

UNIVERSIDAD NACIONAL DE SAN CRISTÓBAL DE
HUAMANGA FACULTAD DE INGENIERÍA DE MINAS
GEOLOGÍA Y CIVIL ESCUELA PROFESIONAL DE
INGENIERÍA DE SISTEMAS



SOFTWARE GEOGRÁFICO QUE DESCRIBE LA
ACTIVIDAD DELICTIVA, CIUDAD DE AYACUCHO, 2017

Tesis para optar el Título Profesional de:

INGENIERO DE SISTEMAS

Presentado por:

Bach. Junior Grover Flores Martínez

AYACUCHO – PERÚ

2019

DEDICATORIA

A mis padres, por ser el pilar fundamental en mi formación personal y profesional; por su apoyo incondicional perfectamente mantenido a través del tiempo.

A mis hermanos, por apoyarme en todo momento y ser parte de mis logros.

La perseverancia y determinación son clave para conseguir todo lo que se desea.

AGRADECIMIENTO

A la Universidad Nacional San Cristóbal de Huamanga, por brindarme sus conocimientos y permitirme conocer a buenos catedráticos y amigos.

A mis compañeros, quienes me dieron perspectivas diferentes y con sus aportaciones y experiencias me ayudaron a incrementar la calidad del trabajo.

CONTENIDO

DEDICATORIA.....	I
AGRADECIMIENTO.....	II
CONTENIDO.....	III
RESUMEN.....	V
INTRODUCCIÓN.....	VI

CAPÍTULO I

PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1 DIAGNÓSTICO Y ENUNCIADO DEL PROBLEMA	1
1.2 FORMULACIÓN DEL PROBLEMA DE INVESTIGACIÓN	4
PROBLEMA GENERAL	4
PROBLEMAS ESPECÍFICOS	4
1.3 OBJETIVOS DE LA INVESTIGACIÓN.....	4
1.4 JUSTIFICACIÓN Y DELIMITACIÓN DE LA INVESTIGACIÓN.....	5

CAPÍTULO II REVISIÓN DE LA

LITERATURA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN.....	7
2.2. MARCO TEÓRICO.....	8
2.2.1. ACTIVIDAD DELICTIVA	8
2.2.2. MAPA DEL DELITO	10
2.2.3. PUNTO CALIENTE.....	11
2.2.4. SISTEMA DE INFORMACIÓN GEOGRÁFICA	11
2.2.5. PROGRAMACIÓN EXTREMA.....	13
2.2.6. PROGRAMACIÓN ORIENTADA A OBJETOS.....	25
2.2.7. TECNOLÓGICAS DE INTERNET	26

CAPÍTULO III METODOLOGÍA DE LA

INVESTIGACIÓN

3.1 TIPO DE INVESTIGACIÓN	29
3.2 NIVEL DE INVESTIGACIÓN	29
3.3 DISEÑO DE INVESTIGACIÓN.....	29
3.4 POBLACIÓN Y MUESTRA DE LA INVESTIGACIÓN	29
3.4.1 POBLACIÓN.....	29
3.4.2 MUESTRA.....	30

3.5	VARIABLES E INDICADORES DE LA INVESTIGACIÓN	30
3.5.1	DEFINICIÓN CONCEPTUAL DE LAS VARIABLES	30
3.5.2	DEFINICIÓN OPERACIONAL DE LAS VARIABLES	30
3.6	OPERACIONALIZACIÓN DE LA VARIABLE	31
3.7	TÉCNICA E INSTRUMENTO PARA RECOLECTAR INFORMACIÓN	31
3.7.1	TÉCNICA PARA RECOLECTAR INFORMACIÓN	31
3.7.2	INSTRUMENTO PARA RECOLECTAR INFORMACIÓN	31
3.8	HERRAMIENTA PARA EL TRATAMIENTO DE DATOS E INFORMACIÓN..	31
3.9	TÉCNICAS PARA APLICAR PROGRAMACIÓN EXTREMA.....	40

CAPÍTULO IV

ANÁLISIS Y RESULTADOS DE LA INVESTIGACIÓN

4.1	ARTEFACTOS DE SOFTWARE APLICANDO PROGRAMACIÓN EXTREMA	43
4.1.1	FASE DE EXPLORACIÓN	43
4.1.2	FASE DE PLANIFICACIÓN	61
4.1.3	FASE DE ITERACIÓN	62

CAPÍTULO V CONCLUSIONES Y

RECOMENDACIONES

5.1	CONCLUSIONES	158
5.2	RECOMENDACIONES	159
	REFERENCIA BIBLIOGRÁFICA.....	160

RESUMEN

La inseguridad ciudadana es un problema social que durante años aqueja a los peruanos. La ciudad de Ayacucho no es ajena a este problema, ya que presenta un elevado índice de inseguridad ciudadana, las encuestas realizadas por la INEI posicionan a la ciudad de Ayacucho como una de las ciudades más peligrosas del Perú al contar con diversos puntos críticos en donde la probabilidad de ocurrencia de un incidente delictivo es mayor al promedio. En la presente tesis se construyó un medio que describe la actividad delictiva y los muestra en un mapa del delito, del mismo modo, muestra estadísticas sobre la tipicidad, antijuridicidad y la culpabilidad del delito.

El objetivo fue desarrollar un software geográfico para describir la actividad delictiva en la ciudad de Ayacucho, el estudio se desarrolló con los registros de incidentes delictivos de los distritos de Jesús Nazareno y Carmen Alto del año 2017. El tipo de investigación es descriptivo, retrospectivo, longitudinal y observacional.

En la investigación, se utilizó la metodología de Programación Extrema (XP) que presenta un ciclo de vida iterativo e incremental y los pasos necesarios para el desarrollo del software geográfico, se utilizó la georreferenciación para registrar y manipular datos espaciales, posteriormente, se mostró los resultados en un mapa del delito de la ciudad de Ayacucho.

Finalmente, se describió la tipicidad presentando el mapa del delito del distrito de Jesús Nazareno y Carmen Alto del año 2017, del mismo modo, se mostró los tipos de delito representados en la nube de delitos de ambos distritos; por otro lado, la frecuencia de ocurrencia en los diferentes sectores y durante los días de la semana muestran el comportamiento de la antijuridicidad y la culpabilidad de ambos distritos.

Palabras claves: Actividad Delictiva, Tipicidad, Antijuridicidad, Culpabilidad, Mapa Delictivo, Georreferenciación, Programación Extrema.

INTRODUCCIÓN

La actividad delictiva se presenta como una conducta típica, antijurídica y culpable que se adecua al presupuesto jurídico de una ley penal, dicha actividad representa un problema para la seguridad ciudadana que afecta a la sociedad. La utilización de un software geográfico ofrece la posibilidad de visualizar mejor el comportamiento delictivo de un área geográfica, ayudando a tomar decisiones de forma rápida, exacta y estratégica; dichas acciones permiten el alcance de la convivencia pacífica, la erradicación de la actividad delictiva y la utilización pacífica de las vías y espacios públicos, que son algunos de los objetivos de la Ley del Sistema Nacional de Seguridad Ciudadana (ley n° 27933).

Actualmente, la información que manejan los distritos de Jesús Nazareno y Carmen Alto respecto a la actividad delictiva no son exactas y representativas, por tal motivo, las estrategias que se toman para enfrentar la inseguridad ciudadana no son del todo efectivas. En ese sentido, se plantea una solución geográfica que describa la actividad delictiva de los distritos de Jesús Nazareno y Carmen Alto, ofreciendo de forma rápida, exacta, visual y oportuna la información de los incidentes delictivos.

Mis valores como la responsabilidad social, con respecto a los problemas en mi entorno, me motivaron a la elección del tema a investigar, el aportar con una solución tecnológica que permita representar de forma adecuada la actividad delictiva en la ciudad de Ayacucho, dará oportunidad a que las autoridades responsables puedan optar por estrategias más acertadas al momento de abordar este problema social.

Los objetivos específicos son: a) Describir la tipicidad presentando el mapa delictivo y los tipos de delito, b) Describir la antijuridicidad presentando por sectores sus frecuencias y periodos de ocurrencia y c) Describir la culpabilidad presentando por sectores sus frecuencias y periodos de ocurrencia.

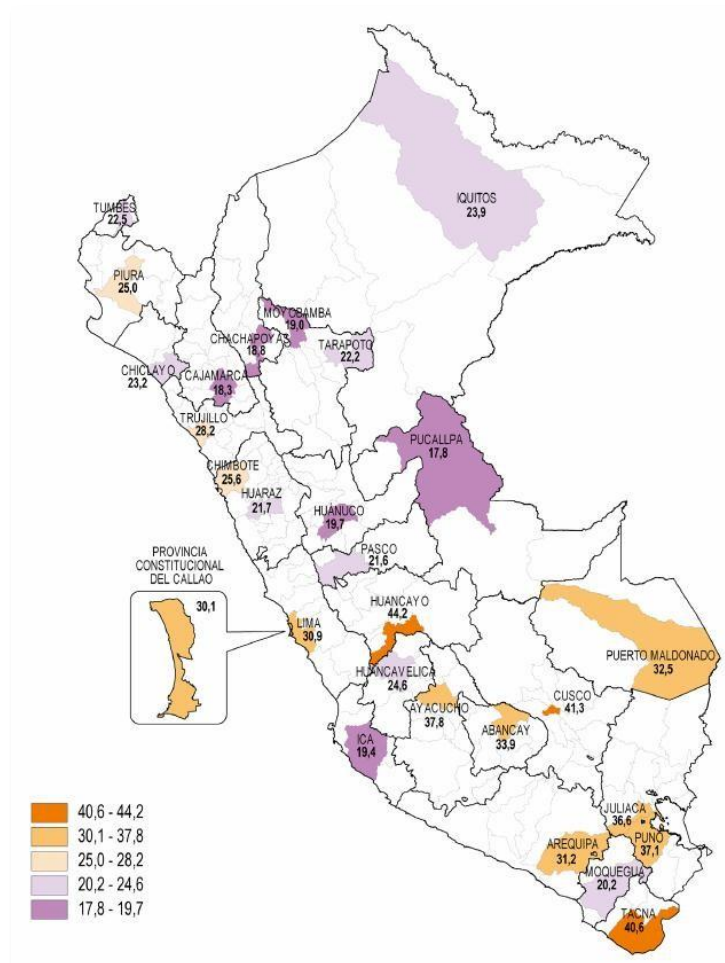
CAPÍTULO I PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1 DIAGNÓSTICO Y ENUNCIADO DEL PROBLEMA

La inseguridad ciudadana es un problema recurrente que forma parte de la convivencia cotidiana de los ciudadanos de Ayacucho, según la encuesta nacional de programas presupuestales de los años 2016-2017 (INEI), se afirma que la ciudad de Ayacucho, cuenta con un 37.8% de víctimas de algún hecho delictivo, lo cual es elevado, considerando que a nivel nacional el promedio de víctimas de algún hecho delictivo es de 30.2%, también para los años 2016 al 2017 la ciudad de Ayacucho tuvo un aumento del 2.2%, mientras que a nivel nacional se tuvo un descenso del 1.7%.

Figura 1.1

Ciudadanos de 15 y más años de edad, víctimas de algún hecho delictivo

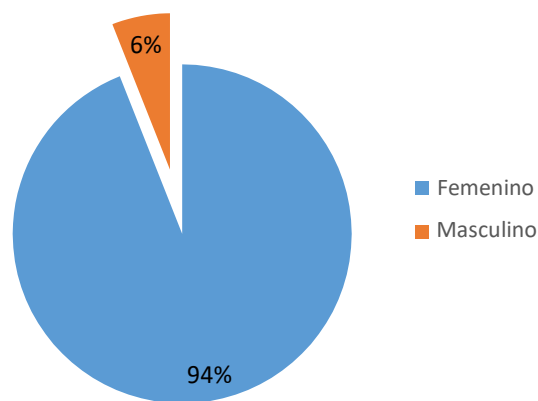


Según la INEI (2017)

Adicionalmente a lo dicho, en los últimos años los distritos metropolitanos de la ciudad de Ayacucho reportaron un aumento en la actividad delictiva, principalmente en la violencia familiar, robos y hurtos, lesiones contra la vida, el cuerpo y la salud y casos de disuasión de personas libando alcohol en la vía pública.

De las indagaciones se ha observado que, la sub gerencia de seguridad ciudadana del distrito de San Juan Bautista (SGSC - SJB) en su plan local de seguridad ciudadana (2018), reportó, que el principal problema que afronta el distrito es la violencia familia, con 158 casos en el año 2017, de estos casos se aprecia que el género femenino es el más vulnerable a las agresiones producto de la violencia familiar.

Gráfico 1.1
Porcentaje de víctimas de violencia familiar por género



Según el plan local de seguridad ciudadana, distrito de San Juan bautista (2018).

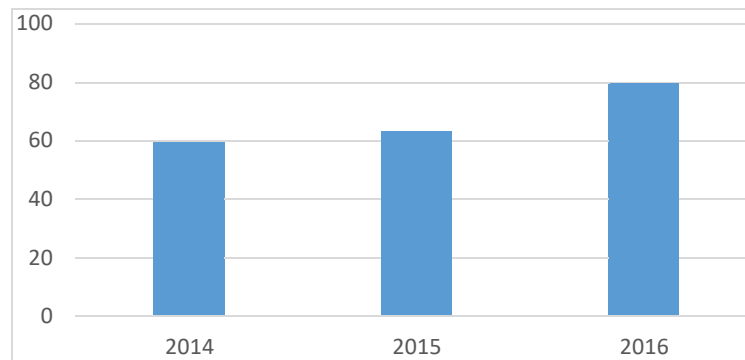
El segundo problema que afronta el distrito de San Juan Bautista, son las grescas callejeras, con un total de 110 casos en el año 2017, estos casos se presentan en su mayoría por el alto consumo de bebidas alcohólicas entre jóvenes y adultos; el tercer problema que afronta el distrito de San Juan Bautista, es la disuasión de personas libando alcohol en la vía pública, con un total de 112 casos registrados, plan local de seguridad ciudadana del distrito de San Juan Bautista (2018).

Del mismo modo, la sub gerencia de seguridad ciudadana del distrito de Carmen Alto, en su plan local de seguridad ciudadana (2017), señala, que la violencia familiar es el delito que encabeza las estadísticas delictivas del distrito, con un total de 91 casos registrados en el año 2016, es preciso señalar, que los casos denunciados van en aumento a partir del año 2014 en

un promedio de 8 casos por año, los casos registrados mantienen el mismo comportamiento victimilógico que el distrito de San Juan Bautista, con relación a las causas y circunstancias que conllevan la violencia familiar.

Gráfico 1.2

Violencia familiar registrada en los años 2014, 2015 y 2016

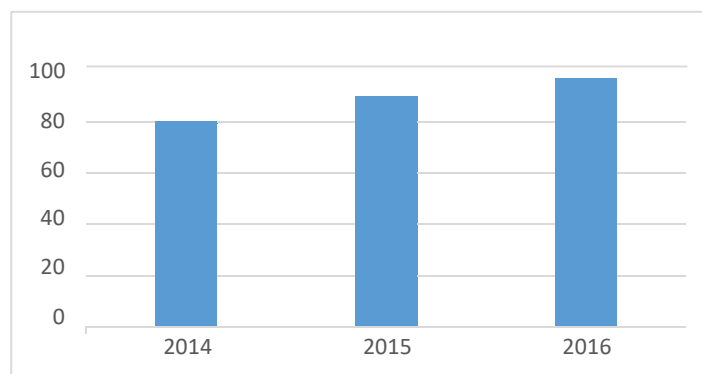


Según el plan local de seguridad ciudadana, distrito de Carmen Alto (2018)

Por otra parte, el delito que por tercer año consecutivo se ubica en el segundo, de acuerdo al registro de faltas y ocurrencias de la sub gerencia de seguridad ciudadana del distrito de Carmen Alto, son las lesiones contra la vida, el cuerpo y la salud con un total de 78 casos registrados en el año 2016. En este rubro se encuentran las peleas y las grescas que suscitan los fines de semana como consecuencias del alto consumo de bebidas alcohólicas en los recreos, bares cantinas y discotecas, el hurto es el tercer tipo de delito que más predomina en el registro del personal de serenazgo, sumando un total de 49 casos en el año 2016 y con un comportamiento ascendente en relación a los años anteriores.

Gráfico 1.3

Lesiones contra la vida, el cuerpo y la salud en los años 2014, 2015 y 2016



Según el plan local de seguridad ciudadana, distrito de Carmen Alto (2018).

En ese sentido, el comportamiento delictivo de los distritos de San Juan Bautista, Carmen Alto, Jesús Nazareno, Andrés Avelino Cáceres Dorregaray tienen el mismo comportamiento, a diferencia del distrito de Ayacucho, el principal problemas que afronta el distrito son los elevados índices delictivos con relación a robos y hurtos en la zona céntrica de la ciudad, la violencia familiar, infantil y alcoholismo se presentan con mayor frecuencia en los Asentamientos Humanos, Asociaciones y Barrios colindantes con los distritos aledaños, mientras que las grescas callejeras se presentan en todo el distrito, plan local de seguridad ciudadana del distrito de Ayacucho, (2018).

En este contexto y ante el aumento de los actos delictivos en la ciudad de Ayacucho, nace una problemática relacionada con la gestión de información en materia delictiva, a fin de crear un mapa del delito que describa de forma clara y oportuna los incidentes delictivos de la ciudad de Ayacucho. Todo ello con la intención de brindar un mejor servicio a la comunidad, pues el tamaño y la población de los gobiernos locales son grandes en relación a la cantidad de recursos logísticos y operativos que poseen.

1.2 FORMULACIÓN DEL PROBLEMA DE INVESTIGACIÓN

1.2.1. PROBLEMA GENERAL

¿Cómo es la actividad delictiva en la ciudad de Ayacucho, 2017?

1.2.2. PROBLEMAS ESPECÍFICOS

a) ¿Cómo es la tipicidad, antijuridicidad y culpabilidad?

1.3 OBJETIVOS DE LA INVESTIGACIÓN

1.3.1. OBJETIVO GENERAL

Desarrollar un Software Geográfico mediante técnicas e instrumentos; programación extrema, georreferenciación, base de datos espacial, programación orientada a objetos, tecnologías de internet, con la finalidad de describir la actividad delictiva en la ciudad de Ayacucho, 2017.

1.3.2. OBJETIVOS ESPECÍFICOS

- a) Describir la tipicidad presentando el mapa del delito y los tipos de delito.
- b) Describir la antijuridicidad presentando por sectores sus frecuencias y periodos de ocurrencia.

- c) Describir la culpabilidad presentando por sectores sus frecuencias y periodos de ocurrencia.

1.4 JUSTIFICACIÓN Y DELIMITACIÓN DE LA INVESTIGACIÓN

1.4.1. IMPORTANCIA DEL TEMA

A. IMPORTANCIA TÉCNICA

La investigación tendrá como producto un mapa del delito, que podrá ser utilizado por las autoridades locales, como; policías, fiscales, jueces, bomberos y serenazgo como una herramienta fundamental para la descripción del comportamiento delictivo y patrones delincuenciales, permitiendo mejorar su presencia en los lugares y momentos necesarios. Así mismo, la investigación dispondrá de un modelo de desarrollo de un Sistema de Información Geográfico (GIS) para los estudiantes de ingeniería de sistemas, criminólogos e investigadores interesados.

B. IMPORTANCIA ECONÓMICA

Al contar con un entendimiento claro sobre los actos delictivos presentes en la ciudad de Ayacucho, la investigación tendrá como resultado un mapa del delito actualizado, en tiempo real, las autoridades descritas líneas antes, podrán reducir el tiempo y costo a la hora de tomar decisiones sobre el uso de recursos logísticos y operativos disponibles, dicha información permitirá mejorar el servicio sobre la seguridad ciudadana de los gobiernos locales.

C. IMPORTANCIA SOCIAL

Finalmente, la propuesta de desarrollar un “software geográfico que describe la actividad delictiva, ciudad de Ayacucho, 2017”, permitirá establecer una cooperación entre los diferentes gobiernos locales, instituciones de la administración pública, sin distinción del nivel de desarrollo tecnológico de estas; utilizar más fácilmente estándares abiertos y aplicaciones tecnológicas de distinta generación. Contar con estadísticas de criminalidad en tiempo real, permitirá comprender mejor el comportamiento delictivo de la ciudad de Ayacucho, de esta manera, las autoridades locales podrán mejorar la presencia de los servicios de seguridad ciudadana, estas medidas repercutirán de forma positiva en la población de Ayacucho, logrando mejorar la percepción del trabajo que realizan los agentes de seguridad en los gobiernos locales.

1.4.2. JUSTIFICACIÓN

La presente investigación es de gran importancia para los gobiernos locales, ya que la lucha contra la inseguridad ciudadana es de interés social y económico. Por tal motivo, es necesario contar con mecanismos que ayuden a comprender y dimensionar mejor el problema, en tal medida, el mapa del delito, permite expresar en cifras la relación entre los delitos y el lugar donde son cometidos, permitiendo direccionar los recursos humanos y logísticos hacia las zonas más afectadas, así como, desplegar propuestas preventivas; para el uso de los gobiernos locales y la población en general.

1.4.3. DELIMITACIÓN

La investigación se realizará en los distritos de Jesús Nazareno y Carmen Alto de la ciudad de Ayacucho, utilizando los registros almacenados del año 2017.

CAPÍTULO II

REVISIÓN DE LA LITERATURA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN

En la tesis para optar el grado de Magíster en Ciencia Política y Gobierno con mención en Gestión Pública y Políticas Públicas denominado “Uso de tecnologías para sistematización de la información sobre el crimen (usos, problemas de georreferencia y demás)”, de la Universidad Pontificia la Católica del Perú, se concluyó que el uso de las tecnologías de georreferenciación en las comisarías del Cercado de Lima es delicado, algunas comisarías emplean las herramientas gratuitas de Internet, para la elaboración y la actualización de mapas del delito, pero, en otros casos, aun se sigue empleando solo mapas impresos. Por otro lado, la Policía Nacional del Perú (PNP) no cuenta con un sistema único de información georreferenciado ni de plataformas similares para registrar la actividad delictiva del Cercado de Lima, dificultando la obtención y manejo de datos delictivos georreferenciados y la construcción de mapas del delito. La presencia de un sistema con información en materia delictiva es de gran utilidad para el diagnóstico y diseño de políticas públicas de seguridad ciudadana (Mora, 2015).

En la investigación “El mapa del crimen. Herramientas geográficas para policías y criminólogos” realizada en la ciudad de Málaga, se concluye que es posible prevenir la delincuencia en las ciudades con una buena gestión y planificación de las labores policiales, el patrullaje y la seguridad ciudadana. Para poder llegar a tales aseveraciones, fue necesario introducir sobre el plano de la ciudad de Málaga los datos recogidos de las encuestas de victimación, las llamadas telefónicas realizadas a la Policía Nacional y Local, así como las denuncias presentadas en la Policía Nacional (actividad delictiva) y las actuaciones de la Policía Local. Además, se introdujeron la tasa de desempleo y el censo de la ciudad de Málaga por secciones. Junto a estos datos, los autores utilizaron un Sistema de información Geográfica (SIG.), consiguiendo un mapa del delito exacto y variado (Stangeland y Garrido, 2004)

Según la investigación de Riveros y Zevallos (2016), en su tesis para optar el Título Profesional de Ingeniero de Sistemas denominado “Business Intelligence para el área de seguridad ciudadana en el distrito de Villa El Salvador mediante la metodología de Ralph

Kimball”, de la Universidad Nacional Mayor de San Marcos, concluye que los sistemas de soporte a la toma de decisiones son una necesidad no solo para organizaciones con fines de lucro como empresas sino también para organizaciones del estado como la Policía, el Serenazgo, bomberos, etc., a través del desarrollo de una solución en Inteligencia de Negocios que tiene como fuente principal la actividad delictiva presente en las incidencias registradas, se logró identificar las zonas más peligrosas del distrito de Villa El Salvador; las cuales deberían de tener mayor vigilancia y atención por las autoridades.

Para Jaulis y Vilcarromero (2015), en sus tesis para optar el título profesional de Ingeniero de Computación y Sistemas denominado “Sistema de predicción de hechos delictivos para la mejora del proceso de prevención del delito en el distrito de la Molina utilizando minería de datos”, de la Universidad de San Martín de Porres, concluye que la información de la actividad delictiva unificada y consolidada de las tres comisarías del distrito permitieron agilizar el proceso de prevención de delitos, del mismo modo permite simplificar las actividades a los efectivos policiales a tomar mejores decisiones con respecto a las zonas donde podría ocurrir un hecho delictivo, esto gracias a la implementación del sistema de predicción.

2.2. MARCO TEÓRICO

2.2.1. ACTIVIDAD DELICTIVA

Según Mezger (1955), El delito es una acción antijurídica, pero, al mismo tiempo, típicamente antijurídica, la decisión de si una determinada conducta incide en la parcela de derecho penal resulta de la consideración de que, como fundamento de la exigencia penal del Estado, no es bastante cualquier acción antijurídica, sino que es preciso una antijurídica tipificada y culpable.

Según Roxin (1997), el delito es toda acción (manifestación de la personalidad), típica (“nullum crimen”), antijurídica (soluciones sociales de conflictos), culpable (necesidad de pena, más cuestiones preventivas) y que cumple otros eventuales presupuestos de punibilidad

Según Zaffaroni, Plagia y Slokar (2005), el delito es una conducta humana individualizada mediante un dispositivo legal (tipo) que revela su prohibición (típica), que por no estar permitida por ningún precepto jurídico (causas de justificación), es contraria al orden

jurídico (antijurídica) y que, por serle exigible al autor que actuase de otra manera en esa circunstancia, le es reprochable (culpable).

A. TIPICIDAD

Para Rodríguez, Ugaz, Gamero y Schönbohm (2012), la tipicidad es una formula abstracta para señalar una conducta cuyo desvalor la hace acreedora de una pena, por lo general está constituido por todos aquellos elementos que caracterizan a una acción humana (antijuridicidad y culpabilidad) contraviniente a una normal, dicha acción es considerada como típica cuando es prohibida por el ordenamiento jurídico penal.

Según Muñoz y García (2002), la tipicidad es la adecuación del acto humano voluntario ejecutado por el sujeto a la figura descrita por la ley como delito. Es el encaje del acto humano voluntario al tipo penal. Si se adecua es indicio de que es delito, si la adecuación no es completa no hay delito.

Para Peña (2010), la tipicidad es una figura que crea el legislador para hacer una valoración de determinada conducta delictiva, se puede decir que es una descripción abstracta de la conducta prohibida, que tiene por función la individualización de las conductas humanas penalmente relevantes.

B. ANTIJURIDICIDAD

Según Muñoz y García (2002, p341), “la antijuridicidad expresa la contradicción entre la acción realizada y las exigencias del ordenamiento jurídico”.

Según Barja (2004), la antijuridicidad es el acto voluntario típico que contraviene el presupuesto de la norma penal, lesionando o poniendo en peligro bienes e intereses tutelados por el derecho. La antijuridicidad es un juicio impersonal objetivo sobre la contradicción existente entre el hecho y el ordenamiento jurídico.

Según Peña (2010), la antijuridicidad es la contradicción de la realización del tipo de norma prohibida, es un juicio de valor objetivo, en tanto se pronuncia sobre la conducta típica.

C. CULPABILIDAD

Para Rodríguez et al. (2012), una vez que se encuentra ante una acción típica y

antijurídica, la culpabilidad es la reprochabilidad de la conducta de una persona imputable y responsable, que pudiendo haberse conducido de otra manera no lo hizo, por lo cual es merecedor de una pena.

Para Roxin (1997), la acción típica y antijurídica ha de ser culpable, poniéndose responsable de ella al autor. Para ello es presupuesto la imputabilidad o capacidad de culpabilidad y la ausencia de causas de exculpación.

