



Universidad de Tarapacá
Facultad de Ingeniería
Departamento de Ingeniería en
Computación e Informática

Ingenierí@
Computación e Informática

SISTEMA ASISTENCIAL BASADO EN IOT DE RECONOCIMIENTO DE AGENTES EXTERNOS PARA PERSONAS EN CONDICIÓN DE DISCAPACIDAD VISUAL “NFTOUCH”

Autor(es): Vranika Santiago Yovich
Luciano Vera Norambuena
Bastián Vega Devia
Andrew Campos Seguel

Asignatura: Proyecto II

Profesor: Diego Aracena Pizarro

ARICA, 28 DE NOVIEMBRE 2023

HISTORIAL DE CAMBIOS

Fecha	Versión	Descripción	Autor(es)
07/09/2023	1.0	Versión preliminar del Informe I (planificación)	Andrew Campos Vranika Santiago Luciano Vera Bastián Vega
23/09/2023	1.1	Revisión y modificación del plan	Andrew Campos Vranika Santiago
27/10/2023	1.2	Correcciones formulación del plan	Andrew Campos Vranika Santiago Luciano Vera Bastián Vega
13/11/2023	1.3	Versión preliminar del Informe II (avance)	Vranika Santiago
17/11/2023	1.4	Añadidura de entregables (requerimientos, casos de uso, mockups, arquitectura)	Andrew Campos Vranika Santiago Luciano Vera Bastián Vega
23/11/2023	1.5	Revisión y modificación del avance	Andrew Campos Vranika Santiago

TABLA DE CONTENIDOS

ÍNDICE DE TABLAS.....	5
ÍNDICE DE ILUSTRACIONES.....	5
RESUMEN	6
ABSTRACT	6
1. PANORAMA GENERAL	7
1.1. <i>Introducción.....</i>	<i>7</i>
1.2. <i>Escenario del Problema.....</i>	<i>8</i>
1.3. <i>Escenario de la Solución.....</i>	<i>8</i>
1.4. <i>Propósito</i>	<i>9</i>
1.5. <i>Alcance.....</i>	<i>9</i>
1.6. <i>Objetivos.....</i>	<i>9</i>
1.6.1. <i>Objetivo General</i>	<i>9</i>
1.6.2. <i>Objetivos Específicos.....</i>	<i>9</i>
1.7. <i>Suposiciones.....</i>	<i>9</i>
1.8. <i>Restricciones.....</i>	<i>10</i>
1.9. <i>Entregables</i>	<i>10</i>
2. ORGANIZACIÓN DEL PERSONAL	11
2.1. <i>Personal y entidades internas.....</i>	<i>11</i>
2.2. <i>Roles y responsabilidades.....</i>	<i>11</i>
2.3. <i>Mecanismos de comunicación.....</i>	<i>12</i>
3. PLANIFICACIÓN DE LOS PROCESOS DE GESTIÓN	13
3.1. <i>Planificación inicial del proyecto</i>	<i>13</i>
3.1.1. <i>Planificación de estimaciones.....</i>	<i>13</i>
3.1.1.1. <i>Costos de Hardware</i>	<i>13</i>
3.1.1.2. <i>Costos de Software.....</i>	<i>13</i>
3.1.2. <i>Planificación de Recursos Humanos.....</i>	<i>14</i>
3.1.2.1. <i>Costos de RRHH.....</i>	<i>14</i>
3.1.3. <i>Planificación de costos totales.....</i>	<i>16</i>
3.2. <i>Lista de actividades (Carta Gantt)</i>	<i>17</i>
3.2.1. <i>Actividades de trabajo.....</i>	<i>17</i>
3.2.2. <i>Asignación de tiempo.....</i>	<i>20</i>
3.3. <i>Planificación de la gestión de riesgos.....</i>	<i>21</i>
4. PLANIFICACIÓN DE LOS PROCESOS TÉCNICOS.....	24
4.1. <i>Modelo de Proceso</i>	<i>24</i>
4.1.1. <i>Especificación de Requerimiento</i>	<i>24</i>
4.1.2. <i>Descripción de la Arquitectura.....</i>	<i>25</i>
4.1.3. <i>Diseño de Interface de Usuario</i>	<i>26</i>
4.1.4. <i>Caso de Uso de Contexto.....</i>	<i>28</i>
4.1.5. <i>Caso de Uso: Listar registros NFC (App móvil)</i>	<i>29</i>
4.1.5.1. <i>Diagrama de secuencia: Nivel 0</i>	<i>30</i>

4.1.5.2. Diagrama de Secuencia: Nivel 1	31
4.1.6. Caso de Uso: Gestionar registros NFC (App móvil).....	32
4.1.6.1. Diagrama de Secuencia: Nivel 0	33
4.1.6.2. Diagrama de Secuencia: Nivel 1	34
4.1.7. Caso de Uso: Gestionar instancias de la base de datos	35
4.1.7.1. Diagrama de Secuencia: Nivel 0	36
4.1.7.2. Diagrama de Secuencia: Nivel 1	37
4.1.8. Caso de Uso: Retornar instancias de la base de datos	38
4.1.8.1. Diagrama de Secuencia: Nivel 0	38
4.1.8.2. Diagrama de Secuencia: Nivel 1	39
4.1.9. Diagrama de clases	40
4.2. Herramientas y técnicas.....	40
5. CONCLUSIONES.....	42
REFERENCIAS	42

ÍNDICE DE TABLAS

Tabla 1: Roles y responsabilidades.....	11
Tabla 2: Costos de Hardware.....	13
Tabla 3: Costos de Software.....	13
Tabla 4: Costos de RRHH.....	15
Tabla 5: Costos totales del proyecto.....	16
Tabla 6: Actividades de trabajo.....	19
Tabla 7: Tipos de Riesgos.....	21
Tabla 8: Gestión de Riesgos.....	23
Tabla 9: CUS: Listar registros NFC (App Móvil).....	29
Tabla 10: CUS: Gestionar registros NFC (App Móvil).....	32
Tabla 11: CUS: Gestionar instancias de la Base de Datos.....	35
Tabla 12: CUS: Retornar instancias de la Base de datos.....	38

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Escenario del Problema.....	8
Ilustración 2: Escenario de la Solución.....	8
Ilustración 3: Carta Gantt.....	20
Ilustración 4: Diagrama Caso de Uso de Contexto.....	28
Ilustración 5: Listar Registros NFC (App Móvil) - Diagrama nivel 0.....	30
Ilustración 6: Listar registros NFC (App Móvil) - Diagrama nivel 1.....	31
Ilustración 7: Gestionar registros NFC (App Móvil) - Diagrama nivel 0.....	33
Ilustración 8: Gestionar registros NFC - Diagrama nivel 1.....	34
Ilustración 9: Gestionar instancias de la Base de Datos - Diagrama nivel 0.....	36
Ilustración 10: Gestionar instancia de la Base de Datos - Diagrama nivel 1.....	37
Ilustración 11: Retornar instancia de la Base de Datos - Diagrama nivel 0.....	38
Ilustración 12: Retornar instancia de la Base de Datos - Diagrama nivel 1.....	39
Ilustración 13: Diagrama de clases.....	40

RESUMEN

Las personas con discapacidad visual severa representan un grupo de población vulnerable, debido a que a menudo utilizan la memoria táctil en su vida diaria para identificar y diferenciar objetos. Sin embargo, esta estrategia puede ser peligrosa en situaciones como el manejo de medicamentos. Para mejorar la autonomía y la confianza de estas personas al interactuar con su entorno, se ha propuesto el desarrollo de una aplicación móvil de fácil uso que emplea tecnología IoT y Raspberry-Pi. Dicha aplicación permite escanear chips NFC. Estos chips contendrán información detallada sobre los objetos escaneados y serán reproducidas por medio del dispositivo móvil a través de una función de lectura de texto a voz. Esta solución tiene como objetivo proporcionar una alternativa segura y efectiva para la identificación de objetos y promover la independencia de las personas con discapacidad visual severa.

PALABRAS CLAVES: *Discapacidad visual, aplicación móvil, tecnología IoT, Raspberry-Pi, tarjetas NFC.*

ABSTRACT

People with severe visual impairment represent a vulnerable population group, as they often rely on tactile memory in their daily lives to identify and differentiate objects. However, this strategy can be dangerous in situations such as medication handling. To improve the autonomy and confidence of these people when interacting with their environment, the development of a user-friendly mobile application using IoT and Raspberry-Pi technology has been proposed. This application allows users to scan NFC chips. These chips will contain detailed information about the scanned objects, which will be reproduced through the mobile device through a text-to-speech function. This solution aims to provide a safe and effective alternative for object identification and to promote the independence of people with severe visual impairments.

KEYWORDS: *Visual impairment, mobile application, IoT technology, Raspberry-Pi, NFC chips.*

1. PANORAMA GENERAL

1.1. Introducción

Internet de las Cosas, también conocido como IoT por sus siglas en inglés, se caracteriza por ser una evolución tecnológica que permite la interacción y comunicación [1] entre objetos y dispositivos a través de la conexión a Internet.

No obstante, no cualquier objeto que esté conectado de manera inalámbrica a Internet es considerado IoT. De acuerdo con uno de los principales productores mundiales de software (SAP) *IoT se refiere más a cosas u objetos que estén equipados con sensores, software y tecnología que tengan la capacidad de transmitir y recibir datos* [2]. Estos sensores recopilan datos del entorno del objeto que luego son procesados por el software y enviados a través de Internet.

En este contexto tecnológico, surge la siguiente interrogante: ¿cómo se puede aprovechar el IoT para mejorar la calidad de vida de las personas? Una de las posibles respuestas es que el IoT tiene la funcionalidad de ayudar a incrementar la independencia y autonomía de las personas. Un grupo que se vería beneficiado, especialmente cuando ingresan agentes externos en el entorno hogareño, sería el de los no videntes.

