

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

Desarrollo de una aplicación móvil para la plataforma Android que despliegue mapas accesibles para personas con discapacidad visual usando el formato SVG.

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS INFORMATICOS Y DE COMPUTACION

CALO ORTIZ DARIO JAVIER
dario.calo@epn.edu.ec

HIGUERA ALQUEDAN ANGEL JAVIER
angel.higuera@epn.edu.ec

DIRECTOR: PhD. CALLE JIMENEZ TANIA ELIZABETH
tania.calle@epn.edu.ec

Quito, 12 de marzo de 2020

DECLARACIÓN

Nosotros, Ángel Javier Higuera Alquedán y Darío Javier Calo Ortiz, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos los derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Ángel Javier Higuera Alquedán

Darío Javier Calo Ortiz

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Ángel Javier Higuera Alquedán y Darío Javier Calo Ortiz, bajo mi supervisión.

PhD. Tania Elizabeth Calle Jiménez

DIRECTOR DE PROYECTO

AGRADECIMIENTOS

Me gustaría agradecer en estas líneas a las personas me nos han colaborado durante el proceso de desarrollo y posterior redacción de este proyecto. En primer lugar, a mis padres por todo su amor, comprensión y apoyo, pero sobre todo gracias infinitas por la paciencia que me han tenido.

A mis amigos que supieron ayudarme y apoyarme cuando lo necesitaba.

A mi tutora, Tania Calle, quien ha guiado con su paciencia, y su rectitud como docente, en todos los momentos que necesité sus consejos.

Agradezco a los todos docentes que, con su sabiduría, conocimiento y apoyo, motivaron a desarrollarme como persona y profesional. A la Escuela Politécnica Nacional por ser la sede de todo el conocimiento adquirido en estos años.

Ángel Higuera

DEDICATORIA

Dedico esta tesis a todos aquellos que creyeron en mí, a aquellos que me ayudaron con la culminación de mis estudios, a todos aquellos que me solucionaron mis dudas, a todos los que me dijeron que lo lograría, a todos ellos les dedico esta tesis.

Ángel Higuera

AGRADECIMIENTOS

Quiero agradecer a mi esposa Jessica y mi hijo Martín, que con su ánimo, apoyo y paciencia permitieron que avanzara cada día en el desarrollo de este proyecto. Su motivación es esencial para culminar cada uno de mis proyectos.

A mis padres y hermanos, que con cada una de sus acciones me apoyaron en cada una de mis tareas profesionales y personales de una forma directa o indirecta.

A mi tutora, PhD Tania Calle, que con su guía me permitió avanzar en el desarrollo del proyecto y más que nada, a entender el mundo de las personas con discapacidad visual y sus necesidades especiales.

A mis amigos Juan, Erika, Ángel y Jonathan, que con su apoyo, guía y consejos supieron impulsarme en cada aspecto de la vida tanto profesional como personal.

Y a cada uno de los docentes de la FIS. Que siempre supieron guiarme en este proceso de aprendizaje.

Darío Calo

DEDICATORIA

Quiero dedicar este proyecto a mi esposa y mi hijo por darme la fuerza para seguir adelante cada día, a mis padres y hermanos que siempre me han guiado y apoyado. Y a mis amigos por todo su apoyo en cada aspecto de la vida.

Darío Calo

RESUMEN

En el Ecuador existe un gran número de personas con discapacidad visual a las cuales un mapa tradicional les brindaría poca información para que puedan orientarse dentro de un del lugar. En base a eso, el presente proyecto desarrolla una aplicación que presenta la información en línea de un mapa a personas no videntes de forma auditiva. Este proyecto también muestra el proceso de creación del servidor que aloja y procesa los mapas y su integración con las operaciones de la aplicación.

El proyecto hace uso del formato de imagen SVG en su versión Tiny, este formato permite que se almacene información textual dentro de la imagen lo cual permite proveer a sistemas de lectura de pantalla información adicional sobre el contenido de dicha imagen. La versión SVG Tiny es la más optimizada para dispositivos móviles. En este caso se construyó los mapas en este formato con el fin de proveer a una aplicación móvil la información necesaria para orientar a un usuario no vidente.

Las pautas de accesibilidad definidas en las WCAG 2.1 ayudan a las páginas a ser más compatibles con los lectores de pantalla. Estas pautas permitieron orientar las tecnologías y herramientas utilizadas en la creación del servidor, el diseño de los mapas y la creación de la aplicación hacia las necesidades de las personas con discapacidad visual.

Este proyecto se desarrolló con la metodología Agile utilizando el Framework SCRUM. Este tipo metodología permite un control y distribución adecuada del esfuerzo necesario para la creación de un producto. Mediante los artefactos y eventos del framework SCRUM se realizó un control de las características del producto que se desarrolló en este proyecto.

Para el desarrollo de las pruebas se obtuvo la ayuda de diversas personas no videntes las cuales dieron una respuesta favorable al comportamiento y uso de la aplicación. Adicionalmente se realizaron pruebas del sistema a través de páginas de evaluación de la accesibilidad, se realizaron también pruebas con conexiones múltiples para asegurar un mínimo de soporte de usuarios simultáneos las cuales dieron resultados satisfactorios.

Palabras clave: SVG, SVG Tiny, mapas, accesibilidad, aplicación móvil, WCAG.

SUMMARY

In Ecuador there is a large number of people with visual disabilities which a traditional map provides them less information, so these people can't orient themselves in one of the places of the map. Based on that, the present project develops an application that presents the information of a map to blind people in an auditive way. This project also shows the process of creating the server that hosts and processes the maps and their integration with the operations of the application.

The project makes use of the SVG image format in its Tiny version, this format allows textual information to be stored within the image which allows to provide screen reading systems additional information on the content of the image. The SVG Tiny version is the most optimized for mobile devices. In this case, the maps were constructed in this format in order to provide a mobile application with the necessary information to guide a blind user.

The accessibility guidelines defined in WCAG 2.1 help pages to be more compatible with screen readers. These guidelines allowed to guide the technologies and tools used in the creation of the server, the design of the maps and the creation of the application towards the needs of the visually impaired.

This project was developed using the Agile methodology using the SCRUM Framework. This type of methodology allows an adequate control and distribution of the effort necessary for the creation of a product. Through the artifacts and rituals of the SCRUM framework, a control of the characteristics of the product that was developed in this project was carried out.

For the development of the tests, the help of various blind people was obtained, which gave a favorable response to the behavior and use of the application. Additionally, system tests were carried out through accessibility assessment pages, tests with multiple connections were also performed to ensure a minimum of simultaneous user support which gave satisfactory results.

Keywords: SVG, SVG Tiny, maps, accessibility, mobile application, WCAG.

TABLA DE CONTENIDO

1. INTRODUCCIÓN.....	1
1.1. Antecedentes.....	1
1.2. Análisis del entorno	1
1.3. Trabajos Relacionados	5
1.4. Objetivos	6
1.4.1. Objetivo general	6
1.4.2. Objetivos específicos.....	6
1.4.3. Alcance.....	6
1.5. Motivación	7
2. Pautas de Accesibilidad para el Contenido Web (WCAG)	8
2.1. WCAG Versión 2.1	8
2.2. Secciones de la WCAG 2.1	8
2.2.1. Diferencias con la versión 2.0.....	9
2.2.2. Pautas consideradas de la WCAG.....	11
2.2.3. Pautas no consideradas.	15
2.3. Criterios de éxito.....	17
2.3.1. Criterios de éxito según las WCAG.....	17
2.3.2. Medidas consideradas para aplicar las pautas WCAG 2.1.....	17
3. Descripción del proceso.....	20
3.1. Diseño de los mapas	20
3.1.1. Scalable Vector Graphics (SVG).....	20
3.1.2. SVG Tiny	20
3.1.3. Proceso de creación de los mapas	20
3.2. Herramientas Tecnológicas.	22
3.3. Estructura de la aplicación móvil.....	23
3.3.1. La plataforma Android	23
3.3.2. Compatibilidad de dispositivos.....	23

3.3.3.	Integración con Android TTS.	24
3.4.	Flujo de trabajo de la aplicación.....	25
3.4.1.	Operaciones con las coordenadas.....	25
3.4.2.	Estructura del formato SVG.	26
3.4.3.	Proceso en la aplicación.....	27
4.	Metodología de desarrollo	28
4.1.	Selección de metodología.....	29
4.2.	Aplicación de la metodología al proyecto.....	30
4.2.1.	Marco de trabajo SCRUM.....	30
4.2.2.	Tratamiento de los roles de SCRUM	33
4.2.3.	Estimación de las historias y prioridades.	33
4.2.4.	Integration continua y Test-driven development (TDD).	36
5.	Desarrollo del sistema.	37
5.1.	Planeación del desarrollo.....	37
5.2.	Desarrollo de los Sprint	43
5.2.1.	Sprint 0: Configuración del entorno de desarrollo	43
5.2.2.	Sprint 1: Levantamiento de la aplicación.....	50
5.2.3.	Sprint 2: Creación de la aplicación móvil	60
5.2.4.	Sprint 3: Interacción de la aplicación	70
5.2.5.	Sprint 4: Funciones adicionales	77
5.3.	Funcionalidad de la aplicación.....	82
6.	Evaluación de la aplicación.....	86
6.1.	Planeación de las pruebas.....	86
6.2.	Pruebas de accesibilidad web	86
6.2.1.	WAVE.....	86
6.2.2.	Examinator	88
6.2.3.	PowerMapper - Accessibility Checker and Validator	88
6.3.	Pruebas de usuario.....	90
6.4.	Discusión de resultados.....	93

7. Conclusiones	94
8. Recomendaciones	96

FIGURAS

Figura 1. Cuadro estadístico de personas con discapacidad en el Ecuador.....	4
Figura 2. Creación de mapas usando QGis.	21
Figura 3. Vinculación de aplicación QGis con la Base de datos.....	21
Figura 4. Ventana de información del polígono actual.....	22
Figura 5. Flujo de trabajo de la aplicación.....	25
Figura 6: Estructura de la imagen SVG.....	26
Figura 7. Características del servidor alojado en Heroku.	44
Figura 8. Repositorio en GitHub del proyecto.	45
Figura 9. Estructura del proyecto.	45
Figura 10. Perfil para las pruebas unitarias en Travis.ci.....	46
Figura 11. Ejecución de las pruebas unitarias en Travis.ci.....	47
Figura 12. Configuración para integrar Heroku con Travis.ci.....	48
Figura 13: Tabla Burndown de la ejecución del sprint 0.	48
Figura 14. Diagrama relacional de la base de datos.	51
Figura 15. Archivos para la creación de tablas.....	52
Figura 16. Estructura del archivo para la creación de las tablas.	52
Figura 17. Estructura de los modelos.....	53
Figura 18. Lista de mapas.....	54
Figura 19. Creación de mapa utilizando QGIS.....	56
Figura 20. lectura de los parámetros de la URL.	57
Figura 21. Vista del mapa de pruebas.	57
Figura 22. Versión inicial de la aplicación.	58
Figura 23. Tabla Burndown de la ejecución del sprint 1.	59
Figura 24. Página del mapa vista desde la aplicación.....	61
Figura 25. Diagrama de relaciones de la base de datos del dispositivo.	62
Figura 26. Permisos de la aplicación.	63
Figura 27. Flujo de trabajo de la aplicación.....	64
Figura 28. Comprobación de coordenadas.	64
Figura 29. Cambio de color del lugar actual en el mapa.	65
Figura 30. Función para el armado del URL correcto en la aplicación.	66
Figura 31. Flujo de trabajo de la aplicación.....	66

Figura 32. Resaltado del lugar en la aplicación.....	67
Figura 33. Configuración motora TTS.	68
Figura 34. Tabla Burndown de la ejecución del sprint 2.	68
Figura 35. respuesta de la prueba de los lugares cercanos.	71
Figura 36. Metadata del SVG.....	71
Figura 37. Uso de los botones para la interacción de los lugares.	72
Figura 38. Cambio de lugar con las teclas de volumen.	73
Figura 39. Respuesta auditiva de la aplicación.	74
Figura 40. Cambio del resaltado del mapa mediante las teclas de volumen.	75
Figura 41. Tabla Burndown de la ejecución del sprint 3.	75
Figura 42. Vista del mapa vista desde el CEC.	78
Figura 43. Vista del mapa de los alrededores de Procodis.	79
Figura 44. Navegación en la pantalla de inicio.	80
Figura 45. Tabla Burndown de la ejecución del sprint 4.	81
Figura 46: Mapa inicial de la aplicación.....	82
Figura 47: Actualización del lugar	82
Figura 48: Lugar más cercano al usuario dentro del mapa	83
Figura 49: Lista de mapas	84
Figura 50: Selección del mapa.....	84
Figura 51: Lugar más cercano al usuario al iniciar el mapa.....	85
Figura 52: Lugar más cercano relativo.....	85
Figura 53. Prueba de la página de lista de mapas con WAVE.	87
Figura 54. Prueba de la vista del mapa con WAVE.....	87
Figura 55. Prueba de la vista del mapa con Examiner.	88
Figura 56: Resultados - Accessibility Checker and Validator.....	89
Figura 57. Prueba en varios dispositivos.....	90
Figura 58. Explicación del funcionamiento de la aplicación.....	104
Figura 59. Prueba en la Carolina con persona no vidente.....	104
Figura 60. Prueba en la Carolina con persona no vidente.....	105
Figura 61. Prueba en la Carolina con persona no vidente.....	105
Figura 62. Prueba en la Fundación Procodis con persona no vidente.....	106
Figura 63. Prueba en la UTI con usuario no vidente.	106

TABLAS

Tabla 1. Secciones WCAG 2.1	8
Tabla 2. nuevas pautas en la WCAG 2.1	10
Tabla 3. Pautas WCAG correspondientes al proyecto. Perceptible.....	12
Tabla 4. Pautas WCAG correspondientes al proyecto. Operabilidad.	14
Tabla 5. Pautas WCAG correspondientes al proyecto. Comprensibilidad.	14
Tabla 6. Pautas WCAG correspondientes al proyecto. Robustez.....	15
Tabla 7: Pautas WCAG 2.1 no consideradas en el proyecto.....	16
Tabla 8. Acciones del proyecto con respecto a las WCAG. Perceptibilidad.	18
Tabla 9. Acciones del proyecto con respecto a las WCAG. Operabilidad.....	19
Tabla 10. Acciones del proyecto con respecto a las WCAG. Comprensibilidad.	19
Tabla 11. Acciones del proyecto con respecto a las WCAG. Compatible.....	19
Tabla 12: Porcentaje de usuarios Android por versión [26, 19]	24
Tabla 13. Comparación metodologías ágiles y tradicionales.....	28
Tabla 14. Comparación de características de las metodologías.	29
Tabla 15. Etiquetas de las historias.	38
Tabla 16. Historia épica CNF01.....	38
Tabla 17. Historia épica SRV01.....	38
Tabla 18. Historia épica MAP01.....	39
Tabla 19. Historia épica APP01.	39
Tabla 20. Historias contenidas en la historia épica CNF01.	40
Tabla 21. Historias contenidas en la historia épica SRV01.	40
Tabla 22. Historias contenidas en la historia épica MAP01.....	41
Tabla 23. Historias contenidas en la historia épica APP01.....	42
Tabla 24. Distribución de las historias en los sprint.....	42
Tabla 25. Historias Sprint 0.....	42
Tabla 26. Sprint backlog del sprint 0.	43
Tabla 27. Descripción de la historia CNF02.	43
Tabla 28. Descripción de la historia CNF03.	44
Tabla 29. Descripción de la historia CNF04.....	46
Tabla 30. Descripción de la historia CNF06.....	47
Tabla 31: Estado del tablero Kanban Sprint 0.....	49
Tabla 32. Sprint backlog del sprint 1.	50
Tabla 33. Descripción de la historia CNF05.....	51
Tabla 34. Descripción de la historia SRV02.....	53

Tabla 35. Descripción de la historia SVR04.....	54
Tabla 36. Descripción de la historia MAP02.....	55
Tabla 37. Descripción de la historia SRV03.....	56
Tabla 38. Descripción de la historia APP02.....	58
Tabla 39. Estado del tablero Kanban Sprint 1.....	59
Tabla 40. Sprint backlog del sprint 2.....	60
Tabla 41. Descripción de la historia APP07.....	61
Tabla 42. Descripción de la historia APP05.....	62
Tabla 43. Descripción de la historia APP03.....	63
Tabla 44. Descripción de la historia SRV05.....	64
Tabla 45. Descripción de la historia APP04.....	66
Tabla 46. Descripción de la historia APP06.....	67
Tabla 47. Estado del tablero Kanban Sprint 2.....	69
Tabla 48. Sprint backlog del sprint 3.....	70
Tabla 49. Descripción de la historia SRV06.....	71
Tabla 50. Descripción de la historia APP08.....	72
Tabla 51. Descripción de la historia APP09.....	73
Tabla 52. Descripción de la historia SRV06.....	74
Tabla 53. Estado del tablero Kanban Sprint 3.....	76
Tabla 54. Sprint backlog del sprint 4.....	77
Tabla 55. Descripción de la historia SRV08.....	77
Tabla 56. Descripción de la historia MAP03.....	79
Tabla 57. Descripción de la historia APP10.....	80
Tabla 58. Estado del tablero Kanban Sprint 4.....	81
<i>Tabla 59. Encuesta de uso de aplicación a usuarios finales.....</i>	<i>92</i>
Tabla 60. Opiniones de los usuarios.....	93
Tabla 61: Acciones para las sugerencias de los usuarios.....	93

1. INTRODUCCIÓN

1.1. Antecedentes

Existe un problema de la inclusión social en distintas partes del mundo, siempre ha existido un estado de marginación o separación, especialmente de personas que tenían alguna discapacidad, sea de tipo psicológica, física o genética. Con el paso del tiempo, estas circunstancias que cruzaban las personas con discapacidad, fueron evolucionando a través de los aportes que la sociedad fue incluyendo en los sistemas y formas de convivencia [1].

Actualmente se ha estructurado cambios que el Estado como órgano regulador de los gobiernos ha implementado. La creación de nuevas políticas que favorezcan a personas con algún tipo de discapacidad busca que se ampare a personas que por muchos años estuvieron abandonados o únicamente protegidos y/o asistidos por sus familiares [2].

Analizar esta problemática desde una perspectiva general y/o tecnológico, nos permitiría contribuir a que sea más fácil incluir a las personas con discapacidad en nuestro medio, buscando la equidad, para que sean valorados dentro del marco social por lo que aportan. Con la finalidad de que las personas con discapacidad visual se sientan involucrados en la sociedad mediante el fortalecimiento de la independencia personal [2].

1.2. Análisis del entorno

El contenido web que existen hoy en día es muy variado en cuanto a contenido multimedia dinámico. Cualquier persona que posea una computadora o un teléfono celular con capacidades multimedia puede acceder o producir contenido de este tipo, como por ejemplo acceder a medios sociales, crear imágenes y editarlas, crear rutas cotidianas de acceso, etc. De la misma forma se puede acceder al contenido que otras personas publican. Un caso específico es el uso de mapas que, mediante aplicaciones de escritorio o móviles, permiten ubicarse en el espacio. Estos mapas son muy útiles al momento de realizar viajes o localizar un lugar en particular de forma remota. Conocer una dirección, orientarse en un barrio, son usos comunes de este tipo de contenido [3].

La mayoría de este contenido está generalizado. Es decir, no se considera a las personas con discapacidad (problemas de visión, motriz, audición, etc.) las cuales también son usuarios consumidores y productores de contenido multimedia dinámico. Que, para el caso particular, también tienen la necesidad de usar mapas que se ajusten a sus necesidades y capacidades [3].

El usuario ciego accede al contenido web de una manera distinta a como lo hacen el resto de usuarios. Mientras que las personas sin discapacidad exploran y toman decisiones en base a la organización visual del contenido, el ciego que accede mediante un lector de pantalla. Además, le cuesta mucho tiempo analizar la totalidad de la página con facilidad [4].

Los elementos visualmente atractivos que generalmente se usan para atraer la atención hacia los elementos más importantes de la página, no pueden ser destacados de forma similar por los intérpretes o asistentes digitales puesto que el contenido al que acceden las personas con discapacidades visuales es presentado de forma lineal y basado en su totalidad en texto [4].

Por lo tanto, el posicionamiento de los elementos en la página o el diseño gráfico, ni ayudan, ni dificultan de manera inherente el acceso de este colectivo a la información que contiene el sitio web. Es más importante el orden por programación, una sintaxis correcta, el uso de la semántica o la posibilidad de saltar bloques de información entre otros factores, los que realmente suponen una óptima experiencia de usuario para este colectivo [4].

Un usuario con discapacidad que accede mediante un lector de pantalla escucha en primer lugar el título de la página y, a continuación, el resto de los elementos presentes en el nodo según su orden de aparición en el código fuente [3].

A nivel mundial, se estima que aproximadamente 1300 millones de personas viven con alguna forma de deficiencia visual. Con respecto a una visión más general, 188,5 millones de personas tienen una deficiencia visual moderada, 217 millones tienen una deficiencia visual de moderada a grave y 36 millones son ciegas [5].

Gracias a la tecnología actual, las personas con pérdida de visión pueden cubrir estas necesidades además de que pueden realizar nuevas cosas, como escribir documentos, navegar por Internet y enviar y recibir correos electrónicos. El software de lectura de pantalla, los dispositivos especiales de conversación y braille, software de interpretación y conversión a audio permiten a quienes no tienen visión usar computadoras, teléfonos celulares y otros dispositivos electrónicos de forma independiente. Del mismo modo, las personas con baja visión pueden usar software y dispositivos de aumento de pantalla que les permitirán ver letras, imágenes y otros objetos sin tener que luchar o forzar su visión restante. Esta tecnología, comúnmente conocida como tecnología asistencial o adaptativa, evoluciona continuamente y ha eliminado muchas barreras de acceso para las personas con pérdida de visión.

Además, permiten llevar a cabo tareas de rutina en el trabajo y la escuela. La tecnología de asistencia también permite que las personas con discapacidad visual sean más independientes en el hogar. Ahora se puede leer el correo, escuchar audiolibros, obtener instrucciones paso a paso para caminar a lugares desconocidos, registrar información importante y mucho más con dispositivos independientes especiales diseñados para personas sin visión o con baja visión [6].

En el área de soluciones informáticas para personas con discapacidad visual, se ven instrumentos que poseen un teclado para introducir la información con signos braille y la salida de los datos se da en braille o voz sintetizada [3].

Regulaciones en el Ecuador

En el Ecuador la organización encargada de la regularización e inclusión de las personas con discapacidad es el Consejo Nacional para la Igualdad de Discapacidades (CONADIS). Este organismo tiene como objetivo el observar, realizar el seguimiento y la evaluación de las políticas públicas en materia de discapacidades, en todo el territorio nacional, en todos los niveles de gobiernos y en los ámbitos público y privado; con el fin de asegurar la plena vigencia y el ejercicio de los derechos de las personas con discapacidad y sus familias; promoviendo, impulsando, protegiendo y garantizando el respeto al derecho de igualdad y no discriminación [7].

Este organismo cuenta con una iniciativa conjunta que busca promover el cumplimiento de la Normativa NTE INEN-ISO/IEC 40500 de accesibilidad para el contenido web, con el fin de apoyar la inclusión y la igualdad de oportunidades para las personas con discapacidades en el acceso a la información, principalmente en portales web del Estado Ecuatoriano [8].

Dicha normativa para los medios digitales que son consumidos por las personas con discapacidad visual está basada en las normas internacionales WCAG 2.0 [9]. Este documento agrupa una gran cantidad de pautas que cubren un amplio rango de recomendaciones para crear contenido Web más accesible.

TOTAL DE PERSONAS CON DISCAPACIDAD REGISTRADAS EN EL REGISTRO NACIONAL DE DISCAPACIDAD

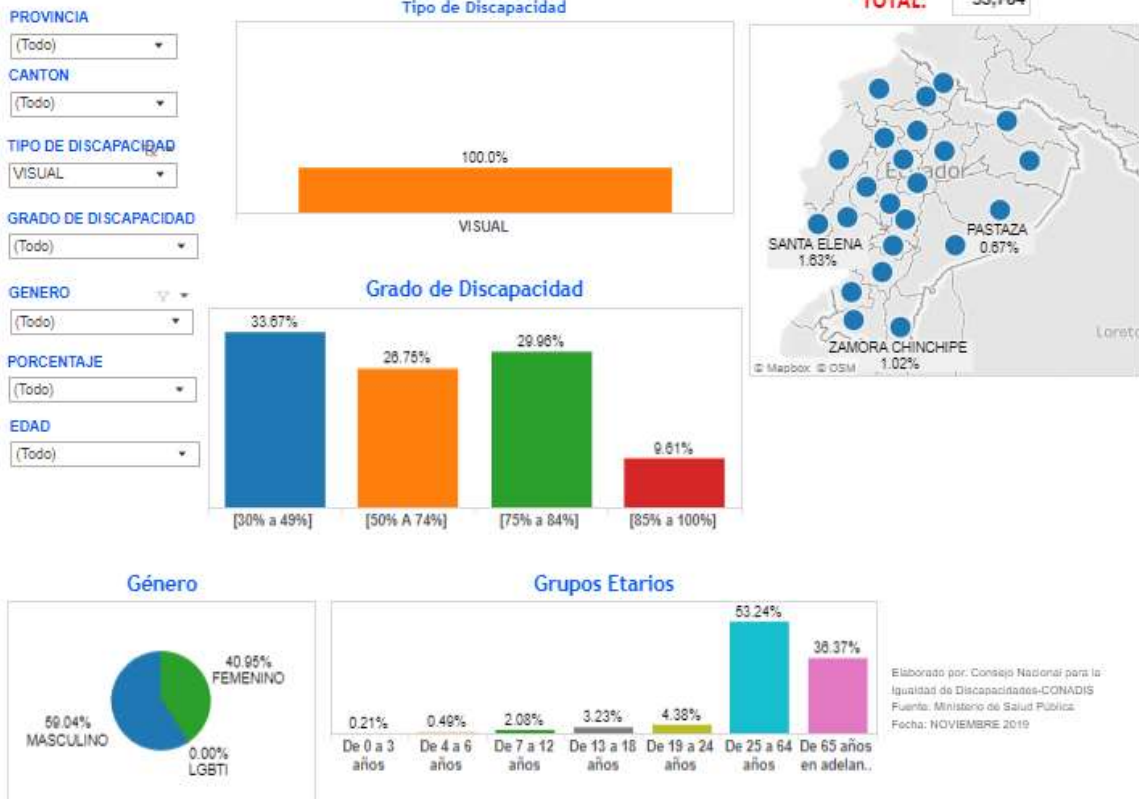


Figura 1. Cuadro estadístico de personas con discapacidad en el Ecuador.

En el Ecuador existe más de 52000 personas con discapacidad visual, que representa un 11.88% de las personas con un tipo de discapacidad [10]. En la Figura 1 se muestra un gráfico provisto por la página de la CONADIS que muestra la cantidad de personas con discapacidad visual la cual llega a alrededor de 55000 personas. Esta cantidad de personas da un motivo importante para incluir a estas personas en el mundo tecnológico, ya que presentan las mismas necesidades de movilidad, adicionalmente pueden aportar al contenido web.