Para Portocarrero (2017), la culpabilidad es la conducta que se debe reprochar jurídicamente al sujeto por no haber hecho lo que debía hacer, cuando sabía que estaba haciendo algo distinto de lo obligado por el mandato o lo prohibido por Ley. Las condiciones dentro de las que actuó u omitió son consideradas por el derecho suficientes para permitirle optar entre cumplir el mandato o violarlo.

2.2.2. MAPA DEL DELITO

Según Pervesi (2007), el mapa del delito consiste en geo-referenciar los hechos delictivos, obteniendo una visualización geográfica que contempla no solo la distancia entre hechos, sino también el equipamiento urbano (bancos, comercios, plazas, etc) y entre las demarcaciones territoriales como comisarias, barrios, zonas marginales, etc.

Del mismo modo para Behar y Lucilli (2003), sugieren que la elaboración del mapa del delito debe verse como un proceso continuo y sistemático, ya que los delitos no se distribuyen al azar ni de forma uniforme sobre la geografía urbana, sino que se localizan o concentran en determinadas zonas y momentos. Estos puntos críticos de la delincuencia se pueden examinar en un mapa y, posteriormente analizar las causas que generan estas concentraciones, los mapas delictivos tienen distintas unidades de análisis; puntos, líneas, áreas y grillas.

Para Vázquez y Soto (2013), El mapa delictivo, principalmente, sirve para ayudar al personal operativo y administrativo en el diagnóstico situacional en la que se encuentra respecto a la seguridad ciudadana, la evaluación de la misma, la formulación de políticas nuevas o reformular las nuevas políticas existentes para la prevención de los hechos delictivos y constantemente monitorización de estos, también ayuda a la gestión policial, que comprende la identificación y solución de los problemas.

2.2.3. PUNTO CALIENTE

Según Vozmediano (2011), el punto caliente es un área que supera el número medio de eventos delictivos, donde el riesgo de ser víctima de un delito es superior a la media.

Del mismo modo Ruiz (2012), una zona caliente representa un área localizada que contiene una gran porción del volumen total de incidentes delictivos de toda esa región, es un área que presenta un nivel de delitos o desorden más elevado que el promedio.

Pero para Sherman (1995), los puntos calientes son lugares pequeños en los que la incidencia del delito es tan frecuente que es muy predecible, por lo menos durante un período de 1 año.

2.2.4. SISTEMA DE INFORMACIÓN GEOGRÁFICA

Según la INEGI (2014), un Sistema de información geográfico es un conjunto de herramientas diseñadas para obtener, almacenar, recuperar y desplegar datos espaciales del mundo real. Conjunto de mapas, de la misma porción del territorio, donde un lugar concreto tiene la misma localización (las mismas coordenadas) en todos los mapas.

Un sistema de información geográfico (GIS) es un conjunto de herramientas para unir, almacenar y presentar datos espaciales sobre el mundo que nos rodea (Burrough, 1986).

A GEORREFERENCIACIÓN

Según ARCGIS, La georreferenciación es el uso de coordenadas de mapa para asignar una ubicación espacial a entidades cartográficas. Todos los elementos de una capa de mapa tienen una ubicación geográfica y una extensión específicas que permiten situarlos en la superficie de la Tierra. La capacidad de localizar de manera precisa las entidades geográficas es fundamental tanto en la representación cartográfica como en sistema de información geográfico.

La georreferenciación es un proceso que permite determinar la posición de un elemento en un sistema de coordenadas espacial diferente al que se encuentra. Existen por tanto dos sistemas de coordenadas: el sistema origen y el sistema destino. Este proceso es determinado con una relación de posiciones entre elementos espaciales en ambos sistemas, de manera que, conociendo la posición en uno de los sistemas de coordenadas es posible obtener la posición homóloga en el otro sistema (Martínez, Francisco y Camacho, 2012).

B SISTEMA GESTOR DE BASE DE DATOS GEOGRÁFICAS


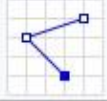
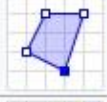

Para ESRI (1998), una base de datos espacial o geográfica, es una colección de datos organizados de tal manera que sirvan efectivamente para una o varias aplicaciones GIS (sistema de información geográfica). Esta base de datos comprende la asociación entre sus dos principales componentes: datos espaciales y atributos o datos no espaciales.

Según Haithcoat et al (2001), una base de datos espaciales es una colección de datos referenciados espacialmente que actúan como un modelo de la realidad, en el sentido de que representa una serie o aproximación de fenómenos. Esos fenómenos seleccionados son representados en forma digital. La representación digital se puede referir al pasado, presente o futuro.

C WELL KNOWN TEXT (WKT)

Es una codificación o sintaxis en formato ASCII estandarizada diseñada para describir objetos espaciales expresados de forma vectorial. Los objetos que es capaz de describir el formato WKT son los siguientes: puntos, multipuntos, líneas, multilíneas, polígonos, multipolígonos. Su especificación ha sido promovida por un organismo internacional Open Geospatial Consortium (OGC), siendo su sintaxis muy fácil de utilizar, de forma que su uso se encuentra muy generalizado en la industria geo informática (ARCGIS).

Figura N° 2.1
Geometrías primitivas WKT.

Type	Ejemplos	
Punto		<code>POINT (30 10)</code>
Línea		<code>LINESTRING (30 10, 10 30, 40 40)</code>
Polígono		<code>POLYGON ((30 10, 40 40, 20 40, 10 20, 30 10))</code>
		<code>POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))</code>

Según ARCGIS (2013)

2.2.5. PROGRAMACIÓN EXTREMA

Programación extrema (XP), es una metodología ágil para el desarrollo de software y consiste básicamente en ajustarse estrictamente a una serie de reglas que se centran en las necesidades del cliente para lograr un producto de buena calidad en poco tiempo, centrada en potenciar las relaciones interpersonales como clave para el éxito del desarrollo de software (Borja, 2013).

Es una metodología ligera de desarrollo de aplicaciones que se basa en la simplicidad, la comunicación y la realimentación del código desarrollado, teniendo como objetivo la satisfacción del cliente, potenciar el trabajo en grupo y minimizar el riesgo actuando sobre las variables del proyecto: costo, tiempo, calidad, alcance (Meléndez, Gaitán y Pérez, 2016, p.26).

A VALORES XP

A.1. SIMPLICIDAD

La programación extrema, como metodología ágil, apuesta a la sencillez, en su máxima expresión. Sencillez en el diseño, en el código, en los procesos, etc. La sencillez es esencial para que todos puedan entender el código, y se trata de mejorar mediante recodificaciones continuas (Joskowicz, 2008, p.16).

“La simplicidad ayuda a que los desarrolladores de software encuentren soluciones más simples a problemas, según el cliente lo estipula. Los desarrolladores también crean características en el diseño que pudieran ayudar a resolver problemas en un futuro” (Borja, 2013).

A.2. COMUNICACIÓN

Prevalece en todas las prácticas de Extreme Programming. Comunicación cara a cara es la mejor forma de comunicación, entre los desarrolladores y el cliente. Método muy ágil. Gracias a esto el equipo esta pude realizar cambios que al cliente no le gustaron (Borja, 2013).

Muchos de los problemas que existen en proyectos de software (así como en muchos otros ámbitos) se deben a problemas de comunicación entre las personas. La comunicación permanente es fundamental en XP. Dado que la documentación es escasa, el diálogo frontal,

cara a cara, entre desarrolladores, gerentes y el cliente es el medio básico de comunicación. Una buena comunicación tiene que estar presente durante todo el proyecto (Joskowicz, 2008, p.16).

A.3. CORAJE

Cuando se encuentran problemas serios en el diseño, o en cualquier otro aspecto, se debe tener el coraje suficiente como para encarar su solución, sin importar que tan difícil sea. Si es necesario cambiar completamente parte del código, hay que hacerlo, sin importar cuanto tiempo se ha invertido previamente en el mismo (Joskowicz, 2008, p.16).

Muchas de las prácticas implican valentía. Una de ellas es siempre diseñar y programar para hoy y no para mañana. Esto es un esfuerzo para evitar empantanarse en el diseño y requerir demasiado tiempo y trabajo para implementar todo lo demás del proyecto. La valentía le permite a los desarrolladores que se sientan cómodos con reconstruir su código cuando sea necesario. Esto significa revisar el sistema existente y modificarlo si con ello los cambios futuros se implementaran más fácilmente (Bustamante y Rodríguez, 2014, p.06).

A.4. RETROALIMENTACIÓN

La retroalimentación debe funcionar en forma permanente. El cliente debe brindar retroalimentación de las funciones desarrolladas, de manera de poder tomar sus comentarios para la próxima iteración, y para comprender, cada vez más, sus necesidades. Los resultados de las pruebas unitarias son también una retroalimentación permanente que tienen los desarrolladores acerca de la calidad de su trabajo (Joskowicz, 2008, p.16).

A.5. RESPETO

El equipo debe trabajar como uno, sin hacer decisiones repentinas. Extreme Programming promueve el trabajo del equipo. Cada integrante del proyecto (cliente, desarrolladores, etc.) forman parte integral del equipo encargado de desarrollar software de calidad. El equipo debe trabajar como uno, sin hacer decisiones repentinas (Borja, 2013).

B ROLES PROGRAMACIÓN EXTREMA

B.1. PROGRAMADOR

El programador escribe las pruebas unitarias y produce el código del sistema. Debe existir una comunicación y coordinación adecuada entre los programadores y otros

miembros del equipo (Borja, 2013).

Es el responsable de implementar las historias de usuario por el cliente. Además, estima el tiempo de desarrollo de cada historia de usuario para que el cliente pueda asignarle prioridad dentro de la iteración. Cada iteración incorpora nueva funcionalidad de acuerdo a las prioridades establecidas por el cliente. El programador también es responsable de diseñar y ejecutar los test de unidad del código que ha implementado o modificado (Meléndez, et al., 2016, p.31)

B.2. CLIENTE

Determina la funcionalidad que se pretende en cada iteración y define las prioridades de implementación según el valor de negocio que aporta cada historia. El Cliente también es responsable de diseñar y ejecutar los test de aceptación (Meléndez, et al., 2016, p.31).

El cliente escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio. El cliente es sólo uno dentro del proyecto, pero puede corresponder a un interlocutor que está representando a varias personas que se verán afectadas por el sistema (Borja, 2013).

B.3. ENCARGADO DE PRUEBAS (TESTER)

Es el Encargado de ejecutar las pruebas regularmente, difunde los resultados dentro del equipo y es también el responsable de las herramientas de soporte para pruebas (Meléndez, et al., 2016, p.31).

El encargado de pruebas ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas (Borja, 2013).

B.4. ENCARGADO DE SEGUIMIENTO (TRACKER)

El encargado de seguimiento proporciona realimentación al equipo en el proceso XP. Su responsabilidad es verificar el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, comunicando los resultados para mejorar futuras estimaciones. También realiza el seguimiento del progreso de cada iteración y evalúa si los objetivos son

alcanzables con las restricciones de tiempo y recursos presentes (Borja, 2013).

Una de las tareas más importante del tracker, consiste en seguir la evolución de las estimaciones realizadas por los programadores y compararlas con el tiempo real de desarrollo. De esta forma, puede brindar información estadística en lo que refiere a la calidad de las estimaciones para que puedan ser mejoradas (Meléndez, et al., 2016, p.32).

B.5. ENTRENADOR (COACH)

Es responsable del proceso en general. Se encarga de iniciar y de guiar a las personas del equipo en poner en marcha cada una de las prácticas de la metodología XP (Meléndez, et al., 2016, p.32).

Es responsable del proceso global. Es necesario que conozca a fondo el proceso XP para proveer guías a los miembros del equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente (Borja, 2013).

B.6. CONSULTOR

Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto. Guía al equipo para resolver un problema específico (Borja, 2013).

B.7. GESTOR (BIG BOSS)

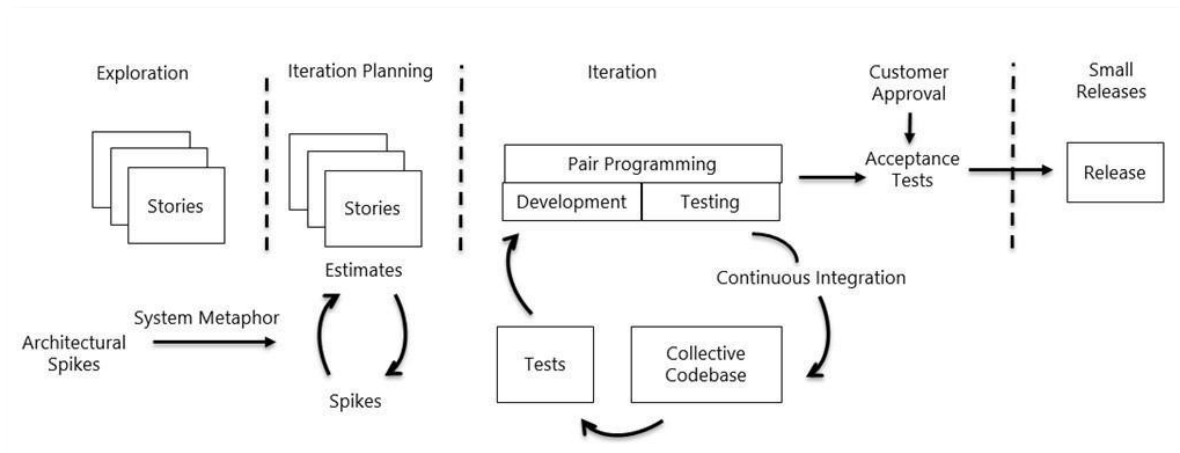
Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación (Borja, 2013).

Experto en tecnología y labores de gestión. Construye el plantel del equipo, obtiene los recursos necesarios y maneja los problemas que se generan. Administra a su vez las reuniones (planes de iteración, agenda de compromisos, etc.). Su labor fundamental es de coordinación (Meléndez, et al., 2016, p.32).

C CICLO DE VIDA DE UN PROYECTO XP

El ciclo de vida del proceso XP se encuentra enmarcado por las fases de exploración, planificación, iteración, producción, mantenimiento y muerte del proyecto.

Figura N° 2.2
Ciclo de vida de programación extrema.



Según Microsoft (2014).

C.1. EXPLORACIÓN

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología (Borja, 2013).

En esta fase, el cliente define lo que necesita mediante la redacción de sencillas “historias de usuarios”. Los programadores estiman los tiempos de desarrollo en base a esta información. Debe quedar claro que las estimaciones realizadas en esta fase son primarias (ya que estarán basadas en datos de muy alto nivel), y podrían variar cuando se analicen más en detalle en cada iteración (Joskowicz, 2008, p.08).

C.2. PLANIFICACIÓN

La planificación es una fase corta, en la que el cliente, los gerentes y el grupo de desarrolladores acuerdan el orden en que deberán implementarse las historias de usuario, y, asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones

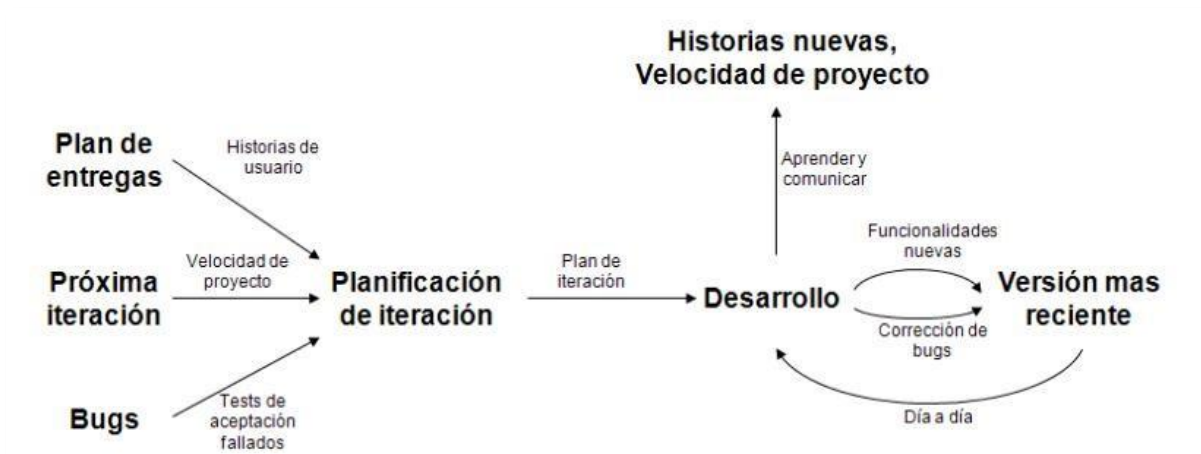
grupales de planificación. El resultado de esta fase es un Plan de Entregas, o “Release Plan” (Joskowicz, 2008, p.09).

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días (Borja, 2013).

C.3. ITERACIONES

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El Plan de Entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias que fueren la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide qué historias se implementarán en cada iteración (para maximizar el valor de negocio). Al final de la última iteración el sistema estará listo para entrar en producción (Borja, 2013).

Figura N° 2.3
Planificación de iteración.



Según Borja (2013).

C.4. PRODUCCIÓN

La fase de producción requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente. Al mismo tiempo, se deben

tomar decisiones sobre la inclusión de nuevas características a la versión actual, debido a cambios durante esta fase (Borja, 2013).

Si bien al final de cada iteración se entregan módulos funcionales y sin errores, puede ser deseable por parte del cliente no poner el sistema en producción hasta tanto no se tenga la funcionalidad completa. En esta fase no se realizan más desarrollos funcionales, pero pueden ser necesarias tareas de ajuste “fine tuning” (Joskowicz, 2008, p.09).

C.5. MANTENIMIENTO

Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Para realizar esto se requiere de tareas de soporte para el cliente. De esta forma, la velocidad de desarrollo puede bajar después de la puesta del sistema en producción. La fase de mantenimiento puede requerir nuevo personal dentro del equipo y cambios en su estructura (Borja, 2013).

C.6. MUERTE DEL PROYECTO

Es cuando el cliente no tiene más historias para ser incluidas en el sistema. Esto requiere que se satisfagan las necesidades del cliente en otros aspectos como rendimiento y confiabilidad del sistema. Se genera la documentación final del sistema y no se realizan más cambios en la arquitectura. La muerte del proyecto también ocurre cuando el sistema no genera los beneficios esperados por el cliente o cuando no hay presupuesto para mantenerlo (Borja, 2013).

D REGLAS Y PRACTICAS

D.1. PLANIFICACIÓN

La metodología XP plantea la planificación como un dialogo continuo entre las partes involucradas en el proyecto, incluyendo al cliente, a los programadores y a los coordinadores o gerentes. El proyecto comienza recopilando “Historias de usuarios”, las que sustituyen a los tradicionales “casos de uso”. Una vez obtenidas las “historias de usuarios”, los programadores evalúan rápidamente el tiempo de desarrollo de cada una (Borja, 2013).

HISTORIAS DE USUARIO

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del

software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales (Borja, 2013).

PLAN DE ENTREGAS (“RELÉASE PLAN”)

El cronograma de entregas establece qué historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas. Este cronograma será el resultado de una reunión entre todos los actores del proyecto: cliente, desarrolladores, gerentes, etc. (Borja, 2013).

PLAN DE ITERACIONES (“ITERATION PLAN”)

Las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido. Al comienzo de cada ciclo, se realiza una reunión de planificación de la iteración. Cada historia de usuario se traduce en tareas específicas de programación. Así mismo, para cada historia de usuario se establecen las pruebas de aceptación (Joskowicz, 2008, p.11).

REUNIONES DIARIAS DE SEGUIMIENTO (“STAND-UP MEETING”)

El objetivo de tener reuniones diarias es mantener la comunicación entre el equipo, y compartir problemas y soluciones. En la mayoría de estas reuniones, gran parte de los participantes simplemente escuchan, sin tener mucho que aportar. Para no quitar tiempo innecesario del equipo, se sugiere realizar estas reuniones en círculo y de pie (Joskowicz, 2008, p.11).

D.2. DISEÑO

La metodología XP hace especial énfasis en los diseños simples y claros. Los conceptos más importantes de diseño en esta metodología son los siguientes:

SIMPLICIDAD

Un diseño simple se implementa más rápidamente que uno complejo. Por ello XP propone implementar el diseño más simple posible que funcione. Se sugiere nunca adelantar la implementación de funcionalidades que no correspondan a la iteración en la que se esté trabajando (Joskowicz, 2008, p.11).

SOLUCIONES “SPIKE”

Cuando aparecen problemas técnicos, o cuando es difícil de estimar el tiempo para implementar una historia de usuario, pueden utilizarse pequeños programas de prueba (llamados “spike”), para explorar diferentes soluciones. Estos programas son únicamente para probar o evaluar una solución, y suelen ser desechados luego de su evaluación (Joskowicz, 2008, p.11).

RECODIFICACIÓN

La recodificación (“refactoring”) consiste en escribir nuevamente parte del código de un programa, sin cambiar su funcionalidad, a los efectos de hacerlo más simple, conciso y/o entendible. Muchas veces, al terminar de escribir un código de programa, pensamos que, si lo comenzáramos de nuevo, lo hubiéramos hecho en forma diferente, más clara y eficientemente (Borja, 2013).

METÁFORAS

Una “metáfora” es algo que todos entienden, sin necesidad de mayores explicaciones. La metodología XP sugiere utilizar este concepto como una manera sencilla de explicar el propósito del proyecto, y guiar la estructura y arquitectura del mismo. Por ejemplo, puede ser una guía para la nomenclatura de los métodos y las clases utilizadas en el diseño del código. Tener nombres claros, que no requieran de mayores explicaciones, redundará en un ahorro de tiempo (Joskowicz, 2008, p.11).

D.3. DESARROLLO

DISPONIBILIDAD DEL CLIENTE

Uno de los requerimientos de XP es tener al cliente disponible durante todo el proyecto. No solamente como apoyo a los desarrolladores, sino formando parte del grupo. El involucramiento del cliente es fundamental para que pueda desarrollarse un proyecto con la metodología XP. Al comienzo del proyecto, el cliente debe proporcionar las historias de usuarios (Borja, 2013).

USO DE ESTÁNDARES

Si bien esto no es una idea nueva, XP promueve la programación basada en estándares, de manera que sea fácilmente entendible por todo el equipo, y que facilite la recodificación (Joskowicz, 2008, p.13).

PROGRAMACIÓN DIRIGIDA POR LAS PRUEBAS (“TEST-DRIVEN PROGRAMMING”)

En las metodologías tradicionales, la fase de pruebas, incluyendo la definición de los tests, es usualmente realizada sobre el final del proyecto, o sobre el final del desarrollo de cada módulo. La metodología XP propone un modelo inverso, en el que, lo primero que se escribe son los test que el sistema debe pasar. Luego, el desarrollo debe ser el mínimo necesario para pasar las pruebas previamente definidas (Joskowicz, 2008, p.13).

PROGRAMACIÓN EN PARES

XP propone que se desarrolle en pares de programadores, ambos trabajando juntos en un mismo ordenador. Si bien parece que ésta práctica duplica el tiempo asignado al proyecto (y por ende, los costos en recursos humanos), al trabajar en pares se minimizan los errores y se logran mejores diseños, compensando la inversión en horas. El producto obtenido es por lo general de mejor calidad que cuando el desarrollo se realiza por programadores individuales (Borja, 2013).

INTEGRACIONES PERMANENTES

Todos los desarrolladores necesitan trabajar siempre con la “última versión”. Realizar cambios o mejoras sobre versiones antiguas causan graves problemas, y retrasan al proyecto. Es por eso que XP promueve publicar lo antes posible las nuevas versiones, aunque no sean las últimas, siempre que estén libres de errores. Idealmente, todos los días deben existir nuevas versiones publicadas. Para evitar errores, solo una pareja de desarrolladores puede integrar su código a la vez (Joskowicz, 2008, p.14).

PROPIEDAD COLECTIVA DEL CÓDIGO

En un proyecto XP, todo el equipo puede contribuir con nuevas ideas que apliquen a cualquier parte del proyecto. Asimismo, cualquier pareja de programadores puede cambiar el código que sea necesario para corregir problemas, agregar funciones o recodificar. En otras metodologías, este concepto puede parecer extraño. Muchas veces se asume que, si hay algo de propiedad colectiva, la responsabilidad también es colectiva. Y que “todos sean responsables”, muchas veces significa que “nadie es responsable” (Borja, 2013).

RITMO SOSTENIDO

Anteriormente, ésta práctica se denominaba “Semana de 40 horas”. Sin embargo, lo

importante no es si se trabajan, 35, 40 o 42 horas por semana. El concepto que se desea establecer con esta práctica es el de planificar el trabajo de manera de mantener un ritmo constante y razonable, sin sobrecargar al equipo (Joskowicz, 2008, p.14).

D.4. PRUEBAS UNITARIAS

Las pruebas unitarias son una de las piedras angulares de XP. Todos los módulos deben de pasar las pruebas unitarias antes de ser liberados o publicados. Las pruebas deben ser definidas antes de realizar el código “Test-driven programming” (Borja, 2013).

DETECCIÓN Y CORRECCIÓN DE ERRORES

Cuando se encuentra un error (“bug”), éste debe ser corregido inmediatamente, y se deben tener precauciones para que errores similares no vuelvan a ocurrir. Asimismo, se generan nuevas pruebas para verificar que el error haya sido resuelto (Joskowicz, 2008, p.15).

PRUEBAS DE ACEPTACIÓN

Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada (Joskowicz, 2008, p.15).

E ARTEFACTOS XP

E.1. HISTORIAS DE USUARIO

Las Historias de Usuario representan una breve descripción del comportamiento del sistema, se realizan por cada característica principal del sistema y son utilizadas para cumplir estimaciones de tiempo y el plan de lanzamientos, así mismo reemplazan un gran documento de requisitos y presiden la creación de las pruebas de aceptación. Cada historia de usuario debe ser lo suficientemente comprensible y delimitada para que los programadores puedan implementarlas en unas semanas. (Meléndez, et al., 2016, p.27).

Figura N° 2.4
Plantilla de historia de usuario

Historia de Usuario	
Número: [Id.]	Nombre Historia de Usuario:
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: [Quien la genera o usa]	Iteración Asignada:
Prioridad en Negocio: (Alta / Media / Baja)	Puntos Estimados: [Semanas de desarrollo, un punto es una semana efectiva de desarrollo]
Riesgo en Desarrollo: (Alto / Medio / Bajo)	Puntos Reales:
Descripción: [Descripción de la tarea o proceso que da lugar a esta historia de usuario]	
Observaciones:	

Según Hanatek (2010).

E.2. TAREAS DE INGENIERÍA (TASK CARD)

Ferreira, E. (citado por Meléndez, et al., 2016) menciona que una historia de usuario se descompone en varias tareas de ingeniería, las cuales describen las actividades que se realizarán en cada historia de usuario, así mismo las tareas de ingeniería se vinculan más al desarrollador, ya que permite tener un acercamiento con el código (p.28).

Figura N° 2.5
Plantilla de tarea de ingeniería.

Tarea de Ingeniería	
Número Tarea: [Id]	Historia de Usuario (No. y Nombre):
Nombre Tarea:	
Tipo de Tarea:	Puntos Estimados:
Fecha Inicio: [Inicio]	Fecha Fin: [Fin]
Programador Responsable: [Nombre del responsable]	
Descripción: [descripción u orden para efectivizar la tarea].	

Según Hanatek (2010).

E.3. TARJETAS CRC (CLASE-RESPONSABILIDAD-COLABORADORES)

Las Tarjetas CRC (Clase-Responsabilidades-Colaboradores), permiten conocer que clases componen el sistema y cuales interactúan entre sí. Se dividen en tres secciones:

Nombre de la Clase, Responsabilidades y Colaboradores (Meléndez, et al., 2016, p.30).

Figura N° 2.6
Plantilla de tarjetas CRC.

Nombre de clase:	
Responsabilidades	Colaboradores
Responsabilidad1()	Colaborador1
Responsabilidad2()	Colaborador2
....	...
ResponsabilidadN()	ColaboradorM

Según Hanatek (2010).

