] R. Ivanov, 'Indoor navigation system for visually impaired,' in *Proceedings of the 11th International Conference on Computer Systems and Technologies and Workshop for PhD Stu-*

- dents in Computing on International Conference on Computer Systems and Technologies, ser. CompSysTech '10. New York, NY, USA: ACM, 2010, pp. 143-149. [Online]. Available: <http://doi.acm.org/10.1145/1839379.1839405>.
- [14] L. Badman, 'Bluetooth 4.1 aims for internet of things,' *Network Computing*, Dec. 2013. [Online]. Available: <http://www.networkcomputing.com/bluetooth-41-aims-for-internet-of-things/a/d-id/1234594>.
- [15] Y. Wang, X. Yang, Y. Zhao, Y. Liu, and L. Cuthbert, 'Bluetooth positioning using rssi and triangulation methods,' in *Consumer Communications and Networking Conference (CCNC)*, 2013 IEEE, Jan 2013, pp. 837-842.
- [16] J. J. M. Diaz, R. de A. Maues, R. B. Soares, E. F. Nakamura, and C. M. S. Figueiredo, 'Bluepass: An indoor bluetooth-based localization system for mobile applications', in *Symposium on Computers and Communications (ISCC)*. IEEE, pp. 778-783.
- [17] Z. 0001, Q. Li, and G. Chen, 'Apt: Accurate outdoor pedestrian tracking with smartphones.' in *INFOCOM*. IEEE, pp. 2508-2516.
- [18] Blake, S., Galea, E., Westeng, H., and Dixon, A. (2004). 'An analysis of human behavior during the WTC disaster of 11 September 2001 based on published survivor accounts.' *Proc., 3rd Int. Symp. on Human Behavior in Fire*, Fire Safety Engineering Group, Univ. of Greenwich, 192-181.
- [19] Kuligowski, E., Peacock, R., Wiess, E., and Hoskins, B. (2013). 'Stair evacuation of older adults and people with mobility impairments.' *Fire Safety J.*, 62, 230-237.
- [20] Chen, T., Pan, L., Zhang, H., Narayanan, S., and Soldner, N. (2013). 'Experimental study of evacuation from a 4-story building.' *Procedia Eng.*, 62(1), 538-547.
- [21] Hashemi, M. and Karimi, H. (2015). 'Indoor Spatial Model and Accessibility Index for Emergency Evacuation of People with Disabilities.' *J. Comput. Civ. Eng.* , 10.1061/(ASCE)CP.1943-5487.0000534 , 04015056.
- [22] Proulx, G. (1995). 'Evacuation time and movement in apartment buildings.' *Fire Safety J.*, 24(3), 229-246.
- [23] Swobodzinski, M., and Raubal, M. (2009). 'An indoor routing algorithm for the blind: development and comparison to a routing algorithm for the sighted.' *Int. J. Geog. Inf. Sci.*, 23(10), 1315-1343.
- [24] Hangli Ge, Masahiro Bessho, Noboru Koshizuka, Ken Sakamura, 'Model Design of Generating Path with Accessibility Semantics for Assisting Indoor Mobility', *DSDIS, 2015*, 2015 IEEE International Conference on Data Science and Data Intensive Systems (DSDIS), 2015 IEEE International Conference on Data Science and Data Intensive Systems (DSDIS) 2015, pp. 139-146, doi:10.1109/DSDIS.2015.55.
- [25] M. A. Keller, P. Sao, and A. S.-K. Pang, 'Ubiquitous computing,' fecha de consulta: julio 2014. [Online]. Available: <http://www-sul.stanford.edu/weiser/Ubiqu.html>.