1.3. Trabajos Relacionados

El formato SVG se ha utilizado para contenido web desde hace algún tiempo para la representación de imágenes escalables. Este uso es analizado en [11] y se concluye que este formato es ideal para aplicaciones multiplataforma debido a que las imágenes no sufren pudelación por operaciones de escalado. El uso para la representación de mapas móviles mediante SVG se puede ver en el trabajo de Wu Binzhao. En [12], él desarrolla una aplicación para representación de información GIS en dispositivos móviles basados en J2ME usando el formato SVG. Feng Lin posteriormente en [13] se basa en el trabajo de Binzhao para mejorar la interfaz para adaptarse a diferentes tipos de pantallas e incorporar otros tipos de procesos. Si bien el uso del SVG se recoge en varios trabajos este no se enfocaba en la accesibilidad.

Una aplicación diferente de las imágenes SVG es propuesta en [14] por Christin Engel la cual a partir de datos crea una imagen SVG la cual va a pasar a imprimirse para que se pueda distribuir a personas con discapacidad visual. Christin creó una interfaz en JavaFx que permite a los usuarios con discapacidad visual el crear dichas tablas y gráficos para luego procesarlas. Si bien la impresión de las tablas fue exitosa se tubo problema con la interacción con la interfaz de usuario para la introducción y diseño de los datos. En un estudio posterior [15] se explora diferentes tipos de mapas interactivos tanto físicos y digitales con el fin de ver la eficiencia de los mapas.

Tania Calle junto con Sergio Lujan en [16] analizan como los lectores de pantalla reconocen las imágenes del formato SVG y la información que se puede incluir dentro de las imágenes en dicho formato. Esto se lleva a cabo mediante la construcción de una página HTML que contiene la imagen. Adicionalmente mediante el análisis de los mapas con las pautas de accesibilidad WCAG 2.0 se logra que el contenido HTML se adapte de mejor manera a los lectores de pantalla. Posteriormente Tania Calle junto con Adrián Egúez en el prototipo propuesto en [17] logra un mapa accesible de un sistema de metro. Este sistema se basa en tres componentes: el componente de base de datos es responsable de almacenar los mapas web, el componente de servidor web accesible es responsable de convertir un mapa web en un mapa web accesible, y el componente del lado del usuario es responsable de presentar la web accesible mapa para el usuario.

Posteriormente Sergio Armero en [18] crea una aplicación web para proveer un mapa de la Universidad de Alicante. Este mapa se basa en OpenStreetMap para obtener la los edificios para componer la imagen SVG. Este sistema se basa en operaciones con PostGIS para ubicar al usuario y proveer la información de los lugares cercanos de la ubicación

usuario. Adicionalmente hace un análisis basándose en las pautas WCAG 2.0 para comprobar el nivel de accesibilidad de la página.

En base a estos estudios relacionados el presente trabajo tiene como objetivo llevar mapas accesibles a la plataforma Android basándose en las pautas de accesibilidad WCAG 2.1.

1.4. Objetivos

1.4.1. Objetivo general

Desarrollar una aplicación móvil para la plataforma Android que despliegue mapas accesibles para personas ciegas usando el formato SVG.

1.4.2. Objetivos específicos

Los objetivos específicos para este proyecto se muestran a continuación:

- Levantamiento de requerimientos de los usuarios para el desarrollo de la aplicación.
- Desarrollar una plataforma que provea un mapa de las inmediaciones de la Fundación Procodis, que contenga la información geográfica necesaria para desplegarla en la aplicación móvil.
- Incorporar criterios de accesibilidad web usando la metadata del formato SVG Tiny en la plataforma.
- Desarrollar una aplicación móvil que despliegue la información de la plataforma y ubique al usuario en el mapa.
- Desarrollar y ejecutar pruebas con los usuarios provistos por Procodis para la recolección de información crucial para el mejoramiento de la aplicación durante su fase de desarrollo.
- Proponer recomendaciones de uso del formato SVG Tiny para futuras aplicaciones móviles accesibles que puedan ser desarrolladas.

1.4.3. Alcance

El proyecto busca la creación de una aplicación móvil que permita indicar las cercanías de un sitio con respecto a la ubicación del usuario en un mapa específico. Esto mediante el uso de información dentro de imágenes SVG.

1.5. Motivación

Existen muchos dispositivos que permiten la ayuda al humano, facilitan ciertas actividades cotidianas. Pero no se debería limitar a la ayuda común, deben orientarse a cada una de las personas, independientemente de sus habilidades, destrezas y limitaciones. Si observamos de un sistema comunitario, de igualdad e inclusión es necesario que la tecnología se adapte a las necesidades de cada una de las personas y no solo a las necesidades de la sociedad promedio. Por tanto, esta tesis busca incluir en el mundo tecnológico a personas con discapacidad visual, explotando otros de sus sentidos y habilidades. Esto, aprovechando el avance de la tecnología. El contenido web generado mediante SVG nos facilita cumplir los objetivos planteados. Este formato nos permite diseñar y moldear el contenido de forma fácil. A fin de proveer un software más inclusivo.

2. Pautas de Accesibilidad para el Contenido Web (WCAG)

Las Guías de contenido web accesible (Web Content Accessibility Guidelines WCAG) cubre un amplio rango de recomendaciones para hacer contenido web más accesible [19]. Estas pautas abordan la accesibilidad del contenido web en computadoras de escritorio, computadoras portátiles, tabletas y dispositivos móviles.

Seguir estas pautas hará que el contenido sea más accesible para una gama más amplia de personas con discapacidades, incluidas las adaptaciones para ceguera y baja visión, sordera y pérdida de audición, etc. Seguir estas pautas también hará que el contenido web sea más usable para los usuarios en general [19].

2.1. WCAG Versión 2.1

WCAG 2.1 se inició con el objetivo de mejorar la guía de accesibilidad para tres grupos principales: usuarios con discapacidades cognitivas o de aprendizaje, usuarios con baja visión y usuarios con discapacidades en dispositivos móviles. Los requisitos estructurales heredados de WCAG 2.0, la claridad y el impacto de las propuestas y el cronograma llevaron al conjunto final de criterios de éxito incluidos en esta versión. El Grupo de trabajo considera que WCAG 2.1 avanza de manera incremental en la guía de accesibilidad del contenido web para todas estas áreas, pero subraya que estas pautas no satisfacen todas las necesidades de los usuarios [19].

2.2. Secciones de la WCAG 2.1

La Tabla 1 muestra como la guía divide sus pautas en 4 secciones [19]:

Perceptible	La información y los componentes de la interfaz de usuario deben ser presentables a los usuarios de manera que puedan percibirlos.
Operable	Los componentes de la interfaz de usuario y la navegación deben ser operables.
Comprensible	La información y el funcionamiento de la interfaz de usuario deben ser comprensibles.
Robusto	El contenido debe ser lo suficientemente robusto como para ser interpretado por una amplia variedad de agentes de usuario, incluidas las tecnologías de asistencia.

Tabla 1. Secciones WCAG 2.1

Este proyecto maneja imágenes SVG, audio generado y texto. En el caso del audio generado sólo es accesible desde la aplicación móvil, la cual va a dictar los datos requeridos por el usuario. Con el fin de cumplir con las recomendaciones descritas en la WCAG 2.1 se analiza cada una de las secciones de la guía con respecto a cómo el proyecto está implementado.

2.2.1. Diferencias con la versión 2.0

WCAG 2.1 extiende la WCAG 2.0 añadiendo nuevos criterios de aceptación, definiciones que lo soportan. Los criterios de aceptación presentes en la Tabla 2 son incluidos en la versión WCAG 2.1 [20].

Sección	Título	Descripción
1.3.4	Orientation	El contenido no restringe su vista y funcionamiento a una sola orientación de visualización, como vertical u horizontal, a menos que una orientación de visualización específica sea esencial.
1.3.5	Identify Input Purpose	El propósito de cada campo de entrada que recopila información sobre el usuario puede determinarse mediante programación.
1.3.6	Identify Purpose	En el contenido implementado usando lenguajes de marcado, el propósito de los Componentes de interfaz de usuario, iconos y regiones se puede determinar mediante programación.
1.4.10	Reflow	El contenido puede presentarse sin pérdida de información o funcionalidad, y sin necesidad de desplazarse en dos dimensiones.
1.4.11	Non-Text Contrast	La presentación visual de lo siguiente tiene una relación de contraste de al menos 3: 1 contra los colores adyacentes.
1.4.12	Text Spacing	En el contenido implementado utilizando lenguajes de marcado que tienen estilos como alto de línea, espaciado, espaciado entre letras espaciado de palabras no deben presentar pérdida de contenido o funcionalidad al cambiar una propiedad de estilo

1.4.13	Content on Hover or Focus	Cuando un puntero del mouse o el foco del teclado activa contenido adicional o lo hace visible este debe ser descartable, seleccionable y persistente.
2.1.4	Character Key Shortcuts	Si se implementa un atajo de teclado que contenga solo letras o números este debe ser deshabilitado, remapeable y activo solo cuando sea señalado.
2.2.6	Timeouts	Se advierte a los usuarios sobre la duración de cualquier inactividad del usuario que pueda causar la pérdida de datos, a menos que los datos se conserven durante más de 20 horas cuando el usuario no tome ninguna medida.
2.3.3	Animation from Interactions	La animación de movimiento activada por la interacción se puede deshabilitar
2.5.1	Pointer Gestures	Toda la funcionalidad que utiliza gestos multipunto se puede operar con un solo puntero
2.5.2	Pointer Cancellation	Las acciones de un solo puntero deben ser: no inmediatas, cancelables, reversibles.
2.5.3	Label in Name	Para los componentes de la interfaz de usuario con etiquetas que incluyen texto o imágenes de texto, el nombre contiene el texto que se presenta visualmente.
2.5.4	Motion Actuation	La funcionalidad que puede ser operada por el movimiento del dispositivo o el movimiento del usuario también puede ser operada por los componentes de la interfaz del usuario y la respuesta al movimiento se puede deshabilitar para evitar la activación accidental.
2.5.5	Target Size	El tamaño del objeto para las entradas de puntero es de al menos 44 por 44 píxeles CSS.
2.5.6	Concurrent Input Mechanisms	El contenido web no restringe el uso de modalidades de entrada disponibles en una plataforma, excepto cuando la restricción es esencial
4.1.3	Status Messages	En el contenido implementado usando lenguajes de marcado, los mensajes de estado se pueden determinar mediante programación.

Tabla 2. nuevas pautas en la WCAG 2.1

Muchos de estos criterios de aceptación son incluidos en la lista de las normativas requeridas para un completo cumplimiento de la norma.

En las tablas a continuación podemos ver las pautas de accesibilidad que el proyecto que corresponden a las características de la ampliación que se desarrolla en este proyecto.

2.2.2. Pautas consideradas de la WCAG.

En este proyecto no puede implementar todas las pautas de accesibilidad debido a que la aplicación o bien no tiene elementos a los cuales afecta o el elemento es indispensable tal cual está definido.

Pautas de Perceptibilidad (Sección 1 WCAG 2.1)

En las Tabla 3 se puede ver los criterios más relevantes que se toman en cuenta en este proyecto con respecto a la sección 1 de la WCAG 2.1.

Textos Alternativos	1.1.1. Cada una de las imágenes debe tener un texto alternativo adecuado, el cual debe contener una idea exacta de la imagen.
	1.2.2. Cada archivo de audio debe tener subtítulos adecuados de acuerdo que permita visualizar el texto que se está escuchando.
	1.2.4. Los mensajes de audio tienen un correspondiente subtítulo.
Adaptable	1.3.1. Toda la estructura debe ser fácilmente accesible por tecnologías de asistencia.
	1.3.2. Los elementos siguen un orden significativo.
	1.3.4. No limitar la orientación de la pantalla a menos que sea necesario.
	1.3.6. Todos los elementos deben ser etiquetados correctamente y fácilmente diferenciables del su entorno.
Distinguible	1.4.1. Los avisos deben tener un color consistente con la información que está presentado.
	1.4.2. Los elementos que contengan audio deben ser fácilmente controlables.
	1.4.4. El tamaño del texto puede ser modificado por tecnologías de asistencia.
	1.4.6. El contraste manejado en la página para las secciones de texto debe ser de al menos 7:1

	1.4.7. Los sonidos de ambiente de la página deben ser fácilmente desactivarles y no deben sobrepasar un nivel 20db menor que los archivos de audios principales.
	1.4.8. La presentación de texto debe ser personalizable: <ul style="list-style-type: none"> • Los colores del texto deben ser fácilmente seleccionables • Las líneas no deben tener más de 80 caracteres por línea • El texto no debe estar justificado • Se debe tener un espaciado de al menos espacio y medio • El texto debe ser redimensionado sin la necesidad de usar una tecnología de asistencia.
	1.4.9. No se deben agregar imágenes con texto fijo sin añadir una etiqueta que contenga toda la información del texto de la imagen
	1.4.11. El contraste manejado para los elementos de la interfaz gráfica debe ser de al menos 3:1.
	1.4.12. Con respecto a el formato del texto <ul style="list-style-type: none"> • El espaciado de línea debe ser al menos 1.5 veces el tamaño de la fuente • El espacio entre párrafos debe ser 2 veces el tamaño de la fuente • El espacio entre letras debe ser al menos 0.12 veces el tamaño de la fuente • El espacio entre palabras debe ser de al menos 0.16 veces el tamaño de la fuente.

Tabla 3. Pautas WCAG correspondientes al proyecto. Perceptible.

Pautas de Operatividad (Sección 2 WCAG 2.1)

En las Tabla 4 se puede ver los criterios más relevantes que se toman en cuenta en este proyecto con respecto a la sección 2 de la WCAG 2.1.

Operaciones con teclado	2.1.1. Todas las funciones del sistema deben ser fácilmente accesibles desde un teclado.
	2.1.2. Las operaciones del sistema se pueden acceder fácilmente desde un teclado sin la necesidad de usar tiempos específicos o combinaciones de teclas complejos.
	2.1.3. Todos los elementos funcionales deben poder accederse desde el teclado.

	2.1.4. Los atajos de teclado o combinaciones de teclas deben ser: fácilmente desactivables, pueden ser remapeados, debe tener un objeto en específico con el cual debe interactuar.
Tiempo suficiente	2.2.1. Para acciones automáticas de scroll se debe tener un tiempo de espera antes de empezar la operación.
	2.2.3. Las operaciones de actualización que se hacen en segundo plano no deben afectar a la navegación del contenido actual.
	2.2.4. Las acciones pueden ser canceladas por el usuario.
	2.2.5. Las operaciones de algún usuario deben tener una persistencia de datos de al menos 20 horas para que el usuario pueda retomar las operaciones que quedaron pendientes.
Reacciones físicas	2.3.1. El contenido debe tener menos de 3 flashes en caso de necesitar de alguno.
	2.3.2. Las animaciones de elementos del sistema deben ser fácilmente desactivables sin afectar a la información contenida en ellos.
Navegable	2.4.1. Bloques de textos repetitivos pueden ser saltados en la navegación.
	2.4.2. Todas las páginas tienen un título que sea descriptivo al propósito de su contenido.
	2.4.3. La navegación no debe afectar al entendimiento del sistema.
	2.4.4. Los links deben tener una clara descripción de hacia dónde dirigen o que acción hacen.
	2.4.6. Los encabezados deben ser descriptivos.
	2.4.7. Las operaciones a través de teclado son resaltadas en la página.
	2.4.9. Se debe evitar etiquetas ambiguas en los links, con el fin de que estos sean reconocidos por una tecnología de asistencia o por el usuario.
	2.4.10. Las cabeceras de cada sección describen adecuadamente su contenido.

Modalidades de ingreso	2.5.1. Todas las operaciones multipunto deben ser también realizables con operaciones de un solo punto.
	2.5.2. Los eventos táctiles deben ser controlados para que no sean activados por error.
	2.5.3. Las imágenes deben contener su etiqueta al inicio de su nombre.
	2.5.4. Los eventos de scrolling pueden ser accedidos tanto por las acciones del hardware como por la interfaz del sistema.

Tabla 4. Pautas WCAG correspondientes al proyecto. Operabilidad.

Pautas de Comprensibilidad (Sección 3 WCAG 2.1)

En las Tabla 5 se puede ver los criterios más relevantes que se toman en cuenta en este proyecto con respecto a la sección 3 de la WCAG 2.1.

Leíble	3.1.1. La información y las operaciones realizables deben ser entendibles.
	3.1.2. El idioma en todo el contenido de la página debe ser programáticamente determinar.
	3.1.3. Se debe proveer una forma de identificar abreviaciones palabras regionales (jergas).
Predecible	3.2.1. Cuando un usuario señala un objeto este no debe cambiar su comportamiento de forma inesperada.
	3.2.2. En el caso de necesitar una entrada de información esta debe ser consistente en todo el ciclo de una página. Es decir, no debe cambiar su contexto durante la introducción de la información.
	3.2.3. La página debe tener una forma de navegación consistente.
	3.2.4. La página debe ser consistente con respecto a la funcionalidad de elementos similares.

Tabla 5. Pautas WCAG correspondientes al proyecto. Comprensibilidad.

Pautas de Robustez (Sección 4 WCAG 2.1)

En la Tabla 6 se puede ver los criterios más relevantes que se toman en cuenta en este proyecto con respecto a la sección 4 de la WCAG 2.1.

Compatible	4.1.1. La información presentada en la página debe ser parseable, para esto se debe tener un buen diseño con respecto a la estructura del etiquetado.
	4.1.2. Los elementos no deben tener atributos duplicados y cualquier identificador tiene que ser único.
	4.1.3. Los mensajes de estado deben ser programáticamente determinados y deben por ser reconocidos por tecnologías de asistencia.

Tabla 6. Pautas WCAG correspondientes al proyecto. Robustez.

2.2.3. Pautas no consideradas.

Debido a que la aplicación no contiene algunos de los elementos multimedia que se menciona en la WCAG 2.1 estos se excluyen del análisis. Adicionalmente de otros aspectos que alterarían el uso de la aplicación. La Tabla 7 describe las razones por las cuales no se toma en cuenta todas las pautas de la WCAG 2.1.

Pauta	Razón de exclusión
Perceptibilidad (Sección 1 WCAG 2.1)	
1.2.1 Audio-only and Video-only (Prerecorded)	La aplicación no cuenta con archivos de audio pregrabado. Debido a que toda la información en audio es generada por el motor de texto a voz de Android.
1.2.2 Captions (Prerecorded)	
1.2.3 Audio Description or Media Alternative (Prerecorded)	
1.2.5 Audio Description (Prerecorded)	
1.2.6 Sign Language (Prerecorded)	
1.2.7 Extended Audio Description (Prerecorded)	
1.2.8 Media Alternative (Prerecorded)	
1.3.3 Sensory Characteristics	
1.3.4 Orientation	La aplicación va a estar fijada a solo una orientación debido a que se necesita

	visualizar todos los elementos en la pantalla.
1.3.5 Identify Input Purpose	La aplicación no cuenta con campos para la introducción de texto.
Operatividad (Sección 2 WCAG 2.1)	
2.2.5 Re-authenticating	La aplicación no cuenta con un sistema de autenticación.
2.2.6 Timeouts	
2.3.1 Three Flashes or Below Threshold	La aplicación no cuenta con flashes cuando se hace alguna transición.
2.3.2 Three Flashes	
2.3.3 Animation from Interactions	La aplicación no cuenta con animaciones.
2.4.5 Multiple Ways	La aplicación no cuenta con entradas de múltiples
Comprensibilidad (Sección 3 WCAG 2.1)	
3.1.4 Abbreviations	La aplicación no tiene abreviaciones o nombre complejos en las etiquetas.
3.1.5 Reading Level	
3.1.6 Pronunciation	La aplicación solo tiene una salida de audio generada automáticamente por lo tanto está exenta de acento.
3.2.5 Change on Request	La aplicación se basa en interacciones con la locación del dispositivo por lo tanto la interacción no es hecha por completo por el usuario.
3.3.1 Error Identification	Los errores se comunican al usuario por medio de mensajes de voz.
3.3.2 Labels or Instructions	Las instrucciones de la aplicación son dadas antes de iniciar su uso.
3.3.3 Error Suggestion	Los errores se comunican al usuario por medio de mensajes de voz. La aplicación no maneja datos sensibles del usuario.
3.3.4 Error Prevention (Legal, Financial, Data)	
3.3.5 Help	Las instrucciones de la aplicación son dadas antes de iniciar su uso.
3.3.6 Error Prevention (All)	Los errores se comunican al usuario por medio de mensajes de voz.

Tabla 7: Pautas WCAG 2.1 no consideradas en el proyecto.

2.3. Criterios de éxito

2.3.1. Criterios de éxito según las WCAG

Según la sección 5 de las WCAG nos indica como ver interpretar el cumplimiento de las guías y como se debe calificar la totalidad del cumplimiento:

- Para la conformidad con el Nivel A (el nivel mínimo de conformidad), la página web cumple con todos los Criterios de éxito del Nivel A, o proporcionar una versión alternativa aceptable.
- Para la conformidad con el Nivel AA, la página web cumple con todos los Criterios de éxito del Nivel A, AA, o proporcionar una versión alternativa aceptable.
- Para la conformidad con el Nivel AAA, la página web cumple con todos los Criterios de éxito del Nivel A, AA, AAA, o proporcionar una versión alternativa aceptable.

2.3.2. Medidas consideradas para aplicar las pautas WCAG 2.1.

Para cumplir con los Criterios de éxito de se han tomado las medidas descritas en la Tabla 8, Tabla 9, Tabla 10 y Tabla 11. El proceso de implementación de las medidas se lleva a cabo en la sección de ejecución de las historias de usuario.

Sección	Perceptibilidad (Sección 1 WCAG)
Medida	Criterios de la que abarca
Las imágenes tienen un texto alternativo. La imagen a utilizar es un SVG la cual tiene embebida la información de cada una de las secciones en la imagen.	1.1 Text Alternatives 1.1.1 Non-text Content
Los audios que se mandan a TTS se muestran en pantalla también.	1.2.3 Audio Description or Media Alternative (Prerecorded) 1.2.4 Captions (Live)
Definir una estructura secuencial para los elementos con tal de facilitar el acceso. El mapa va a listar los lugares cercanos comenzando desde el más cercano	1.3.2 Meaningful Sequence
La orientación de pantalla debe ser limitada a una orientación vertical para no afectar la experiencia del usuario. La rotación de la aplicación está fijada en vertical mediante	1.3.4 Orientation

los parámetros definidos en el “manifest” de la App Android.	
Los colores para la interfaz de usuario son blanco y negro. Los el mapa está definido solo como contornos para evitar la malinterpretación de color.	1.4.1 Use of Color
Las descripciones de las locaciones van a ser cortas.	1.3.6 Identify Purpose 1.4.8 Visual Presentation 1.4.12 Text Spacing
La fuente por defecto va a ser Arial 12pt con espaciado simple a 1.5	1.4.3 Contrast (Minimum) 1.4.4 Resize text
Todas las locaciones en el mapa no van a tener un texto dentro de ellas, en cambio va a tener un recuadro en la parte superior que contenga dicha información.	1.4.9 Images of Text (No Exception) 1.4.13 Content on Hover or Focus

Tabla 8. Acciones del proyecto con respecto a las WCAG. Perceptibilidad.

Sección	Operatividad (WCAG Sección #2)
Medida	Criterios de éxito que abarca
Los la selección de los mapas va a poder hacerse desde el teclado. De encontrarse dentro de un mapa se puede navegar por los lugares cercanos mediante las teclas de volumen.	2.1.3 Keyboard (No Exception) 2.1.4 Character Key Shortcuts
Se va a ubicar en el mapa correspondiente al momento de iniciar la ubicación.	2.2.3 No Timing 2.2.4 Interruptions
El sistema no va a manejar persistencia de sesiones.	2.2.5 Re-authenticating
Los mapas no van a cambiar inesperadamente cuando se encuentre dentro de ellos.	2.2.6 Timeouts 2.3.1 Three Flashes or Below Threshold
Las páginas están etiquetadas adecuadamente en las cabeceras.	2.4.2 Page Titled 2.4.3 Focus Order
Los links tienen el nombre de la sección en su texto.	2.4.4 Link Purpose (In Context)

El mapa puede hacerse zoom con operaciones de doble-tap.	2.5.1 Pointer Gestures
No hay un scroll automático al iniciar.	2.5.4 Motion Actuation

Tabla 9. Acciones del proyecto con respecto a las WCAG. Operabilidad.

Sección	Comprensibilidad (WCAG Sección #3)	
Medida	Criterios de éxito que abarca	
La aplicación solo va a estar en español.	3.1.1 Language of Page	
No se van a utilizar abreviaciones debido a que los textos van a ser utilizados por un TTS.	3.1.4 Abbreviations 3.1.6 Pronunciation	
Los mapas solo cambian cuando se cambia de locación.	3.2 Predictable	
Se tiene opciones de retroceder a la lista de mapas en la vista web.	3.2.3 Consistent Navigation 3.2.4 Consistent Identification	

Tabla 10. Acciones del proyecto con respecto a las WCAG. Comprensibilidad.

Sección	Compatible (WCAG Sección #4)	
Medida	Criterios de éxito que abarca	
Todas las etiquetas son diferenciables por lo cual son fácilmente identificables para un motor TTS con analizador de páginas.	4.1.1 Parsing	
El contenido de los mensajes está definido en una parte del HTML.	4.1.3 Status Messages	

Tabla 11. Acciones del proyecto con respecto a las WCAG. Compatible.

3. Descripción del proceso.

En este capítulo se va a detallar las herramientas que se va a usar para la creación tanto de los mapas, el servidor web y la aplicación. Es necesario comprender cada una de las herramientas que se va a utilizar antes de empezar con la creación de las historias de usuario.

3.1. Diseño de los mapas

Uno de los puntos centrales de la creación del sistema en este proyecto es mostrar mapas simples que permitan introducir información dentro de ellos. Para lograr esto se va a usar las imágenes de tipos SVG.

3.1.1. Scalable Vector Graphics (SVG)

SVG es un formato para describir gráficos 2D mediante XML. Este formato puede contener formas de vectores gráficos, imágenes y textos. Los objetos gráficos pueden ser agrupados y transformados en objetos previamente renderizados. Una imagen en formato SVG puede ser escalada sin perder su calidad ni deformarse, además soporta la aplicación de animaciones [21].

La importancia del formato SVG en la accesibilidad web es debido a que este puede almacenar información adicional dentro de la propia imagen, información la cual puede ser indexable. El manejo de la información dentro de una imagen SVG se la puede hacer con diferentes librerías [21].

3.1.2. SVG Tiny

SVG 1.2 Tiny es un perfil de SVG diseñado para su implementación en dispositivos móviles y PDA, este también puede ser usado en computadoras portátiles y de escritorio, y por lo tanto este incluye un subconjunto de las características incluidas en SVG 1.1 Full, junto con nuevas características para extender las capacidades del SVG [22].

3.1.3. Proceso de creación de los mapas

Para el diseño de los mapas se utilizó QGIS, que es una aplicación profesional de SIG que está construida sobre software libre y de código abierto (FOSS). En la Figura 2 se muestra la interfaz de QGIS y como esta interactúa con el plugin de Google Maps que nos ofrece información general de los lugares que queremos transformar en imagen SVG.