Según la Organización Mundial de la Salud (OMS), hay alrededor de 45 millones de personas ciegas en el mundo. En Chile, en tanto, aproximadamente 850 mil personas tienen algún tipo de discapacidad visual y alrededor de 84 mil tienen ceguera total [3]. Es decir, la ceguera afecta a un total del 4% de la población chilena. Al pertenecer a una comunidad reducida, las grandes empresas y productores no tienen como prioridad atender a sus necesidades específicas.

Este proyecto, denominado “NFTouch”, nace para facilitar las tareas diarias de las personas con discapacidad visual. Además, potenciará su autonomía con el uso de herramientas tecnológicas como chips NFC y Raspberry-Pi. La aplicación está dirigida a personas que formen parte de la comunidad de invidentes y que dispongan de un dispositivo móvil. Es así como, sensores integrados en objetos que puedan ser utilizados en su vida diaria podrían proporcionar información relevante sobre el entorno en el que conviven. El escaneo de chip NFC (Near Field Communication) en objetos, como en medicamentos o prendas de vestir.

1.2. Escenario del Problema



Ilustración 1: Escenario del Problema.

Las personas que padecen una discapacidad visual enfrentan desafíos para reconocer agentes externos que ingresan a sus hogares y a su vez, los cambios que son producidos por estos mismos. Esto es debido a las formas, tamaños y texturas similares que pueden tener. Por ejemplo, en el caso de los fármacos, el individuo podría tomar un medicamento con un envase parecido al que inicialmente deseaba consumir, causándole efectos no deseados. Por consecuencia, se dificulta la realización de tareas cotidianas reduciendo su autonomía e independencia.

1.3. Escenario de la Solución

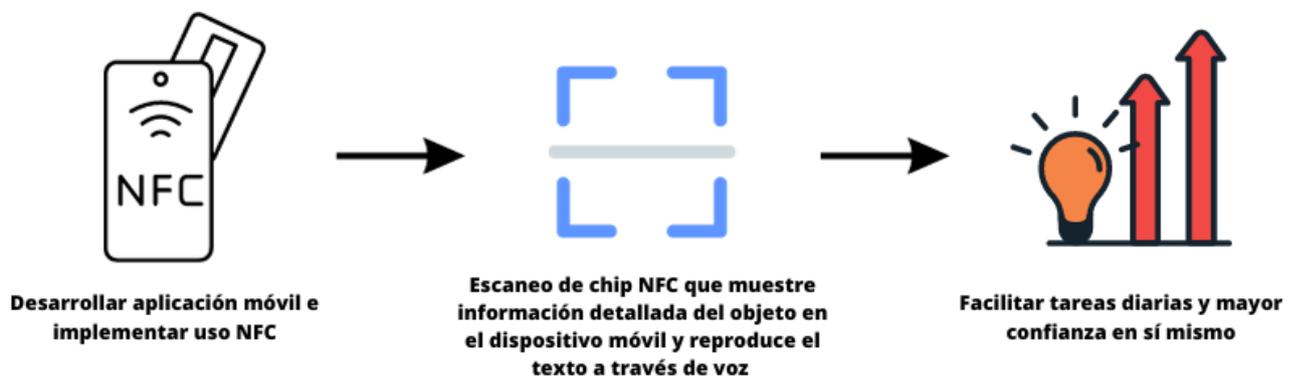


Ilustración 2: Escenario de la Solución.

Para solucionar el problema establecido en el punto anterior ([1.2. Escenario del Problema](#)) se ha propuesto desarrollar una aplicación móvil. Su función principal sería la de escanear el chip NFC, para luego mostrar la información detallada del objeto. Esta información se agregaría mediante la voz y se guardaría en una Raspberry-Pi. Primero, la Raspberry-Pi organizará la información recibida en una estructura de datos en donde cada instancia de dicha estructura contendrá un identificador único. Luego, el chip NFC almacenará el identificador previamente mencionado. Al momento de escanear el chip NFC, este llevaría hacia la instancia específica con el identificador único que tenga en la Raspberry-Pi y retornará toda la información almacenada en la estructura de datos utilizada. Cabe agregar que la información sería reproducida a

través de voz por el micrófono del dispositivo, para así proveer de una mayor independencia al invidente al momento de identificar objetos que le supongan mayor dificultad de reconocimiento.

1.4. Propósito

El propósito será mejorar la calidad de vida de las personas con discapacidad visual al facilitarles el reconocimiento y detección de objetos, además se les brinda una mayor independencia.

1.5. Alcance

El alcance del proyecto "NFTouch" se centra en diseñar e implementar una aplicación móvil orientada a personas con discapacidad visual severa para mejorar su autonomía. La aplicación permitirá a los usuarios escanear chips NFC con su dispositivo móvil y acceder a información detallada sobre objetos previamente introducidos en la base de datos. Esta información se reproducirá a través de voz utilizando la función de lectura de texto a voz del dispositivo.

1.6. Objetivos

1.6.1. Objetivo General

El objetivo general es desarrollar una aplicación móvil asistencial basada en IoT para personas con discapacidad visual que les permita reconocer y detectar agentes externos en el entorno hogareño.

1.6.2. Objetivos Específicos

- i. Investigar las necesidades de las personas con discapacidad visual en cuanto a funciones que necesitan acceder en sus dispositivos móviles.
- ii. Comprender los desafíos que enfrentan las personas con discapacidad visual al utilizar dispositivos móviles.
- iii. Desarrollar un prototipo de la aplicación móvil que sea cómoda y de fácil uso utilizando Ionic como framework.
- iv. Evaluar el prototipo de la aplicación móvil con personas con discapacidad visual para obtener comentarios sobre la funcionalidad, facilidad de uso y accesibilidad.

1.7. Suposiciones

- i. Se asume que los usuarios que utilicen la aplicación móvil padecen de alguna discapacidad visual.
- ii. Se asume que los usuarios tienen acceso a un dispositivo móvil (smartphone) que cuente con un sistema operativo Android.
- iii. Se asume que los usuarios tienen acceso a Internet.

- iv. Se asume que los dispositivos móviles que los usuarios utilizarán son compatibles con el escáner NFC.
- v. Se asume que por ser un dispositivo con el sistema operativo Android, el usuario tendrá activada la función de Google Talk-Back.

1.8. Restricciones

Existe un conjunto de limitaciones para que el desarrollo de NFTouch sea exitoso, las cuales se pueden observar en la siguiente lista:

- **Raspberry-Pi:** conocimientos del funcionamiento de Raspberry-Pi.
- **Temática:** debe ser apto para personas con discapacidad móvil o física.
- **Control:** el manejo de la aplicación debe ser vía smartphone.
- **Maqueta:** diseñar y construir una maqueta con materiales reciclables (en su mayoría).
- **Sistema operativo:** debe ser Android.

1.9. Entregables

- Informe y presentación de la planificación del proyecto.
- Informe y presentación del avance del proyecto.
- Informe y presentación final del proyecto.
- Maqueta.
- Bitácora semanal.
- Wiki y manual de usuario.
- Producto final.

2. ORGANIZACIÓN DEL PERSONAL

2.1. Personal y entidades internas

A cada integrante del grupo se le designó una responsabilidad, donde estarán a cargo de cumplir en el tiempo estimado. Cabe recalcar que por mucho que existan encargados para cada eje, todos los integrantes deben velar por la realización tanto de sus tareas como la de sus compañeros.

- **Jefe de grupo:** es el líder del equipo, responsable de la supervisión, dirección y toma de decisiones estratégicas.
- **Analista:** este rol implica entender los requisitos del negocio, analizarlos y convertirlos en especificaciones técnicas para el equipo de desarrollo. También puede implicar pruebas y validación para asegurar que el producto final cumpla con los requisitos.
- **Desarrollador Backend:** se encarga de la lógica del negocio, las operaciones de la base de datos, la implementación del servidor y la integración con el frontend.
- **Desarrollador Frontend:** se encarga de la interfaz de usuario y la experiencia del usuario. Trabaja en el diseño y la interactividad de la aplicación móvil.

2.2. Roles y responsabilidades

Rol	Responsable	Involucrados
Jefe de proyecto	Vranika Santiago	Vranika Santiago y Andrew Campos
Analista	Andrew Campos	Vranika Santiago, Luciano Vera, Bastián Vega y Andrew Campos
Desarrollador Backend	Bastián Vega	Andrew Campos y Bastián Vega
Desarrollador Frontend	Luciano Vera	Luciano Vera, Vranika Santiago y Bastián Vega

Tabla 1: Roles y responsabilidades.

2.3. Mecanismos de comunicación

Para garantizar una buena organización y comunicación, se han establecido las siguientes herramientas y plataformas:

- **WhatsApp:** Se utilizará para programar y anunciar horarios de reuniones, llevar un registro de la asistencia diaria de los integrantes y compartir archivos. Se estableció un horario pertinente para compartir mensajes, el cual es de lunes a viernes desde las 8:00 hasta las 18:00.
- **Discord:** Se utilizará como un espacio de discusión y toma de decisiones para las propuestas de mejora del proyecto.
- **Google Drive:** Se utilizará como respaldo para todos los archivos generados.
- **Redmine:** Plataforma oficial donde se suben los archivos.
- **Correos electrónicos:**
 - Vranika Santiago Yovich – vranika.santiago.yovich@alumnos.uta.cl
 - Luciano Vera Norambuena – luciano.vera.norambuena@alumnos.uta.cl
 - Bastián Vega Devia – bastian.vega.devia@alumnos.uta.cl
 - Andrew Campos Seguel – andrew.campos.seguel@alumnos.uta.cl

3. PLANIFICACIÓN DE LOS PROCESOS DE GESTIÓN

3.1. Planificación inicial del proyecto

En la parte de hardware, los productos a utilizar fueron los siguientes:

- Arriendo de equipos
- Samsung Galaxy A53
- Raspberry-Pi3 model B
- Tarjeta NFC NTAG215

Mientras que para la parte de software:

- Licencia Microsoft Office

3.1.1. Planificación de estimaciones

A continuación, se especificarán los costos estimados (en CLP) tanto para hardware como para software. Además, de los valores de cada rol.