- [26] V. Coroama, 'The chatty environment - a world explorer for the visually impaired,' in Adjunct Proceedings of Ubicomp 2003, Seattle, Washington, Oct. 2003.
- [27] V. Coroama and F. Röthenbacher, 'The chatty environment,' in Demo at the Second Conference on Pervasive Computing and Communications (PerCom 2004), Orlando, Florida, Mar. 2004.
- [28] 38. M. Banâtre, P. Couderc, J. Pauty, and M. Becus, 'Ubibus: Ubiquitous computing to help blind people in public transport,' in Mobile Human-Computer Interaction - MobileHCI 2004, ser. Lecture Notes in Computer Science, S. Brewster and M. Dunlop, Eds. Springer Berlin Heidelberg, 2004, vol. 3160, pp. 310314.
- [29] J. Stewart, S. Bauman, M. Escobar, J. Hilden, K. Bihani, and M. W. Newman, 'Accessible contextual information for urban orientation,' in In Proc. UbiComp 2008. ACM Press (2008), pp. 332-335.13.
- [30] L. Chen, I. Hussain, R. Chen, W. Huang, and G. Chen, 'Blueview: A perception assistant system for the visually impaired,' in Proceedings of the 2013 ACM Conference.
- [31] Billi M, Burzagli L, Catarci T, et al. A Unified Methodology for the Evaluation of Accessibility and Usability of Mobile Applications. Universal Access in the Information Society. 2010;9:337e56.
- [32] G. C. J. N. J. S.-D. W. Carol Barnum, Nigel Bevan, "The "magic number 5": Is it enough for web testing?" pp. 698–699, 2003.
- [33] L. Faulkner, "Beyond the five-user assumption: Benefits of increased sample sizes in usability testing," pp. 379–383, 2003.
- [34] S. Soy, "The case study as a research method," <https://www.ischool.utexas.edu/~ssoy/users/users/l391d1b.htm>, fecha de consulta: junio 2017.
- [35] W. Tellis, "Introduction to case study," <http://nsuworks.nova.edu/tqr/vol3/iss2/4/>, fecha de consulta: junio 2017.
- [36] Boyce, K. E., Shields, T. J., and Silcock, G. W. (1999a). 'Toward the characterization of building occupancies for fire safety engineering: Capabilities of disabled people moving horizontally and on an incline.' *Fire Technol.*, 35(1), 51–67.
- [37] Christensen, K. M., Collins, S. D., Holt, J. M., and Phillips, C. N. (2006). 'The relationship between the design of the built environment and the ability to egress of individuals with disabilities.' *Rev. Disability Stud.*, 2(3), 24–34.
- [38] Jiang, C. S., Zheng, S. Z., Yuan, F., Jia, H. J., Zhan, Z. N., and Wang, J. J. (2012). 'Experimental assessment on the moving capabilities of mobility-impaired disabled.' *Safety Sci.*, 50(4), 974–985.
- [39] Shi, L., Xie, Q., Cheng, X., Chen, L., Zhou, Y., and Zhang, R. (2009). 'Developing a database for emergency evacuation model.' *Build. Environ.*, 44(8), 1724–1729.

- [40] Church, R. L., and Marston, J. R. (2003). 'Measuring accessibility for people with a disability.' *Geog. Anal.*, 35(1), 83–96.
- [41] Sakkas, N., and Perez, J. (2006). 'Elaborating metrics for the accessibility of buildings.' *Comput. Environ. Urban Syst.*, 30(5), 661–685.
- [42] Thill, J.-C., Dao, T. H., and Zhou, Y. (2011). 'Traveling in the threedimensional city: Applications in route planning, accessibility assessment, location analysis and beyond.' *J. Transp. Geog.*, 19(3), 405–421.
- [43] Abu Doush I, Pontelli E. Non-visual navigation of spreadsheet tables. In Proceedings of the 12th International Conference on Computers Helping People with Special Needs (ICCHP 2010), Vienna, Austria, July 14e16, 2010; 108e15.
- [44] Ahmed F. Assistive Web browsing with touch interfaces. In Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS'10), Orlando, FL, USA, October 25e27, 2010; 235e6.
- [45] Johnson KL, Dudgeon B, Amtmann D. Assistive technology in rehabilitation. *Phys Med Rehabil Clin N Am.* 1997;8:389e403.
- [46] Phillips B, Zhao H. Predictors of assistive technology abandonment. *Assist Technology.* 1993;5:36e45.
- [47] Scherer MJ, Lane JP. Assessing consumer profiles of 'ideal' assistive technologies in ten categories: an integration of quantitative and qualitative methods. *Disability Rehabilitation.* 1997;19:528e35.
- [48] OMS, "Ceguera y discapacidad visual," <http://www.who.int/mediacentre/factsheets/fs282/es/>, fecha de consulta: febrero 2017.

Anexos y apéndices

Anexos y apéndices A

Pauta Evaluación de Usabilidad

	Poco					Mucho				
	1	2	3	4	5	6	7	8	9	10
Me gusta la aplicación										
La aplicación es útil										
La aplicación me hace estar activo										
Volvería a usar esta aplicación										
Recomendaría esta aplicación a otros										
Aprendí con esta aplicación										
Me sentí en control de la situación										
La aplicación es interactiva										
La aplicación es fácil de utilizar										
La aplicación es motivadora										
La aplicación se adapta a mi ritmo										
La aplicación me permitió reconocer el espacio										
Me gustan los sonidos de la aplicación										
Los sonidos de la aplicación son claramente identificables										
Los sonidos de la aplicación me transmiten información										
Los mensajes me ayudan a comprender mi entorno										
Preguntas abiertas										
¿La aplicación es útil para ti?										
¿Qué es lo que más te gustó de la aplicación?										
¿Qué cree que le falta a la aplicación para mejorarla?										
¿Entendiste los mensajes o te confundieron?										
¿Te sentiste seguro durante el trayecto?										
¿Te ayudaron los mensajes a comprender tu entorno?										

Figura A.1: Pauta de evaluación de usabilidad basada en pauta de profesor Jaime Sanchez

Anexos y apéndices B

Tipos de mensajes emitidos

Tipo mensaje	Exp1	Exp2	Exp3	Exp4	Exp5	Exp6	Total
# Mensajes con información contextual errados	1	2	5	3	1	1	13
# Mensajes sin información contextual acertados	1	2	5	4	1	5	18
# Mensajes con información contextual acertados	20	26	25	20	24	26	141
# Mensajes con y sin información contextual fuera de recorrido	2	6	4	6	1	8	27
# Mensajes no emitidos	2	0	2	5	4	2	15

Tabla B.1: Mensajes contabilizados por tipo