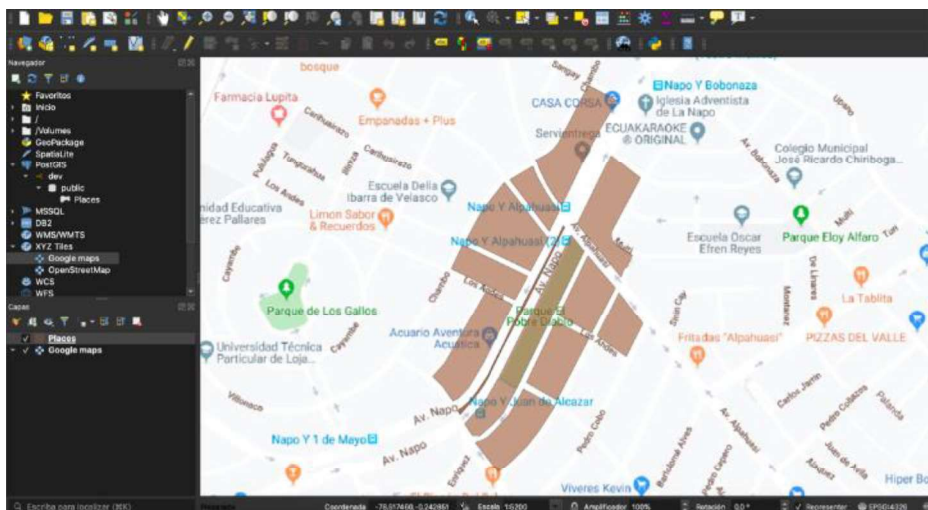


Figura 2. Creación de mapas usando QGis.

Primero, como se muestra en la Figura 3, se vincula QGis con una base de datos relacional, que para el caso empleamos PostgreSQL y su plugin PostGIS, para el manejo de datos geográficos. La vinculación permitirá almacenar el mapa dibujado en modo de datos geográficos que podrá ser fácilmente accedida por el software mediante un ORM que para el caso es SequelizeJS.

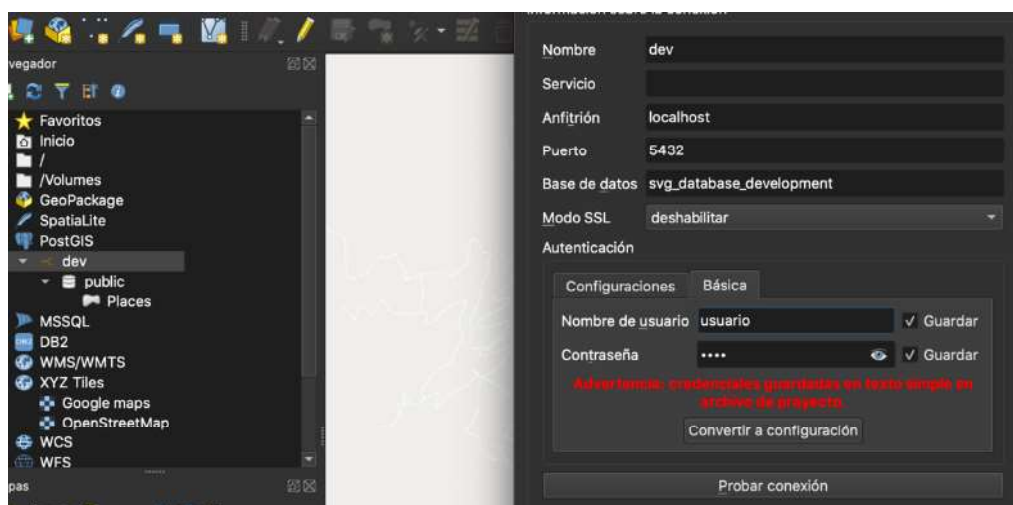


Figura 3. Vinculación de aplicación QGis con la Base de datos.

El proceso de dibujo se realiza calcando el perfil del mapa de Google Maps vinculado a QGis, para poder usar las coordenadas en la geodata. La Geodata está relacionada a los puntos geográficos, esto permite comparar con la posición del GPS que captura el dispositivo móvil y permitirá verificar en que polígono y posición se encuentra la persona. Mediante esto, podemos informar a la persona sobre su mapa actual y los polígonos cercanos a su posición. En la Figura 4 se puede ver como el mapa de prueba del parque La Carolina se dibuja con la herramienta QGIS.

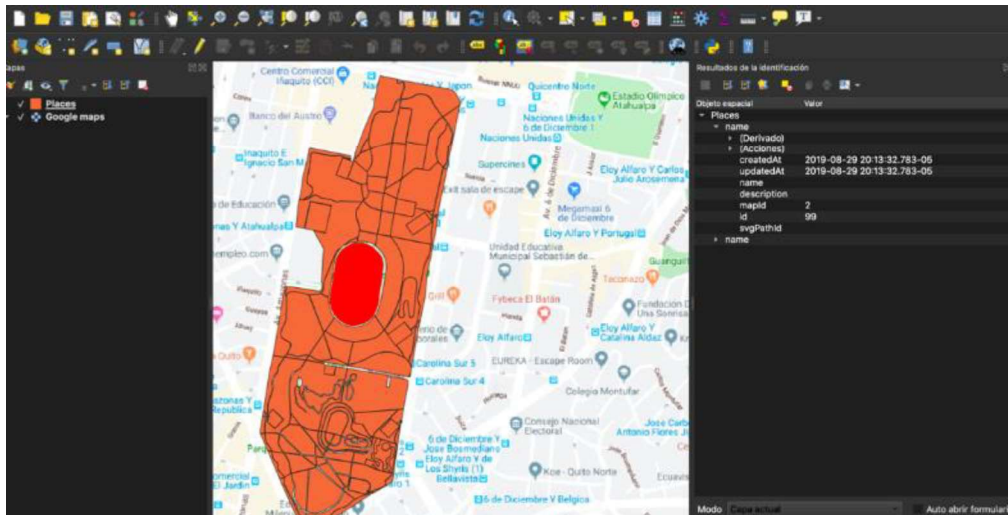


Figura 4. Ventana de información del polígono actual.

El proceso de dibujo se asemeja a el proceso de dibujo de un técnico graficador de AutoCAD, con la ventaja de la geo-posición y datos de referencia como nombres de polígonos, fecha de creación, identificadores únicos, etcétera, que facilita el desarrollo y análisis en la aplicación móvil.

3.2. Herramientas Tecnológicas.

Las siguientes herramientas detalladas a continuación fueron cuidadosamente seleccionadas para alcanzar el objetivo deseado. Además de que por su funcionalidad las hace adecuadas para desplegar un ambiente web controlable y mantenible.

Node JS y Express JS

Node JS es el motor de ejecución en un solo hilo (thread) para JavaScript que usaremos para este proyecto. Este motor fue construido sobre una versión moderna de V8 [23]. Express JS es el framework que emplearemos para usar los beneficios que nos provee Node JS. Se caracteriza principalmente por ser minimalista y flexible.

JavaScript

JavaScript es un lenguaje de scripting principalmente orientado a navegadores web, el cual se ha convertido en esencia para la construcción de aplicaciones web modernas. JavaScript está diseñado para admitir el estándar ECMAScript, el cual incluye un modelo de objeto particular y funciones integradas [24].

TypeScript

TypeScript es un súper set de JavaScript que se compila a JavaScript, empieza desde la misma sintaxis y semánticas de ECMAScript ya que usa el mismo código de JavaScript agregado tipos (TYPES) para una mayor productividad y buenas prácticas como la revisión estática de código y refactorizaciones [25].

PostgreSQL y PostGIS

PostgreSQL un motor de base de datos, relacional, open source que usa y extiende el lenguaje SQL combinado con varias características. PostgreSQL es altamente extensible, ya que nos provee de nuestros propios tipos o incluso permite usar plugin para propósitos específicos como PostGIS [26].

PostGIS es una extensión de para PostgreSQL que permite el manejo de información geográfica y/o espacial, agrega soporte y funciones para manipular objetos espaciales con comandos entendibles para lenguajes SQL (consultas SQL) [27].

3.3. Estructura de la aplicación móvil

3.3.1. La plataforma Android

Android es un sistema operativo que corre en una gran variedad de dispositivos (teléfonos, tablets, televisores, etc.). Esta es una plataforma totalmente libre basada en Linux que permite desarrollar aplicaciones con lenguaje de Java. Tiene una gran cantidad de API proporcionadas por Google entre las cuales vamos a utilizar la Interfaz TTS de los dispositivos [28].

La plataforma Android es una de las más extendidas alrededor del mundo. Siendo adoptada por muchos fabricantes de terminales móviles teniendo alrededor de 2.000 millones usuarios [29]. Por lo tanto, la opción de desarrollar una aplicación para dispositivos Android es lo más viable para la difusión de una aplicación móvil.

3.3.2. Compatibilidad de dispositivos

Con la llegada de Android 5.0 (API 21) de Android se introduce cambios muy grandes en la API de desarrollo y las funcionalidades, haciendo que la compatibilidad con versiones anteriores de Android sea más complicada [30]. Al momento de crear una aplicación se debe acaparar la mayor compatibilidad de versiones. En este proyecto se escoge una compatibilidad base con Lollipop (API 21) tenemos una compatibilidad del 89.3%. En la

Tabla 12 podemos ver el porcentaje de usuarios en 2019 por cada una de las distribuciones de Android [31].

5.0	Lollipop	21	3.0%
5.1		22	11.5%
6.0	Marshmallow	23	16.9%
7.0	Nougat	24	11.4%
7.1		25	7.8%
8.0	Oreo	26	12.9%
8.1		27	15.4%
9	Pie	28	10.4%

Tabla 12: Porcentaje de usuarios Android por versión [31, 19]

3.3.3. Integración con Android TTS.

El sintetizador de voz de Android permite convertir bloques de texto en notas de voz o archivos de audio. La plataforma Android permite una fácil integración de esta característica dentro de las aplicaciones desarrolladas por usuarios ya que cuenta con un paquete que permite la conexión con dicha característica del sistema [32]. Esta permite también una gran personalización debido a que la API recibe una gran cantidad de parámetros. Cabe aclarar que las desde aplicaciones solo podemos hacer llamadas a las operaciones TTS y estas se ejecutan de manera asíncrona a la aplicación que realizo la petición [32], por lo tanto, debemos tomar en cuenta aplicar los controles necesarios.

Con este se va a mostrar la información al usuario enviando mensajes de voz de los lugares contenidos en el mapa SVG.

3.4. Flujo de trabajo de la aplicación.

La aplicación se basa en un navegador web personalizado el cual va a permitir la comunicación entre la página web y el motor TTS de el dispositivo móvil.

La aplicación necesitara de los permisos para la obtención de la información de la ubicación del dispositivo. Esta localización será enviada al servidor como parámetro de la petición HTTP. Esta aplicación obtendrá la página web desde el servidor. Posteriormente la página será analizada y enviada hacia el motor TTS.

La página web contendrá un SVG con los datos necesarios para ofrecer al usuario una respuesta auditiva de todos los lugares del mapa en el cual se encuentra.

En la Figura 5, se puede ver de manera general el proceso que sigue la aplicación:

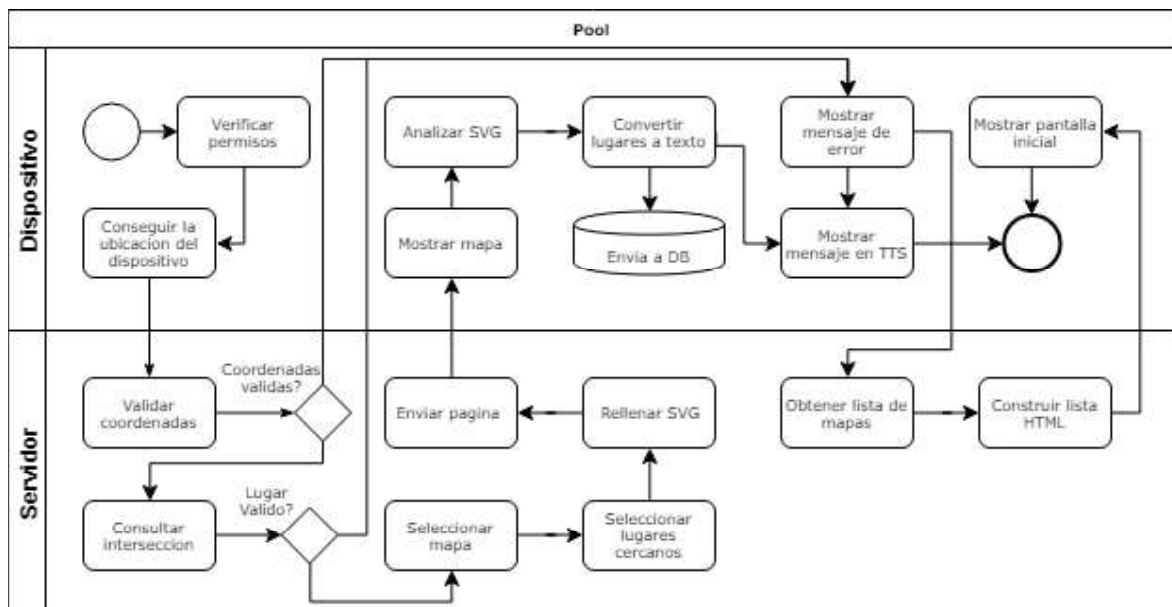


Figura 5. Flujo de trabajo de la aplicación.

3.4.1. Operaciones con las coordenadas

Al momento de recibir la petición HTTP con las coordenadas sean números válidos. En caso de no ser así la petición será redirigida a la lista de mapas disponibles.

En caso de ser correcta la información, se realizará una operación de intersección de las coordenadas incluidas den la petición con los polígonos guardados en la base de datos. Lo cual nos indicara en que mapa está el usuario actualmente.

Con el lugar identificado, se procede a localizar los lugares cercanos con respecto al punto que se extrajo de la petición. Estas distancias serán incluidas en la imagen SVG de la página de respuesta.

3.4.2. Estructura del formato SVG.

La estructura de la información adicional para cada uno de los paths en el SVG se puede ver en la Figura 6:

```
<svg>
  <title>(Nombre del mapa)</title>
  <desc>(Descripcion del mapa)</desc>
  <metadata id="place-information">
    <rdf:about>
      <cc:place rdf:about="(codigo-lugar)">
        <dc:id>(place-id)</dc:id>
        <dc:distance>(distancia)</dc:distance>
        <dc:nombre>(nombre)</dc:nombre>
        <dc:descripcion>(descripcion)</dc:descripcion>
        <dc:svgid>(svgid)</dc:svgid>
      </cc:place>
    </rdf:about>
  </metadata>
</svg>
```

Figura 6: Estructura de la imagen SVG.

A partir de esta estructura de los metadatos se puede obtener los mensajes que se van a procesar por el motor de voz del dispositivo del usuario.

Los datos incluidos en el SVG dentro de la etiqueta metadata son:

- Nombre del mapa: se registra el nombre en la etiqueta <title> para que los lectores de pantalla lo puedan hallar con facilidad.
- Descripción del mapa: Dentro de este se incluye una pequeña descripción del lugar que representa ese SVG.
- Código del lugar: código único que nos va a servir para identificar más fácilmente el lugar al que pertenece la información.
- Nombre del lugar: Nombre que nos servirá para construir el mensaje que se va a procesar por la aplicación.
- Distancia: Distancia a la que se encuentra el usuario del lugar.
- Svg-id: identificador que nos muestra el identificador de el paths dentro del SVG. En su mayoría tienen el mismo valor que el valor del código del lugar.
- Id del lugar: Información adicional que permite identificar el lugar en la base de datos.

- Descripción del lugar: Información adicional que contiene una breve descripción del lugar.

3.4.3. Proceso en la aplicación

Al recibir la respuesta desde el servidor con la página correspondiente, la aplicación analizará los datos incluidos en la metadata del SVG y los almacenará en una base de datos local para presentar información auditiva más personalizada.

Una vez terminado la carga del mapa el usuario podrá listar los lugares cercanos con las teclas de volumen del dispositivo. Los lugares se mostrarán en el mapa y se enviará una respuesta auditiva al usuario.

Permisos en la aplicación.

La aplicación al estar basada en la información de la ubicación del dispositivo es necesario que esta conceda los permisos necesarios para acceder a la información de la ubicación.

Para esto es necesario que se ingresa en los ajustes de la aplicación y se conceda los permisos para la ubicación.

Configuración del motor de voz.

La configuración del motor de voz se debe realizar antes de la ejecución de la aplicación para evitar problemas al momento de enviar los mensajes al usuario.

Para configurar este parámetro dirigirse a la sección de Ajustes, en esta dirigirse a la sesión de Accesibilidad, luego seleccionar la opción Salida de texto a voz. Una vez aquí seleccionar e instalar el paquete de voz en español.

Una vez instalado el paquete proceder a con la selección y realizar una prueba para comprobar que este funcione adecuadamente.

4. Metodología de desarrollo

El proceso de gestión de un proyecto de desarrollo de software sigue las reglas básicas de gestión de proyectos, pero también incluye características particulares. Seleccionar la metodología de desarrollo de software adecuada para un proyecto depende en gran medida del tamaño del equipo, los objetivos y otros factores.

La metodología tradicional de proyectos está establecida donde los proyectos se ejecutan en un ciclo secuencial. Mantienen una secuencia fija: iniciación, planificación, ejecución, monitoreo y cierre. El enfoque tradicional de poner especial énfasis en procesos lineales, documentación, planificación inicial y priorización. Según el método tradicional, el tiempo y el presupuesto son variables y los requisitos se fijan debido a que a menudo se enfrentan a problemas de presupuesto y cronología.

En cambio, las metodologías ágiles son un enfoque general utilizado para el desarrollo de software, depende en gran medida del trabajo en equipo, la colaboración, las tareas de lidiar con el tiempo y la flexibilidad para responder al cambio lo más rápido posible.

En la Tabla 13 se muestra una comparación entre metodologías ágiles y metodologías tradicionales [33]:

	Métodos ágiles	Métodos tradicionales
Enfoque	Adaptativo	Predictivo
Medición de éxito	Valor agregado al negocio	Conformado en planificar
Tamaño del proyecto	Pequeño	Largo
Estilo de gestión	Descentralizado	Autocrático
Perspectiva de cambio	Adaptable al cambio	Sostenible al cambio
Cultura	Liderazgo / Colaboración	Comando / Control
Documentación	Poca	Mucha
Énfasis	Orientado a las personas	Orientado al proceso
Ciclos	Numerosos	Limitados
Dominio	Impredecible / Exploratorio	Predecible
Planificación Inicial	Mínima	Comprehensiva
Retorno de la inversión	Al principio del proyecto	Al final del proyecto
Tamaño del equipo	Pequeño / Creativo	Largo

Tabla 13. Comparación metodologías ágiles y tradicionales.

Debido a lo expuesto anteriormente, es necesario implementar una metodología ágil para el desarrollo de este proyecto, ya que necesitamos las facilidades de colaboración de equipo y una ágil respuesta al cambio en corto plazo a fin de otorgar entrega continua del producto final que generen valor agregado a la conclusión del proyecto.

4.1. Selección de metodología

Cinco de los primeros métodos ágiles publicados entre 1995 y 2002 fueron seleccionados para análisis y comparación: Métodos DSDM, XP, SCRUM, LSD, FDD y Kanban. El análisis de cinco métodos proporciona suficientes datos para una comparación significativa de similitudes y diferencias entre cada uno de los métodos, y, por último, es probable que contengan la mayor parte de técnicas utilizadas en métodos ágiles publicados posteriormente.

El detalle comparativo de los marcos de trabajo se muestra en la Tabla 14 [34]:

Criterio \ Metodología	SCRUM	XP	DSDM	FDD	Kanban
Tamaño de equipo	43713	43740	43740	43922	-
Tamaño de proyecto	todos	pequeño	todos	largo	todos
Iteraciones (semanal)	4	2	-	2	1
Roles y responsabilidades	definido	definido	definido	definido	definido
Procesos centrados	no definido	no definido	definido	definido	definido
Personas centradas	definido	definido	no definido	no definido	no definido
Soporte de equipo virtual	definido	no definido	no definido	definido	definido
Mitigación de riesgos altos	definido	no definido	definido	no definido	no definido
Mitigación de riesgos medianos	no definido	definido	no definido	definido	definido
Documentación	básica	básica	existe	importante	no definido
Reuniones diarias	definido	definido	no definido	no definido	definido
Compartición de información (vía documentos)	no definido	no definido	definido	definido	no definido
Reuniones cara a cara	no definido	no definido	no definido	no definido	no definido
Basado en modelos	no definido	no definido	no definido	no definido	no definido

Tabla 14. Comparación de características de las metodologías.

Para este proyecto se empleará SCRUM, uno de los marcos de trabajo más populares de la metodología de implementación ágil utilizada por las compañías en todo el mundo. SCRUM está diseñado para equipos más pequeños y se basa en ciclos de desarrollo de dos semanas, llamados sprints.

SCRUM brinda una mayor responsabilidad y permite a las personas trabajar en una variedad de especializaciones. Esto es gracias a que todo el equipo se comunica efectivamente y todos están en la misma página. [35]

El equipo debe pensar enfocado en todo aspecto para resolver problemas, lo que a menudo da como resultado soluciones más creativas e ideas nuevas.

Lo más importante, SCRUM nos permite entregar actualizaciones y nuevos productos a nuestros clientes de manera efectiva y eficiente, lo cual es vital en un mercado en constante cambio. La metodología no solo ofrece incrementos oportunos, sino que también nos permite abordar los problemas de funcionalidad con cada iteración, asegurando que lo hagamos bien antes de continuar. Las metodologías ágiles como SCRUM fueron diseñadas para resolver los errores inevitables tanto en el software en sí como en la forma en que los equipos trabajan juntos, lo que resulta en un mejor producto para el cliente. [36]

4.2. Aplicación de la metodología al proyecto

4.2.1. Marco de trabajo SCRUM

El presente proyecto tiene como objetivo ayudar a personas no videntes, por lo tanto, los requerimientos del producto definidos en una planeación inicial podrían no abarcar todo lo necesitado por el usuario. Debido a esto se debe optar por una metodología que permita una respuesta rápida en cuanto a la inclusión de cambios. Las metodologías ágiles nos permiten resolver estos cambios y entre los framework que utilizan metodologías ágiles, SCRUM nos permite tener un mejor control del avance del producto final mediante las entregas al final de cada cierto periodo.

Por lo expuesto anteriormente la metodología de desarrollo que se va a utilizar en este proyecto es SCRUM, esta basa principalmente en la premisa de ejecutar un proyecto en entregas parciales y regulares del producto. SCRUM es ideal para equipos con pocas personas las cuales van a desempeñar un rol específico en el proceso [37].

Con esta metodología se busca que el desarrollo se realice en el tiempo planificado, con entregas constantes para poder mantener un buen estado del producto. SCRUM tiene la ventaja de ser flexible con respecto a la utilización de herramientas [38].

Roles

Los roles en SCRUM son bastante diferentes de los métodos de software tradicionales. Los roles y expectativas claramente definidos ayudan a las personas a realizar sus tareas de manera eficiente. En SCRUM, hay tres roles: propietario del producto (product owner), equipo de desarrollo (dev team) y SCRUM Master. Juntos, estos se conocen como el Equipo SCRUM [39].

- SCRUM Master: ayuda al grupo a comprender y aplicar SCRUM para lograr un valor comercial. El SCRUM master hace todo lo que está a su alcance para ayudar al equipo, al product owner y a la organización a tener éxito. El SCRUM Master no es el gerente de los miembros del Equipo, ni tampoco es un gerente de proyecto, líder de equipo o representante del equipo. En cambio, el SCRUM master sirve al equipo; él o ella ayuda a eliminar impedimentos, protege al equipo de interferencias externas y lo ayuda a adoptar prácticas de desarrollo ágiles. Él o ella educa, entrena y guía al product owner, al equipo y al resto de la organización en el uso hábil de SCRUM.
- Propietario del producto: es responsable de identificando las características del producto, traduciéndolas en una lista priorizada, decidiendo cuál debe estar en la parte superior de la lista para el próximo Sprint, y continuamente re-priorizando y refinando la lista. El Product Owner en Agile es como un portavoz de los clientes y necesita representarlos.
- Equipo de desarrollo: Un equipo de desarrollo es una colección de personas que trabajan juntas para desarrollar y entregar los incrementos de producto solicitados y comprometidos. Se compone de miembros multifuncionales que son capaces de lograr los objetivos del sprint. Esto podría incluir ingenieros de software, arquitectos, programadores, analistas, administradores de sistemas, expertos en control de calidad, diseñadores de interfaces de usuario, etc.

Artefactos

Los artefactos SCRUM proporcionan información clave que el equipo y las partes interesadas deben conocer para comprender el producto en desarrollo, las actividades que se planifican y las actividades realizadas en el proyecto.

Estos son los artefactos más comunes en un proyecto SCRUM y los artefactos del proyecto no están limitados por estos [40].

- **Visión del producto:** La visión del producto es un artefacto para definir el objetivo a largo plazo del proyecto / producto. Establece la dirección general y guía al equipo SCRUM. Todos deberían poder memorizar la Visión del Producto; por lo tanto, debe ser breve y preciso.
- **Gol Sprint:** El objetivo de Sprint ayuda a enfocar en el mismo. Es el objetivo que se alcanzará dentro del Sprint a través de la implementación de los elementos previstos del Backlog del Producto, y proporciona orientación al Equipo de Desarrollo sobre por qué está construyendo el Incremento del Producto.
- **Pila de Producto:** Una pila (stack) de productos es una lista de todas las cosas que se requieren en el producto y es un requisito dinámico y mejor entendido para cualquier cambio que se realice en el producto. La pila de productos del propietario del producto (PO), que consta de una lista de todas las características, funciones, requisitos, mejoras y correcciones que constituyen los cambios que se realizarán en el producto en futuras versiones.
- **Sprint Backlog:** El Sprint Backlog es el conjunto de elementos del Backlog del Producto seleccionados para el Sprint más un plan para entregar el Incremento del producto y alcanzar el Objetivo Sprint. El Sprint Backlog es un pronóstico del Equipo de Desarrollo sobre qué funcionalidad estará en el próximo Incremento y el trabajo necesario para brindar esa funcionalidad. El Sprint Backlog define el trabajo que realizará el Equipo de Desarrollo para convertir los elementos del Sprint Backlog en un "Listo". El Sprint Backlog hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para cumplir con el Objetivo del Sprint.
- **Definición de Hecho:** Cada elemento de la Lista de Producto tiene criterios de aceptación que definen de manera medible lo que debe cumplirse cuando se declara que el elemento está hecho. Muchos criterios se aplican a todos o muchos elementos de la Lista de Producto. En lugar de definir repetidamente estos criterios con cada elemento, ha resultado útil recopilar estos criterios en un solo lugar: la Definición de Hecho. Por lo tanto, la Definición de Hecho es una comprensión compartida del Equipo SCRUM sobre el significado del trabajo a completar. Por lo general, contiene criterios de calidad, restricciones y requisitos generales no funcionales.
- **El Burndown Chart:** Los Burndown Charts son gráficos que ofrecen una visión general del progreso a lo largo del tiempo mientras se completa un proyecto. A medida que se completan las tareas, el gráfico "se quema" a cero. Se utiliza como una herramienta para guiar al equipo de desarrollo a completar con éxito un Sprint a tiempo con un producto final que funcione. Si un equipo decide que ha movido

más objetivos de los posibles para completar desde el Backlog del Producto al Sprint Backlog, el Burndown Chart puede ayudarlos a determinar qué tareas no pueden completar de manera realista para que estas tareas puedan volver al Producto Reserva.

Eventos

A continuación, se detallan los cinco eventos importantes en SCRUM [41]:

1. Planificación del sprint: se produce un evento en un margen de tiempo al comienzo de un sprint en el que el equipo determina los elementos de la cartera de productos en los que trabajarán durante ese sprint.
2. SCRUM daily meeting: es un evento de 15 minutos para el Equipo de Desarrollo para sincronizar actividades y crear un plan para las próximas 24 horas.
3. Revisión de Sprint: se lleva a cabo un evento con un margen de tiempo al final del Sprint para inspeccionar el Incremento y adaptar la Cartera de Producto si es necesario.
4. Retrospectiva de Sprint: un evento con un margen de tiempo para proporcionar una oportunidad para que el Equipo SCRUM se inspeccione a sí mismo y cree un plan de mejoras para el próximo Sprint.

