



Universidad Tecnológica Nacional
Facultad Regional Santa Fe

Proyecto Final

Informe:

“Prototipo de software destinado a la carga,
visualización y gestión de denuncias
ciudadanas para fuerzas de seguridad de la
provincia de Entre Ríos”

Alumnos: Muñoz, Sonia Antonella
Ronchi, Hugo Nicolás

Carrera: Ingeniería en Sistemas de Información

Director: Ing. Ledesma, Rodrigo

Docentes: Bracalenti, Claudio
Villarreal, Pablo
Ale, Mariel

Año 2020

Índice

Contenido

Índice	1
1. Introducción	5
1.1 Objetivo general	5
1.2 Objetivos específicos.....	5
1.3 Alcance del proyecto	6
1.4 Estructura del informe final	6
2. Terminología del modelo de negocio	7
2.1 Ciudadano	7
2.2 Usuario	7
2.3 Denuncia	7
2.4 Dependencia policial	7
2.5 Dispositivo	8
3. Herramientas y tecnologías a utilizar en el proyecto	8
3.1 Servidor.....	8
3.2 Aplicación Android Nativa	8
3.3 Servicio Web	8
3.4 Control de versiones de código fuente.....	9
3.5 Entorno de desarrollo integrado (IDE)	9
3.6 Librerías de código fuente	9
4. Cambios en la planificación del proyecto	9
4.1 Cambios realizados en actividades y tareas planificadas	10
4.2 Cambios realizados en la implementación de la solución	11
4.3 Gestión de riesgos del proyecto	12
4.3.1 Riesgo en planificación del proyecto	13
4.3.2 Riesgo en definición de requerimientos.....	13
4.3.3 Riesgos imprevistos	13
5. Investigación y análisis	14
5.1 Entrevistas.....	14
5.2 Lectura de leyes y resoluciones.....	15
5.3 Selección de metodología de trabajo.....	18
5.4 Requerimientos de la solución.....	19
5.4.1 Historias de usuario	21

5.4.2 Tareas.....	28
6. Diseño de base de datos.....	41
6.1 Definir los datos necesarios.....	41
6.2 Realizar un diagrama Entidad-Relación y de tablas de base de datos.....	44
6.2.1 Tabla usuarios	45
6.2.2 Tabla roles	46
6.2.3 Tabla permisos	46
6.2.4 Tabla revisiones.....	46
6.2.5 Tabla categorias_denuncia	47
6.2.6 Tabla subcategorias_denuncia.....	47
6.2.7 Tabla tipo_denuncia_app	48
6.2.8 Tabla dispositivos.....	48
6.2.9 Tabla denuncias.....	49
6.2.10 Tabla activity_log	50
6.2.11 Tabla password_reset.....	51
6.2.12 Tablas intermedias.....	51
7. Tecnologías y arquitectura	52
7.1 Arquitectura del proyecto.....	52
7.1.1 Objetivos y restricciones	52
7.1.2 Vista de módulos.....	53
7.1.3 Vista de Componente y conector	58
7.2 Tecnologías utilizadas para la solución	61
7.3 Motor de base de datos	61
7.4 Lenguajes de desarrollo de software	62
7.4.1 Sistema web y API	62
7.4.2 Aplicación móvil	62
7.5 Librería para gráficos.....	63
7.6 Librería para mapas	63
7.7 Herramientas adicionales	64
7.7.1 Visual Studio Code.....	64
7.7.2 Node.js.....	64
7.7.3 Postman.....	64
7.7.4 Git y GitLab	64
7.7.5 Android Studio	65
8. Implementación.....	65

8.1 Sprints	65
8.1.1 Sprint Planning.....	65
8.1.2 Sprint 0	65
8.1.3 Sprint 1	66
8.1.4 Sprint 2	68
8.1.5 Sprint 3	69
8.2 Instalación de tecnologías y herramientas	70
8.2.1 Sistema web	70
8.2.2 API del proyecto.....	70
8.2.3 Aplicación móvil	70
8.3 Definición de módulos de la solución.....	71
8.3.1 Módulos sistema web.....	71
8.3.2 Módulos aplicación móvil	71
8.4 Creación de la base de datos	72
8.5 Creación de tablas en la base de datos.....	72
8.6 Generación de código fuente y documentación	72
8.6.1 Aplicación móvil	73
8.6.2 API.....	73
8.6.3 Sistema Web.....	74
8.7 Verificación y corrección de errores de los productos de la solución	75
8.7.1 Pruebas funcionales de módulos del sistema web	76
8.7.2 Metodología para pruebas de la API	79
8.8 Validación de la solución con el cliente.....	82
9. Funcionalidades principales	82
9.1 Aplicación móvil.....	82
9.1.1 Realizar denuncia	82
9.1.2 Mapa de dependencias policiales	84
9.1.3 Listado de dependencias policiales.....	85
9.1.4 Lógica para la identificación de dispositivos	86
9.2 Sistema web.....	87
9.2.1 Auditar denuncias	87
9.2.2 Analizar denuncias.....	89
9.2.3 Reportes	89
9.2.4 Estadísticas.....	90
9.2.5 Mapas	92

9.2.6 Gestión	93
9.2.7 Política de baneo de dispositivos	95
10. Conclusiones	96
11. Anexos.....	97
11.1 Anexo 1: Entrevistas.....	97
11.1.1 Entrevista stakeholder.....	97
11.1.2 Entrevista Ciudadano (20 - 30 años).....	99
11.1.3 Entrevista Ciudadano (30 - 40 años).....	101
11.1.4 Entrevista Ciudadano (50 - 60 años).....	102
11.2 Anexo 2: Diagrama de Entidad-Relación	104
11.3 Anexo 3: Diagrama de Base de Datos.....	105
11.4 Anexo 4: Pruebas funcionales	106
11.4.1 Aplicación móvil	106
11.4.2 Sistema web	109
11.4.3 API.....	123
12. Bibliografía	127

1. Introducción

El presente proyecto surge de la necesidad de disminuir la cantidad de hechos delictivos no denunciados por los ciudadanos. Ésta es una problemática que preocupa a las fuerzas de seguridad, debido a que afecta la gestión y asignación de recursos a una zona geográfica como consecuencia de la falta de información sobre la misma.

Es por ello que se le propuso a la Policía de Entre Ríos, donde Sonia Muñoz se desempeña, el desarrollo de una solución informática que permita tratar dicha problemática; los mismos se mostraron interesados en la idea y nos brindaron su apoyo para llevar adelante el proyecto.

Cabe destacar que el concepto de “denuncia ciudadana a través de un sistema informático” no es una temática innovadora; existe, por ejemplo, un formulario web en el sitio del Ministerio de Seguridad de la Nación, pero el mismo es precario en cuanto a la cantidad de datos que maneja y el curso de acción de la denuncia es lento.

Las ventajas que nuestra solución ofrece, en comparación con sistemas de la misma temática, es la de garantizar el anonimato de las denuncias, debido a que no se registra información de los usuarios, y la creación de una base de conocimiento sobre información útil para el ciudadano con temática relacionada a la seguridad. Por otro lado, para la Policía de Entre Ríos, brinda la posibilidad de generar informes, estadísticas y mapas que den soporte a la toma de decisiones para la asignación de los recursos disponibles.

1.1 Objetivo general

El objetivo general del presente proyecto final de carrera es el desarrollo de un sistema de denuncia ciudadana, con el fin de orientar al usuario ante un ilícito llevado a cabo contra su persona (o contra terceros) y generar, para uso de las fuerzas de seguridad, reportes estadísticos y geográficos que den soporte a la toma de decisiones en materia de políticas de seguridad a implementarse en el ámbito local.

1.2 Objetivos específicos

Para lograr el objetivo general detallado previamente, se proponen los siguientes objetivos específicos:

- Investigar y analizar protocolos utilizados por las fuerzas de seguridad para recibir denuncias. Esto comprende la documentación que se genera en el procesamiento y almacenamiento de la misma.
- Recopilar y seleccionar los datos que se utilizarán para los reportes a generar en base a los requerimientos previamente capturados.
- Definir la arquitectura y tecnología a utilizar en la implementación del proyecto en base a las características del mismo, recursos disponibles y conocimientos adquiridos. Investigar y definir las herramientas tecnológicas (lenguajes de programación, frameworks, librerías, etc.) que se consideran necesarias para la implementación del proyecto.
- Implementar el sistema con la información y los conocimientos obtenidos en las etapas anteriores del proyecto.

1.3 Alcance del proyecto

El presente proyecto surge de la necesidad de brindarle al ciudadano una herramienta que facilite el proceso de realización de una denuncia policial por su parte. Sin embargo, debido a la magnitud del mismo, consideramos que, a fines académicos, la implementación de un prototipo de software es lo más adecuado.

Esta decisión tiene su fundamento en los procesos políticos y burocráticos (convenios, resoluciones, etc.) necesarios para llevar a cabo el despliegue del producto final. Además, el prototipado nos permite utilizar datos no reales para mostrar el funcionamiento del sistema y no utilizar información verídica, que el ciudadano brinda, sin su consentimiento. Desde un punto de vista técnico, la construcción de un prototipo evita el proceso de migración de datos, como así también, la integración con sistemas existentes

1.4 Estructura del informe final

En el capítulo 2 se brindan al lector conceptos básicos para entender la terminología utilizada en el proyecto; el capítulo 3 introduce las herramientas y tecnologías utilizadas; el capítulo 4 detalla la nueva planificación que se realizó respecto al plan de proyecto original; el capítulo 5 recopila toda la investigación realizada; en el capítulo 6 se definieron los datos a utilizar en los productos a realizar; en el capítulo 7 se describen las tareas asociadas a las elecciones de tecnologías y arquitectura de la solución; en el capítulo 8 se describirán las tareas asociadas a la implementación del proyecto; el capítulo 9 brinda una descripción de las

principales funcionalidades desarrolladas; por último, el capítulo 10 recopila las conclusiones que obtuvimos de todo el proceso.

2. Terminología del modelo de negocio

En este capítulo serán definidos conceptos relacionados al proyecto que el lector podrá encontrar en el transcurso del informe final. Cabe destacar que no serán definiciones gramaticales de los términos, sino las interpretaciones de los mismos que fueron utilizadas para realizar el trabajo.

2.1 Ciudadano

Con este término determinamos a todas las personas que utilizarán la aplicación móvil de denuncia ciudadana. Un funcionario policial que utilice dicha aplicación, también será considerado un ciudadano.

2.2 Usuario

Término que hace referencia al personal policial que utilizará el sistema web que permite procesar las denuncias ciudadanas cargadas a través de la aplicación móvil. Decidimos utilizar esta distinción (a pesar de que un ciudadano también es un usuario de la aplicación móvil) con el fin de facilitar al lector la distinción entre los dos productos.

2.3 Denuncia

Utilizamos este término para definir a todos los hechos cargados en la aplicación móvil por los ciudadanos. Cabe destacar que el mismo no es homónimo a una denuncia policial, debido a su forma de realización, su anonimato obligado y que puede no desencadenar una investigación, sino ser tomada como soporte para el análisis de políticas de seguridad implementadas o a implementar. Es decir, inicialmente puede no tener la misma relevancia que una denuncia policial realizada en una comisaría, aunque esto puede cambiar de acuerdo al contenido que presente el hecho denunciado.

2.4 Dependencia policial

Utilizamos este término para referirnos no solamente a comisarías, sino a todo lugar que esté bajo control policial y brinde atención al ciudadano. Creemos importante realizar esta aclaración debido a la existencia de establecimientos como la comisaría de la mujer, por ejemplo, a la cual se puede recurrir ante hechos de violencia de género, entre otras.

2.5 Dispositivo

Este término hace referencia a los dispositivos electrónicos en los cuales se puede instalar la aplicación móvil, es decir, teléfonos celulares o tabletas. Consideramos oportuno definirlo para que no se preste a confusión con otros dispositivos electrónicos en los cuales puede correr el sistema web.

3. Herramientas y tecnologías a utilizar en el proyecto

En este capítulo se hará mención a conceptos técnicos los cuales creemos convenientes para una mejor comprensión del presente informe por parte del lector. Algunos de ellos serán muy básicos con el fin de brindar completitud al conjunto de los mismos.

3.1 Servidor

Un servidor es un equipo informático que se encarga de atender peticiones de otros equipos informáticos denominados clientes. Los clientes solicitan información al servidor (archivos, sistemas informáticos, bases de datos, etc.) y él se encarga de retornar lo solicitado. El flujo de información no solamente es bidireccional desde los dispositivos clientes y los servidores, sino que, también, la comunicación puede realizarse de un servidor a otro si es requerido.

Utilizamos un servidor para alojar los sistemas necesarios para el proyecto, junto con su base de datos.

3.2 Aplicación Android Nativa

Las aplicaciones nativas, en principio, se desarrollan utilizando el lenguaje específico para cada plataforma, por ejemplo, Java + XML en Android, Objective-C y Swift en iOS o C# y Visual Basic en Windows. Esto nos permite acceder a las API's (Interfaz de Programación de Aplicaciones) disponibles y nos debería ofrecer un mejor rendimiento. [1]

Para el presente proyecto, se propuso el desarrollo de una aplicación Android Nativa para uso de los ciudadanos.

3.3 Servicio Web

Un servicio web (en inglés, web service o web services) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y

ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. [2].

Fue necesaria la implementación de servicios web en nuestro proyecto para que la aplicación Android pueda interactuar con la base de datos alojada en el servidor.

3.4 Control de versiones de código fuente

Los programas para control de versiones son un grupo de aplicaciones originalmente ideadas para gestionar ágilmente los cambios en el código fuente de los programas y poder revertirlos, cuyo ámbito ha sido ampliado pasando del concepto control de versiones al de gestión de configuración de software, en el que se engloban todas las actividades que pueden realizarse por un equipo sobre un gran proyecto software u otra actividad que genere ficheros digitales (por ejemplo: documentos, ofertas, dibujos, esquemas, etcétera). [3]

3.5 Entorno de desarrollo integrado (IDE)

Un entorno de desarrollo integrado es un programa que cuenta con las herramientas necesarias para el desarrollo de software.

Dicho entorno puede ser personalizado o mejorado con la instalación de complementos según el criterio del programador. Los mismos pueden ayudar en diferentes tareas, por ejemplo, depuración de código fuente, su compilación o su ejecución.

3.6 Librerías de código fuente

Una librería de código fuente es un conjunto de funciones ya definidas que permiten realizar acciones estandarizadas particulares en un proyecto de software.

En el proyecto se utilizaron librerías para realizar gráficos, mapas, reportes, etcétera.

4. Cambios en la planificación del proyecto

El presente proyecto final de carrera fue realizado en base a una planificación que fue modificada respecto a la presentada a la cátedra con anterioridad. El objetivo de este capítulo será informar al lector los cambios realizados en el mismo y su fundamentación.

4.1 Cambios realizados en actividades y tareas planificadas

Se optó por agrupar las tareas necesarias en cuatro actividades principales, las cuales contemplan desde entrevistas con los interesados a la corrección de errores del producto de software final.

Con el fin de mejorar el desarrollo del proyecto y las estimaciones de esfuerzo del mismo, se realizaron modificaciones a las actividades y tareas planificadas en el año 2017.

A continuación, se presenta una tabla comparando las estimaciones en horas de las actividades, tareas y estimaciones de esfuerzo planificadas en año 2017 y las que fueron finalmente realizadas en el año 2019:

Actividad	Tarea	Estimación de esfuerzo 2017 [hs]	Estimación de esfuerzo 2019 [hs]	Duración final del esfuerzo [hs]
Investigar y analizar	Entrevistas	25	15	10
	Redacción de documento de requerimientos	14	5	2,5
	Lectura de leyes y resoluciones	12	8	10
	Definir metodología de trabajo	5	4	4
Diseñar base de datos	Definir datos necesarios	5	4	7
	Realizar un diagrama Entidad-Relación	3	3	4
Definir arquitectura del proyecto y tecnologías a utilizar	Definir arquitectura de la solución	4	10	5
	Seleccionar tecnologías a utilizar	4	2	5
	Definir motor de base de datos	3	2	0,5
	Definir lenguaje/s de desarrollo	3	3	0,7
	Definir librería/s gráfica/s	4	2	0,5
	Definir librería/s de mapas	5	2	0,5

Implementar solución propuesta	Planificar proceso de desarrollo	-	4	20
	Instalar o agregar la tecnología necesaria	8	2	3,5
	Definir módulos de la solución	8	2	20
	Crear base de datos	8	2	0,5
	Crear tablas de base de datos	8	4	1
	Generar código fuente y la documentación	320	270	233
	Generar casos de prueba	12	20	30
	Verificar el correcto funcionamiento	12	20	15
	Validar con el stakeholder	8	8	0
	Corregir errores	20	-	-
Redacción de documentos	Escribir informe final de proyecto	-	140	80
TOTAL		491	532	452,7

Tabla 1. Estimación de esfuerzo entre planes 2017 y 2019

4.2 Cambios realizados en la implementación de la solución

Por la experiencia adquirida de quienes implementan el presente proyecto y la necesidad de una mejora en los procesos de denuncia para el ciudadano (en materia de tiempo requerido y anonimato principalmente), así como también un uso eficiente y estandarizado de los lenguajes de programación a utilizar, se proponen los siguientes cambios:

1. **Eliminar el proceso de rectificación de la denuncia por parte del ciudadano:** en la solución planificada originalmente el ciudadano, luego de cargar los datos denunciados en la aplicación, debía asistir a la comisaría para rectificar lo denunciado (en denuncias no anónimas). Este proceso es engorroso y confuso, por lo que probablemente traiga aparejado un bajo índice de denuncia, se propuso entonces que todas las denuncias sean

anónimas y revisadas por un funcionario policial con el fin de determinar la correcta confección de la misma.

2. **Rapidez en el registro de una denuncia:** se redujeron al mínimo los datos obligatorios requeridos por trámite, de esa forma, cargar una denuncia en la aplicación debería tomar pocos minutos, permitiendo al ciudadano un ahorro significativo en su tiempo y una mejor predisposición a usar la aplicación.
3. **Obtener información relevante respecto a delitos no denunciados:** en sintonía con los puntos anteriores, al facilitar al ciudadano un canal de denuncia simple y rápido, se busca captar delitos que anteriormente eran poco denunciados (hurto de celulares, bicicletas, artículos personales, etc.).
4. **Garantizar el anonimato del denunciante:** en el anterior enfoque de la solución, se proponía que el ciudadano, al realizar denuncias no anónimas, obtenga un código de seguimiento que le permita monitorear el avance de su trámite. Sin embargo, dicho código puede traer aparejados inconvenientes con el anonimato del denunciante. Es por ello que se optó por dejar de lado esta funcionalidad y centrarse en el dispositivo desde el cual se emite la denuncia en lugar de la persona que la realiza, es decir, todas las denuncias son anónimas y están asociadas a un dispositivo a través de un código interno. Este enfoque brinda al ciudadano la posibilidad de realizar denuncias sin necesidad de credenciales de autenticación y que un mismo dispositivo pueda ser utilizado por múltiples personas físicas.
5. **Optimización de tecnologías a utilizar:** la solución planificada inicialmente contemplaba el uso de tres frameworks de desarrollo; Ionic Framework para la aplicación móvil; Laravel Framework para la API que le permite a la aplicación comunicarse con la base de datos y .NET Framework, en el cual se desarrollaría el sistema web a utilizar por los funcionarios policiales. Con el nuevo enfoque propuesto, y basados en la experiencia adquirida, proponemos utilizar Android Nativo y Laravel solamente. De esta manera la solución alcanza un estado homogéneo que facilita el mantenimiento de la misma.

4.3 Gestión de riesgos del proyecto

Los proyectos de investigación y desarrollo involucran una serie de actividades, las cuales son llevadas a cabo por equipos de trabajo utilizando tecnologías conocidas o por conocer. La naturaleza de los mismos trae aparejadas consecuencias que atentan contra su avance e

implementación, ya sea en materia de tiempos, costos económicos, tecnológicos, etc. dichas consecuencias son inherentes a todo proyecto y se denominan **riesgos**.

En el desarrollo de este proyecto, nos encontramos con una serie de riesgos detallados en la planificación que se hicieron presentes. A continuación, comentaremos cómo lidiamos con ellos.

4.3.1 Riesgo en planificación del proyecto

Debido a la falta de experiencia en el desarrollo de proyectos de software, se puede fallar en la planificación de tareas. Es decir, se pueden terminar tareas más rápidamente o más lentamente de lo previsto.

En nuestro caso, hubo que ajustar y reestimar las actividades y tareas no solamente del desarrollo de software, sino además del proyecto en sí. Esto se debió a que la primera planificación fue realizada en el año 2017 y no contábamos con la experiencia suficiente en las herramientas que utilizamos hoy en día. Como consecuencia, el tiempo estimado fue ajustado y permitió llevar adelante el proyecto en mucho menos tiempo del estimado.

4.3.2 Riesgo en definición de requerimientos

Cuando definimos los requerimientos con el cliente existe la posibilidad de malinterpretar sus requisitos, esto sucede porque nosotros y el cliente podemos tener distintas formas de interpretar el mismo requerimiento.

En nuestro caso, uno de los requerimientos iniciales era que el usuario pudiera realizar denuncias de forma no anónima, con lo que debíamos almacenar datos personales con el fin de no tener que rellenarlos ante cada denuncia. Luego, debido a no poder garantizar la seguridad del proceso, se decidió que todas las denuncias sean anónimas e identificar al dispositivo y no a la persona denunciante.

Esto trajo consigo cambios en la manera de plantear un eje clave del proyecto, por lo que nos retrasó en el proceso de planificación. Por fortuna, dicho cambio se originó antes de comenzar con la fase de implementación, por lo que su impacto no fue demasiado elevado en materia de tiempos.

4.3.3 Riesgos imprevistos

La pandemia mundial, y su cuarentena en el país a partir del mes de marzo de 2020 fue un riesgo totalmente imprevisto para el desarrollo del proyecto. La misma, trajo aparejada muchos

cambios a nivel mundial y de índole personal para los que llevamos adelante el proyecto, con consecuencias muy graves tanto en lo económico como en lo emocional.

Debimos dejar de lado el proyecto final de carrera por varios meses hasta poder estabilizarnos desde lo emocional y lo económico, pero pudimos sobreponernos a la crisis y darle finalmente un cierre a esta etapa de nuestras vidas.

5. Investigación y análisis

5.1 Entrevistas

En todo proyecto relacionado a sistemas de información, el recabar información sobre los objetivos a contemplar y los procesos necesarios para alcanzar el mismo, es de vital importancia. Para ello, suelen realizarse entrevistas o charlas con los interesados y, si corresponde, con los usuarios finales del proyecto a desarrollar.

En nuestro caso, lo primero que hicimos fue presentar una propuesta inicial al Director de Ayudantía General de la Policía de Entre Ríos a través de una charla informal (debido a su apretada agenda). Como resultado de dicha presentación, obtuvimos el visto bueno para la realización de un prototipo del proyecto.

Luego de dicha reunión, nos vimos en la necesidad de encontrar un stakeholder para poder interiorizarnos sobre los protocolos de denuncia y los procesos derivados de la misma.

Debido al escaso tiempo con el que cuentan los jefes y autoridades policiales, decidimos contactar al Ex Director de Asuntos Internos de la Policía de Entre Ríos, el cual sirvió por más de treinta años en la misma y posee una experiencia más que idónea para nuestra investigación.

La entrevista que le hicimos puede consultarse en el Anexo 11.1, y de la misma obtuvimos la idea general del proceso de denuncia, los tipos de hecho que, por lo general, la gente no suele denunciar en las comisarías y la importancia de la anonimidad de los datos del denunciante en los tiempos que corren. Además, nos sugirió sobre las posibilidades del proyecto a futuro, enfatizando en el control de la efectividad de las operaciones realizadas por la Fuerza y el entrecruzamiento de datos entre sistemas ya existentes.

Por último, nos planteamos la idea de hacer una entrevista a un usuario final de la aplicación móvil, es decir, un ciudadano.

Para la elección de los entrevistados nos basamos en la población objetivo del proyecto, es por ello que decidimos entrevistar a una persona entre 20 - 30 años, otra entre 30 - 40 años y

por último 50 - 60 años. De esta manera, pudimos obtener información sobre la persona respecto a su experiencia en el uso de herramientas de mapeo, su nivel de confianza en la Fuerza Policial ante un hecho que lo perjudique y su predisposición respecto a la utilización de una aplicación de denuncia ciudadana. Las entrevistas realizadas pueden consultarse en el Anexo 11.1.

5.2 Lectura de leyes y resoluciones

Basándonos en el Código Penal de la Nación Argentina Ley 11.179 [4], en la tipificación existente de los delitos y en el sistema de denuncias con el que actualmente se maneja la Policía de Entre Ríos. Se obtuvieron las siguientes categorías de delitos:

- Contra las Personas: son aquellos delitos que se cometen contra la integridad física de las personas, causando la muerte o lesiones, en sus distintos tipos agravados, como el delito de homicidio agravado o asesinato o lesiones.
 - Homicidio doloso consumado
 - Homicidio doloso en grado de tentativa
 - Muerte en accidentes viales
 - Homicidio culposo por otros hechos
 - lesiones dolosas
 - lesiones culposas en accidentes viales
 - lesiones culposas en otros hechos
 - Otros delitos contra las personas
- Contra la Integridad sexual y el honor: son aquellos que atacan la libertad y la voluntad sexual de una persona
 - Contra el honor
 - Abuso sexual con acceso carnal
 - Otros delitos contra la integridad sexual
- Contra el Estado Civil: Este bien jurídico responde a un concepto establecido por el Derecho Civil. Es la condición de cada persona en relación con los derechos y obligaciones civiles.
 - Matrimonios ilegales
 - Supresión y suposición del estado civil y la identidad
- Contra la libertad: aquella que atente contra la libertad individual, que consiste en la facultad constante que tiene el hombre para ejercer sus actividades, tanto morales como físicas, en servicio de sus propias necesidades y con el fin de alcanzar su destino en la vida terrestre.
 - Amenazas
 - Trata de personas simple
 - Trata de personas agravado
 - Otros delitos contra la libertad
- Contra la propiedad: son aquellos delitos que atenten contra la propiedad. Entendiendo a la propiedad como “todo lo que es susceptible de valor económico o apreciable en

dinero, sea objeto corporal, sea incorporal, como un crédito o un invento, ya se adquiriera en manera originaria, ya derivadas, es decir por contrato, ley, sentencia", "o una decisión administrativa".