3.1.1.1. Costos de Hardware

Arriendos de equipo				
Nombre producto	Cantidad	Arriendo por mes	Tiempo en meses	Costo total
Equipos personales	4	\$80.000	4	\$320.000
Hardware general				
Nombre producto	Costo unidad	Cantidad	Costo total	
Samsung Galaxy A53	\$400.000	1	\$400.000	
Raspberry-Pi3 model B	\$60.000	1	\$60.000	
Tarjeta NFC NTAG215	\$1.590	10	\$15.000	
			Total	\$796.000

Tabla 2: Costos de Hardware.

3.1.1.2. Costos de Software

Nombre producto	Costo unidad	Cantidad	Costo total
Licencia Microsoft Office	\$5306	4	\$21.224
Total por 4 meses			\$84.897

Tabla 3: Costos de Software.

3.1.2. Planificación de Recursos Humanos

3.1.2.1. Costos de RRHH

Puntos para tomar en consideración:

- La fecha final estimada del proyecto es el 6 de diciembre del año 2023, es decir, en total son 16 semanas de trabajo (equivalentes a 4 meses). Considerando, que por semana son 6 horas de trabajo en clases y una estimación de 6 horas de trabajo autónomo, se obtiene un total de 192 horas totales.
- El valor hora para cada rol es un promedio de la información encontrada en Internet. Hay que destacar que cada responsable gana un 10% extra (*).
- Para calcular el valor total mensual de cada encargado, se debe multiplicar el valor hora por las horas trabajadas mensualmente, que en este caso son 48 (se dividen las horas totales por los meses trabajados). En caso de que sea responsable, se le agrega el 10% nombrado anteriormente. Para más detalles véase [tabla 4: Costos de RRHH](#).

Trabajador	Rol	Valor por hora	Horas mensuales totales (48)	Costo total por mes
Vranika Santiago	Jefe proyecto ¹ (*)	\$10.154	8	\$81.232
	Analista ²	\$5.538	7	\$38.766
	Frontend ³	\$8.000	33	\$264.000
Luciano Vera	Analista	\$5.538	7	\$38.766
	Frontend (*)	\$8.800	41	\$360.800
Andrew Campos	Analista (*)	\$6.092	7	\$42.644
	Sub-jefe proyecto	\$9.231	8	\$73.848
	Backend ⁴	\$8.615	33	\$284.295
Bastián Vega	Analista	\$5.538	7	\$38.766
	Frontend	\$8.000	17	\$136.000
	Backend (*)	\$9.476	24	\$227.424
Total por 1 mes				\$1.586.541
Total por 4 meses				\$6.346.164

Tabla 4: Costos de RRHH.

¹ Talent (s.f.). Salario para Jefe De Proyecto en Chile - Salario Medio. Accedido el 24 de septiembre de 2023. <https://cl.talent.com/salary?job=Jefe+de+proyecto>

² Talent (s.f.). Salario para Analista Programador en Chile - Salario Medio. Accedido el 24 de septiembre de 2023. <https://cl.talent.com/salary?job=Analista+programador>

³ Talent (s.f.). Salario para Desarrollador Frontend en Chile - Salario Medio. Accedido el 24 de septiembre de 2023. <https://cl.talent.com/salary?job=Desarrollador+frontend>

⁴ Talent (s.f.). Salario para Desarrollador Backend en Chile - Salario Medio. Accedido el 24 de septiembre de 2023. <https://cl.talent.com/salary?job=Desarrollador+backend>

3.1.3. Planificación de costos totales

Tipo coste	Costo
Hardware	\$796.000
Software	\$84.896
Recursos humanos + holgura (30%)	\$8.250.013
Costo total del proyecto	\$9.130.909

Tabla 5: Costos totales del proyecto.

Por lo tanto, la estimación del proyecto durante 16 semanas (o 4 meses) de trabajo tendrá un presupuesto total de \$9.130.909

3.2. Lista de actividades (Carta Gantt)

3.2.1. Actividades de trabajo

Actividad	Descripción	Responsable
Bitácoras	Registro de todas las actividades que se desarrollan semanalmente.	Vranika Santiago
Wiki	Se capturan y comparten ideas e información del proyecto.	Vranika Santiago
Organización	Designación de la actividad que estará encargado cada integrante.	Vranika Santiago
Establecer problemática	Identificar y definir es el problema que el proyecto busca abordar o resolver.	Vranika Santiago
Analizar diversas soluciones	Evaluación de diversas alternativas para resolver la problemática identificada.	Vranika Santiago
Presentación idea	Presentación acerca de la problemática escogida y su respectiva solución.	Luciano Vera
Diseñar maqueta	Realizar bosquejo, decidir los materiales reciclables a utilizar y justificación de la maqueta.	Luciano Vera
Construcción maqueta	Armado de maqueta con materiales reciclables (en su mayoría).	Luciano Vera
Presentación maqueta	Se presenta el desarrollo, la justificación, escenarios experimentales y su resultado.	Bastián Vega
Informe I	Crear informe para la parte I: formulación del proyecto.	Andrew Campos
Análisis de framework	Analizar lenguajes de programación para utilizar en la aplicación e investigar los distintos framework que son útiles	Bastián Vega

	para el desarrollo de aplicaciones Android.	
Estimar costos	Calcular el presupuesto total del proyecto.	Bastián Vega
Presentación I	Elaborar presentación de la formulación del proyecto y luego exponerlo.	Luciano Vera
Realizar Casos de Uso	Realizar diagramas de casos de uso de sistema y casos de uso de interfaz.	Andrew Campos
Creación de mockups	Material que simula el producto final tanto en lo estético como en lo funcional.	Luciano Vera
Definir técnica de prueba	Mecanismo que se utiliza para encontrar defectos a través de pruebas.	Vranika Santiago
Establecer requerimientos	Definir requerimientos no funcionales y funcionales.	Bastián Vega
Establecer arquitectura	Diagrama para visualizar de mejor manera la arquitectura de los sistemas utilizados.	Bastián Vega
Informe II	Redacción del informe del avance del proyecto.	Vranika Santiago
Presentación II	Creación del material de apoyo sobre el avance del proyecto.	Luciano Vera
Diseñar fronted	Creación de la interfaz de usuario y la experiencia del usuario.	Luciano Vera
Diseñar backend	Proporcionar los servicios y las funcionalidades necesarias para que la aplicación móvil funcione correctamente.	Andrew Campos

Codificar reconocimiento de voz a texto	-	Vranika Santiago
Depuración de códigos	Proceso en el cual se identificarán y corregirán errores en el algoritmo.	Bastián Vega
Pruebas de funcionamiento	Chequear que la aplicación móvil este cumpliendo su propósito.	Bastián Vega
Redactar manual de usuario	Creación de material para facilitar el uso de la interfaz y robot.	Andrew Campos
Confeccionar póster	Apoyo visual sobre las características del proyecto.	Vranika Santiago
Informe III	Redacción del informe final.	Andrew Campos
Presentación III	Creación del material para la presentación final.	Luciano Vera

Tabla 6: Actividades de trabajo.

3.2.2. Asignación de tiempo

La Carta Gantt fue creada con el propósito de planificar y gestionar de manera más eficaz las actividades a lo largo del semestre. Además, se puede registrar el tiempo dedicado para cada tarea y compararlo con el tiempo estimado.

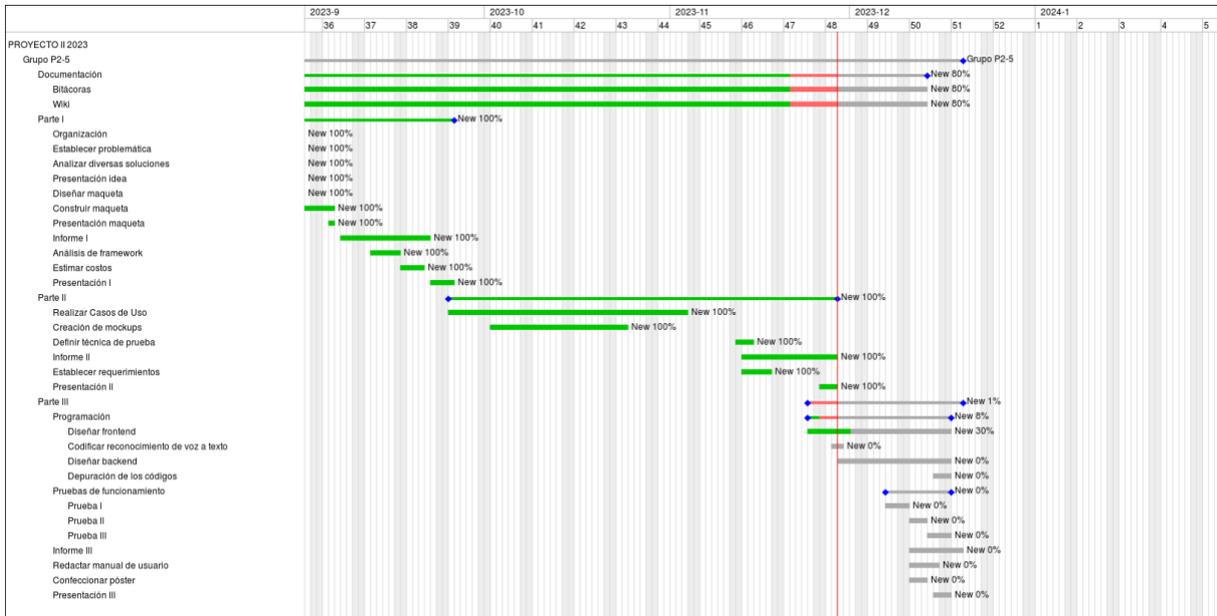


Ilustración 3: Carta Gantt.

El proyecto se dividirá en 3 fases, las cuales son:

- **Parte I:** planificación del proyecto, que está comprendida del 21/08/2023 hasta el 25/09/2023.
- **Parte II:** ejecución del proyecto, que está comprendida del 27/09/2023 hasta el 28/11/2023.
- **Parte III:** cierre del proyecto, que está comprendida del 29/11/2023 hasta el 19/12/2023.