4.2.2. Tratamiento de los roles de SCRUM

La asignación de roles dentro del marco de trabajo se definió en una reunión previa a la ejecución de los Sprint. Los roles de SCRUM dentro de este proyecto se asignan de la siguiente manera:

- Dueño del producto: Fundación Procodis y usuarios no videntes, los cuales proveerán toda la información relevante para el desarrollo del sistema.
- SCRUM master: Ángel Higuera. Sera el encargado de la redacción y organización de las historias
- Equipo de desarrollo: Ángel Higuera y Darío Calo. Encargados de la ejecución de las historias y desarrollo del sistema. Adicionalmente estimaran las historias asignándoles un valor de esfuerzo.

4.2.3. Estimación de las historias y prioridades.

Estimacion de las historias.

El valor de los puntos es definido por consenso en la organización debido a que no todas las personas trabajan al mismo ritmo. El objetivo es llegar a un consenso general ya que

cada persona conoce tanto sus habilidades como limitaciones y por lo tanto dos personas no tienen por qué coincidir en el tiempo que crean necesario para realizar la misma historia de usuario [42].

Debido a limitaciones de tiempo debido a otras actividades se estima un tiempo aproximado de 2 horas por día destinado a la ejecución de las historias de este proyecto. Por lo tanto, por consenso del equipo se definió que 1 punto equivale a una jornada de trabajo.

Con la utilización de una métrica basada en puntos el equipo no está tan presionado, las personas no tienen tanto miedo a equivocarse, es decir, miedo de no terminar a tiempo la historia de usuario en el en un tiempo determinado. El estrés que siente el equipo es menor porque ahora no tienen que trabajar pendientes del reloj sino del esfuerzo [42].

Priorización de las historias.

Para asignar la prioridad de cada una de las historias se optó por una valoración según la técnica Moscow, esquema el cual obtiene deriva su nombre de las primeras letras de las frases “Must Have”, “Should Have”, “Could Have” y “Won’t Have” [43].

Este método de priorización es generalmente más efectivo que los esquemas simples. Los niveles están en orden de prioridad decreciente, siendo más importantes aquellas sin las que el producto no tendría valor. Las historias que son de baja prioridad son aquellas que, aunque sería bueno tenerlas, no es necesario incluirlas [43].

En base a lo que se explicó anteriormente de esta técnica se puede priorizar las historias de la siguiente manera:

- Prioridad alta: Historias las cuales son una parte esencial del producto y que sin ellas el producto no funcionaría.
- Prioridad media: Historias que representan una funcionalidad complementaria a las primeras. Estas historias si bien son importantes se puede aplazar su ejecución.
- Prioridad baja: Historias que representan algo adicional al producto pero que no afectan a las demás partes del producto si no están presentes.

En base a estos 3 niveles se puede clasificar la importancia de cada una de las historias en la definición de las mismas.

Uso del Sprint de planeación (Sprint 0)

El inicio del proyecto debe tener una estructura ya definida para empezar el desarrollo de manera óptima. Para esto es necesario definir un tiempo previo a la iniciación del proyecto para la planificación y estructuración de los elementos del mismo. Este tiempo se lo puede denominar Sprint 0 y tendría las siguientes características [44]:

- Crear la estructura básica del proyecto.
- Tener un diseño mínimo de los procesos.
- Desarrollar historias de configuración ligeras.
- Tener una duración corta.

El desarrollo de este sprint permite que al inicio del proyecto las historias puedan ser iniciadas sin retrasos por configuración.

4.2.4. Integración continua y Test-driven development (TDD).

La integración continua (de las siglas CI comúnmente usadas en por los desarrolladores de software) es la práctica de integrar rutinariamente los cambios de código en la rama principal de un repositorio, y probar los cambios, tan pronto como sea posible. Idealmente, los desarrolladores integrarán su código diariamente, si no varias veces al día. Mediante el uso de software de terceros en la implementación de pruebas unitarias, pruebas de aceptación y/o pruebas de integración [45]

El objetivo de TDD es un código limpio que funcione. Consiste escribir una prueba, hacerlas correr y verificar que todo esté acorde a lo aceptado anteriormente. Las preguntas abiertas que debe abordar para garantizar el TDD son [46]:

- ¿Cómo se aplican los enfoques de TDD a las bases de datos, las GUI y los sistemas distribuidos?
- ¿Cuáles son los puntos de referencia cuantitativos que pueden demostrar el valor de TDD
- ¿Cuáles son los mejores enfoques para resolver el problema ubicuo de la escalabilidad?

5. Desarrollo del sistema.

En el capítulo anterior se seleccionó la metodología de desarrollo SCRUM, en este capítulo se va a describir cómo se desarrolla cada una de las historias del sistema y como se organizan a lo largo de la duración del proyecto.

5.1. Planeación del desarrollo.

El desarrollo de la aplicación se va a dividir en dos partes:

- La construcción de un servidor que provea las imágenes en formato SVG con la metadata necesaria.
- La aplicación móvil que permita el despliegue de imágenes y provea una interfaz auditiva para el usuario.

Los requerimientos funcionales del proyecto se pueden ver a continuación:

- Servidor que pueda desplegar páginas con imágenes SVG de mapas.
- Servidor que pueda identificar si un punto está dentro de los mapas guardados.
- Servidor que permita el almacenamiento de imágenes vectoriales.
- Aplicación móvil que permita el despliegue de los mapas.
- Aplicación móvil que se tenga una interfaz de voz para comunicar la información al usuario.

Los requerimientos funcionales se relacionan con la implementación de los estándares para personas con discapacidad visual:

- Implementación de estándares W3C a la página web mostrada en el dispositivo móvil para que esta se pueda mostrar más fácilmente en otros navegadores.
- Implementación de una interfaz responsiva en la página a desplegar.
- Implementación de la metadata en los las imágenes SCG a mostrar.

En resumen, con el fin de cumplir con los requerimientos se propone los siguientes puntos a cumplir:

- Crear la configuración necesaria para iniciar el proyecto.
- Servidor que provea la interfaz web para proveerlos mapas a la aplicación.
- Aplicación móvil Android que presente los mapas en SVG.
- Crear los mapas para realizar las pruebas.

Definición de Historias

Con el fin de seguir la metodología SCRUM se va a estructurar varias historias épicas con el fin de realizar una planificación correcta de todo el esfuerzo. Dentro de cada una de estas historias épicas se definirá cada una de las características que se puedan abordar con una historia. Con el fin de clasificar a cada una de las historias de mejor manera se añadió a estas un prefijo de acuerdo a la Tabla 15.

Etiqueta	Áreas que abarca
CNF	Configuración de la estructura del proyecto. Configuración de plataforma de código.
SRV	Desarrollo del código de la aplicación. Definían de configuraciones para Node.js.
MAP	Diseño de los mapas.
APP	Desarrollo de código de la aplicación móvil.

Tabla 15. Etiquetas de las historias.

Historias épicas

Se denomina Epic o historias Épica a una historia de usuario que, por su gran tamaño, el equipo descompone en historias con un tamaño más adecuado para ser gestionada con los principios y técnicas ágiles: estimación y seguimiento cercano (normalmente diario). Es adecuado construir una historia épica por cada uno de los objetivos generales con el fin de organizar y realizar seguimiento del progreso de forma más exacta [47].

Con el fin de cubrir los puntos básicos de la aplicación se va a definir las historias épicas definidas en la Tabla 16, Tabla 17, Tabla 18, Tabla 19, las cuales abarcan un aspecto general del desarrollo del sistema.

Historia épica	CNF01
Título: Levantar la configuración necesaria para ejecutar el servidor.	
Descripción: Con el fin de empezar el desarrollo de la aplicación se necesita una infraestructura en la cual se pueda desplegar de manera eficiente el código.	

Tabla 16. Historia épica CNF01.

Historia épica	SRV01
Título: Desarrollar una página que muestre el mapa con el mapa en formato SVG.	
Descripción: La función principal del servidor es mostrar una página web la cual contenga la imagen del mapa en formato SVG.	

Tabla 17. Historia épica SRV01.

Historia épica	MAP01
Título: Diseñar una imagen SVG de los mapas que se requiera para la aplicación.	
Descripción: La pieza principal de la página es el SVG de cada uno de los mapas por lo tanto es necesario crear los mimos con los programas descritos con anterioridad.	

Tabla 18. Historia épica MAP01.

Historia épica	APP01
Título: Desarrollar una aplicación Android capaz de mostrar mapas y lugares.	
Descripción: Para que la metadata de la aplicación sea interpretada adecuadamente es necesario crear una aplicación que sirva de interfaz para la página que despliega los mapas.	

Tabla 19. Historia épica APP01.

Creación de historias

Cada una de las historias épicas abarca un grupo de historias las cuales se detallan a continuación. La estimación y prioridad de las historias se definió de acuerdo a lo descrito en la sección 3.4.3.

La historia épica CON01 que se centra en la configuración inicial de la aplicación del servidor basada en Node.js. Como se puede ver en la Tabla 20 esta historia épica abarca todas las historias con el prefijo CNF.

Historia épica	Titulo		
CNF01	Levantar la configuración necesaria para ejecutar el servidor.		
Historias			
Identificador	Titulo	Prioridad	Esfuerzo
CNF02	Configurar un servidor remoto para desplegar la aplicación.	Alta	1
CNF03	Implementar una estructura base para la aplicación Node.js en un repositorio de código.	Alta	2
CNF04	Crear una estructura para ejecutar pruebas unitarias para la aplicación de Node.js	Alta	2
CNF05	Crear una base de datos en PostgreSQL para almacenar los datos de la aplicación	Alta	3
CNF06	Crear un proceso de despliegue de código al servidor.	Media	1

Tabla 20. Historias contenidas en la historia épica CNF01.

La historia épica SRV01 abarca el desarrollo como de la aplicación del servidor. Adicionalmente se abarca la configuración de los plugin de PostgreSQL. Como se puede ver en la Tabla 21 esta historia épica abarca todas las historias con el prefijo SRV.

Historia épica	Titulo		
SRV01	Desarrollar una página que muestre el mapa con el mapa en formato SVG.		
Historias			
Identificador	Titulo	Prioridad	Esfuerzo
SRV02	Implementación de la estructura de las rutas mediante express.js.	Alta	3
SRV03	Implementar la vista para el mapa de prueba	Alta	1
SRV04	Implementar los modelos necesarios para levantar la aplicación con su respectiva correlación a la base de datos.	Alta	3
SRV05	Implementar la operación de detección de polígonos mediante los parámetros de coordenadas.	Media	3
SRV06	Listar los lugares cercanos	Media	3
SRV07	Resaltar el lugar seleccionado en a imagen SVG	Media	2
SRV08	Implementar una función que me señale el lugar más cercano al mapa cuando estoy fuera de él.	Media	2

Tabla 21. Historias contenidas en la historia épica SRV01.

La historia épica MAP01 agrupa las historias relacionadas con la creación de las imágenes con sus respectivas coordenadas. Como se puede ver en la Tabla 22 esta historia épica abarca todas las historias con el prefijo MAP.

Historia épica	Titulo		
MAP01	Diseñar una imagen SVG de los mapas que se requiera para la aplicación.		
Historias			
Identificador	Titulo	Prioridad	Esfuerzo
MAP02	Crear un mapa de prueba para iniciar el desarrollo de la aplicación.	Alta	3
MAP03	Crear un mapa de las cercanías de la Fundación Procodis.	Media	3

Tabla 22. Historias contenidas en la historia épica MAP01.

La historia épica APP01 descrita en la siguiente tabla abarca todo lo relacionado con funcionalidades de la aplicación móvil. Como se puede ver en la Tabla 23 la historia épica abarca todas las historias con el prefijo APP.

Historia épica	Titulo		
APP01	Desarrollar una aplicación Android capaz de mostrar mapas y lugares.		
Historias			
Identificador	Titulo	Prioridad	Esfuerzo
APP02	Crear una aplicación base para iniciar el desarrollo.	Alta	3
APP03	Obtener las coordenadas de la posición geográfica.	Media	3
APP04	Implementar una función que llame al servidor remoto.	Alta	2
APP05	Implementar una base de datos en la aplicación para guardar las configuraciones.	Alta	3
APP06	Implementar la conexión para el uso del motor TTS del teléfono.	Media	2
APP07	Mostrar la página con el SVG.	Alta	1
APP08	Implementar la navegación de los lugares cercanos mediante los botones de volumen.	Media	2
APP09	Mostrar mediante mensajes de audio los lugares cercanos.	Media	3

APP10	Implementar la navegación en el listado de mapas mediante las teclas de volumen.	Media	3
-------	--	-------	---

Tabla 23. Historias contenidas en la historia épica APP01.

En total se tienen 19 historias las cuales como se describe en la Tabla 24 se distribuyen a lo largo de 4 Sprint.

Sprint 1	Sprint 2	Sprint 3	Sprint 4
SRV02	APP03	SRV06	SRV08
MAP02	APP04	APP08	MAP03
SRV03	SRV05	APP09	APP10
SRV04	APP05	SRV06	
APP02	APP06		
CNF05	APP07		

Tabla 24. Distribución de las historias en los sprint.

Adicionalmente se ejecutaron 6 historias que corresponden a las configuraciones iniciales del proyecto en un sprint corto antes de la ejecución del primer Sprint. La Tabla 25 muestra las historias que se tomaron en cuenta en el Sprint 0.

Sprint 0
CNF02
CNF03
CNF04
CNF05
CNF06
CNF07

Tabla 25. Historias Sprint 0.

5.2. Desarrollo de los Sprint

5.2.1. Sprint 0: Configuración del entorno de desarrollo

Planificación del Sprint

Para este Sprint previo se va a enfocar a las configuraciones iniciales de la aplicación del servidor y código con el fin de empezar la implementación de características del servicio. La Tabla 26 describe las historias que se ejecutaron en el sprint 0 con sus respectivos puntos de esfuerzo.

Sprint 0	Titulo	Puntos
CNF02	Configurar un servidor remoto para desplegar la aplicación.	1
CNF03	Implementar una estructura base para la aplicación Node.js en un repositorio de código.	2
CNF04	Crear una estructura para ejecutar pruebas unitarias para la aplicación de Node.js	2
CNF06	Crear un proceso de despliegue de código al servidor.	1
Total		6

Tabla 26. Sprint backlog del sprint 0.

El total de puntos de este sprint es de 6 los cuales se reparten en la semana de ejecución del sprint 0.

Descripción y ejecución de las historias

Descripción de la historia

La Tabla 27 muestra las características, prioridad y esfuerzo de la historia CNF02.

Identificador	Titulo	Prioridad	Esfuerzo
CNF02	Configurar un servidor remoto para desplegar la aplicación.	Alta	1
Descripción	La página web debe estar disponible desde cualquier red, por lo tanto, esta debe estar alojada en un servidor en la nube.		
Criterios de Aceptación	<ul style="list-style-type: none">- La aplicación debe estar alojada en un servidor en la plataforma de Heroku.- La aplicación tiene acceso a una base de datos de PostgreSQL.		

Tabla 27. Descripción de la historia CNF02.

Implementación de la historia:

Dentro de la plataforma Heroku se creó un Dino de tipo Hobby Dev. Adicionalmente se implementó el plugin para la base de datos PostgreSQL. En la Figura 7 se puede ver la pantalla principal del dinamo de Heroku en el cual se visualiza el plugin de PostgreSQL.

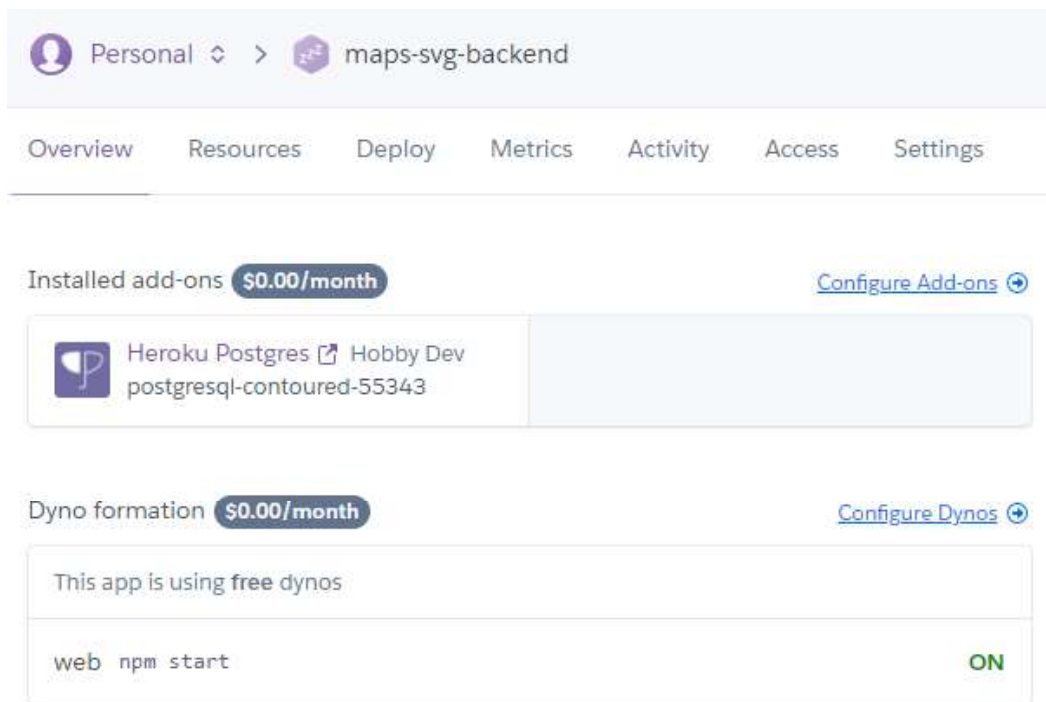


Figura 7. Características del servidor alojado en Heroku.

Descripción de la historia

La Tabla 28 muestra las características, prioridad y esfuerzo de la historia CNF03.

Identificador	Título	Prioridad	Esfuerzo
CNF03	Implementar una estructura base para la aplicación Node.js en un repositorio de código.	Alta	2
Descripción	La estructura del proyecto debe estar almacenada en un repositorio de código con el fin de poder organizar los cambios a lo largo del tiempo. Esta estructura debe corresponder con una aplicación Node.js		
Criterios de Aceptación	<ul style="list-style-type: none">- La aplicación debe estar alojada en un repositorio en GitHub- La aplicación corresponde a una aplicación de Node.js		

Tabla 28. Descripción de la historia CNF03.

Implementación de la historia:

Para la sincronización del código entre los colaboradores del proyecto se creó el repositorio “svgmaps” en la plataforma Github. Esto nos permitirá adicionalmente implementar un proceso de integración continua según la metodología TDD. La Figura 8 muestra información del repositorio que se creó para el proyecto.

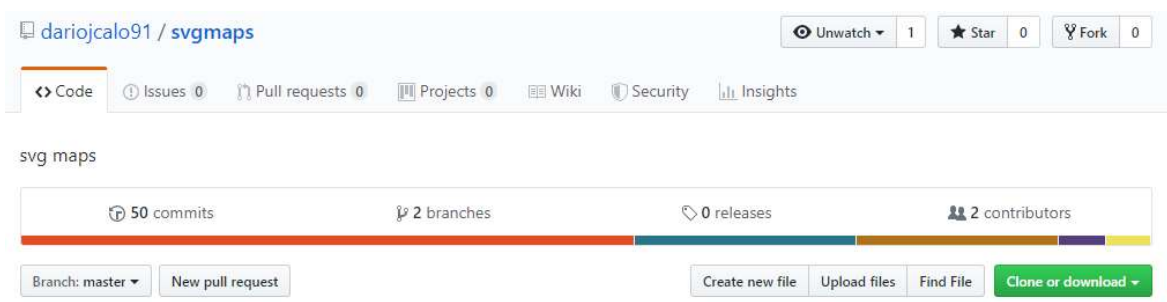


Figura 8. Repositorio en GitHub del proyecto.

El primer commit del proyecto incluye una estructura de una aplicación Node.js con una estructura que básica para levantar un servicio web la cual se puede ver en la Figura 9. Adicionalmente se especifica los paquetes necesarios para levantar dichos servicios.

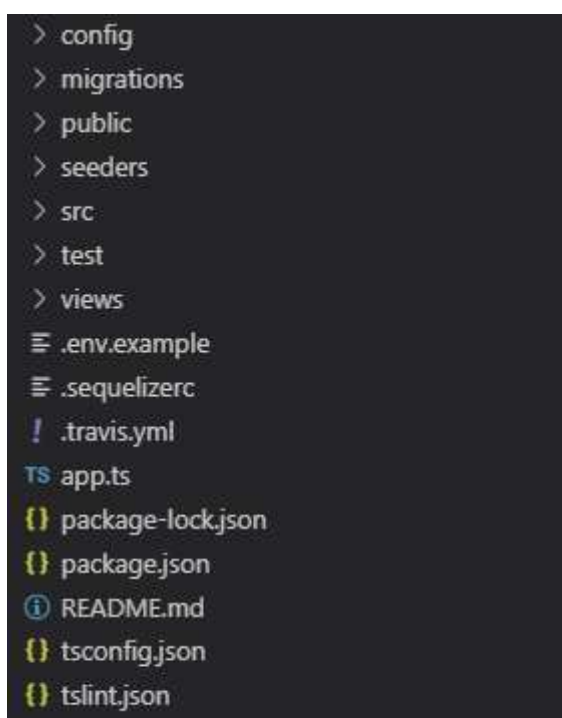


Figura 9. Estructura del proyecto.

Descripción de la historia

La Tabla 29 muestra las características, prioridad y esfuerzo de la historia CNF04.

Identificador	Título	Prioridad	Esfuerzo
CNF04	Crear una estructura para ejecutar pruebas unitarias para la aplicación de Node.js	Alta	2
Descripción	Las pruebas unitarias permiten a la aplicación comprobar la integridad de la funcionalidad con el fin de evitar errores graves en la aplicación.		
Criterios de Aceptación	- La aplicación debe estar conectada la plataforma Travis con el fin de ejecutar los test unitarios.		

Tabla 29. Descripción de la historia CNF04.

Implementación de la historia:

La plataforma Travis nos permite que los test unitarios se ejecuten cada vez que se abra un *Pull request*. Se ejecutará una comprobación de las funcionalidades de la aplicación del servidor que impide que suba a producción un cambio si falla en cualquier aspecto y notifica del error para que sea corregido. En la Figura 10 se puede ver un ejemplo de cómo se visualiza el estado de las pruebas.



Figura 10. Perfil para las pruebas unitarias en Travis.ci.

La configuración necesaria para que esta plataforma funcione es la mostrada en la Figura 11.

```

1  language: node_js
2  sudo: false
3  notifications:
4  |   email: false
5  services:
6  -  postgresql
7  addons:
8  |   postgresql: '9.5'
9  |   apt:
10 |     packages:
11 |     -  postgresql-9.5
12 |     -  postgresql-client-9.5
13 |     -  postgresql-9.5-postgis-2.4
14 node_js:
15 -  8.11.3
16 install:
17 -  npm install -g typescript
18 -  npm install
19 before_script:
20 -  psql -c 'create database svg_database_development;' -U postgres
21 -  psql -U postgres -d 'svg_database_development' -c "create extension postgis;"
22 -  psql -c 'create database svg_database_test;' -U postgres
23 -  psql -U postgres -d 'svg_database_test' -c "create extension postgis;"
24 -  npm run sequelize -- db:migrate
25 -  npm run sequelize -- db:seed:all
26 script:
27 -  npm test

```

Figura 11. Ejecución de las pruebas unitarias en Travis.ci.

Descripción de la historia

La Tabla 30 muestra las características, prioridad y esfuerzo de la historia CNF06.

Identificador	Título	Prioridad	Esfuerzo
CNF06	Crear un proceso de despliegue de código al servidor.	Baja	2
Descripción	Con la implementación de la estructura de integración continua podemos automatizar el despliegue de nuestro código hacia al servidor por medio de las configuraciones de Travis CI.		
Criterios de Aceptación	- La aplicación debe desplegarse hacia el servidor cada vez que se añadía nuevas características a la aplicación.		

Tabla 30. Descripción de la historia CNF06.

Implementación de la historia:

Mediante el archivo de configuración de Travis.ci podemos especificar a qué tipo de servidor vamos a desplegar el código. Para esto se obtuvo un api-key la cual va a servir para permitir a nuestro proceso de integración continua el accede al servidor de Heroku.

Adicionalmente de debe especificar el nombre del servidor y desde que rama del repositorio se va a iniciar el proceso, la Figura 12Figura 11 muestra esas configuraciones.

```

deploy:
  provider: heroku
  api_key:
    secure: soHet6fm1d3mYWkwMuU0M0Rov0zmAr3FzH
  app: maps-svg-backend
  on:
    repo: dariojcalo91/svgmaps
    #all_branches: true
    branch: master

```

Figura 12. Configuración para integrar Heroku con Travis.ci

Burndown chart Sprint 0

Como podemos ver en la Figura 13 la ejecución de este sprint se llevó a cabo de manera que no se distancio mucho de lo estimado.

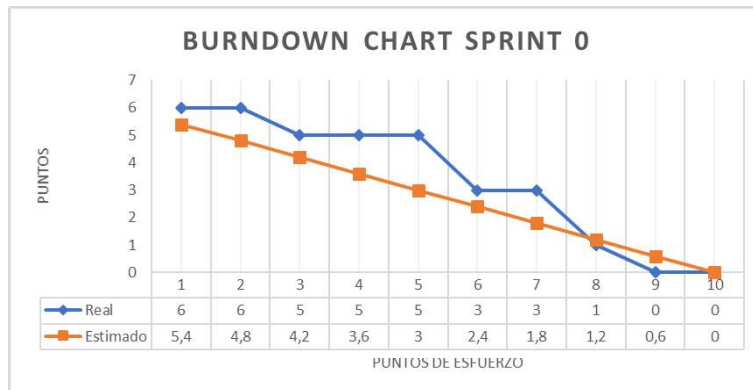


Figura 13: Tabla Burndown de la ejecución del sprint 0.

La línea de la estimación no se aleja significativamente de la ejecución real de el sprint. Lo cual nos dice que se ejecutaron las historias a tiempo.

Evolución del tablero Kanban Sprint 0

Las historias que necesitan ser probadas más adecuadamente se quena en la columna *Pruebas*, las cuales se van a probar a lo largo del siguiente sprint. Las historias que abarcan la configuración del sistema se pasan directamente a la columna de finalizado puesto que al ser operaciones esenciales se asume que funcionan puesto que la aplicación se ejecuta de manera correcta. La Tabla 31 muestra como progresaron las historias en el tablero Kanban.

Pendiente		En ejecución	Pruebas	Finalizado	
SRV02	MAP02			CNF02	
SRV03	APP02			CNF03	
SRV04	CONF05			CNF04	
SRV05	APP03			CNF06	
SRV06	APP04				
SRV07	APP05				
SRV08	APP06				
APP08	APP07				
APP09	SRV05				
SRV09					
MAP03					

Tabla 31: Estado del tablero Kanban Sprint 0.

Revisión y retrospectiva Sprint 0

Con este sprint el servidor y las configuraciones para el desarrollo quedan preparados para iniciar con la integración de nuevas características. El despliegue de la aplicación en el servidor se realizará de forma automática. Esta opción nos ahorra el trabajo de desplegar los cambios al servidor con cada nueva característica agregada.