- Robos
- Tentativa de robo
- Robo agravado por el resultado de lesiones y/o muerte
- Tentativa de robo agravado por el resultado de lesiones y/o muerte
- Hurtos
- Tentativa de hurto
- Otros delitos contra la propiedad
- Contra la Seguridad Pública: desde el punto de vista de la conducta incriminada, en la creación de un peligro para la conservación de los bienes materialmente considerados y las personas (vida y salud más apropiadamente) de un modo indeterminado. Es decir, que los bienes de que se trata, los de cualquiera, los de todos y de cada uno, son expuestos a resultar dañados o destruidos, y suprimida o lesionada la integridad biológica de las personas, como consecuencia de aquélla. Es característica, además, la posibilidad de propagación del daño a un número indeterminado de víctimas.
 - Explosivos y otros
 - Tenencia ilegal de armas de fuego
 - Portación ilegal de armas de fuego
 - Acopio y fabricación ilegal de armas, piezas y municiones
 - Entrega y Comercialización ilegal de armas de fuego
 - Omisión, adulteración y suspensión de mareaje
 - Otros delitos contra la seguridad pública
- Contra el orden público: Las infracciones que constituyen los delitos contra el orden público (en los que incluye la rebelión, sedición, atentado, resistencia y desobediencia, desacatos, desórdenes públicos y tenencia y depósito de armas y municiones y tenencia de explosivos) se caracterizan por el mantenimiento de la tranquilidad o la paz en las manifestaciones colectivas de la vida ciudadana
 - Instigación a cometer delitos
 - Asociación ilícita
 - Intimidación pública
 - Apología del delito
 - Otros atentados contra el orden público
- Contra la seguridad de la Nación: Las conductas previstas como delito atentan contra la seguridad de la nación, es decir, que crean un peligro para la conservación de la existencia del estado como ente soberano e independiente de los otros poderes constituidos en el contexto internacional.
 - Traición
 - Delitos que comprometen la paz y la dignidad de la Nación

- Contra los poderes públicos y el orden constitucional: Nos limitaremos a decir que desde el punto de vista de la objetividad jurídica esta categoría es para la protección de la comunidad jurídica y políticamente organizada en su orden interno.
 - Afectados al orden constitucional y la vida democrática
 - Sedición
- Contra la administración pública: son aquellos delitos que se cometen contra el normal, ordenado, legal y decoroso desenvolvimiento de la función administrativa.
 - Afectado y resistencia contra la autoridad
 - Falsa denuncia
 - Usurpación de autoridad, títulos u honores
 - Abuso de autoridad y violación de los deberes de funcionarios públicos
 - Violación de sellos y documentos
 - Cohecho y tráfico de influencias
 - Malversación de caudales públicos
 - Negociaciones incompatibles con el ejercicio de funciones públicas
 - Exacciones ilegales
 - Enriquecimiento ilícito de funcionarios y empleados
 - Prevaricato
 - Denegación y retardo de justicia
 - Falso testimonio
 - Encubrimiento
 - Evasión y quebrantamiento de pena
- Contra la fe pública: delitos que atentan sobre aquello que da autenticidad y veracidad: objetos y declaraciones que suscitan el crédito de los demás fundándose en signos exteriores o en la práctica de la vida de relación en comunidad.
 - Falsificación de moneda
 - Falsificación de sellos, timbres y marcas
 - Falsificación de documentos en general
 - Materias o instrumentos destinados a cometer falsificaciones legisladas
 - Fraudes al comercio y a la industria
 - Pago de cheques sin provisión de fondos
- Contra el orden económico y financiero: es la tipificación de una diversidad de envueltos en investigaciones criminales por "asociación delictiva", "encubrimiento", y hasta lavado de dinero y activos, legitimación y blanqueo de capitales.
- Infracciones a la Ley 23737 (estupefacientes): según la ley se detallan los siguientes:
 - Siembra y producción de estupefacientes
 - Comercialización y entrega
 - Tenencia o entrega atenuada
 - Desvío de importación de Drogas
 - Organización y Financiación
 - Tenencia simple
 - Tenencia simple atenuada para uso personal

- Confabulación
- Contrabando de estupefacientes
- Otras infracciones a la Ley 23737
- Contrabando:
 - Obstrucción de control aduanero
 - Contrabando Simple
 - Contrabando agravado
 - Contrabando de elementos nucleares, agresivos químicos, armas y municiones
- Delitos Ambientales
- Delitos migratorios
- Otros Delitos previstos en leyes especiales: leyes especiales que están aparte de las demás categorías.
 - Ley de protección integral a las mujeres
 - Ley de protección contra la violencia familiar
 - Ley Penal Tributaria
 - Ley Contra los tratos y actos de crueldad a los animales
- Figuras contravencionales
- Suicidios

5.3 Selección de metodología de trabajo

Para el desarrollo del proyecto decidimos utilizar una metodología ad-hoc, es decir, una metodología adaptada o personalizada. Esta decisión se fundamentó debido a que el sistema a desarrollar no presenta una complejidad que justifique el uso de una metodología tradicional con los procesos, informes y diagramas que las mismas implican. A su vez, ninguna metodología ágil de las conocidas por los integrantes del proyecto puede aplicarse fielmente al ciclo de desarrollo debido a los recursos con los que contamos.

Si bien el desarrollo del proyecto fue realizado de manera personalizada, fueron utilizados productos de la metodología ágil SCRUM solamente para la gestión del proyecto [5] (Product Backlog, Sprint Backlog, etc.) debido a nuestra experiencia con la misma y su probada robustez. Sin embargo, al estar compuesto el grupo por dos integrantes, no podíamos cumplir con los roles necesarios para adaptar la metodología por completo.

La metodología adoptada contempló la realización de diagramas y documentos que se consideraron esenciales para el modelado de los datos, reuniones de progreso diario o semanal, gestión de actividades en base a tareas y la redacción de la documentación inherente al proyecto.

Para un control de tiempos, tareas y resolución de errores se decidió adoptar una metodología de control de versionado con Git utilizando su implementación Gitlab (detallada en la sección 7.7, “Herramientas adicionales”). La misma nos permite crear diferentes versiones de la aplicación con el fin de garantizar prolijidad, escalabilidad y contingencia ante errores.

5.4 Requerimientos de la solución

Basado en la información recabada a través de las entrevistas del sub capítulo anterior y lo hablado previamente con los interesados, se procedió a la definición de los requerimientos del proyecto.

Para ello, de la metodología SCRUM, utilizamos el Product Backlog, el cual se encarga de definir actores involucrados en la solución, actividades necesarias para llevar adelante el proyecto y su esfuerzo estimado, prioridad e iteración en la cual será implementada.

En cuanto a los actores involucrados, fueron identificados los siguientes:

- **Ciudadano:** persona que tenga intención de reportar un delito dentro de los límites de la provincia de Entre Ríos.
- **Auditor:** funcionario policial. Encargado de revisar las denuncias realizadas por ciudadanos, re-tipificarlas según la definición de delito legislada y sancionar al denunciante por uso indebido de la plataforma si lo amerita.
- **Analista:** funcionario policial. Encargado de filtrar y procesar los datos almacenados con el fin de generar información de interés para la toma de decisiones.
- **Administrador:** funcionario policial. Realiza la gestión de recursos del producto y se encarga de que todo funcione correctamente.
- **Desarrollador:** encargado de llevar adelante el proyecto y de su mantenimiento.
- **Usuario:** usuarios del sistema, engloba a los actores ciudadano, auditor, analista y administrador.

Para el esfuerzo estimado por actividad se utilizó una escala del 1 al 5, quedando en manos de los desarrolladores la elección del número de dicho atributo.

La prioridad fue determinada por los desarrolladores de acuerdo a una escala alta, media o baja.

En SCRUM, el desarrollo de un producto se realiza de una manera iterativa [6], es decir, se dividen las actividades en iteraciones con el fin de obtener un resultado de valor para el cliente al concluir cada iteración y, a su vez, poder organizar el trabajo de los desarrolladores.

Para este proyecto y, basados en la cantidad y complejidad de las actividades aparejadas al mismo, se propuso un total de dos iteraciones. Además, existe una iteración cero, la cual permite a los desarrolladores la puesta a punto necesaria para comenzar sus tareas.

Basados en lo expuesto anteriormente, se presenta el Product Backlog obtenido:

PRODUCT BACKLOG				
ID	Enunciado	Esfuerzo	Iteración	Prioridad
1	Como Ciudadano , necesito realizar denuncias de forma anónima, con la finalidad de conservar mi derecho a preservar mi identidad	5	1	alta
2	Como Ciudadano , necesito conocer donde debo radicar una denuncia ante un hecho, con la finalidad de poder realizarla rápida y correctamente	5	2	baja
3	Como Ciudadano , debo poder acceder a información sobre dependencias policiales y gubernamentales (N° de teléfono, ubicación, horarios de atención, etc.)	4	2	baja
4	Como Auditor , necesito obtener un listado de denuncias por auditar. El mismo debe poder filtrarse por identificador de dispositivo y rango de fechas.	2	1	media
5	Como Auditor , necesito poder cambiar el estado, re-tipificar o desestimar una denuncia realizada por un ciudadano según el contenido de la misma	3	1	alta
6	Como Auditor , debo poder darle un strike a un dispositivo que haya incluido contenido inapropiado en una denuncia. Cuando un dispositivo cuente con tres strikes, toda denuncia futura será desestimada automáticamente	1	2	baja
7	Como Analista , necesito poder visualizar todas las denuncias, ver cantidades y porcentajes y visualizar gráficas según diferentes filtros para tener una síntesis de	4	2	media

	estadísticas delictuales			
8	Como Analista , necesito poder visualizar un mapa de delitos según filtros de rango de fechas y tipo, para analizar las zonas problemáticas.	4	2	baja
9	Como Administrador , debo poder gestionar usuarios del sistema	3	1	media
10	Como Desarrollador , debo garantizar que cada usuario del sistema acceda solamente a las secciones a las cuales esté autorizado.	3	1	media
11	Como Desarrollador , debo brindar un modo autenticado y seguro de acceso al sistema para los funcionarios policiales	2	1	baja
12	Como Desarrollador , debo asegurar que la aplicación móvil esté disponible para la mayor cantidad de ciudadanos posibles	1	1	media
13	Como Desarrollador , debo garantizar el acceso al sistema a través de los navegadores web más utilizados	1	1	media
14	Como Desarrollador , debo garantizar que no se borren de manera física los datos almacenados	2	1	alta
15	Como Usuarios , necesitamos que los productos de software tengan un aspecto simple y sean fáciles de utilizar	5	1	alta

Tabla 2. Product Backlog

5.4.1 Historias de usuario

Las historias de usuario se definen utilizando los siguientes datos:

- ID: identificador único de cada historia, el formato del mismo es “HU99”, dónde 99 simboliza dos números enteros.
- Nombre: nombre corto para identificar coloquialmente a la historia.
- Prioridad: nivel de importancia de la historia para la concreción del proyecto. Se define en conjunto con el usuario. Toma uno de los siguientes valores: alta, media y baja.
- Esfuerzo: estimación por parte de los desarrolladores sobre la realización de la historia.
- Descripción: explicación breve y ordenada de la historia.

- Validación: condiciones que deben cumplirse para que la historia se considere finalizada.

HU01	
ID	HU01
Nombre	Realizar denuncia
Prioridad	Alta
Esfuerzo	5
Descripción	Como Ciudadano , necesito realizar denuncias de forma anónima, con la finalidad de conservar mi derecho a preservar mi identidad
Validación	<p>1 - Datos que contendrá la denuncia:</p> <ul style="list-style-type: none"> - Fecha hecho - Fecha denuncia - Tipo de hecho denunciado - Coordenadas del hecho - Descripción del hecho - Archivo de audio (opcional) <p>2 - Implementar un mecanismo para identificar un dispositivo que realiza una denuncia</p>

Tabla 3. Historia de usuario 01

HU02	
ID	HU02
Nombre	Mapa dependencias policiales
Prioridad	Baja
Esfuerzo	5
Descripción	Como Ciudadano , necesito conocer donde debo radicar una denuncia ante un hecho, con la finalidad de poder realizarla rápida y correctamente
Validación	<p>1 - Implementar un mapa con la ubicación y el área de acción de cada dependencia policial de la provincia. Cuando el usuario cliquee un punto de dependencia, debería desplegarse una ventana emergente con la siguiente información:</p> <ul style="list-style-type: none"> - Nombre - Dirección - Localidad - Teléfono <p>2 - Cuando se mantenga presionado el punto de la dependencia, debería permitir llamar por teléfono automáticamente a la misma.</p>

Tabla 4. Historia de usuario 02

HU03	
ID	HU03
Nombre	Información dependencias policiales
Prioridad	Baja
Esfuerzo	4
Descripción	Como Ciudadano , debo poder acceder a información sobre dependencias policiales y gubernamentales (N° de teléfono, ubicación, horarios de atención, etc.)
Validación	<p>1 - Implementar un listado de dependencias policiales de la provincia que brinde la siguiente información:</p> <ul style="list-style-type: none"> - Nombre - Dirección - Localidad - Departamento - Teléfono <p>2 - Desde el listado se debe permitir llamar por teléfono automáticamente a cada dependencia.</p>

Tabla 5. Historia de usuario 03

HU04	
ID	HU04
Nombre	Listado de denuncias por auditar
Prioridad	Media
Esfuerzo	2
Descripción	Como Auditor , necesito obtener un listado de denuncias por auditar. El mismo debe poder filtrarse por identificador de dispositivo y rango de fechas.
Validación	<p>1 - Implementar un listado de denuncias que no hayan sido rectificadas o desestimadas. Ordenarlas por fecha de realización más reciente y mostrar por cada una los siguientes datos:</p> <ul style="list-style-type: none"> - Botón para visualizar denuncia - Fecha de realización de denuncia - Fecha del hecho - Tipo de hecho denunciado - Descripción

Tabla 6. Historia de usuario 04

HU05	
ID	HU05
Nombre	Acciones a realizar con las denuncias realizadas por un ciudadano
Prioridad	Alta

Esfuerzo	3
Descripción	Como Auditor , necesito poder cambiar el estado, re-tipificar o desestimar una denuncia realizada por un ciudadano según el contenido de la misma
Validación	1 - Ante una denuncia realizada por un ciudadano, un auditor debe revisar para verificar su correcta confección. Para ello, debe contar con las siguientes opciones: <ul style="list-style-type: none"> - Elegir una o varias categorías de delito - Elegir una o varias subcategorías de delito - Observaciones (opcional) - Opción de desestimar denuncia - Opción de aceptar denuncia - Opción de dar un strike al dispositivo denunciante

Tabla 7. Historia de usuario 05

HU06	
ID	HU06
Nombre	Desestimar denuncias de dispositivos baneados
Prioridad	Baja
Esfuerzo	1
Descripción	Como Auditor , debo poder darle un strike a un dispositivo que haya incluido contenido inapropiado en una denuncia. Cuando un dispositivo cuente con tres strikes, toda denuncia futura será desestimada automáticamente
Validación	1 - El sistema internamente debe poder desestimar automáticamente las denuncias que provengan de dispositivos que cuenten con tres strikes, es decir, baneados.

Tabla 8. Historia de usuario 06

HU07	
ID	HU07
Nombre	Generar reportes y estadísticas basadas en denuncias
Prioridad	Media
Esfuerzo	4
Descripción	Como Analista , necesito poder visualizar todas las denuncias, ver cantidades y porcentajes y visualizar gráficas según diferentes filtros para tener una síntesis de estadísticas delictuales

Validación	<p>1 - El sistema debe poder generar reportes en formato Excel o pdf mediante una tabla con los siguientes campos:</p> <ul style="list-style-type: none"> - Fecha de realización de denuncia - Fecha del hecho - Tipo de hecho denunciado - Descripción - Estado - Categoría hecho - Subcategoría hecho - Observaciones - Usuario revisor <p>2 - El sistema debe generar un tablero de control con gráficos que reflejen una síntesis de interés para las autoridades policiales. Los mismos serán:</p> <ul style="list-style-type: none"> - Cantidad total de hechos denunciados - Cantidad total de hechos aceptados - Cantidad total de hechos desestimados - Cantidad total de denuncias revisadas - Cantidad total de usuarios baneados - Cantidad de hechos denunciados por mes en el año actual - Cantidad de hechos denunciados por año - Cantidad de hechos denunciados por tipo - Cantidad de hechos denunciados por categoría - Cantidad de hechos denunciados por subcategoría <p>3 - Los filtros para los reportes será:</p> <ul style="list-style-type: none"> - Intervalo de fechas - Categoría delito - Subcategoría delito <p>3 - Los filtros para las estadísticas serán:</p> <ul style="list-style-type: none"> - Intervalo de fechas
------------	---

Tabla 9. Historia de usuario 07

HU08	
ID	HU08
Nombre	Generar un mapa con delitos denunciados
Prioridad	Baja
Esfuerzo	4
Descripción	Como Analista , necesito poder visualizar un mapa de delitos según filtros de rango de fechas y tipo, para analizar las zonas problemáticas.

Validación	<p>1 - El sistema debe permitir generar un mapa de puntos con la geolocalización de denuncias aceptadas. Cuando un usuario cliquee sobre un punto, se mostrará la siguiente información:</p> <ul style="list-style-type: none"> - Fecha hecho - Fecha denuncia - Categoría hecho denunciado - Subcategoría hecho denunciado - Botón "ver detalles". Redirige hacia todos los datos de la denuncia. <p>2 - Los filtros para la generación del mapa serán:</p> <ul style="list-style-type: none"> - Intervalo de fechas - Categoría delito - Subcategoría delito
------------	--

Tabla 10. Historia de usuario 08

HU09	
ID	HU09
Nombre	Administrar usuarios del sistema web
Prioridad	Media
Esfuerzo	3
Descripción	Como Administrador , debo poder gestionar usuarios del sistema
Validación	<p>1 - El sistema debe permitir crear, mostrar, editar y eliminar usuarios. Cada usuario deberá contener la siguiente información:</p> <ul style="list-style-type: none"> - Nombre y apellido - Dependencia - Nombre de usuario - Password - Fecha de alta - Correo electrónico - Autorizado [Sí/No] - Fecha de validación de cuenta - Borrado [Sí/No] <p>2 - El borrado de usuarios debe ser de forma lógica, no física.</p> <p>3 - Al momento de crear un usuario, el administrador debe colocar la contraseña de la cuenta y notificar al mismo a través del medio que corresponda.</p>

Tabla 11. Historia de usuario 09

HU10	
ID	HU10

Nombre	Gestión de roles y permisos de sistema web
Prioridad	Media
Esfuerzo	3
Descripción	Como Desarrollador , debo garantizar que cada usuario del sistema acceda solamente a las secciones a las cuales esté autorizado.
Validación	1 - El sistema debe permitir crear, mostrar, editar, eliminar y asociar roles y permisos a usuarios. Cada usuario puede tener múltiples roles o permisos y cada rol debe estar conformado por uno o más permisos de manera obligatoria.

Tabla 12. Historia de usuario 10

HU11	
ID	HU11
Nombre	Autenticar ingreso al sistema web de forma segura
Prioridad	Baja
Esfuerzo	2
Descripción	Como Desarrollador , debo brindar un modo autenticado y seguro de acceso al sistema para los funcionarios policiales
Validación	1 - El sistema debe garantizar que no pueda accederse a ninguna sección del mismo sin que previamente el usuario haya proporcionado su nombre de usuario y contraseña. 2 - Debe haber un mecanismo de restablecimiento de contraseña vía email para el usuario ante un olvido de su contraseña e imposibilidad de ingreso al sistema. 3 - El usuario debe poder cambiar su contraseña en cualquier momento una vez ingresa al sistema exitosamente.

Tabla 13. Historia de usuario 11

HU12	
ID	HU12
Nombre	Garantizar disponibilidad de aplicación móvil para ciudadanos
Prioridad	Media
Esfuerzo	1
Descripción	Como Desarrollador , debo asegurar que la aplicación móvil esté disponible para la mayor cantidad de ciudadanos posibles
Validación	1 - La aplicación móvil deberá poder descargarse desde Google PlayStore.

Tabla 14. Historia de usuario 12

HU13	
ID	HU13

Nombre	Garantizar acceso al sistema web a través de navegadores modernos
Prioridad	Media
Esfuerzo	1
Descripción	Como Desarrollador , debo garantizar el acceso al sistema a través de los navegadores web más utilizados
Validación	1 - El sistema web deberá poder ser accedido desde Google Chrome y Mozilla Firefox

Tabla 15. Historia de usuario 13

HU14	
ID	HU14
Nombre	Borrado de datos de manera lógica
Prioridad	Alta
Esfuerzo	2
Descripción	Como Desarrollador , debo garantizar que no se borren de manera física los datos almacenados
Validación	1 - El borrado de datos deberá ser de manera lógica, no física.

Tabla 16. Historia de usuario 14

HU15	
ID	HU15
Nombre	Interfaz y experiencia de usuario amigable en los productos a desarrollar
Prioridad	Media
Esfuerzo	5
Descripción	Como Usuarios , necesitamos que los productos de software tengan un aspecto simple y sean fáciles de utilizar
Validación	1 - Diseñar las interfaces de los productos de manera simple y amigable para el usuario

Tabla 17. Historia de usuario 15

5.4.2 Tareas

Las tareas que componen la concreción de cada historia de usuario, tienen los siguientes atributos:

- ID: identificador único de cada tarea, el formato del mismo es “T99”, donde 99 simboliza dos números enteros.
- Historia asociada: identificador de la historia de usuario de la que forma parte la tarea.

- Sprint: número de sprint en el que se incluyó la tarea. Al comienzo dicho valor se encontraba vacío hasta la reunión de planificación del sprint, pero para evitar duplicar la información el mismo será incluido en esta sección.
- Nombre: nombre corto para identificar coloquialmente a la tarea.
- Descripción: explicación breve y ordenada de la tarea.

Tarea	
ID	T01
Historia asociada	-
Sprint	0
Nombre	Crear el proyecto para el sistema web
Descripción	Crear el proyecto en Laravel para el sistema que ha ser utilizado por funcionarios policiales

Tabla 18. Tarea 1.

Tarea	
ID	T02
Historia asociada	-
Sprint	0
Nombre	Crear el repositorio de Gitlab para el sistema web
Descripción	Crear el repositorio en Gitlab para el control de versiones del sistema

Tabla 19. Tarea 2.

Tarea	
ID	T03
Historia asociada	-
Sprint	0
Nombre	Crear el proyecto para la API
Descripción	Crear el proyecto en Laravel para la implementación de la API

Tabla 20. Tarea 3.

Tarea	
ID	T04
Historia asociada	-
Sprint	0
Nombre	Crear el repositorio de Gitlab para la API
Descripción	Crear el repositorio en Gitlab para el control de versiones del proyecto que implementa la API

Tabla 21. Tarea 4.

Tarea	
ID	T05
Historia asociada	-
Sprint	0
Nombre	Crear el proyecto para la aplicación móvil
Descripción	Crear el proyecto en Android Studio

Tabla 22. Tarea 5.

Tarea	
ID	T06
Historia asociada	-
Sprint	0
Nombre	Crear el repositorio de Gitlab para la aplicación móvil
Descripción	Crear el repositorio en Gitlab para el control de versiones del proyecto que implementa la aplicación móvil

Tabla 23. Tarea 6.

Tarea	
ID	T07
Historia asociada	-
Sprint	0
Nombre	Crear la base de datos
Descripción	Crear la base de datos en PostgreSQL

Tabla 24. Tarea 7.

Tarea	
ID	T08
Historia asociada	HU15
Sprint	1
Nombre	Realizar bocetos de interfaces
Descripción	Diagramar las interfaces del sistema web y la aplicación móvil.

Tabla 25. Tarea 8.

Tarea	
ID	T09
Historia asociada	HU01
Sprint	1
Nombre	Formulario alta denuncia

Descripción	Datos que contendrá la denuncia: - Fecha hecho - Fecha denuncia - Tipo de hecho denunciado - Coordenadas del hecho - Descripción del hecho - Archivo de audio (opcional) - Identificador del dispositivo que realiza la denuncia
-------------	---

Tabla 26. Tarea 9.

Tarea	
ID	T10
Historia asociada	HU01
Sprint	1
Nombre	Almacenar datos denuncia
Descripción	Persistir los datos de la denuncia en la base de datos a través de la API.

Tabla 27. Tarea 10.

Tarea	
ID	T11
Historia asociada	HU01
Sprint	1
Nombre	Mensaje de voz en denuncia
Descripción	Permitir que se pueda grabar un mensaje de voz y adjuntarlo a la denuncia

Tabla 28. Tarea 11.

Tarea	
ID	T12
Historia asociada	HU01
Sprint	1
Nombre	Almacenar mensaje de voz de denuncia
Descripción	Almacenar el mensaje de audio de las denuncias en el storage

Tabla 29. Tarea 12.

Tarea	
ID	T13
Historia asociada	HU01
Sprint	1
Nombre	Permitir identificar un dispositivo denunciante
Descripción	Implementar un mecanismo para identificar un dispositivo que realiza una denuncia

Tabla 30. Tarea 13.

Tarea	
ID	T14
Historia asociada	HU02
Sprint	2
Nombre	Mapa de dependencias policiales
Descripción	Implementar un mapa con el área de acción de cada dependencia policial de la provincia. Cuando el usuario cliquee un punto, debería desplegarse una ventana emergente con la siguiente información: - Nombre - Dirección - Localidad - Teléfono

Tabla 31. Tarea 14.

Tarea	
ID	T15
Historia asociada	HU02
Sprint	2
Nombre	Llamar a dependencia policial desde el mapa
Descripción	Cuando se mantenga presionado el punto de la dependencia, debería permitir llamar por teléfono automáticamente a la misma.

Tabla 32. Tarea 15.

Tarea	
ID	T16
Historia asociada	HU03
Sprint	2
Nombre	Listado de dependencias policiales
Descripción	Implementar un listado de dependencias policiales de la provincia que brinde la siguiente información: - Nombre - Dirección - Localidad - Departamento - Teléfono

Tabla 33. Tarea 16.

Tarea	
ID	T17
Historia asociada	HU03

Sprint	2
Nombre	Llamar a dependencia policial desde el listado
Descripción	Desde el listado se debe permitir llamar por teléfono automáticamente a cada dependencia.

Tabla 34. Tarea 17.

Tarea	
ID	T18
Historia asociada	HU04
Sprint	1
Nombre	Listado de denuncias por auditar
Descripción	Implementar un listado de denuncias que no hayan sido rectificadas o desestimadas. Ordenarlas por fecha de realización más reciente y mostrar por cada una los siguientes datos: - Botón para visualizar denuncia - Fecha de realización de denuncia - Fecha del hecho - Tipo de hecho denunciado - Descripción

Tabla 35. Tarea 18.

Tarea	
ID	T19
Historia asociada	HU05
Sprint	1
Nombre	Revisión de denuncia
Descripción	Ante una denuncia realizada por un ciudadano, un auditor debe revisar para verificar su correcta confección. Para ello, debe contar con las siguientes opciones: - Elegir una o varias categorías de delito - Elegir una o varias subcategorías de delito - Observaciones (opcional) - Opción de desestimar denuncia - Opción de aceptar denuncia - Opción de dar un strike al dispositivo denunciante

Tabla 36. Tarea 19.

Tarea	
ID	T20
Historia asociada	HU06

Sprint	2
Nombre	Desestimar automáticamente denuncias de dispositivos baneados
Descripción	El sistema internamente debe poder desestimar automáticamente las denuncias que provengan de dispositivos que cuenten con tres strikes, es decir, baneados.

Tabla 37. Tarea 20.

Tarea	
ID	T21
Historia asociada	HU07
Sprint	2
Nombre	Generar reportes de hechos denunciados
Descripción	El sistema debe poder generar reportes en formato Excel o pdf mediante una tabla con los siguientes campos: <ul style="list-style-type: none"> - Fecha de realización de denuncia - Fecha del hecho - Tipo de hecho denunciado - Descripción - Estado - Categoría hecho - Subcategoría hecho - Observaciones - Usuario revisor

Tabla 38. Tarea 21.

Tarea	
ID	T22
Historia asociada	HU07
Sprint	2
Nombre	Exportar reportes generados en PDF o Excel
Descripción	Permitir la generación y descarga de archivos con los resultados obtenidos

Tabla 39. Tarea 22.

Tarea	
ID	T23
Historia asociada	HU07
Sprint	2
Nombre	Generar gráficos y estadísticas basados en denuncias

Descripción	<p>El sistema debe generar dos tableros de control con gráficos estadísticos que reflejen una síntesis de interés para las autoridades policiales. Los mismos estarán divididos en dos categorías: gráficos con intervalos de fechas dinámicas y gráficos anuales.</p> <p>1- Los gráficos con intervalo de fechas deberán tener como filtro una "fecha inicial" y una "fecha final" y mostrar la siguiente información:</p> <ul style="list-style-type: none"> - Cantidad total de hechos denunciados - Cantidad total de hechos aceptados - Cantidad total de hechos desestimados - Cantidad total de denuncias revisadas - Cantidad total de dispositivos baneados - Cantidad de hechos denunciados por tipo - Cantidad de hechos denunciados por categoría - Cantidad de hechos denunciados por subcategoría <p>2- Los gráficos anuales poseerán un filtro con un listado desplegable de años. Además, deberán mostrar la siguiente información:</p> <ul style="list-style-type: none"> - Cantidad de denuncias realizadas por mes en el año seleccionado - Cantidad de denuncias revisadas por mes en el año seleccionado - Cantidad de dispositivos registrados por mes en el año seleccionado - Cantidad de dispositivos baneados por mes en el año seleccionado - Cantidad de hechos denunciados por año (gráfico fijo)
-------------	---

Tabla 40. Tarea 23.

Tarea	
ID	T24
Historia asociada	HU08
Sprint	2
Nombre	Generar un mapa de delitos denunciados
Descripción	<p>El sistema debe permitir generar un mapa de puntos con la geolocalización de denuncias aceptadas. Cuando un usuario cliquee sobre un punto, se mostrará la siguiente información:</p> <ul style="list-style-type: none"> - Fecha hecho - Fecha denuncia - Categoría hecho denunciado - Subcategoría hecho denunciado - Botón "ver detalles". Redirige hacia todos los datos de la denuncia. <p>Los filtros para la generación del mapa serán:</p> <ul style="list-style-type: none"> - Intervalo de fechas - Categoría delito

	- Subcategoría delito
--	-----------------------

Tabla 41. Tarea 24.