3.3. Planificación de la gestión de riesgos

Para la gestión de riesgos, primeramente, se categorizaron los tipos de riesgos y sus factores:

Tipo de Riesgo	
Tecnológico	Entrega retrasada o problemas relacionados con el hardware o de la ayuda del software, muchos problemas tecnológicos reportados.
Humano	Baja moral del personal, mala salud, malas relaciones entre los miembros del equipo, disponibilidad de empleo.
Herramientas	Rechazo de los miembros del equipo para utilizar herramientas, quejas acerca de las herramientas CASE, peticiones de estaciones de trabajo más potentes.
Requerimientos	Peticiones de muchos cambios en los requerimientos, quejas del cliente.
Organizacional	Chismorreos organizacionales, falta de acciones por el administrador principal.
Estimación	Fracaso en el cumplimiento de los tiempos acordados y en la eliminación de defectos reportados.

Tabla 7: Tipos de Riesgos.

Mientras que para los riesgos latentes que se podrían presentar en el proyecto, se clasificaron en 4 niveles de impacto:

1. Catastrófico
2. Crítico
3. Marginal
4. Despreciable

Riesgo	Tipo	Probabilidad de Ocurrencia	Nivel de Impacto	Acción Remedial
Falta de conocimiento y comprensión de la metodología de trabajo por parte del cliente	Herramientas	40%	Catastrófico	Entregar documentación temática y precisa sobre la metodología en cascada.
Presupuesto afectado por el entorno económico	Estimación	20%	Catastrófico	Solicitar una reunión con el cliente en la que se discuta la situación económica del proyecto y se llegue a un acuerdo en cuanto a la necesidad de un mayor apoyo financiero.
Daño o pérdida en la tarjeta SD	Tecnológico	25%	Catastrófico	Comprar otra tarjeta SD y respaldar la información regularmente.
Detenimientos de las actividades por recesos o paros	Humano	5%	Crítico	Reprogramación de actividades.
Quejas del cliente	Requerimientos	30%	Catastrófico	Trabajar con el cliente para conocer su descontento y encontrar una solución.
Falta de comunicación entre las áreas de trabajo	Organizacional	20%	Crítico	Realizar reuniones periódicas para mantener al equipo informado sobre las actualizaciones en la carta Gantt y la asignación de nuevas tareas.

Atraso en actividades de la Carta Gantt	Estimación	10%	Crítico	Agrupar las tareas atrasadas y darle prioridad por su relevancia en el desarrollo de nuevas tareas.
Enfermedades del personal	Humano	10%	Marginal	Un involucrado se encargará temporalmente de las tareas del personal faltante.
Daños físicos en la Raspberry-Pi3	Tecnológico	20%	Catastrófico	Reemplazar Raspberry-Pi3
Pérdida de avances	Tecnológico	10%	Catastrófico	Realizar respaldos frecuentemente del avance del desarrollo del software.

Tabla 8: Gestión de Riesgos.

4. PLANIFICACIÓN DE LOS PROCESOS TÉCNICOS

4.1. Modelo de Proceso

4.1.1. Especificación de Requerimientos

Requerimientos funcionales

1. El sistema recibe y muestra la información de un chip NFC.
2. El sistema está comunicado a la base de datos (Raspberry).
3. El sistema es capaz de realizar operaciones DML en la base de datos.
4. El sistema es capaz de asignar y retornar una ID a cada instancia automáticamente.
5. El sistema es capaz de reproducir mensajes de texto a voz.

Requerimientos no funcionales

1. La aplicación debe estar programada en JavaScript ES12.
2. La aplicación debe ser hecha con el framework Ionic.
3. La base de datos debe ser programada con Python 3.12.0.
4. El sistema operativo de la Raspberry debe ser Raspbian.
5. La aplicación debe reconocer el texto correctamente para reproducirlo en voz.
6. La aplicación debe ser instalada en un dispositivo Android.
7. La aplicación debe ser de fácil acceso y manipulación para el usuario.
8. El dispositivo para la aplicación debe tener activada la opción de Google Talk-Back.
9. La base de datos debe ser hecha con SQL.
10. La base de datos debe estar disponible en una red local.

4.1.2. Descripción de la Arquitectura

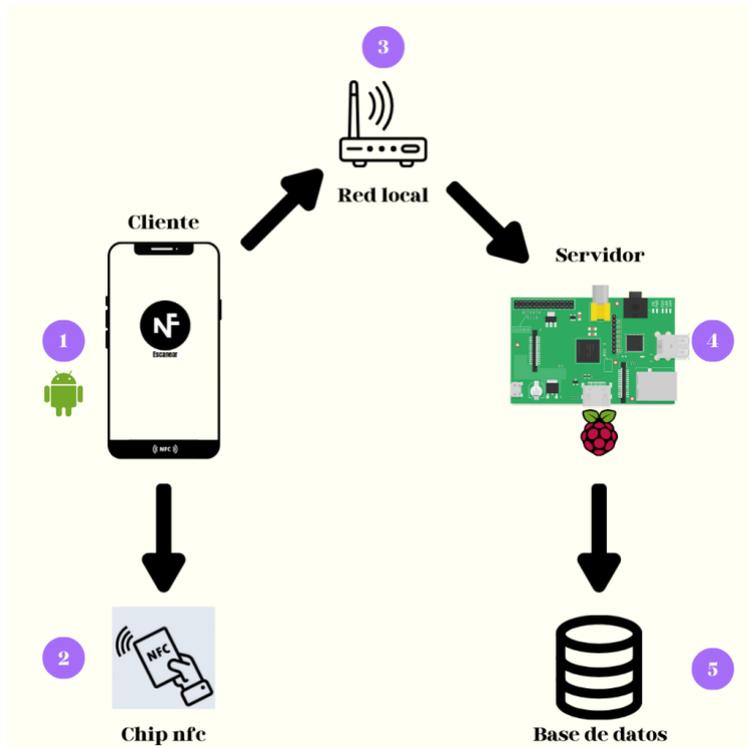


Ilustración 4: Descripción de la Arquitectura.

1. Software (cliente) diseñado para el usuario con discapacidad visual que permitirá acceder al sensor NFC integrado en el dispositivo, con el que se podrá escanear la tarjeta nfc.
2. Tarjeta NFC NTAG215 asignada a un objeto, contendrá el identificador del objeto almacenado a la base de datos.
3. Red local WIFI que permitirá a la aplicación realizar solicitudes HTTP al servidor (Raspberry-Pi3) para recibir o enviar información.
4. Servidor alojado en una Raspberry-Pi3. Maneja solicitudes HTTP para recibir o emitir datos, y controla la base de datos.
5. Base de datos dentro de la Raspberry-Pi3 que almacena las instancias de los objetos.

4.1.3. Diseño de Interface de Usuario



Ilustración 5: Principal.

Al momento de iniciar la aplicación móvil (pantalla “Principal”), aparece un botón en medio de la pantalla que dice “Escanear” (1), el cual se debe presionar para escanear un objeto ya registrado (*Ilustración 4*).

En caso de encontrarse en la pantalla “Registrar objeto” o “Lista de objetos” y el usuario desee acceder a la pantalla principal, debe deslizar hasta que Google Talk-Back le indique “Se encuentra en la pantalla principal”.



Ilustración 6: Registrar objeto.

En la pantalla “Registrar objeto”, primeramente, se tiene una pantalla con las etiquetas “Título” (1), la cual en una caja de entrada de texto se ingresa el título del objeto. En segundo lugar, está la etiqueta “Descripción” (2), donde se ingresa la descripción del objeto a agregar. Por último, se tiene una etiqueta denominada “Tipo” (3), donde en un combo box se presentan diferentes opciones para clasificar al objeto. Cada información por agregar es comunicada por el usuario mediante su voz y para ser guardada, se encuentra el botón “Escanear” (4) (*Ilustración 5*).

En caso de encontrarse en la pantalla “Principal” o “Lista de objetos”, y el usuario desee acceder a la pantalla “Registrar objeto”, debe deslizar hasta que Google Talk-Back le indique “Se encuentra en la pantalla registrar objeto”.



Ilustración 7: Lista de objetos.

En la pantalla “Lista de objetos”, por medio de la reproducción de audio, cada objeto registrado se listará por su título. Por cada objeto, se ofrecerán dos opciones: Eliminar (1) o Ver detalle (2), las cuales deben ser comunicadas al dispositivo móvil mediante la voz del usuario.

En caso de ser escogida la opción “Eliminar [título del objeto]”, el objeto será removido de la lista. De lo contrario, si es escogida la opción “Ver detalle” véase Ilustración 7.

En caso de encontrarse en la pantalla “Principal” o “Registrar objeto”, y el usuario desee acceder a la pantalla “Lista de objetos” debe deslizar hasta que Google Talk-Back le indique “Se encuentra en la pantalla lista de objetos”.



Ilustración 8: Lista de objetos - Ver detalle.

En caso de haber sido escogida la opción “Ver detalle [título del objeto]”, mediante la reproducción de audio, se indicará la descripción y el tipo de objeto (1).

4.1.4. Caso de Uso de Contexto

En el diseño grueso se hace mención a la etapa inicial del proceso de diseño de software, el cual permite definir los componentes principales y las interacciones clave del sistema. En el caso de este trabajo, el enfoque estará dirigido en crear diagramas secuenciales de nivel 0 y 1 para visualizar las interacciones entre los actores y el sistema en diferentes escenarios en base a los casos de usos definidos previamente. Estos diagramas proporcionarán una perspectiva detallada de las acciones y respuestas del sistema, lo que facilitará la identificación de puntos clave de interacción y flujos de información.

Asimismo, se emplearán los diagramas de clases para representar la estructura estática del sistema. Estos diagramas funcionarán como una herramienta para facilitar la visualización de las clases, sus atributos y las relaciones entre ellas.