E.4. PRUEBAS DE ACEPTACIÓN

Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada (Joskowicz, 2008, p.15)

2.2.6. PROGRAMACIÓN ORIENTADA A OBJETOS

Durán, Gutiérrez y Pimentel (2007), mencionan que en la programación orientada a objetos el elemento básico de trabajo es el objeto. Un objeto se puede considerar como una representación de un objeto real (por ejemplo, una manzana) o aun no siendo tan real podemos hacernos una idea de que es un elemento existente con ciertas propiedades (por ejemplo, una línea en un dibujo o el estado meteorológico) (p.80)

La orientación a objetos puede describirse como el conjunto de disciplinas que desarrollan y modelan software para facilitar la construcción de sistema complejos a partir de componentes más sencillos. Su atractivo radica en que proporciona conceptos y herramientas con los cuales se modela y representa el mundo tan fielmente como sea posible (Osorio, 2008, p.35).

Bell y Parr (2003), mencionan “un programa se construye a partir de objetos, los que a su vez se crean a partir de clases, esto promueve la modularidad de los programas” (p.546).

La programación orientada a objetos se define como un paradigma que permite realizar una abstracción de la realidad, que se puede implementar en una aplicación de software con el fin de resolver problemas mediante el uso de un lenguaje de programación. El paradigma de orientación a objetos comprende una gran cantidad de conceptos que permite el desarrollo de aplicaciones robustas (Flórez, 2012).

2.2.7. TECNOLÓGICAS DE INTERNET

Internet, también llamado autopista de información, designa un conjunto de redes informáticas relacionadas entre sí, para permitir que los usuarios puedan comunicarse entre sí; es una red abierta. Su principio básico es la transmisión de datos de manera fiable entre ordenadores (Colección Esencial, 2011).

Luján (2001), asevera que contrario a otros servicios online, que se controlan de forma centralizada, la Internet posee un diseño descentralizado. Dado que cada ordenador (host) en la Internet es independiente. Los operadores pueden elegir qué servicio usar y qué servicios locales proporcionar.

Cafassi (1998) asevera que Internet es una tecnología. Sin embargo, no es la acepción más correcta y su consecuencia se ven a la hora de analizar las implicaciones sociales que conlleva su uso.

A APLICACIÓN WEB

Para Luján (2001), una aplicación web es un tipo especial de aplicación cliente / servidor, donde el cliente (el navegador, explorador o visualizador) como el servidor (el servidor web) y el protocolo mediante el que se comunican (HTTP) están estandarizados y no han de ser creados por el programador de aplicaciones.

Una aplicación web generalmente se compone de los siguientes elementos: a) Recursos estáticos: paginas HTML, imágenes, sonidos, hojas de estilo, etc., b) Recursos dinámicos: servlets, JSP, Java Bean, c) Librerías de clases y d) Descriptor de despliegue para definir los parámetros de funcionamiento de la aplicación en el servidor (Groussard, 2010).

B PROTOCOLO

“Proviene del griego <<protókolon>>, que significa <<la primera hoja o tapa,

encolada, de un manuscrito importante con notas sobre su contenido>>” (Antonio, 2001).

Según Gutiérrez y Tena (2003), un protocolo es un conjunto definido de etapas, que implica a dos o más partes, designado para realizar una tarea específica.

C HTTP

Luján (2001), el protocolo forma parte de la familia de protocolos de comunicación TCP / IP, que son los empleados en internet. Permiten la conexión de sistemas heterogéneos, facilitando el intercambio de información ente distintos ordenadores.

“El protocolo de transferencia de hipertexto es un sencillo protocolo cliente–servidor que articula los intercambios de información entre los clientes Web y los servidores HTTP” (Romero, 1997, p. 203).

Para Kroenke (2003), hay dos características importantes en ese protocolo: a) Orientado a peticiones, pues los servidores HTTP esperan peticiones para realizar alguna acción y generar una respuesta y b) No mantienen un estado, ya que reciben una pregunta, la procesan y luego olvidan cuál cliente la formuló.

D TRANSMISSION CONTROL PROTOCOL (TCP)

Según Colección Esencial (2011), este protocolo, descompone el mensaje en paquetes y asegura la fiabilidad de la transmisión. Cuando los paquetes llegan a su destino, se agrupan automáticamente para formar un único mensaje, idéntico al original.

TCP utiliza mensajes IP para lograr una transferencia de datos libre de errores. Ambos establecen un diálogo con otro sistema a base de enviar servicios de mensajes IP (Romero, 1997, p.22).

E INTERNET PROTOCOL (IP)

En el protocolo de internet, cada equipo posee una sola dirección única en la red a la que pertenece y cada red posee una dirección única en internet. Por lo tanto, cada ordenador tiene asignado un nombre y un número IP, y en algunos casos un nombre de dominio (Colección Esencial, 2011).

“IP es capaz de enviar mensajes de pequeño tamaño (denominado datagrama) entre dos ordenadores conectados en red. No ofrece garantías de que los mensajes alcancen su destino, debido a los posibles fallos de las redes de comunicaciones.” (Romero, 1997, p.20).

F ARQUITECTURA EN TRES CAPAS

Para Sommerville (2005), asevera que, en esta arquitectura, la presentación, el procesamiento de la aplicación y la gestión de los datos son procesos lógicamente separados que se ejecutan obre procesadores diferentes.

“Los métodos utilizados para conectar los servidores Web con los SGBD hicieron que surgiese la arquitectura de tres capas, siendo las siguientes: a) Capa visible para el usuario, b) Capa intermedia y c) Capa subyacente” (Ramos, A. y Ramos, J., 2007, p.179).

CAPÍTULO III

METODOLOGÍA DE LA INVESTIGACIÓN

3.1 TIPO DE INVESTIGACIÓN

La investigación es del tipo observacional, porque no existe intervención en los incidentes delictivos, solo se observa su comportamiento.

La investigación es del tipo retrospectiva, porque se analizan y estudian los registros de incidentes delictivos ocurridos en el pasado, los registros fueron recopilados por personas ajenas a la investigación.

La investigación es longitudinal, porque se hace mediciones repetidas sobre la variable de estudio, todos los incidentes delictivos durante el 2017.

Finalmente, la investigación es del tipo descriptiva, por ser una investigación de una sola variable de interés y tener como fin el describir parámetros en la población de estudio.

3.2 NIVEL DE INVESTIGACIÓN

La investigación es de nivel descriptiva, porque busca describir la actividad delictiva en una circunstancia temporal (2017) y geográfica (distrito de Jesús Nazareno y Carmen Alto) obteniendo como resultado el mapa del delito.

3.3 DISEÑO DE INVESTIGACIÓN

Se recolectaron los datos mediante el análisis de los documentos que se tiene los registros de la actividad delictiva en los distritos de Jesús Nazareno y Carmen Alto, los datos recolectados pertenecen a los incidentes u ocurrencias de los meses de enero a diciembre del 2017, estos datos se procesaran mediante la estadística descriptiva presentándolos en mapas, gráficos y cuadros.

3.4 POBLACIÓN Y MUESTRA DE LA INVESTIGACIÓN

3.4.1 POBLACIÓN

La población de estudio estuvo conformada por todos los incidentes delictivos registrados en los distritos de Jesús Nazareno y Carmen Alto de la ciudad de Ayacucho del

año 2017.

3.4.2 MUESTRA

No existe muestra, se ha realizado es un censo, porque se estudió todos los incidentes delictivos registrados en los distritos de Jesús Nazareno y Carmen Alto de la ciudad de Ayacucho.

3.5 VARIABLES E INDICADORES DE LA INVESTIGACIÓN

3.5.1 DEFINICIÓN CONCEPTUAL DE LAS VARIABLES

VARIABLE DE INTERÉS

ACTIVIDAD DELICTIVA

La actividad delictiva es una conducta típica, antijurídica y culpable que se adecua al presupuesto jurídico de una ley penal, en donde solo una acción u omisión puede ser típica, sólo una acción u omisión típica puede ser antijurídica y sólo una acción u omisión antijurídica puede ser culpable.

INDICADORES DE LA VARIABLE DE INTERÉS

TIPICIDAD

La tipicidad es la verificación de si una actividad realizada coincide con lo descrito en la ley penal, todo acto humano para considerarse como delito debe adecuarse al tipo penal.

ANTI JURIDICIDAD

La antijuridicidad es el acto voluntario típico que contraviene el presupuesto de la norma penal, lesionando o poniendo en peligro bienes e intereses tutelados por el derecho, es un juicio de valor objetivo, en tanto se pronuncia sobre la conducta típica.

CULPABILIDAD

La culpabilidad es la reprochabilidad jurídica de la conducta típica y antijurídica de una persona imputable, por no haber hecho lo que debía hacer, cuando sabía que estaba haciendo algo distinto de lo obligado por el mandato o lo prohibido por ley.

3.5.2 DEFINICIÓN OPERACIONAL DE LAS VARIABLES

VARIABLE DE INTERÉS

X: Actividad delictiva

INDICADORES DE LA VARIABLE DE INTERÉS

X1: Tipicidad

X2: Antijuridicidad

X3: Culpabilidad

3.6 OPERACIONALIZACIÓN DE LA VARIABLE

Se muestra en el anexo 1.

3.7 TÉCNICA E INSTRUMENTO PARA RECOLECTAR INFORMACIÓN

3.7.1 TÉCNICA PARA RECOLECTAR INFORMACIÓN

Se aplicó la técnica de análisis documental a los registros de los incidentes ocurridos en los distritos de Jesús Nazareno y Carmen Alto.

3.7.2 INSTRUMENTO PARA RECOLECTAR INFORMACIÓN


Se usó el instrumento de registro para levantar los datos de los incidentes u ocurrencias, el instrumento se muestra en el anexo 2.





3.8 HERRAMIENTA PARA EL TRATAMIENTO DE DATOS E INFORMACIÓN




Las herramientas tecnológicas que se utilizaron fueron seleccionadas de acuerdo a los requerimientos del sistema geográfico, también, se consideró si la herramienta era open source y su capacidad multiplataforma.


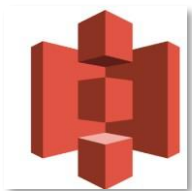

Tabla N° 3.1




Herramientas para el tratamiento de datos e información



SOFTWARE	FABRICANTE	DESCRIPCIÓN
Kde Neon 5.15 	KDE neon.kde.org	KDE neon es una distribución Linux y un sistema operativo derivado de Ubuntu, pero centrado en el desarrollo de software para la comunidad de código abierto KDE.
Red Hat 7.6	Red Hat	Red Hat Enterprise Linux también conocido por sus



	www.redhat.com/rhel/	<p>siglas RHEL es una distribución comercial de GNU/Linux desarrollada por Red Hat. Es la versión comercial basada en Fedora que a su vez está basada en el anterior Red Hat Linux, de forma similar a como SUSE Enterprise (SUSE Linux Enterprise Desktop y SLE Server) lo es respecto de OpenSUSE o Mandriva Corporate respecto de Mandriva Linux One.</p>
Pycharm 	JetBrains	<p>Pycharm es un entorno de desarrollo integrado (IDE) utilizado específicamente para la programación en el lenguaje Python. Está desarrollado por la compañía checa JetBrains. Proporciona análisis de código, un depurador gráfico, un comprobador de unidades integrado, integración con sistemas de control de versiones y es compatible con el desarrollo web con Django.</p>
WebStorm 	JetBrains	<p>WebStorm es un entorno de desarrollo integrado (IDE) utilizado en la programación de computadoras, específicamente para los lenguajes de JavaScript y frameworks derivados de NodeJs. Está desarrollado por la compañía checa JetBrains. Proporciona análisis de código, un depurador gráfico, un comprobador de unidades integrado, integración con sistemas de control de versiones y es compatible con el desarrollo web con Angular, Angularjs, React y React Native.</p>
Visual Studio Code 	Microsoft	<p>Visual Studio Code (vscode) 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. Es gratuito y de código abierto, aunque la descarga oficial está bajo software propietario.</p>
DBaver	Open-source	<p>DBeaver es un cliente SQL y una herramienta de</p>





	community	<p>administración de base de datos. Para las bases de datos relacionales, utiliza la API de JDBC para interactuar con las bases de datos a través de un controlador JDBC. Para otras bases de datos (NoSQL) utiliza controladores de base de datos propietarios. Proporciona una arquitectura de complementos (basada en la arquitectura de complementos de Eclipse) que permite a los usuarios modificar gran parte del comportamiento de la aplicación para proporcionar funcionalidades o funciones específicas de la base de datos que son independientes de la base de datos. Esta es una aplicación de escritorio escrita en Java y basada en la plataforma Eclipse.</p>
<p>Dia</p> 	Proyecto GNU	<p>Dia es una aplicación informática de propósito general para la creación de diagramas, creada originalmente por Alexander Larsson, y desarrollada como parte del proyecto GNOME. Está concebido de forma modular, con diferentes paquetes de formas para diferentes necesidades.</p>
<p>Postman</p> 	Postdot Technologies Pvt. Ltd.	<p>Postman es un ApiTest, es un de software que implica probar las interfazs de programación de aplicaciones (API) directamente y como parte de las pruebas de integración para determinar si cumplen con las expectativas de funcionalidad, confiabilidad, rendimiento y seguridad. Como las API carecen de una GUI, las pruebas de API se realizan en la capa de mensajes.</p>

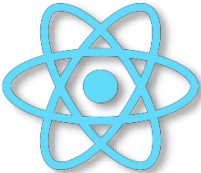


<p>Amazon EC2</p> 	<p>Amazon</p>	<p>Amazon Elastic Compute Cloud (Amazon EC2) es una parte central de la plataforma de cómputo en la nube de la empresa Amazon.com denominada Amazon Web Services (AWS). EC2 permite a los usuarios alquilar computadores virtuales en los cuales pueden ejecutar sus propias aplicaciones. Este tipo de servicio supone un cambio en el modelo informático al proporcionar capacidad informática con tamaño modificable en la nube, pagando por la capacidad utilizada. En lugar de comprar o alquilar un determinado procesador para utilizarlo varios meses o años, en EC2 se alquila la capacidad por horas.</p>
<p>Amazon S3</p> 	<p>Amazon</p>	<p>Amazon S3 o Amazon Simple Storage Service es un "servicio de almacenamiento simple" ofrecido por Amazon Web Services (AWS) que proporciona almacenamiento de objetos a través de una interfaz de servicio web. Amazon S3 usa la misma infraestructura de almacenamiento escalable que Amazon.com utiliza para ejecutar su red global de comercio electrónico.</p> <p>Amazon S3 puede emplearse para almacenar cualquier tipo de objeto que permita usos como almacenamiento para aplicaciones de Internet, respaldo y recuperación, recuperación de desastres, archivos de datos, lagos de datos para análisis y almacenamiento en la nube híbrida.</p>
<p>Amazon RDS</p> 	<p>Amazon</p>	<p>El servicio de base de datos relacional de Amazon (o Amazon RDS) es un servicio de base de datos relacional distribuido por Amazon Web Services (AWS). Es un servicio web que se ejecuta "en la nube" diseñado para simplificar la configuración, el funcionamiento y el escalamiento de una base de datos relacional para su uso en aplicaciones. Los procesos de administración, como parchear el software de la base de datos, realizar copias</p>




		de seguridad de las bases de datos y habilitar la recuperación en un momento dado se administran automáticamente. Los recursos de cálculo y almacenamiento de escala pueden realizarse mediante una sola llamada a la API, ya que AWS no ofrece una conexión ssh a instancias de RDS.
Amazon Elastic Beanstalk 	Amazon	AWS Elastic Beanstalk es un PaaS (Plataforma como Servicio) servicio de Amazon Web Services que permite crear aplicaciones y desplegarlas a un conjunto definido de Servicios de AWS, incluyendo Amazon EC2, Amazon S3, Amazon Simple Notification Service (SNS), Amazon CloudWatch, auto escalamiento, y balanceador de carga elásticos.
PostgreSQL 	PostgreSQL Global Development Group	<p>PostgreSQL es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto, publicado bajo la licencia PostgreSQL, similar a la BSD o la MIT.</p> <p>Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una empresa o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, altruista, libre o apoyados por organizaciones comerciales. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group)</p>
PostGis 	Refractions Research	PostGIS convierte al sistema de administración de bases de datos PostgreSQL en una base de datos espacial mediante la adición de tres características: tipos de datos espaciales, índices espaciales y funciones que operan sobre ellos. Debido a que está construido sobre PostgreSQL, PostGIS hereda automáticamente las

		características de las bases de datos empresariales, así como los estándares abiertos que implementan un Sistema de Información Geográfica dentro del motor de base de datos.
<p>Apache</p> 	<p>Apache Software Foundation</p>	<p>El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual según la normativa RFC 2616.</p>
<p>Python 3.6</p> 	<p>Python Software Foundation</p>	<p>Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.</p> <p>Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma.</p>
<p>Django</p> 	<p>Django Software Foundation</p>	<p>Django es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como Modelo–vista–template. Fue desarrollado en origen para gestionar varias páginas orientadas a noticias de la World Company de Lawrence, Kansas, y fue liberada al público bajo una licencia BSD en julio de 2005. En junio de 2008 fue anunciado que la recién formada Django Software Foundation se haría cargo de Django en el futuro.</p>
<p>Geo Django</p> 	<p>Django Software Foundation</p>	<p>GeoDjango es un módulo contribuido por Django que lo transforma en un sistema geográfico web a gran escala. El objetivo de GeoDjango es que sea lo más fácil posible crear aplicaciones web basadas en información geográfica, como los servicios basados en la ubicación. Sus características incluyen:</p>

		<p>Los campos de la plantilla de Django para OGC objetos geométricos rz datos raster.</p> <p>Extensiones de Django ORM para consultas y manipulaciones de datos espaciales.</p> <p>Las interfaces de Python están ligeramente vinculadas y son de alto nivel para operaciones geométricas de SIG y matrices, y para la manipulación de datos en varios formatos.</p>
Django Rest Framework 	Tom Christie.	<p>Django REST framework (DRF) es una de las apps de tercero más usadas en Django para la liberación de una API REST.</p> <p>Recientemente se ha liberado la versión 3 de esta librería soportada principalmente por Tom Christie. Esta versión incluye muchas novedades y suponen un salto de calidad y de madurez en la herramienta.</p>
GeoJson	geojson.org	<p>GeoJSON Es un formato estándar abierto diseñado para representar elementos geográficos sencillos, junto con sus atributos no espaciales, basado en JavaScript Object Notation (JSON). El formato es ampliamente utilizado en aplicaciones de cartografía en entornos web al permitir el intercambio de datos de manera rápida, ligera y sencilla.</p>
JavaScript 	Netscape Communications Corp	<p>JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.</p> <p>Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.</p>
ECMAScript 6	Sun Microsystems,	<p>ECMAScript es una especificación de lenguaje de programación publicada por ECMA International. El</p>

	Ecma International	<p>desarrollo empezó en 1996 y estuvo basado en el popular lenguaje JavaScript propuesto como estándar por Netscape Communications Corporation. Actualmente está aceptado como el estándar ISO 16262.</p> <p>ECMAScript define un lenguaje de tipos dinámicos ligeramente inspirado en Java y otros lenguajes del estilo de C. Soporta algunas características de la programación orientada a objetos mediante objetos basados en prototipos y pseudoclases.</p>
<p>Css</p> 	CSS Working Group	<p>CSS (Cascading Style Sheets - Hojas de estilo en cascada), es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web, e interfazs de usuario escritas en HTML o XHTML; el lenguaje puede ser aplicado a cualquier documento XML, incluyendo XHTML, SVG, XUL, RSS, etcétera. También permite aplicar estilos no visuales, como las hojas de estilo auditivas.</p>
<p>NPM</p> 	npm, Inc.	<p>npm es el manejador de paquetes por defecto para Node.js, un entorno de ejecución para JavaScript.</p>
<p>WebPack</p> 	obias Koppers, Sean Larkin, Johannes Ewald, Juho Vepsäläinen, Kees Kluskens and Webpack contributors	<p>Webpack (webpack estilizado) es un paquete de código abierto de JavaScript. Su objetivo principal es agrupar los archivos JavaScript para su uso en un navegador, pero también es capaz de transformar, agrupar o empaquetar casi cualquier recurso o activo. Webpack toma módulos con dependencias y genera activos estáticos que representan esos módulos. Es un agrupador de módulos principalmente para JavaScript, pero puede</p>

		transformar activos front-end como HTML, CSS, incluso imágenes si se incluyen los complementos correspondientes.
Reactjs 	Facebook Inc.	<p>React (ReactJS) es una biblioteca Javascript de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página (SPA). Es mantenido por Facebook y la comunidad de software libre, han participado en el proyecto más de mil desarrolladores diferentes.</p> <p>React intenta ayudar a los desarrolladores a construir aplicaciones que usan datos que cambian todo el tiempo. Su objetivo es ser sencillo, declarativo y fácil de combinar.</p>
Redux 	Dan Abramov and Andrew Clark	<p>Redux es una biblioteca de código abierto de JavaScript para administrar el estado de la aplicación. Se usa más comúnmente con bibliotecas como React o Angular para crear interfaces de usuario. Similar a (e inspirado por) la arquitectura Flux de Facebook.</p>
Leaflet 	Vladimir Agafonkin	<p>Leaflet es una biblioteca de código fuente abierto de JavaScript que se utiliza para crear aplicaciones de mapas web. Lanzado por primera vez en 2011, es compatible con la mayoría de las plataformas móviles y de escritorio, y admite HTML5 y CSS3. Junto con OpenLayers y la API de Google Maps, es una de las bibliotecas de mapas de JavaScript más populares y es utilizada por los principales sitios web como FourSquare, Pinterest y Flickr.</p>
MapBox	Eric Gundersen	<p>Mapbox es un proveedor de mapas en línea, realizado por encargo para páginas web como Foursquare, Pinterest, Evernote, Financial Times, EThe Weather Channel y Uber Technologies. Desde 2010, he expandido rápidamente su nicho de mapas.</p>

		
D3 	Mike Bostock y Jeffrey Heer	D3.js (Data-Driven Documents) es una biblioteca de JavaScript para producir, a partir de datos, infogramas dinámicos e interactivos en navegadores web. Hace uso de tecnologías bien sustentadas como SVG, HTML5, y CSS. Esta biblioteca es sucesora de la biblioteca Protovis. En contraste con muchas otras bibliotecas, D3.js permite tener control completo sobre el resultado visual final.
ChartJs 	Chart.js	Charts.js es una librería de JavaScript que utiliza el canvas de HTML5 para mostrar gráficos web. Chart está bajo la licencia MIT license.

Elaboración propia.

3.9 TÉCNICAS PARA APLICAR PROGRAMACIÓN EXTREMA

Siguiendo el ciclo de vida de un proyecto XP, iniciamos con la fase de Exploración, seguido de la fase de Planificación, Iteración, Producción, Mantenimiento y Muerte del proyecto. En cada fase se utilizan artefactos que la metodología recomienda.

Tabla N° 3.2
Artefactos de la fase de observación.

TAREA	ARTEFACTO	TÉCNICA	RESPONSABLE
Escribir historia de usuario	Historia de usuario	<ul style="list-style-type: none"> · Describir brevemente las necesidades del usuario (con sus propias palabras). · Dividir grandes historias de usuario en pequeños. 	Cliente

Estimar esfuerzo para las historias de usuario	Plan de alto nivel	<ul style="list-style-type: none"> Conocer las historias de usuario Estimar el esfuerzo (días) para desarrollar la historia de usuario 	Cliente Equipo de desarrollo
Probar las tecnologías a utilizar	Arquitectura técnica inicial	<ul style="list-style-type: none"> Explorar las tecnologías que cubran las necesidades técnicas del proyecto Probar el rendimiento de las tecnologías Definir la arquitectura inicial 	Cliente Equipo de desarrollo

Elaboración propia.

Tabla N° 3.3

Artefactos de la fase de planificación.

TAREA	ARTEFACTO	TÉCNICA	RESPONSABLE
Formular el plan de entregas	Plan de entrega	<ul style="list-style-type: none"> Establecer la prioridad de cada historia de usuario. Realizar una estimación de esfuerzo de cada historia de usuario. Tomar acuerdos sobre la primera entrega y el cronograma de entregas. 	Cliente Equipo de desarrollo

Elaboración propia.

Tabla N° 3.4

Artefactos de la fase de iteración.

TAREA	ARTEFACTO	TÉCNICA	RESPONSABLE
Definir la arquitectura técnica	Arquitectura técnica	<ul style="list-style-type: none"> Actualizar la arquitectura técnica inicial Integrar las tecnologías y frameworks a la arquitectura técnica. Representar la arquitectura técnica en el diagrama despliegue. 	Equipo de desarrollo
Formular el plan de iteraciones	Plan de iteración	<ul style="list-style-type: none"> Estimar y asignar esfuerzo para implementar cada tarea de ingeniería 	Cliente Equipo de desarrollo

		<ul style="list-style-type: none"> Asignar las tareas de ingeniería a cada iteración. Utilizar el plan de versión. Dividir cada historia de usuario en tareas de ingeniería 	
Escribir tareas de ingeniería	Tarea de ingeniería	<ul style="list-style-type: none"> Describir cada tarea de ingeniería (descripción técnica). Asignar las tareas de ingeniería a cada desarrollador. 	Equipo de desarrollo
Implementar interfaz	Interfaz de usuario (GUI)	<ul style="list-style-type: none"> Diseñar las interfaces de usuario relacionadas a las historias de usuario. 	Cliente Equipo de desarrollo
Escribir las tarjetas CRC	Tarjeta CRC	<ul style="list-style-type: none"> Diseñar una tarjeta de crc de forma simple Identificar al responsable de la tarjeta Identificar colaboradores 	Equipo de desarrollo
Implementar la base de datos física	Base de datos física	<ul style="list-style-type: none"> Implementar la base de datos a partir del diagrama lógico de base de datos 	Equipo de desarrollo
Implementar pruebas unitarias	Reporte pruebas unitarias	<ul style="list-style-type: none"> Implementar las pruebas unitarias. 	Equipo de desarrollo
Implementar pruebas de integración	Reporte de pruebas de integración	<ul style="list-style-type: none"> Integrar continuamente al concluir las iteraciones. Verificar que la integración es correcta con la última iteración. 	Cliente Equipo de desarrollo
Implementar pruebas de aceptación	Reporte de pruebas de aceptación	<ul style="list-style-type: none"> Verificar con el cliente que las pruebas de aceptación sean satisfactorias. 	Cliente Equipo de desarrollo

Elaboración propia.

CAPÍTULO IV

ANÁLISIS Y RESULTADOS DE LA INVESTIGACIÓN

4.1 ARTEFACTOS DE SOFTWARE APLICANDO PROGRAMACIÓN EXTREMA

4.1.1 FASE DE EXPLORACIÓN

A. HISTORIAS DE USUARIO

Tabla N° 4.1
Historias de usuarios.

N.º	HISTORIA DE USUARIO	DESCRIPCIÓN CORTA
01	Acceso al sistema	El usuario (sereno, administrador y personal administrativo) podrá acceder al sistema.
02	Registro de permisos	El usuario administrador podrá registrar los permisos de los usuarios de su sub gerencia de seguridad ciudadana.
03	Gestión de permisos	El usuario administrador podrá actualizar los permisos de los usuarios de su sub gerencia de seguridad ciudadana.
04	Registro de usuarios	El usuario administrador podrá registrar nuevos usuarios (sereno, personal administrativo) que pertenecen a su sub gerencia de seguridad ciudadana, validando que los usuarios no se repitan.
05	Gestión de usuarios	El usuario (administrador, sereno, personal administrativo) podrá actualizar los datos de su cuenta El usuario administrador podrá actualizar los datos de los usuarios (sereno, personal administrativo) que pertenecen a su sub gerencia de seguridad ciudadana.
06	Registro de marco legal	El usuario root podrá registrar los tipos de delito provenientes del código penal, los delitos registrarán a todas las sub gerencias seguridad ciudadana y mantendrán una relación con los incidentes y sub gerencias registradas.
07	Gestión del marco legal	El usuario root podrá actualizar los tipos de delito provenientes del código penal, los delitos registrarán a todas las sub gerencias seguridad ciudadana.
08	Registro de sub gerencias	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá registrar a la sub gerencia al sistema.