5.2.2. Sprint 1: Levantamiento de la aplicación

Planificación del Sprint

Este primer sprint se va a enfocar a fabricar un primer prototipo. Como, por ejemplo, la estructura de la base de datos, las primeras rutas y una primera versión de la aplicación.

Con estas historias, al final tendremos una primera versión de la página cargada en una aplicación móvil.

Sprint backlog

La Tabla 32 describe las historias que se ejecutaron en el sprint 1 con sus respectivos puntos de esfuerzo.

Sprint 1	Título	Puntos
CNF05	Crear una base de datos en PostgreSQL para almacenar los datos de la aplicación	2
SRV02	Crear una aplicación base para iniciar el desarrollo.	3
SRV04	Implementar los modelos necesarios para levantar la aplicación con su respectiva correlación a la base de datos.	3
MAP02	Crear un mapa de prueba para iniciar el desarrollo de la aplicación.	3
SRV03	Implementar la vista para el mapa de prueba	1
APP02	Crear una aplicación base para iniciar el desarrollo.	3
Total		15

Tabla 32. Sprint backlog del sprint 1.

El total de puntos de este sprint es de 15 los cuales se reparten en las dos semanas de ejecución del sprint 1.

Descripción y ejecución de las historias

Descripción de la historia

La Tabla 33 muestra las características, prioridad y esfuerzo de la historia CNF05.

Identificador	Título	Prioridad	Esfuerzo
CNF05	Crear una base de datos en PostgreSQL para almacenar los datos de la aplicación	Alta	2
Descripción	Se debe implementar una base de datos que permita almacenar la información de los mapas y los lugares de los medimos con el fin de que estos sean accesibles.		
Criterios de Aceptación	<ul style="list-style-type: none"> - Se tiene un diseño para la base de datos. - Se tiene una estructura que permita agregar fácilmente más modelos mediante migraciones. - Se tiene las migraciones para cada uno de los modelos necesarios. 		

Tabla 33. Descripción de la historia CNF05.

Implementación de la historia:

Para el fin de mostrar los modelos se creó un modelo entidad relación en el cual se define las tablas que esta primera versión del sistema va a utilizar. El modelo se muestra en la Figura 14.

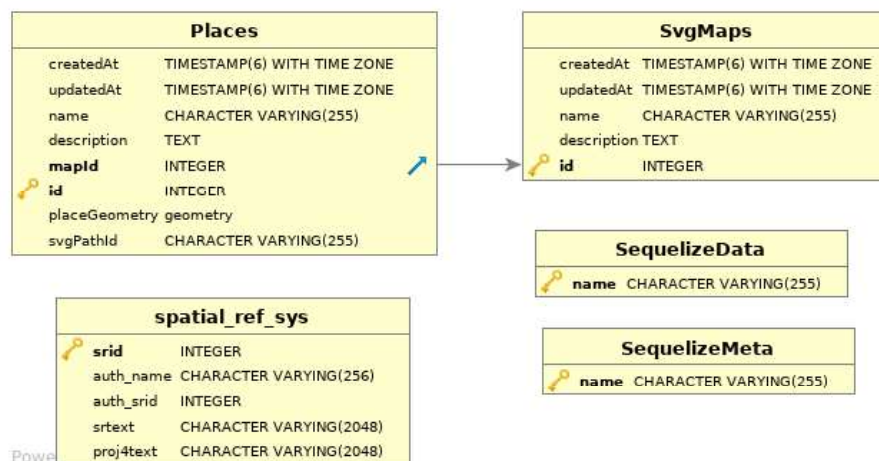


Figura 14. Diagrama relacional de la base de datos.

El sistema inicialmente utiliza estas dos tablas *Places* y *SvgMaps*. Estas dos tablas contienen la información para cada uno de los lugares y su respectivo mapa.

La tabla *spatial_ref_sys* es necesaria para el plugin PostGIS el cual es utilizado para las operaciones referentes a las coordenadas geográficas. Cabe aclarar que esta tabla es creada automáticamente al momento de instalar el plugin PostGIS.

Las tablas *SequelizeData* y *SequelizeMeta* son creadas automáticamente cuando se ejecuta las migraciones en la aplicación Node.js.

Las migraciones necesarias son las que se muestran en la Figura 15.

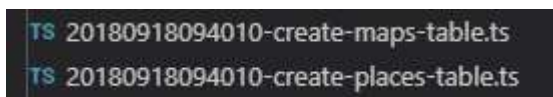


Figura 15. Archivos para la creación de tablas.

Cada una de estas migraciones tiene una estructura definida la cual se muestra en la Figura 16. Esta configuración está acorde a los campos especificados el diagrama.

```
module.exports = {
  up: function (queryInterface: QueryInterface, sequelize: SequelizeStatic) {
    return queryInterface.createTable(tableName, {
      createdAt: {
        type: sequelize.DATE
      },
      updatedAt: {
        type: sequelize.DATE
      },
      name: {
        type: sequelize.STRING
      },
      description: {
        type: sequelize.TEXT
      },
      id: {
        allowNull: false,
        autoIncrement: true,
        primaryKey: true,
        type: sequelize.INTEGER,
        unique: true
      },
    }).then(() => {
      return queryInterface.addIndex(tableName, ["name"]);
    });
  },
  down: function(queryInterface: QueryInterface) {
    return queryInterface.dropTable(tableName);
  }
};
```

Figura 16. Estructura del archivo para la creación de las tablas.

Cada una de estos archivos de migraciones crearan la tabla correspondiente y registrarán el nombre de la migración en la tabla *SequelizeMeta*.

Descripción de la historia

La Tabla 34 muestra las características, prioridad y esfuerzo de la historia SRV02.

Identificador	Título	Prioridad	Esfuerzo
SRV02	Crear una aplicación base para iniciar el desarrollo.	Alta	3
Descripción	Crear la arquitectura para acceder a las rutas.		
Criterios de Aceptación	<ul style="list-style-type: none">- Las rutas de la aplicación deben estar implantadas mediante Express.js- Se debe tener una página de inicio que permita listar los mapas disponibles.		

Tabla 34. Descripción de la historia SRV02.

Implementación de la historia

Para el fin de agregar las rutas adecuadas se implementó una estructura basada en el paquete Express.js el cual nos da las herramientas necesarias para dicho fin. En esta primera parte se va a definir una estructura que separa cada uno de los modelos en diferentes carpetas. La Figura 17 muestra las carpetas creadas.

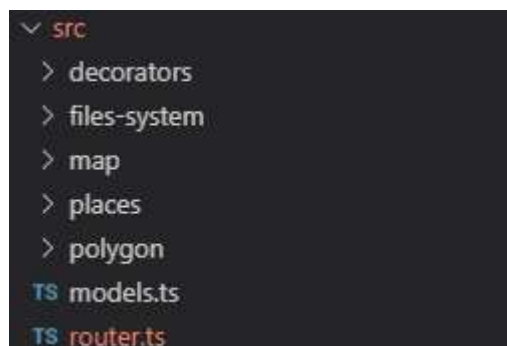


Figura 17. Estructura de los modelos.

Como aún no tenemos registrados los mapas vamos a tener una primera vista en blanco hasta que el mapa de prueba esté listo.

Descripción de la historia

La Tabla 35 muestra las características, prioridad y esfuerzo de la historia SVR04.

Identificador	Título	Prioridad	Esfuerzo
SVR04	Implementar los modelos necesarios para levantar la aplicación con su respectiva correlación a la base de datos.	Alta	2
Descripción	Los modelos son necesarios para almacenar		
Criterios de Aceptación	<ul style="list-style-type: none">- Para cada uno de los modelos que se tiene en la base de datos debe implementarse una estructura en la aplicación que permita su fácil manejo.- Se debe listar la cantidad de mapas en la página para comprobar que la pagina funciona.		

Tabla 35. Descripción de la historia SVR04.

Implementación de la historia

Como requerimiento de Sequelize se necesita un archivo Typescript que represente cada uno de los campos que la tabla en la base de datos tiene. Este modelo también tiene una representación de las conexiones con los demás modelos.

Con el fin de probar que el modelado funciona se creó una lista de los mapas que se encuentran en el sistema con el número de lugares que contiene cada uno de ellos, esto se puede ver en la Figura 18. El listado de lugares dentro del mapa comprueba que la relación entre mapa y lugar está funcionando de manera correcta.

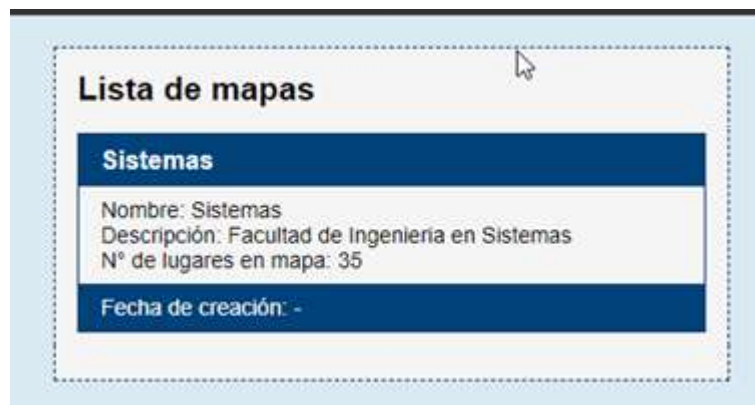


Figura 18. Lista de mapas.

Descripción de la historia

La Tabla 36 muestra las características, prioridad y esfuerzo de la historia MAP02.

Identificador	Título	Prioridad	Esfuerzo
MAP02	Crear un mapa de prueba para iniciar el desarrollo de la aplicación.	Alta	3
Descripción	Las pruebas unitarias permiten a la aplicación comprobar la integridad de la funcionalidad con el fin de evitar errores graves en la aplicación.		
Criterios de Aceptación	<ul style="list-style-type: none">- Crear una imagen SVG que represente el mapa de prueba necesario para la aplicación.- Crear un listado de los puntos que conforman cada uno de los polígonos que conforman el mapa.- Migrar los puntos de los mapas a la base de datos.		

Tabla 36. Descripción de la historia MAP02.

Implementación de la historia

Para la creación del mapa de prueba emplearemos QGis como herramienta de dibujo / calcado de un mapa pequeño siguiendo el perfil de Google Maps. Este mapa quedara almacenado en la geodatabase (PostgreSQL – PostGIS) el cual permite ejecutar operaciones desde el sistema, tanto como vista desde la aplicación, pruebas, cálculos, etc.

El resultado también permite tener una conversión de polígonos los cuales proveerán de información para la generación del archivo SVG. Esta conversión se la realiza al tener los puntos en formato JSON que generaran mediante funciones un polígono en la base de datos. En la Figura 19 podemos ver la interfaz de desarrollo de los mapas.

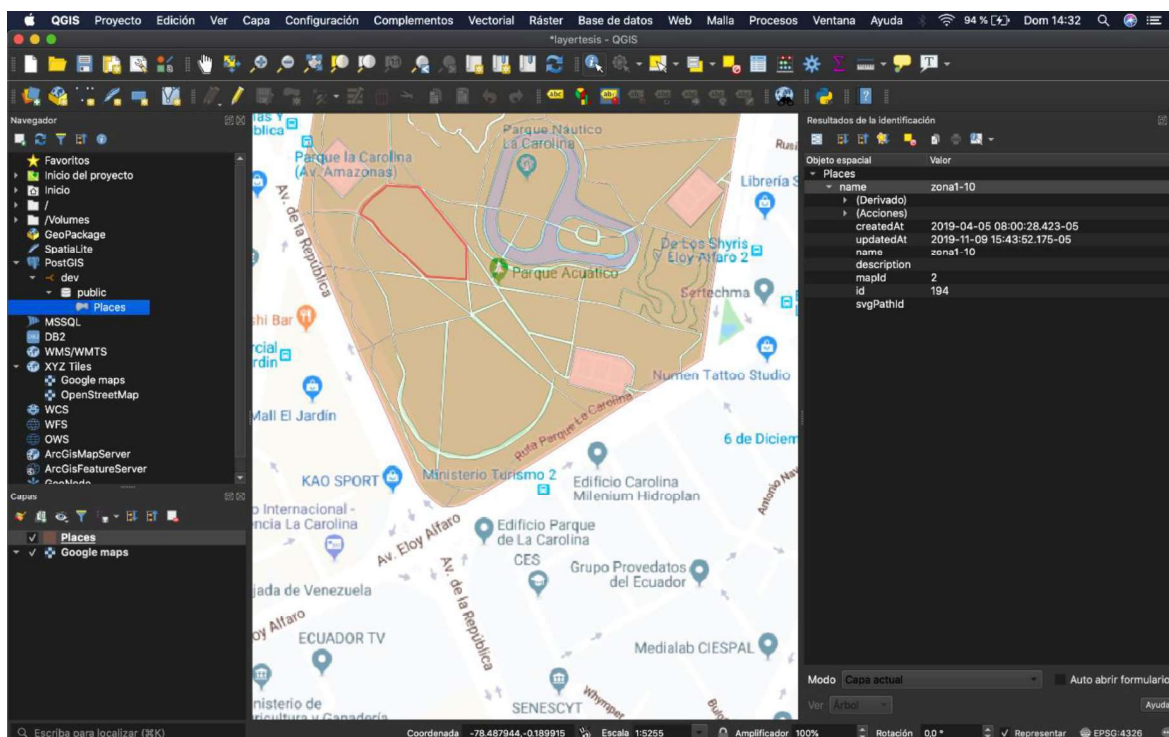


Figura 19. Creación de mapa utilizando QGIS.

Descripción de la historia

La Tabla 37 muestra las características, prioridad y esfuerzo de la historia SRV03.

Identificador	Título	Prioridad	Esfuerzo
SRV03	Implementar la vista para el mapa de prueba	1	SRV03
Descripción	Para que el mapa sea renderizado en la página web es necesario crear una ruta que redirija a una vista que contenga la imagen SVG.		
Criterios de Aceptación	<ul style="list-style-type: none"> - El mapa es mostrado en una página independiente. - El mapa tiene una ruta que contenga un identificador. 		

Tabla 37. Descripción de la historia SRV03.

Implementación de la historia

Se inicio con la creación de la ruta en el archivo *router.ts* en este vamos a definir el identificador del mapa como parte de la ruta. Adicionalmente como se ve en la Figura 20 se va a tener como parte de la ruta un conjunto de coordenadas las cuales se va a utilizar en futuras historias.

```
router.get("/mapa/:mapId", function (req, res) {  
  const lon = req.query.idx ? req.query.idx : "0";  
  const lat = req.query.idy ? req.query.idy : "0";  
  const zoom = req.query.zoom;  
  const mapId = req.params.mapId ? req.params.mapId : 0;  
}
```

Figura 20. lectura de los parámetros de la URL.

Con este identificador se llama a la base de datos, esta va a retornar la información del mapa y los lugares dentro de este. La Figura 21 muestra la página llamada por la ruta.

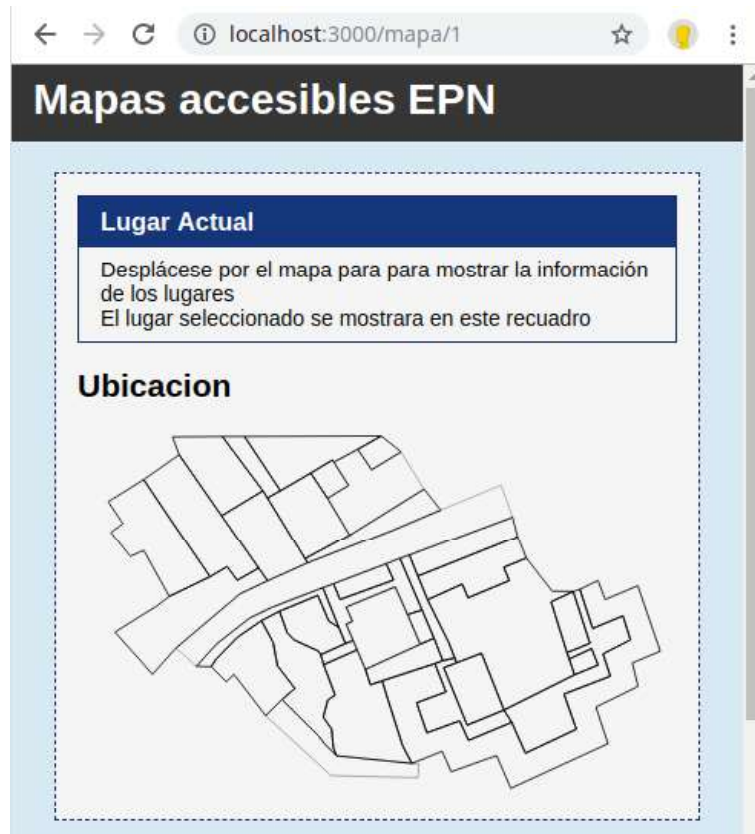


Figura 21. Vista del mapa de pruebas.

Dentro de la vista HTML se llama al SVG creado en el la historia MAP02 la cual va a contener una representación gráfica de los lugares en el mapa.

Descripción de la historia

La Tabla 38 muestra las características, prioridad y esfuerzo de la historia APP02.

Identificador	Título	Prioridad	Esfuerzo
APP02	Crear una aplicación base para iniciar el desarrollo.	Alta	3
Descripción	La aplicación móvil va a mostrar los mapas del servidor y va a permitir realizar operaciones con los mapas.		
Criterios de Aceptación	<ul style="list-style-type: none">- Crear una ampliación que simule un navegador web.- La aplicación muestra el listado de los mapas.		

Tabla 38. Descripción de la historia APP02.

Implementación de la historia

Se creó una aplicación con una actividad Android que tiene como elemento principal un WebView como se muestra en la Figura 22 que va a contener la página obtenida del servidor.



Figura 22. Versión inicial de la aplicación.

La WebView se va a llenar con los datos que se reciben como respuesta de la petición HTTP del servidor.

Burndown chart Sprint 1

Como podemos ver en la Figura 23 la ejecución de este sprint se llevó a cabo de manera que no se distancio mucho de lo estimado.

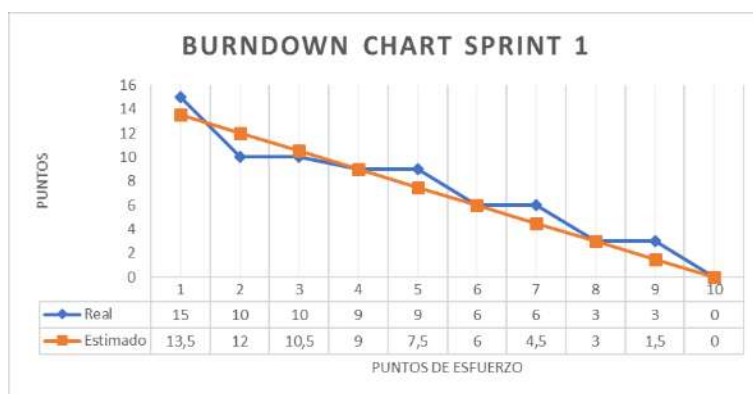


Figura 23. Tabla Burndown de la ejecución del sprint 1.

La línea de la estimación no se aleja significativamente de la ejecución real de el sprint. Lo cual nos dice que se ejecutaron las historias a tiempo.

Evolución del tablero Kanban Sprint 1

Las historias que necesitan ser probadas más adecuadamente se quena en la columna *Pruebas*, las cuales se van a probar a lo largo del siguiente sprint. Las historias que abarcan la configuración del sistema se pasan directamente a la columna de finalizado puesto que al ser operaciones esenciales se asume que funcionan puesto que la aplicación se ejecuta de manera correcta. La tabla 38 muestra como progresaron las historias en el tablero Kanban.

Pendiente		En ejecución	Pruebas	Finalizado	
SRV05	APP08		APP05	MAP02	CNF02
SRV06	APP09		APP06	APP02	CNF03
SRV07			SRV05	CONF05	CNF04
SRV08				APP03	CNF06
SRV09				APP04	SRV02
MAP03				APP07	SRV03
					SRV04

Tabla 39. Estado del tablero Kanban Sprint 1.

Revisión y retrospectiva Sprint 1

En el primer sprint se ejecutó de manera efectiva, la implementación de las historias se hizo dentro del tiempo de ejecución del sprint 2. Las historias que requieren ser probadas con más tiempo se las movió a la columna de pruebas en el tablero Kanban.

Con respecto a la ejecución del sprint 1 se debe mejorar la organización de las historias puesto que la primera selección no abarcaba las configuraciones necesarias para la ejecución de la aplicación en el servidor.

La cantidad de puntos de esfuerzo es el adecuado puesto que se pudo terminar la ejecución de historias al mismo tiempo que se terminó el tiempo de ejecución del sprint.

5.2.3. Sprint 2: Creación de la aplicación móvil

Planificación del Sprint

Este sprint se centró en la implementación de las funciones de la aplicación móvil por lo tanto se va a tomar las historias de la historia épica APP.

Al final de este sprint se espera tener la aplicación móvil usando las rutas de la aplicación del servidor.

Sprint backlog

La Tabla 40 describe las historias que se ejecutaron en el sprint 2 con sus respectivos puntos de esfuerzo.

Sprint 2	Titulo	Puntos
APP03	Obtener las coordenadas de la posición geográfica.	3
APP04	Implementar una función que llame al servidor remoto.	2
SRV05	Implementar la operación de detección de polígonos mediante los parámetros de coordenadas.	3
APP05	Implementar una base de datos en la aplicación para guardar las configuraciones.	3
APP06	Implementar la conexión para el uso del motor TTS del teléfono.	2
APP07	Mostrar la página con el SVG.	1
Total		14

Tabla 40. Sprint backlog del sprint 2.

El total de puntos de este sprint es de 6 los cuales se reparten en las dos semanas de ejecución del sprint 2.

Descripción y ejecución de las historias

Descripción de la historia

La Tabla 41 muestra las características, prioridad y esfuerzo de la historia APP07.

Identificador	Título	Prioridad	Esfuerzo
APP07	Mostrar la página con el SVG.	Alta	1
Descripción	Desde la aplicación se llama al servidor y este responderá con el mapa correspondiente. Este mapa se mostrará de manera correcta en la aplicación.		
Criterios de Aceptación	- La página se renderiza en la aplicación cuando se llama al servidor.		

Tabla 41. Descripción de la historia APP07.

Implementación de la historia

La aplicación debe llamar por defecto a la página del mapa, si la página del mapa no está disponible se debe mostrar la lista de mapas. El mapa se adapta al ancho de la pantalla como se puede ver en la Figura 24.

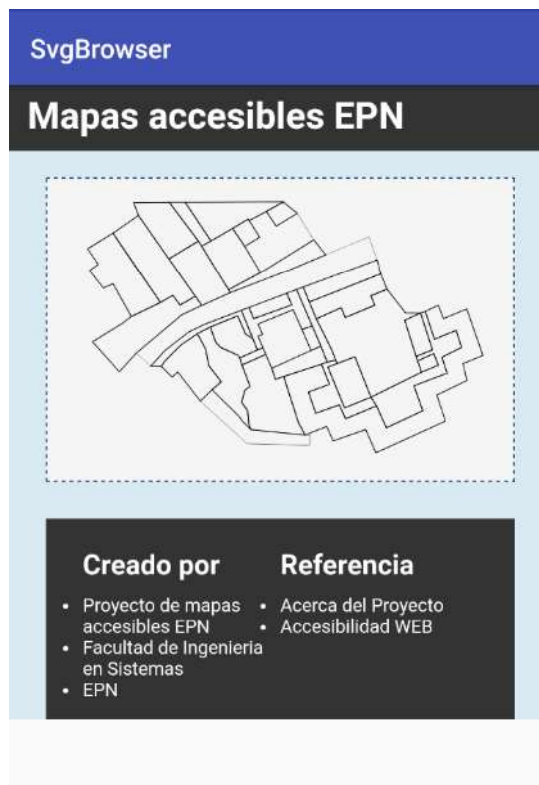


Figura 24. Página del mapa vista desde la aplicación.

Descripción de la historia

La Tabla 42 muestra las características, prioridad y esfuerzo de la historia APP05.

Identificador	Título	Prioridad	Esfuerzo
APP05	Implementar una base de datos en la aplicación para guardar los datos de la aplicación y sus configuraciones.	Alta	3
Descripción	Crear una base de datos local en el almacenamiento del dispositivo con el fin de persistir las configuraciones a utilizar y datos obtenidos del servidor.		
Criterios de Aceptación	- Crear una base de datos. - Crear una forma de acceder a la base de datos de manera eficiente.		

Tabla 42. Descripción de la historia APP05.

Implementación de la historia

La base de datos local de la aplicación está implantada para guardar etiquetas calve-valor con el fin de que los valores sean fácilmente accesible. En la Figura 25 se puede ver la base de datos básica de la aplicación.



Figura 25. Diagrama de relaciones de la base de datos del dispositivo.

La base de datos va a tener 2 tablas:

- dictionary: esta nos va a servir para almacenar los valores etiquetándolos con un identificador. El campo *name* va a almacenar la etiqueta de y los campos *int_value*, *double_value*, *string_value*, *float_value*, *boolean_value* van a almacenar el valor de esa etiqueta según qué tipo de dato sea.

- near_places: esta tabla va a almacenar los datos de los lugares cercanos con el fin de utilizarlos sin necesidad de volver a procesar la página que contiene el mapa.

Cada una de las bases de datos va a contener una serie de getters y setters.

Descripción de la historia

La Tabla 43 muestra las características, prioridad y esfuerzo de la historia APP03.

Identificador	Título	Prioridad	Esfuerzo
APP03	Obtener las coordenadas de la posición geográfica.	Media	3
Descripción	La aplicación móvil debe obtener las coordenadas del dispositivo.		
Criterios de Aceptación	<ul style="list-style-type: none"> - Implementar la obtención de las coordenadas a través de la aplicación. - Refrescar constantemente la ubicación con el fin de detectar cambios de ubicación. 		

Tabla 43. Descripción de la historia APP03.

Implementación de la historia

Para implementar la obtención de la ubicación del dispositivo se debe añadir en el manifiesto de la aplicación que se requiere esta característica del dispositivo. La Figura 26 muestra la parte del manifiesto de la ampliación que contiene los permisos necesarios.