A continuación, se muestra el caso de uso de contexto, donde se puede visualizar la interacción del usuario con discapacidad visual con el sistema y la Raspberry-Pi3.

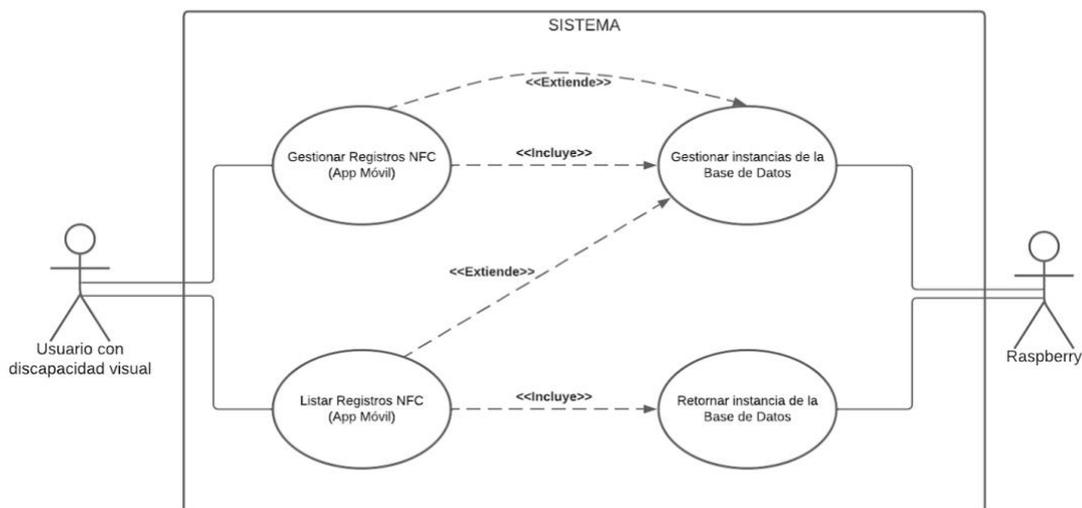


Ilustración 9: Diagrama Caso de Uso de Contexto.

4.1.5. Caso de Uso: Listar registros NFC (App móvil)

C U S: Listar Registros NFC (app móvil)	
Autor / Fecha: Andrew Campos / 24 - 10 - 2023	
Descripción: Se listan todas las instancias registradas en la base de datos	
Actor: Usuario con discapacidad visual.	
Precondición: Instancias para los chips NFC ya existentes en la base de datos y base de datos operativa y tener una comunicación entre la app.	
Flujo Principal: Usuario	Flujo Principal: Sistema 1. <<Incluye C U S: Retornar instancia de la Base de Datos>> 2. Indica por cada instancia: <ul style="list-style-type: none"> • Título • Opción "Ver detalles" • Opción "Eliminar"
Flujo Alternativo: "Ver detalles" 3.1. Selecciona la opción "Ver detalles" de una instancia en específico.	Flujo Alternativo: 3.1.1 Muestra la siguiente información de la instancia: <ul style="list-style-type: none"> • ID • Título • Categoría • Descripción
Flujo Alternativo: "Eliminar" 3.2 Selecciona la opción "Eliminar" de una instancia en específico.	Flujo Alternativo: 3.2.1 <<Extiende C U S: Gestionar Registros NFC (Eliminar Instancia)>>
Postcondiciones: -	

Tabla 9: CUS: Listar registros NFC (App Móvil).

4.1.5.1. Diagrama de Secuencia: Nivel 0

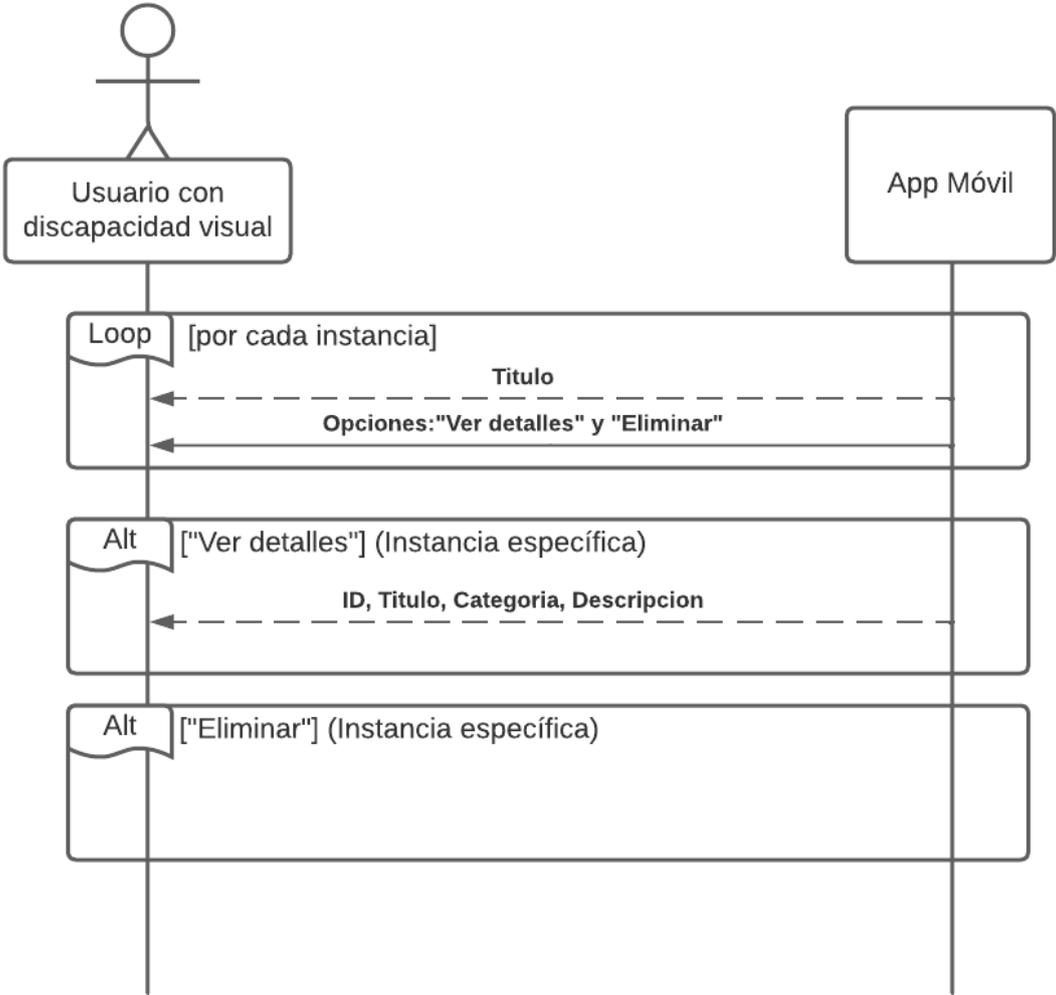


Ilustración 10: Listar Registros NFC (App Móvil) - Diagrama nivel 0.

4.1.5.2. Diagrama de Secuencia: Nivel 1

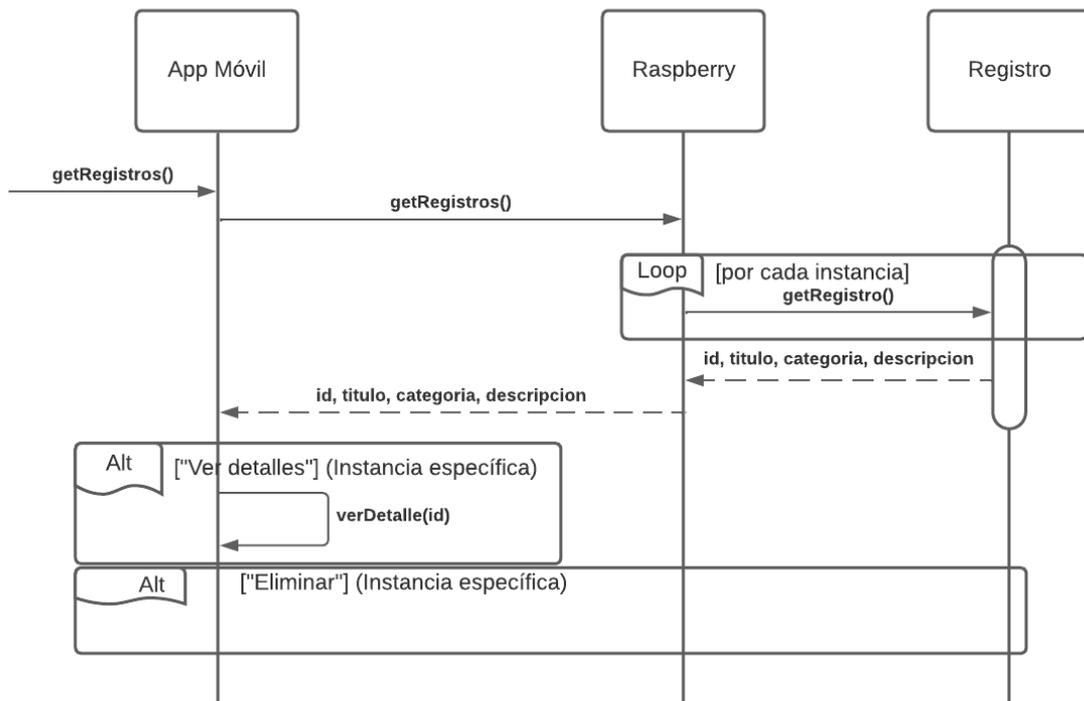


Ilustración 11: Listar registros NFC (App Móvil) - Diagrama nivel 1.