09	Gestión de sub gerencias	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá actualizar los datos de la sub gerencia al sistema.
10	Registro de sectores y sub sectores	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá registrar los sectores y sub sectores de su jurisdicción.
11	Gestión de sectores y sub sectores	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá actualizar los datos de los sectores y sub sectores de su jurisdicción.
12	Registro de puestos	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá registrar los puestos de control que cuenta en su jurisdicción.
13	Gestión de puestos	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá actualizar los datos de los puestos de control que cuenta en su jurisdicción.
14	Registro de turnos	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá registrar los turnos de patrullaje.
15	Gestión de turnos	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá actualizar los datos de los turnos de patrullaje.
16	Registro de cargos	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá registrar los cargos que cuenta su sub gerencia.
17	Gestión de cargos	El usuario root en coordinación de la sub gerencia de seguridad ciudadana podrá actualizar los datos de los cargos que cuenta su sub gerencia.
18	Registro de incidentes	El usuario administrador y sereno podrá registrar los incidentes delictivos que se presenten en su jurisdicción.
19	Gestión de incidentes	El usuario administrador y sereno podrán actualizar datos de los incidentes delictivos que se presenten en su jurisdicción.
20	Gestión mapa de incidentes	El usuario administrador podrá hacer consultas en el mapa del delito que el sistema genera.
21	Registro de base patrullaje	El usuario administrador podrá registrar las bases para el patrullaje que realiza la sub gerencia.
22	Gestión de base patrullaje	El usuario administrador podrá actualizar los datos de las bases de patrullaje que realiza la sub gerencia.
23	Gestión de tipos de delito	El usuario administrador podrá hacer consultas sobre los tipos de delito, tipicidad y culpabilidad que se presentan en su jurisdicción.

Elaboración propia.

B. PLAN DE ALTO NIVEL

Tabla N° 4.2
Plan de alto nivel

N.º	HISTORIA DE USUARIO	ESTIMACIÓN
01	Acceso al sistema	3
02	Registro de permisos	3
03	Gestión de permisos	4
04	Registro de usuarios	4
05	Gestión de usuarios	3
06	Registro de marco legal	3
07	Gestión del marco legal	3
08	Registro de sub gerencias	3
09	Gestión de sub gerencias	3
10	Registro de sectores y sub sectores	3
11	Gestión de sectores y sub sectores	4
12	Registro de puestos	3
13	Gestión de puestos	3
14	Registro de turnos	3
15	Gestión de turnos	4
16	Registro de cargos	3
17	Gestión de cargos	3
18	Registro de incidentes	5
19	Gestión de incidentes	3
20	Gestión mapa de incidentes	6
21	Registro de base patrullaje	6

22	Gestión de base patrullaje	4
23	Gestión de tipos de delito	7

Elaboración propia.

C. HISTORIAS DE USUARIO EN DETALLE

Tabla N° 4.3

Historia de usuario: Acceso al sistema.

HISTORIA DE USUARIO	
Número: 01	Nombre: Acceso al sistema
Modificación o extensión de la h.u. : Registro de usuarios (04)	
Usuario: sereno, personal administrativo y administrador	Iteración asignada : 03
Prioridad en el negocio : Alto	Puntos estimados (días) : 04
Riesgo en desarrollo : Alto	Puntos reales (días): 03
Descripción: El usuario (sereno, administrador, personal administrativo) ingresa su nombre de usuario y contraseña, si los datos son correctos, el sistema permite el ingreso al usuario. Si los datos son incorrectos, el sistema informa que ocurrió un error al validar los datos.	
Observaciones: Solo los usuarios registrados y activos podrán ingresar al sistema. Cada usuario necesariamente debe estar adherido a una sub gerencia de seguridad ciudadana.	

Elaboración propia.

Tabla N° 4.4

Historia de usuario: Registro de permisos.

HISTORIA DE USUARIO	
Número: 02	Nombre: Registro de permisos
Modificación o extensión de la h.u. : Registro de cargos (16), Registro de usuarios (04)	

Usuario : administrador	Iteración asignada : 03
Prioridad en el negocio : Alto	Puntos estimados (días): 03
Riesgo en desarrollo : Alto	Puntos reales (días): 04
Descripción: El usuario administrador podrá registrar los permisos de los usuarios (sereno y personal administrativo) según su cargo. El sistema gestionará los accesos en las dos capas (frontend y backend).	
Observaciones: Cada usuario administrador solo tiene jurisdicción en su sub gerencia.	

Elaboración propia.

Tabla N° 4.5

Historia de usuario: Gestión de permisos.

HISTORIA DE USUARIO	
Número: 03	Nombre: Gestión de permisos
Modificación o extensión de la h.u. : Registro de permisos(02)	
Usuario : administrador	Iteración asignada : 03
Prioridad en el negocio : Medio	Puntos estimados (días): 04
Riesgo en desarrollo : Medio	Puntos reales (días): 05
Descripción: El usuario administrador podrá actualizar los permisos de los usuarios (sereno y personal administrativo) según su cargo. El sistema gestionará los accesos en las dos capas (frontend y backend).	
Observaciones: Cada usuario administrador solo tiene jurisdicción en su sub gerencia.	

Elaboración propia.

Tabla N° 4.6

Historia de usuario: Registro de usuarios.

HISTORIA DE USUARIO	
Número: 04	Nombre: Registro de usuarios
Modificación o extensión de la h.u. : Registro de sub gerencias (10)	
Usuario : administrador	Iteración asignada : 03
Prioridad en el negocio : Alto	Puntos estimados (días): 04
Riesgo en desarrollo : Alto	Puntos reales (días): 05
Descripción: El usuario administrador podrá registrar a los usuarios (sereno, personal administrativo) que sean parte de su sub gerencia de seguridad ciudadana. Los datos que se guardan son el nombre, apellidos, sexo, fecha de nacimiento, cargo (importante), control de acceso (permiso).	
Observaciones: Al registrar un usuario el sistema asignara la sub gerencia que pertenece por medio de la sesión del usuario administrador.	

Elaboración propia.

Tabla N° 4.7

Historia de usuario: Gestión de usuarios.

HISTORIA DE USUARIO	
Número: 05	Nombre: Gestión de usuarios
Modificación o extensión de la h.u. : Registro de usuarios (04)	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 03
Prioridad en el negocio : Medio	Puntos estimados (días): 03
Riesgo en desarrollo : Medio	Puntos reales (días): 03
Descripción: El usuario (administrador, sereno, personal administrativo) podrá acceder a su perfil y modificar los datos básicos de su cuenta, datos como el: nombre, apellidos, descripción, fecha de nacimiento, preferencias.	

El usuario administrador, podrá modificar los datos sensibles de los usuarios (sereno y personal administrativo), los datos sensibles son: la contraseña, documento de identidad, tipo de documento de identidad, nombre de usuario, alias.

Observaciones:

Cada usuario administrador solo tiene jurisdicción en su sub gerencia.

Elaboración propia.

Tabla N° 4.8

Historia de usuario: Registro de marco legal.

HISTORIA DE USUARIO	
Número: 06	Nombre: Registro de marco legal
Modificación o extensión de la h.u. : ninguno	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 02
Prioridad en el negocio : Alto	Puntos estimados (días): 03
Riesgo en desarrollo : Alto	Puntos reales (días): 05
Descripción: El usuario root podrá registrar los tipos de delito provenientes del código penal, los delitos serán categorizados según los capítulos, artículos, su vigencia y si es un delito o falta.	
Observaciones: El tipo de delito registrado será el marco de referencia para todas las sub gerencias.	

Elaboración propia.

Tabla N° 4.9

Historia de usuario: Gestión del marco legal.

HISTORIA DE USUARIO	
Número: 07	Nombre: Gestión del marco legal
Modificación o extensión de la h.u. : Registro de marco legal (06)	
Usuario : sereno, personal administrativo y	Iteración asignada : 02

administrador	
Prioridad en el negocio : Medio	Puntos estimados (días): 03
Riesgo en desarrollo : Medio	Puntos reales (días): 04
Descripción: El usuario root podrá actualizar los tipos de delito provenientes del código penal, los delitos podrán ser eliminados siempre y cuando no tengan incidentes asociados. De ser derogada la ley, solo se ocultara para mantener persistencia con los incidentes delictivos.	
Observaciones : Ninguno	

Elaboración propia.

Tabla N° 4.10

Historia de usuario: Registro de sub gerencias.

HISTORIA DE USUARIO	
Número: 08	Nombre: Registro de sub gerencias
Modificación o extensión de la h.u. : ninguno	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 01
Prioridad en el negocio : Alto	Puntos estimados (días): 03
Riesgo en desarrollo : Alto	Puntos reales (días): 04
Descripción: El usuario root podrá registrar las sub gerencias de seguridad ciudadana, al registrar una sub gerencia se ingresarán los datos de la sub gerencia y su logo. Es usuario root deberá controlar la delimitación geográfica de las sub gerencias, la delimitación geografía deberá ser un polígono que servirá como control para el registro de incidentes, patrullaje y ocurrencias.	
Observaciones: La delimitación geográfica de la sub gerencia deberá ser un PolygonField. La delimitación geográfica de la sub gerencia no deberá intersectar con otras delimitaciones.	

Elaboración propia.

Tabla N° 4.11

Historia de usuario: Gestión de sub gerencias.

HISTORIA DE USUARIO	
Número: 09	Nombre: Gestión de sub gerencias
Modificación o extensión de la h.u. : Registro de sub gerencias (08)	
Usuario : administrador	Iteración asignada : 01
Prioridad en el negocio : Medio	Puntos estimados (días): 03
Riesgo en desarrollo : Medio	Puntos reales (días): 04
Descripción: El usuario root podrá actualizar los datos de las sub gerencias de seguridad ciudadana. El usuario root no podrá eliminar una sub gerencia que tenga delitos registrados, podrá deshabilitarlo para no generar problemas de persistencia. El usuario root podrá modificar la delimitación de la sub gerencia, siempre y cuando no intersecte con otra sub gerencia.	
Observaciones : Ninguno	

Elaboración propia.

Tabla N° 4.12

Historia de usuario: Registro de sectores y sub sectores.

HISTORIA DE USUARIO	
Número: 10	Nombre: Registro de sectores y sub sectores
Modificación o extensión de la h.u. : Registro de sub gerencias (08)	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 01
Prioridad en el negocio : Alto	Puntos estimados (días): 03
Riesgo en desarrollo : Alto	Puntos reales (días): 04

<p>Descripción:</p> <p>El usuario root podrá registrar los sectores y sub sectores que forman parte de la sub gerencia. La unión de todos los sectores debe ser igual a la delimitacion territorial de la sub gerencia. Los sub sectores tienen el mismo comportamiento que los sectores con la delimitacion territorial, en este caso, la suma de sub sectores es igual al sector.</p>
<p>Observaciones:</p> <p>El sector y sub sector son polígonos representados por PoligonField en la base de datos, es imperativo utilizar tecnología GIS.</p>

Elaboración propia.

Tabla N° 4.13

Historia de usuario: Gestión de sectores y sub sectores.

HISTORIA DE USUARIO	
Número: 11	Nombre: Gestión de sectores y sub sectores
Modificación o extensión de la h.u. : Registro de sectores y sub sectores (10)	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 01
Prioridad en el negocio : Medio	Puntos estimados (días): 04
Riesgo en desarrollo : Medio	Puntos reales (días): 04
<p>Descripción:</p> <p>El usuario root podrá modificar los datos básicos de los sectores y sub sectores de las diferentes sub gerencias.</p> <p>Si se modifica un sector que tiene conflicto con otro sector, se respetará la decisión de la sub gerencia para la no intersección de sectores.</p>	
Observaciones : Ninguno	

Elaboración propia.

Tabla N° 4.14

Historia de usuario: Registro de puestos.

HISTORIA DE USUARIO	
Número: 12	Nombre: Registro de puestos
Modificación o extensión de la h.u. : 08 y 10	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 01
Prioridad en el negocio : Alto	Puntos estimados (días): 03
Riesgo en desarrollo : Alto	Puntos reales (días): 04
Descripción: El usuario root podrá registrar los puestos de control en los sectores o sub sectores registrados en la sub gerencia de seguridad ciudadana. Un puesto de control es determinado por la sub gerencia y sirve como punto de concentración y control en los sectores, el sistema tendrá que almacenar la ubicación, nombre, descripción de los puntos de control.	
Observaciones: El puesto de control es una ubicación exacta y debe pertenecer a un sector o sub sector, se utiliza un PointField en la base de datos.	

Elaboración propia.

Tabla N° 4.15

Historia de usuario: Gestión de puestos.

HISTORIA DE USUARIO	
Número: 13	Nombre: Gestión de puestos
Modificación o extensión de la h.u. : Registro de puestos (12)	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 01
Prioridad en el negocio : Medio	Puntos estimados (días): 03
Riesgo en desarrollo : Medio	Puntos reales (días): 03
Descripción: El usuario root podrá actualizar los datos de los puestos de control en los sectores o sub	

sectores registrados en las sub gerencia de seguridad ciudadana. El usuario root podrá modificar la ubicación del puesto siempre y cuando la ubicación del puesto cambia físicamente.
Observaciones : Ninguno

Elaboración propia.

Tabla N° 4.16

Historia de usuario: Registro de turnos.

HISTORIA DE USUARIO	
Número: 14	Nombre: Registro de turnos
Modificación o extensión de la h.u. : Registro de sub gerencias (08)	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 02
Prioridad en el negocio : Alto	Puntos estimados (días): 03
Riesgo en desarrollo : Alto	Puntos reales (días): 04
Descripción: El usuario root podrá registrar los turnos en formato 24 horas que se determina en la sub gerencia. La unión de todos los turnos debe ser igual a las 24 horas del día, no debe existir un periodo de tiempo que no pertenezca a un turno, por otro lado, no debe existir intersección entre los turnos.	
Observaciones : El turno se registra en un formato TimeField.	

Elaboración propia.

Tabla N° 4.17

Historia de usuario: Gestión de turnos.

HISTORIA DE USUARIO	
Número: 15	Nombre: Gestión de turnos
Modificación o extensión de la h.u. : Registro de turnos (14)	

Usuario : sereno, personal administrativo y administrador	Iteración asignada : 02
Prioridad en el negocio : Medio	Puntos estimados (días) : 04
Riesgo en desarrollo : Medio	Puntos reales (días) : 04
Descripción: El usuario root podrá actualizar los datos de los turnos de las sub gerencia de seguridad ciudadana. El usuario root podrá modificar la hora de inicio y fin del turno siempre y cuando no se genere intersección entre los turnos y la unión complete las 24 horas.	
Observaciones: La suma de turnos debe completar las 24 horas y no debe haber intersección entre los turnos.	

Elaboración propia.

Tabla N° 4.18

Historia de usuario: Registro de cargos

HISTORIA DE USUARIO	
Número: 16	Nombre: Registro de cargos
Modificación o extensión de la h.u. : Registro de sub gerencias (08)	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 02
Prioridad en el negocio : Alto	Puntos estimados (días) : 03
Riesgo en desarrollo : Alto	Puntos reales (días) : 04
Descripción: El usuario root podrá registrar los cargos genéricos que las sub gerencias utilizarán en la gestión de sus usuarios. El usuario root podrá registrar los cargos que cada sub gerencia maneja de forma predeterminada y adicionalmente se le asignara el nivel de acceso a cada cargo, los cargos genéricos son (sereno, administrados, personal administrativo).	
Observaciones:	

Un sereno puede tener cambiar de cargo (sereno motorizado, jefe de brigada, monitor).
Para los patrullajes, los serenos cambian de roles (cargos).

Elaboración propia.

Tabla N° 4.19

Historia de usuario: Gestión de cargos

HISTORIA DE USUARIO	
Número: 17	Nombre: Gestión de cargos
Modificación o extensión de la h.u. : Registro de cargos (16)	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 02
Prioridad en el negocio : Medio	Puntos estimados (días): 03
Riesgo en desarrollo : Medio	Puntos reales (días): 04
Descripción: El usuario root podrá actualizar los datos de los cargos genéricos, siempre y cuando no se tenga actividad por parte de las sub gerencias. El usuario administrador podrá actualizar los cargos que cada sub gerencia, podrá modificar el nivel de acceso de los cargos.	
Observaciones : Ninguno	

Elaboración propia.

Tabla N° 4.20

Historia de usuario: Registro de incidentes.

HISTORIA DE USUARIO	
Número: 18	Nombre: Registro de incidentes
Modificación o extensión de la h.u. : 04, 06, 08, 10, 12, 14 y 16.	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 04
Prioridad en el negocio : Alto	Puntos estimados (días): 05

Riesgo en desarrollo : Alto	Puntos reales (días): 08
Descripción: El usuario del sistema (sereno, personal administrativo), podrá registrar el incidente, el tipo de delito, la ubicación, el momento en que ocurrió, quienes intervinieron y a qué sector pertenece el incidente. El usuario del sistema adicionalmente podrá registrar archivos extra (fotos, video, audio) a los incidentes para almacenarlos como pruebas o imagen referente.	
Observaciones: El sistema registrará la ubicación con el tipo de dato PointField. Se podrán registrar múltiples delitos como múltiples serenitos.	

Elaboración propia.

Tabla N° 4.21

Historia de usuario: Gestión de incidentes.

HISTORIA DE USUARIO	
Número: 19	Nombre: Gestión de incidentes
Modificación o extensión de la h.u. : Registro de incidentes (18)	
Usuario : sereno y administrador	Iteración asignada : 04
Prioridad en el negocio : Medio	Puntos estimados (días): 03
Riesgo en desarrollo : Medio	Puntos reales (días): 03
Descripción: El usuario del sistema (administrador y personal administrativo), podrá actualizar los datos de los incidentes, corregir si algo se modificó o adicionar información. El usuario administrador actualizará los datos bajo su responsabilidad, el sistema guardará un historial de modificaciones.	
Observaciones: El sistema tendrá que mantener un historial de modificaciones por cada instancia de incidente.	

Elaboración propia.

Tabla N° 4.22

Historia de usuario: Gestión mapa de incidentes.

HISTORIA DE USUARIO	
Número: 20	Nombre: Gestión mapa de incidentes
Modificación o extensión de la h.u. : 04, 06, 08, 10, 12, 14, 16 y 18	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 06
Prioridad en el negocio : Alto	Puntos estimados (días): 06
Riesgo en desarrollo : Alto	Puntos reales (días): 08
Descripción: Como usuario del sistema (administrador y personal administrativo), me gustaría poder visualizar el mapa del delito de todos los delitos registrados en mi sub gerencia, poder filtrar por los serenitos que participaron, delitos que se registraron y sectores que presenta; me gustaría poder hacer múltiples filtros en el mapa. Como sereno, me gustaría ver el mapa del delito de los delitos que participe, así como filtrar por tipos de delito y sectores.	
Observaciones : Ninguno	

Elaboración propia.

Tabla N° 4.23

Historia de usuario: Registro de base patrullaje.

HISTORIA DE USUARIO	
Número: 21	Nombre: Registro de base patrullaje
Modificación o extensión de la h.u. : 04, 14 y 16	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 05
Prioridad en el negocio : Alto	Puntos estimados (días): 06
Riesgo en desarrollo : Alto	Puntos reales (días): 08
Descripción: Como usuario del sistema (administrador y personal administrativo) quiero registrar el base	

de patrullaje para periodo de tiempo (normalmente una semana), en el cual tengo que asignar a los serenos el turno, días de descanso y el cargo momentáneo que ocupan.

Al registrar el base de patrullaje, es importante que cada turno cuente con un mínimo de un sereno, para asegurar la disponibilidad de serenos al distrito.

Observaciones:

Se debe validar que cada turno tenga un sereno suscrito a la base, un sereno no puede estar en dos turnos simultáneamente.

Elaboración propia.

Tabla N° 4.24

Historia de usuario: Gestión de base patrullaje.

HISTORIA DE USUARIO	
Número: 22	Nombre: Gestión de base patrullaje
Modificación o extensión de la h.u. : Registro de base patrullaje (21)	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 05
Prioridad en el negocio : Medio	Puntos estimados (días): 04
Riesgo en desarrollo : Medio	Puntos reales (días): 05
Descripción: Como usuario del sistema (administrador y personal administrativo) quiero poder actualizar los datos de la base patrullaje, siempre que se respete la presencia de un mínimo de un sereno en cada turno y que un sereno no esté en dos turnos simultáneamente.	
Observaciones : Ninguno	

Elaboración propia.

Tabla N° 4.25

Historia de usuario: Gestión de tipos de delito.

HISTORIA DE USUARIO	
Número: 23	Nombre: Gestión de tipos de delito

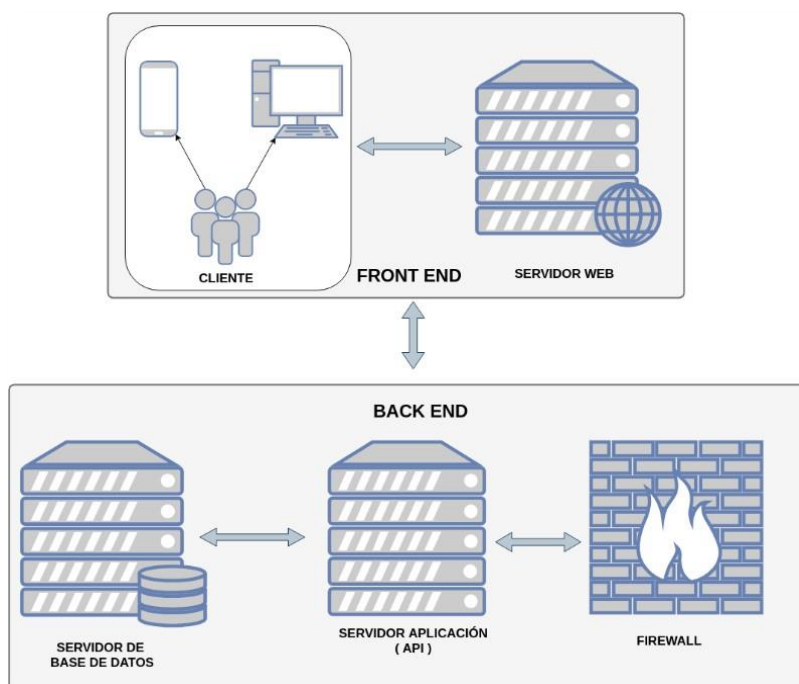
Modificación o extensión de la h.u. : 04, 06 y 18	
Usuario : sereno, personal administrativo y administrador	Iteración asignada : 05
Prioridad en el negocio : Medio	Puntos estimados (días): 04
Riesgo en desarrollo : Medio	Puntos reales (días): 05
Descripción: Como usuario del sistema (administrador y personal administrativo) quiero poder ver cuál es el delito que más presencia tiene en mi sub gerencia, poder saber en qué sector se concentra más, que días ocurre y en que horarios. Del mismo modo, quiero conocer que sectores y días tienen mayor antijuridicidad y culpabilidad.	
Observaciones : Ninguno	

Elaboración propia.

D. ARQUITECTURA TÉCNICA INICIAL

Figura N° 4.1

Arquitectura técnica inicial.



Elaboración propia.

4.1.2 FASE DE PLANIFICACIÓN

A. PLAN DE ENTREGA

Tabla N° 4.26

Plan de entrega (reléase plan).

N.º	HISTORIA DE USUARIO	PRIORIDAD	RIESGO	ESFUERZO (DÍAS)	ITERACIÓN
01	Acceso al sistema	ALTO	ALTO	04	03
02	Registro de permisos	ALTO	ALTO	04	03
03	Gestión de permisos	MEDIO	MEDIO	05	03
04	Registro de usuarios	ALTO	ALTO	05	03
05	Gestión de usuarios	MEDIO	MEDIO	03	03
06	Registro de marco legal	ALTO	ALTO	05	03
07	Gestión del marco legal	MEDIO	MEDIO	04	02
08	Registro de sub gerencias	ALTO	ALTO	04	01
09	Gestión de sub gerencias	MEDIO	MEDIO	04	01
10	Registro de sectores y sub sectores	ALTO	ALTO	04	01
11	Gestión de sectores y sub sectores	MEDIO	MEDIO	04	01
12	Registro de puestos	ALTO	ALTO	04	01
13	Gestión de puestos	MEDIO	MEDIO	03	01
14	Registro de turnos	ALTO	ALTO	04	01
15	Gestión de turnos	MEDIO	MEDIO	04	02
16	Registro de cargos	ALTO	ALTO	04	02
17	Gestión de cargos	MEDIO	MEDIO	03	02
18	Registro de incidentes	ALTO	ALTO	08	04
19	Gestión de incidentes	MEDIO	MEDIO	03	04
20	Gestión mapa de incidentes	ALTO	ALTO	08	06

21	Registro de base patrullaje	ALTO	ALTO	08	05
22	Gestión de base patrullaje	MEDIO	MEDIO	05	05
23	Gestión de tipos de delito	MEDIO	MEDIO	08	06

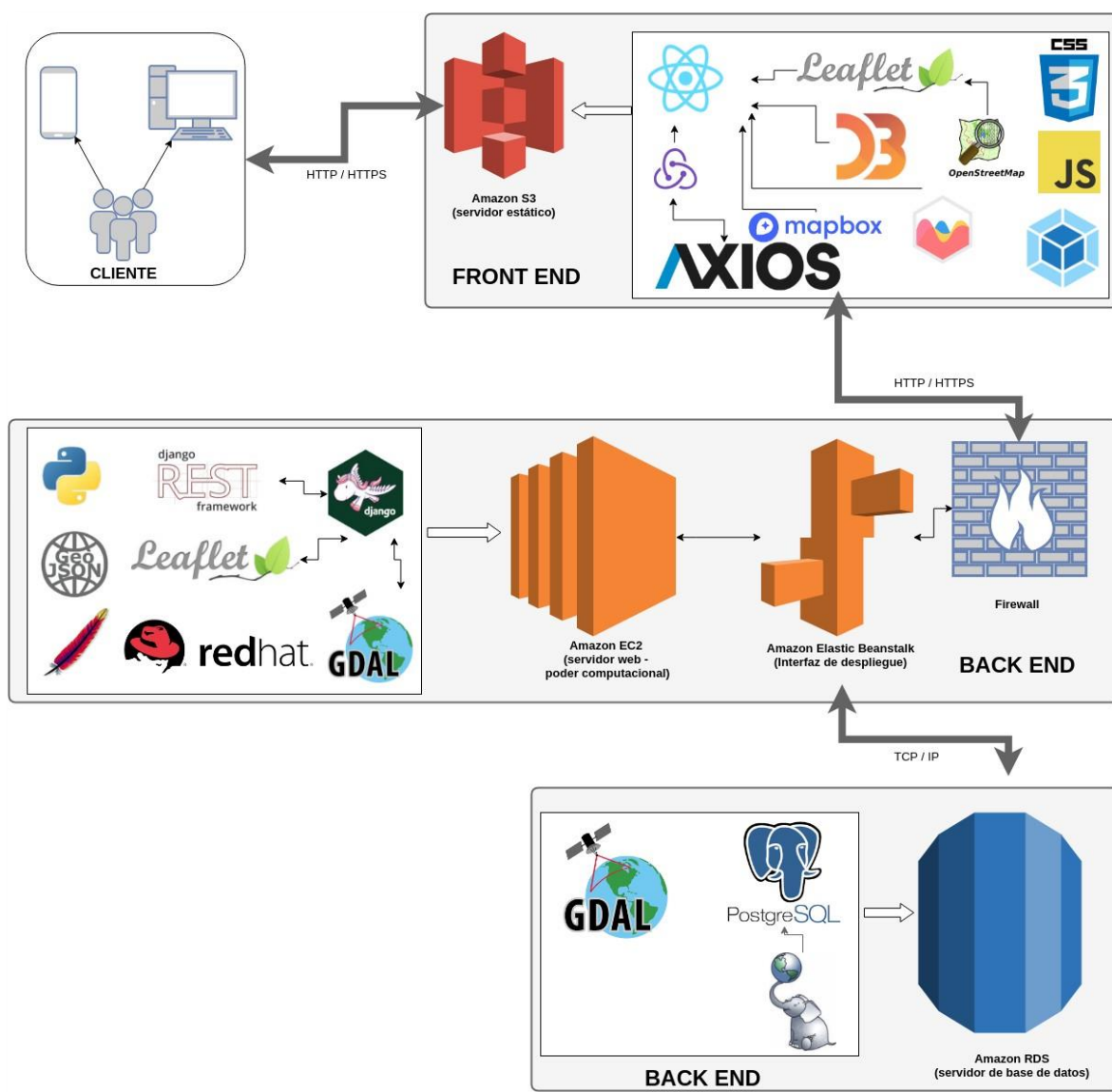
Elaboración propia.