Tarea	
ID	T25
Historia asociada	HU09
Sprint	1
Nombre	Alta de usuarios sistema web
Descripción	<p>Cada usuario deberá contener la siguiente información:</p> <ul style="list-style-type: none"> - Nombre y apellido - Dependencia - Nombre de usuario - Password - Fecha de alta - Correo electrónico - Autorizado [Sí/No] - Fecha de validación de cuenta - Borrado [Sí/No] <p>Al momento de crear un usuario, el administrador debe colocar la contraseña de la cuenta y notificar al mismo.</p>

Tabla 42. Tarea 25.

Tarea	
ID	T26
Historia asociada	HU09
Sprint	1
Nombre	Editar usuarios sistema web
Descripción	Editar los datos almacenados de los usuarios.

Tabla 43. Tarea 26.

Tarea	
ID	T27
Historia asociada	HU09
Sprint	1
Nombre	Eliminar usuarios sistema web
Descripción	El borrado de usuarios debe ser de forma lógica, no física. Cambiando el atributo "Borrado" de los mismos.

Tabla 44. Tarea 27.

Tarea	
ID	T28
Historia asociada	HU10
Sprint	1
Nombre	Alta permisos sistema web
Descripción	Atributos permisos: - Nombre

Tabla 45. Tarea 28.

Tarea	
ID	T29
Historia asociada	HU10
Sprint	1
Nombre	Editar permisos sistema web
Descripción	Atributos permisos: - Nombre

Tabla 46. Tarea 29.

Tarea	
ID	T30
Historia asociada	HU10
Sprint	1
Nombre	Eliminar permisos sistema web
Descripción	

Tabla 47. Tarea 30.

Tarea	
ID	T31
Historia asociada	HU10
Sprint	1
Nombre	Alta roles sistema web

Descripción	Atributos permisos: - Nombre - Permisos asociados
-------------	---

Tabla 48. Tarea 31.

Tarea	
ID	T32
Historia asociada	HU10
Sprint	1
Nombre	Editar roles sistema web
Descripción	Atributos permisos: - Nombre - Permisos asociados

Tabla 49. Tarea 32.

Tarea	
ID	T33
Historia asociada	HU10
Sprint	1
Nombre	Eliminar roles sistema web
Descripción	

Tabla 50. Tarea 33.

Tarea	
ID	T34
Historia asociada	HU11
Sprint	1
Nombre	Login sistema web
Descripción	Para ingresar al sistema un usuario debe proporcionar: - Nombre de usuario - Contraseña.

Tabla 51. Tarea 34.

Tarea	
ID	T35
Historia asociada	HU11
Sprint	1
Nombre	Restablecer contraseña sistema web
Descripción	Mecanismo de restablecimiento de contraseña vía email para el usuario ante un olvido de su contraseña e imposibilidad de ingreso al sistema.

Tabla 52. Tarea 35.

Tarea	
ID	T36
Historia asociada	HU11
Sprint	1
Nombre	Pantalla "Mi perfil" usuario sistema web
Descripción	Brindarle al usuario la posibilidad de visualizar los datos que se tienen del mismo. Además, desde esta pantalla el usuario debe poder cambiar su contraseña.

Tabla 53. Tarea 36.

Tarea	
ID	T37
Historia asociada	HU12
Sprint	1
Nombre	Aplicación móvil disponible en Google Play Store
Descripción	Subir la aplicación a Google Play Store

Tabla 54. Tarea 37.

Tarea	
ID	T38
Historia asociada	HU13
Sprint	1
Nombre	Sistema web disponible para navegadores modernos
Descripción	Hacer pruebas del sistema web en Google Chrome y Mozilla Firefox

Tabla 55. Tarea 38.

Tarea	
ID	T39
Historia asociada	HU14
Sprint	1
Nombre	Garantizar el borrado lógico de datos
Descripción	Corroborar que la información de la base de datos referida a usuarios, dispositivos y denuncias se borran de manera lógica solamente.

Tabla 56. Tarea 39.

Tarea	
ID	T40
Historia asociada	HU07
Sprint	2
Nombre	Editar datos revisión

Descripción	Los datos de la revisión pueden ser editados por un analista ante la presencia de errores o falta de información
-------------	--

Tabla 57. Tarea 40.

Tarea	
ID	T41
Historia asociada	HU01
Sprint	3
Nombre	Securizar rutas de API
Descripción	Las rutas de la API, con excepción de la que permite identificar el dispositivo denunciante, deben ser accedidas solamente si la petición entrante contiene un token de autenticación válido. Caso contrario devolver una respuesta HTTP con código de error 400 (Bad request).

Tabla 58. Tarea 41.

Tarea	
ID	T42
Historia asociada	HU01
Sprint	3
Nombre	Configurar código captcha en API
Descripción	Colocar las claves necesarias para la utilización de Google Recaptcha en el archivo de configuración de la API.

Tabla 59. Tarea 42.

Tarea	
ID	T43
Historia asociada	HU01
Sprint	3
Nombre	Anexar la validación de código captcha en la identificación de dispositivos de la API
Descripción	Validar que una petición hacia la ruta dispositivo cuente con un parámetro captcha y el mismo sea válido.

Tabla 60. Tarea 43.

Tarea	
ID	T44
Historia asociada	HU01
Sprint	3
Nombre	Pantalla de inicio de la aplicación móvil

Descripción	La pantalla de inicio de la aplicación móvil deberá contar con un código captcha para que complete el usuario y un botón para ingresar a la aplicación, el cual solo podrá ser presionado una vez completado el código que lo precede.
-------------	--

Tabla 61. Tarea 44.

Tarea	
ID	T45
Historia asociada	HU01
Sprint	3
Nombre	Configurar código captcha en la pantalla de inicio de la aplicación móvil
Descripción	Colocar la clave pública de Google Recaptcha en la aplicación móvil para que las peticiones puedan ser validadas por los servidores de Google y luego enviar la respuesta de dichos servidores a la API.

Tabla 62. Tarea 45.

Tarea	
ID	T46
Historia asociada	HU15
Sprint	3
Nombre	Actualizar estética nuevas pantallas aplicación móvil
Descripción	Actualizar las nuevas pantallas propuestas según las pautas de diseño que fueron definidas.

Tabla 63. Tarea 46.

Tarea	
ID	T47
Historia asociada	HU15
Sprint	3
Nombre	Pantalla "cargando datos" de la aplicación móvil
Descripción	Realizar una pantalla que permita indicar al usuario cuando se están procesando datos. La misma deberá hacer su aparición cada vez que una petición es realizada a un servidor.

Tabla 64. Tarea 47.

6. Diseño de base de datos

6.1 Definir los datos necesarios

En esta sección, recopilamos los datos necesarios para el funcionamiento de los sistemas y que fueron persistidos en las tablas de la base de datos. Cabe destacar, que ciertas tablas ya se

encontraban precargadas previamente a la primera ejecución del sistema. Las mismas, son aquellas que permiten brindar opciones preseleccionadas a los usuarios, por ejemplo, categorías de delitos, y perduran en el tiempo con valores fijos. A continuación, describimos las tablas principales de la base de datos y sus valores por defecto si corresponde:

- Denuncia: entidad que modela los datos ingresados por un ciudadano desde la aplicación móvil. Para este proyecto entenderemos a las exposiciones como denuncias, ya que una “denuncia anónima” no se puede definir, en el ámbito jurídico, como denuncia a falta de una parte esencial, el denunciante. Contendrá los siguientes datos: delito según app móvil, el/los delitos/s según la revisión, descripción, fecha y hora aproximada del hecho, detalles y ubicación
- Revisión: Entidad que modela la tarea que realiza un auditor policial con los datos que ingresó el ciudadano. Contendrá los siguientes datos: el revisor (usuario del sistema web), fecha de la revisión, observaciones y la relación con la denuncia.
- Categoría: entidad que proporciona una categoría del delito de acuerdo a los criterios relevados para la denuncia. Se definieron los datos que componen esta entidad en la sección 5.2, “[Leyes y resoluciones](#)”
- Subcategoría: entidad que proporciona una subcategoría del delito de acuerdo a los criterios relevados para la denuncia. Se definieron los datos que componen esta entidad en la sección 5.2, “[Leyes y resoluciones](#)”
- Categoría app: entidad que modela los tipos de hechos que puede notificar el ciudadano a través de la aplicación móvil. Se define en conjunto con la institución una tipificación simple para que el ciudadano pueda reconocer: “Robo/Hurto”, “Pelear”, “Amenazas”, “Violencia de género”, “Otros”.
- Dispositivo: entidad que contiene los datos asociados al dispositivo desde el cual se realiza una denuncia.
- Usuario: entidad que contiene los datos relacionados identificar a los agentes policiales que utilizan el sistema web (nombre y apellido, documento, dependencia de trabajo, email). Se especifican los datos que contendrá está en la sección próxima
- Rol: contiene los datos asociados a los roles del sistema web. Especifica qué conjunto de permisos en el sistema web tendrá el usuario.

- Permiso: contiene los datos asociados a los permisos del sistema web. Lectura, escritura y modificación de datos de cierta entidad. Solo para los usuarios del sistema web. Los permisos definidos fueron:
 - Generar-mapas
 - Generar-reportes
 - Generar-estadísticas
 - Auditar-denuncias
 - Gestionar-permisos
 - Gestionar-roles
 - Gestionar-usuarios
 - Visualizar-denuncias
 - Analizar-denuncias
 - Dar-strike-dispositivo
 - Desestimar-denuncias
- Rol: contiene los datos asociados a los roles del sistema web. Especifica qué conjunto de permisos en el sistema web tendrá el usuario. Fueron definidos los siguientes roles con sus permisos asociados:
 - Auditor-policía: se asignó este rol a todos los agentes que auditan las denuncias realizadas por el ciudadano. Los mismos pueden visualizar denuncias, desestimarlas y dar un strike a un dispositivo si lo consideran necesario. Permisos asociados:
 - Auditar-denuncias
 - Visualizar-denuncias
 - Desestimar-denuncias
 - Dar-strike-dispositivo
 - Analista-policía: rol asignado a los agentes encargados de manipular la información recolectada por el sistema. Se les permite generar reportes, gráficos y mapas con el fin de redactar informes que brindan soporte para la toma de decisiones. Permisos asociados:
 - Generar-estadísticas
 - Generar-mapas
 - Generar-reportes
 - Visualizar-denuncias

- Analizar-denuncias
- Desestimar-denuncias
- Dar-strike-dispositivo
- Administrador-policía: rol asignado a los agentes que se dedican a gestionar los usuarios del sistema web. Por razones de seguridad no poseen permisos para desarrollar las acciones de otros roles. Permisos asociados:
 - Gestionar-usuarios
- Administrador-informática: rol asignado a los desarrolladores de la División Informática de la Policía de Entre Ríos. Pueden realizar todas las acciones disponibles en el sistema. Permisos asociados:
 - Auditar-denuncias
 - Visualizar-denuncias
 - Desestimar-denuncias
 - Dar-strike-dispositivo
 - Generar-estadísticas
 - Generar-mapas
 - Generar-reportes
 - Analizar-denuncias
 - Gestionar-usuarios

6.2 Realizar un diagrama Entidad-Relación y de tablas de base de datos

Un diagrama de entidad relación es utilizado para representar el dominio de un sistema de información, modelando el mismo a través de entidades, las cuales cuentan con atributos y se relacionan entre sí.

En el Anexo 11.1, puede observarse el diagrama completo correspondiente al proyecto, basado en las entidades definidas en la sección anterior.

Además del Modelo Entidad-Relación, realizamos un diagrama de tablas de base de datos (ver Anexo 11.2), para definir la estructura y los atributos de la misma. A continuación, detallaremos la definición de cada tabla con sus correspondientes atributos.

6.2.1 Tabla usuarios

La tabla usuarios está destinada a almacenar datos sobre el personal policial que utilizará el sistema web. Además de datos personales, se dedica al almacenamiento de datos de la cuenta de usuario y sobre el estado de la misma, es decir, si está autorizada, validada o eliminada.

Cabe destacar que el atributo email debe ser único, ya que es el medio por el cual un usuario puede restablecer su contraseña.

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador numérico incremental de la tabla
nombre_usuario	VARCHAR (30)	SI	SI	Nombre del usuario en el sistema
password	VARCHAR (100)	SI	NO	Contraseña del usuario, se almacena encriptada
nombre	VARCHAR (30)	SI	NO	Nombre del usuario
apellido	VARCHAR (30)	SI	NO	Apellido del usuario
dni	INTEGER (8)	SI	SI	Documento nacional de identidad del usuario
email	VARCHAR (30)	SI	SI	Email del usuario
dependencia	VARCHAR (100)	SI	NO	Dependencia a la que pertenece el usuario
autorizado	TINYINTEGER	SI	NO	Determina si el usuario puede ingresar o no al sistema. 0: no autorizado 1: autorizado default:1
validado	DATETIME	NO	NO	Fecha en la que el usuario validó su cuenta dentro del sistema. default: null
borrado	TINYINTEGER	SI	NO	Determina si el usuario fue borrado del sistema. 0: no borrado 1: borrado default:0

fecha_borrado	DATETIME	NO	NO	Fecha en la que se eliminó la cuenta del usuario del sistema. default: null
---------------	----------	----	----	--

Tabla 65. Tabla de base de datos "usuarios".

6.2.2 Tabla roles

La tabla Roles está destinada a almacenar datos sobre los roles que se definen para acceder a secciones del sistema web. Cada uno de ellos debe tener permisos asociados y deberían estar asociados a, al menos, un usuario.

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador numérico incremental de la tabla
nombre	VARCHAR (30)	SI	SI	Nombre del rol

Tabla 66. Tabla de base de datos "roles".

6.2.3 Tabla permisos

La tabla permisos está destinada a almacenar datos sobre los permisos que se definen para acceder a secciones del sistema web. Cada uno de ellos podría formar parte de un rol y podrían estar asociados a un usuario directamente.

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador numérico incremental de la tabla
nombre	VARCHAR (30)	SI	SI	Nombre del permiso

Tabla 67. Tabla de base de datos "permisos".

6.2.4 Tabla revisiones

La tabla revisiones está destinada a almacenar datos sobre los procesos de revisión que realizan los auditores policiales sobre las denuncias de ciudadanos.

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador numérico incremental de la tabla

id_denuncia	INTEGER	SI	SI	Identificador de la denuncia revisada.
id_usuario	INTEGER	SI	NO	Identificador del usuario auditor que realiza la revisión.
observaciones	VARCHAR (2500)	NO	NO	Datos adicionales sobre la denuncia
fecha_revision	DATETIME	SI	NO	Fecha en la que fue realizada la revisión.

Tabla 68. Tabla de base de datos “revisiones”.

6.2.5 Tabla categorias_denuncia

La tabla categorias_denuncia está destinada a almacenar datos sobre las categorías delictuales a las que puede pertenecer un hecho denunciado. Se utilizan al momento de revisar una denuncia. Cada denuncia revisada debe contar con, al menos, una categoría asociada obligatoriamente.

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador numérico incremental de la tabla
nombre	VARCHAR (80)	SI	SI	Nombre de la categoría de delito.
borrado	TINYINT	SI	NO	Determina si la categoría fue borrada del sistema. 0: no borrada 1: borrada default:0

Tabla 69. Tabla de base de datos “categorias_denuncia”.

6.2.6 Tabla subcategorias_denuncia

La tabla subcategorias_denuncia está destinada a almacenar datos sobre las subcategorías delictuales a las que puede pertenecer un hecho denunciado. Se utilizan al momento de revisar una denuncia. Cada denuncia revisada puede o no tener una/s subcategoría/s asociada/s.

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
--------	-----------------------	-----------	-------	-------------

id	INTEGER	SI	SI	Identificador numérico incremental de la tabla
id_categoria_denuncia	INTEGER	SI	SI	Identificador de la categoría a la que pertenece la subcategoría.
nombre	VARCHAR (80)	SI	SI	Nombre de la subcategoría de delito.
borrado	TINYINT	SI	NO	Determina si la subcategoría fue borrada del sistema. 0: no borrada 1: borrada default:0

Tabla 70. Tabla de base de datos “subcategorias_denuncia”.

6.2.7 Tabla tipo_denuncia_app

La tabla tipo_denuncia_app está destinada a almacenar datos sobre los tipos de hechos que podrán denunciar los ciudadanos a través de la aplicación móvil.

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador numérico incremental de la tabla
nombre	VARCHAR (30)	SI	SI	Nombre del tipo de hecho a denunciar.

Tabla 71. Tabla de base de datos “tipo_denuncia_app”.

6.2.8 Tabla dispositivos

La tabla dispositivos está destinada a almacenar datos sobre los dispositivos móviles que utilizan los ciudadanos para realizar denuncias. Cada dispositivo tiene asociado un token numérico generado por el sistema que sirve para identificar las denuncias realizadas por el mismo dentro del sistema (ver Capítulo 9.1). Además, cuenta los strikes del dispositivo y si el mismo se encuentra baneado en el sistema (ver Capítulo 9.2).

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador

				numérico incremental de la tabla
token	VARCHAR (100)	SI	SI	
token_instalacion	VARCHAR (100)	SI	SI	
fecha_alta	DATETIME	SI	NO	
fecha_ultima_denuncia	DATETIME	NO	NO	
strikes	TINYINT	SI	NO	
baneado	TINYINT	SI	NO	
fecha_baneo	DATETIME	NO	NO	

Tabla 72. Tabla de base de datos “dispositivos”.

6.2.9 Tabla denuncias

La tabla denuncias está destinada a almacenar datos sobre las denuncias realizadas por los ciudadanos. Cada denuncia posee un dispositivo asociado y un estado único, los cuales, como se mencionó en la sección 7.9, son: Por auditar, Desestimada y Aceptada.

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador numérico incremental de la tabla
id_tipo_denuncia_app	INTEGER	SI	SI	Identificador de la categoría a la que pertenece la subcategoría.
id_dispositivo	INTEGER	SI	SI	Identificador de la categoría a la que pertenece la subcategoría.
fecha	DATETIME	NO	NO	Fecha del hecho reportada por el ciudadano
coordenadas	POINT	SI	NO	Coordenadas donde ocurrió el hecho

estado	VARCHAR (30)	SI	NO	Estado actual de la denuncia.
descripcion	VARCHAR (2500)	SI	NO	Descripción del hecho denunciado.
fecha_alta	DATETIME	SI	NO	Fecha en la que la denuncia fue escrita en la base de datos.
audio_url	VARCHAR (191)	NO	NO	Dirección web donde se aloja el audio enviado por el ciudadano (si lo hubiere).

Tabla 73. Tabla de base de datos “denuncias”.

6.2.10 Tabla activity_log

La tabla activity_log está destinada a almacenar datos sobre las operaciones que los usuarios policiales realizan sobre la base de datos. La misma no figura en el diagrama de base de datos porque fue generada por un paquete de Laravel [7], el cual decidimos utilizar debido a la automatización que brinda a las operaciones y sus buenas prácticas de implementación. Una descripción detallada de la misma puede verse en la siguiente tabla:

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
id	INTEGER	SI	SI	Identificador numérico incremental de la tabla
log_name	VARCHAR (255)	SI	NO	Tipo de operación realizada.
description	VARCHAR (255)	NO	NO	Observaciones adicionales respecto a la operación realizada.
subject_id	INTEGER	SI	NO	Identificador del modelo con el que trabajó el usuario del sistema.
subject_type	VARCHAR (255)	SI	NO	Nombre del modelo con el que trabajó el usuario del sistema.
causer_id	INTEGER	SI	NO	Identificador del usuario del sistema que realizó la operación.
causer_type	VARCHAR	SI	NO	Nombre del modelo del usuario

	(255)			del sistema que realizó la operación.
properties	TEXT	SI	NO	Atributos del modelo que fueron modificados y sus valores.
created_at	DATETIME	SI	NO	Fecha de creación del registro.
updated_at	DATETIME	SI	NO	Fecha de actualización del registro.

Tabla 74. Tabla de base de datos "activity_log".

6.2.11 Tabla password_reset

La tabla password_reset está destinada a la recuperación de las contraseñas de los usuarios del sistema web. La misma es una funcionalidad proporcionada por Laravel [8].

Nombre	Tipo de dato (tamaño)	Requerido	Único	Descripción
email	VARCHAR (80)	SI	NO	Email del usuario del sistema que quiere restablecer su contraseña.
token	VARCHAR (100)	SI	NO	Token que permite vincular el usuario y la cuenta al momento de realizar la operación.
created_at	DATETIME	NO	NO	Fecha en la que se solicitó el restablecimiento de contraseña.

Tabla 75. Tabla de base de datos "password_reset".

6.2.12 Tablas intermedias

Como puede observarse del diagrama de base de datos, el mismo cuenta con varias tablas intermedias. Debido a que los atributos de las mismas son claves primarias de las tablas a las que hacen referencia, sólo describiremos las mismas a continuación:

- rol_usuario
- permiso_usuario
- permiso_rol
- categoria_denuncia
- subcategoria_denuncia

7. Tecnologías y arquitectura

7.1 *Arquitectura del proyecto*

La arquitectura de un proyecto de software tiene como objetivo representar los componentes del mismo de una manera clara y estandarizada. Para ello se vale de diferentes modelos llamados vistas, los cuales representan de manera homogénea los diferentes aspectos del sistema a implementar a través de componentes y sus relaciones entre sí.

Para este proyecto se optó por las siguientes vistas para una comprensión más global de las conexiones y división del sistema:

- Vista módulo: lista los módulos que representan un paquete de funcionalidades del sistema final, que requieran una gran cobertura arquitectónica o aquellos que impliquen algún punto especialmente delicado de la arquitectura.
- Vista componente y conector: definen modelos compuestos por elementos que tienen alguna presencia en tiempo de ejecución (procesos, objetos, clientes, servidores, almacenamiento de datos, etc.) y el camino de interacción entre ellos (links, protocolos de comunicación, flujo de información, acceso a almacenamiento compartido, etc.).

7.1.1 Objetivos y restricciones

Al momento de diseñar la arquitectura del proyecto, realizamos un análisis sobre los atributos de calidad que serían deseables de cumplir con la implementación. Los mismos se listan a continuación:

7.1.1.1 Escalabilidad

Es necesario que el proyecto sea escalable, con el fin de que en un futuro se puedan agregar funcionalidades según se requiera, por ejemplo, una adaptación para hacer comparaciones con las denuncias formales que sirven para prevención de hechos y calificar la efectividad de las denuncias anónimas.

7.1.1.2 Usabilidad

Como un componente del sistema es una aplicación móvil de uso público y general, no es posible llevar adelante una capacitación a usuarios. Esto hace que sea importante la facilidad de su uso para todo tipo de personas.

7.1.1.3 Interoperabilidad

El sistema móvil y el web deben interoperar para comunicar la información, además, en la Fuerza existen sistemas que en un futuro podrían necesitar conectarse para mejorar la predicción y prevención de delitos

7.1.1.4 Seguridad

Al tratarse de datos de denuncias anónimas es importante asegurar el anonimato del denunciante, como así también, salvaguardar la información para que no atente contra la alteración de la estadística real.

7.1.2 Vista de módulos

En esta vista se busca representar los módulos importantes que puedan resolver los objetivos anteriormente especificados.

7.1.2.1 Descripción de módulos de App móvil

Layer de Presentación (Layer)

- Menú: módulo en el que se encuentra la vista gráfica de las opciones que ofrece el sistema y la presentación del mismo
- Mapa de comisarías: es una vista en un mapa de las ubicaciones geográficas de las comisarías, sus jurisdicciones y la ubicación del dispositivo
- Guías Telefónicas: se utiliza un buscador básico para la información de las distintas dependencias policiales (nombre, dirección y teléfono)
- Trámites: lista de los trámites que se realizan en la policía para público general y como realizarlos.
- Denuncias: vista dividida para facilitar el alta de denuncias anónimas en 4 pasos: tipo de delito (se especifica el delito general), fecha y hora, ubicación (selecciona la ubicación en un mapa, es opcional), descripción (permite escribir o grabar un audio de lo sucedido)

Lógica de Negocios

- Carga mapa: módulo con funciones para la carga de mapas y la llamada al servicio externo de Google Maps
- Listar Teléfonos: aquí se hace la búsqueda y conexión con el servicio externo de Google FireBase de la Policía de Entre Ríos y los listados telefónicos.

- Listar Trámites: conexión con el servicio externo de Google FireBase de la Policía de Entre Ríos con los listados de trámites.
- Alta Denuncia: módulo que permite obtener los tipos de delito que pueden ser registrados a través de la aplicación (vía web services) y enviar la denuncia completa a un servidor.
- Validación: se trata de un módulo con las funciones generales de validación de datos ingresados.

Acceso a Datos

- SQL local: maneja las conexiones con la base de datos local.
- Shared Preferences: contiene datos de configuración local de la aplicación.

7.1.2.2 Descripción de módulos de Página Web

Layer de Presentación (Layer)

- Login: módulo de inicio de acceso a la página
- Lista de denuncia: módulo con listados y búsqueda de las denuncias cargadas
- Estadísticas: sección que contiene estadísticas y gráficos básicos del sistema, presentando de forma sencilla los números de denuncias previamente revisados y aceptados (que un usuario de control realizó la revisión y categorización correspondiente del hecho)
- Revisión de denuncias: sección dedicada a los usuarios de control que deben leer y/o escuchar la descripción de la denuncia cargada a través de la aplicación móvil para clasificarla y aceptarla o rechazarla (si es spam o inentendible).

Lógica de Negocios

- Login: módulo con las funciones correspondientes a la autenticación de usuarios y validez de los datos ingresados
- Filtrado y listado: parte de la lógica que permite realizar las búsquedas, filtrados y listados de las denuncias
- Reportes: muy similar al módulo anterior, pero con estadísticas, mapas, números y gráficas
- Alta y modificación de revisiones: funciones dedicadas a la modificación de denuncias revisadas anteriormente y alta de revisiones de denuncias del sistema móvil con sus correspondientes validaciones.

Acceso a Datos

- PostgreSQL: maneja las conexiones con la base de datos del sistema, altas, modificaciones y selecciones.

7.1.2.3 Descripción de módulos de API

Lógica de Negocios

- Alta denuncia: funciones de post para el alta de denuncia y validaciones
- Listados de delitos: función get de listado para la app móvil de los delitos generales.

Acceso a Datos

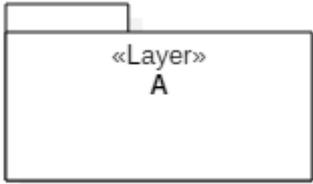
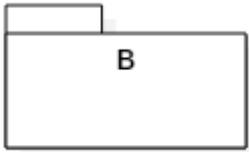
- PostgreSQL: maneja las conexiones con la base de datos del sistema, altas y selecciones.

7.1.2.4 Descripción del módulo Cross

Tanto la app móvil como el sistema web tienen su módulo cross correspondiente que maneja:

- Seguridad: contiene diversos servicios de seguridad para cada layer y las diversas amenazas que puede encontrar en cada uno.
- Logs: Maneja los diversos errores y da aviso o guarda la información dependiendo cada caso.

7.1.2.5 Representación de componentes de un diagrama en vista de módulos

Nombre componente	Representación
Banda	
Comunicación	
Módulo	

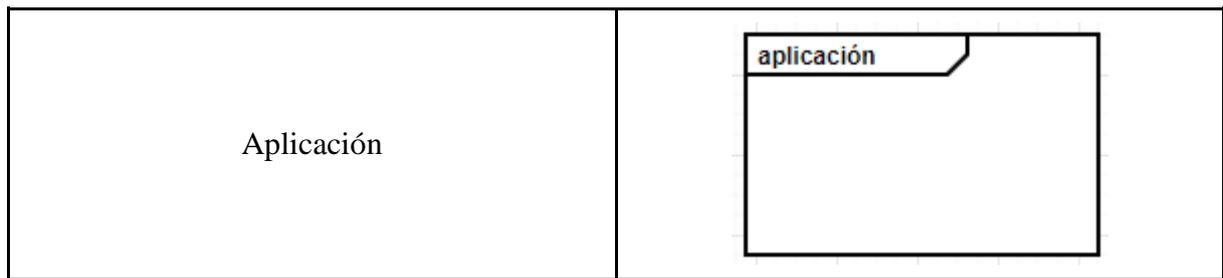


Tabla 76. Representación de componentes de un diagrama de módulos.