4.1.6. Caso de Uso: Gestionar registros NFC (App móvil)

C U S: Gestionar registros NFC (app móvil)	
Autor / Fecha: Andrew Campos / 30 - 10 - 2023	
Descripción: Se gestionan los registros de los elementos NFC con una instancia de la base de datos.	
Actor: Usuario con discapacidad visual.	
Precondición: Base de datos en funcionamiento.	
Flujo Principal: Rellenar elemento NFC 2. Ingresar los datos solicitados.	Flujo Principal: Sistema 1. Solicita los siguientes datos del objeto: <ul style="list-style-type: none"> • Título • Categoría • Descripción 3. Indica que la instancia se ha creado exitosamente, se genera y muestra el ID generado para la instancia. 4. Agrega el ID al chip NFC 5. <<Incluye C U S: Gestionar Instancias de la Base de Datos (Crear Instancia)>>.
Flujo Alternativo: Modificar información del elemento NFC 3.1.3. Ingresar "Si".	Flujo Alternativo: 3.1.1. Agrega el ID del chip relacionado. 3.1.2. Indica: "¿Está seguro de reescribir esta información? (Si/No)". 3.1.4. <<Extiende C U S: Gestionar Instancias de la Base de Datos (Modificar Instancia)>>.
Flujo Alternativo: Eliminar instancia 3.2.3. Ingresar "Si".	Flujo Alternativo: 3.2.1. Agrega el ID del chip relacionado. 3.2.2. Indica: "¿Está seguro de eliminar esta información? (Si/No)". 3.2.4. <<Extiende C U S: Gestionar Instancias de la Base de Datos (Eliminar Instancia)>>.
Postcondiciones: Nuevo estado en la base de datos	

Tabla 10: CUS: Gestionar registros NFC (App Móvil).

4.1.6.1. Diagrama de Secuencia: Nivel 0

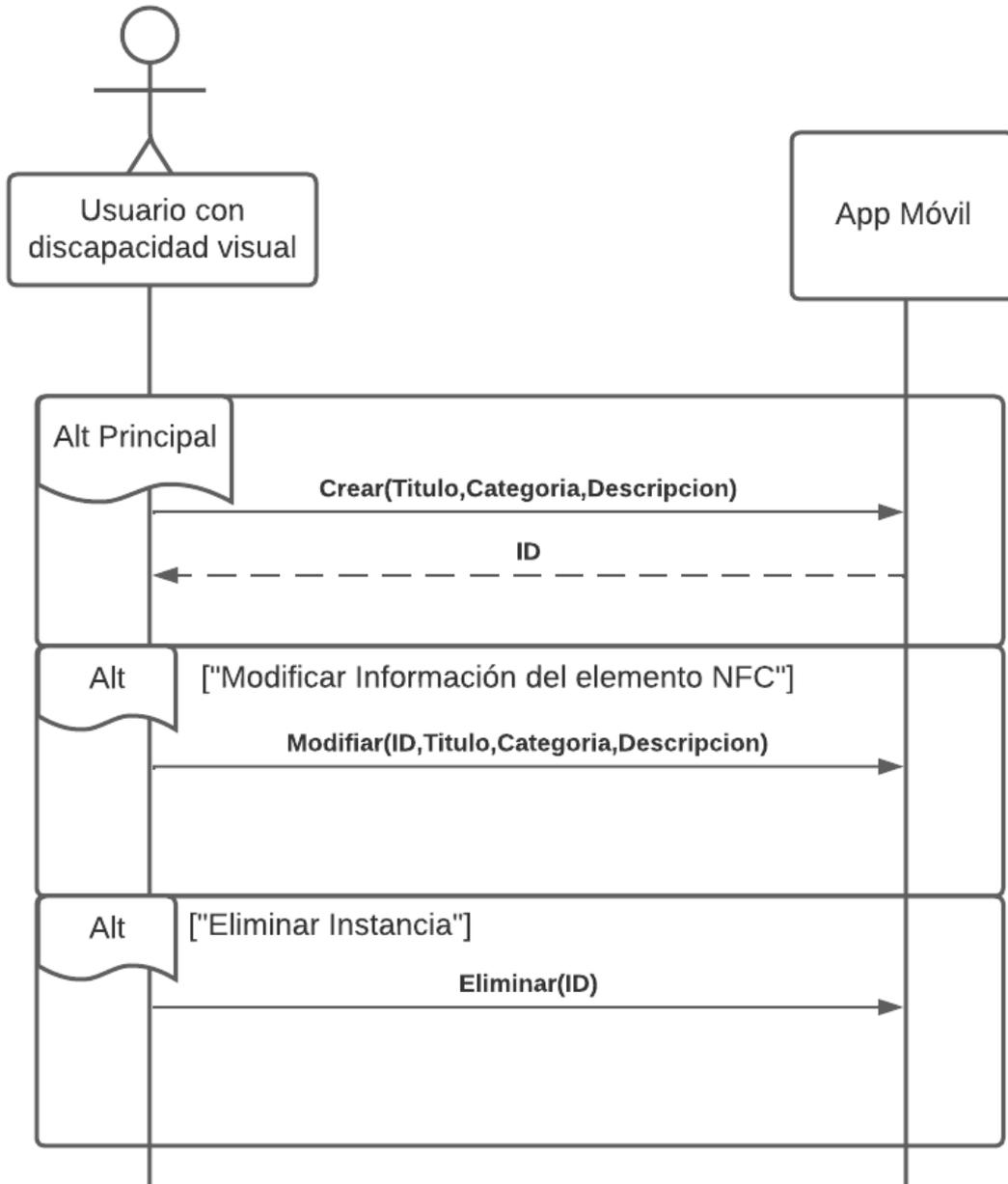


Ilustración 12: Gestionar registros NFC (App Móvil) - Diagrama nivel 0.

4.1.6.2. Diagrama de Secuencia: Nivel 1

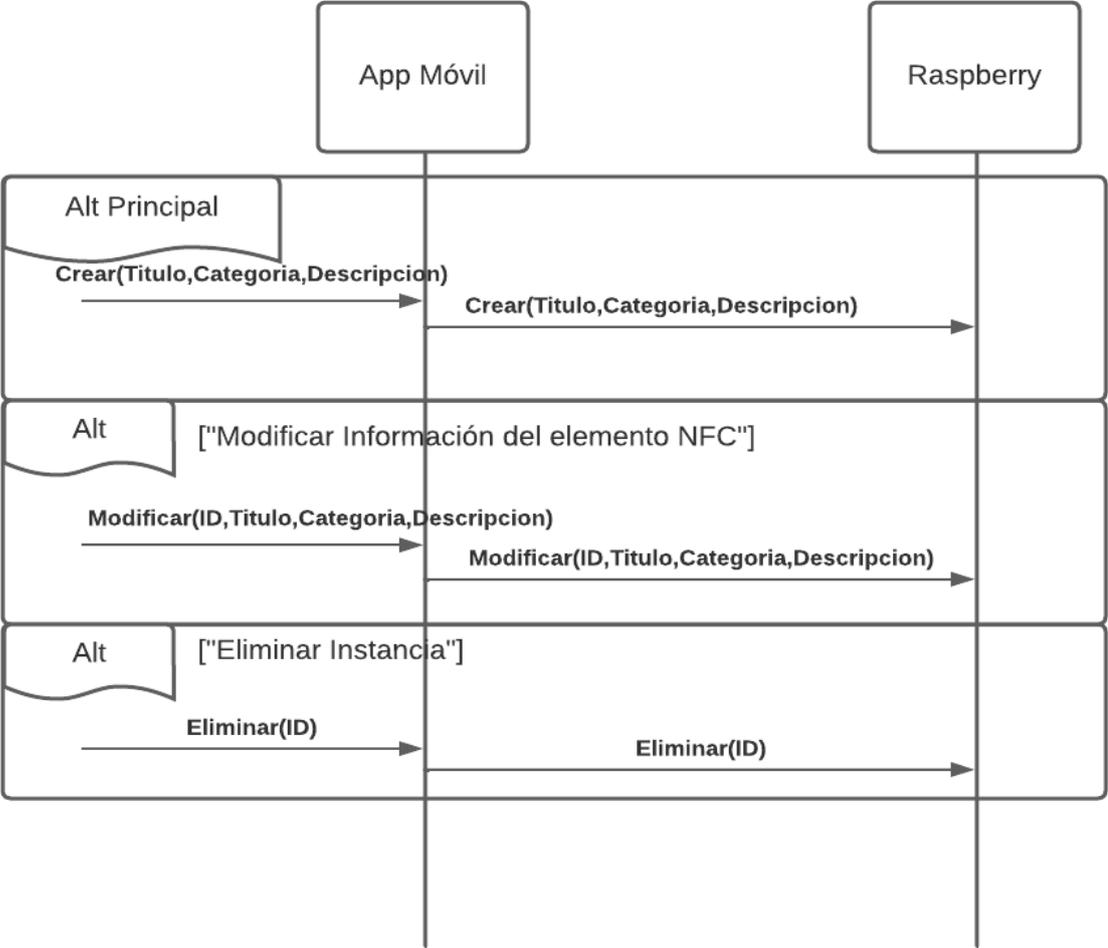


Ilustración 13: Gestionar registros NFC - Diagrama nivel 1.

4.1.7. Caso de Uso: Gestionar instancias de la base de datos

C U S: Gestionar Instancias de la Base de Datos	
Autor / Fecha: Andrew Campos / 2 - 11 - 2023	
Descripción: Este caso de uso existe para las transacciones que se realizan en la base de datos.	
Actor: Aplicación Móvil	
Precondición: Base de datos en funcionamiento	
<p>Flujo Principal: Crear Instancia</p> <p>1. Ingresar:</p> <ul style="list-style-type: none"> • Título • Categoría • Descripción 	<p>Flujo Principal: Sistema</p> <p>2. Acepta la Información recibida y crea una instancia con un ID único y propio (el cual lo retorna)</p>
<p>Flujo Alternativo: Modificar Instancia</p> <p>1.1.1. Ingresar:</p> <ul style="list-style-type: none"> • ID • Título • Categoría • Descripción 	<p>Flujo Alternativo:</p> <p>1.1.2. Modificada la información que contiene la instancia con el ID idéntico al ingresado</p>
<p>Flujo Alternativo: Eliminar Instancia</p> <p>1.2.1. Ingresar ID</p>	<p>Flujo Alternativo:</p> <p>1.2.2. Busca la instancia que tenga su ID idéntico al ingresado y elimina la instancia de la base de datos</p>
Postcondiciones: Nuevo estado de la base de datos	

Tabla 11: CUS: Gestionar instancias de la Base de Datos.