4.1.3 FASE DE ITERACIÓN

A. ARQUITECTURA TÉCNICA

Figura N° 4.2

Arquitectura técnica.



Elaboración propia.

En la figura 4.2 “arquitectura técnica”, se muestran cuatro secciones representadas por rectángulos grises los cuales son sectores de la arquitectura técnica, se detallan de abajo hacia arriba.

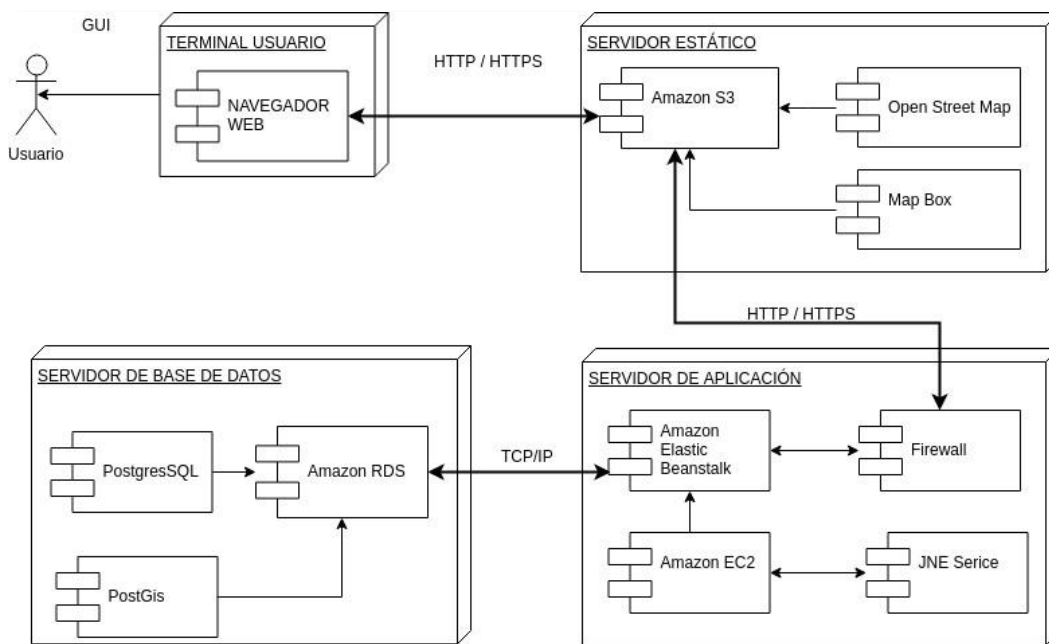
El primer sector representa la lógica de datos, se utiliza RDS de Amazon como servicio proveedor de base de datos relacional, sobre el cual se instancia un PostgreSQL como gestor de base de datos y Postgis como paquete GIS para la manipulación de datos espaciales para PostgreSQL, Postgis utiliza librerías de Gdal (Geospatial Data Abstraction Library).

El segundo sector representa la lógica de negocios, se utiliza el servicio de Elastic Beanstalk como interfaz de despliegue, la interfaz crea una instancia en el servicio Amazon Elastic Compute Cloud (EC2) con grupos de seguridad que funcionan como firewall virtual entre las instancias y el exterior, ambos servicios son de Amazon; como lenguaje de programación principal se tiene a Python en su tercera versión, como framework web a Django en su segunda versión y su paquete geográfico GeoDjango, como paquete gestor de los servicios api rest se tiene a Django-rest-framework que utiliza Json y GeoJson como formato de intercambio de datos, este último permite interpretar los datos espaciales de geoDjango y los libera en el api rest, se utiliza a Leaflet como gestor de mapas en backend y por la integración con OpenStreetMap; geoDjango utiliza librerías de Gdal. Todas las tecnologías y paquetes mencionados corren sobre el sistema operativo Red Hat Enterprise Linux (RHEL) y como servidor web se tiene un apache server.

El tercer sector derecho representa el frontend, se utiliza el servicio S3 de Amazon para despachar archivos multimedia del backend y ser el servidor estático para la aplicación frontend, como framework frontend se tiene a React, como manejador de estados se tiene a Redux y axios como paquete que utiliza el protocolo HTTP para hacer promesas con el backend, como paquetes para manejar datos y crear gráficos estadísticos se tiene a D3 y Chartjs, como proveedor de mapas se vuelve a utilizar Leaflet y Mapbox, finalmente, como empaquetador se utiliza a WebPack.

El tercer sector izquierdo representa al usuario final, el cual puede consumir la aplicación por medio de una terminal desktop o mobile. Al descomponer la aplicación en los diferentes servicios de Amazon, se abre la posibilidad crear aplicaciones mobile.

Figura N° 4.3
Arquitectura técnica, diagrama de despliegue.



Elaboración propia.

B. PLAN DE ITERACIÓN

Tabla N° 4.27
Plan de iteraciones (primera iteración).

N° H.U.	HISTORIA DE USUARIO	N° T. I.	TAREA DE INGENIERÍA
08	Registro de sub gerencias	01	Registrar sub gerencia
		02	Registrar polígono sub gerencia
10	Registro de sectores y sub sectores	03	Registrar sector
		04	Registrar polígono sector
		05	Registrar sub sector
		06	Registrar polígono sub sector
12	Registro de puestos	07	Registrar puesto
		08	Registrar punto para el puesto
09	Gestión de sub gerencias	09	Mantener sub gerencia

11	Gestión de sectores y sub sectores	10	Mantener sector
		11	Mantener sub sector
13	Gestión de puestos	12	Mantener puesto

Elaboración propia.

Tabla N° 4.28

Plan de iteraciones (segunda iteración).

N° H.U.	HISTORIA DE USUARIO	N° T. I.	TAREA DE INGENIERÍA
06	Registro de marco legal	13	Estructurar código penal
		14	Registrar código penal
14	Registro de turnos	15	Registrar turno
16	Registro de cargos	16	Registrar cargo
07	Gestión del marco legal	17	Mantener marco legal
15	Gestión de turnos	18	Mantener turno
17	Gestión de cargos	19	Mantener cargo

Elaboración propia.

Tabla N° 4.29

Plan de iteraciones (tercera iteración).

N° H.U.	HISTORIA DE USUARIO	N° T. I.	TAREA DE INGENIERÍA
04	Registro de usuarios	20	Registrar usuario
		21	Registrar sereno
01	Acceso al sistema	22	Login usuario
		23	Logout usuario
		24	Refresh token
		25	Integrar redux con el token
02	Registro de permisos	26	Registrar permiso

03	Gestión de permisos	27	Mantener permiso
05	Gestión de usuarios	28	Mantener usuario
		29	Mantener sereno

Elaboración propia.

Tabla N° 4.30

Plan de iteraciones (cuarta iteración).

N° H.U.	HISTORIA DE USUARIO	N° T. I.	TAREA DE INGENIERÍA
18	Registro de incidentes	30	Registrar incidente
		31	Registrar delito (s)
		32	Registrar punto geográfico
		33	Registrar archivo adicional (s)
		34	Registrar persona involucrada
19	Gestión de incidentes	35	Mantener incidente
		36	Detallar incidente

Elaboración propia.

Tabla N° 4.31

Plan de iteraciones (quinta iteración).

N° H.U.	HISTORIA DE USUARIO	N° T. I.	TAREA DE INGENIERÍA
21	Registro de base patrullaje	37	Registrar base patrullaje
		38	Integrar redux con el árbol de base patrullaje
22	Gestión de base patrullaje	39	Mantener base patrullaje

Elaboración propia.

Tabla N° 4.32

Plan de iteraciones (sexta iteración).

N° H.U.	HISTORIA DE USUARIO	N° T. I.	TAREA DE INGENIERÍA
---------	---------------------	----------	---------------------

20	Gestión mapa de incidentes	40	Generar mapa de incidente
		41	Filtrar mapa de incidente
		42	Integrar redux con el mapa de incidente
23	Gestión de tipos de delito	43	Generar nube de delitos
		44	Filtrar nube de delitos (dashboard)
		45	Integrar redux con la nube de delitos
		46	Generar cuadros de antijuridicidad
		47	Integrar redux a los cuadros de antijuridicidad
		48	Generar cuadros de culpabilidad
		49	Integrar redux a los cuadros de culpabilidad

Elaboración propia.

C. TAREAS DE INGENIERÍA

C.1. PRIMERA ITERACIÓN

Tabla N° 4.33

Tarea de ingeniería: Registrar sub gerencia.

TAREA DE INGENIERÍA	
Número de tarea : 01	Historia de usuario: Registro de sub gerencias (08)
Nombre de tarea : Registrar sub gerencia	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2018-10-01	Fecha fin : 2018-10-03
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario del sistema (root) selecciona módulo SGSC del administrador backend del sistema; el sistema despliega un menú de opciones. El usuario en sesión selecciona el menú SGSC , botón AÑADIR SGSC , el sistema muestra	

<p>una interfaz con el formulario «AÑADIR SGSC» para agregar la instancia de SGSC con los campos (nombre, misión, visión, logo, color, imagen, mapa y un desplegable pre seleccionado con el usuario que está en sesión).</p> <p>El usuario en sesión completa los datos del formulario y hace clic en el botón GRABAR, el sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente la «nombre de la sgsc» ”.</p>
Observaciones: Ninguno

Elaboración propia

Tabla N° 4.34

Tarea de ingeniería: Registrar polígono sub gerencia.

TAREA DE INGENIERÍA	
Número de tarea : 02	Historia de usuario: Registro de sub gerencias (08)
Nombre de tarea : Registrar polígono sub gerencia	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2018-10-03	Fecha fin : 2018-10-05
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El sistema al detectar que se guarda una instancia del tipo SGSC, verifica la existencia del campo ÁREA.</p> <p>El sistema almacena en memoria el valor del campo área enviado por el formulario AÑADIR SGSC.</p> <p>El sistema valida el campo área formando un polígono geográfico y que no tenga intersección con las demás instancias SGSC (atributo área) registrados en la base de datos.</p>	
<p>Observaciones:</p> <p>El polígono validado no debe intersectar con las áreas de las demás instancias de SGSC.</p> <p>El polígono debe ser un objeto del tipo PolygonField.</p>	

Elaboración propia.

Tabla N° 4.35

Tarea de ingeniería: Registrar sector.

TAREA DE INGENIERÍA	
Número de tarea : 03	Historia de usuario: Registro de sectores y sub sectores (10)
Nombre de tarea : Registrar sector	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2018-10-05	Fecha fin : 2018-10-08
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona módulo SGSC del administrador backend del sistema; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú SECTOR, botón AÑADIR SECTOR, el sistema muestra una interfaz con el formulario «AÑADIR SECTOR» para agregar la instancia de sector con los campos (nombre, color, imagen, un desplegable con las sub gerencias registradas (SGSC), sector y un desplegable pre seleccionado con el usuario que está en sesión).</p> <p>El usuario en sesión completa los datos del formulario y hace clic en el botón GRABAR, el sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente la «nombre del sector »” .</p>	
Observaciones: Ninguno	

Elaboración propia.

Tabla N° 4.36

Tarea de ingeniería: Registrar polígono sector.

TAREA DE INGENIERÍA	
Número de tarea : 04	Historia de usuario: Registro de sectores y sub sectores (10)
Nombre de tarea : Registrar polígono sector	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2018-10-08	Fecha fin : 2018-10-09
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	

<p>Descripción:</p> <p>El sistema al detectar que se guarda una instancia del tipo SECTOR, verifica la existencia del campo SECTOR.</p> <p>El sistema almacena en memoria el valor del campo sector enviado por el formulario AÑADIR SECTOR.</p> <p>El sistema valida el campo sector formando un polígono geográfico, que no tenga intersección con las demás instancias SECTOR (atributo sector) registrados en la base de datos y que el polígono este incluido en la SGSC (atributo área).</p>
<p>Observaciones:</p> <p>El polígono creado no debe intersectar con los sectores de las demás instancias de sector.</p> <p>El polígono creado debe estar incluido en el área de la SGSC.</p> <p>El polígono debe ser un objeto del tipo PolygonField.</p>

Elaboración propia.

Tabla N° 4.37

Tarea de ingeniería: Registrar sub sector.

TAREA DE INGENIERÍA	
Número de tarea : 05	Historia de usuario: Registro de sectores y sub sectores (10)
Nombre de tarea : Registrar sub sector	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2018-10-09	Fecha fin : 2018-10-10
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona módulo SGSC del administrador backend del sistema; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú SUB SECTOR, botón AÑADIR SUB SECTOR, el sistema muestra una interfaz con el formulario «AÑADIR SUB SECTOR» para agregar la instancia de sub sector con los campos (nombre, color, un desplegable con los sectores registrados, sector y un desplegable pre seleccionado con el usuario que está en sesión).</p> <p>El usuario en sesión completa los datos del formulario y hace clic en el botón GRABAR, el</p>	

sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente la « nombre del sub sector » .
Observaciones: Ninguno

Elaboración propia.

Tabla N° 4.38

Tarea de ingeniería: Registrar polígono sub sector.

TAREA DE INGENIERÍA	
Número de tarea : 06	Historia de usuario: Registro de sectores y sub sectores (10)
Nombre de tarea : Registrar polígono sub sector	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2018-10-10	Fecha fin : 2018-10-11
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El sistema al detectar que se guarda una instancia del tipo SUB SECTOR , verifica la existencia del campo SECTOR . El sistema almacena en memoria el valor del campo sector enviado por el formulario AÑADIR SUB SECTOR . El sistema valida el campo sector formando un polígono geográfico, que no tenga intersección con las demás instancias SUB SECTOR (atributo sector) registrados en la base de datos y que el polígono este incluido en el SECTOR (atributo sector).	
Observaciones: El polígono creado no debe intersectar con los sectores de las demás instancias de sub sector. El polígono creado debe estar incluido en el sector del modelo sector. El polígono debe ser un objeto del tipo PolygonField.	

Elaboración propia.

Tabla N° 4.39

Tarea de ingeniería: Registrar puesto.

TAREA DE INGENIERÍA	
Número de tarea : 07	Historia de usuario: Registro de puestos (12)
Nombre de tarea : Registrar puesto	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2018-10-11	Fecha fin : 2018-10-15
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona módulo SGSC del administrador backend del sistema; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú PUESTO, botón AÑADIR PUESTO, el sistema muestra una interfaz con el formulario «AÑADIR PUESTO» para agregar la instancia de puesto con los campos (nombre, color, imagen, un desplegable con los sectores registrados, ubicación, un desplegable pre seleccionado con el usuario que está en sesión y isvisible).</p> <p>El usuario en sesión completa los datos del formulario y hace clic en el botón GRABAR, el sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente la «nombre del puesto » ”.</p>	
<p>Observaciones:</p> <p>El campo isvisible es un booleano que por default marca como True, sirve como medida de prevención para eliminar instancias y no tener problemas de consistencia.</p>	

Elaboración propia.

Tabla N° 4.40

Tarea de ingeniería: Registrar punto para el puesto.

TAREA DE INGENIERÍA	
Número de tarea : 08	Historia de usuario: Registro de puestos (12)
Nombre de tarea : Registrar punto para el puesto	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02

Fecha de inicio : 2018-10-15	Fecha fin : 2018-10-17
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El sistema al detectar que se guarda una instancia del tipo PUESTO , verifica la existencia del campo UBICACIÓN . El sistema almacena en memoria el valor del campo sector enviado por el formulario AÑADIR PUESTO . El sistema valida el campo sector formando un punto geográfico, contener latitud y longitud cono propiedades y que el punto este incluido en el SECTOR (atributo sector).	
Observaciones: El punto creado debe estar incluido en el sector del modelo sector. El punto debe ser un objeto del tipo PointField.	

Elaboración propia.

Tabla N° 4.41

Tarea de ingeniería: Mantener sub gerencia

TAREA DE INGENIERÍA	
Número de tarea : 09	Historia de usuario: Gestión de sub gerencias (09)
Nombre de tarea : Mantener sub gerencia	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 04
Fecha de inicio : 2018-10-17	Fecha fin : 2018-10-23
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario del sistema (root) selecciona módulo SGSC del administrador backend; el sistema despliega un menú de opciones. El usuario en sesión selecciona el menú SGSC , el sistema lista las instancias de SGSC creadas. Si el usuario en sesión quiere actualizar una SGSC hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario « MODIFICAR SGSC », el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR , el sistema actualiza los datos en	

la base de datos y retorna un mensaje de confirmación “Se actualizo la «nombre de la sgsc»”.

Si el usuario en sesión quiere eliminar una **SGSC** selecciona en la instancia (s) que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «numero de instancias» registros”.

Observaciones:

Como alternativa a la eliminación de registros, se puede optar por marcar **FALSE** el field flag. Al actualizar una instancia se validan con los mismos parámetros que al crearlo.

Elaboración propia.

Tabla N° 4.42

Tarea de ingeniería: Mantener sector.

TAREA DE INGENIERÍA	
Número de tarea : 10	Historia de usuario: Gestión de sectores y sub sectores (11)
Nombre de tarea : Mantener sector	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2018-10-23	Fecha fin : 2018-10-25
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona módulo SGSC del administrador backend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú SECTOR, el sistema lista las instancias de SECTOR creadas.</p> <p>Si el usuario en sesión quiere actualizar un SECTOR hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario «MODIFICAR SECTOR», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo la «nombre del sector»”.</p> <p>Si el usuario en sesión quiere eliminar un SECTOR selecciona en la instancia (s) que desea</p>	

eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «**numero de instancias**» registros”.

Observaciones:

Como alternativa a la eliminación de registros, se puede optar por marcar **FALSE** el field flag. Al actualizar una instancia se validan con los mismos parámetros que al crearlo.

Elaboración propia.

Tabla N° 4.43

Tarea de ingeniería: Mantener sub sector.

TAREA DE INGENIERÍA	
Número de tarea : 11	Historia de usuario: Gestión de sectores y sub sectores (11)
Nombre de tarea : Mantener sub sector	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2018-10-25	Fecha fin : 2018-10-29
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona módulo SGSC del administrador backend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú SUB SECTOR, el sistema lista las instancias de SUB SECTOR creadas.</p> <p>Si el usuario en sesión quiere actualizar un SUB SECTOR hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario «MODIFICAR SUB SECTOR», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo la «nombre del sub sector»”.</p> <p>Si el usuario en sesión quiere eliminar un SUB SECTOR selecciona en la instancia (s) que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «numero de instancias»</p>	

registros”.
Observaciones: Como alternativa a la eliminación de registros, se puede optar por marcar FALSE el field flag. Al actualizar una instancia se validan con los mismos parámetros que al crearlo.

Elaboración propia.

Tabla N° 4.44

Tarea de ingeniería: Mantener puesto

TAREA DE INGENIERÍA	
Número de tarea : 12	Historia de usuario: Gestión de puestos (13)
Nombre de tarea : Mantener puesto	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 03
Fecha de inicio : 2018-10-29	Fecha fin : 2018-11-01
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario del sistema (root) selecciona módulo SGSC del administrador backend; el sistema despliega un menú de opciones. El usuario en sesión selecciona el menú PUESTO el sistema lista las instancias de SUB PUESTO creadas. Si el usuario en sesión quiere actualizar un PUESTO hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario « MODIFICAR PUESTO », el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR , el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo la « nombre del puesto »”. Si el usuario en sesión quiere eliminar un PUESTO selecciona en la instancia (s) que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó « numero de instancias » registros”.	
Observaciones: Como alternativa a la eliminación de registros, se puede optar por marcar FALSE el field flag.	

Al actualizar una instancia se validan con los mismos parámetros que al crearlo.
Elaboración propia.

C2. SEGUNDA ITERACIÓN

Tabla N° 4.45

Tarea de ingeniería: Estructurar código penal.

TAREA DE INGENIERÍA	
Número de tarea : 13	Historia de usuario: Registro de marco legal (06)
Nombre de tarea : Estructurar código penal	
Tipo de tarea : Diseño	Puntos estimados (días) : 03
Fecha de inicio : 2018-11-01	Fecha fin : 2018-11-06
Diseñador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El equipo de desarrollo abstrae la estructura del código penal peruano. Para registrar el tipo de delito (delito y faltas), se crean estructuras superiores (CATEGORIATIPODELITO y SUBCATEGORIATIPODELITOS), las estructuras superiores sirven como clasificador del TIPO DELITO . Para mantener una mayor independencia entre las diferentes sub gerencias y el tipo de delito, se crea un objeto intermedio (INCIDENCIATIPODELITO), el objeto sirve como puente y utiliza los datos descriptivos (tiempo y lugar) de los incidentes. El objeto (INCIDENCIATIPODELITO), tiene relación con la sgsc, tipodelito, serenos, turno y sector.	
Observaciones: Ninguno	

Elaboración propia.

Tabla N° 4.46

Tarea de ingeniería: Registrar del código penal.

TAREA DE INGENIERÍA	
Número de tarea : 14	Historia de usuario: Registro de marco legal (06)

Nombre de tarea : Registrar del código penal	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2018-11-06	Fecha fin : 2018-11-08
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario del sistema (root) selecciona modulo LEGAL del administrador backend del sistema; el sistema despliega un menú de opciones. El usuario en sesión selecciona el menú TIPO DELITO , botón AÑADIR TIPO DELITO , el sistema muestra una interfaz con el formulario « AÑADIR TIPO DELITO » para agregar la instancia de tipo delito con los campos (artículo, nombre, un desplegable con la sub categorías registrados, descripción, estado y número de casos que se inicializa con cero). El usuario en sesión completa los datos del formulario y hace clic en el botón GRABAR , el sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente la « nombre del tipo delito » .	
Observaciones: EL número de casos es un field que se inicializa en cero.	

Elaboración propia.

Tabla N° 4.47

Tarea de ingeniería: Registrar turno.

TAREA DE INGENIERÍA	
Número de tarea : 15	Historia de usuario: Registro de turnos (14)
Nombre de tarea : Registrar turno	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 04
Fecha de inicio : 2018-11-08	Fecha fin : 2018-11-14
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario del sistema (root) selecciona módulo SGSC del administrador backend del sistema; el sistema despliega un menú de opciones.	

El usuario en sesión selecciona el menú **TURNO**, botón **AÑADIR TURNO**, el sistema muestra una interfaz con el formulario «**AÑADIR TURNO**» para agregar la instancia de turno con los campos (nombre, hora inicio, hora fin, un desplegable con las sgsc, un desplegable pre seleccionado con el usuario que está en sesión, isvigente).

El usuario en sesión completa los datos del formulario y hace clic en el botón **GRABAR**, el sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente la «**nombre del turno** »”.

Observaciones:

El campo isvigente es un booleano que por default marca como True, sirve como medida de prevención para no eliminar instancias.

La hora inicio y hora fin son fields del tipo TimeField que utilizan el **TIME_ZONE** especificado en la configuración del proyecto (América/Lima).

Elaboración propia.

Tabla N° 4.48

Tarea de ingeniería: Registrar cargo.

TAREA DE INGENIERÍA	
Número de tarea : 16	Historia de usuario: Registro de cargos (16)
Nombre de tarea : Registrar cargo	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 04
Fecha de inicio : 2018-11-14	Fecha fin : 2018-11-20
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona módulo SGSC del administrador backend del sistema; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú CARGO, botón AÑADIR CARGO, el sistema muestra una interfaz con el formulario «AÑADIR CARGO» para agregar la instancia de cargo con los campos (nombre, isvehiculo, vehículo, canpatrullaje, canincidente, un desplegable con las sgsc, un desplegable pre seleccionado con el usuario que está en sesión).</p> <p>El usuario en sesión completa los datos del formulario y hace clic en el botón GRABAR, el</p>	

sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente la «**nombre del cargo** »”.

Observaciones: Ninguno

Elaboración propia.

Tabla N° 4.49

Tarea de ingeniería: Mantener marco legal.

TAREA DE INGENIERÍA	
Número de tarea : 17	Historia de usuario: Gestión del marco legal (07)
Nombre de tarea : Mantener marco legal	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 04
Fecha de inicio : 2018-11-20	Fecha fin : 2018-11-26
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona modulo LEGAL del administrador backend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú TIPO DELITO, el sistema lista las instancias de TIPO DELITO creadas.</p> <p>Si el usuario en sesión quiere actualizar un TIPO DELITO hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario «MODIFICAR TIPO DELITO», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo la «nombre del tipo delito»”.</p> <p>Si el usuario en sesión quiere eliminar un TIPO DELITO selecciona en la instancia (s) que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «numero de instancias» registros”.</p>	
<p>Observaciones:</p> <p>Como alternativa a la eliminación de registros, se puede optar por marcar FALSE el field flag.</p>	

Al actualizar una instancia se validan con los mismos parámetros que al crearlo.

Elaboración propia.

Tabla N° 4.50

Tarea de ingeniería: Mantener turno.

TAREA DE INGENIERÍA	
Número de tarea : 18	Historia de usuario: Gestión de turnos (15)
Nombre de tarea : Mantener turno	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 04
Fecha de inicio : 2018-11-23	Fecha fin : 2018-11-30
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona módulo SGSC del administrador backend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú TURNO, el sistema lista las instancias de TURNO creadas.</p> <p>Si el usuario en sesión quiere actualizar un TURNO hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario «MODIFICAR TURNO», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo la «nombre del turno»”.</p> <p>Si el usuario en sesión quiere eliminar una TURNO selecciona en la instancia (s) que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «numero de instancias» registros”.</p>	
<p>Observaciones:</p> <p>Como alternativa a la eliminación de registros, se puede optar por marcar FALSE el field flag.</p> <p>Al actualizar una instancia se validan con los mismos parámetros que al crearlo.</p>	

Elaboración propia.

Tabla N° 4.51

Tarea de ingeniería: Mantener cargo.

TAREA DE INGENIERÍA	
Número de tarea : 19	Historia de usuario: Gestión de cargos (17)
Nombre de tarea : Mantener cargo	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 03
Fecha de inicio : 2018-11-30	Fecha fin : 2018-12-05
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona módulo SGSC del administrador backend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú CARGO, el sistema lista las instancias de CARGO creadas.</p> <p>Si el usuario en sesión quiere actualizar un CARGO hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario «MODIFICAR CARGO», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo la «nombre del turno»”.</p> <p>Si el usuario en sesión quiere eliminar una CARGO selecciona en la instancia (s) que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «numero de instancias» registros”.</p>	
<p>Observaciones:</p> <p>Como alternativa a la eliminación de registros, se puede optar por marcar FALSE el field flag. Al actualizar una instancia se validan con los mismos parámetros que al crearlo.</p>	

Elaboración propia.

C3. TERCERA ITERACIÓN

Tabla N° 4.52

Tarea de ingeniería: Registrar usuario.

TAREA DE INGENIERÍA	
Número de tarea : 20	Historia de usuario: Registro de usuarios (04)
Nombre de tarea : Registrar usuario	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2018-12-05	Fecha fin : 2018-12-07
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona modulo USUARIO del administrador backend del sistema; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú USER, botón AÑADIR USER, el sistema muestra una interfaz con el formulario «AÑADIR USER» para agregar la instancia del user con los campos (apellido_paterno,apellido_materno, nombres, dni, sexo, nacimiento, foto,descripcion, cel, issereno, username, password).</p> <p>El usuario en sesión completa los datos del formulario y hace clic en el botón GRABAR, el sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente el «nombre del user»”.</p>	
<p>Observaciones:</p> <p>El username es un campo único, requerido y agnóstico a la sub gerencia que se suscriba.</p> <p>El dni se valida con un servicio tercero para obtener el nombre y apellidos.</p> <p>El password se encripta con PBKDF1 (Password-Based Key Derivation Function 1) con SHA256.</p>	

Elaboración propia.