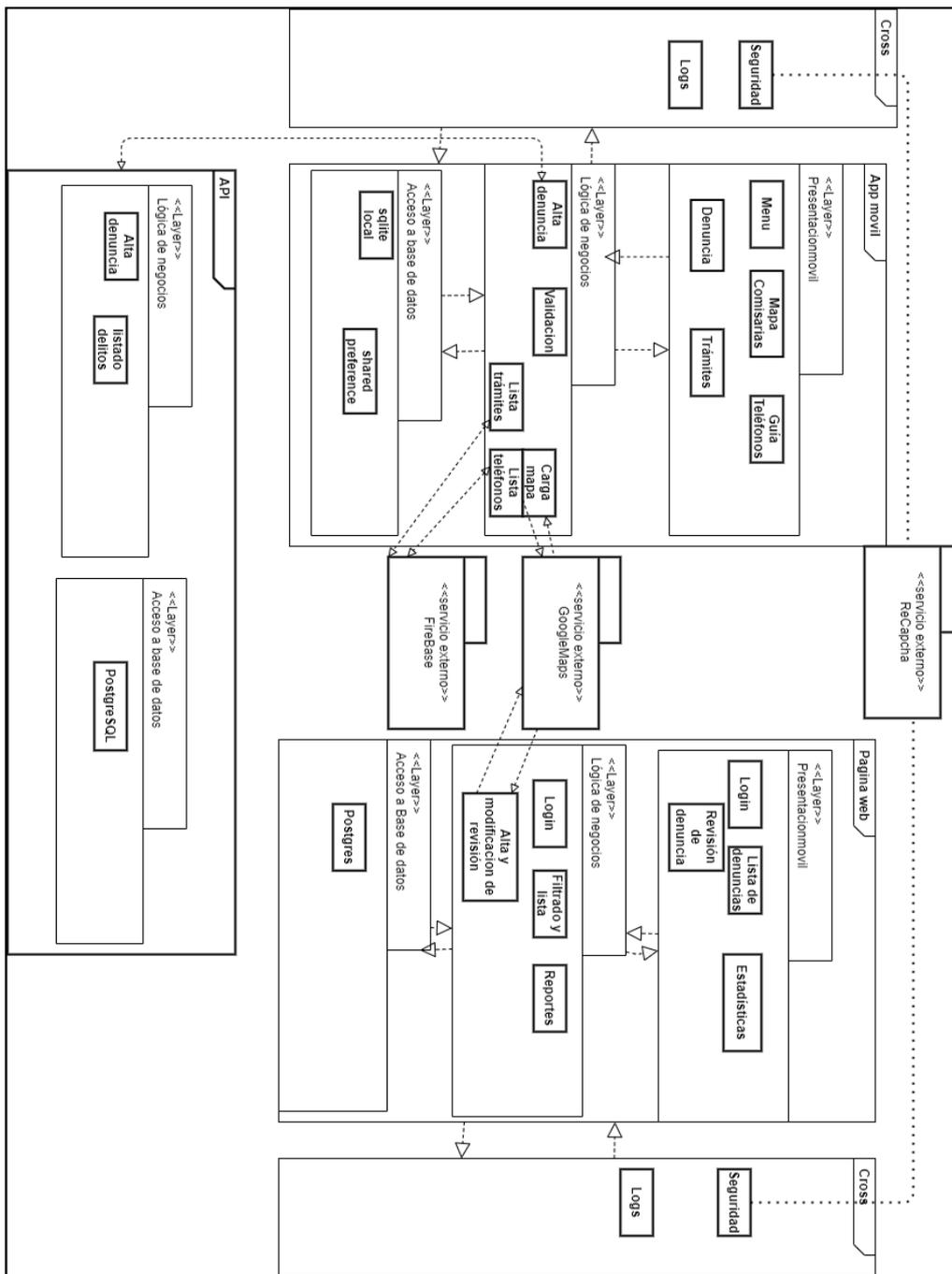


Imagen 1. Diagrama de módulos del proyecto.

7.1.3 Vista de Componente y conector

7.1.3.1 Objetivo

Mostrar el funcionamiento dinámico de algunos componentes del sistema y estudiar las vistas que puedan resolver los problemas que se plantearon en los escenarios de calidad.

7.1.3.2 Representación de elementos de la vista

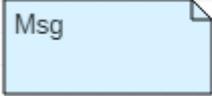
Nombre componente	Representación
Request Mensaje	
Respuesta del mensaje	
Componente	
Interfaz del componente	
Request	

Tabla 77. Representación de elementos de la vista.

7.1.3.3 Vista autenticación

Se utiliza un servicio externo para comprobar que sea un humano el que utiliza la app, para evitar el spam y/o otras amenazas. Comunica con un mensaje que pasó la comprobación y recibe una clave encriptada. Luego, esta es enviada junto con una clave especial que identifica la aplicación. Para responder al mensaje la API primero verifica con el servicio de Google Recaptcha que se hayan enviado los datos correctos, los guards asociados a un usuario y responde con una clave de autenticación que será necesaria para que consuma los otros servicios de la API de denuncias

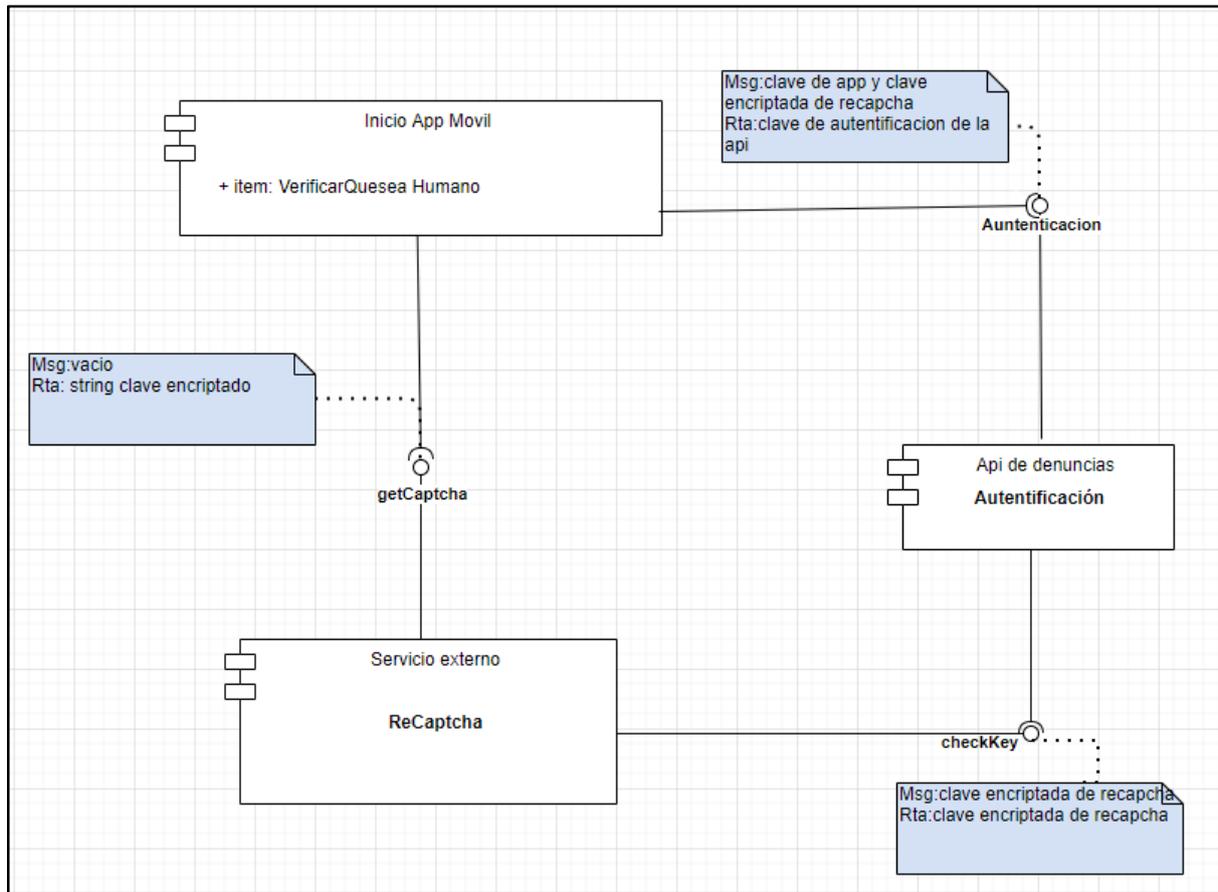


Imagen 2. Vista autenticación.

7.1.3.4 Vista alta de denuncias y lista de delitos

El propósito de esta comunicación es dar de alta una denuncia en la base de datos del sistema y obtener una lista actualizada de los tipos de delitos que se pueden cargar.

Para esto se envían mensajes con cabecera de autenticación como se muestra en el gráfico

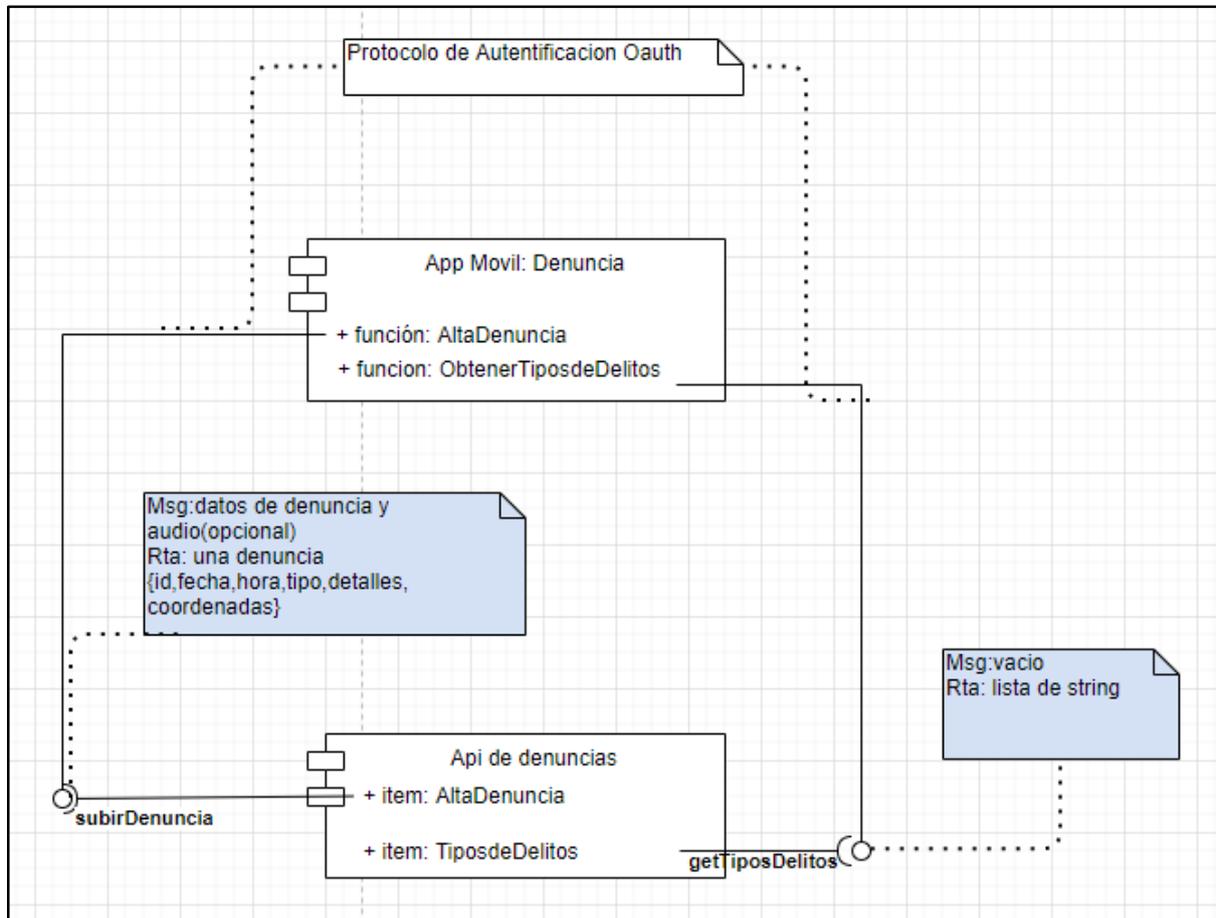


Imagen 3. Vista alta de denuncias y lista de delitos.

7.1.3.5 Vista teléfonos y trámites

Los datos de teléfonos y trámites de las dependencias de la policía están en una base de datos de Google FireBase, entonces usamos la forma de comunicación que nos ofrece servicio y llamamos a las funciones que nos traerán los datos de la base

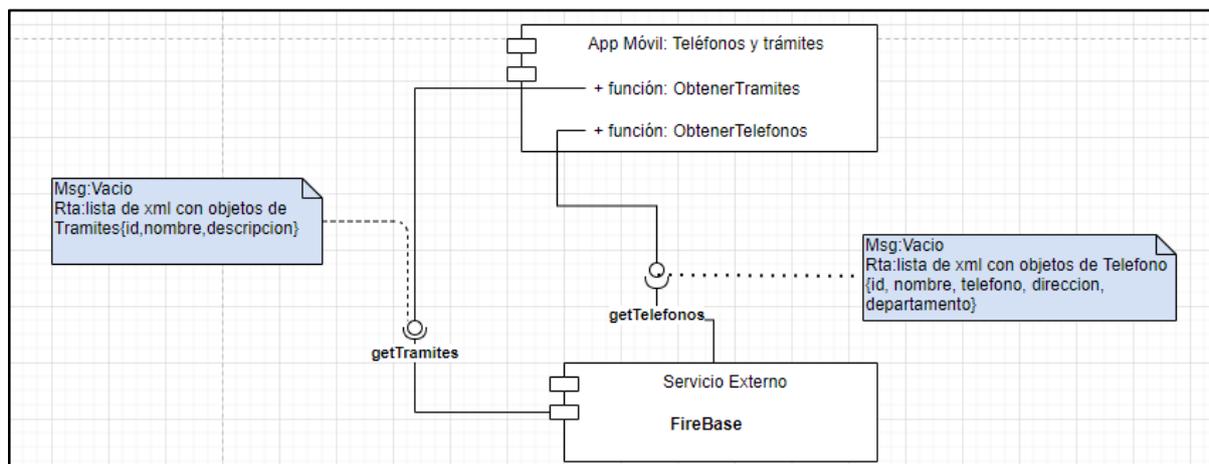


Imagen 4. Vista teléfonos y trámites

7.2 Tecnologías utilizadas para la solución

Basados en las necesidades del proyecto, decidimos utilizar dos tecnologías principales para la solución del mismo, aplicaciones para dispositivos móviles y sistemas web.

La elección de la primera de ellas se justifica en la comodidad, facilidad de uso y costumbre por parte de los usuarios, en este caso, los habitantes de la provincia de Entre Ríos. La proliferación de los dispositivos móviles ha crecido exponencialmente con el paso de los años, la brecha generacional entre usuarios de los mismos ha disminuido considerablemente, a tal punto que personas de la tercera edad se sienten más cómodos con un dispositivo de este tipo comparado con una computadora personal. Además, debido a su bajo costo y popularidad, es altamente probable que una familia disponga de al menos un dispositivo en su hogar.

La elección de un sistema web donde se puedan rectificar datos y obtener información sobre los mismos se basa en los recursos con que ya cuenta la Fuerza Policial de la provincia, la extensión de territorio a cubrir y la facilidad con la que se puede distribuir el producto al añadir nuevas funcionalidades al mismo.

7.3 Motor de base de datos

Al momento de elegir el motor de base de datos a utilizar, optamos por uno en el cual tuviéramos experiencia previa, de código abierto y con una amplia comunidad que lo utilice. Basados en dichas características iniciales, se propusieron dos opciones a considerar: MySQL y PostgreSQL.

MySQL se utiliza cuando las consultas suelen ser pequeñas y, si es necesario hacer modificaciones, no demorar mucho tiempo en ello. En otras palabras, es ideal para bases de datos pequeñas y medianas.

PostgreSQL brinda opciones más complejas que MySQL, ya que suele estar orientado para bases de datos más grandes y con consultas más largas y complejas. Además, cuenta con extensiones muy útiles y probadas que permiten extender sus funcionalidades.

En este caso, las consultas a la base de datos son simples y se estima que el crecimiento de la base de datos escale rápidamente en el tiempo (cada ciudadano puede realizar muchas denuncias desde un dispositivo móvil). Además, PostgreSQL cuenta con una extensión denominada PostGIS, la cual permite optimizar y ampliar las funcionalidades del motor para trabajar con datos geográficos, lo cual es muy útil para los requerimientos del proyecto.

Es por lo expresado anteriormente que el motor de base de datos utilizado para la solución fue PostgreSQL.

7.4 Lenguajes de desarrollo de software

7.4.1 Sistema web y API

Como se mencionó anteriormente, se utilizaron sistemas web en la solución final. Debido a su interacción constante con la base de datos y su falta de necesidad de acceso a través de aplicaciones de terceros, se decidió utilizar el lenguaje PHP (Hypertext Preprocessor), ya que es el lenguaje de código abierto más utilizado para esta clase de proyectos y cuenta con soporte que facilita el despliegue y mantenimiento del mismo.

Basado en los requerimientos previamente planteados, la seguridad de acceso y la cantidad de datos a gestionar, se propuso el uso de Laravel Framework en su versión 5.4. Laravel es el esquema de trabajo (con base PHP) más usado en el mundo, cuenta con políticas de seguridad ya integradas, integración sencilla con múltiples motores de bases de datos y metodologías de organización de archivos eficiente para la mantenibilidad de sistemas que escalen en el tiempo.

7.4.2 Aplicación móvil

Para determinar el lenguaje de programación a utilizar para realizar la aplicación móvil, realizamos un análisis de dos alternativas: aplicaciones híbridas y nativas.

Las aplicaciones móviles nativas son las que se desarrollan específicamente para cada sistema operativo, iOS, Android o Windows Phone, adaptando a cada uno el lenguaje con el

que se desarrolla: lenguaje Objective-C para iOS, Java para Android, y .Net para Windows Phone. Es decir, hay que realizar el desarrollo de aplicaciones móviles para cada plataforma.

Entre las ventajas de este tipo de aplicaciones destacan que aprovechan las funcionalidades del dispositivo y que pueden funcionar sin conexión a Internet.

Las aplicaciones híbridas se desarrollan bajo lenguaje Javascript, CSS o HTML, al igual que los sistemas web, lo cual permite la adaptación a cualquier sistema operativo (Android o iOS); y, por otro lado, como sucede con las apps nativas, permiten el acceso a las funcionalidades del dispositivo a través de librerías.

Basado en los requerimientos previamente planteados, la experiencia de quienes realizamos el proyecto y la magnitud de mercado que poseen los dispositivos con sistema operativo Android, decidimos realizar la aplicación móvil de manera nativa, utilizando lenguaje Java y Android Studio como entorno de desarrollo integrado.

7.5 Librería para gráficos

Como su nombre lo indica, una librería gráfica permite utilizar Javascript para realizar gráficos en sistemas web. Su principal utilidad radica en posibilitar la conexión con los datos almacenados (a través de un lenguaje de servidor), permitiendo actualizar los gráficos en tiempo real o solo con actualizar el navegador web.

Hay infinidad de librerías para graficar datos en sistemas web. En este caso, elegimos utilizar Highcharts [9], debido a que es gratuita, cuenta con documentación bien detallada con ejemplos y una amplia comunidad de desarrolladores que la utilizan.

7.6 Librería para mapas

Este tipo de librerías permite generar e insertar mapas en sistemas web utilizando Javascript. Su principal utilidad radica en posibilitar la conexión con los datos almacenados (a través de un lenguaje de servidor), permitiendo actualizar los datos que se muestran en los mapas con actualizar el navegador web.

En este caso, elegimos utilizar LeafletJS [10], debido a que es gratuita, cuenta con documentación bien detallada con ejemplos, una amplia comunidad de desarrolladores que la utilizan y plugins útiles para generar distintos tipos de mapas.

7.7 Herramientas adicionales

En esta sección presentaremos las herramientas que utilizamos para el desarrollo del proyecto.

7.7.1 Visual Studio Code

Visual Studio Code [11] es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. También es personalizable, por lo que los usuarios pueden cambiar el tema del editor, los atajos de teclado y las preferencias. Es gratuito y de código abierto.

7.7.2 Node.js

Node.js [12] es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación ECMAScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como, por ejemplo, servidores web.

7.7.3 Postman

Postman [13] es una plataforma colaborativa para el desarrollo y testing de APIs. Construye y gestiona múltiples APIs propias o de terceros. Organiza las peticiones en colecciones y automatiza la integración de servicios API Rest con las aplicaciones desarrolladas.

7.7.4 Git y GitLab

Git [14] es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.

Gitlab [15] es un servicio web de control de versiones y desarrollo de software colaborativo basado en Git. Además de gestor de repositorios, el servicio ofrece también alojamiento de wikis y un sistema de seguimiento de errores, todo ello publicado bajo una Licencia de código abierto.

Además, la herramienta logra facilitar la metodología de versionado que adoptamos y permite controlar las tareas por hacer, en proceso y terminadas. Y conectarse con los distintos entornos de desarrollo que se proponen para este proyecto.

7.7.5 Android Studio

Android Studio [16] es el entorno de desarrollo integrado oficial para la plataforma Android.

Está disponible para las plataformas Microsoft Windows, macOS y Linux y ha sido diseñado específicamente para el desarrollo de Android.

8. Implementación

8.1 Sprints

El Sprint [17] es la unidad básica de trabajo para un equipo Scrum. Es una simple iteración llevada a cabo por los miembros del equipo. Un equipo puede completar varios sprints durante el desarrollo del proyecto. Un Sprint inicia con un equipo que se compromete a realizar el trabajo y finaliza con la demostración de un entregable. El tiempo mínimo para un Sprint es de una semana y el máximo es de 4 semanas.

8.1.1 Sprint Planning

El Sprint Planning [18] es el primer evento de Scrum en dónde se planifican las tareas a realizar en el Sprint en curso. En esta reunión participan, de manera colaborativa, todo el equipo Scrum: Scrum Master, Product Owner y Equipo de Desarrollo.

En este caso, la reunión de planeamiento de los sprint las llevamos a cabo entre los miembros del equipo de trabajo del proyecto, con el fin de evitar atrasos debido a complicaciones en la agenda de los product owner.

8.1.2 Sprint 0

Se conoce como Sprint 0 a la fase inicial del proyecto y es independiente del resto de sprints. En esta etapa es donde se prepara técnica y metodológicamente al equipo para que el proyecto tenga un buen comienzo.

En nuestro caso, utilizamos este sprint con el fin de poner a punto las herramientas con las que íbamos a trabajar en el proyecto y sus configuraciones, principalmente los repositorios de código fuente y creación de la base de datos que utilizamos y el cliente a través de la cual acceder a la misma.

Cabe destacar que las tareas que componen este sprint consideramos no asociarlas a ninguna historia de usuario, ya que las mismas hacen referencia a actividades propias de puesta a punto del proyecto.

Un resumen del sprint puede verse a continuación:

Sprint Backlog 0				
Historia de usuario	Tarea	Horas estimadas	Horas realizadas	Horas restantes
-	Crear el proyecto para el sistema web	0,25	0,25	0
-	Crear el repositorio de Gitlab para el sistema web	0,25	0,25	0
-	Crear el proyecto para la API	0,25	0,25	0
-	Crear el repositorio de Gitlab para la API	0,25	0,25	0
-	Crear el proyecto para la aplicación móvil	2	3	-1
-	Crear el repositorio de Gitlab para la aplicación móvil	0,5	1	-0,5
-	Crear la base de datos	2	2	0

Tabla 78. Sprint Backlog 0.

Duración del sprint [días]: **2**

Total de horas de trabajo [horas]: **7**

Promedio de horas de trabajo por día [horas/día]: **3,5**

8.1.3 Sprint 1

Para el sprint 1 consideramos que el producto entregable debería permitir a un ciudadano cargar una denuncia en la aplicación móvil y a un funcionario policial poder auditar la misma. Para poder realizar esta última opción se trabajó en todo el esquema de gestión del sistema web, con el fin de poder dar de alta usuarios y asignarle sus correspondientes roles y permisos.

Cabe destacar que, al tratarse de un prototipo, decidimos no subir la aplicación móvil a la Playstore de Google para no incurrir en gastos económicos.

Un resumen del sprint puede verse a continuación:

Sprint Backlog 1				
Historia de usuario	Tarea	Horas estimadas	Horas realizadas	Horas restantes
HU15	Realizar bocetos de interfaces	10	8	2

HU01	Formulario alta denuncia	15	27	-12
	Almacenar datos denuncia	3	2	1
	Mensaje de voz en denuncia	20	5	15
	Almacenar mensaje de voz de denuncia	3	2	1
	Permitir identificar un dispositivo denunciante	1	1,5	-0,5
HU05	Revisión de denuncia	3	5,75	-2,75
HU14	Garantizar el borrado lógico de datos	2	5	-3
HU13	Sistema web disponible para navegadores modernos	3	6	-3
HU10	Alta permisos sistema web	1	2,5	-1,5
	Editar permisos sistema web	1	0,5	0,5
	Eliminar permisos sistema web	1	0,5	0,5
	Alta roles sistema web	1	0,25	0,75
	Editar roles sistema web	1	0,25	0,75
	Eliminar roles sistema web	1	0,25	0,75
HU09	Alta de usuarios sistema web	2	2,5	-0,5
	Editar usuarios sistema web	1	1,5	-0,5
	Eliminar usuarios sistema web	1	0,5	0,5
HU04	Listado de denuncias por auditar	1,5	2,5	-1
HU12	Aplicación móvil disponible en Google Play Store	5	0	5
HU11	Login sistema web	1	1,5	-0,5
	Restablecer contraseña sistema web	2	3	-1
	Pantalla "Mi perfil" usuario sistema web	1	3	-2

Tabla 79. Sprint Backlog 1.

Duración del sprint [días]: **23**

Total horas de trabajo [horas]: **81**

Promedio de horas de trabajo por día [horas/día]: **4,05**

8.1.4 Sprint 2

El sprint 2 fue planificado con la intención de implementar todas las funcionalidades restantes del proyecto, las cuales consisten principalmente en la generación de información (reportes, estadísticas, mapas, etc.) y la posibilidad de brindarle al ciudadano un medio de comunicación con dependencias policiales a través de la aplicación móvil.

Un resumen del sprint puede verse a continuación:

Sprint Backlog 2				
Historia de usuario	Tarea	Horas estimadas	Horas realizadas	Horas restantes
HU07	Generar reportes de hechos denunciados	15	5,5	9,5
	Exportar reportes generados en PDF o Excel	12	1	11
	Generar gráficos y estadísticas basados en denuncias	12	12	0
	Editar datos de revisiones realizadas	5	7,5	-2,5
HU06	Desestimar automáticamente denuncias de dispositivos baneados	2	0,5	1,5
HU08	Generar un mapa de delitos denunciados	13	7	6
HU02	Mapa de dependencias policiales	30	29	1
	Llamar a dependencia policial desde el mapa	10	12	-2
HU03	Listado de dependencias policiales	30	28	2
	Llamar a dependencia policial desde el listado	10	11	-1

Tabla 80. Sprint Backlog 2.

Duración del sprint [días]: **31**

Total horas de trabajo [horas]: **113,5**

Promedio de horas de trabajo por día [horas/día]: **3,66**

Este promedio de horas diarias elevado se explica debido a que ambos estudiantes entramos en periodo de vacaciones laborales con el fin de acelerar la ejecución del proyecto, además de ello se sumó la cuarentena obligatoria por coronavirus en el país, por lo que contamos con más tiempo libre para dedicarle a la finalización de la etapa.

8.1.5 Sprint 3

Este sprint surge principalmente de la necesidad de mejorar la seguridad de la API del proyecto, lo cual es sumamente importante debido a que la misma cuenta con acceso público no restringido. Además, se propusieron tareas que ayudan a mejorar la experiencia de usuario y el diseño de interfaces propuesto.