4.1.7.1. Diagrama de Secuencia: Nivel 0

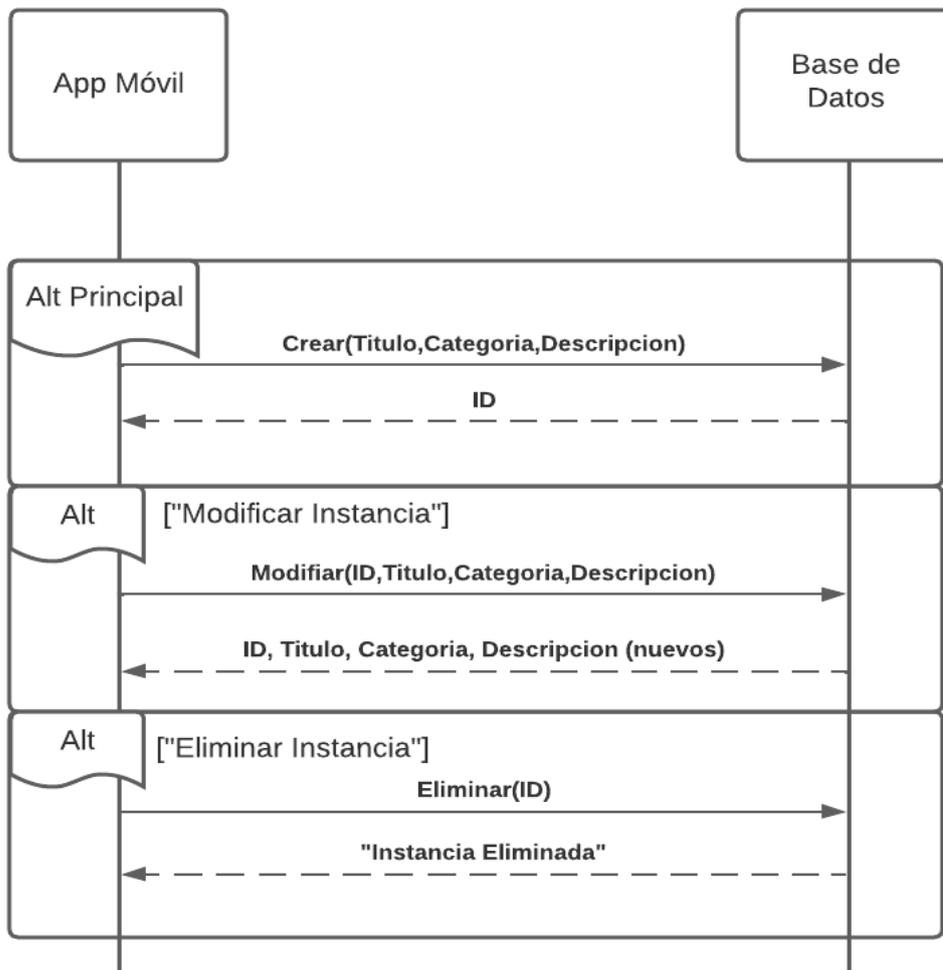


Ilustración 14: Gestionar instancias de la Base de Datos - Diagrama nivel 0.

4.1.7.2. Diagrama de Secuencia: Nivel 1

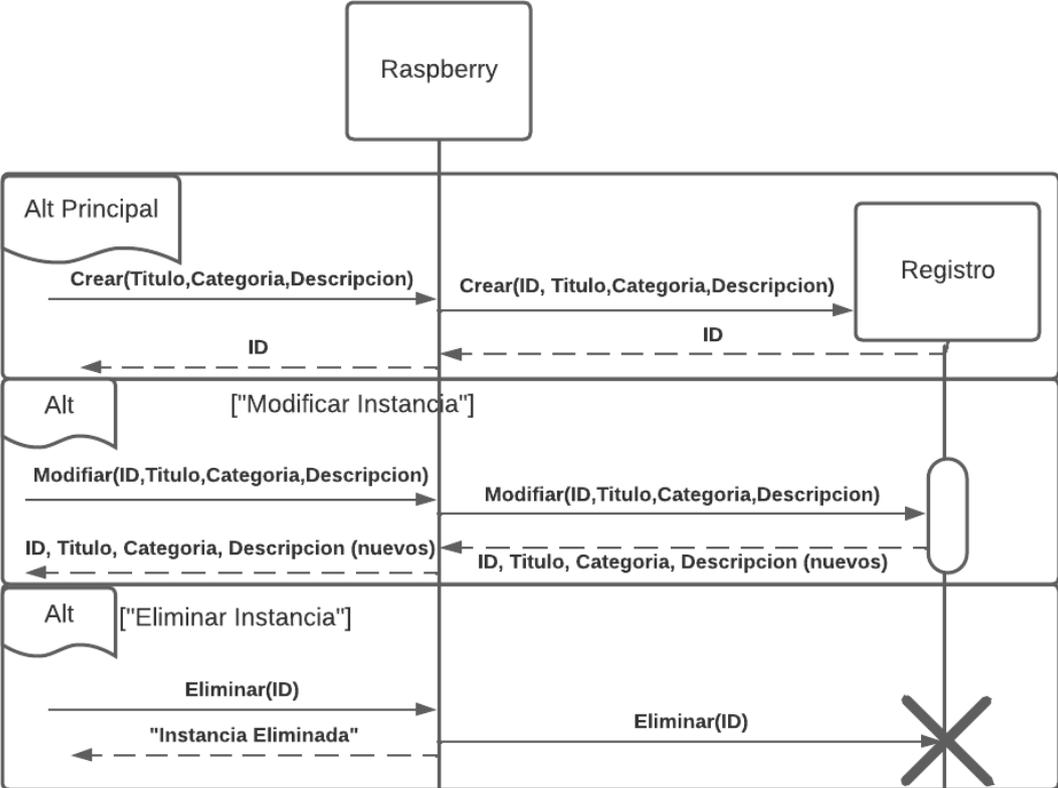


Ilustración 15: Gestionar instancia de la Base de Datos - Diagrama nivel 1.

4.1.8. Caso de Uso: Retornar instancias de la base de datos

C U S: Retornar Instancia de la Base de datos	
Autor / Fecha: Andrew Campos / 01 - 11 - 2023	
Descripción: Este caso de uso existe para describir la comunicación que tiene la aplicación móvil del proyecto con la base de datos presente en la raspberry pi	
Actor: Aplicación Móvil	
Precondición: Base de datos en funcionamiento	
Flujo Principal: Aplicación móvil 1. Solicita la información referente al ID	Flujo Principal: Sistema 2. Retorna: <ul style="list-style-type: none"> • ID • Título • Categoría • Descripción
Postcondiciones: -	

Tabla 12: CUS: Retornar instancias de la Base de datos.

4.1.8.1. Diagrama de Secuencia: Nivel 0

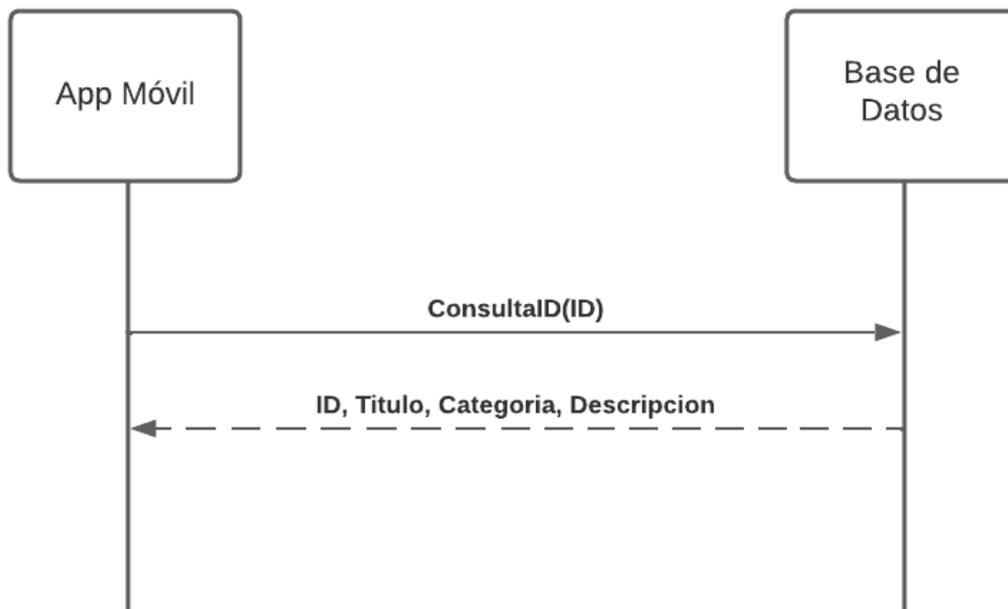


Ilustración 16: Retornar instancia de la Base de Datos - Diagrama nivel 0.