Tabla N° 4.53

Tarea de ingeniería: Registrar sereno.

TAREA DE INGENIERÍA	
Número de tarea : 21	Historia de usuario: Registro de usuarios (04)
Nombre de tarea : Registrar sereno	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 03

Fecha de inicio : 2018-12-07	Fecha fin : 2018-12-12
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (administrador) selecciona módulo ADMINISTRACIÓN de la aplicación frontend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú REGISTRAR SERENO, el sistema carga una interfaz con una secuencia de pasos (3) para el registro de serenos.</p> <p>El primer paso (AUTENTICACION), se ingresa los datos (nombre de usuario, alias, correo electrónico), el sistema valida en tiempo real cada campo (el nombre de usuario es un campo único para todo el sistema, el alias es un campo único para las sub gerencias, el correo es un campo único para todo el sistema); si los campos son válidos, el sistema permite pasar al siguiente paso.</p> <p>El segundo paso (DATOS PERSONALES) ingresa los datos (nombre, apellidos, número de documento de identidad, fecha de nacimiento, sexo, numero de contacto y una descripción), el sistema valida el número de dni y carga los datos de nombre y apellidos; si los campos son válidos, el sistema permite pasar al siguiente paso.</p> <p>El tercer paso (CARGO EN LA SGSC) ingresa los datos (cargo, fecha de asignación y descripción), el sistema lista los cargos que fueron registrados indicando si cada cargo puede participar en patrullaje, incidentes o si es un personal administrativo.; si los campos son válidos, la aplicación frontend por medio de redux envía los datos al sistema backend para su validación y posterior registro.</p> <p>El sistema valida los datos, si ocurre un error, responde con un estado 40X(dependiendo del tipo de error).</p> <p>Si los datos son correctos, el sistema crea las instancias necesarias (User, Sereno, serenoID, SerenoCargo) y los registra en la base de datos, luego se regresa un estado 201 (created).</p> <p>El sistema recepciona la respuesta y dependiendo del estado que devuelva, muestra un mensaje de confirmación o error.</p>	
<p>Observaciones:</p> <p>Se tiene que tener creado la instancia Sgsc.</p> <p>Se tiene que tener creado la instancia Cargo.</p> <p>Se tiene que tener creado la instancia User.</p>	

Elaboración propia.

Tabla N° 4.54

Tarea de ingeniería: Login usuario.

TAREA DE INGENIERÍA	
Número de tarea : 22	Historia de usuario: Acceso al sistema (01)
Nombre de tarea : Login usuario	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 1
Fecha de inicio : 2018-12-12	Fecha fin : 2018-12-13
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario (administrador, personal administrativo, sereno), ingresa a la interfaz inicial «/», el sistema muestra la portada de la aplicación frontend.</p> <p>El usuario en sesión hace clic en el botón INGRESAR, el sistema redirige a la interfaz «/login» con el formulario LOGIN.</p> <p>El usuario en sesión ingresa, envía su username y password, el sistema valida la autenticidad del usuario en sesión.</p> <p>Si la autenticación es favorable, el sistema redirige a la interfaz «/admin/dash» y por medio de redux, carga datos del usuario al estado de la aplicación.</p> <p>Si la autenticación no es favorable, el sistema muestra un mensaje de error y agrega en 1 al contador de intentos.</p> <p>El sistema valida que el contador no exceda los 4 intentos para hacer la validación, en caso contrario, el sistema bloquea la interfaz por 1 minuto.</p>	
<p>Observaciones:</p> <p>El username y el password son campos obligatorios.</p> <p>Si la autenticación es favorable, recién se accede al jwt del usuario.</p> <p>El sistema almacena en memoria el estado de la autenticación.</p>	

Elaboración propia.

Tabla N° 4.55

Tarea de ingeniería: Logout usuario.

TAREA DE INGENIERÍA	
Número de tarea : 23	Historia de usuario: Acceso al sistema (01)
Nombre de tarea : Logout usuario	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2018-12-13	Fecha fin : 2018-12-14
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario (administrador, personal administrativo, sereno), se encuentra en cualquier interfaz de la aplicación excepto « /; /login ». El usuario en sesión hace clic en su menú de opciones (superior derecho o superior izquierdo dependiendo de la resolución de su interfaz), el sistema muestra un menú de opciones. El usuario en sesión hace clic en el botón SALIR , el sistema elimina la sesión del usuario y redirige a la interfaz « /login ». El sistema detecta que se eliminó el estado de autenticación, elimina las variables del LOCALSTORAGE .	
Observaciones: El sistema elimina el token que está en memoria. El sistema elimina la AUTHORIZATION de la cabecera axios. El sistema agrega a la blacklist el jwt.	

Elaboración propia.

Tabla N° 4.56

Tarea de ingeniería: Refresh token.

TAREA DE INGENIERÍA	
Número de tarea : 24	Historia de usuario: Acceso al sistema (01)
Nombre de tarea : Refresh token	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2018-12-14	Fecha fin : 2018-12-17
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	

<p>Descripción:</p> <p>El sistema detecta expira el JWT (ACCESS) se acerca, hace una petición y renueva el token.</p> <p>El sistema registra el JWT (ACCESS Y REFRESH) expirado en la blacklist para que no pueda ser utilizado posteriormente.</p> <p>Si en la renovación del JWT ocurre un error, el sistema realiza el logout del usuario.</p> <p>Existe la posibilidad que el JWT sea válido pero ya haya expirado, en este paso se responde con un estado 401 (UNAUTHORIZED) y se hace un logout del usuario.</p>
<p>Observaciones:</p> <p>El sistema utiliza el refresh token para hacer la renovación.</p>

Elaboración propia.

Tabla N° 4.57

Tarea de ingeniería: Integrar redux con el token.

TAREA DE INGENIERÍA	
Número de tarea : 31	Historia de usuario: Acceso al sistema (01)
Nombre de tarea : Integrar redux con el token	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2018-12-17	Fecha fin : 2018-12-18
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El sistema detecta que el estado de autenticación es favorable, decodifica el token y obtiene los datos del usuario (username, foto, nombre y apellido del usuario), por medio de redux los agrega al STORE para su rehúso en las demás interfaces.</p> <p>El sistema por medio de redux guarda el TOKEN (ACCESS) en la cabecera de axios para que las futuras peticiones tengan acceso al backend.</p> <p>El sistema guarda en el LOCALSTORAGE el uuid de la SGSC.</p> <p>Al momento de hacer logout REDUX elimina el estado de autenticación del STORE y las variables del LOCALSTORAGE.</p>	
<p>Observaciones:</p> <p>Si ocurre un error en la decodificación del token, el sistema por seguridad hace logout y agrega</p>	

el jwt a la blacklist.

Elaboración propia.

Tabla N° 4.58

Tarea de ingeniería: Registrar permiso.

TAREA DE INGENIERÍA	
Número de tarea : 26	Historia de usuario: Registro de permisos (02)
Nombre de tarea : Registrar permiso	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 04
Fecha de inicio : 2018-12-18	Fecha fin : 2018-12-26
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario del sistema (root) selecciona modulo USUARIO del administrador backend del sistema; el sistema despliega un menú de opciones. El usuario en sesión selecciona el menú PERMISOS Y GRUPOS , botón AÑADIR PERMISO , el sistema muestra una interfaz con el formulario « AÑADIR PERMISOS » para agregar la instancia del permiso con los campos (un desplegable con los grupos, un desplegable con los permisos por modelo creados (multi seleccion), un desplegable con los usuarios registrados y un desplegable pre seleccionado con el usuario que está en sesión). El usuario en sesión completa los datos del formulario y hace clic en el botón GRABAR , el sistema guarda los datos en la base de datos y envía un mensaje de “se registró correctamente el permiso.	
Observaciones: Ninguno	

Elaboración propia.

Tabla N° 4.59

Tarea de ingeniería: Mantener permiso.

TAREA DE INGENIERÍA	
Número de tarea : 27	Historia de usuario: Gestión de permisos (03)

Nombre de tarea : Mantener permiso	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 05
Fecha de inicio : 2018-12-26	Fecha fin : 2019-01-09
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona modulo USUARIOS del administrador backend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú PERMISOS Y GRUPOS, el sistema lista las instancias de PERMISOS creadas.</p> <p>Si el usuario en sesión quiere actualizar un PERMISO hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario «MODIFICAR PERMISO», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo los permisos”.</p> <p>Si el usuario en sesión quiere eliminar una PERMISO selecciona en la instancia (s) que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «numero de instancias» registros”.</p>	
<p>Observaciones:</p> <p>Al actualizar una instancia se validan con los mismos parámetros que al crearlo.</p>	

Elaboración propia.

Tabla N° 4.60

Tarea de ingeniería: Mantener usuario.

TAREA DE INGENIERÍA	
Número de tarea : 28	Historia de usuario: Gestión de usuarios (05)
Nombre de tarea : Mantener usuario	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-01-09	Fecha fin : 2019-01-10

Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ
<p>Descripción:</p> <p>El usuario del sistema (root) selecciona modulo USUARIO del administrador backend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú USER, el sistema lista las instancias de USER creadas.</p> <p>Si el usuario en sesión quiere actualizar un USER hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el formulario «MODIFICAR USER», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo la «nombre del user»”.</p> <p>Si el usuario en sesión quiere eliminar una USER selecciona en la instancia (s) que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «numero de instancias» registros”.</p>
<p>Observaciones:</p> <p>Como alternativa a la eliminación de registros, se puede optar por marcar FALSE el field flag. Al actualizar una instancia se validan con los mismos parámetros que al crearlo.</p>

Elaboración propia.

Tabla N° 4.61

Tarea de ingeniería: Mantener sereno.

TAREA DE INGENIERÍA	
Número de tarea : 29	Historia de usuario: Gestión de usuarios (05)
Nombre de tarea : Mantener sereno	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2019-01-10	Fecha fin : 2019-01-14
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (administrador, sereno, personal administrativo) se encuentra</p>	

<p>autenticado en el sistema.</p> <p>El usuario en sesión hace clic en su menú de opciones (superior derecho o superior izquierdo dependiendo de la resolución de su interfaz), el sistema muestra un menú de opciones.</p> <p>El usuario en sesión hace clic en el botón MI PERFIL, el sistema carga la interfaz MI PERFIL con el formulario «MODIFICAR PERFIL», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Perfil actualizado”.</p> <p>El usuario del sistema (administrador) selecciona módulo ADMINISTRACIÓN de la interfaz frontend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú SERENOS, el sistema carga una interfaz con un listado de serenos registrados.</p> <p>Si el usuario (administrador) desea actualizar los datos de otro personal administrativo, hace clic en la instancia de su interés y el sistema carga una interfaz pre cargada con una secuencia de pasos (3) (similar al registro de sereno).</p> <p>Si el usuario (administrador) desea eliminar una SERENO selecciona en la instancia que desea eliminar y en la parte superior marca la acción “Eliminar registros”, el sistema elimina los registros y retorna un mensaje de confirmación “Se eliminó «nombre y alias del sereno»”.</p> <p>Observaciones:</p> <p>Como alternativa a la eliminación de registros, se puede optar por marcar FALSE el field flag. Al actualizar una instancia se validan con los mismos parámetros que al crearlo.</p>
--

Elaboración propia.

C4. CUARTA ITERACIÓN

Tabla N° 4.62

Tarea de ingeniería: Registrar incidente

TAREA DE INGENIERÍA	
Número de tarea : 30	Historia de usuario: Registro de incidentes (18)
Nombre de tarea : Registrar incidente	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02

Fecha de inicio : 2019-01-14	Fecha fin : 2019-01-16
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (personal administrativo, sereno) selecciona modulo INCIDENTE de la aplicación frontend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú AGREGAR, el sistema muestra una interfaz con el formulario «AGREGAR INCIDENTE», el formulario tiene los campos (motivo, delitos presentes (desplegable de los tipos del delito), serenos participantes (desplegable de los serenos registrados), sector (desplegable de los sectores registrados), puesto (desplegable de los puestos registrados), fecha del incidente (pre cargado al día actual), unidad (desplegable con las unidades registradas), hora llamada, hora llegada, hora finalización, descripción del incidente, botón agregar persona involucrada, botón agregar archivo adicional, mapa).</p> <p>El usuario en sesión completa los datos obligatorios del formulario (motivo, serenos participantes, sector, puesto, delitos presentes, descripción, ubicación, fecha, hora (llamada, llegada, final)), hace clic en el botón ENVIAR, la aplicación frontend valida los datos y los envía a la aplicación backend para su segunda validación.</p> <p>Si la segunda validación es correcta, el sistema registra en la base de datos y retorna un mensaje de confirmación “Se registró el incidente con motivo: «motivo del incidente»”.</p>	
<p>Observaciones:</p> <p>Dependiendo de los datos ingresados, el sistema puede desencadenar las tareas de ingeniería N.º 31, 32, 33 y 34.</p> <p>La ubicación es un punto geométrico se utiliza un PointField en la base de datos.</p>	

Elaboración propia.

Tabla N° 4.63

Tarea de ingeniería: Registrar delito (s).

TAREA DE INGENIERÍA	
Número de tarea : 31	Historia de usuario: Registro de incidentes (18)
Nombre de tarea : Registrar delito (s)	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02

Fecha de inicio : 2019-01-16	Fecha fin : 2019-01-18
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El sistema al detectar que se guarda una instancia del tipo INCIDENTE , verifica la cantidad de elementos en la lista de delitos_presentes. El sistema lista los delitos presentes, crea una instancia del tipo INCIDENCIATIPODELITO ; por cada elemento de la lista agrega el uuid de la sgsc , fecha del incidente, hora de llamada , sector , puesto y los registra en la base de datos.	
Observaciones: Ninguno	

Elaboración propia.

Tabla N° 4.64

Tarea de ingeniería: Registrar punto geográfico.

TAREA DE INGENIERÍA	
Número de tarea : 32	Historia de usuario: Registro de incidentes (18)
Nombre de tarea : Registrar punto geográfico	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-01-18	Fecha fin : 2019-01-21
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El sistema al detectar que se guarda una instancia del tipo INCIDENTE , verifica la existencia de la ubicación. El sistema convierte el GEOJSON de ubicación en un PointField , el field se registra en la base de datos.	
Observaciones: El PointField debe pertenecer a un sector que se registró en el incidente. El sector que se elija tiene que pertenecer a la sgsc.	

Elaboración propia.

Tabla N° 4.65

Tarea de ingeniería: Registrar archivo adicional (s).

TAREA DE INGENIERÍA	
Número de tarea : 33	Historia de usuario: Registro de incidentes (18)
Nombre de tarea : Registrar archivo adicional (s)	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-01-21	Fecha fin : 2019-01-22
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El sistema al detectar que se guarda una instancia del tipo INCIDENTE, verifica la cantidad de elementos en la lista de archivos_adicionales.</p> <p>El sistema lista los archivos_adicionales, crea una instancia del tipo INCIDENTEARCHIVOS; por cada elemento de la lista agrega el uuid de la Sgsc, uuid del incidente, formato del archivo, tipo de archivo y nombre, finalmente los guarda en la base de datos.</p> <p>Si el archivo es una imagen, la aplicación backend crea dos réplicas de la imagen con diferentes tamaños (medio, pequeño) para optimizar la carga de imágenes.</p>	
<p>Observaciones:</p> <p>Los archivos se guardan en el servidor de archivos estáticos.</p> <p>La carpeta donde se guardan los archivos tiene la siguiente ruta «uuid_sgsc/imagenes_incidentes».</p>	

Elaboración propia.

Tabla N° 4.66

Tarea de ingeniería: Registrar persona involucrada.

TAREA DE INGENIERÍA	
Número de tarea : 34	Historia de usuario: Registro de incidentes (18)
Nombre de tarea : Registrar persona involucrada	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2019-01-22	Fecha fin : 2019-01-24

Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ
<p>Descripción:</p> <p>El sistema al detectar que se guarda una instancia del tipo INCIDENTE, verifica la cantidad de elementos en la lista de personas_involucradas.</p> <p>El sistema lista las personas_involucradas, crea una instancia del tipo PERSONA INVOLUCRADA, por cada elemento de la lista agrega el dni (si es que se obtiene), apellido paterno, apellido materno, nombre, edad, alias, situación y descripción, finalmente los guarda en la base de datos.</p> <p>Si la instancia tiene el campo dni, la aplicación frontend verifica el nombre y apellidos por medio de un servicio de tercero.</p>
<p>Observaciones:</p> <p>Las instancias de persona involucrada se guardan sin relación a la sgsc.</p> <p>Para mantener relación a la sgsc, se crea un modelo intermedio.</p>

Elaboración propia.

Tabla N° 4.67

Tarea de ingeniería: Mantener incidente

TAREA DE INGENIERÍA	
Número de tarea : 35	Historia de usuario: Gestión de incidentes (19)
Nombre de tarea : Mantener incidente	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2019-01-24	Fecha fin : 2019-01-28
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (administrador, sereno) selecciona modulo INCIDENTE de la aplicación frontend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú LISTAR, el sistema lista las instancias de INCIDENTE registradas (si es un sereno, el sistema muestra los incidentes en los que participo; en caso contrario lista todos los incidentes de su sub gerencia).</p> <p>Si el usuario en sesión quiere actualizar un INCIDENTE hace clic en la instancia que desea</p>	

actualizar, el sistema muestra una interfaz con el formulario «**ACTUALIZAR INCIDENTE**», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón **ACTUALIZAR**, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo el incidente: «**motivo del incidente**»”.

Si el usuario en sesión quiere eliminar una **INCIDENTE** selecciona la instancia que desea eliminar y selecciona la opción para eliminar, el sistema elimina pide confirmación y según la respuesta elimina la instancia “Se eliminó el incidente «**motivo del incidente**» ”.

Observaciones:

Si se modifica un incidente, la aplicación backend hace una copia de la instancia inicial y lo guarda en el historial.

Los incidentes no se eliminan, se ocultan para la gestión de la sub gerencia y se almacenan como historial.

Elaboración propia.

Tabla N° 4.68

Tarea de ingeniería: Detallar incidente.

TAREA DE INGENIERÍA	
Número de tarea : 36	Historia de usuario: Registro de incidentes (18)
Nombre de tarea : Detallar incidente	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-01-28	Fecha fin : 2019-01-29
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (administrador personal administrativo, sereno) selecciona modulo INCIDENTE de la aplicación frontend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú LISTAR, el sistema lista las instancias de INCIDENTE registradas (si es un sereno, el sistema muestra los incidentes en los que participo; en caso contrario lista todos los incidentes de su sub gerencia).</p> <p>Si el usuario en sesión selecciona la instancia a detallar, el sistema muestra una interfaz</p>	

« DETALLE INCIDENTE », con los datos del incidente. El punto se representa con un mapa y el sector que lo engloba, si se tiene archivos adicionales, el sistema los muestra en formato pequeño, reproductor de video o sonido.
Observaciones: Ninguno

Elaboración propia.

C5. QUINTA ITERACIÓN

Tabla N° 4.69

Tarea de ingeniería: Registrar base patrullaje.

TAREA DE INGENIERÍA	
Número de tarea : 37	Historia de usuario: Registro de base patrullaje (21)
Nombre de tarea : Registrar base patrullaje	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 05
Fecha de inicio : 2019-01-29	Fecha fin : 2019-01-05
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (personal administrativo) selecciona modulo PATRULLAJE de la aplicación frontend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú AGREGAR BASE, el sistema muestra una interfaz con el formulario «AGREGAR BASE», el formulario tiene un calendario para seleccionar la fecha de inicio y fin del periodo de base.</p> <p>El usuario en sesión, selecciona el periodo base, el sistema muestra el botón de consultar.</p> <p>El usuario en sesión, hace clic en el botón “CONSULTAR”, el sistema consulta a la aplicación backend por la disponibilidad de serenos que pueden hacer patrullaje y muestra una tabla (nombre, cargo y alias) con la relación de serenos disponibles.</p> <p>El usuario en sesión, selecciona el sereno que desea agregar a la base, el sistema muestra una modal con las opciones (turno, días descanso, cargo); se completa los datos y el sereno se agrega a memoria.</p> <p>El usuarios en sesión, hace clic en el botón “VERIFICAR Y REGISTRAR BASE”, el</p>	

sistema valida los datos, registra en la base de datos y muestra un mensaje con el resultado.

Observaciones:

El sistema valida que cada turno tiene como un mínimo un sereno.

Las fechas de descanso de los serenitos deben estar en el periodo de base.

La fecha mínima del periodo base debe ser la última del periodo base que le precede.

Elaboración propia.

Tabla N° 4.70

Tarea de ingeniería: Integrar redux con el árbol de base patrullaje.

TAREA DE INGENIERÍA	
Número de tarea : 38	Historia de usuario: Registro de base patrullaje (21)
Nombre de tarea : Integrar redux con el árbol de base patrullaje	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 03
Fecha de inicio : 2019-02-05	Fecha fin : 2019-02-08
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El sistema detecta que se está registrando una BASE PATRULLAJE , crea un árbol con el primer nodo (rango del periodo base) y por medio de REDUX los agrega al STORE. Cada vez que agrega o quita un sereno de la base patrullaje, redux modifica el store para actualizar el árbol de base patrullaje.	
Observaciones: Ninguno	

Elaboración propia.

Tabla N° 4.71

Tarea de ingeniería: Mantener base patrullaje.

TAREA DE INGENIERÍA	
Número de tarea : 39	Historia de usuario: Gestión de base patrullaje (22)
Nombre de tarea : Mantener base patrullaje	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 05

Fecha de inicio : 2019-02-08	Fecha fin : 2019-02-15
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario del sistema (personal administrativo) selecciona modulo PATRULLAJE de la aplicación frontend; la aplicación despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú BASES, el sistema lista las instancias de BASE PATRULLAJE.</p> <p>Si el usuario en sesión quiere actualizar una BASE PATRULLAJE hace clic en la instancia que desea actualizar, el sistema muestra una interfaz con el árbol de base patrullaje y el formulario «MODIFICAR BASE PATRULLAJE», el formulario se encuentra pre cargado con datos iniciales (datos actuales de la instancia); se actualizan los datos y hace clic en el botón ACTUALIZAR, el sistema actualiza los datos en la base de datos y retorna un mensaje de confirmación “Se actualizo la base patrullaje del «fecha inicio – fecha fin»”.</p> <p>Si el usuario en sesión quiere eliminar un BASE PATRULLAJE selecciona en la instancia que desea eliminar y marca el botón “Eliminar”, el sistema elimina el registros y retorna un mensaje de confirmación “Se eliminó la base patrullaje del «fecha inicio – fecha fin» ”.</p>	
<p>Observaciones:</p> <p>No se puede eliminar un base patrulla que ya paso su periodo.</p> <p>No se puede modificar un base patrulla que ya paso su periodo.</p>	

Elaboración propia.

C6. SEXTA ITERACIÓN

Tabla N° 4.72

Tarea de ingeniería: Generar mapa de incidente.

TAREA DE INGENIERÍA	
Número de tarea : 40	Historia de usuario: Gestión mapa de incidentes (20)
Nombre de tarea : Generar mapa de incidente	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 04
Fecha de inicio : 2019-02-15	Fecha fin : 2019-02-21

Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ
<p>Descripción:</p> <p>El usuario del sistema (personal administrativo, administrador) selecciona modulo INCIDENTE de la aplicación frontend; el sistema despliega un menú de opciones.</p> <p>El usuario en sesión selecciona el menú MAPA INCIDENTE, el sistema muestra una interfaz con un mapa a lado izquierdo y 3 desplegables (tipos delito, serenos participantes, sectores) a lado derecho.</p> <p>El sistema clusteriza las ubicaciones de los incidentes, dependiendo del nivel de zoom que se tenga.</p> <p>Si se tiene el máximo zoo, desaparece la clusterización y se muestran los markers por individual.</p> <p>El usuarios en sesión, selecciona un marker, el sistema carga el detalle del incidente (motivo, tipo delito, sereno participante, sector, hora, fecha y detalle) en el lado derecho.</p>
<p>Observaciones:</p> <p>Por default se cargan los incidente de los últimos 365 (1 año) días.</p>

Elaboración propia.

Tabla N° 4.73

Tarea de ingeniería: Filtrar mapa de incidente.

TAREA DE INGENIERÍA	
Número de tarea : 41	Historia de usuario: Gestión mapa de incidentes (20)
Nombre de tarea : Filtrar mapa de incidente	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2019-02-21	Fecha fin : 2019-02-25
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario (personal administrativo, administrador) se encuentra en la interfaz MAPA INCIDENTE, del menú INCIDENTE.</p> <p>El usuario en sesión selecciona cualquiera de los desplegables (tipos delito, serenos participantes, sectores) y selecciona una o varias de sus opciones, el sistema filtra los</p>	

resultados y actualiza las ubicaciones cumpliendo el filtro ingresado por el usuario.
El sistema hace el filtro en memoria y no consulta a la aplicación backend.

Observaciones:

Se tiene en memoria solo opciones que puede satisfacer (delitos que fueron registrados y no todo el código penal).

Elaboración propia.

Tabla N° 4.74

Tarea de ingeniería: Integrar redux con el mapa de incidente.

TAREA DE INGENIERÍA	
Número de tarea : 42	Historia de usuario: Gestión mapa de incidentes (20)
Nombre de tarea : Integrar redux con el mapa de incidente	
Tipo de tarea : Desarrollo frontend	Puntos estimados (días) : 02
Fecha de inicio : 2019-02-25	Fecha fin : 2019-02-27
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El sistema detecta que se está filtrando los incidentes en la interfaz MAPA INCIDENTES. El sistema crea el cluster con los incidentes que se tiene en memoria y por medio de REDUX los agrega al STORE. El sistema recorre los incidentes custerizados y los agrega al mapa (markers y geojson). Cada vez que filtra, redux modifica el store para actualizar el cluster de incidentes.</p>	
<p>Observaciones:</p> <p>Se utiliza redux para disminuir la carga del sistema (menos consumo de memoria).</p>	

Elaboración propia.

Tabla N° 4.75

Tarea de ingeniería: Generar nube de delitos

TAREA DE INGENIERÍA	
Número de tarea : 43	Historia de usuario: Gestión de tipos de delito (23)

Nombre de tarea : Generar nube de delitos	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2019-02-27	Fecha fin : 2019-03-01
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario del sistema (personal administrativo, administrador) selecciona modulo LEGAL de la aplicación frontend; la aplicación despliega un menú de opciones. El usuario en sesión selecciona el menú ESTADÍSTICAS , el sistema muestra una interfaz con una nube de palabras (tipos de delito), la nube de delitos permite visualizar de forma dinámica la presencia de tipos de delito. El usuarios en sesión, hace clic a las opciones superiores (incidentes, ocurrencias, patrullaje), el sistema procesa la petición y actualiza la nube de delitos cambiando el repositorio de origen al proveedor seleccionado.	
Observaciones: Se utiliza redux para disminuir la carga del sistema (menos consumo de memoria).	

Elaboración propia.

Tabla N° 4.76

Tarea de ingeniería: Filtrar nube de delitos (dashboard).

TAREA DE INGENIERÍA	
Número de tarea : 44	Historia de usuario: Gestión de tipos de delito (23)
Nombre de tarea : Filtrar nube de delitos (dashboard)	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 02
Fecha de inicio : 2019-03-01	Fecha fin : 2019-03-05
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario (personal administrativo, administrador) se encuentra en la interfaz ESTADÍSTICAS , del menú LEGAL de la aplicación frontend. El usuario en sesión hace clic en el tipo de delito, el sistema procesa la petición y actualiza la	

nube de delitos. Si en memoria se tiene un tipo de delito seleccionado, el sistema hace cálculos y muestra cuadros estadísticos. 1. La presencia del delito en los sectores 2. La presencia del delito en los días de la semana 3. la presencia del delito en las horas del día (intervalos de 3 horas). Si no se tiene el detalle en memoria, la aplicación solicita datos de la aplicación backend.
Observaciones: Ninguno

Elaboración propia.