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
```

Figura 26. Permisos de la aplicación.

La aplicación en este momento debe usar los siguientes permisos.

- ACCESS_FINE_LOCATION: Permite obtener la ubicación usando los servicios complementarios al GPS que permiten una localización más breve.
- ACCESS_COARSE_LOCATION: Permite obtener la ubicación mediante el uso de el sensor de GPS.
- INTERNET: Permite que la aplicación acceda a internet.
- ACCESS_WIFI_STATE: Permite saber si una conexión inalámbrica está disponible.

Con esto se implementa una función de obtención de la ubicación que se va a ejecutar de manera recursiva mediante un temporizador. Esta función como se ve en la Figura 27 debe almacenar la ubicación dentro de la base de datos local.

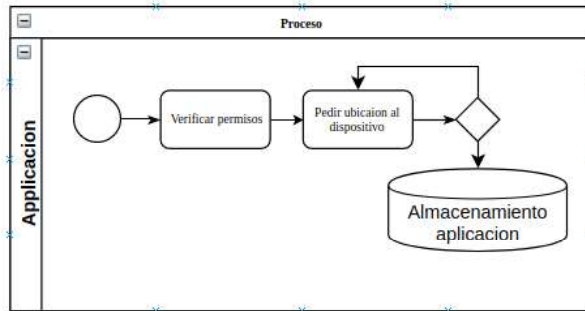


Figura 27. Flujo de trabajo de la aplicación.

Con para comprobar que esta función este correctamente implantada se añadió un toast como se ve en la Figura 28 para que la ubicación se muestre a lo largo del desarrollo.



Figura 28. Comprobación de coordenadas.

Para almacenar la ubicación de manera que esta sea obtenida en las diferentes operaciones se guarda en la tabla *diccionario* con las etiquetas CURRENT_LAT y CURRENT_LON.

Descripción de la historia

La Tabla 44 muestra las características, prioridad y esfuerzo de la historia SRV05.

Identificador	Titulo	Prioridad	Esfuerzo
SRV05	Implementar la operación de detección de polígonos mediante los parámetros de coordenadas.	Media	3
Descripción	Cada vez que la coordenada este dentro de un polígono este se resaltara para que la ubicación.		
Criterios de Aceptación	- Cuando se llame un mapa con la ubicación incluida se debe resaltar el polígono que cura esa ubicación.		

Tabla 44. Descripción de la historia SRV05.

Implementación de la historia

Las coordenadas ingresadas son extraídas desde la ruta. Estas coordenadas se incluyen en la operación Intersect del plugin PostGIS.

A la ruta de los mapas se agrega los siguientes parámetros como una consulta:

- idx: Parámetro que define la latitud de la ubicación.
- idy: Parámetro que define la longitud de la ubicación.

Así formamos una ruta parecida a la siguiente:

- (servidor)/mapa/1?idx=-0.21025674&idy=-78.4891124846

Y el servidor responde con la página que se muestra en la Figura 29. Puesto que se modifica el color del parámetro path que corresponde a la sección del dibujo.

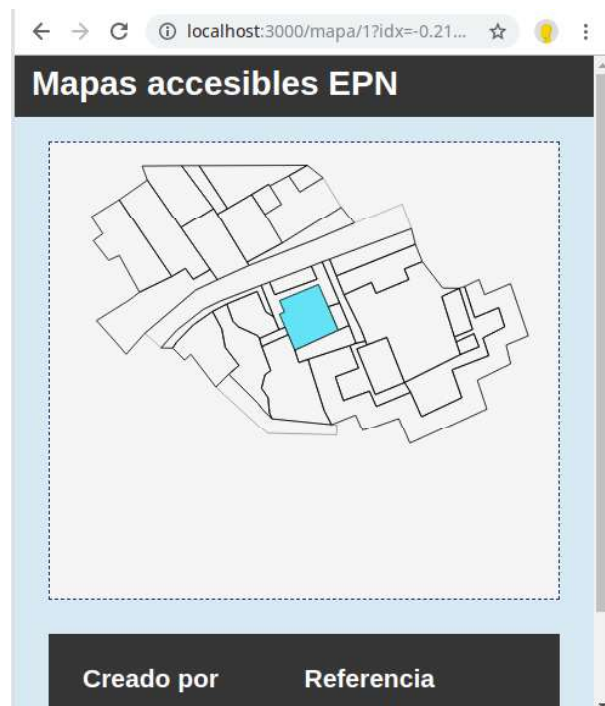


Figura 29. Cambio de color del lugar actual en el mapa.

Descripción de la historia

La Tabla 45 muestra las características, prioridad y esfuerzo de la historia APP04.

Identificador	Título	Prioridad	Esfuerzo
APP04	Implementar una función que llame al servidor remoto.	Alta	2
Descripción	Quiero que la aplicación llame de manera dinámica cuando la ubicación cambie.		
Criterios de Aceptación	<ul style="list-style-type: none"> - La aplicación debe llamar al servidor agregando los parámetros de ubicación. - La llamada al servidor se debe hacer cuando la ubicación cambie. 		

Tabla 45. Descripción de la historia APP04.

Implementación de la historia

Para que agregar la ubicación a la URL se crea una función que toma los datos de la ubicación. La Figura 30 muestra dicha función.

```
public static String getMapUrl(int map, double latitude, double longitude){  
    String base=getBaseUrl();  
    return base+"/mapa/"+map+"?idx="+latitude+"&idy="+longitude;  
}
```

Figura 30. Función para el armado del URL correcto en la aplicación.

Esta función va a ser llamada por la operación recursiva que va a manejar la actualización de la ubicación. Esto cambia ligeramente el comportamiento anterior haciendo que la aplicación ahora solo llame al servidor cuando la ubicación cambie. La Figura 31 muestra cómo se modificó la lógica de la función.

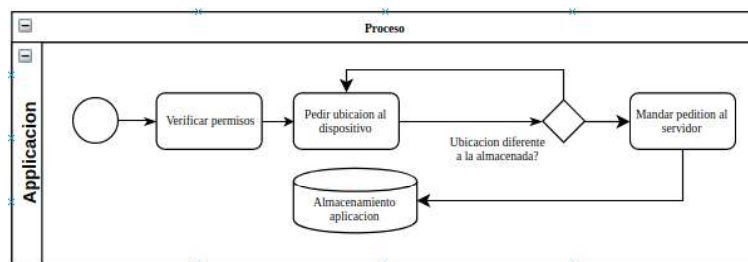


Figura 31. Flujo de trabajo de la aplicación.

Con este cambio al llamar a la ruta que acepta los parámetros de ubicación tenemos que el polígono que representa el lugar se va a resaltar. Y cada vez que la ubicación caiga en dentro de un polígono que este dentro del mapa este se resaltara como se ve en la Figura 32.



Figura 32. Resaltado del lugar en la aplicación.

Descripción de la historia

La Tabla 46 muestra las características, prioridad y esfuerzo de la historia APP06.

Identificador	Titulo	Prioridad	Esfuerzo
APP06	Implementar la conexión para el uso del motor TTS del teléfono.	Media	2
Descripción	La aplicación debe usar el motor TTS para informar al usuario los mensajes necesarios.		
Criterios de Aceptación	<ul style="list-style-type: none"> - Detallar la configuración del motor TTS en los teléfonos Android. - Implementar un mecanismo que permita encolar mensajes para que el motor TTS los interprete. 		

Tabla 46. Descripción de la historia APP06.

Implementación de la historia

Primero se debe realizar la instalación de los paquetes de voz en el dispositivo. Esto se realiza mediante las configuraciones de accesibilidad. En la Figura 33 se muestra cómo se debe ver la configuración al terminar de instalar dichos paquetes de voz.

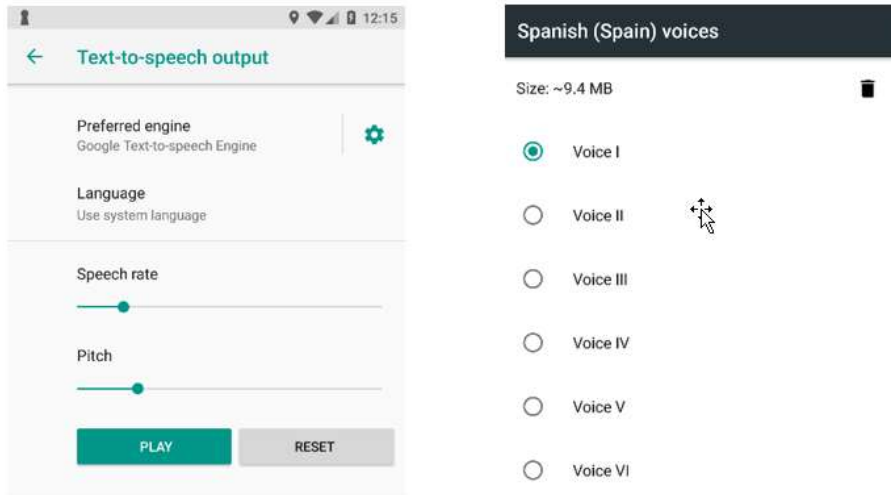


Figura 33. Configuración motora TTS.

Para el proceso de los mensajes se creó la clase *TextToSpeechManager* la cual contiene una lista de los mensajes a procesar y estos son ejecutados secuencialmente hasta vaciar la cola.

Burndown chart Sprint 2

Como se muestra en la Figura 34 la ejecución de este sprint se llevó a cabo de manera que no se distancio mucho de lo estimado.

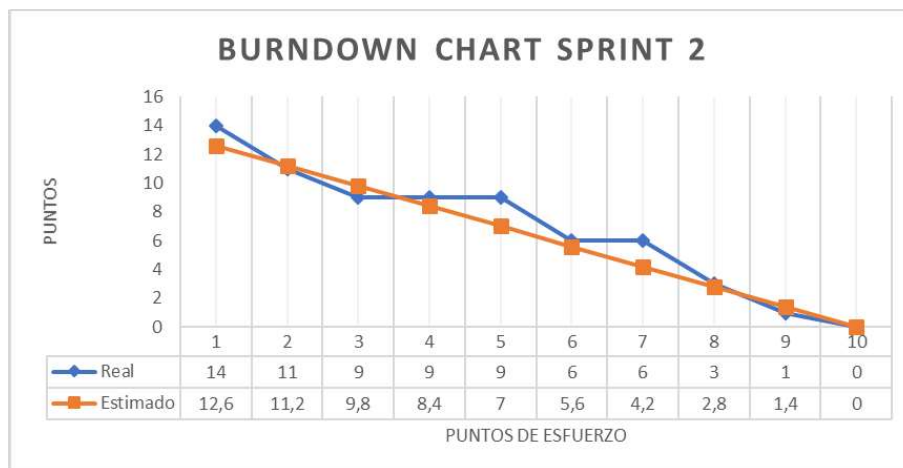


Figura 34. Tabla Burndown de la ejecución del sprint 2.

La línea de la estimación no se aleja significativamente de la ejecución real de el sprint. Lo cual nos dice que se ejecutaron las historias a tiempo.

Evolución del tablero Kanban

Las historias que necesitan ser probadas más adecuadamente se quena en la columna *Pruebas*, las cuales se van a probar a lo largo del siguiente sprint. Las historias que abarcan la configuración del sistema se pasan directamente a la columna de finalizado puesto que al ser operaciones esenciales se asume que funcionan puesto que la aplicación se ejecuta de manera correcta. La Tabla 47 muestra como evolucionaron las historias en el tablero Kanban.

Pendiente		En ejecución	Pruebas	Finalizado	
SRV05	APP03		SRV02	MAP02	CNF02
SRV06	APP04		SRV03	APP02	CNF03
SRV07	APP05		SRV04	CONF05	CNF04
SRV08	APP06				CNF06
SRV09	APP07				
MAP03	APP08				
	APP09				

Tabla 47. Estado del tablero Kanban Sprint 2.

Revisión y retrospectiva Sprint 2

En el segundo sprint se ha ejecutado de manera relativamente efectiva puesto que no hubo desviaciones graves con respecto a la estimación inicial. La implementación de las historias se hizo dentro del tiempo de ejecución del sprint 2. Las historias que requieren ser probadas se las movió a la columna de pruebas en el tablero Kanban.

5.2.4. Sprint 3: Interacción de la aplicación

Planificación del Sprint

Este tercer sprint se va a enfocar a la interacción de la aplicación móvil con el servidor de datos. Para esto se va a ejecutar las historias que tengan que ver con la obtención de los lugares cercanos a la ubicación del usuario.

Con estas historias, al final tendremos una primera versión de la página cargada en una aplicación móvil.

Sprint backlog

La Tabla 48 describe las historias que se ejecutaron en el sprint 3 con sus respectivos puntos de esfuerzo.

Sprint 3	Titulo	Puntos
SRV06	Listar los lugares cercanos e introducir la información de los lugares en el SVG	3
APP08	Implementar la navegación de los lugares cercanos mediante los botones de volumen.	3
APP09	Mostrar mediante mensajes de audio los lugares cercanos.	3
SRV06	Resaltar el lugar seleccionado en la imagen SVG	2
Total		11

Tabla 48. Sprint backlog del sprint 3.

Descripción y ejecución de las historias

Descripción de la historia

La Tabla 49 muestra las características, prioridad y esfuerzo de la historia SRV06.

Identificador	Titulo	Prioridad	Esfuerzo
SRV06	Listar los lugares cercanos	Media	3
Descripción	Es necesario una función que permita el listado de los lugares cercanos a la ubicación del usuario, para que esta información sea incluida en la imagen SVG a modo de meta data.		
Criterios de Aceptación	<ul style="list-style-type: none">- Crear un método que mida la distancia entre un punto o polígono con respecto a un polígono almacenado.- Crear un método que liste los lugares cercanos a una ubicación.- Agregar la información a la imagen SVG en el tag de meta data.		

Tabla 49. Descripción de la historia SRV06.

Implementación de la historia

Con el fin de probar el correcto funcionamiento de este método como se ve en la Figura 35 se implementó un endpoint en la API del servidor de forma temporal, esta nos va a devolver los lugares cercanos a la coordenada que se incluya en el URL.



Figura 35. respuesta de la prueba de los lugares cercanos.

Para incluir esta información en la imagen SVG nos vamos a guiar en el formato explicado en la sección 3.4.2 la cual nos va a permitir una estructura más acorde a lo que un intérprete de página espera. La Figura 36 muestra la metadata del SVG.

```
<metadata
  id="metadata5">
  <rdf:ABOUT>
    <cc:place rdf:about=facultad-sistemas>
      <dc:id>320</dc:id>
      <dc:distance>0</dc:distance>
      <dc:nombre>facultad-sistemas</dc:nombre>
      <dc:descripcion></dc:descripcion>
      <dc:svgid>facultad-sistemas</dc:svgid>
    </cc:place>
    <cc:place rdf:about=salida-sistemas>
      <dc:id>322</dc:id>
      <dc:distance>7.91759474881736</dc:distance>
      <dc:nombre>pasillo-medio</dc:nombre>
      <dc:descripcion></dc:descripcion>
      <dc:svgid>salida-sistemas</dc:svgid>
    </cc:place>
```

Figura 36. Metadata del SVG.

Con esto la aplicación puede describir al usuario la lista de los lugares cercanos con su respectiva distancia.

Descripción de la historia

La Tabla 50 muestra las características, prioridad y esfuerzo de la historia APP08.

Identificador	Título	Prioridad	Esfuerzo
APP08	Implementar la navegación de los lugares cercanos mediante los botones de volumen.	Alta	3
Descripción	La interacción con el mapa es una de las funciones principales de la aplicación por lo tanto es necesario implementar una forma de movernos en el mapa dibujado.		
Criterios de Aceptación	<ul style="list-style-type: none"> - Implementar una función que capture las teclas de volumen del teléfono celular y muestre los lugares que se van a usar como referencia - Mostrar el cambio de lugar en el recuadro que se encuentra en la parte superior del mapa. 		

Tabla 50. Descripción de la historia APP08.

Implementación de la historia.

La API de la plataforma Android provee una forma de detectar el uso de las teclas con las que cuenta el dispositivo. Por lo tanto, debemos crear una función que permita que la aplicación llame al lugar marcado como siguiente en la base de datos local. La Figura 37 muestra las funciones que capturan las funciones de los botones.