4.1.8.2. Diagrama de Secuencia: Nivel 1

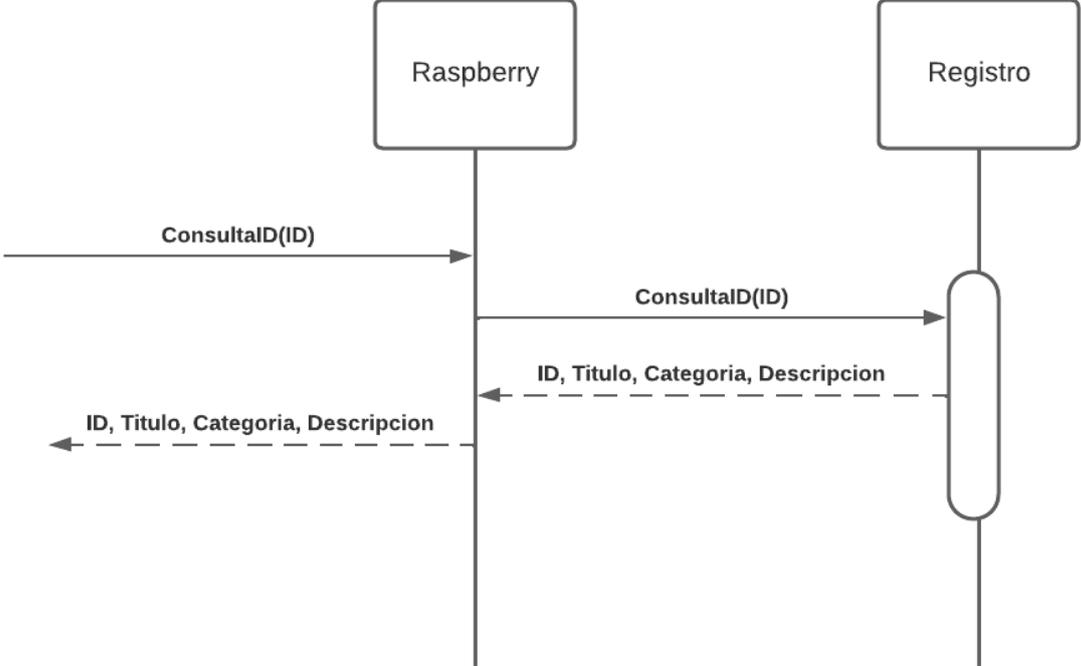


Ilustración 17: Retornar instancia de la Base de Datos - Diagrama nivel 1.

4.1.9. Diagrama de clases

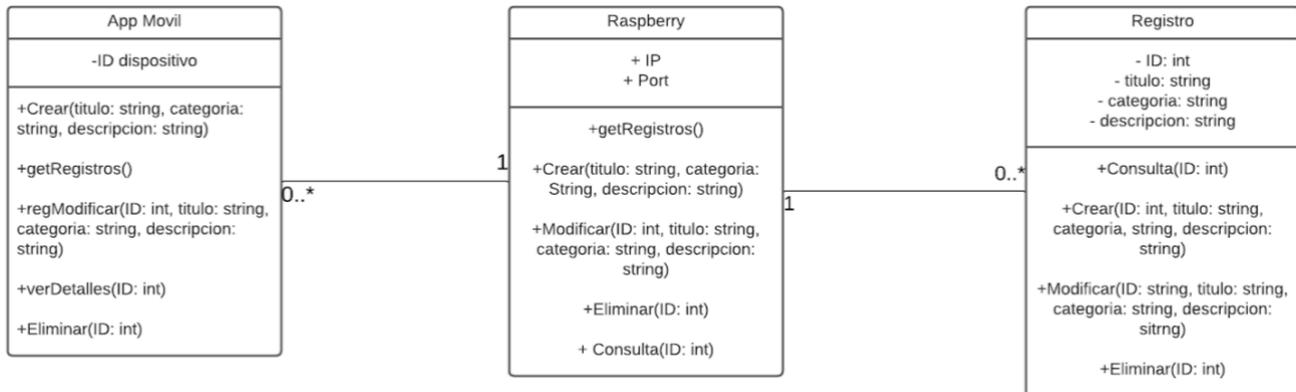


Ilustración 18: Diagrama de clases.

4.2. Herramientas y técnicas

Las herramientas a utilizar para el desarrollo del servidor (Raspberry Pi - Flask):

1. *Python*: lenguaje de programación principal para el desarrollo del servidor.
2. *Flask*: framework web ligero para Python que facilita la creación de aplicaciones web.
3. *Flask-SQLAlchemy*: extensión de Flask que proporciona integración con SQLAlchemy, una biblioteca de mapeo objeto-relacional (ORM) para Python.
4. *SQLite (o MySQL/PostgreSQL)*: base de datos que puedes utilizar para almacenar la información en el servidor. SQLite es una opción liviana y fácil de usar para proyectos más pequeños.
5. *Flask-CORS*: Extensión de Flask para manejar los encabezados CORS (Cross-Origin Resource Sharing) y permitir solicitudes desde dominios diferentes al del servidor.

Para el desarrollo de la Aplicación Móvil (Ionic):

1. *Node.js y npm*: Node.js es necesario para instalar y ejecutar herramientas relacionadas con el desarrollo web, y npm es el gestor de paquetes de Node.js.
2. *Ionic Framework*: framework para el desarrollo de aplicaciones móviles con tecnologías web (Angular, HTML, CSS).
3. *Angular*: framework de desarrollo web utilizado por Ionic para construir la parte frontend de la aplicación móvil.
4. *Cordova*: plataforma para el desarrollo de aplicaciones móviles híbridas que permite acceder a las API nativas del dispositivo desde JavaScript.
5. *HttpClient (Angular)*: módulo de Angular para realizar solicitudes HTTP desde la aplicación móvil.

Herramientas Adicionales:

1. *Visual Studio Code (VSCode)*: editor de código ligero y potente que es compatible con el desarrollo de proyectos web y móviles.
2. *Git*: sistema de control de versiones para el seguimiento de cambios en el código.
3. *SSH (Secure Shell)*: para acceder a la Raspberry Pi de forma remota desde una máquina de desarrollo.
4. *SCP/SFTP*: para transferir archivos entre una máquina de desarrollo y la Raspberry-Pi.
5. *Sensor activo⁵: Dispositivo móvil con NFC. Puede generar señales NFC para comunicarse con otros dispositivos NFC, como tarjetas, etiquetas o incluso otros teléfonos móviles.*
6. *Sensor pasivo⁶: Chip NFC NTAG215. No generan activamente señales, pero responden cuando son alimentadas por la energía de la señal NFC emitida por el dispositivo activo, como un teléfono móvil.*

La técnica a utilizar en el proyecto para realizar pruebas y ver los defectos que tiene la aplicación móvil es la de extremo a extremo. Según el sitio web parasoft.com, metodología de prueba que emplean los probadores para probar el flujo de trabajo de una aplicación de principio a fin. El objetivo de las pruebas de extremo a extremo es validar la integración del sistema y asegurarse de que los datos estén seguros. [4]

⁵ Mundo Sensor (s.f.). *Sensores activos y pasivos* - Accedido el 27 de noviembre de 2023. <https://mundosensor.blogspot.com/2014/11/sensores-activos-y-pasivos.html>

⁶ Krypton Solid (s.f.). *¿Qué es sensor activo?* - Definición de Krypton Solid - Accedido el 27 de noviembre de 2023. <https://mundosensor.blogspot.com/2014/11/sensores-activos-y-pasivos.html>

5. CONCLUSIONES

En conclusión, respondiendo a la pregunta inicial, el IoT puede utilizarse para mejorar la calidad de vida de las personas no videntes gracias al desarrollo de una aplicación móvil. Se logró planificar un proyecto para desarrollar una aplicación móvil asistencial enfocada a personas con discapacidad visual que pueda reconocer y detectar agentes externos en el entorno hogareño. Además, se analizaron los diversos desafíos que tienen las personas con discapacidad visual con el propósito de solucionar un problema específico en sus vidas.

Para esto, por un lado, se trabajó con una persona no vidente y en base a ello, se logró identificar un problema y encontrarle una solución; se desarrolló el diseño de una aplicación móvil utilizando herramientas tales como IoT, chips NFC y Raspberry-Pi. Por otro lado, se establecieron las suposiciones y restricciones de este proyecto, lo que ha proporcionado una mayor claridad sobre los desafíos que se enfrentaran durante el desarrollo de la fase dos del proyecto.

La planificación de procesos técnicos consistió en especificar los requerimientos, describir su arquitectura, describir el diseño de la interfaz del usuario y exhibir paso a paso los casos de uso. En cuanto a los casos de uso de contexto se utilizaron diagramas secuenciales de nivel 0 y 1 para representar la estructura estática del sistema y mostrar una perspectiva detallada de las acciones y respuestas de este. Mientras que el identificar la diferencia entre los requerimientos no funcionales y funcionales garantizó que la aplicación móvil fuese más efectiva y segura.

La necesidad de crear nuevas y más innovadoras formas de mejorar la calidad de vida humana es una tarea ardua. La importancia que recae en las nuevas tecnologías en concordancia con la creatividad humana es la clave para ayudar a grupos humanos específicos y que sus necesidades no sean olvidadas. Este proyecto es una iniciativa más en la proyección de una mejor calidad de vida para todos, aunque se enfoque solamente en una comunidad. Sin embargo, es relevante destacar que existen un sinnúmero de otros problemas para esta y muchas otras comunidades que pueden ser resueltos con la ayuda de la tecnología e ingenio humano.

REFERENCIAS

- [1] A. Beltramen, «Prototipado rápido de proyectos IoT sin programación,» SEDICI - Repositorio de la Universidad Nacional de La Plata, Septiembre 2018. [En línea]. Available: <http://sedici.unlp.edu.ar/handle/10915/71753>.
- [2] G. R. Pontiggia, «Las soluciones tecnológicas que mejoran la vida de las personas ciegas,» El Dínamo, 28 Abril 2023. [En línea]. Available: <https://www.eldinamo.cl/buen-dato/2023/04/28/las-soluciones-tecnologicas-que-mejoran-la-vida-de-las-personas-ciegas/>.
- [3] SAP, «¿Qué es el internet de las cosas (IoT)?,» SAP, [En línea]. Available: <https://www.sap.com/latinamerica/products/artificial-intelligence/what-is-iot.html>.
- [4] “Las pruebas de extremo a extremo mejoran la calidad del software”. Parasoft. Accedido el 28 de noviembre de 2023. [En línea]. Available: <https://es.parasoft.com/solutions/end-to-end-testing/#:~:text=La%20prueba%20de%20extremo%20a,que%20los%20datos%20estén%20seguros>