Un resumen del sprint puede verse a continuación:

Sprint Backlog 3				
Historia de usuario	Tarea	Horas estimadas	Horas realizadas	Horas restantes
HU01	Securizar rutas de API	5	1,5	3,5
	Configurar código captcha en API	3	0,5	2,5
	Anexar la validación de código captcha en la identificación de dispositivos de la API	2	0,5	1,5
	Pantalla de inicio de la aplicación móvil	1	4	-3
	Configurar código captcha en la pantalla de inicio de la aplicación móvil	25	15	10
HU15	Actualizar estética nuevas pantallas aplicación móvil	5	3	2
	Pantalla "cargando datos" de la aplicación móvil	4	7	-3

Tabla 81. Sprint Backlog 3.

Duración del sprint [días]: **12**

Total horas de trabajo [horas]: **31,5**

Promedio de horas de trabajo por día [horas/día]: **2,63**

Para el diseño de las tablas presentadas nos basamos en plantillas publicadas en [19] y las personalizamos de acuerdo a nuestras necesidades.

8.2 Instalación de tecnologías y herramientas

En esta sección detallaremos las herramientas y tecnologías que fueron instaladas para el desarrollo del proyecto.

8.2.1 Sistema web

Para realizar el sistema web, se procedió con la instalación de:

- Nuevo proyecto laravel, en su versión 5.4
- Cliente de PostgreSQL, PgAdmin en su versión 4
- Plataforma Postman, versión 7.2
- Entorno NodeJS, versión 10.16.1
- Entorno de desarrollo Visual Studio Code, versión 1.42.1
- Creación de repositorio para alojar el código fuente del proyecto en Gitlab

8.2.2 API del proyecto

Para realizar la API del proyecto, se procedió con la instalación de:

- Nuevo proyecto laravel, en su versión 5.4
- Creación de repositorio para alojar el código fuente del proyecto en Gitlab

8.2.3 Aplicación móvil

Para realizar la aplicación móvil, se procedió con la instalación de:

- Android Studio 4.1 (ultima versión)
- Plugin de para Android Studio de GitLab
- Librerías de las versiones 19(KitKat) a la 28(Pie) del sistema de Android
- Emulador de Android Studio
- Librerías de Google Maps
- Plataforma Postman, versión 7.2

- Librerías de Retrofit

8.3 Definición de módulos de la solución

Un módulo se define como un conjunto de funciones asociadas a un recurso del producto de software a desarrollar. En esta sección brindaremos una breve descripción de los módulos que consideramos necesarios para cumplir con los requerimientos del proyecto.

8.3.1 Módulos sistema web

El sistema web cuenta con los siguientes módulos:

- Permisos: permite crear, editar y eliminar permisos asociados a recursos dentro del sistema.
- Roles: permite crear, editar, eliminar roles y asociar permisos a los mismos.
- Usuarios: permite crear, editar, eliminar usuarios y asociar roles a los mismos.
- Denuncias: permite revisar, auditar y listar una denuncia de acuerdo a su estado y atributos pertinentes.
- Reportes: permite generar un listado de denuncias de acuerdo a filtros de búsqueda. Además, dicho listado puede ser exportado a formato Excel o CSV.
- Estadísticas: permite generar gráficos y estadísticas de acuerdo a filtros de búsqueda.
- Mapas: permite generar mapas y agrupar cantidades de denuncias de acuerdo a filtros de búsqueda.

8.3.2 Módulos aplicación móvil

La aplicación móvil cuenta con los siguientes módulos:

- Denuncia: permite generar una denuncia de acuerdo a los atributos especificados para la misma y, luego, transmitirla al servidor para su auditoría.
- Mapa de dependencias: permite mostrar las áreas de acción de las comisarías y dependencias policiales. Además, un usuario puede comunicarse con las mismas con pulsar dentro de su área de acción.
- Listado de dependencias: listado que permite observar los datos correspondientes a dependencias policiales. Además, un usuario puede comunicarse con las mismas pulsando su botón asociado.
- Trámites: Lista de trámites que se realizan en las distintas dependencias policiales para público en general

8.4 Creación de la base de datos

La base de datos fue creada con el cliente web de PostgreSQL llamado PgAdmin 4. Dentro del servidor por defecto que posee su instalación se procedió a la utilización del asistente para la creación de bases de datos, nombrando a la misma “deciper”. Además, utilizando nuevamente el asistente se habilitó la extensión “postgis”, para mejorar las operaciones asociadas a los datos geográficos dentro de la base creada.

8.5 Creación de tablas en la base de datos

Para crear las tablas de la base de datos se utilizó la funcionalidad llamada “migration” que provee Laravel [20]. A través de ella se pueden definir los atributos que tendrá cada tabla de la base de datos. Además, permite definir propiedades de la tabla y/o atributos como: claves foráneas, nombres, restricciones, etc.

Una vez generadas todas las tablas del sistema, con el comando **php artisan migrate**, se ejecutan las consultas necesarias dentro del motor de base de datos automáticamente, obteniendo como resultado, las tablas de la base de datos sin registros.

Como complemento para la funcionalidad anterior, el framework cuenta con otra función llamada “seeders” [21], la cual permite definir datos con los que inicialmente contarán ciertas tablas, por ejemplo, los permisos del sistema, categorías de delitos, etc.

Una vez generados los archivos seeders necesarios, se pueden ejecutar todos a la vez con el comando **php artisan db:seed** rellenando las tablas correspondientes con datos establecidos.

Para la base de datos del proyecto, se utilizaron seeders para las siguientes tablas:

- categorias_denuncia
- subcategorias_denuncia
- permisos
- roles
- usuarios
- tipo_denuncia_app

8.6 Generación de código fuente y documentación

Como se explicó con anterioridad, para el desarrollo del presente proyecto utilizamos una metodología ad-hoc. Por ello, decidimos realizar reuniones periódicas, pero de carácter

informal, es decir, no llevábamos un registro escrito como suele hacerse en la metodología SCRUM.

Debido a nuestros horarios laborales, estas reuniones eran principalmente de tarde en forma física o remota a través de videollamada.

Decidimos dividir el desarrollo basado en las tecnologías como web y móvil. Luego, basados en la experiencia de cada miembro del equipo, Hugo tomó las tareas relacionadas a la programación web (Sistema y API) y Sonia las tareas relacionadas a programación móvil.

Para llevar adelante las tareas, además de las herramientas mencionadas en la sección 7.7, fueron necesarios los siguientes paquetes/plugins:

8.6.1 Aplicación móvil

Para el desarrollo de la aplicación móvil se requirieron los siguientes plugins:

- **retrofit:** Es una librería por la cual se consumen y dan respuesta de forma simple a APIs
- **recaptcha:** librería que crea, verifica y valida captcha. Por medio de una API que envía una clave encriptada cuando el captcha ha sido resuelto y guarda la clave para que un tercero pueda verificar que no es un bot.
- **Google Maps API:** API para la creación de mapas geoespaciales y la obtención de los datos actualizados del mismo
- **Firebase Base de datos:** de esta aplicación se usaron los plugin para comunicarse con la base de datos que contiene la información de las dependencias de la Policía de Entre Ríos

8.6.2 API

Para el desarrollo de la API se requirieron los siguientes paquetes:

- **laravel/passport:** permite utilizar el protocolo OAuth para autenticar usuarios que se conecten a la API y proteger las rutas de usuarios no autenticados. Lo utilizamos para proteger casi la totalidad de las rutas de acceso al servidor.
- **spatie/flysystem-dropbox:** permite utilizar una cuenta de Dropbox como storage de archivos. La utilizamos para simular un servidor de storage y añadir más realismo al caso práctico almacenando allí los audios enviados en denuncias.

- **anhskohbo/no-captcha:** permite integrar Google Recaptcha a una aplicación. Además, valida automáticamente que un formulario protegido por esta metodología es correcto enviando datos a los servidores de Google y procesando la respuesta. Utilizado para validar un dispositivo que requiere ingresar a la aplicación móvil.

8.6.3 Sistema Web

Para el desarrollo del sistema web se requirieron los siguientes paquetes/librerías:

- **box/spout:** paquete que permite generar archivos Excel y csv. Utilizado para exportar reportes de denuncias.
- **phaza/laravel-postgis:** amplía las funcionalidades de las migraciones de laravel para almacenar formas geográficas en bases de datos postgis. Utilizado para trabajar con las coordenadas de las denuncias enviadas por ciudadanos.
- **spatie/laravel-activitylog:** permite realizar logs del sistema de forma automatizada. Utilizado en múltiples operaciones del sistema para llevar un control de las acciones realizadas sobre los recursos.
- **spatie/laravel-permission:** permite generar un esquema de roles y permisos y asociarlos con los usuarios del sistema. Utilizado para garantizar el acceso a los usuarios a las secciones del sistema que le correspondan.
- **bootstrap:** framework de css que posee clases y componentes predefinidos logrando muy buenos acabados estéticos de forma rápida y escalable.
- **select2:** adiciona funcionalidades a los componentes de html llamados select. Entre las mismas podemos mencionar búsquedas blandas, multi-selección de opciones, etc.
- **fontawesome:** librería de iconos predefinidos de buena calidad que pueden ser importados de forma rápida.
- **sweetalert2:** librería de estilos predefinidos para alertas de navegador web.
- **datatables:** permite añadir funcionalidades a tablas html como búsqueda, ordenamiento, paginación, etc.
- **daterangepicker:** permite seleccionar fechas en calendarios en un campo de entrada de texto.

- **jquery:** librería con funcionalidades de javascript predefinidas. Utilizada debido a su potencialidad y ahorro de tiempo.
- **highcharts:** librería para realizar gráficos con javascript. Permite configurar absolutamente todos los parámetros del mismo, colores, títulos, leyendas, etc.
- **audiojs:** permite utilizar un reproductor de audio en una página web a través de javascript. Utilizada para escuchar audios de denuncias.
- **moment:** librería de javascript utilizada para dar formato a fechas.
- **leaflet:** librería que permite generar mapas personalizados utilizando javascript.
- **leaflet-easy-print:** permite imprimir un mapa generado por el usuario.
- **leaflet-markercluster:** agrupa marcadores sobre un mapa basado en el zoom del mismo, mostrando el número de puntos agrupados y pintando los grupos en base a la cantidad de los mismos.

8.7 Verificación y corrección de errores de los productos de la solución

En los proyectos de desarrollo de software, es de vital importancia que el producto final cumpla con los requerimientos obtenidos. Sin embargo, eso no es todo lo que se espera de una solución, sino que además deberían contemplar casos donde se puedan presentar errores (generados por el usuario o no tenidos en cuenta en la programación), es por ello que decimos que nuestro producto debe estar bien **validado** y **verificado**.

Existen muchos tipos de pruebas que pueden ser realizadas a un software para acercarse a un resultado aceptable en cuanto a validación y verificación, por ejemplo:

- **Pruebas funcionales:** Las pruebas funcionales verifican que cada función de la aplicación de software funciona de acuerdo con la especificación de requisitos. Muestran “Lo que hace el sistema”. El objetivo de esta prueba es verificar si el sistema es funcionalmente perfecto [22].
- **Pruebas no funcionales:** Se ocupan de los requisitos no funcionales. Las pruebas no funcionales ayudan a estimar la preparación de un sistema de acuerdo con los diferentes criterios que no están cubiertos por las pruebas funcionales. A diferencia de las pruebas funcionales, muestra “Qué bueno funciona el sistema” [22].

- **Pruebas estructurales:** verifica la implementación del programa o código a través de pruebas de la estructura del sistema de software o sus componentes. El probador se concentra en el trabajo del software durante las pruebas estructurales [22].
- **Pruebas relacionadas con el cambio:** se proporciona para garantizar que los errores erradicados previamente se hayan solucionado y para detectar errores que pueden haber aparecido accidentalmente en una nueva versión [22].

Con el fin de no extender demasiado este informe final y brindarle al lector la posibilidad de que visualice resultados de manera coloquial, es que decidimos realizar pruebas respecto a las funcionalidades principales de nuestra solución.

Hoy en día existen muchas herramientas que permiten realizar pruebas funcionales de manera automatizada, pero las mismas generalmente involucran sintaxis acorde al lenguaje de programación utilizado, es por ello que decidimos utilizar un proceso manual personalizado [23] para que el lector pueda comprender fácilmente cómo se plantearon las pruebas funcionales de los productos desarrollados.

8.7.1 Pruebas funcionales de módulos del sistema web

Para realizar estas pruebas, nos enfocamos en los módulos del sistema web que definimos anteriormente. Para cada una de las funcionalidades de los mismos, generamos dos (o tres, según corresponda) tablas cuyos atributos nos permiten recabar los datos necesarios a utilizar durante el proceso.

A modo de ejemplo, se utilizará a continuación el módulo de ingreso al sistema web, con sus funcionalidades correspondientes.

Para iniciar el proceso de pruebas, comenzamos generando una tabla que denominamos “cabecera de caso de prueba”, en la cual se colocan los datos que permiten identificar claramente el módulo del producto que contiene la funcionalidad/es a probar y las personas involucradas en el proceso.

ID caso de prueba	SISWEB-CP1
Módulo	Ingreso al sistema
Descripción	Comprobar que el usuario pueda ingresar al sistema
Preparado por	Sonia

Probado por	Sonia
--------------------	-------

Tabla 82. Cabecera de caso de prueba.

Donde:

- **ID caso de prueba:** identificador único que distingue el producto y la funcionalidad del mismo que se está probando de manera unívoca.
- **Módulo:** sección del producto a la que pertenecen las funcionalidades que se va a probar.
- **Descripción:** breve reseña del módulo al cual pertenecen las funcionalidades.
- **Preparado por:** nombre de quien preparó las pruebas.
- **Probado por:** nombre de quien realizó las pruebas.

Luego, continuamos con la tabla “funcionalidad de módulo”, la cual permite detallar todos los pasos necesarios y los resultados esperados y obtenidos para las pruebas realizadas.

Funcionalidad	Ingresar al sistema			
Pasos previos	1. Dar de alta un usuario para realizar la prueba 2. Tomar nota de sus credenciales de ingreso			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar si el sitio funciona en el dominio especificado	Ingresar la url: https://deciper.per.gov.ar	El sitio debería abrir	Pasa	
Comprobar cómo reacciona el sistema con los distintos casos de prueba	1. Para cada fila de la tabla " Conjunto de casos de prueba ", seleccionar los valores indicados en la misma 2. Presionar el botón "ingresar"	Resultado esperado de acuerdo a la fila que se esté probando de la tabla "Conjunto de casos de prueba"	Resultado actual de acuerdo a la fila que se esté probando de la tabla "Conjunto de casos de prueba"	

Tabla 83. Datos necesarios para probar una funcionalidad.

Donde:

- **Funcionalidad:** nombre de la funcionalidad a probar.
- **Pasos previos:** listado de pasos que deben llevarse adelante para generar el entorno donde se ejecutarán las pruebas.
- **Descripción:** breve descripción de la prueba.

- **Pasos:** pasos necesarios para comenzar la prueba.
- **Resultado esperado:** resultado que se espera que arroje la prueba.
- **Resultado actual:** resultado que arrojó la prueba.
- **Notas:** comentarios adicionales.

Con el fin de no agregar más filas en la tabla “funcionalidad”, se decidió separar las pruebas de datos en una tabla aparte, de tal manera de facilitar su lectura y escalabilidad. Debido a ello surge la tabla “conjunto de datos de prueba”, la cual solo será completada si es requerido.

Conjunto de datos de prueba				
Nro. conjunto	Valores conjuntos de datos	Comentarios	Resultado esperado	Resultado actual
1	Usuario= HR34495578 Contraseña=PER1 23654	Parámetros correctos	Pasar	Pasa
2	Usuario= hronchi Contraseña=PER1 23654	Parámetro usuario formato inválido.	Fallar	Mensaje error: “Credenciales inválidas”
3	Usuario= HR34495578 Contraseña=12345 6	Parámetro Contraseña incorrecto.	Fallar	Mensaje error: “Credenciales inválidas”
4	Usuario= HR34495578 Contraseña=per	Parámetro Contraseña incorrecto.	Fallar	Mensaje error: “Credenciales inválidas”
5	Usuario= HR34495578 Contraseña=PER1	Parámetro Contraseña incorrecto.	Fallar	Mensaje error: “Credenciales inválidas”
6	Usuario= Contraseña=PER1 23654	Parámetro obligatorio faltante.	Fallar	Mensaje error:” Atención! El campo Usuario es requerido.”
7	Usuario= HR34495578 Contraseña=	Parámetro obligatorio faltante.	Fallar	Mensaje error:” Atención! El campo Contraseña es requerido.”

Tabla 84. Conjunto de datos de prueba necesarios para probar una funcionalidad.

Donde:

- **Nro. conjunto:** identificador numérico único del conjunto de datos de prueba.
- **Valores conjuntos de datos:** nombre y valor de los parámetros con los que se ejecutará la prueba.
- **Comentarios:** indican la razón sobre los valores que toman los parámetros.
- **Resultado esperado:** resultado que se espera que arroje la prueba.
- **Resultado actual:** resultado que arrojó la prueba.

8.7.2 Metodología para pruebas de la API

Para realizar las pruebas a la API, utilizamos el mismo enfoque que para los módulos, pero modificando algunos campos de las tablas generadas, de tal manera que se adapten a la información requerida para el proceso. Además, se decidió dividir las pruebas por ruta de acceso, de tal manera de mejorar el orden y legibilidad de los resultados obtenidos.

Para ejemplificar la metodología propuesta, se procede al análisis de la ruta “dispositivo”, la cual es aquella que permite identificar un dispositivo que quiera ingresar a la aplicación móvil, validando que los datos con los que cuenta sean correctos y retornando el token de acceso correspondiente.

Nuevamente, comenzamos con una tabla de cabecera, la cual puede verse a continuación:

ID caso de prueba	API-CP1
Ruta	/api/v1/dispositivo
Método HTTP	POST
Descripción	Comprobar la funcionalidad que permite obtener un token de sesión al ingresar a la aplicación móvil
Preparado por	Sonia
Probado por	Sonia

Tabla 85. Cabecera de caso de prueba para API.

Donde:

- **ID caso de prueba:** identificador único que distingue el producto y la funcionalidad

del mismo que se está probando de manera unívoca.

- **Ruta:** nombre de la ruta que se va a probar.
- **Método HTTP:** método a través del cual se envían los parámetros al servidor.
- **Descripción:** breve reseña del módulo al cual pertenecen las funcionalidades.
- **Preparado por:** nombre de quien preparó las pruebas.
- **Probado por:** nombre de quien realizó las pruebas.

Luego, continuamos con la tabla “funcionalidad”, la cual permite detallar todos los pasos necesarios y los resultados esperados y obtenidos para las pruebas realizadas. La misma es una versión acotada de la tabla homónima para un caso de prueba de módulos, debido a la naturaleza de la forma de las peticiones.

Funcionalidad	Comprobar que un dispositivo móvil pueda obtener un token de sesión
Pasos previos	<ol style="list-style-type: none"> 1. Utilizar un cliente HTTP para realizar las pruebas 2. Colocar la dirección web de la API seguido del parámetro "ruta" de la tabla anterior 3. Ejecutar llamadas HTTP por cada fila de la tabla "Conjunto de datos de prueba" <p>IMPORTANTE: los valores "APP_KEY", "APP_INSTALLATION_TOKEN", "TOKEN_VALIDO" y "TOKEN_SESION_VALIDO" son de carácter ilustrativo para este documento.</p>

Tabla 86. Datos necesarios para probar una funcionalidad.

Donde:

- **Funcionalidad:** descripción de la funcionalidad a probar en la ruta.
- **Pasos previos:** listado de pasos que deben llevarse adelante para generar el entorno donde se ejecutarán las pruebas.

Por último, se cuenta con la tabla de conjunto de datos de prueba, la cual es muy similar a la utilizada en las pruebas de módulos web, con la salvedad que la columna “Resultado actual” arroja un código HTTP y un mensaje de respuesta, el cual debe ser interpretado por la aplicación móvil e informarle al usuario sobre el resultado de su consulta.

Conjunto de datos de prueba				
Nro. conjunto	Valores conjuntos de datos	Comentarios	Resultado esperado	Resultado actual

1	appKey=APP_KEY appInstallationToken=APP_INSTALLATION_TOKEN g-recaptcha-response=TOKEN_VALIDO	Parámetros de la petición correctos	Pasar	HTTP Status:200 Mensaje: "token:TOKEN_SESSION_VALIDO"
2	appKey=APP_KEY appInstallationToken=APP_INSTALLATION_TOKEN g-recaptcha-response=	Parámetro obligatorio faltante.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"
3	appKey=APP_KEY appInstallationToken= g-recaptcha-response=TOKEN_VALIDO	Parámetro obligatorio faltante.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"
4	appKey= appInstallationToken=APP_INSTALLATION_TOKEN g-recaptcha-response=TOKEN_VALIDO	Parámetro obligatorio faltante.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"
5	appKey= appInstallationToken= g-recaptcha-response=	Parámetro obligatorio faltante.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"
6	appKey=APP_KEY appInstallationToken=APP_INSTALLATION_TOKEN g-recaptcha-response=1	Parámetro obligatorio incorrecto.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"

Tabla 87. Conjunto de datos de prueba necesarios para probar una ruta de la API.

Donde:

- **Nro. conjunto:** identificador numérico único del conjunto de datos de prueba.
- **Valores conjuntos de datos:** nombre y valor de los parámetros con los que se ejecutará la prueba.

- **Comentarios:** indican la razón sobre los valores que toman los parámetros.
- **Resultado esperado:** resultado que se espera que arroje la prueba.
- **Resultado actual:** resultado que arrojó la prueba. Debe contar con un código HTTP y un mensaje coloquial para notificar al usuario de ser necesario.

Para concluir la sección, cabe destacar que el resto de las pruebas relacionadas a todos los productos desarrollados, pueden ser consultadas en el Anexo 11.3.

8.8 Validación de la solución con el cliente

Una vez superada con éxito la fase de pruebas, debíamos proceder a solicitar una reunión con los funcionarios policiales para mostrar la solución implementada. Sin embargo, debido a la situación de emergencia generada por la pandemia global, decidimos aplazar la misma ya que consideramos que la Fuerza tiene otras prioridades más urgentes que atender.

9. Funcionalidades principales

En este capítulo, mostraremos las funciones principales del sistema web y la aplicación móvil, detallando los aspectos claves y la lógica aplicada a procesos importantes.

9.1 Aplicación móvil

9.1.1 Realizar denuncia

Para realizar una denuncia desde la aplicación móvil se ingresa a esta y se va a la opción de denuncia, aquí ingresan los datos en una seguidilla de pasos: 1- ingresar fecha y hora, 2- seleccionar un tipo de delito, 3- dar la ubicación en un mapa del hecho, 4- describir en formato de texto y/o mensaje de voz el hecho. Luego se envía la denuncia a través de la API de denuncia con la correspondiente autenticación, para guardarla en la base de datos, y devolverá un mensaje de si se realizó con éxito o no (mensaje de error) según sea necesario.

Las interfaces de pasos se hicieron para que sean simples y fáciles de entender. Por ejemplo, el puntero del mapa se mantiene centrado en la pantalla para que el usuario navegue con comodidad (como en aplicaciones de pedidos de taxi).



Imagen 5. Cargar una denuncia en la aplicación móvil. Paso 1.

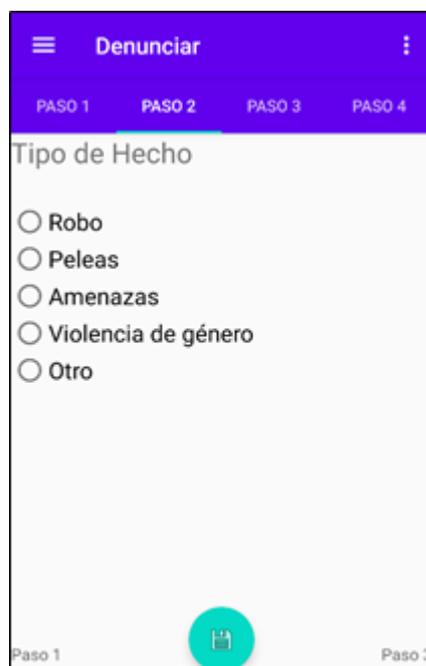


Imagen 6. Cargar una denuncia en la aplicación móvil. Paso 2.

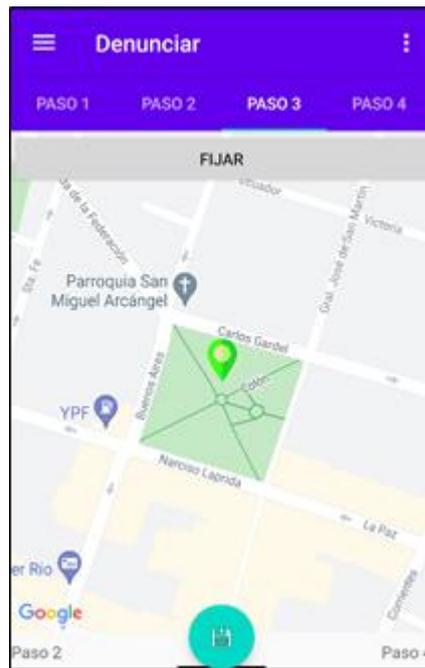


Imagen 7. Cargar una denuncia en la aplicación móvil. Paso 3.

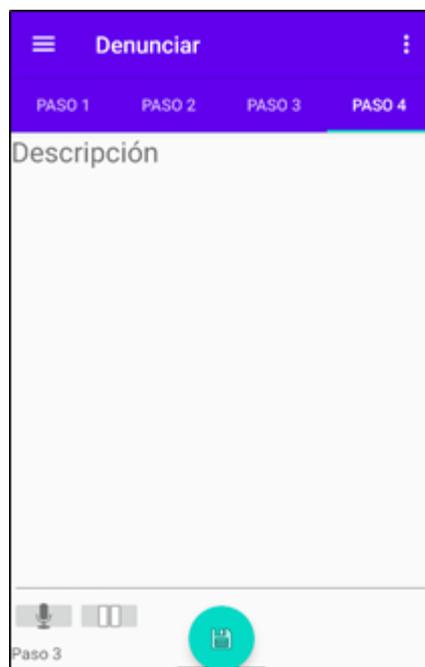


Imagen 8. Cargar una denuncia en la aplicación móvil. Paso 4.

9.1.2 Mapa de dependencias policiales

Esta funcionalidad permite visualizar el área de acción de comisarías de la ciudad de Paraná a través de un mapa (el objetivo es abarcar toda la provincia, pero por cuestiones de tiempo se decidió comenzar con la capital). De esta forma, si un ciudadano fue víctima de un delito, puede ver a qué comisaría le corresponde realizar la denuncia correspondiente con presionar sobre el polígono en el cual se dibujó el marcador en el mapa.

Además, la ventana emergente del polígono, permite realizar una llamada telefónica automáticamente a la comisaría pertinente.

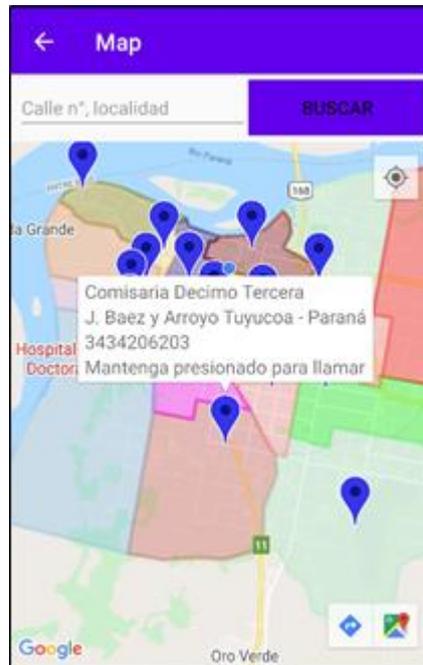


Imagen 9. Mapa de dependencias policiales de la ciudad de Paraná.

9.1.3 Listado de dependencias policiales

Esta funcionalidad permite buscar dependencias policiales provinciales y visualizar su información. Además, permite realizar un llamado telefónico a la misma con pulsar sobre un elemento de la lista obtenida como resultado de la búsqueda realizada.