```

public boolean onKeyDown(int keyCode, KeyEvent event) {
    boolean active = db.getBooleanValue(CustomConst.ActiveRecursive);
    if ((keyCode == KeyEvent.KEYCODE_VOLUME_DOWN)){
        operaciones.loadNextPevLocation( operation: 2);
    }
    if ((keyCode == KeyEvent.KEYCODE_VOLUME_UP)){
        operaciones.loadNextPevLocation( operation: 1);
    }
    return true;
}

```

Figura 37. Uso de los botones para la interacción de los lugares.

Esto actualizará el estado de los lugares y proveerá la información del lugar actual, la cual se mostrará en el recuadro superior como se puede ver en la Figura 38.

Para que el recuadro muestre la información enviada desde el la aplicación es necesario la creación de una función en JavaScript que ejecute este proceso de actualización, a esta

función dentro de aplicación la marcaremos como @JavascriptInterface. Así el teléfono buscara esta función dentro del archivo JavaScript de la página.



Figura 38. Cambio de lugar con las teclas de volumen.

En la parte izquierda de la Figura 38 podemos ver que la aplicación antes de presionar la tecla de subir volumen, a la izquierda podemos ver la aplicación después de presionar la tecla. Como se muestra la información se ha actualizado.

Descripción de la historia

La Tabla 51 muestra las características, prioridad y esfuerzo de la historia APP09.

Identificador	Titulo	Prioridad	Esfuerzo
APP09	Mostrar mediante mensajes de audio los lugares cercanos.	Alta	3
Descripción	La aplicación debe mostrar los lugares cercanos mediante el motor TTS del teléfono.		
Criterios de Aceptación	- Implementar una función que complemente al listado de lugares con el motor TTS del teléfono.		

Tabla 51. Descripción de la historia APP09.

Implementación de la historia

En la historia APP06 se llevó a cabo la configuración dentro de la aplicación para el uso del motor de texto a voz del dispositivo móvil. En esta historia se implementó una función que al momento de presionar las teclas de volumen. La Figura 39 muestra los textos generados por la creación de los mensajes auditivos de la aplicación.

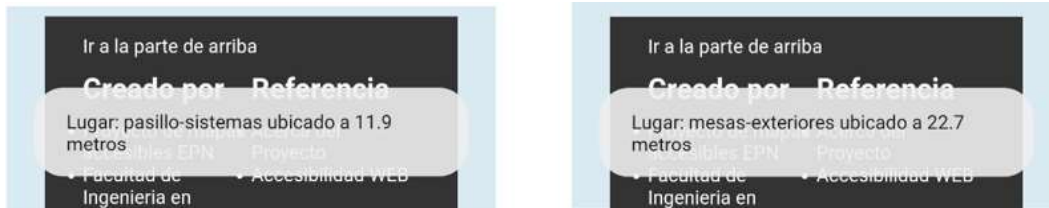


Figura 39. Respuesta auditiva de la aplicación.

Como se mostró función implementada en APP06 muestra un breve toast con la información que se presenta mediante la salida de voz.

Descripción de la historia

La Tabla 52 muestra las características, prioridad y esfuerzo de la historia SRV06.

Identificador	Título	Prioridad	Esfuerzo
SRV06	Resaltar el lugar seleccionado en a imagen SVG	Media	2
Descripción	La aplicación debe resaltar el lugar correspondiente a la información en el recuadro de dialogo en el SVG.		
Criterios de Aceptación	- Implementar una función que resalte el lugar mostrado en el recuadro superior en el SVG.		

Tabla 52. Descripción de la historia SRV06.

Implementación de la historia

De una manera similar a la historia que modifica el recuadro superior del mapa vamos a poder una función que modifique el color de un path de la imagen SVG. Esta parte de la imagen o path está marcado por la etiqueta svgId la cual es única para cada una de las partes de la imagen. Por lo tanto, solo debemos enlazar la información que la aplicación manda con el mencionado Id.

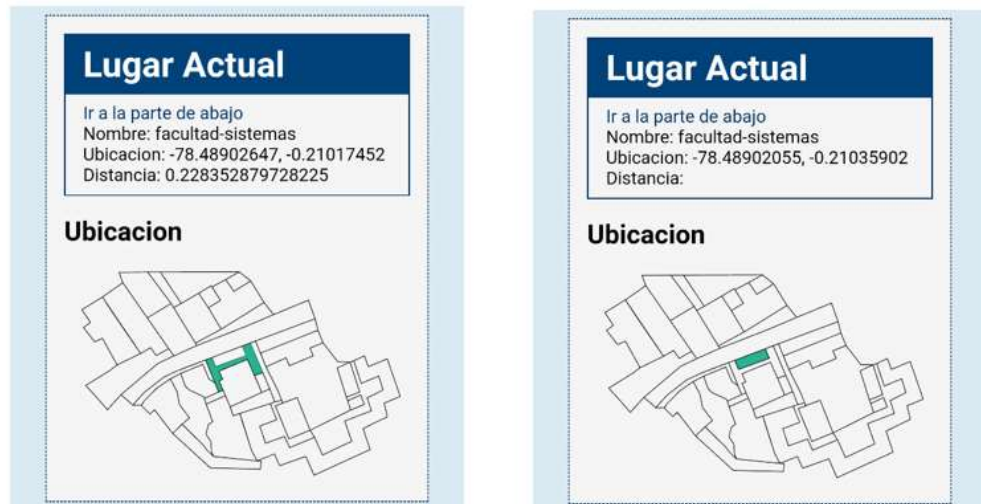


Figura 40. Cambio del resaltado del mapa mediante las teclas de volumen.

Como se muestra en la Figura 40 ver la función reacciona de forma correcta al momento de pulsar las teclas de volumen.

Burndown chart Sprint 3

Como podemos ver en la Figura 41 la ejecución de este sprint se llevó a cabo de manera que no se distancio mucho de lo estimado.

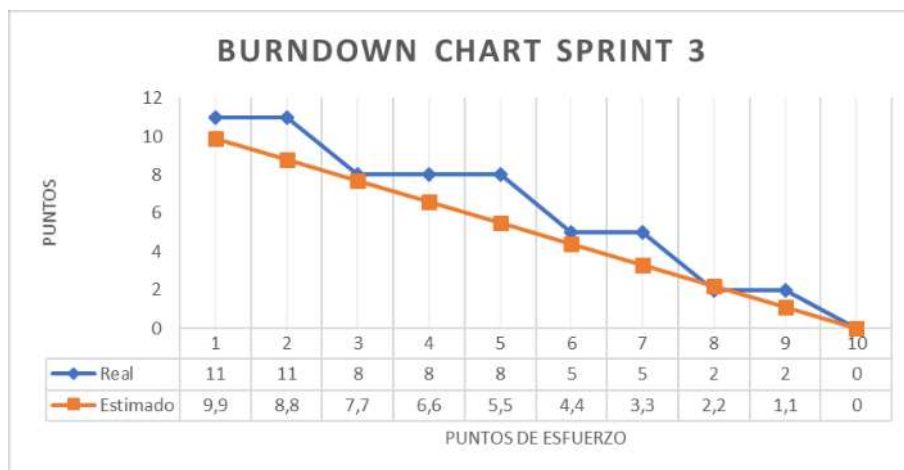


Figura 41. Tabla Burndown de la ejecución del sprint 3.

La línea de la estimación en la primera mitad del sprint se sitúa por debajo de la de ejecución. Lo cual nos dice que se quizá las historias se sobreestimaron y se las acabo antes. Pero luego la ejecución se iguala con la estimación lo cual nivela el esfuerzo del sprint.

Evolución del tablero Kanban

Las historias que necesitan ser probadas más adecuadamente se quena en la columna *Pruebas*. Las historias del sprint 2 se pasan a la columna de finalizado puesto que se ha probado su funcionamiento durante la ejecución del sprint 3. La Tabla 53 muestra como evolucionaron las historias en el tablero Kanban.

Pendiente		En ejecución	Pruebas	Finalizado	
MAP03	APP08		SRV06	MAP02	CNF02
	APP09		APP08	APP02	CNF03
			APP09	CONF05	CNF04
			SRV07	APP03	CNF06
				APP04	SRV02
				APP07	SRV03
				APP05	
				APP06	
				SRV05	

Tabla 53. Estado del tablero Kanban Sprint 3.

Revisión y retrospectiva Sprint 3

La implementación de las historias se hizo dentro del tiempo de ejecución del sprint 3. Las historias que requieren ser probadas se las movió a la columna de pruebas en el tablero Kanban las cuales se van a probar durante la ejecución del sprint 4.

5.2.5. Sprint 4: Funciones adicionales

Planificación del Sprint

Este sprint se va a enfocar en implementar las características finales de la aplicación. Adicionalmente se corrigieron errores que se presentaron en las pruebas de las historias del sprint pasado.

Sprint backlog

La Tabla 54 describe las historias que se ejecutaron en el sprint 4 con sus respectivos puntos de esfuerzo.

Sprint 4	Título	Puntos
SRV08	Implementar una función que me señale el lugar más cercano al mapa cuando estoy fuera de él.	3
MAP03	Crear un mapa de las cercanías de la Fundación Procodis.	3
APP10	Implementar la navegación en el listado de mapas mediante las teclas de volumen.	2
Total		8

Tabla 54. Sprint backlog del sprint 4.

El total de puntos de este sprint es de 8 los cuales se reparten en las dos semanas de ejecución del sprint 4.

Descripción y ejecución de las historias

Descripción de la historia

La Tabla 55 muestra las características, prioridad y esfuerzo de la historia SRV08.

Identificador	Título	Prioridad	Esfuerzo
SRV08	Implementar una función que me señale el lugar más cercano al mapa cuando estoy fuera de él.	SRV08	3
Descripción	Con el fin de evitar errores al momento de ingresar a un mapa se debe señalar un lugar de inicio.		
Criterios de Aceptación	<ul style="list-style-type: none">- Recibir las coordenadas en los parámetros de la selección de mapas.- Buscar el lugar más cercano a esas coordenadas y marcarlo.		

Tabla 55. Descripción de la historia SRV08.

Implementación de la historia

Con la información de las coordenadas que se encuentran en la URL podemos averiguar el lugar más cercano al punto donde se encuentra el usuario con respecto al mapa que el usuario selecciona.

En este ejemplo que se ve en la Figura 42 se va a utilizar las coordenadas del CEC el cual no está dentro del SVG de la politécnica.

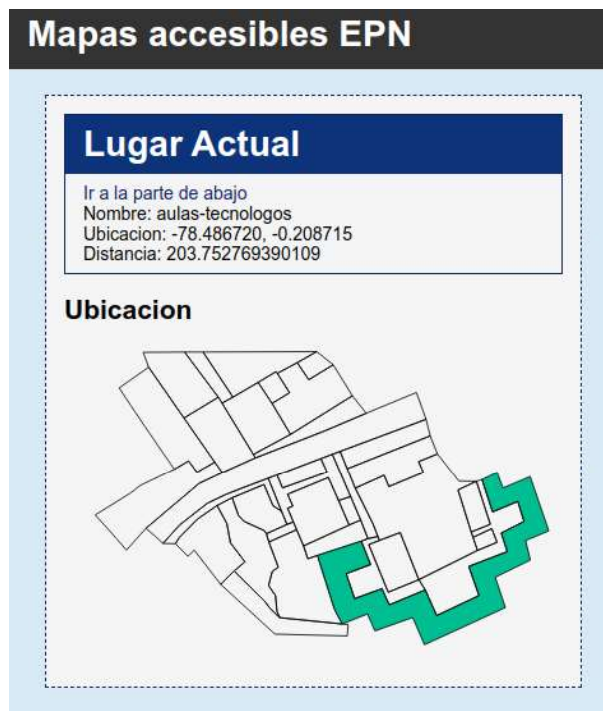


Figura 42. Vista del mapa vista desde el CEC.

El mapa señala que el lugar más cercano es las aulas de tecnológicos y que se ubica a aproximadamente 203 metros.

Descripción de la historia

La Tabla 56 muestra las características, prioridad y esfuerzo de la historia MAP03.

Identificador	Titulo	Prioridad	Esfuerzo
MAP03	Crear un mapa de las cercanías de la Fundación Procodis.	Alta	3
Descripción	Es necesario un mapa con la información de los lugares cercanos a la Fundación Procodis.		
Criterios de Aceptación	- Mostrar un mapa adicional con la información de la fundación Procodis.		

Tabla 56. Descripción de la historia MAP03.

Implementación de la historia

Siguiendo el mismo proceso que se usó para la creación del mapa de pruebas de la politécnica, se creó una página que muestre una representación en SVG de los lugares cercanos de la fundación Procodis. La Figura 43 muestra la página con el SVG creado.



Figura 43. Vista del mapa de los alrededores de Procodis.

Adicionalmente se creó la información que enlaza los identificadores de cada una de las partes del SVG con su correspondiente en la base de datos. Lo cual nos permite ejecutar las operaciones de resaltado en la imagen.

Descripción de la historia

La Tabla 57 muestra las características, prioridad y esfuerzo de la historia APP10.

Identificador	Título	Prioridad	Esfuerzo
APP10	Implementar la navegación en el listado de mapas mediante las teclas de volumen.	Media	2
Descripción	El listado de mapas debe integrar una navegación de la misma manera que los mapas por medio de las teclas.		
Criterios de Aceptación	- Implementar una navegación que permita señalar el mapa del cual se quiera obtener la información.		

Tabla 57. Descripción de la historia APP10.

Implementación de la historia

Por medio de la combinación de funciones JS y de funciones dentro de la aplicación móvil se resaltó la cabecera del mapa que se señale con la tecla de bajar volumen. La Figura 44 muestra cómo se resalta el enlace al mapa.

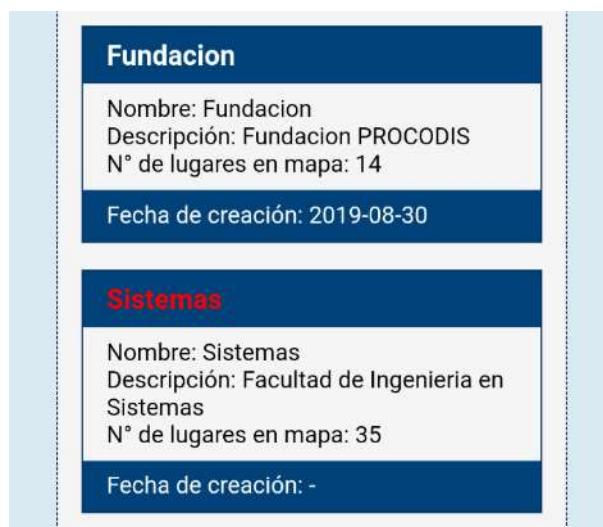


Figura 44. Navegación en la pantalla de inicio.

Si se presiona la tecla de subir volumen en el mapa señalado la aplicación nos mostrara la pantalla con el mapa. Adicionalmente como se implementó en la historia SRV08 se marcará el lugar más cercano a la ubicación actual del usuario.

Burndown chart Sprint 4

Como podemos ver en la Figura 45 la ejecución de este sprint se llevó a cabo de manera que se distanció un poco de lo estimado debido a que son pocas historias y la gráfica no se puede acercar tanto.

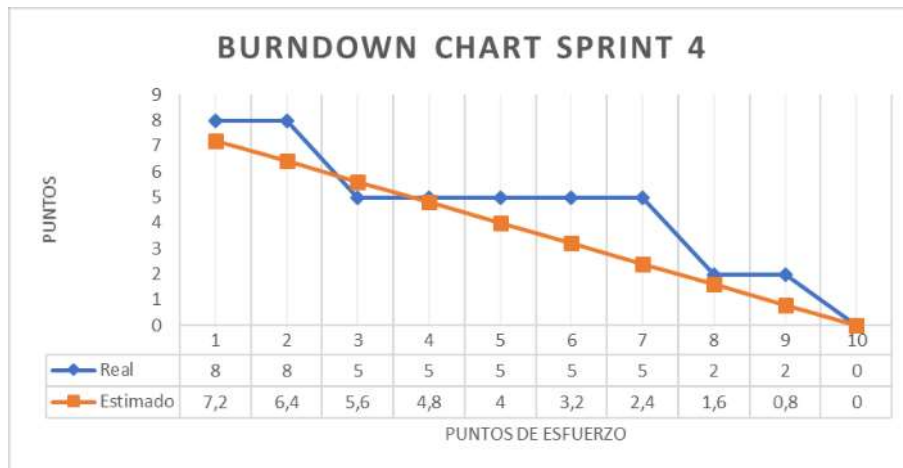


Figura 45. Tabla Burndown de la ejecución del sprint 4.

Evolución del tablero Kanban

Las historias cuyas pruebas se heredaron del sprint 3 se lograron probar dentro del este Sprint, adicionalmente de corregir ciertos errores en las mismas. La Tabla 58 muestra como evolucionaron las historias en el tablero Kanban.

Pendiente	En ejecución	Pruebas	Finalizado
		SRV08	MAP02 CNF02
		MAP03	APP02 CNF03
		APP10	CONF05 CNF04
			APP03 CNF06
			APP04 SRV02
			APP07 SRV03
			APP05 SRV06
			APP06 SRV07
			SRV05

Tabla 58. Estado del tablero Kanban Sprint 4.

Revisión y retrospectiva Sprint 4

Este último Sprint al tener que cerrar con la ejecución del proyecto se encargó de solucionar ciertos errores que surgieron a partir de la implementación de las historias del Sprint 3. Con el fin de poder probar las historias de el sprint 4 antes de comenzar con las pruebas de

accesibilidad se llevará a cabo una separación de una semana entre estas dos actividades. Con el fin de asegurar que la aplicación funcione correctamente.

5.3. Funcionalidad de la aplicación

Al finalizar la ejecución de los sprint se llevó a cabo una comprobación de la funcionalidad previo a la ejecución de las pruebas.

El primer escenario de la aplicación es con el usuario dentro de un mapa y explorando los lugares que están cerca. Al iniciar la aplicación esta se ubicará dentro del lugar que corresponde a las coordenadas. Como se muestra en la Figura 46 se puede ver que se resalta el lugar y la información del lugar se muestra en el cuadro de información, así como también se mostrara la información del lugar a través de la interfaz de audio del dispositivo.

Si el usuario se empieza a mover por el mapa la aplicación actualizara el lugar actual. En la Figura 47 se muestra el cambio al nuevo lugar que la aplicación muestra cuando la ubicación se actualiza.



Figura 46: Mapa inicial de la aplicación.

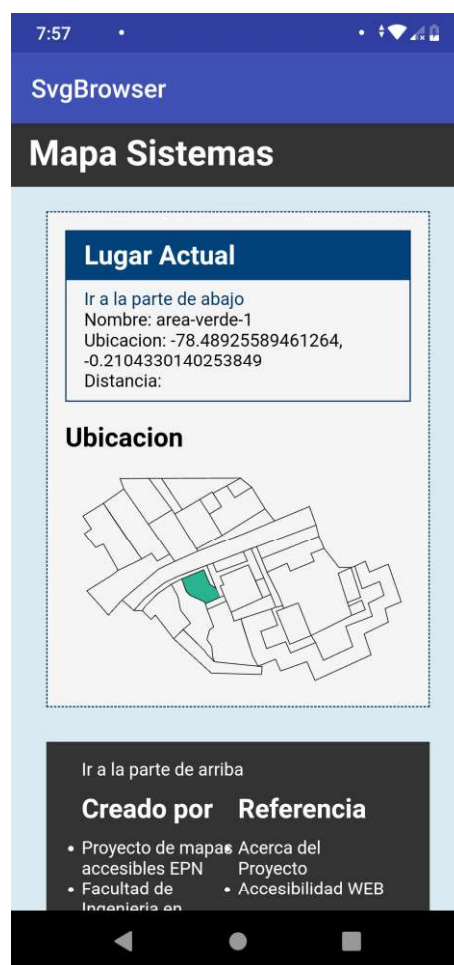


Figura 47: Actualización del lugar

Una vez iniciado el mapa el usuario puede empezar a navegar por los lugares cercanos con las teclas de volumen. En la Figura 48 se visualiza como la aplicación se comporta al pulsar el botón de subir el volumen. Adicionalmente se muestra un mensaje a través de la interfaz de audio con la distancia del lugar.

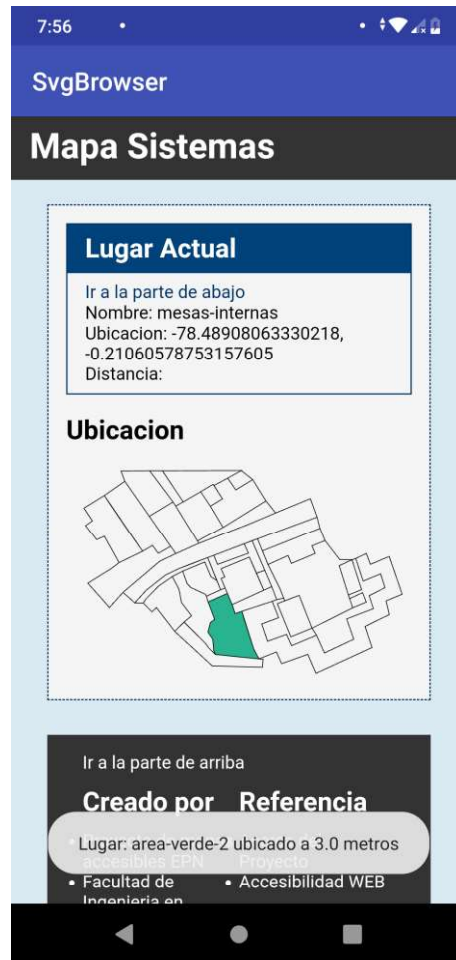


Figura 48: Lugar más cercano al usuario dentro del mapa

Otro caso que se puede dar es cuando el usuario no se encuentra dentro de un mapa preestablecido. En este caso la aplicación mostrara la lista de mapas disponibles para consultar la información. En la Figura 49 podemos ver la lista de mapas que la aplicación.

Si el usuario presiona el botón de bajar el volumen se empezará a seleccionar cada uno de los mapas. Como podemos ver en la Figura 50 se resalta el enlace al mapa al presionar la tecla de bajar volumen.



Figura 49: Lista de mapas



Figura 50: Selección del mapa.

Si se presiona la tecla de subir volumen la aplicación va a ingresar al mapa que se tiene seleccionado. El mapa muestra la información del lugar más cercano al usuario y la distancia con respecto al mismo. En la Figura 51 se muestra como la aplicación muestra la distancia. Si se navega de la misma forma que en caso anterior la aplicación mostrara la distancia de los lugares con respecto al punto más cercano al usuario dentro del mapa como se puede ver en la Figura 52 .

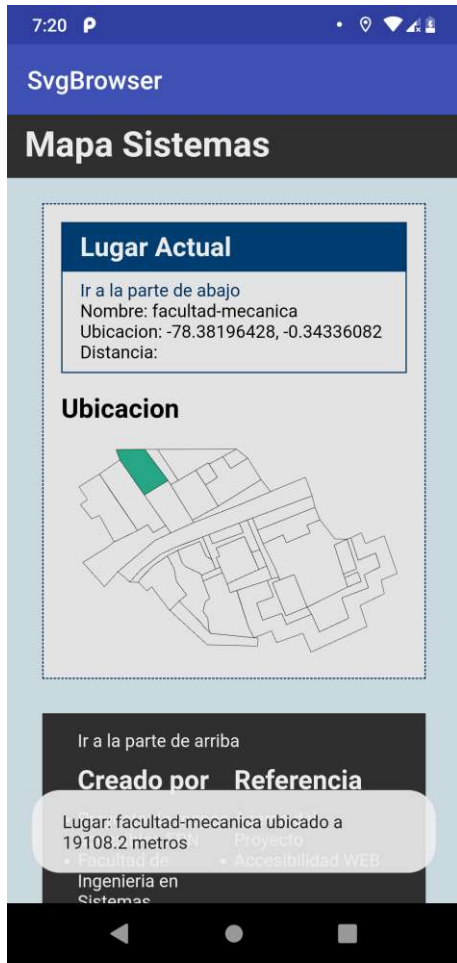


Figura 51: Lugar más cercano al usuario al iniciar el mapa.

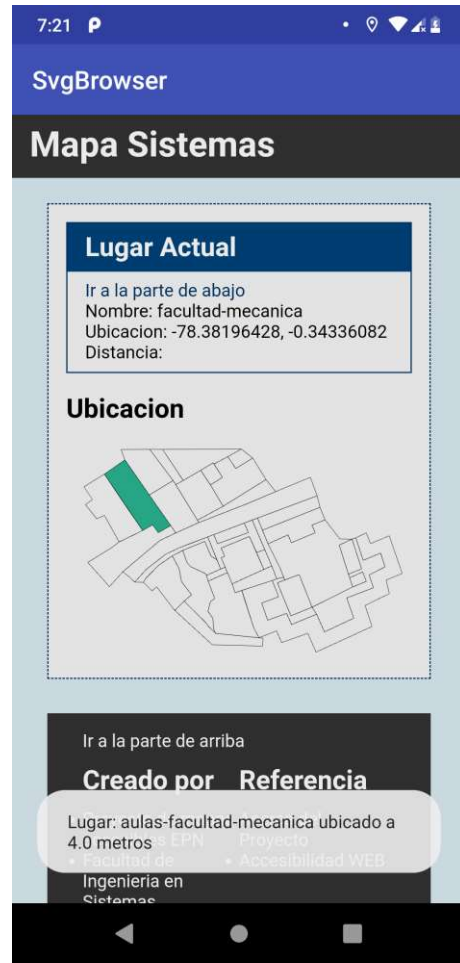


Figura 52: Lugar más cercano relativo

Con esta ejecución previa se puede ver como la aplicación se comporta en los escenarios previstos. Una comprobación más profunda de la aplicación se aborda en la sección de pruebas.

6. Evaluación de la aplicación

Para garantizar que la aplicación cumpla con los aspectos que rigen la W3C, se realizaron evaluaciones con herramientas en línea como: WAVE, EXAMINATOR, PowerMapper. Estas herramientas entregan reportes y sugerencias (en formato de código y/o líneas de código marcadas) de arreglos para cumplir con un mínimo de accesibilidad.

6.1. Planeación de las pruebas.

Para la evaluación de la aplicación y del servidor se ejecuto las pruebas en dos fases:

Pruebas de Accesibilidad web automatizadas, en esta fase empelaremos software de terceros (WAVE, EXAMINATOR, PowerMapper), en modalidad web, con el cual mediremos el estado del sistema. Estas herramientas fueron seleccionadas ya que sus resultados presentan alta confiabilidad en cuanto a análisis de sitios web, además de que son muy aclamados en cuanto análisis de calidad de software se refiere. Para este proceso, procedemos a ingresar la URL de nuestro sistema en las paginas de evaluación y con los resultados, realizaremos las correcciones pertinentes con el fin de mejorar nuestra calificación. Con esto buscamos garantizar la usabilidad y accesibilidad de nuestro sistema.

Pruebas de usuario, en esta fase nos ayudaremos de usuarios no videntes, los cuales usaran la aplicación de forma asistida por nosotros, para la debida explicación de uso del mismo. El primer escenario a emplear será el parque de la carolina con la ayuda del equipo de ciclistas no videntes, y con el uso de un dispositivo móvil con Android, procederemos a caminar por varios lugares del parque a modo que la aplicación le muestre los resultados. Como segundo escenario está el mapa de los alrededores de la fundación, el cual contara con la ayuda del director de la fundación. Como escenario adicional, buscaremos la opinión de un experto en el tema de movilidad de personas no videntes, que para el caso es Hugo Arias. Al final de estas pruebas esperamos tener los comentarios finales de los usuarios a modo de retroalimentación. Con esto buscaremos recalibrar y/o mejorar detalles que los usuarios consideren pertinentes para su uso.

6.2. Pruebas de accesibilidad web

6.2.1. WAVE

WAVE es un conjunto de herramientas de evaluación que ayudan a los autores a hacer que su contenido web sea más accesible para las personas con discapacidad. WAVE puede identificar muchos errores de accesibilidad y pautas de accesibilidad al contenido web (WCAG), pero también facilita la evaluación humana del contenido web [48].

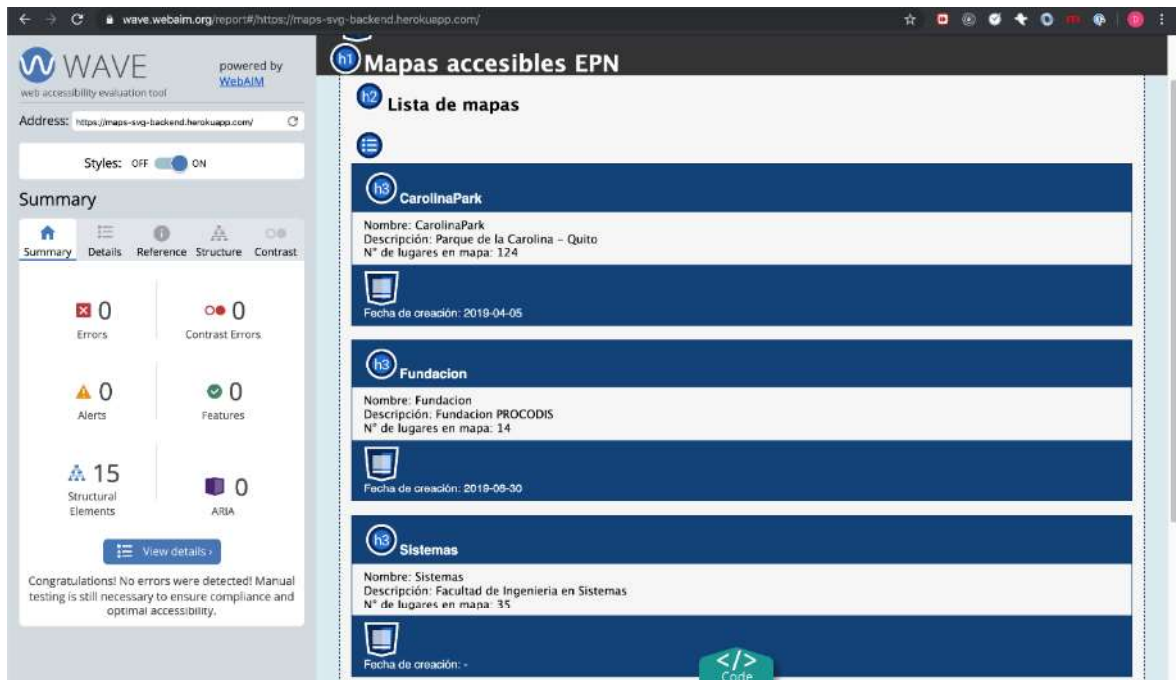


Figura 53. Prueba de la página de lista de mapas con WAVE.

La Figura 53 muestra los resultados de la evaluación de la página de la lista de mapas con la herramienta WAVE.

También se va a probar con la página de uno de los mapas para poder comprobar un nivel mínimo de nivel de accesibilidad.

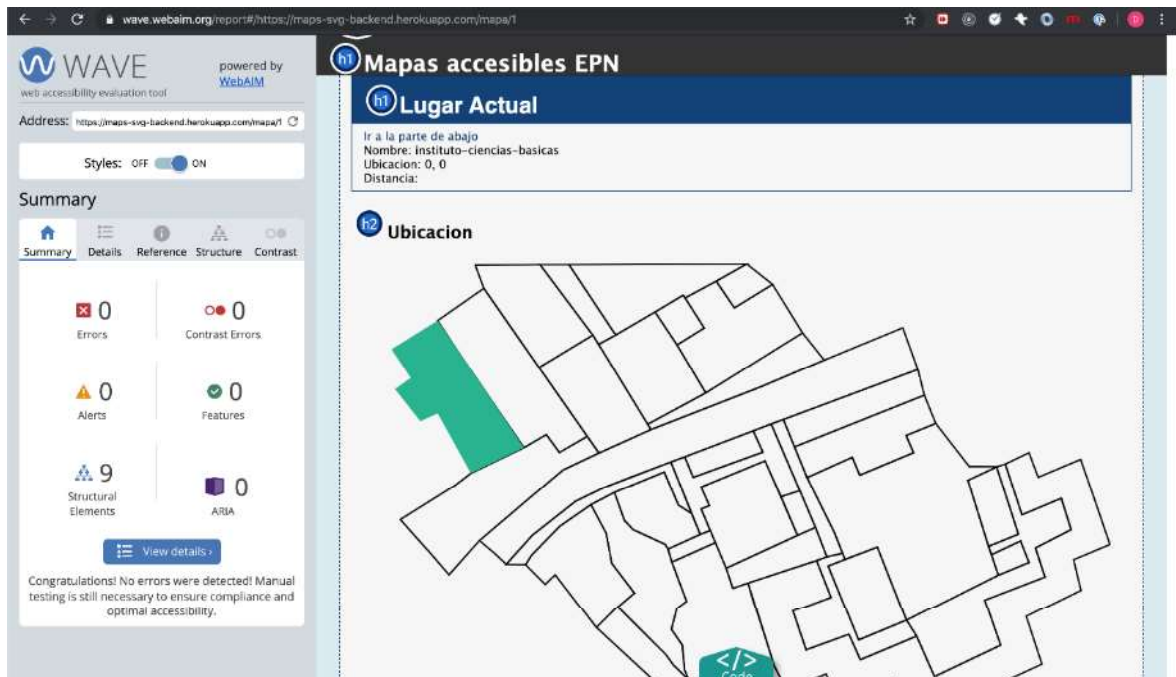


Figura 54. Prueba de la vista del mapa con WAVE.

La Figura 54 muestra los resultados de la evaluación de la página de la lista de mapas con la herramienta WAVE.

Después de evaluar el sistema con Wave, verificamos que cumpla con un mínimo de calidad tanto en la página principal y como en una página de los mapas.

6.2.2. Examinator

Examinator es un servicio en línea utilizado para evaluar de modo automático la accesibilidad de una página web, usando como referencia algunas técnicas recomendadas por las Pautas de Accesibilidad para el Contenido Web 2.0 (WCAG 2.0).

Examinator adjudica una puntuación entre 1 y 10 como un indicador rápido de la accesibilidad de las páginas y proporciona un informe detallado de las pruebas realizadas. La Figura 55 nos muestra la calificación de Examinator.



Figura 55. Prueba de la vista del mapa con Examinator.

Como se mostró, las pruebas de accesibilidad de WAV y Examinator no nos muestra fallos graves con respecto a la estructura general de la página. Con respecto a TAW se puede mover los estilos a un CSV aparte, pero se perdería la habilidad de abrir la imagen en programas exteriores.

6.2.3. PowerMapper - Accessibility Checker and Validator

A continuación, se ejecutó las pruebas de accesibilidad con la herramienta SortSite - Accessibility Checker and Validator de PowerMapper, la cual analiza todas las paginas relacionadas a un proyecto y nos da un informe detallado sobre el cumplimiento de las pautas WCAG 2.1. La herramienta mostró varios errores inicialmente, pero como podemos ver en la Figura 7, los cuales se redujeron al mínimo con la implementación de algunas correcciones en el software.

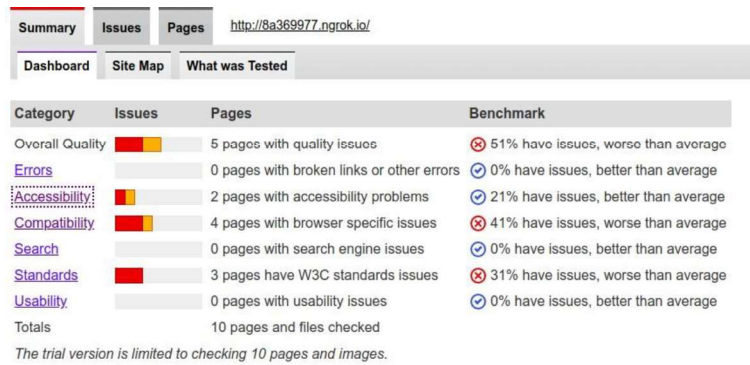


Figura 56: Resultados - Accessibility Checker and Validator.

Los errores restantes son causados por detalles relacionado al uso de imágenes SVG como son:

- Compatibilidad limitada con exploradores antiguos: Este error es omisible puesto que nuestra aplicación va a correr únicamente sobre la aplicación que se ha creado para este proyecto.
- Estilos embebidos: Las imágenes SVG usan estilos para cada uno de sus paths, por lo tanto, el extrapolar los estilos fuera de la imagen haría que esta perdiera la capacidad de funcionar individualmente.

Con estas pruebas automáticas se asegura un nivel aceptable de accesibilidad para poder ejecutar la imagen en los dispositivos móviles que usen la aplicación.

6.3. Pruebas de usuario

Pruebas de ejecución en dispositivos móviles.

Con el fin de probar las funciones de la aplicación directamente en las respuestas en 3 dispositivos móviles a la vez y la aplicación no presenta problemas al momento de presentar los datos a través de la interfaz de voz.

Como podemos ver en la Figura 57 la aplicación se ejecuta sin retrasos en los múltiples dispositivos. Estos dispositivos se van a ser usados por las personas que nos van a prestar su ayuda para las pruebas de usuario.

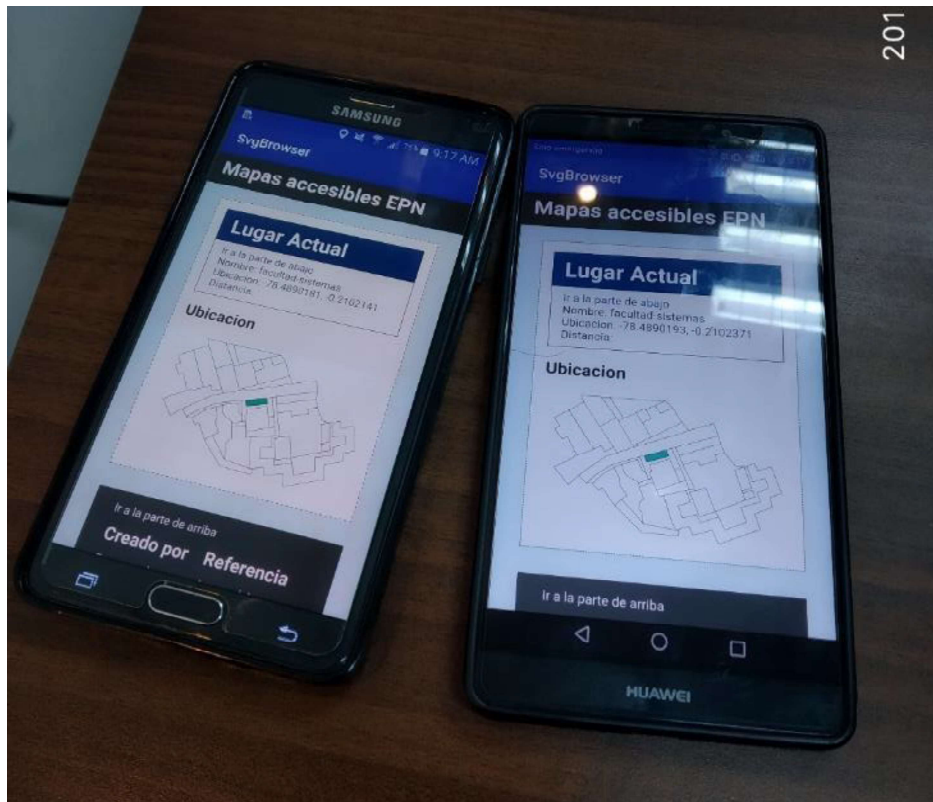


Figura 57. Prueba en varios dispositivos.

Pruebas con los usuarios

Para el desarrollo de las pruebas, se requirió de la ayuda de personas con discapacidad visual parcial o total que realizarían un recorrido por uno de los mapas con la aplicación. El objetivo es poder verificar la utilidad del sistema, presión de los mapas, y recibir cualquier comentario y/o mejora que pudiera necesitar el sistema a fin de demostrar la utilidad del sistema al usuario final.

Se realizaron 4 pruebas en el mapa de la zona sur del parque de la carolina con 3 personas con falta de visión total y un asistente que no tiene discapacidad visual. Se inicio con una explicación sencilla de uso y asistidos de uno de nosotros, se verifico la utilidad del sistema y reacciones de los usuarios al momento de usar la aplicación.

Después de terminar el recorrido con los usuarios se obtuvo los comentarios de los usuarios mediante una encuesta realizada a cada uno. En la Tabla 59 se muestra los resultados de dicha encuesta.

	Usuarios				
	Franklin Ochoa	Anabel Morales	Fausto Laiques	Franklin Anaguano	Hugo Arias
Lugar	Parque la Carolina				UTI
¿Considera que la aplicación cumple con el objetivo de representar a un mapa accesible?	Si	Si	Si	Si	Si
¿Considera que la aplicación es correctamente descriptiva en cuanto a los nombres de los lugares recorridos?	No	No	No	Si	No
¿Considera que la aplicación es de fácil uso?	Si	Si	Si	No	Si
¿Considera que la navegación era automática respecto al cambio de lugares mientras se desplazaba?	Si	Si	Si	Si	-
¿Considera que era clara y lógica la información en formato de audio que entregaba la aplicación?	Si	Si	Si	Si	Si
¿Considera que los botones si funcionaban y entregaban	Si	Si	Si	Si	Si

la información correcta explicada?					
¿Considera que la explicación respecto al funcionamiento fue completa?	Si	Si	Si	Si	Si
¿Conocía de aplicaciones similares a la expuesta?	Si	Si	Si	Si	Si
¿Recomendaría la aplicación?	Si	Si	Si	Si	No

Tabla 59. Encuesta de uso de aplicación a usuarios finales.

Adicionalmente se preguntó por la opinión de los usuarios para poder hacer mejoras al sistema. En la Tabla 60 se muestra dichos comentarios.