Tabla N° 4.77

Tarea de ingeniería: Integrar redux con la nube de delitos.

TAREA DE INGENIERÍA	
Número de tarea : 45	Historia de usuario: Gestión de tipos de delito (23)
Nombre de tarea : Integrar redux con la nube de delitos	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-03-06	Fecha fin : 2019-03-07
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El sistema detecta que se está filtrando los delitos en la interfaz ESTADÍSTICAS del menú LEGAL . El sistema crea una lista con los delitos que se tiene en memoria y por medio de REDUX los agrega al STORE . Cada vez que existe variación en el delito seleccionado, redux modifica el store para actualizar la lista de delitos y generar los datos estadísticos que la interfaz muestra. Si no se tiene los datos suficiente, la aplicación consulta a la aplicación backend por nuevos datos.	
Observaciones: Se utiliza redux para disminuir la carga del sistema (menos consumo de memoria).	

Elaboración propia.

Tabla N° 4.78

Tarea de ingeniería: Generar cuadros de antijuridicidad

TAREA DE INGENIERÍA	
Número de tarea : 46	Historia de usuario: Gestión de tipos de delito (23)
Nombre de tarea : Generar cuadros de antijuridicidad	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-03-06	Fecha fin : 2019-03-07
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El usuario (personal administrativo, administrador) se encuentra en la interfaz ANTI JURIDICIDAD, del menú INCIDENTE de la aplicación frontend.</p> <p>El usuario en sesión visualiza la cantidad de incidentes registrados, tipos de delito, antijuridicidad y culpabilidad, el sistema carga en memoria un resumen de los incidentes registrados.</p> <p>El sistema hace cálculos y muestra cuadros estadísticos.</p> <ol style="list-style-type: none"> 1. La frecuencia de incidentes en los sectores. 2. La frecuencia de incidentes antijurídicos en los sectores. 3. La frecuencia de incidentes no antijurídicos en los sectores por día. 4. La frecuencia de incidentes en los sectores por día. 5. La frecuencia de incidentes antijurídicos en los sectores por día. 6. La frecuencia de incidentes no antijurídicos en los sectores por día. <p>Si no se tiene los datos suficientes, la aplicación consulta a la aplicación backend por nuevos datos.</p>	
Observaciones: Ninguno	

Elaboración propia.

Tabla N° 4.79

Tarea de ingeniería: Integrar redux a los cuadros de antijuridicidad.

TAREA DE INGENIERÍA

Número de tarea : 47	Historia de usuario: Gestión de tipos de delito (23)
Nombre de tarea : Integrar redux a los cuadros de antijuridicidad	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-03-07	Fecha fin : 2019-03-08
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El sistema detecta que se está visualizando los incidentes antijurídicos en la interfaz ANTI JURIDICIDAD del menú INCIDENTE . El sistema crea una lista con los incidentes antijurídicos que se tiene en memoria y por medio de REDUX los agrega al STORE . Para cada cuadro estadístico, redux hace los cálculos necesarios y muestra los cuadros, creando dependencia de la lista original. Si la aplicación solicita nuevos datos, redux actualiza la lista y por ende, todos los cuadros son actualizados.	
Observaciones: Se utiliza redux para disminuir la carga del sistema (menos consumo de memoria).	

Tabla 4.77, Tarea de ingeniería: Integrar redux con la nube de delitos

Tabla N° 4.80

Tarea de ingeniería: Generar cuadros de culpabilidad.

TAREA DE INGENIERÍA	
Número de tarea : 48	Historia de usuario: Gestión de tipos de delito (23)
Nombre de tarea : Generar cuadros de culpabilidad	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-03-08	Fecha fin : 2019-03-11
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
Descripción: El usuario (personal administrativo, administrador) se encuentra en la interfaz CULPABILIDAD , del menú INCIDENTE de la aplicación frontend.	

<p>El usuario en sesión visualiza la cantidad de incidentes registrados, tipos de delito, antijuridicidad y culpabilidad, el sistema carga en memoria un resumen de los incidentes registrados.</p> <p>El sistema hace cálculos y muestra cuadros estadísticos.</p> <ol style="list-style-type: none"> 1. La frecuencia de culpabilidad en los sectores. 2. La frecuencia de culpabilidad antijurídicos en los sectores. 3. La frecuencia de culpabilidad no antijurídicos en los sectores por día. 4. La frecuencia de culpabilidad en los sectores por día. 5. La frecuencia de culpabilidad antijurídicos en los sectores por día. 6. La frecuencia de culpabilidad no antijurídicos en los sectores por día. <p>Si no se tiene los datos suficientes, la aplicación consulta a la aplicación backend por nuevos datos.</p>
<p>Observaciones: Ninguno</p>

Elaboración propia.

Tabla N° 4.81

Tarea de ingeniería: Integrar redux a los cuadros de culpabilidad

TAREA DE INGENIERÍA	
Número de tarea : 49	Historia de usuario: Gestión de tipos de delito (23)
Nombre de tarea Integrar redux a los cuadros de culpabilidad	
Tipo de tarea : Desarrollo	Puntos estimados (días) : 01
Fecha de inicio : 2019-03-11	Fecha fin : 2019-03-12
Programador responsable : JUNIOR GROVER FLORES MARTÍNEZ	
<p>Descripción:</p> <p>El sistema detecta que se está visualizando los incidentes antijurídicos en la interfaz CULPABILIDAD del menú INCIDENTE.</p> <p>El sistema crea una lista con los incidentes con culpabilidad que se tiene en memoria y por medio de REDUX los agrega al STORE.</p> <p>Para cada cuadro estadístico, redux hace los cálculos necesarios y muestra los cuadros, creando dependencia de la lista original.</p>	

Si la aplicación solicita nuevos datos, redux actualiza la lista y por ende, todos los cuadros son actualizados.

Observaciones:

Se utiliza redux para disminuir la carga del sistema (menos consumo de memoria).

Elaboración propia.

D. INTERFACES DE USUARIO

Figura N° 4.4

Añadir sgsc

Id:

Nombre:

Mision:

Vision:

Logo: Ningún archivo seleccionado

Area:

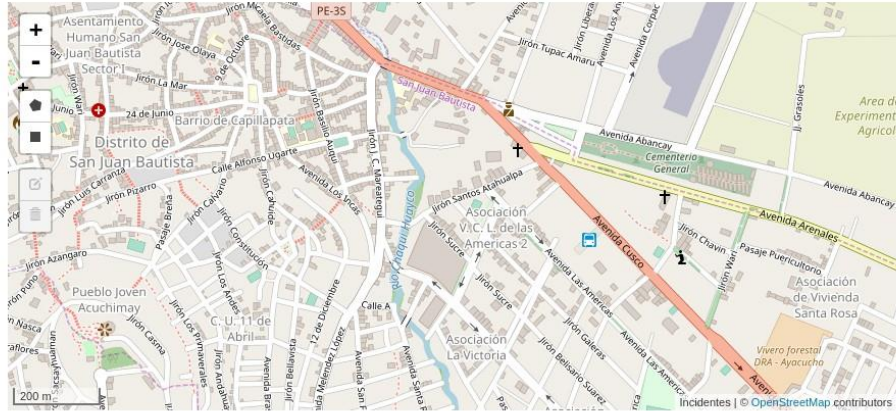


Imagen: Ningún archivo seleccionado

Registro:

☒ Flag

Interfaz de usuario (backend): Registrar sub gerencia.

Elaboración propia.

Figura N° 4.5

Interfaz de usuario (backend): Registrar sector.

Nombre:

Color:




Imagen:

Seleccionar archivo







Ningún archivo seleccionado


Sgsc:

Municipalidad Distrital de Jesús Nazareno



Sector:





50 m

Incidentes | © OpenStreetMap contributors

Jirón Huascar

Jirón Tupac Amaru

Jirón Cangallo

Jirón Pachacutec

Jirón Los Andes

Jirón Basilio Auqui

Jirón Manco Capac

Jirón Bellavista

Jirón Palma

Jirón Los Pedregales

Jirón Loreto

Jirón Perene

Jirón Apurímac

Jirón Sapocsa

Avenida Meléndez López

Avenida San Francisco

Avenida Brasil

Poligación Santa Rosa




María Parado de Belido

C. U. I. Garrilazo

72 de Diciembre

Registro:

rot



☒ Flag

Elaboración propia.

Figura N° 4.6

Interfaz de usuario (backend): Registrar sub sector.

Id:

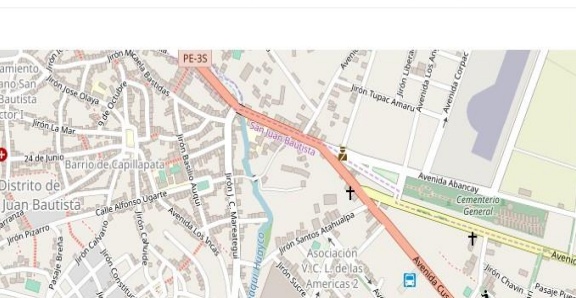
Nombre:

Color:

Imagen: Ningún archivo seleccionado

Sector: **SECTOR III - Municipalidad Distrital de Jesús Nazareno**

Subsector:



Elaboración propia.

Figura N° 4.7

Interfaz de usuario (backend): Registrar puesto.

Añadir puesto

Id:

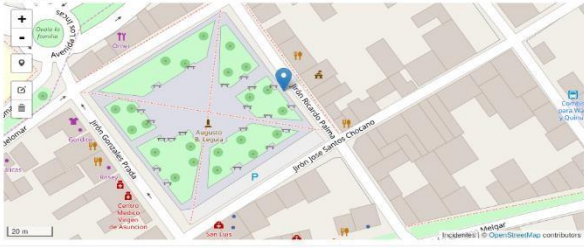
Nombre:

Color:

Imagen: Ningún archivo seleccionado

Sector:

Sgsc:

Ubicación: 

Registro:

☒ Invisible

☒ Flag

Elaboración propia.

Figura N° 4.8

Interfaz de usuario (backend): Mantener sub gerencia.

Modificar sgsc HISTÓRICO

Id:

Nombre:

Misión:

Vision:

Logo: Actualmente: sgsc_e0f68033-5d48-11e9-84cb-2c600c46499/logo/JESUS_NAZARENO.jpg
 Modificar: Ningún archivo seleccionado

Area: 

Imagen: Actualmente: sgsc_e0f68033-5d48-11e9-84cb-2c600c46499/logo/JESUS_NAZARENO_usuario.jpg
 Modificar: Ningún archivo seleccionado

Registro:

☒ Flag

Elaboración propia.




Figura N° 4.9

Interfaz de usuario (backend): Mantener sector.

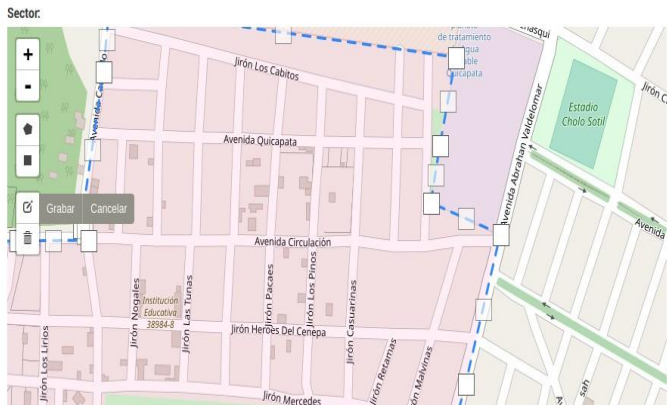
Nombre:

Color:

Imagen: Ningún archivo seleccionado

Sgsc:   

Sector:



Elaboración propia.




Figura N° 4.10

Interfaz de usuario (backend): Mantener sub sector.

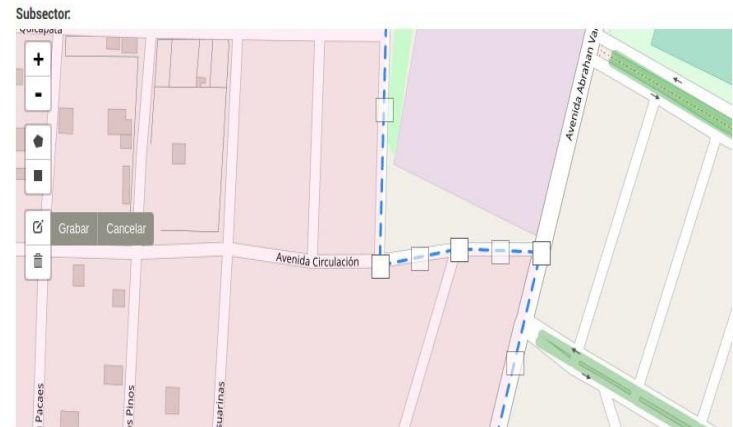
Nombre:

Color:

Imagen: Ningún archivo seleccionado

Sector:   

Subsector:



Elaboración propia.

Interfaz de usuario (backend): Mantener puesto.

Nombre:	<input type="text" value="PALACIO"/>		
Color:	<input type="text"/>		
Imagen:	<input type="button" value="Seleccionar archivo"/> Ningún archivo seleccionado		
Sector:	<input type="button" value="SECTOR II - Municipalidad Distrital de Jesús Nazareno"/> 📍 + ✖		
Sgsc:	<input type="button" value="Municipalidad Distrital de Jesús Nazareno"/> 📍 + ✖		
Ubicación:			

Elaboración propia.

Figura N° 4.12

Interfaz de usuario (backend): Registrar código penal.

Añadir tipo delito

Artículo:

Nombre:

Subcategoría:

HOMICIDIO

+

×

Descripción:

Estado:

VIGENTE

Numcasos:

0

☒ Flag

Elaboración propia.

Figura N° 4.13
Interfaz de usuario (backend): Registrar turno.

Añadir turno

Id:	ee798d11622a11e989622cf	
Nombre:	<input type="text"/>	
Hora inicio:	6:00:00	Ahora 🕒
Hora fin:	14:00:00	Ahora 🕒
<input checked="" type="checkbox"/> Isvigente		
Sgsc:	Municipalidad Distrital de Jesús Nazareno ▼	✎ + ✖
Registro:	rot ▼	✎ + ✖
<input checked="" type="checkbox"/> Flag		

Elaboración propia.

Figura N° 4.14
Interfaz de usuario (backend): Registrar cargo.

Añadir cargo

Id:	1cf6f8e5622b11e989622cf	
Nombre:	<input type="text"/>	
<input type="checkbox"/> Isvehiculo		
Unidad:	Municipalidad Distrital de Jesús Nazareno - Optimus ▼	✎ + ✖
<input type="checkbox"/> Isadministrativo		
<input type="checkbox"/> Canpatrullaje		
<input type="checkbox"/> Caincidente		
Sgsc:	Municipalidad Distrital de Jesús Nazareno ▼	✎ + ✖
<input checked="" type="checkbox"/> Flag		

Elaboración propia.

Figura N° 4.15

Interfaz de usuario (backend): Mantener marco legal.

Modificar tipo delito
HISTÓRICO

Artículo:
ARTÍCULO 121 - B

Nombre:
LESIONES GRAVES POR VIOLENCIA CONT

Subcategoria:
LESIONES

Descripcion:

En los supuestos previstos en el primer párrafo del artículo 121 se aplica pena privativa de libertad no menor de seis ni mayor de doce años e inhabilitación conforme a los numerales 5 y 11 del artículo 36 del presente Código y los artículos 75 y 77 del Código de los Niños y Adolescentes, según corresponda, cuando: 1. La víctima es mujer y es lesionada por su condición de tal en cualquiera de los contextos previstos en el primer párrafo del artículo 108-B. 2. La víctima se encuentra en estado de gestación. 3. La víctima es cónyuge; excónyuge; conviviente; exconviviente; padrastro; madrastra; ascendiente o descendiente por consanguinidad, adopción o afinidad; pariente colateral del cónyuge y conviviente hasta el cuarto grado de consanguinidad y segundo de afinidad; habita en el mismo hogar, siempre que no medien relaciones contractuales o laborales; o es con quien se ha procreado hijos en común, independientemente de que se conviva o no al momento de producirse los actos de violencia, o la violencia se da en cualquiera de los contextos de los numerales 1, 2 y 3 del primer párrafo del artículo 108-B. 4. La víctima mantiene cualquier tipo de relación de dependencia o subordinación sea de autoridad, económica, cuidado, laboral o contractual y el agente se hubiera aprovechado de esta situación. 5. Para cometer el delito se hubiera utilizado cualquier tipo de arma, objeto contundente o instrumento que ponga en riesgo la vida de la víctima.

Estado:
VIGENTE

Numcasos:
329

☒ Flag

Elaboración propia.

Figura N° 4.16

Interfaz de usuario (backend): Mantener turno.

Modificar turno
HISTÓRICO

Id:
f7a22b315d4911e984cb2c6

Nombre:
MAÑANA

Hora inicio:
06:00:00
Ahora

Hora fin:
14:00:00
Ahora

☒ Isvigente

Sgsc:
Municipalidad Distrital de Jesús Nazareno

Registro:
r o t

☒ Flag

Elaboración propia.

Figura N° 4.17
Interfaz de usuario (backend): Mantener cargo.

Modificar cargo HISTÓRICO

Id: e4865edb5d4a11e984cb2cf

Nombre: Sereno

☐ Isvehiculo

Unidad: ----- ✎ + ✖

☐ Isadministrativo

☒ Canpatrullaje

☒ Cainscidente

Sgsc: Municipalidad Distrital de Jesús Nazareno ✎ + ✖

☒ Flag

Elaboración propia.

Figura N° 4.18
Interfaz de usuario (frontend): Registrar sereno – paso 1.

Proceso registro de usuarios
Cada paso define un nivel de seguridad para el usuario que se registra, recuerde que el uso de contraseñas es exclusivamente personal !.

AUTENTICACION **DATOS PERSONALES** **CARGO EN LA SGSC**

Verificación de datos, los datos del usuario son únicos.

NOMBRE DE USUARIO * **ALIAS ***

jujunior Flores

CORREO ELECTRÓNICO *

hello@correoinstitucional.com

* Campos requeridos

Siguiente

Elaboración propia.

Figura N° 4.19

Interfaz de usuario (frontend): Registrar sereno – paso 2.

Proceso registro de usuarios

Cada paso define un nivel de seguridad para el usuario que se registra, recuerde que el uso de contraseñas es exclusivamente personal !.

AUTENTICACION DATOS PERSONALES CARGO EN LA SGSC

Ingrese los datos personales del usuario.

NOMBRE *
Nombre

APELL PATERNO *
Apellido

APELL MATERNO *
Apellido

DOCUMENTO DE IDENTIDAD *
01234567

NACIMIENTO *
Fecha de nacimiento

SEXO *
Select...

NUMERO DE CONTACTO *
966 000 000

DESCRIPCION
Ingrese una descripcion del sereno

Previo Siguiete

Elaboración propia.

Figura N° 4.20

Interfaz de usuario (frontend): Registrar sereno – paso 3.

Proceso registro de usuarios

Cada paso define un nivel de seguridad para el usuario que se registra, recuerde que el uso de contraseñas es exclusivamente personal !.

AUTENTICACION DATOS PERSONALES CARGO EN LA SGSC

Ingrese los datos con respecto al cargo del sereno.

CARGO *
Select...

ADMINISTRATIVO INCIDENTE PATRULLA

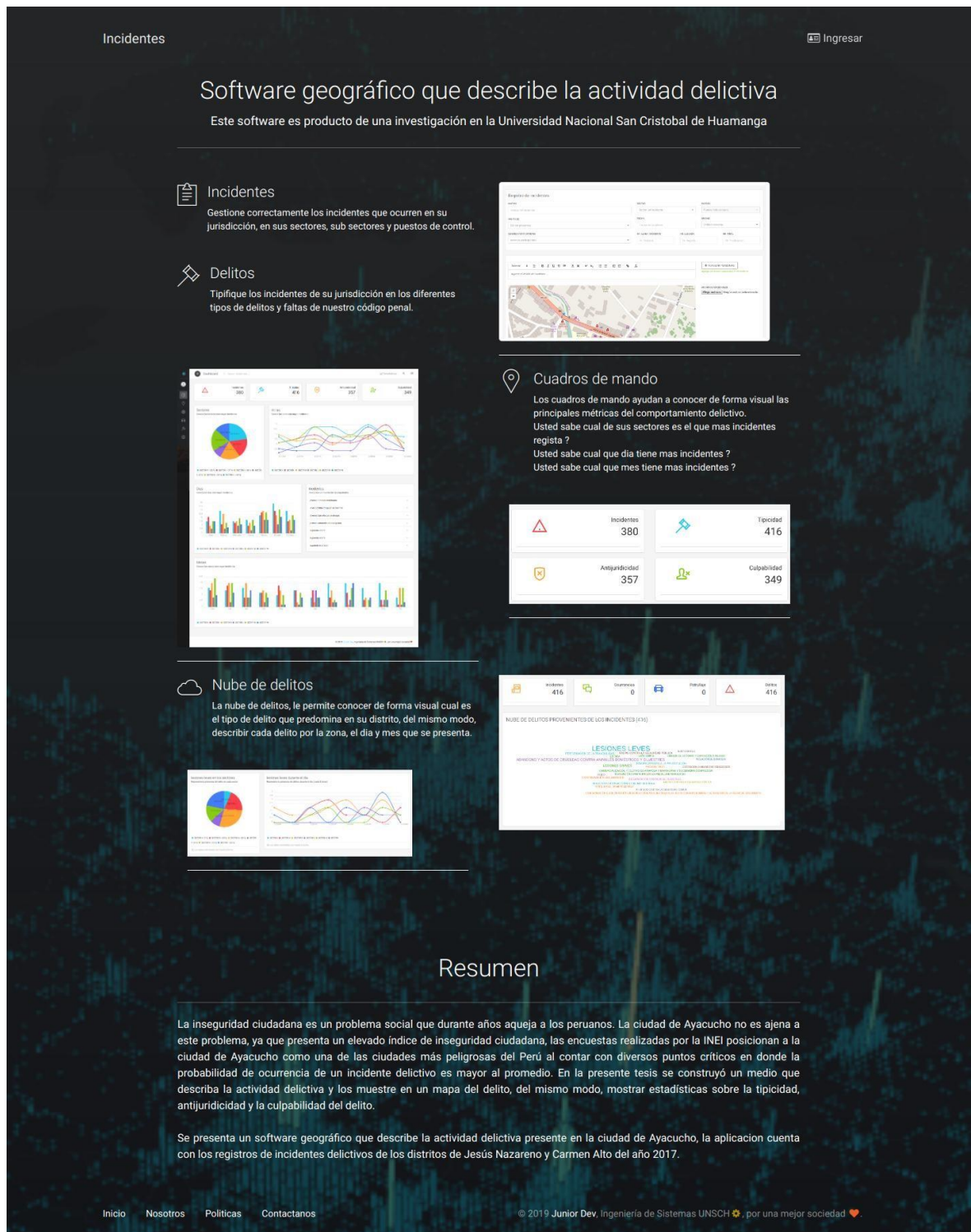
FECHA ASIGNACIÓN *
Fecha de nacimiento

DESCRIPCION
Ingrese una descripcion del sereno

Previo Registrar

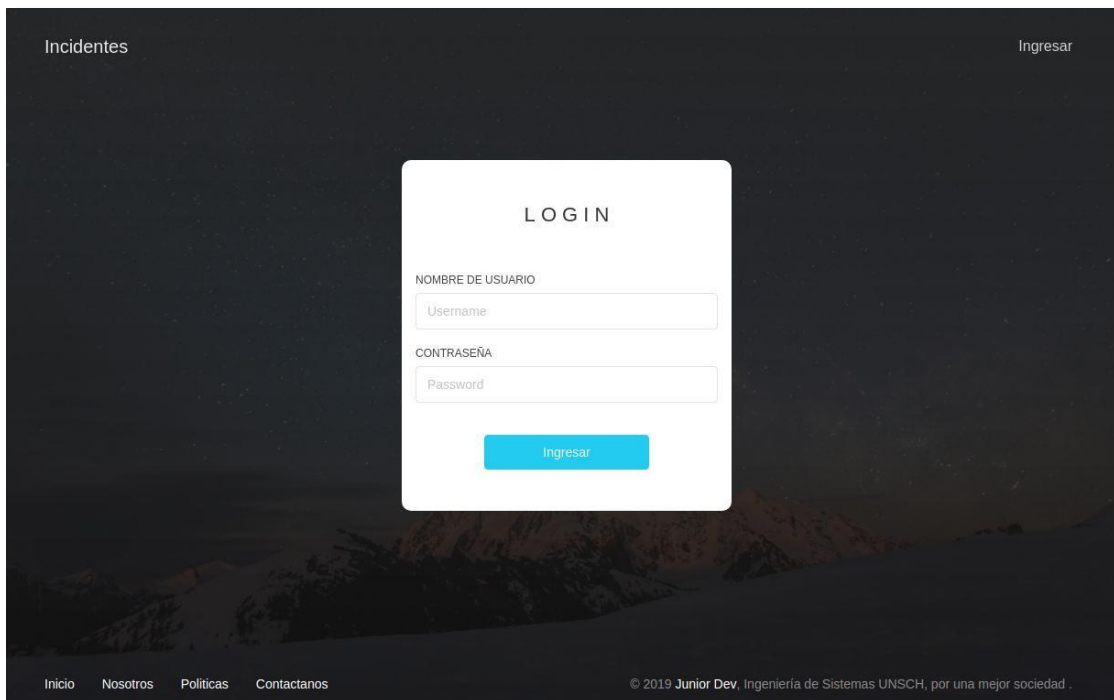
Elaboración propia.

Figura N° 4.21
Interfaz de usuario (frontend): Inicio.



Elaboración propia.

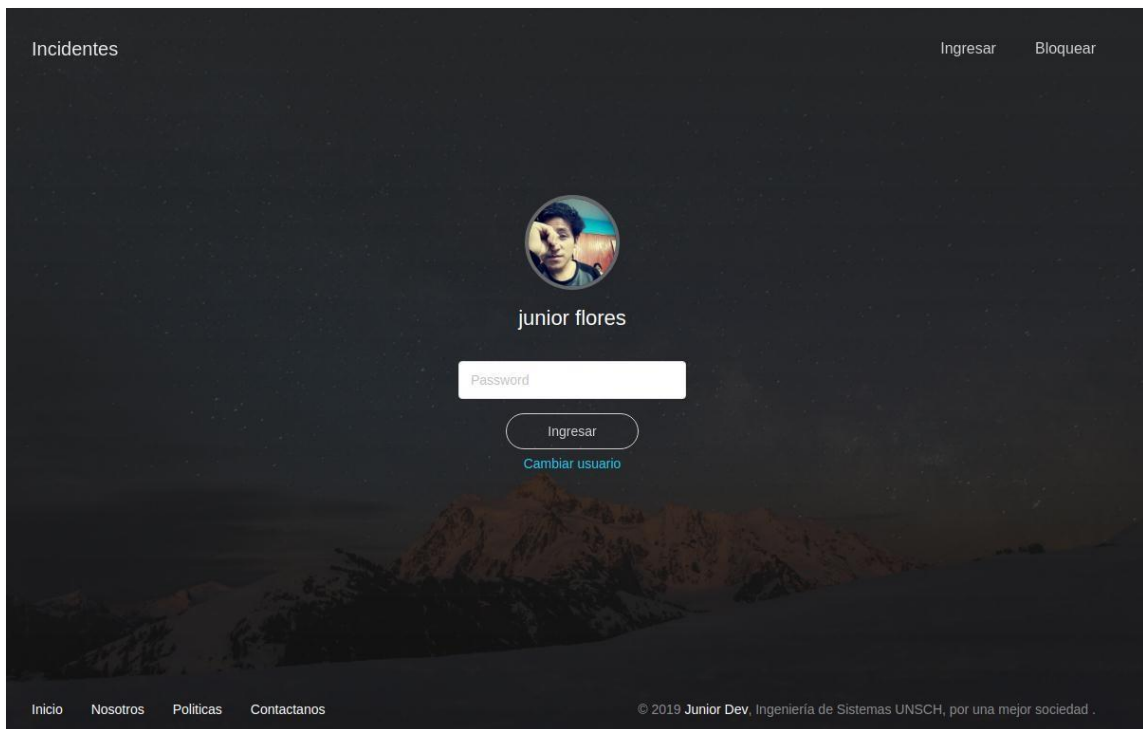
Figura N° 4.22
Interfaz de usuario (frontend): Login usuario.