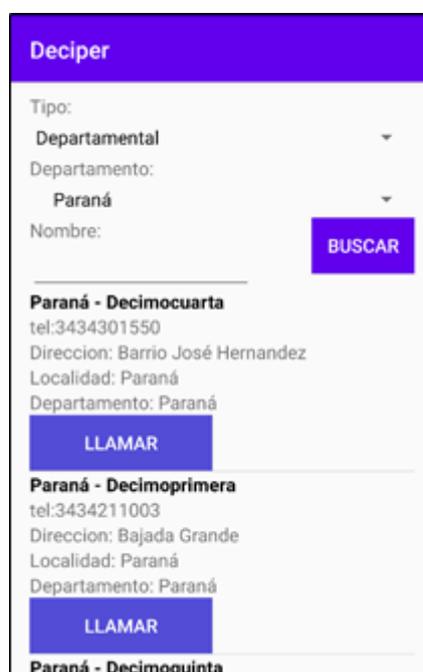


Imagen 10. Listado de dependencias policiales de la ciudad de Paraná.

9.1.4 Lógica para la identificación de dispositivos

Uno de los requerimientos principales de la solución implementada fue la de mantener en todo momento la protección de datos personales y el anonimato de las denuncias realizadas. Sin embargo, al tratarse de una aplicación abierta al público, debimos tomar recaudos para evitar que datos falsos o erróneos, cargados intencionalmente o no, sean tomados en cuenta para la generación de información.

Para reducir este comportamiento indebido, se optó por la identificación del dispositivo denunciante, de manera de brindar la opción a los auditores policiales de banear dispositivos manipulados por ciudadanos malintencionados (política explicada en detalle en la sección 9.2.7).

El mecanismo que pensamos se basa en la utilización de dos identificadores, el primero es el ANDROID ID [24], el cual es único para cada dispositivo y solo cambia ante un restablecimiento de fábrica del dispositivo. El segundo es la generación de un ID único global (GUID) [25], el cual se encuentra almacenado encriptado en el código fuente de la aplicación y en un archivo de configuración del servidor.

Además de los identificadores, se utilizó un código proporcionado por Google Recaptcha [26], el cual garantiza que una petición al servidor es realizada por un ser humano y no un robot. Este mecanismo funciona proporcionando una serie de imágenes y criterios de selección que son fácilmente entendibles para un ser humano, pero no para una máquina. Además, como la validación de los datos implica una clave secreta en el servidor, los estándares de acceso a los recursos protegidos son muy elevados.

Cuando el ciudadano ingresa a la aplicación por primera vez debe completar un código captcha, una vez que lo hace, se realiza una petición a la API con los identificadores y el código de respuesta proporcionado por Google. En el servidor se valida que existan los tres parámetros en la petición, que el GUID sea idéntico al que se encuentra en el archivo de configuración y que el código captcha sea válido (esta evaluación la realizan los servidores de Google). Si los criterios de validación se cumplen, se busca el ANDROID ID suministrado en la tabla de dispositivos de la base de datos.

Si el dispositivo ingresa por primera vez, su ANDROID ID no se encuentra registrado por lo que se procede a darlo de alta en la base de datos, loguea dicho dispositivo en el sistema y retorna como respuesta un token de sesión válido por sesenta minutos.

El token retornado por la API es almacenado dentro del código fuente y deberá ser adjuntado en la cabecera de cada petición posterior que realice la aplicación al servidor.

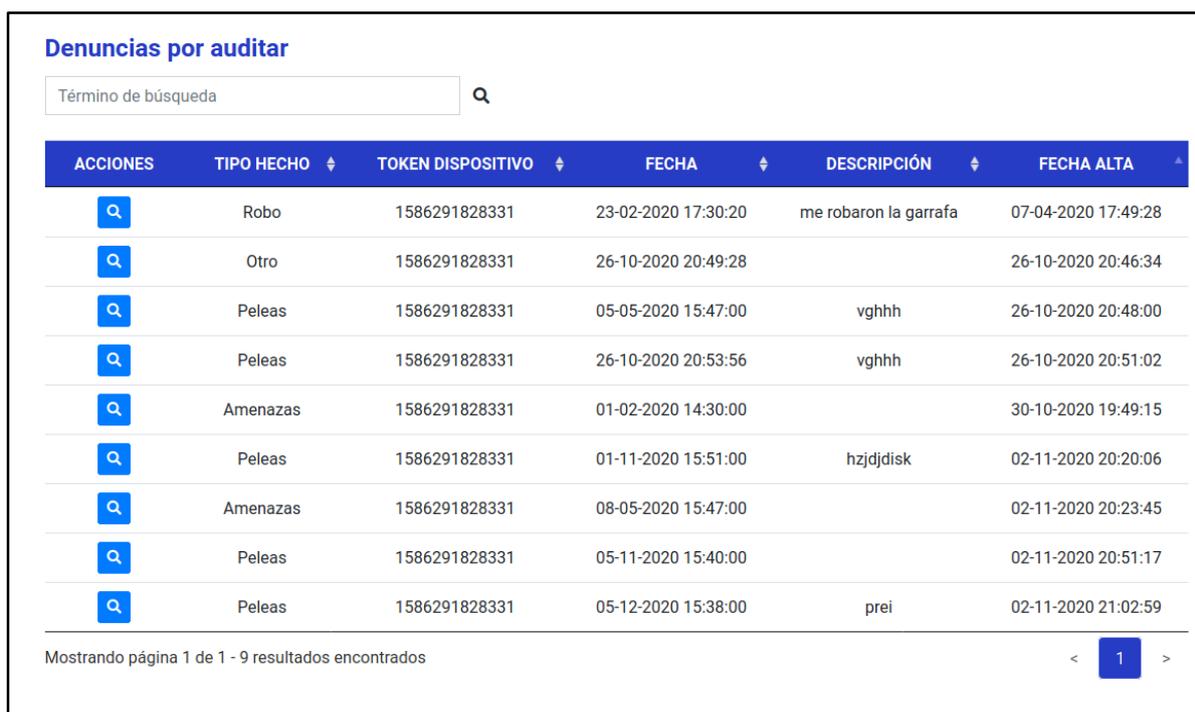
Si el usuario reinstala la aplicación, el ANDROID ID no cambia, por lo que el dispositivo no vuelve a darse de alta en la base de datos, sino que simplemente el servidor lo loguea (de pasar las validaciones correctamente) y retorna el token de sesión. De esta manera, el dispositivo conserva su historia (strikes, baneos, fecha última denuncia, etc.) dentro del sistema.

9.2 Sistema web

9.2.1 Auditar denuncias

Esta funcionalidad permite, a un auditor o analista policial, la posibilidad de aceptar o desestimar una denuncia emitida por un ciudadano.

Para ello, el sistema lista las denuncias por auditar ordenándose de mayor fecha de realización a menor (de esta manera, se evitan que las denuncias queden sin auditar).



Denuncias por auditar

Término de búsqueda

ACCIONES	TIPO HECHO	TOKEN DISPOSITIVO	FECHA	DESCRIPCIÓN	FECHA ALTA
<input type="button" value="Q"/>	Robo	1586291828331	23-02-2020 17:30:20	me robaron la garrafa	07-04-2020 17:49:28
<input type="button" value="Q"/>	Otro	1586291828331	26-10-2020 20:49:28		26-10-2020 20:46:34
<input type="button" value="Q"/>	Peleas	1586291828331	05-05-2020 15:47:00	vghhh	26-10-2020 20:48:00
<input type="button" value="Q"/>	Peleas	1586291828331	26-10-2020 20:53:56	vghhh	26-10-2020 20:51:02
<input type="button" value="Q"/>	Amenazas	1586291828331	01-02-2020 14:30:00		30-10-2020 19:49:15
<input type="button" value="Q"/>	Peleas	1586291828331	01-11-2020 15:51:00	hzjdjdisk	02-11-2020 20:20:06
<input type="button" value="Q"/>	Amenazas	1586291828331	08-05-2020 15:47:00		02-11-2020 20:23:45
<input type="button" value="Q"/>	Peleas	1586291828331	05-11-2020 15:40:00		02-11-2020 20:51:17
<input type="button" value="Q"/>	Peleas	1586291828331	05-12-2020 15:38:00	prei	02-11-2020 21:02:59

Mostrando página 1 de 1 - 9 resultados encontrados < 1 >

Imagen 11. Listado de denuncias por auditar.

El usuario selecciona la denuncia con la que trabajará y el sistema lo dirige a una página donde se desplegarán los datos asociados a la misma. Además, allí mismo puede recategorizar la denuncia, es decir, asignarle una o más categorías o subcategorías delictivas contempladas, como así también observaciones si así lo considera pertinente.

Por último, cuenta con tres botones para aceptar o desestimar la denuncia y, si corresponde, dar un strike al dispositivo denunciante.

Denuncia #1

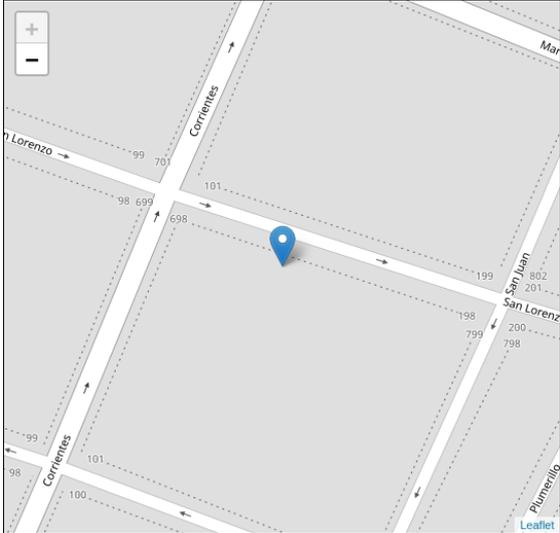
Fecha hecho: 23-02-2020 17:30:20 **Fecha carga:** 07-04-2020 17:46:36

Tipo hecho: Robo **Dispositivo:** 1586291828331

Descripción:
me robaron la garrafa

Audio:

▶ 00:00 / 01:31



Revisión

Observaciones 28/2500 caracteres

Observaciones de la revisión

Categoría: Seleccione una categ... AGREGAR

Subcategoría: Seleccione una subcateg... AGREGAR

ACCIONES	CATEGORÍA
X	[5] - Contra la propiedad

ACCIONES	SUBCATEGORÍA
X	[5] - Robos
X	[5] - Tentativa de robo

ACEPTAR DENUNCIA

DESESTIMAR DENUNCIA

DAR STRIKE A DISPOSITIVO

Imagen 12. Pantalla para auditar denuncias.

9.2.2 Analizar denuncias

Esta funcionalidad es similar a la anterior, auditoría de denuncias, sin embargo, permite visualizar denuncias aceptadas o desestimadas con el fin de detectar errores en el proceso de auditoría realizado (desestimar una denuncia correcta, aceptar denuncias con datos erróneos, etc.) y corregir el estado de la denuncia, sus categorías y/o subcategorías delictuales y observaciones.

Dicha tarea es realizada exclusivamente por los analistas policiales, ya que son los responsables de la generación de información basada en las denuncias que se encuentran en el sistema.

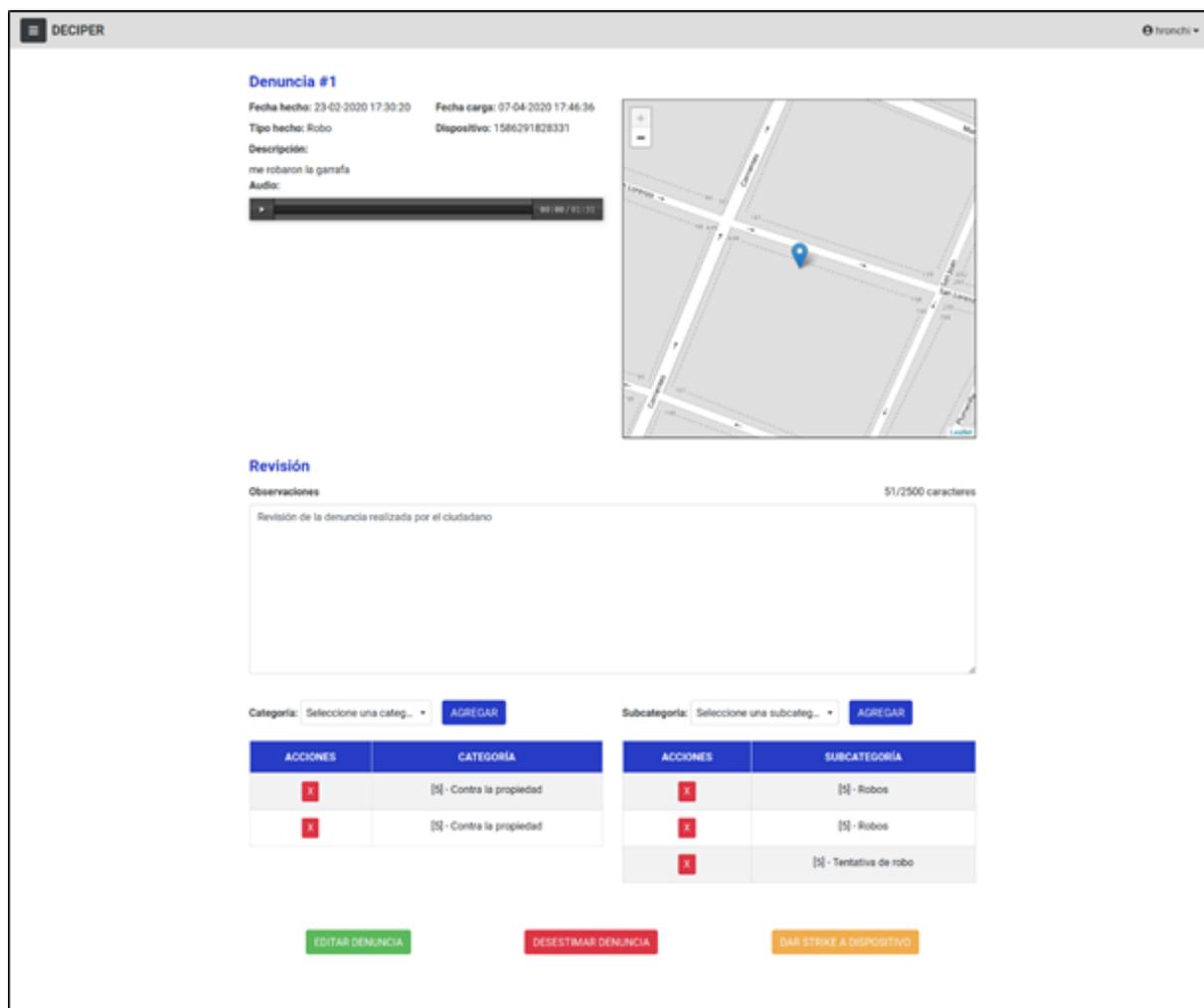


Imagen 13. Pantalla para analizar denuncias.

9.2.3 Reportes

La funcionalidad reportes permite buscar denuncias aceptadas a través de filtros como: rango de fechas, categoría de delito y subcategoría de delito. El listado generado como resultado de

la búsqueda puede ser exportado a formato Excel o CSV con el fin de poder utilizar los datos para realizar gráficos o informes personalizados. Además, esta característica puede utilizarse para ofrecer reportes semanales o mensuales a los encargados de la toma de decisiones. Cabe destacar que esta funcionalidad es utilizada por los analistas policiales solamente.

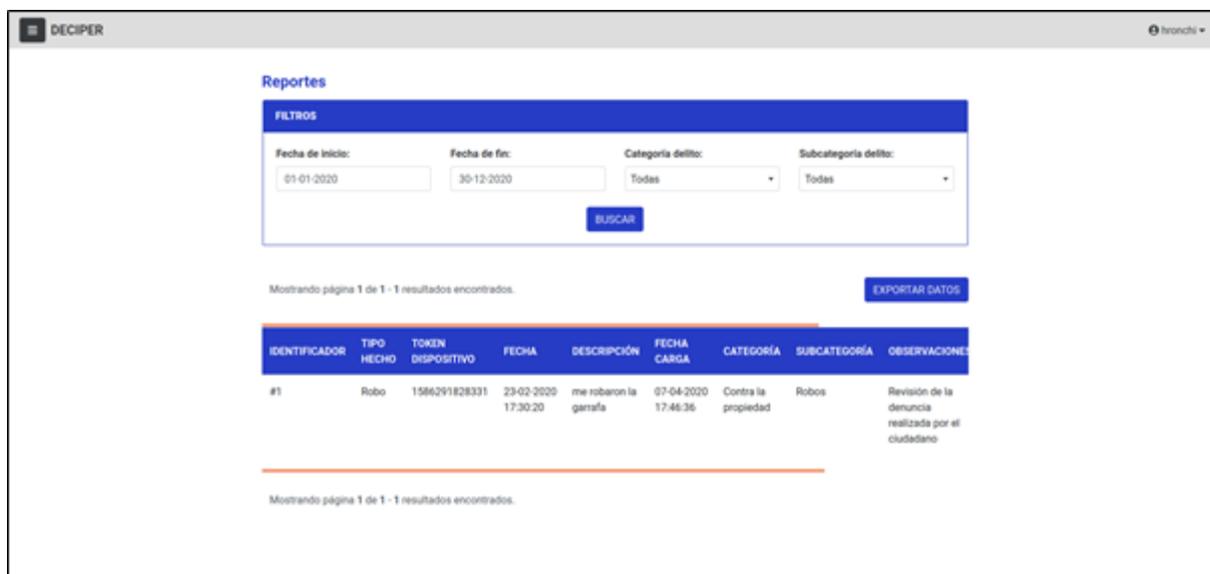


Imagen 14. Pantalla de reportes.

9.2.4 Estadísticas

La funcionalidad estadística, consiste en la generación de información que busca contestar las preguntas frecuentes de interés para facilitar la toma de decisiones de forma rápida y efectiva.

Es por ello que se decidió utilizar gráficos con el fin de agilizar la percepción de la información. Los mismos son divididos en dos categorías dentro del sistema: gráficos con intervalos de tiempo variable y gráficos anuales.

Los gráficos con intervalos de tiempo variable permiten filtrar los datos de acuerdo a un rango de fechas que especifica el usuario, de esa manera, permite contestar preguntas tales como: ¿Cuántas denuncias hubo la última semana/último mes?, ¿Cuántos dispositivos nuevos se registraron en la última quincena?, etc.

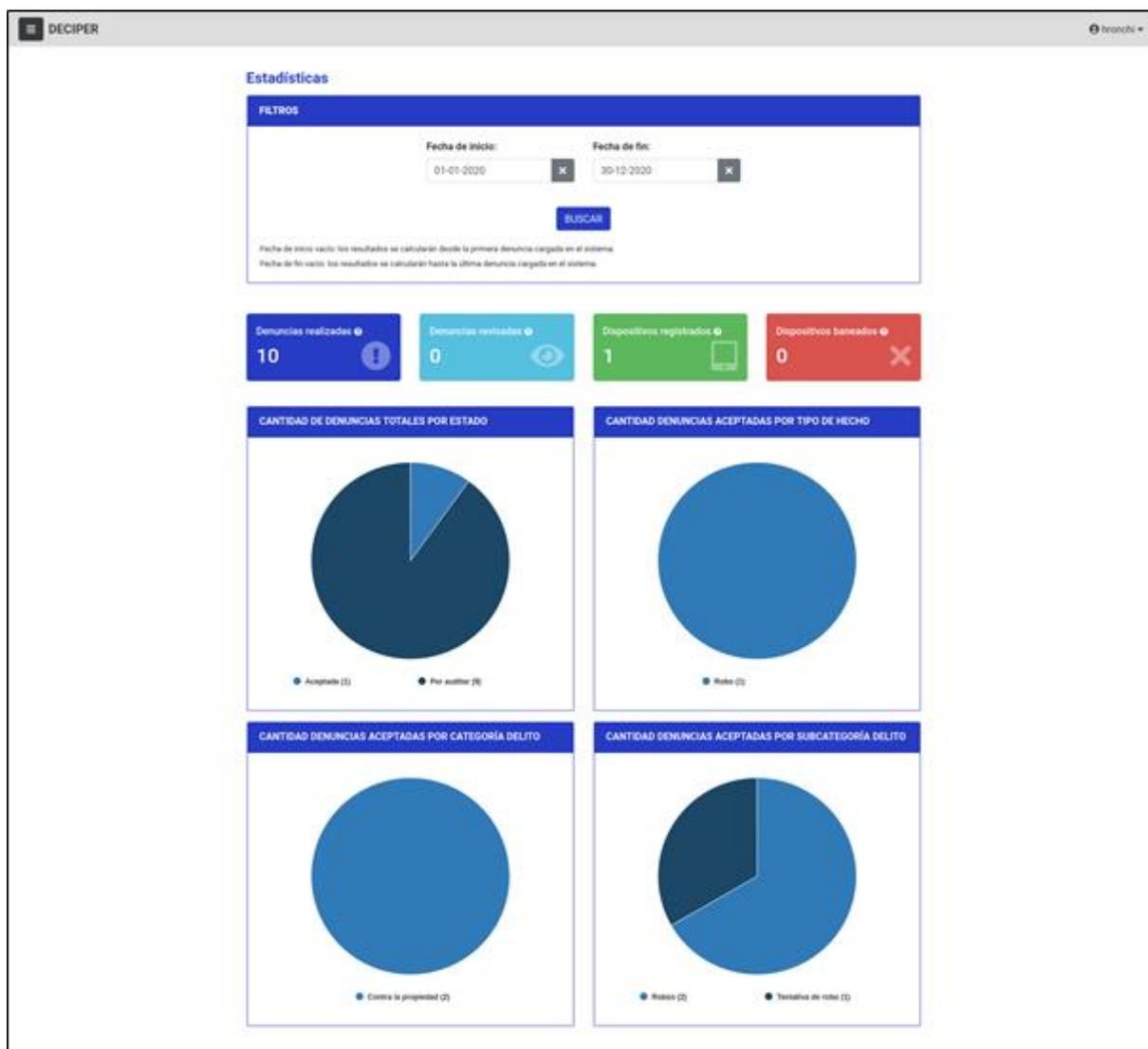


Imagen 15. Pantalla de estadísticas con intervalo de tiempo variable.

Por otro lado, los gráficos anuales, permiten observar la variación interanual de la información. En lugar de un rango de fechas solamente permite seleccionar un año en específico, de manera de poder contrastar resultados y comparar información con el año en curso.

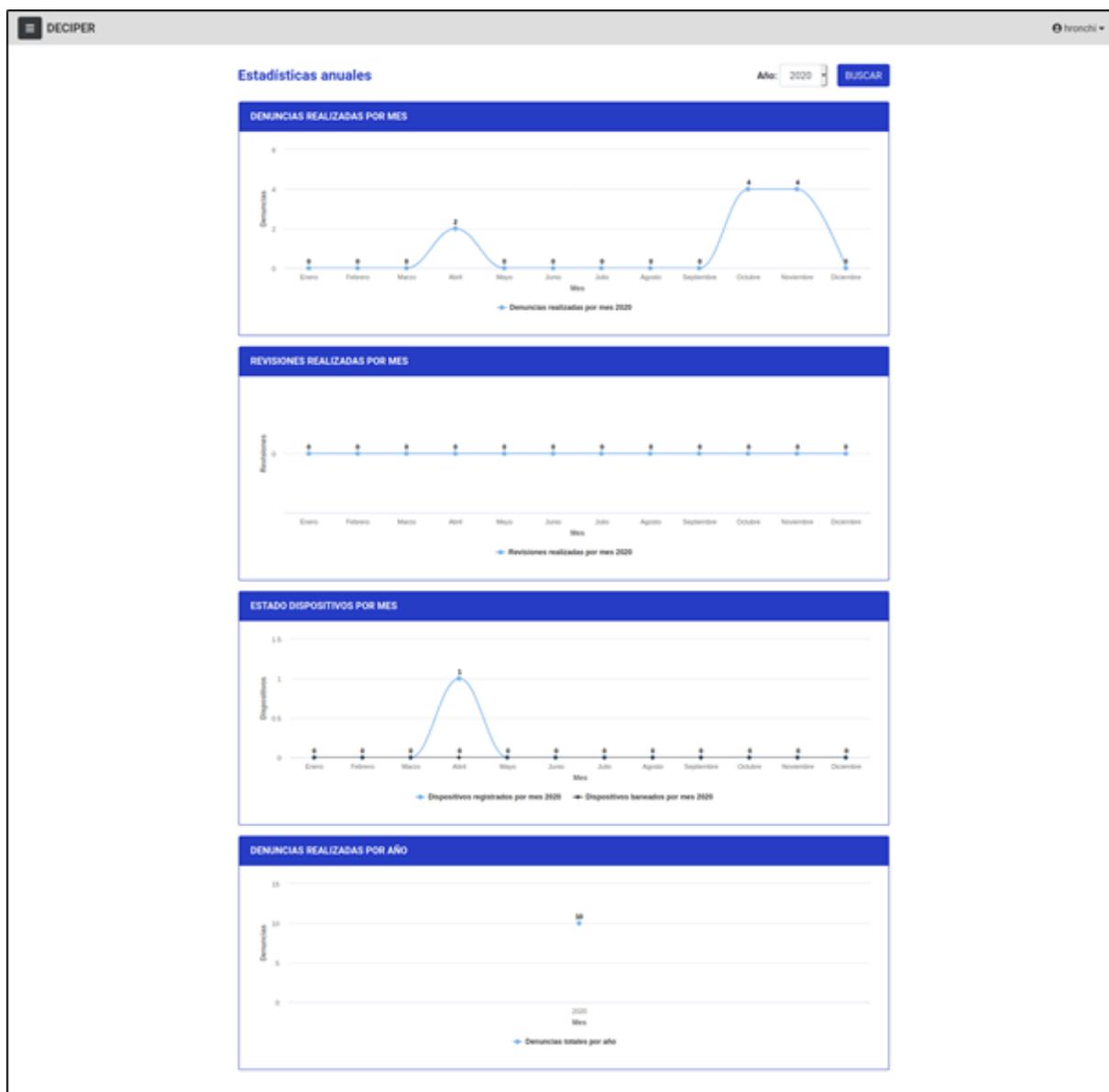


Imagen 16. Pantalla de estadísticas anuales.

9.2.5 Mapas

Esta funcionalidad permite visualizar las denuncias aceptadas a través de su localización en un mapa de la provincia de Entre Ríos. De esta manera se pueden identificar zonas con alta proliferación de delitos. Cabe destacar que los puntos cercanos se auto agrupan, cambiando de color verde, amarillo o rojo, cuanto mayor sea la cantidad de los mismos. Por otro lado, cuando se realiza acercamiento sobre una zona en particular, la visualización se autoajusta solamente con los puntos visibles, permitiendo visualizar una zona específica con amplio nivel de detalle.

Cabe destacar, además, que los delitos utilizados para la generación del mapa pueden ser filtrados por rango de fechas, categoría de delito y subcategoría de delito, lo que permite obtener información detallada de utilidad.

Esta funcionalidad solo puede ser utilizada por los analistas de la Fuerza Policial.