Nombre	Comentario / Observaciones
Franklin Ochoa	Es muy útil para poder movernos y saber en qué sitio nos encontramos. Sería bueno tener un nombre más representativo del lugar. Podría incluir un lector de letreros que ayudaría a ubicarnos a parte del nombre del lugar.
Anabel Morales	No quería manejar el celular por miedo a desconfigurarlo. Me parece útil para saber de dónde partí y hacia dónde dirigirme. Deberían ponerle un nombre adecuado que pueda recordar a las zonas del parque.
Fausto Laiques	Me gusto que la aplicación no consuma tanto recurso, pero si sería un poco desventajoso que tengamos el celular a la mano ya que nos podrían robar. El mapa es claro y fácil de usar, pero el nombre del lugar podría ser otro.
Franklin Anaguano	Como persona sin discapacidad, veo que tienen una muy buena aplicación, es un poco difícil de entender en cuanto a la forma de uso inicialmente. Pero después de su explicación, diría que si se asemeja a un mapa. Sería conveniente y útil para los compañeros, pero deberían ampliar las zonas de cobertura.
Hugo Arias	Si bien la aplicación cumple con su objetivo de mostrar los lugares cercanos sería más útil si diera una descripción más detallada de lo que se encuentra en cada edificio. Adicionalmente la

	implementación del eje Z dentro de la aplicación ayudaría a extender la información de los lugares que tienen más de una planta.
--	--

Tabla 60. Opiniones de los usuarios.

Este conjunto de sugerencias se puede extrapolar a una lista de mejoras al sistema las cuales se detallan en la Tabla 61.

Mejora sugerida	Acción
Nombre más representativo de los lugares.	Se añadió descripción a los lugares.
Descripción más detallada de que son los lugares o que se puede encontrar en los mismos.	Se añadió descripción a los lugares.
Lector de los elementos del entorno (letreros, objetos).	Se encuentra fuera del objetivo de este estudio. Por lo tanto, se agregó a las recomendaciones a futuro.
Uso de la aplicación con manos libres o mediante comandos de voz.	Se encuentra fuera del objetivo de este estudio. Por lo tanto, se agregó a las recomendaciones a futuro.
Tutorial de explicación al inicio.	Se agrego a las recomendaciones a futuro.
Añadir más mapas.	Se agrego a las recomendaciones a futuro.
Añadir el eje z en la aplicación para poder identificar en que piso se encuentra.	Se agrego a las recomendaciones a futuro.

Tabla 61: Acciones para las sugerencias de los usuarios.

6.4. Discusión de resultados.

Las pruebas automatizadas de la sección 6.2 arrojaron resultados positivos con lo cual se garantiza que la pagina cumple con el criterio AA de las WCAG 2.1 que son lineamientos estándares de accesibilidad web con respecto a navegadores web. Exceptuando por un par de puntos inherentes de las imágenes SVG. Esto facilita el análisis de la página para la aplicación móvil.

Con respecto a las pruebas con usuarios, esta al ser una aplicación móvil para el usuario es más fácil acceder a la información de los mapas desde cualquier sitio dentro del mapa. Y al tener una interfaz propia se tiene mayor control de que se presenta como respuesta auditiva al usuario. La aplicación mostró una respuesta positiva al cambiar de zona en el mapa. Si bien la aplicación es usable desde la pantalla principal algunos usuarios opinaron que se debería poder interactuar con la aplicación con los botones de volumen de los

audífonos, debido a que utilizarse en exteriores se es propenso a que el dispositivo sufra algún daño.

Respecto a los trabajos relacionados, se expandió lo visto en el prototipo creado en [17] a los dispositivos móviles de plataforma Android. Esto aporta mayor movilidad a la aplicación ya que puede acceder a información de la ubicación del usuario para ubicarlo en la zona correcta del mapa. De igual forma que se abordó las pruebas en [17] se llegó a resultados exitosos con respecto a los usuarios y los softwares de evaluación de accesibilidad. El sistema creado en a lo largo de este proyecto tiene la capacidad de informar al usuario sobre las informaciones de algún sistema de transporte (paradas y distancia del destino) en tiempo real.

La página web al seguir las pautas de accesibilidad de las WCAG 2.1 al igual que la pagina creada en [16] es accesible para los lectores de pantalla. Adicionalmente la información en la metadata de la imagen es fácilmente interpretable programáticamente, por lo tanto, es fácil proveer esta información a intérpretes de texto en el dispositivo móvil del usuario mediante operaciones definidas dentro de la aplicación.

Si bien la aplicación informa de los lugares cercanos al usuario es necesario de un asistente para que la persona se mueva por el lugar puesto que no se da una ruta específica en tiempo real. Esto puede solucionarse si se implementa un sistema de rutas o se acopla uno de alguna aplicación de terceros.

El sistema de ubicación de los teléfonos Android no da una ubicación exacta del dispositivo al momento de instalar la aplicación por lo que es necesario que la ubicación se calibre previo a la ejecución de la misma.

La aplicación no puede detectar objetos cercanos que no estén en el mapa por lo tanto sigue siendo necesario la ayuda de alguien para el desplazamiento o la ayuda de alguna tecnología de realidad aumentada que reconozca el entorno.

7. Conclusiones

- Mediante el desarrollo de la aplicación móvil de este proyecto se ha provisto de una primera aproximación a un mapa para para personas con capacidades visuales reducidas. El enfoque de esta aplicación es que a partir de la imagen SVG de un lugar se puede se puede aumentar la información dentro de esta y presentarla como un mapa interactivo con el fin de ubicar a los usuarios con respecto de a que distancia se encuentran de cada parte de ese lugar. Esta información generalmente no es tan detallada en servicios de mapas como Google Maps y OpenStreet Maps.

- Esta aplicación se puede complementar con la interacción con otras aplicaciones que provean un servicio de rutas, ya que con los dos puntos que se extraen de esta aplicación se puede trazar una ruta que permita al usuario ir de un punto a otro.
- Gracias a la interfaz otorgada por PostGIS podemos acceder fácilmente a la información de los lugares guardados en la base de datos, por lo tanto, la aplicación móvil puede utilizar dichos datos para informar al usuario de los lugares más cercanos. Adicionalmente, con la utilización de este plugin determinar las distancias entre los lugares y el dispositivo en el cual se seta ejecutando la aplicación se facilita mucho.
- La información incluida en el SVG nos sirve para informar rápidamente al usuario de los lugares, pero para que la aplicación utiliza de mejor manera la red móvil es mejor optar por una petición simple hacia el servidor la cual nos informara si el usuario ha cambiado de lugar.
- La metodología que se utilizó para este proyecto ayudó a progresar y organizar de mejor el desarrollo de la aplicación tanto del servidor como de la aplicación móvil. Si bien esta metodología ayudo al proyecto es preferible aplicarla en un grupo de al menos cinco personas ya que se facilita la asignación de roles y responsabilidades, la asignación de tareas y la resolución de las tareas y/o prioridades que surjan durante el desarrollo del sistema.

8. Recomendaciones

- Ya que la aplicación se encuentra limitada a 3 mapas de interés y polígonos de zonas cercanas, se recomienda ampliar los datos que la aplicación maneja (geográfico. y/o informativo) con el fin de ofrecer una experiencia más completa acerca de los lugares que se encuentran en cada mapa que son accedidos por el usuario final.
- Si se quiere ampliar el número soportado de usuarios simultáneos, se debe optar por un usar servicio de host que permita asignar más recursos al servidor / servicio. Puesto que en este proyecto se utilizó un servidor gratuito provisto por la plataforma Heroku, se encuentran las limitantes de acceso y uso de recursos que pueden ser fácilmente mitigados si se emplea un servicio de paga.
- La estructura de esta aplicación se puede extrapolar para sistemas automatizados que den información por voz de lugares en una zona de la ciudad, como por ejemplo una línea de buses puede utilizar este sistema para proveer una respuesta en tiempo real de las paradas.
- Podemos agregar valor en la información que generamos de forma geográfica esto con el fin de mejorar la experiencia del usuario final que consuma esta información, en este caso empleamos esta característica en los tags SVG y en la descripción de los polígonos generados en QGis. Esto puede ser un aliado comercial para exploración de franquicias que requieran un estudio de mercadeo de zonas para asignar y/o abrir un nuevo local, para sectores públicos, ya que podría asignarse nuevas zonas escolares, de salud, de servicios, todo esto con tan solo ver la información de los mapas generados, su meta información y su posición actual.
- Para aplicaciones móviles es altamente recomendado usar SVG Tiny 1.2 (recomendación de la W3C) ya que la demanda de la industria, el apoyo al grupo de trabajo SVG y las solicitudes de la comunidad de desarrolladores establecieron la necesidad de algún tipo de SVG adecuado para mostrar gráficos vectoriales en dispositivos pequeños (móviles).
- Utilizar tags SVG es beneficioso para proyectos que requieran agregación de información a una imagen, y ya que se pueden manipular esta información con tecnologías como JavaScript, nos permitiría realizar personalizaciones más dinámicas que las imágenes ráster. Por ejemplo, la animación SVG es una técnica popular para agregar interés e interactividad a un sitio web. Y, al ser un proyecto que se ejecutara en un ambiente móvil, SVG Tiny tiene ventajas para desarrollar esta tarea de una forma más adecuada.

- Debido a la naturaleza de los gráficos vectoriales, los datos que determinan el tamaño del archivo son las capas, efectos, máscaras, colores y gradientes utilizados. En comparación con otros formatos de imagen, las imágenes basadas en mapas de bits registran cada píxel de la imagen, así como la cantidad de colores utilizados. Una imagen basada en un mapa de bits será más costosa que una contraparte SVG. Podemos hacer que los archivos SVG sean aún más pequeños comprimiéndolos con gzip, que en caso de aplicaciones móviles significa menor consumo de recursos para presentar un resultado.

REFERENCIAS

- [1] Sandi's View, «How does technology help people who are blind or visually impaired?,» 05 2015. [En línea]. Available: <https://sandysview1.wordpress.com/2015/05/07/how-does-technology-help-people-who-are-blind-or-visually-impaired/>.
- [2] R. Blanco G., «REICE - La equidad y la inclusión social: uno de los desafíos de la educación,» 2006. [En línea]. Available: <https://www.redalyc.org/pdf/551/55140302.pdf>.
- [3] C. Venegas Sandoval y G. M. Mansilla Gomez, «Accesibilidad en web para personas con discapacidad visual,» [En línea]. Available: http://mingaonline.uach.cl/scielo.php?script=sci_arttext&pid=S0718-025X2005000100001&lng=es&nrm=iso&tlng=es.
- [4] R. Alcaraz Martínez y M. Ribera Turró, «Mapas digitales y aplicaciones basadas en la localización: mejoras en su accesibilidad para las personas ciegas,» 9 01 2015. [En línea]. Available: http://www.nosolousabilidad.com/articulos/mapas_digitales.htm.
- [5] S. Liste, «Organizacion mundial de la Salud,» 11 10 2018. [En línea]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/blindness-and-visual-impairment>.
- [6] S. VIEW, «How does technology help people who are blind or visually impaired?,» [En línea]. Available: <https://sandysview1.wordpress.com/2015/05/07/how-does-technology-help-people-who-are-blind-or-visually-impaired/>.
- [7] Consejo Nacional para la Igualdad de Discapacidades, «Consejo Nacional para la Igualdad de Discapacidades - Valores / Misión / Visión,» [En línea]. Available: <https://www.consejodiscapacidades.gob.ec/valores-mision-vision/>.
- [8] Consejo Nacional para la Igualdad de Discapacidades , «Accesibilidad Web en Ecuador,» [En línea]. Available: <https://www.consejodiscapacidades.gob.ec/accesibilidad-web-en-ecuador/>.

- [9] Consejo Nacional para la Igualdad de Discapacidades , «Norma INEN ISO/IEC 40500_Extracto: Directrices de Accesibilidad al Contenido Web del W3C (WCAG) 2.0,» [En línea]. Available: <https://www.consejodiscapacidades.gob.ec/biblioteca/>.
- [10] Consejo Nacional para la Igualdad de Discapacidades, «Consejo Nacional para la Igualdad de Discapacidades,» [En línea]. Available: <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>.
- [11] J. Khalid, «Information Technology & Software Engineering,» 2018. [En línea]. Available: <https://pdfs.semanticscholar.org/aae9/a9d40e799c5ace3688301414a448f97763a2.pdf>.
- [12] W. Binzhao y X. Bin, «IEEE - Mobile phone GIS based on mobile SVG,» 07 2005. [En línea]. Available: <https://ieeexplore.ieee.org/abstract/document/1525253>.
- [13] F. Lin y G. Chaozhen, «IEEE - Raster-Vector Integration Based on SVG on Mobile GIS Platform,» 2011. [En línea]. Available: <https://ieeexplore.ieee.org/document/6106534>.
- [14] C. Engel, E. Franziska Müller y G. Weber, «ACM - SVGPlott: an accessible tool to generate highly adaptable, accessible audio-tactile charts for and from blind and visually impaired people,» 06 2019. [En línea]. Available: <https://dl.acm.org/citation.cfm?id=3316793>.
- [15] J. Ducasse, A. Brock y C. Jouffrais, «Accessible Interactive Maps for Visually Impaired Users,» 10 2017. [En línea]. Available: https://link.springer.com/chapter/10.1007%2F978-3-319-54446-5_17.
- [16] T. Calle Jiménez y S. Luján-Mora, «RUA - Importancia de Accesibilidad Web en Mapas Geográficos para la Educación,» 11 2015. [En línea]. Available: <http://rua.ua.es/dspace/handle/10045/53736>.
- [17] T. Calle-Jimenez, A. Eiguez-Sarzosa y S. Luján-Mora, «Springer - Design of an Architecture for Accessible Web Maps for Visually Impaired Users,» 26 06 2018. [En línea]. Available: https://link.springer.com/chapter/10.1007/978-3-319-94334-3_23.

- [18] S. J. Armero, «Uso de SVG para desarrollar mapas web para personas con discapacidad visual,» 06 2019. [En línea]. Available: <http://rua.ua.es/dspace/handle/10045/93592>.
- [19] «WCAG 2.1 - Background on WCAG 2,» 05 06 2018. [En línea]. Available: <https://www.w3.org/TR/WCAG21/#background-on-wcag-2>. [Último acceso: 01 10 2018].
- [20] WC3, «Web Content Accessibility Guidelines (WCAG) 2.1,» [En línea]. Available: <https://www.w3.org/TR/WCAG21/>.
- [21] World Wide Web Consortium, «Scalable Vector Graphics (SVG) 1.1 (Second Edition),» 2003. [En línea]. Available: <https://www.w3.org/TR/SVG/>.
- [22] World Wide Web Consortium, «Scalable vector graphics (SVG) Tiny 1.2 Specification. W3C Recommendation,» 2008b. [En línea]. Available: <https://www.w3.org/TR/SVGTiny12/>.
- [23] TutorialsPoint, «TutorialsPoint - Node.js,» 2009. [En línea]. Available: https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm.
- [24] MDN contributors, «MDN - About JavaScript,» 2019. [En línea]. Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript.
- [25] TutorialsPoint, «TutorialsPoint - TypeScript - Overview,» 2007. [En línea]. Available: https://www.tutorialspoint.com/typescript/typescript_overview.htm.
- [26] PostGIS Project Steering Committee, «About PostGIS,» [En línea]. Available: <https://postgis.net/>.
- [27] The PostgreSQL Global Development Group, «About PostgreSQL,» 2018. [En línea]. Available: <https://www.postgresql.org/about/>.
- [28] D. Robledo Fernández, Desarrollo de Aplicaciones Android 2, Madrid: Ministerio de Educación, Cultura y Deporte. Área de Educación, 2015.
- [29] B. Poppe, «The Verge - Google announces over 2 billion monthly active devices on Android,» 17 05 2017. [En línea]. Available: <https://www.theverge.com/2017/5/17/15654454/android-reaches-2-billion-monthly-active-users>. [Último acceso: 01 10 2018].

- [30] Google Developers, «Android Lollipop,» Google, 20 02 2018. [En línea]. Available: <https://developer.android.com/about/versions/lollipop?hl=es-419>. [Último acceso: 01 10 2018].
- [31] Google Developers, «Paneles de control,» Google, 28 09 2018. [En línea]. Available: <https://developer.android.com/about/dashboards/?hl=es-419>. [Último acceso: 01 10 2018].
- [32] Google Developers, «Class - TextToSpeech,» Google, 06 06 2018. [En línea]. Available: <https://developer.android.com/reference/android/speech/tts/TextToSpeech>. [Último acceso: 01 10 2018].
- [33] B. Gobin, «Academia.edu - An Agile and Modular Approach for Developing Ontologies,» [En línea]. Available: (1) https://www.academia.edu/7058738/An_Agile_and_Modular_Approach_for_Developing_Ontologies.
- [34] S. MERZOUK, S. ELHADI, H. ENNAJI, A. MARZAK y N. SAEL, «Semanticscholar.org - A Comparative Study of Agile Methods: Towards a New Model-based Method,» [En línea]. Available: <https://pdfs.semanticscholar.org/e5e6/e3974b48248df0f6d952cc7b90a4fe67eed6.pdf>.
- [35] Scrum, «Scrum.org - The home of Scrum,» [En línea]. Available: <https://www.scrum.org/resources/blog/que-es-scrum>.
- [36] U. d. Barcelona, «¿Te conviene utilizar la metodología SCRUM en tus proyectos?,» [En línea]. Available: <https://www.obs-edu.com/int/blog-project-management/temas-actuales-de-project-management/te-conviene-utilizar-la-metodologia-scrum-en-tus-proyectos>.
- [37] Y. D. Amaya Balaguera, «Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles,» Universidad de la Rioja, 2013. [En línea]. Available: <https://dialnet.unirioja.es/servlet/articulo?codigo=6041502>. [Último acceso: 02 10 2018].

- [38] D. Strode, «Agile methods: a comparative analysis,» [En línea]. Available: <https://pdfs.semanticscholar.org/a32b/bff37b09b0e829808d57d2bcfee76d972c3b.pdf>.
- [39] Visual Paradigm, «What Are The Three Scrum Roles?,» [En línea]. Available: <https://www.visual-paradigm.com/scrum/what-are-the-three-scrum-roles/>.
- [40] Visual paradigm, «What are Scrum Artifacts?,» [En línea]. Available: <https://www.visual-paradigm.com/scrum/what-are-scrum-artifacts/>.
- [41] Visual Paradigm, «What are the Scrum Events?,» [En línea]. Available: <https://www.visual-paradigm.com/scrum/what-are-scrum-events/>.
- [42] Scrummanager.net, «Scrum Manager Book - Velocidad,» 10 2014. [En línea]. Available: https://www.scrummanager.net/bok/index.php?title=Velocidad,_trabajo_y_tiempo.
- [43] P. Soneira, «Laboratorio TI.,» [En línea]. Available: <https://www.laboratorioti.com/2016/09/26/tecnica-priorizacion-moscow/>.
- [44] S. Watts, «BMC - What is Sprint Zero? Sprint Zero Explained,» 01 2018. [En línea]. Available: <https://www.bmc.com/blogs/sprint-zero/>.
- [45] Atlassian, «Continuous integration, explained,» ACM library, 2017. [En línea]. Available: <https://www.atlassian.com/continuous-delivery/continuous-integration>.
- [46] S. Fraser, D. Astels, K. Beck, B. U. J. McGregor y J. Newkirk, «Discipline and practices of TDD: (test driven development),» ACM Library, 2003. [En línea]. Available: <https://dl.acm.org/citation.cfm?id=949407>.
- [47] Scrum Manager, «Scrum Manager Book,» 03 2014. [En línea]. Available: <https://www.scrummanager.net/bok/index.php?title=Epic>.
- [48] WebAIM, «WAVE,» [En línea]. Available: <https://wave.webaim.org/report#/https://maps-svg-backend.herokuapp.com/>.
- [49] PivotalTracker Team, «PivotalTracker,» 2017. [En línea]. Available: https://www.pivotaltracker.com/help/articles/quick_start/. [Último acceso: 01 10 2018].

- [50] npatro, «GitHub - javascript-unit-testing-with-mocha,» [En línea]. Available: <https://github.com/npatro/javascript-unit-testing-with-mocha>.
- [51] The SVG Working Group, «Scalable Vector Graphics (SVG) Tiny 1.2 Specification,» The SVG Working Group, 22 12 2008. [En línea]. Available: <https://www.w3.org/TR/SVGTiny12/>.
- [52] CTIC, «Taw,» [En línea]. Available: <https://www.tawdis.net/>.
- [53] J. Unadkat, «BrowserStack,» [En línea]. Available: <https://www.browserstack.com/guide/what-is-selenium-ide..>
- [54] Scrum org, «Scrum.org The home of scrum,» [En línea]. Available: <https://www.scrum.org/resources/blog/que-es-scrum>.
- [55] T. Calle Jiménez, S. Sánchez Gordón y S. Luján-Mora, «RUA - Retos de accesibilidad en Geo-MOOCs,» 2013. [En línea]. Available: <http://rua.ua.es/dspace/handle/10045/41445>.

ANEXOS

Anexo A

Fotografías de las pruebas con los usuarios finales.



Figura 58. Explicación del funcionamiento de la aplicación.



Figura 59. Prueba en la Carolina con persona no vidente.



Figura 60. Prueba en la Carolina con persona no vidente.

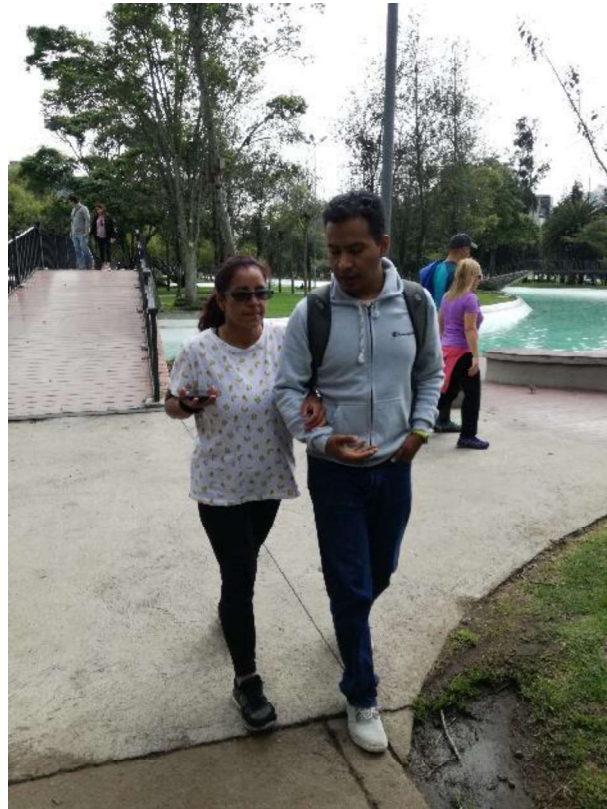


Figura 61. Prueba en la Carolina con persona no vidente.

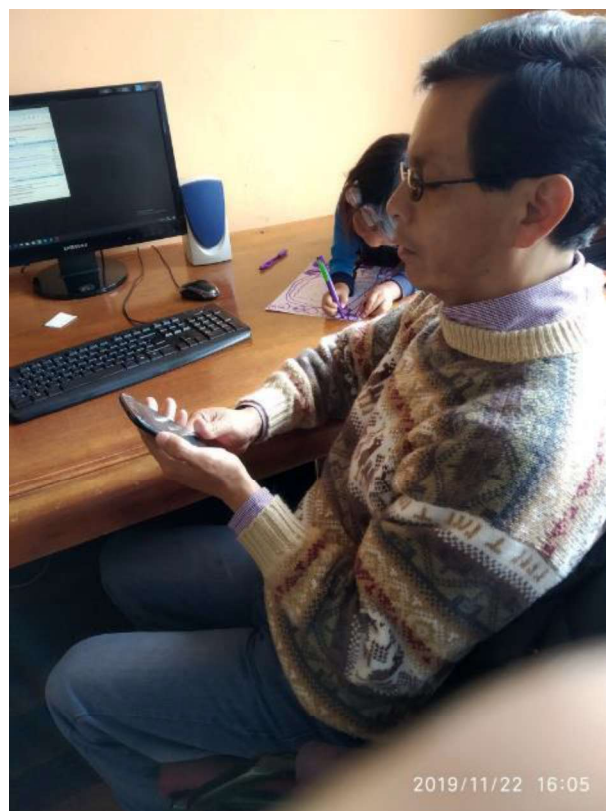


Figura 62. Prueba en la Fundación Procodis con persona no vidente.

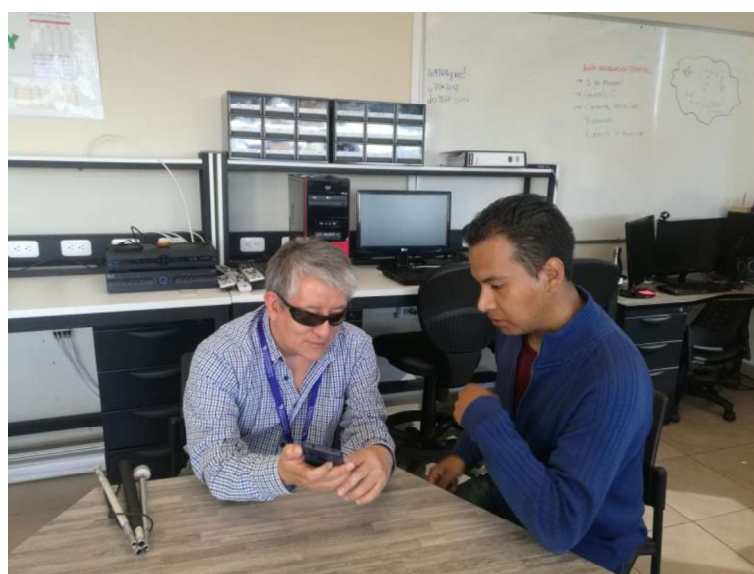


Figura 63. Prueba en la UTI con usuario no vidente.

Anexo B.

Resumen de la encuesta a los usuarios posterior a las pruebas hecha en el Parque de la Carolina. **Encuesta-opiniones-usabilidad.xlsx**

Anexo C.

Control del progreso de los puntos de los sprint. **Graficos-Burndown-seguimiento.xlsx**

Anexo D.

Captura informe Examinator. **Informe-examinator.ws.png**