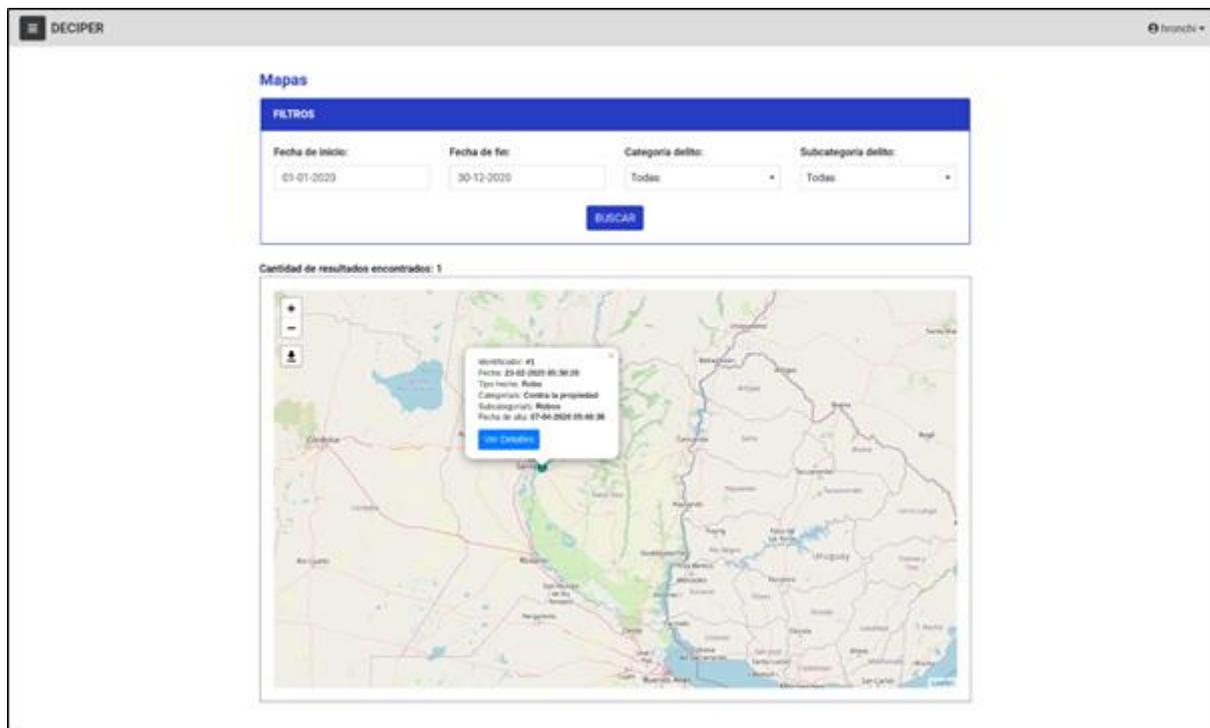
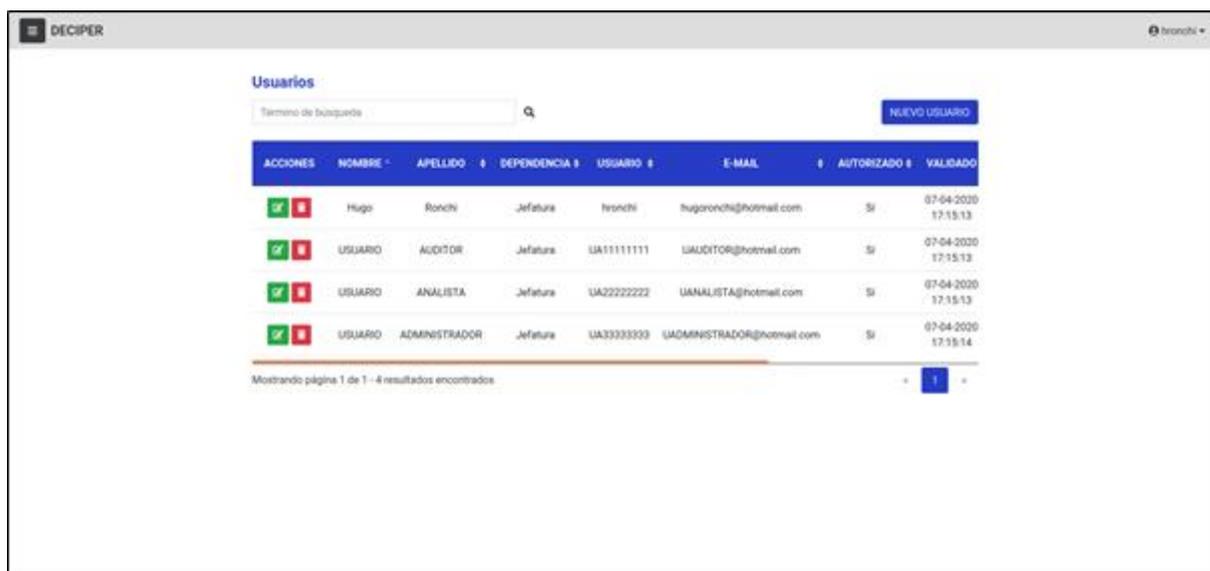


Imagen 17. Pantalla de mapa de denuncias.

9.2.6 Gestión

Esta funcionalidad agrupa aquellas operaciones relacionadas con los recursos usuarios, permisos y roles del sistema. Las mismas involucran dar de alta, editar o eliminar cada uno de los recursos mencionados previamente.

Cada procedimiento comienza con un listado de los recursos actuales y desde allí se puede crear o seleccionar con cuál de ellos operar.



ACCIONES	NOMBRE	APELLIDO	DEPENDENCIA	USUARIO	E-MAIL	AUTORIZADO	VALIDADO
 	Hugo	Ronchi	Jefatura	hronchi	hugoronchi@hotmail.com	SI	07-04-2020 17:15:13
 	USUARIO	AUDITOR	Jefatura	UA11111111	UAUDITOR@hotmail.com	SI	07-04-2020 17:15:13
 	USUARIO	ANALISTA	Jefatura	UA22222222	UANALISTA@hotmail.com	SI	07-04-2020 17:15:13
 	USUARIO	ADMINISTRADOR	Jefatura	UA33333333	UADMINISTRADOR@hotmail.com	SI	07-04-2020 17:15:14

Imagen 18. Pantalla de gestión de usuarios.

El proceso de creación de usuarios se rige por un protocolo particular, en el cual el sistema genera automáticamente un nombre y una contraseña para la cuenta, quitando la posibilidad de errores de tipeo por parte del administrador.

El nombre de usuario se compone de la primera letra del primer nombre de pila, la primera letra del apellido y el documento nacional de identidad completo, de esta manera se garantiza la unicidad y la facilidad de recordar el nombre proporcionado. Cabe destacar que este es el estándar adoptado para los demás sistemas que utiliza la Fuerza Policial, por lo que el usuario está acostumbrado a utilizar la cadena de texto generada.

La contraseña generada por el sistema, se compone de la cadena “PER” (Policía de Entre Ríos) y se concatenan los minutos, segundos y la hora (en ese orden) actuales, de tal manera que pueda ser fácil de recordar debido a su corta longitud.

Una vez finalizada la operación en el sistema, se notifica al nuevo usuario sus credenciales de acceso de forma física y confidencial (atendiendo a los protocolos de seguridad dispuestos para todos los sistemas de la Fuerza Policial) y, por último, el sistema envía automáticamente un email con un link de acceso que expira en 24 horas y sirve para que el usuario pueda colocar una contraseña de su elección.

Un usuario también puede solicitar el restablecimiento de contraseña autogestionado por el mismo vía email, por lo que es importante que cada dirección de correo sea única en la base de datos, lo cual es validado automáticamente por el sistema.

9.2.7 Política de baneo de dispositivos

Cuando una denuncia contiene datos que claramente no corresponden (audios en broma, amenazas o insultos en su descripción, etc.) Los auditores y analistas tienen la posibilidad de sancionar al dispositivo denunciante mediante un strike.

Un strike es una amonestación interna y se asocia al dispositivo a través de su Android Id, de esta manera, cuando se perciben tres strikes, el mismo es baneado permanentemente del sistema.

Cuando un dispositivo es baneado, todas las denuncias que realizó el mismo (y todavía no se auditaron) se desestiman automáticamente, como así también todas las denuncias futuras que realice el mismo. Estas últimas no se escriben siquiera en la base de datos, para evitar usar almacenamiento en datos fraudulentos.

Además, al ciudadano se le seguirá notificando que su denuncia fue aceptada correctamente (a pesar de estar baneado), de esta manera, seguirá enviando datos falsos inútilmente pensando que está logrando su cometido y no será una amenaza para el sistema.

En un futuro, se podría sugerir identificar a un ciudadano que cometa esta infracción, con el fin de sancionar jurídicamente por sus actos, a través de peticiones especiales que realice la aplicación para dispositivos baneados.

10. Conclusiones

Como se ha podido apreciar en capítulos previos del presente informe, el desarrollo de este Proyecto Final de Carrera no fue nada fácil. Hemos tenido retrasos con los tiempos estimados, grandes cambios a nivel planificación, e incluso, tenido que lidiar con una pandemia a nivel global.

En el aspecto técnico, supimos manejar los cambios propuestos en la planificación, adaptándonos a nuevos roles y tareas a llevar adelante por cada uno de nosotros. A su vez, aprendimos a adaptar las herramientas con las que contamos de acuerdo a nuestras necesidades, como fue, por ejemplo, realizar un “sprint 0” con tareas que no están asociadas a historias de usuario, con el fin de poder estimar y organizar con mayor detalle el proyecto.

En un futuro, si el prototipo de solución realizado puede convertirse en un producto funcional real para el ciudadano, nos gustaría mejorar aspectos relativos a la calidad de código, testing automatizado de funcionalidades y seguridad de la información, áreas sobre las cuales seguimos ganando experiencia día a día en nuestros respectivos trabajos.

Queremos mencionar, además, que disfrutamos mucho el trabajar codo a codo y día a día haciendo lo que nos gusta en un proyecto real y poder llevarlo adelante de principio a fin. La gestión de proyectos es una temática apasionante y muy compleja, pero a su vez es muy gratificante construir un producto con buenas prácticas y verlo terminado.

Por último, y a nivel personal, queremos agradecer a todas las personas que han colaborado con nosotros y que nos han formado en el transcurso de la carrera, ya que son una parte muy importante en este logro que estamos alcanzando.

11. Anexos

11.1 Anexo 1: Entrevistas

11.1.1 Entrevista stakeholder

1- ¿Cómo se lleva a cabo el proceso de toma de denuncia por parte de la Fuerza? ¿Cuáles son los datos obligatorios en la misma?

Hay dos aspectos en este proceso, por un lado, si el funcionario policial toma conocimiento que se produjo un hecho delictivo, actúa de oficio, es decir, debe dirigirse al lugar del hecho a comprobar la ocurrencia del mismo y luego redactar una comunicación a su superior y al juzgado informando del hecho, en otras palabras, no es necesario que un ciudadano realice una denuncia.

Luego, el oficial debe comunicarse con el área de investigaciones y la de criminalística para recabar información sobre el hecho, la cual se recopila en un expediente que se remite al juzgado.

El mismo procedimiento se lleva a cabo si una persona se presenta en la comisaría a denunciar un hecho, se le toma la denuncia y, mientras tanto, se puede dar aviso a criminalística e investigaciones de manera informal. Luego de tomar la denuncia, el funcionario se dirige obligadamente al lugar del hecho para constatar la ocurrencia del mismo. Se conforma el expediente y se lo envía al juzgado.

Por otro lado, con las llamadas telefónicas que denuncian un hecho (de manera anónima o no), generalmente no se guarda un registro de las mismas, aunque la policía está obligada a investigar igualmente lo expresado en la llamada. Un ejemplo de este caso es un llamado sobre personas merodeando un lugar concreto, los funcionarios se dirigen a dicho lugar para constatar lo que sucede por una cuestión preventiva.

2- A grandes rasgos, ¿qué ocurre luego de que la denuncia es aceptada?

Como se explicó en la pregunta anterior, toda la información recopilada sobre un hecho delictivo conforma un expediente que se envía al juzgado para su tratamiento y resolución.

3- En base a su opinión y experiencia, ¿Cuáles le parecen que son los hechos que menos denuncia el ciudadano?

La mayoría de hechos que no se denuncian son robos que se cometieron por un descuido de la víctima. Por ejemplo, dejó la puerta abierta de su casa y le sustrajeron una o más pertenencias.

Otro tipo de delito que tiene una baja tasa de denuncia son los relacionados a defraudación, comúnmente conocidos como “cuento del tío”. Generalmente son denunciados estos casos cuando involucran a adultos mayores o lo sustraído es significativo para la víctima, pero la cifra es muy baja comparada con la presunta cantidad de hechos cometidos.

Otro tipo de hecho son los robos de bienes difícilmente identificables, como garrafas de gas, cortadoras de césped, etc.

Otro tipo de hecho no denunciado pueden ser los de carácter sexual o contra las mujeres, debido a razones de diversa índole.

El llamado anónimo, si se registra con fines estadísticos, sirve. Si es con fines preventivos, también sirve. Pero no sirve a efectos judiciales, debido a que se necesita que la denuncia sea formal, que se haga por escrito, determinar la identidad del denunciante, etc.

Las llamadas telefónicas ayudan a entender la situación de una zona geográfica en particular, y es muy interesante armar un mapa del delito con las mismas con el fin de encontrar patrones delictivos.

4- ¿Cuán importante es el anonimato de los datos de un denunciante al momento de realizar una denuncia? ¿En qué tipo de delitos lo considera más importante?

Depende el tipo de denuncia, debido a que, en una formal, por escrito, deben estar los datos del denunciante, el policía que toma la denuncia está obligado a identificarlo, mientras que, en una denuncia anónima, no figuran dichos datos, pero no suelen llegar a tomar un curso legal y formal válido, por lo general.

Por eso este tipo de denuncia tiene importancia con fines estadísticos y preventivos. Además, la denuncia anónima es considerada más importante en delitos contra la propiedad y contra las personas.

En materia de narcotráfico, un juzgado nunca autorizaría un allanamiento por un llamado telefónico anónimo. Si puede servir de manera preventiva si la denuncia corresponde a narcomenudeo en un lugar público, por ejemplo. Pero los procedimientos en el área del narcotráfico son muy engorrosos y requieren de muchas etapas previas para llegar a una resolución.

En materia de corrupción policial, es muy difícil llegar a una resolución debido a que tenes que probar lo que se denuncia del funcionario implicado.

5- ¿Qué le parece la idea de destinar recursos humanos a la tarea de auditar los datos denunciados por un ciudadano? ¿Cree que puede ser una solución provisoria hasta el desarrollo de una herramienta que permita automatizar en gran medida este análisis?

Sí, porque el área de estadísticas que está en la Jefatura Central maneja todos los datos relacionados a lo delictual. Nuevamente, los datos obtenidos a través de las denuncias anónimas sirven en materia de prevención, pero también aportarían a la Fuerza en materia de política institucional, es decir, permitirían que un jefe departamental o de operaciones, por ejemplo, pueda realizar un seguimiento sobre el accionar de los efectivos ante hechos ocurridos que fueron previamente reportados anónimamente.

Por otro lado, el tener estadísticas de estas denuncias anónimas es muy útil para comparar con los hechos denunciados formalmente o de oficio, ya que se pueden mejorar las políticas de asignación de recursos en una zona y horarios determinados (por ejemplo, las rondas de patrullaje), como así también el accionar de los funcionarios policiales que debían darle tratamiento a dicha denuncia, permitiendo un control práctico y una evaluación cuantitativa de la eficacia de la Fuerza.

6- Por último, ¿cuenta la Fuerza con algún mecanismo para estimar la cifra de delitos no denunciados? Si lo hubiera, ¿podría describirlo brevemente?

No, no tiene forma de estimarlo. En lo práctico, en las comisarías nunca querés tener más delitos de los que se tienen que, obligadamente, informar. La estadística puede jugar en contra para los jefes de comisaría también, por ejemplo, si en un determinado mes se hicieron muy buenas tareas de prevención y, en ese mismo mes, se acumulan una cantidad de denuncias anónimas, un jefe departamental puede preguntarle el motivo por el cual le aumentaron los delitos, a pesar de toda la prevención oportunamente realizada. Es por ello que un sistema que permita el registro de denuncias anónimas en otro lugar aparte de las comisarías, permite un doble control y asegura la persistencia de las mismas.

11.1.2 Entrevista Ciudadano (20 - 30 años)

1- ¿Cómo calificaría su nivel de confianza en la Fuerza Policial provincial? ¿Cuál es su opinión respecto al accionar policial ante hechos de inseguridad que ha sufrido usted o sus allegados?

Sinceramente, del uno al diez, uno. Porque las veces que he tenido que confiar en la Fuerza por hechos personales no han estado.

Para mí actúa mal, no estuvo para lo que necesitaba, no cumplió con mis expectativas.

2- De los siguientes tipos de delito, ¿por cuales no considera necesario, o prudente, dirigirse a una comisaría a realizar una denuncia policial?

a- Robo de pertenencias en la vía pública

Sí, es necesario, pero de ahí a que hagan o no las cosas es otro tema.

b- Violencia de género contra usted o un allegado

Sí

c- Robo de objetos en su hogar cuando no se encuentra presente.

Se tendría que hacer, pero como por lo general no hay respuesta de parte de ellos, no se hace.

d- Pelea a golpe de puño en la vía pública con un tercero

Depende del grado de violencia, si no hay pruebas no la haría.

e- Discusión con un tercero en la vía pública seguida de amenazas

No, porque no te van a prestar atención.

3- ¿Estaría dispuesto a utilizar un medio alternativo, rápido, seguro y gratuito para notificar delitos sin moverse de su casa y de tal manera contribuir con las estadísticas delictivas?

Sí, estaría muy bueno. Primero, ahorraría el tiempo de ir hasta allá y, segundo, habría la posibilidad de contabilizar los hechos denunciados y que no queden en la nada.

4- ¿Ha utilizado en alguna ocasión herramientas de mapeo virtuales, por ejemplo, Google maps, Bing, etc.? ¿Cuál es su opinión sobre las mismas?

Sí, siempre. Opino que son muy buenas, debido a que no sé las calles. Yo las uso siempre.

5- Por último, le presentamos un pequeño ejercicio. ¿Podría ubicar, en este celular con Google maps, la plaza Primero de Mayo de la ciudad de Paraná, partiendo desde nuestra ubicación actual? ¿Qué le pareció la experiencia del ejercicio realizado?

Como no sé las calles, me costó un poquito, pero creo que podría ubicar un hecho en el mapa.

11.1.3 Entrevista Ciudadano (30 - 40 años)

1- ¿Cómo calificaría su nivel de confianza en la Fuerza Policial provincial? ¿Cuál es su opinión respecto al accionar policial ante hechos de inseguridad que ha sufrido usted o sus allegados?

Baja, poca confianza. Las veces que me han robado o que he sufrido hechos delictivos la policía no intervino de una manera adecuada.

2- De los siguientes tipos de delito, ¿por cuales no considera necesario, o prudente, dirigirse a una comisaría a realizar una denuncia policial?

a- Robo de pertenencias en la vía pública.

Sí.

b- Violencia de género contra usted o un allegado.

Sí.

c- Robo de objetos en su hogar cuando no se encuentra presente.

Sí.

d- Pelea a golpe de puño en la vía pública con un tercero.

Depende de la gravedad del hecho, pero probablemente sí.

e- Discusión con un tercero en la vía pública seguida de amenazas

No, no iría.

3- ¿Estaría dispuesto a utilizar un medio alternativo, rápido, seguro y gratuito para notificar delitos sin moverse de su casa y de tal manera contribuir con las estadísticas delictivas?

Sí. Porque al tener identificadas las áreas conflictivas se les puede dar prioridad y hacer la tarea más eficiente.

4- ¿Ha utilizado en alguna ocasión herramientas de mapeo virtuales, por ejemplo, Google maps, Bing, etc.? ¿Cuál es su opinión sobre las mismas?

Sí. Son muy útiles y prácticas.

5- Por último, le presentamos un pequeño ejercicio. ¿Podría ubicar, en este celular con Google maps, la plaza Primero de Mayo de la ciudad de Paraná, partiendo desde nuestra ubicación actual? ¿Qué le pareció la experiencia del ejercicio realizado?

(El entrevistado realiza sin inconvenientes el ejercicio)

Me pareció bien, lo hago casi siempre cuando tengo que buscar una dirección. No creo que sea difícil, pero quizás sea más simple escribir la dirección en lugar de mover el mapa.

11.1.4 Entrevista Ciudadano (50 - 60 años)

1- ¿Cómo calificaría su nivel de confianza en la Fuerza Policial provincial? ¿Cuál es su opinión respecto al accionar policial ante hechos de inseguridad que ha sufrido usted o sus allegados?

Yo fui criado teniendo confianza en la policía al ciento por ciento, hoy esa confianza no la tengo. Está todo muy desvirtuado y, como en todos lados, hay gente buena y gente mala dentro de la Fuerza. Ahora si me preguntas si creo firmemente en la policía la respuesta es no, todo el tiempo salen casos de agentes que han descubierto con droga o en situaciones ilícitas, ¿cómo vas a confiar? Plenamente al menos, no.

En cuanto al accionar, una vez me quisieron entrar en la casa, se los llamó y vinieron enseguida, hicieron todo lo que podían, pero con resultado cero, solo se llevaron una constancia de que me habían querido entrar al domicilio.

2- De los siguientes tipos de delito, ¿por cuales no considera necesario, o prudente, dirigirse a una comisaría a realizar una denuncia policial?

a- Robo de pertenencias en la vía pública.

No iría, lo considero una pérdida de tiempo.

b- Violencia de género contra usted o un allegado.

Sí.

c- Robo de objetos en su hogar cuando no se encuentra presente.

Sí.

d- Pelea a golpe de puño en la vía pública con un tercero.

Si me lesionan sí. Si veo a dos personas peleando solo llamaría al 911. En este país uno es reacio a denunciar ya que quedas pegado vos, además te dicen que la denuncia es confidencial pero después resulta que el denunciado se termina enterando y eso no debería ser así.

e- Discusión con un tercero en la vía pública seguida de amenazas

No, eso es enojo del momento. Depende del grado de violencia que exista, lo tomaría como algo normal.

3- ¿Estaría dispuesto a utilizar un medio alternativo, rápido, seguro y gratuito para notificar delitos sin moverse de su casa y de tal manera contribuir con las estadísticas delictivas?

Si.

4- ¿Ha utilizado en alguna ocasión herramientas de mapeo virtuales, por ejemplo, Google maps, Bing, etc.? ¿Cuál es su opinión sobre las mismas?

No las he utilizado, pero las conozco.

5- Por último, le presentamos un pequeño ejercicio. ¿Podría ubicar, en este celular con Google maps, la plaza Primero de Mayo de la ciudad de Paraná, partiendo desde nuestra ubicación actual? ¿Qué le pareció la experiencia del ejercicio realizado?

(El entrevistado realiza sin inconvenientes el ejercicio a pesar de no haber utilizado la herramienta previamente)

No fue difícil realizar el ejercicio, tampoco me parece difícil hacerlo para marcar un hecho.

11.2 Anexo 2: Diagrama de Entidad-Relación

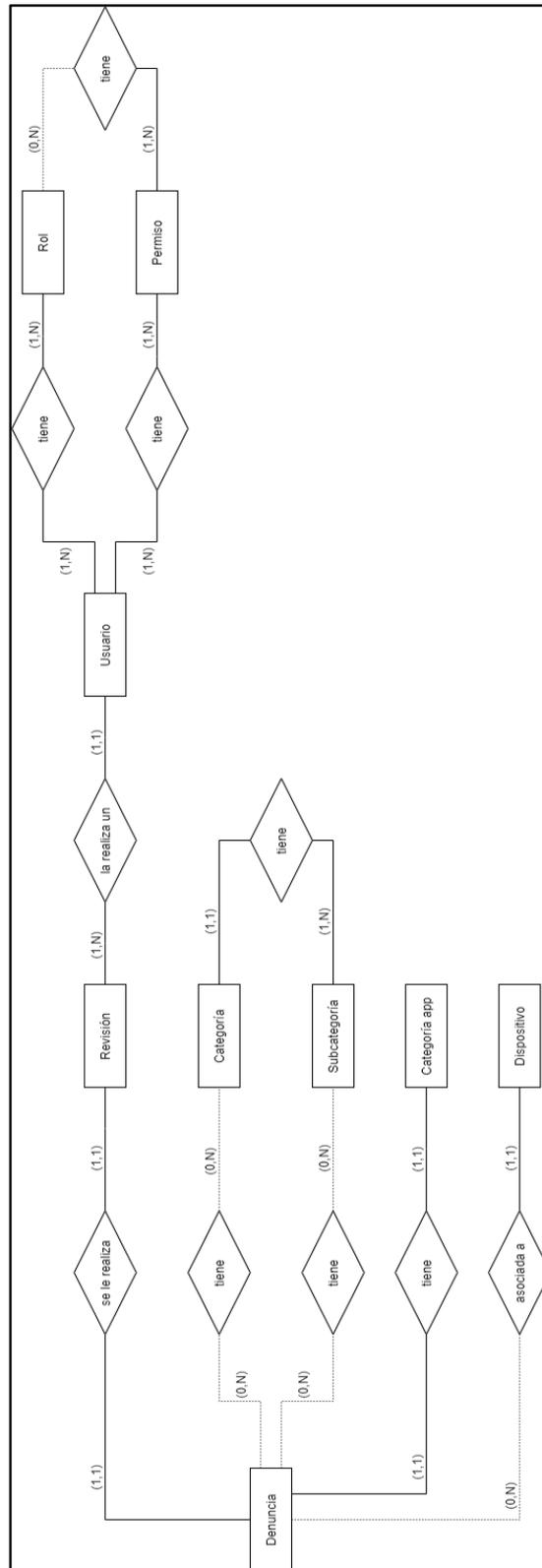


Imagen 19. Diagrama de Entidad Relación del proyecto.

11.3 Anexo 3: Diagrama de Base de Datos

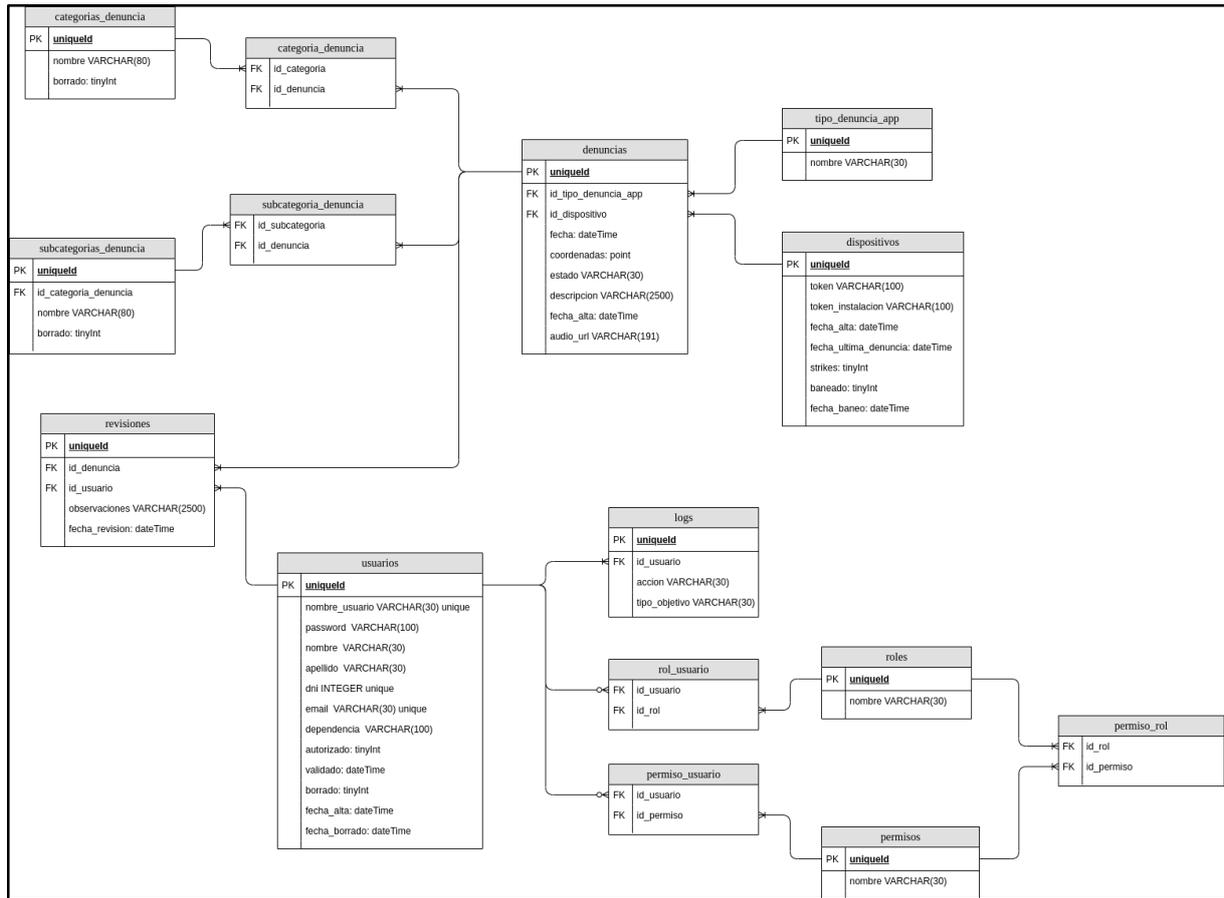


Imagen 20. Diagrama de Base de Datos del proyecto.

11.4 Anexo 4: Pruebas funcionales

11.4.1 Aplicación móvil

11.4.1.1 Módulo alta denuncia

ID caso de prueba	APPMOVIL-CP1
Módulo	Alta denuncia
Descripción	Posicionarse como un ciudadano para realizar un alta de denuncia
Preparado por	Hugo
Probado por	Hugo

Funcionalidad	Alta denuncia			
Pasos previos	1. Haber ingresado a la aplicación móvil			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Ingresar a la sección de denuncias	1. Presionar el botón de denuncias en el menú	Éxito	Éxito	-
Ingresar una denuncia con todos los datos según conjunto 1	1. Ingresar la fecha y luego la hora e ir al siguiente paso 2. Seleccionar de la lista un tipo de delito e ir al siguiente paso 3. Usar la ubicación actual como punto en el mapa e ir al siguiente paso. 4. Escribir los detalles y grabar un mensaje de voz.	Éxito	Éxito	Se presupone que los permisos de ubicación y guardado de archivos multimedia en el dispositivo están dados.

Enviar denuncia	1. Hacer click en el botón de guardado	Éxito	Éxito	El dispositivo debe contar con conexión a internet
-----------------	--	-------	-------	--

Conjunto de datos de prueba				
Nro. conjunto	Valores conjuntos de datos	Comentarios	Resultado esperado	Resultado actual
1	Fecha= 14-03-2020 Hora= 15:51 Tipo= Robo Ubicación= Ubicación actual Descripción= Una persona en moto me robó Mensaje de voz= No	Parámetros correctos	Pasar	Mensaje: "Denuncia enviada"
2	Fecha= Hora= 15:51 Tipo= Robo Ubicación= Ubicación actual Descripción= Una persona en moto me robó Mensaje de voz= No	Parámetro obligatorio fecha faltante.	Fallar	Mensaje error:"¡Atención! El campo Fecha es requerido"
3	Fecha= 14-03-2020 Hora= 15:51 Tipo= Ubicación= Ubicación actual Descripción= Una persona en moto me robó Mensaje de voz= No	Parámetro obligatorio Tipo faltante.	Fallar	Mensaje error:"¡Atención! El campo Tipo es requerido"

4	Fecha= 14-03-2020 Hora= 15:51 Tipo= Robo Ubicación= Descripción= Una persona en moto me robó Mensaje de voz= No	Parámetro obligatorio Ubicación faltante.	Fallar	Mensaje error:"¡Atención! El campo Ubicación es requerido"
5	Fecha= 14-03-2020 Hora= 15:51 Tipo= Robo Ubicación= Ubicación actual Descripción= Mensaje de voz= No	Parámetro obligatorio Descripción faltante.	Fallar	Mensaje error:"¡Atención! El campo Descripción es requerido"
6	Fecha= 14-03-2020 Hora= 28:98 Tipo= Robo Ubicación= Ubicación actual Descripción= Una persona en moto me robó Mensaje de voz= No	Parámetro Hora incorrecto.	Fallar	Mensaje error:"¡Atención! Formato del campo Hora inválido"
7	Fecha= 14-03-2025 Hora= 15:51 Tipo= Robo Ubicación= Ubicación actual Descripción= Una persona en moto me robó Mensaje de voz= No	Parámetro Fecha mayor a fecha actual.	Fallar	Mensaje error:"¡Atención! El campo Fecha no puede ser posterior a la fecha actual"

11.4.2 Sistema web

11.4.2.1 Módulo auditar denuncia

ID caso de prueba	SISWEB-CP2
Módulo	Auditar denuncia
Descripción	Comprobar la auditoría de denuncias y los posibles estados de la misma
Preparado por	Sonia
Probado por	Sonia

Funcionalidad	Aceptar denuncia			
Pasos previos	<ol style="list-style-type: none"> 1. Ingresar al sistema con usuario registrado y con permiso "auditar denuncia" 2. Desde el menú dirigirse a "Auditar denuncia" 3. Seleccionar una denuncia a auditar 			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar si se puede aceptar una denuncia sin categorizarla	<ol style="list-style-type: none"> 1. Deslizarse hacia el fondo de la página 2. Presionar el botón "Aceptar denuncia" 	Fallar	Falla	Mensaje de error: "¡Atención! Debe seleccionar al menos una categoría de delito para aceptar la denuncia."
Comprobar si se puede aceptar una denuncia luego de categorizarla	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Deslizarse hacia el fondo de la página 4. Presionar el botón "Aceptar denuncia" 	Pasar	Pasa	

Comprobar si se puede aceptar una denuncia con subcategoría delito	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Seleccionar una subcategoría delito de la lista desplegable 4. Presionar el botón "agregar" 5. Deslizarse hacia el fondo de la página 6. Presionar el botón "Aceptar denuncia" 	Pasar	Pasa	
Comprobar si se puede aceptar una denuncia con observaciones	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Dirigirse al campo "observaciones" 4. Completar con texto 5. Deslizarse hacia el fondo de la página 6. Presionar el botón "Aceptar denuncia" 	Pasar	Pasa	

Funcionalidad	Desestimar denuncia			
Pasos previos	<ol style="list-style-type: none"> 1. Ingresar al sistema con usuario registrado y con permiso "auditar denuncia" 2. Desde el menú dirigirse a "Auditar denuncia" 3. Seleccionar una denuncia a auditar 			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar si se puede desestimar una denuncia	<ol style="list-style-type: none"> 1. Deslizarse hacia el fondo de la página 2. Presionar el botón "Desestimar denuncia" 	Pasar	Pasa	

Comprobar si se puede desestimar una denuncia luego de categorizarla	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Deslizarse hacia el fondo de la página 4. Presionar el botón "Desestimar denuncia" 	Pasar	Pasa	
Comprobar si se puede desestimar una denuncia con subcategoría delito	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Seleccionar una subcategoría delito de la lista desplegable 4. Presionar el botón "agregar" 5. Deslizarse hacia el fondo de la página 6. Presionar el botón "Desestimar denuncia" 	Pasar	Pasa	
Comprobar si se puede desestimar una denuncia con observaciones	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Dirigirse al campo "observaciones" 4. Completar con texto 5. Deslizarse hacia el fondo de la página 6. Presionar el botón "Desestimar denuncia" 	Pasar	Pasa	

Funcionalidad	Dar strike a dispositivo
Pasos previos	<ol style="list-style-type: none"> 1. Ingresar al sistema con usuario registrado y con permiso "auditar denuncia" 2. Desde el menú dirigirse a "Auditar denuncia" 3. Seleccionar una denuncia a auditar

Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar si se puede dar un strike al dispositivo denunciante	<ol style="list-style-type: none"> 1. Deslizarse hacia el fondo de la página 2. Presionar el botón "Dar strike dispositivo" 3. Presionar "aceptar" en el mensaje de confirmación 	Pasar	Pasa	

11.4.2.2 Módulo analizar denuncia

ID caso de prueba	SISWEB-CP3
Módulo	Analizar denuncia
Descripción	Comprobar el análisis de denuncias y los posibles estados de la misma
Preparado por	Sonia
Probado por	Sonia

Funcionalidad	Editar denuncia			
Pasos previos	<ol style="list-style-type: none"> 1. Ingresar al sistema con usuario registrado y con permiso "analizar denuncia" 2. Desde el menú dirigirse a "Analizar denuncia" 3. Seleccionar una denuncia a analizar 			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar si se puede editar una denuncia sin categorizarla	<ol style="list-style-type: none"> 1. Remover la/s categorías de delito de la denuncia 2. Deslizarse hacia el fondo de la página 3. Presionar el botón "Editar denuncia" 	Fallar	Falla	Mensaje de error: "No puede dejar una revisión sin categorías. Agregue al menos una y luego borre la que necesite."

Comprobar si se puede editar una denuncia luego de recategorizarla	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Deslizarse hacia el fondo de la página 4. Presionar el botón "Editar denuncia" 	Pasar	Pasa	
Comprobar si se puede editar una denuncia con subcategoría delito	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Seleccionar una subcategoría delito de la lista desplegable 4. Presionar el botón "agregar" 5. Deslizarse hacia el fondo de la página 6. Presionar el botón "Editar denuncia" 	Pasar	Pasa	
Comprobar si se puede editar una denuncia con observaciones	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Dirigirse al campo "observaciones" 4. Cambiar el contenido del campo 5. Deslizarse hacia el fondo de la página 6. Presionar el botón "Editar denuncia" 	Pasar	Pasa	

Funcionalidad	Desestimar denuncia			
Pasos previos	<ol style="list-style-type: none"> 1. Ingresar al sistema con usuario registrado y con permiso "analizar denuncia" 2. Desde el menú dirigirse a "Analizar denuncia" 3. Seleccionar una denuncia a analizar 			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas

Comprobar si se puede desestimar una denuncia	<ol style="list-style-type: none"> 1. Deslizarse hacia el fondo de la página 2. Presionar el botón "Desestimar denuncia" 	Pasar	Pasa	
Comprobar si se puede desestimar una denuncia luego de categorizarla	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Deslizarse hacia el fondo de la página 4. Presionar el botón "Desestimar denuncia" 	Pasar	Pasa	
Comprobar si se puede desestimar una denuncia con subcategoría delito	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Seleccionar una subcategoría delito de la lista desplegable 4. Presionar el botón "agregar" 5. Deslizarse hacia el fondo de la página 6. Presionar el botón "Desestimar denuncia" 	Pasar	Pasa	
Comprobar si se puede desestimar una denuncia con observaciones	<ol style="list-style-type: none"> 1. Seleccionar una categoría delito de la lista desplegable 2. Presionar el botón "agregar" 3. Dirigirse al campo "observaciones" 4. Editar el texto del campo 5. Deslizarse hacia el fondo de la página 6. Presionar el botón "Desestimar denuncia" 	Pasar	Pasa	

Funcionalidad	Dar strike a dispositivo
Pasos previos	<ol style="list-style-type: none"> 1. Ingresar al sistema con usuario registrado y con permiso "analizar denuncia" 2. Desde el menú dirigirse a "Analizar denuncia"

	3. Seleccionar una denuncia a analizar			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar si se puede dar un strike al dispositivo denunciante	<ol style="list-style-type: none"> 1. Deslizarse hacia el fondo de la página 2. Presionar el botón "Dar strike dispositivo" 3. Presionar "aceptar" en el mensaje de confirmación 	Pasar	Pasa	

11.4.2.3 Módulo reportes

ID caso de prueba	SISWEB-CP4
Módulo	Reportes
Descripción	Comprobar la generación de reportes y su exportación
Preparado por	Sonia
Probado por	Sonia

Funcionalidad	Generar reportes			
Pasos previos	<ol style="list-style-type: none"> 1. Ingresar al sistema con usuario registrado y con permiso "generar reportes" 2. Desde el menú dirigirse a "Reportes" 			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar que la lista desplegable "Subcategoría delito" se actualiza con las pertenecientes a la "Categoría delito" seleccionada	<ol style="list-style-type: none"> 1. Seleccionar un valor de la lista "Categoría delito" distinto de "Todas" 2. Desplegar la lista "Subcategoría delito" y comprobar si se actualizó con las subcategorías del delito seleccionado 	Pasar	Pasa	

<p>Comprobar que la lista desplegable "Subcategoría delito" se vacía al seleccionar la opción "Todas" en "Categoría delito"</p>	<ol style="list-style-type: none"> 1. Seleccionar un valor de la lista "Categoría delito" distinto de "Todas" 2. Desplegar la lista "Subcategoría delito" y comprobar si se actualizó con las subcategorías del delito seleccionado 3. Seleccionar la opción "Todas" de la lista "Categoría delito" 4. Comprobar que la lista "Subcategoría delito" solo cuenta con su opción "Todas" 	<p>Pasar</p>	<p>Pasa</p>	
<p>Comprobar que se pueden generar reportes</p>	<ol style="list-style-type: none"> 1. Para cada fila de la tabla "Conjunto de casos de prueba", seleccionar los valores indicados en la misma 2. Presionar el botón "buscar" 3. Comprobar si se generan resultados 	<p>Resultado esperado de acuerdo a la fila que se esté probando de la tabla "Conjunto de casos de prueba"</p>	<p>Resultado actual de acuerdo a la fila que se esté probando de la tabla "Conjunto de casos de prueba"</p>	

Conjunto de datos de prueba				
Nro. conjunto	Valores conjuntos de datos	Comentarios	Resultado esperado	Resultado actual
<p>1</p>	<p>Fecha inicio=01-01-2020 Fecha fin=30-03-2020 Categoría delito=Todas Subcategoría delito=Todas</p>	<p>Parámetros correctos</p>	<p>Pasar</p>	<p>Pasa</p>

2	Fecha inicio=01-01-2020 Fecha fin=30-03-2020 Categoría delito=Contra las personas Subcategoría delito=Todas	Parámetros correctos	Pasar	Pasa
3	Fecha inicio=01-01-2020 Fecha fin=30-03-2020 Categoría delito=Contra las personas Subcategoría delito=Homicidio doloso consumado	Parámetros correctos	Pasar	Pasa
4	Fecha inicio=31-03-2020 Fecha fin=30-03-2020 Categoría delito=Todas Subcategoría delito=Todas	Fecha inicio superior a Fecha fin	Fallar	Mensaje error: "¡Atención! La fecha de inicio debe ser menor a la de fin."
5	Fecha inicio=29-03-2019 Fecha fin=30-03-2020 Categoría delito=Todas Subcategoría delito=Todas	Intervalo entre Fecha inicio y Fecha fin superior a 365 días	Fallar	Mensaje error:" Atención! El intervalo máximo permitido es un año."
6	Fecha inicio= Fecha fin=30-03-2020 Categoría delito=Todas Subcategoría delito=Todas	Parámetro obligatorio faltante.	Fallar	Mensaje error:" Atención! Debe completar la fecha de inicio."
7	Fecha inicio=29-03-2020 Fecha fin= Categoría delito=Todas Subcategoría delito=Todas	Parámetro obligatorio faltante.	Fallar	Mensaje error:" Atención! Debe completar la fecha de fin."

Funcionalidad	Exportar reporte generado			
Pasos previos	<ol style="list-style-type: none"> 1. Ingresar al sistema con usuario registrado y con permiso "generar reportes" 2. Desde el menú dirigirse a "Reportes" 3. Generar correctamente un reporte con resultados 4. Presionar el botón "exportar" 			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar si se puede exportar un reporte sin título de archivo	<ol style="list-style-type: none"> 1. Dejar en blanco el campo "nombre de archivo" 2. Presionar el botón "Exportar" 	Fallar	Falla	Mensaje de error: "¡Ha ocurrido un error! Debe colocar un nombre al archivo."
Comprobar si se puede exportar un reporte en formato Excel	<ol style="list-style-type: none"> 1. Colocar texto en el campo "nombre de archivo" 2. Seleccionar formato Excel 3. Presionar el botón "Exportar" 	Pasar	Pasa	
Comprobar si se puede exportar un reporte en formato CSV	<ol style="list-style-type: none"> 1. Colocar texto en el campo "nombre de archivo" 2. Seleccionar formato CSV 3. Presionar el botón "Exportar" 	Pasar	Pasa	

11.4.2.4 Módulo estadísticas

ID caso de prueba	SISWEB-CP5
Módulo	Estadísticas
Descripción	Comprobar la generación de estadísticas
Preparado por	Sonia
Probado por	Sonia

Funcionalidad	Generar estadísticas
Pasos previos	1. Ingresar al sistema con usuario registrado y con permiso "generar estadísticas"

	2. Desde el menú dirigirse a "Estadísticas"			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar que se pueden generar estadísticas	1. Para cada fila de la tabla " Conjunto de casos de prueba ", seleccionar los valores indicados en la misma 2. Presionar el botón "buscar" 3. Comprobar si se generan resultados	Resultado esperado de acuerdo a la fila que se esté probando de la tabla "Conjunto de casos de prueba"	Resultado actual de acuerdo a la fila que se esté probando de la tabla "Conjunto de casos de prueba"	

Conjunto de datos de prueba				
Nro. conjunto	Valores conjuntos de datos	Comentarios	Resultado esperado	Resultado actual
1	Fecha inicio= Fecha fin=	Parámetros correctos	Pasar	Pasa
2	Fecha inicio= Fecha fin=30-03-2020	Parámetros correctos	Pasar	Pasa
3	Fecha inicio=29-03-2020 Fecha fin=	Parámetros correctos	Pasar	Pasa
4	Fecha inicio=31-03-2020 Fecha fin=30-03-2020	Fecha inicio superior a Fecha fin	Fallar	Mensaje error: "¡Atención! La fecha de inicio debe ser menor a la de fin."

11.4.2.5 Módulo estadísticas anuales

ID caso de prueba	SISWEB-CP6
Módulo	Estadísticas anuales

Descripción	Comprobar la generación de estadísticas anuales
Preparado por	Sonia
Probado por	Sonia

Funcionalidad	Generar estadísticas anuales			
Pasos previos	1. Ingresar al sistema con usuario registrado y con permiso "generar estadísticas" 2. Desde el menú dirigirse a "Estadísticas anuales"			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar que se pueden generar estadísticas anuales	1. Seleccionar un valor de la lista "Año" 2. Presionar botón "buscar"	Pasar	Pasa	

11.4.2.6 Módulo mapas

ID caso de prueba	SISWEB-CP7
Módulo	Mapas
Descripción	Comprobar la generación de mapas
Preparado por	Sonia
Probado por	Sonia

Funcionalidad	Generar mapas			
Pasos previos	1. Ingresar al sistema con usuario registrado y con permiso "generar mapas" 2. Desde el menú dirigirse a "Mapas"			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas

Comprobar que la lista desplegable "Subcategoría delito" se actualiza con las pertenecientes a la "Categoría delito" seleccionada	<ol style="list-style-type: none"> 1. Seleccionar un valor de la lista "Categoría delito" distinto de "Todas" 2. Desplegar la lista "Subcategoría delito" y comprobar si se actualizó con las subcategorías del delito seleccionado 	Pasar	Pasa	
Comprobar que la lista desplegable "Subcategoría delito" se vacía al seleccionar la opción "Todas" en "Categoría delito"	<ol style="list-style-type: none"> 1. Seleccionar un valor de la lista "Categoría delito" distinto de "Todas" 2. Desplegar la lista "Subcategoría delito" y comprobar si se actualizó con las subcategorías del delito seleccionado 3. Seleccionar la opción "Todas" de la lista "Categoría delito" 4. Comprobar que la lista "Subcategoría delito" solo cuenta con su opción "Todas" 	Pasar	Pasa	
Comprobar que se pueden generar mapas	<ol style="list-style-type: none"> 1. Para cada fila de la tabla "Conjunto de casos de prueba", seleccionar los valores indicados en la misma 2. Presionar el botón "buscar" 3. Comprobar si se generan resultados 	Resultado esperado de acuerdo a la fila que se esté probando de la tabla "Conjunto de casos de prueba"	Resultado actual de acuerdo a la fila que se esté probando de la tabla "Conjunto de casos de prueba"	

Nro. conjunto	Valores conjuntos de datos	Comentarios	Resultado esperado	Resultado actual
1	Fecha inicio=01-01-2020 Fecha fin=30-03-2020 Categoría delito=Todas Subcategoría delito=Todas	Parámetros correctos	Pasar	Pasa

2	Fecha inicio=01-01-2020 Fecha fin=30-03-2020 Categoría delito=Contra las personas Subcategoría delito=Todas	Parámetros correctos	Pasar	Pasa
3	Fecha inicio=01-01-2020 Fecha fin=30-03-2020 Categoría delito=Contra las personas Subcategoría delito=Homicidio doloso consumado	Parámetros correctos	Pasar	Pasa
4	Fecha inicio=31-03-2020 Fecha fin=30-03-2020 Categoría delito=Todas Subcategoría delito=Todas	Fecha inicio superior a Fecha fin	Fallar	Mensaje error: "¡Atención! La fecha de inicio debe ser menor a la de fin."
5	Fecha inicio=29-03-2019 Fecha fin=30-03-2020 Categoría delito=Todas Subcategoría delito=Todas	Intervalo entre Fecha inicio y Fecha fin superior a 365 días	Fallar	Mensaje error:" Atención! El intervalo máximo permitido es un año."
6	Fecha inicio= Fecha fin=30-03-2020 Categoría delito=Todas Subcategoría delito=Todas	Parámetro obligatorio faltante.	Fallar	Mensaje error:"¡Atención! Debe completar la fecha de inicio."
7	Fecha inicio=29-03-2020 Fecha fin= Categoría delito=Todas Subcategoría delito=Todas	Parámetro obligatorio faltante.	Fallar	Mensaje error:"¡Atención! Debe completar la fecha de fin."

Funcionalidad	Visualizar detalles denuncia			
Pasos previos	1. Ingresar al sistema con usuario registrado y con permiso "generar mapas" 2. Desde el menú dirigirse a "Mapas" 3. Generar correctamente un mapa con resultados			
Descripción	Pasos	Resultado esperado	Resultado Actual	Notas
Comprobar si se puede visualizar los datos de la denuncia	1. Presionar sobre un marcador del mapa generado 2. Presionar el botón "ver detalles"	Se abre una nueva pestaña del navegador con los datos pertenecientes a la denuncia seleccionada	Pasa	

11.4.3 API

11.4.3.1 Ruta denuncia

ID caso de prueba	API-CP2
Ruta	/api/v1/denuncia
Método HTTP	POST
Descripción	Comprobar la funcionalidad que permite almacenar una denuncia enviada desde la aplicación móvil
Preparado por	Sonia
Probado por	Sonia

Funcionalidad	Comprobar que una denuncia pueda ser almacenada en la base de datos
Pasos previos	1. Utilizar un cliente HTTP para realizar las pruebas 2. Colocar la dirección web de la API seguido del parámetro "ruta" de la tabla anterior 3. Ejecutar llamadas HTTP por cada fila de la tabla " Conjunto de datos de prueba " IMPORTANTE: el valor "TOKEN_VALUE" simboliza un token de

	sesión obtenido a través de un ingreso exitoso a la aplicación móvil.
--	---

Conjunto de datos de prueba				
Nro. conjunto	Valores conjuntos de datos	Comentario	Resultado esperado	Resultado actual
1	token_dispositivo=TOKEN_VAL UE fecha=23-02-2020 17:30:20 tipo_denuncia_app=1 coordenadas=POINT (-31.740562, -60.510161) descripcion="Descripción del hecho" audio=audio_hecho.mp3	Parámetros de la petición correctos	Pasar	HTTP Status:200 Mensaje: "Denuncia cargada correctamente"
2	token_dispositivo=TOKEN_VAL UE fecha=23-02-2020 17:30:20 tipo_denuncia_app=1 coordenadas=POINT (-31.740562, -60.510161) descripcion= audio=audio_hecho.mp3	Parámetros de la petición correctos.	Pasar	HTTP Status:200 Mensaje: "Denuncia cargada correctamente"
3	token_dispositivo=TOKEN_VAL UE fecha=23-02-2020 17:30:20 tipo_denuncia_app=1 coordenadas=POINT (-31.740562, -60.510161) descripcion="Descripción del hecho" audio=	Parámetros de la petición correctos.	Pasar	HTTP Status:200 Mensaje: "Denuncia cargada correctamente"

4	token_dispositivo= fecha=23-02-2020 17:30:20 tipo_denuncia_app=1 coordenadas=POINT (-31.740562, -60.510161) descripcion="Descripción del hecho" audio=audio_hecho.mp3	Parámetro obligatorio faltante.	Fallar	HTTP Status:404 Mensaje: "El dispositivo no pudo ser encontrado"
5	token_dispositivo=TOKEN_VAL UE fecha= tipo_denuncia_app=1 coordenadas=POINT (-31.740562, -60.510161) descripcion="Descripción del hecho" audio=audio_hecho.mp3	Parámetro obligatorio faltante.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"
6	token_dispositivo=TOKEN_VAL UE fecha=23-02-2020 17:30:20 tipo_denuncia_app= coordenadas=POINT (-31.740562, -60.510161) descripcion="Descripción del hecho" audio=audio_hecho.mp3	Parámetro obligatorio faltante.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"
7	token_dispositivo=TOKEN_VAL UE fecha=23-02-2020 17:30:20 tipo_denuncia_app=1 coordenadas= descripcion="Descripción del hecho" audio=audio_hecho.mp3	Parámetro obligatorio faltante.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"
8	token_dispositivo=TOKEN_VAL UE fecha=23-02-2020 17:30:20 tipo_denuncia_app=1 coordenadas=POINT (-31.740562, -60.510161) descripcion=	Al menos uno de los dos parámetros faltantes debe estar presente.	Fallar	HTTP Status:400 Mensaje: "Petición incorrecta"

	audio=			
--	--------	--	--	--

12. Bibliografía

- [1] Collado, C. (21 de enero de 2016). Aplicaciones nativas vs. híbridas, ¿qué son y cuáles son mejores? <https://andro4all.com/2016/01/apps-nativas-vs-apps-hibridas-ventajas-desventajas>
- [2] Wikipedia. (s.f.). Servicio web. https://es.wikipedia.org/wiki/Servicio_web
- [3] Wikipedia. (s.f.). Programas para control de versiones. https://es.wikipedia.org/wiki/Programas_para_control_de_versiones
- [4] Ministerio de Justicia y Derechos Humanos. (s.f.) CÓDIGO PENAL DE LA NACIÓN ARGENTINA. LEY 11.179. <http://servicios.infoleg.gob.ar/infolegInternet/anexos/15000-19999/16546/texact.htm>
- [5] Sutherland, J. Ken Schwaber, K. (2013). Scrum guide. Artefactos de Scrum. Lista de Producto (Product Backlog). Página 15. ©2014 Scrum.Org and ScrumInc.
- [6] Sutherland, J. Ken Schwaber, K. (2013). Scrum guide. El equipo Scrum (Scrum Team). El dueño de Producto (Product Owner). Página 6. ©2014 Scrum.Org and ScrumInc.
- [7] Spatie. (s.f.) laravel-activitylog. <https://github.com/spatie/laravel-activitylog>
- [8] Laravel. (s.f.). Passwords. <https://laravel.com/docs/5.4/passwords>
- [9] Highcharts. (s.f.). <https://www.highcharts.com/>
- [10] Leaflet js. (s.f.). <https://leafletjs.com/>
- [11] Visual Studio Code. (s.f.). <https://code.visualstudio.com/>
- [12] Node JS. (s.f.). <https://nodejs.org/es/>
- [13] Postman. (s.f.). <https://www.postman.com/>
- [14] Git. (s.f.). <https://git-scm.com/>
- [15] Gitlab. (s.f.). <https://gitlab.com/>
- [16] Android Studio. (s.f.). <https://developer.android.com/studio>
- [17] Lara, W. (2015). ¿Cómo funciona la metodología Scrum? <https://platzi.com/blog/metodologia-scrum-fases/>
- [18] Be agile my friend. (s.f.). ¿Qué es el Sprint Planning? <https://www.beagilemyfriend.com/sprint-planning/>

- [19] PMO Informática. (25 de noviembre de 2013). Plantillas Scrum: Pila de producto (Product Backlog). <http://www.pmoinformatica.com/2013/11/plantillas-scrum-pila-producto-product.html>
- [20] Laravel. (s.f.). Migrations. <https://laravel.com/docs/5.4/migrations>
- [21] Laravel. (s.f.). Seeding. <https://laravel.com/docs/5.4/seeding>
- [22] Easy QA. (s.f.). Types of software testing. <https://geteasyqa.com/es/qa/software-testing-types/>
- [23] Easy QA. (s.f.). Best Test Case templates with examples. <https://geteasyqa.com/es/qa/best-test-case-templates-examples/>
- [24] Developers Android. (última actualización, 3 de junio de 2020). Prácticas recomendadas para identificadores únicos. Casos prácticos y el identificador adecuado que debe usarse. <https://developer.android.com/training/articles/user-data-ids#common-use-cases>
- [25] Developers Android. (última actualización, 3 de junio de 2020). Prácticas recomendadas para identificadores únicos. Cómo trabajar con los ID de instancia y GUID. <https://developer.android.com/training/articles/user-data-ids#instance-ids-guids>
- [26] Google Recaptcha. (última actualización, 16 de julio de 2021). recaptcha v3. <https://developers.google.com/recaptcha/docs/v3>