The screenshot shows a web application with a dark background featuring a mountain landscape. At the top left is the text "Incidentes" and at the top right is "Ingresar". In the center, there is a white login form titled "LOGIN". The form contains two input fields: "NOMBRE DE USUARIO" with a placeholder "Username" and "CONTRASEÑA" with a placeholder "Password". Below these fields is a blue button labeled "Ingresar". At the bottom of the page, there is a navigation bar with links "Inicio", "Nosotros", "Políticas", and "Contactanos", and a copyright notice "© 2019 Junior Dev, Ingeniería de Sistemas UNSCH, por una mejor sociedad."

Elaboración propia.

Figura N° 4.23
Interfaz de usuario (frontend): Lock usuario.



The screenshot shows the same web application as Figure 4.22, but in a "lock" state. At the top right, there are two links: "Ingresar" and "Bloquear". In the center, there is a circular profile picture of a man, followed by the name "junior flores". Below the name is a white input field labeled "Password". Under the password field is a blue button labeled "Ingresar" and a blue link labeled "Cambiar usuario". The bottom navigation bar and copyright notice are identical to the previous figure.

Elaboración propia

Figura N° 4.24

Interfaz de usuario (frontend): Mantener usuario.

Elaboración propia.

Figura N° 4.25

Interfaz de usuario (frontend): Registrar incidente.

Elaboración propia.

Figura N° 4.26

Interfaz de usuario (frontend): Registrar incidente – agregar persona.

Personas Involucradas

Se encontro los datos de JUNIOR GROVER

DNI DEL SUP.: 70804143

SITUACION DEL SUP.: Situacion del supuesto

SITUACION (OTRO): Ingrese el dni del supue

NOMBRE: JUNIOR GROVER

APE. PATERNO: FLORES

APE. MATERNO: MARTINEZ

EDAD: 0

ALIAS: Ingrese el alias del sup

DIRECCION: Ingrese el alias del sup

DETALLES EXTRA DEL SUPUESTO: Ingrese algun detalle extra del supuesto

Cerrar Agregar

Elaboración propia

Figura N° 4.27

Interfaz de usuario (frontend): Lista de incidentes.

Listar Buscar incidentes... Estadísticas

Listado de los Incidentes segun el codigo penal

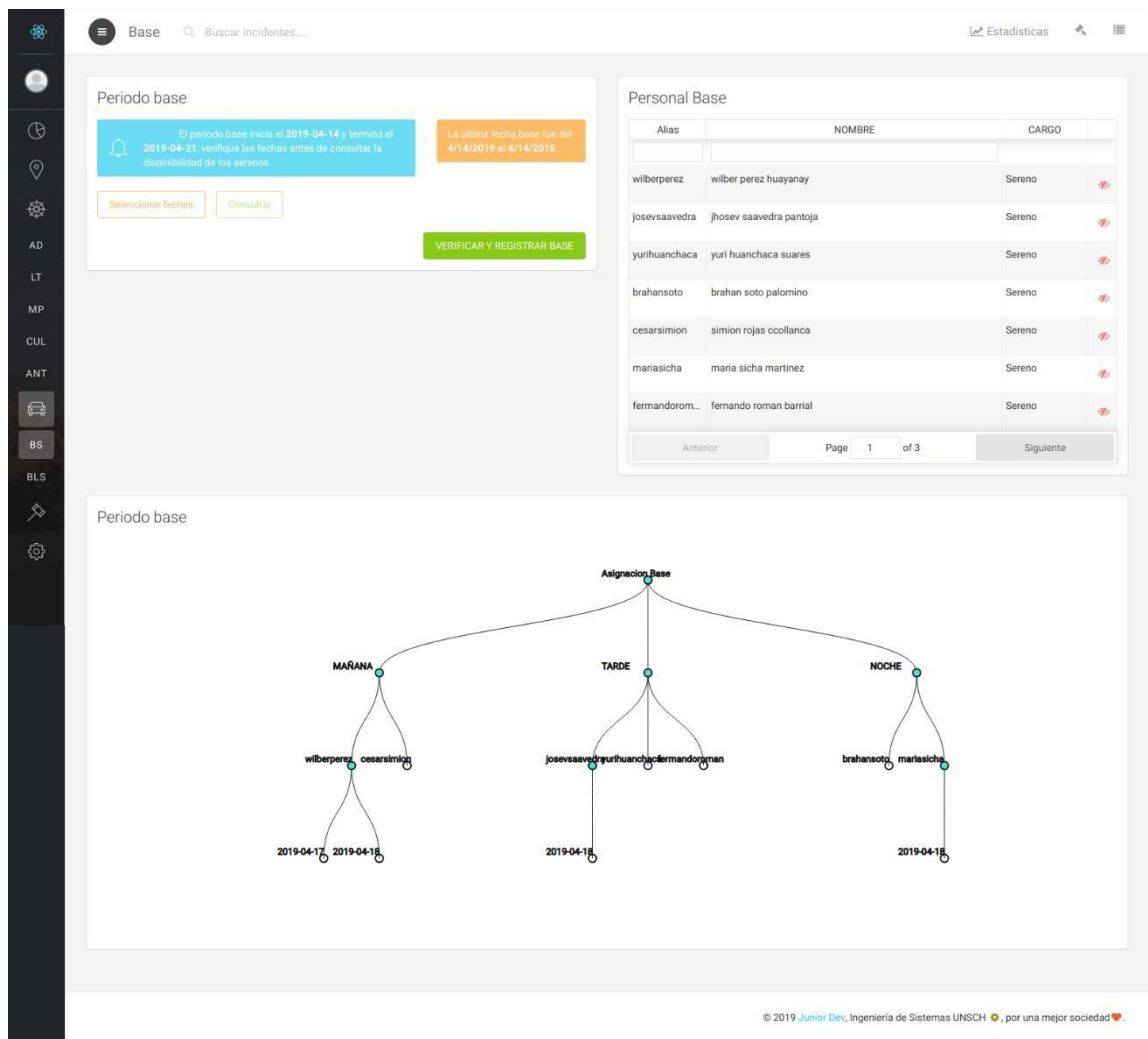
CODIGO	MOTIVO	DESCRIPCION	SECTOR	FECHA
20	violencia familiar	en momento que realizaba el patrullaje se recibio una llamada de un vecino quien dijo que habia u...	SECTOR II	10/02/2017
19	hurto	En momento que se realizaba el servicio se apersono a la base castillo el sr. yuri navarro carbajal (...)	SECTOR IV	10/01/2017
18	pandillaje	En momento que se realizaba el cambio de turno. el personal de base nos comunica que en la dire...	SECTOR IV	08/01/2017
17	violencia familiar	En momento que se realizaba patrullaje se informo por parte de un vecino de un supuesto caso de...	SECTOR VI	08/01/2017
16	accidente de transito	En momento que se realizaba el patrullaje se recibe una llamada de parte del personal de base(ca...	SECTOR VI	12/02/2017
15	violencia familiar	En momento que se realizaba patrullaje se recibio una llamada de base castillo manifestando don...	SECTOR VI	14/02/2017
14	accidente de transito	Al momento de realizar mi patrullaje, se noto se informo de un accidente de transito en la calle señ...	SECTOR I	07/01/2017
13	Ocupacion de via sin autoriza...	Siendo las 22:01 del 07/01/2017, se levanta acta del señor Jose Luis Mendoza Olarte DNI: 447404...	SECTOR II	07/01/2017
12	niño extraviado	En momento que se servia el patrullaje por la zona III, el personal de base nos comunica de un sup...	SECTOR III	08/01/2017
11	supuesta agresion	En momento que se realizaba el patrullaje con la movilidad placa EQQ-931. el personal de base no...	SECTOR V	01/07/2017
10	violencia familiar	en momentos que se realizaba el patrullaje por la zona III se recepciono una llamada de parte del ...	SECTOR VI	11/01/2017
9	colision de vehiculos	En momentos que se realizaba el patrullaje con la movilidad EQQ-931, el personal base nos indica ...	SECTOR III	06/01/2017
8	violencia familiar	Mientras se patrullaba por la asociación Los Mecánicos, el personal de base nos informó sobre un...	SECTOR I	05/01/2017
7	Discusión de pareja	En momentos de patrullaje, se recepciona la llamada telefónica de la doctora Sonia con número d...	SECTOR II	05/01/2017
6	Supuesta violencia familiar	Mientras se realizaba el patrullaje, se recibió una llamada de una vecina quien manifiesta haber un...	SECTOR II	05/01/2017
5	Desaparición de fémina	Mientras los serenos del Patrullaje integrado desarrollaban su respectivo patrullaje a bordo del ve...	SECTOR II	01/02/2017
4	violencia familiar	el personal sereno en instante de realizar el respectivo patrullaje, recepciona comunicado de la ce...	SECTOR II	02/01/2017
3	Mordedura de can	en momentos de patrullaje por la av los incas a bordo del conejo N° 5 se recepciona una llamada ...	SECTOR III	02/01/2017
2	violencia familiar en año nuevo	en momentos de servicio de patrullaje el personal base nos comunica sobre un supuesto violencia...	SECTOR VI	01/01/2017
1	apoyo prestado en traslado a ...	en circunstancias del patrullaje conjuntamente con mi colega CORDERO PARIONA RICHAR, el per...	SECTOR III	01/01/2017

Anterior Page 19 of 19 Siguiente

© 2019 Junior Dev, Ingeniería de Sistemas UNSCH, por una mejor sociedad

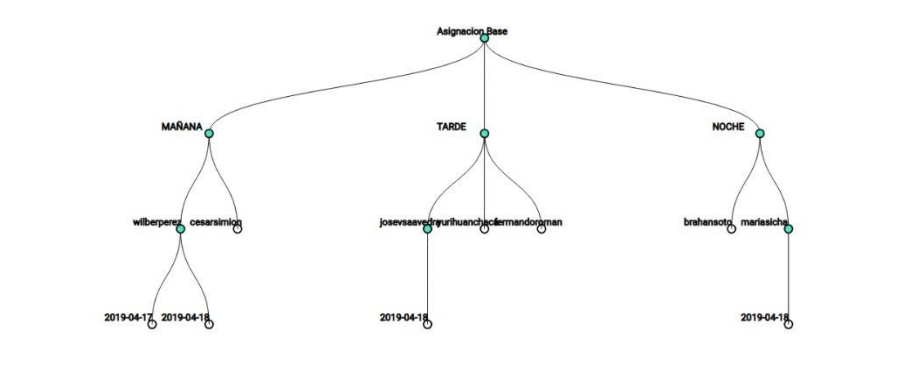
Elaboración propia.

Figura N° 4.28
Interfaz de usuario (frontend): Registrar patrullaje.



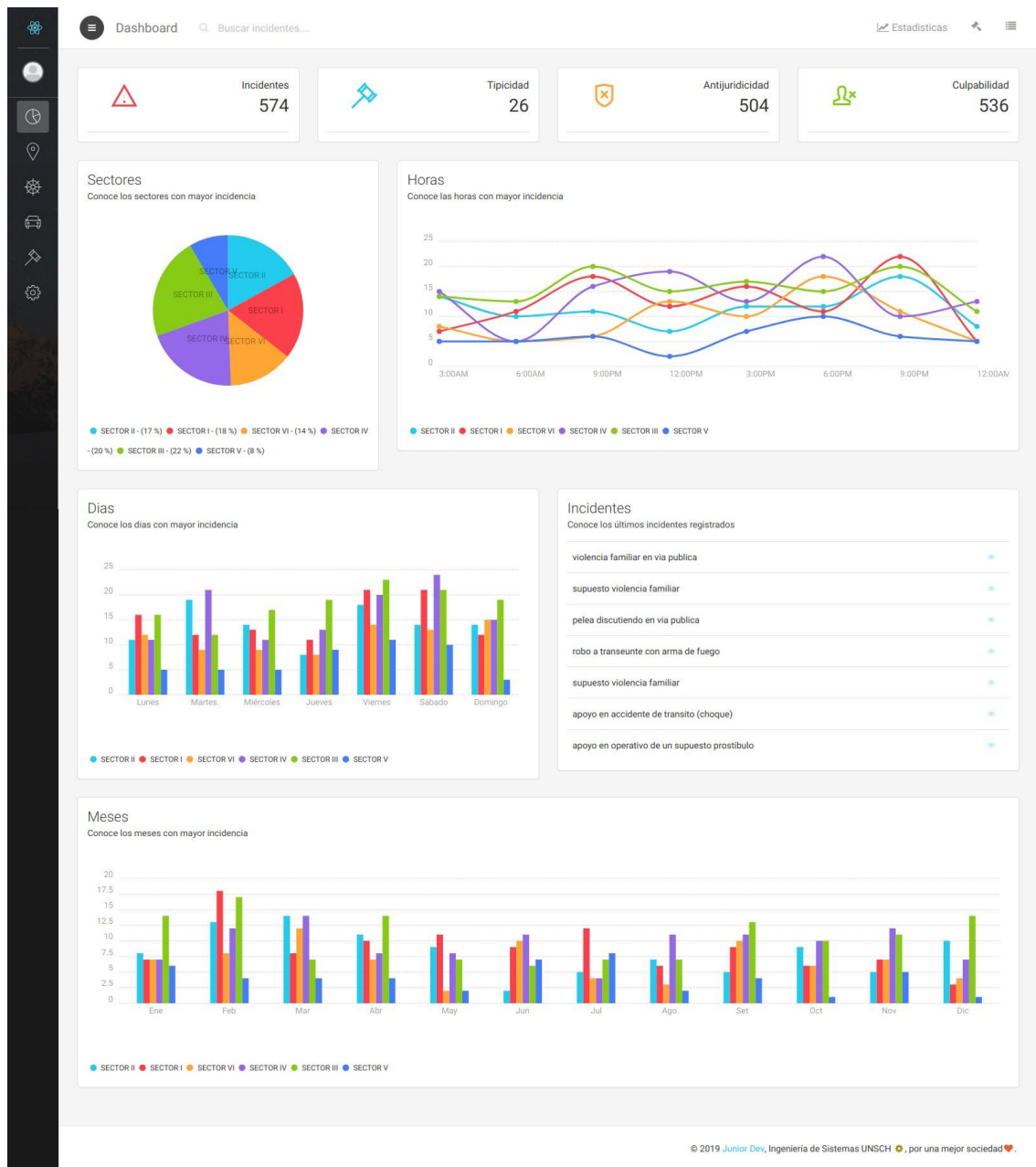
Elaboración propia.

Figura N° 4.29
Árbol base patrullaje.



Elaboración propia.

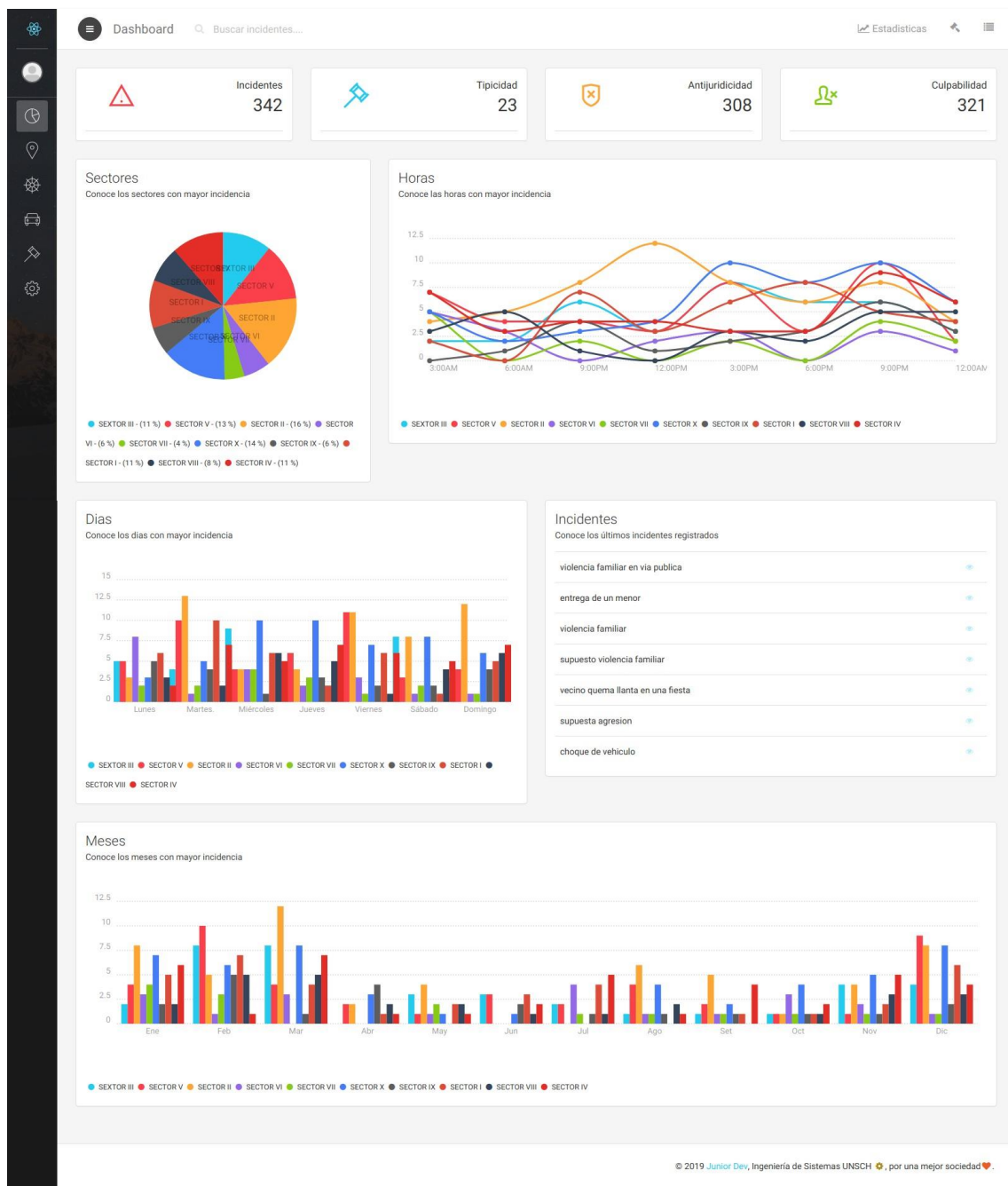
Figura N° 4.30
Interfaz de usuario (frontend): Dashboard – Jesús Nazareno.



Elaboración propia.

La figura 4.30 “Dashboard – Jesús Nazareno”, muestra de forma visual el comportamiento de los incidentes delictivos del distrito de Jesús Nazareno, en la parte superior se visualiza la sumatoria de incidentes, tipos de delito, incidentes antijurídicos e incidentes con culpabilidad, seguido, se tiene la frecuencia de ocurrencia organizado por sectores y horas del día (separadas en intervalos de 3 horas), como complemento se tienen los cuadros de días y meses que reflejan la frecuencia de ocurrencia de los incidentes delictivos del distrito.

Figura N° 4.31
Interfaz de usuario (frontend): Dashboard – Carmen Alto.

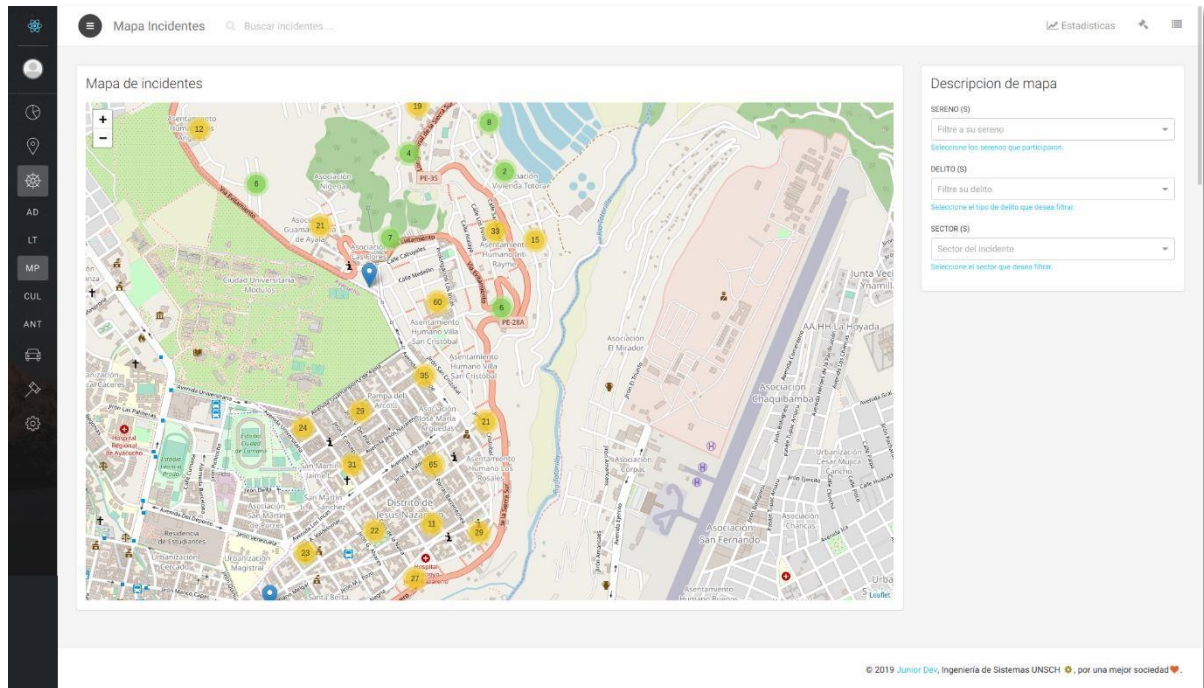


Elaboración propia.

La figura 4.31 “Dashboard – Carmen Alto”, tiene el mismo comportamiento que la figura 4.30 “Dashboard – Jesus Nazareno”, la diferencia radica en el origen de los datos. Se puede notar que, en ambas figuras, los meses de enero, febrero y marzo tienen mayor ocurrencia de incidentes, también, los incidentes mayormente ocurren en el intervalo de 6:00 pm a 9:00 pm.

Figura N° 4.32

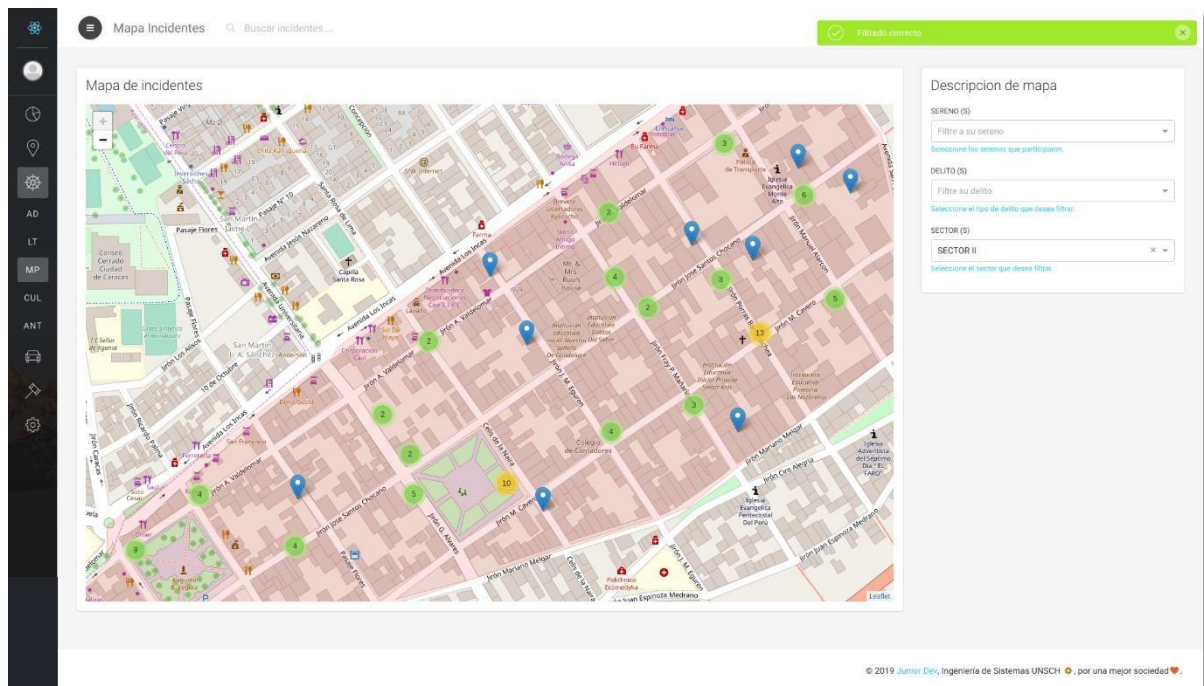
Interfaz de usuario (frontend): Mapa de incidente – Jesús Nazareno.



Elaboración propia.

Figura N° 4.33

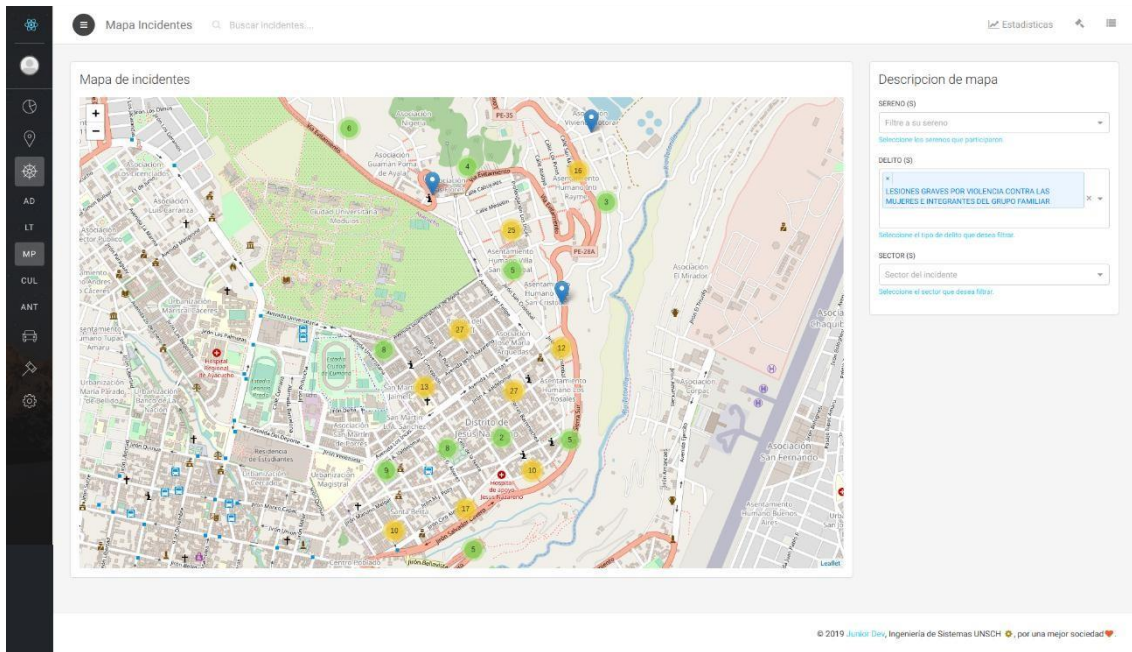
Interfaz de usuario (frontend): Mapa de incidente (filtrado - sector) – Jesús Nazareno.



Elaboración propia.

Figura N° 4.34

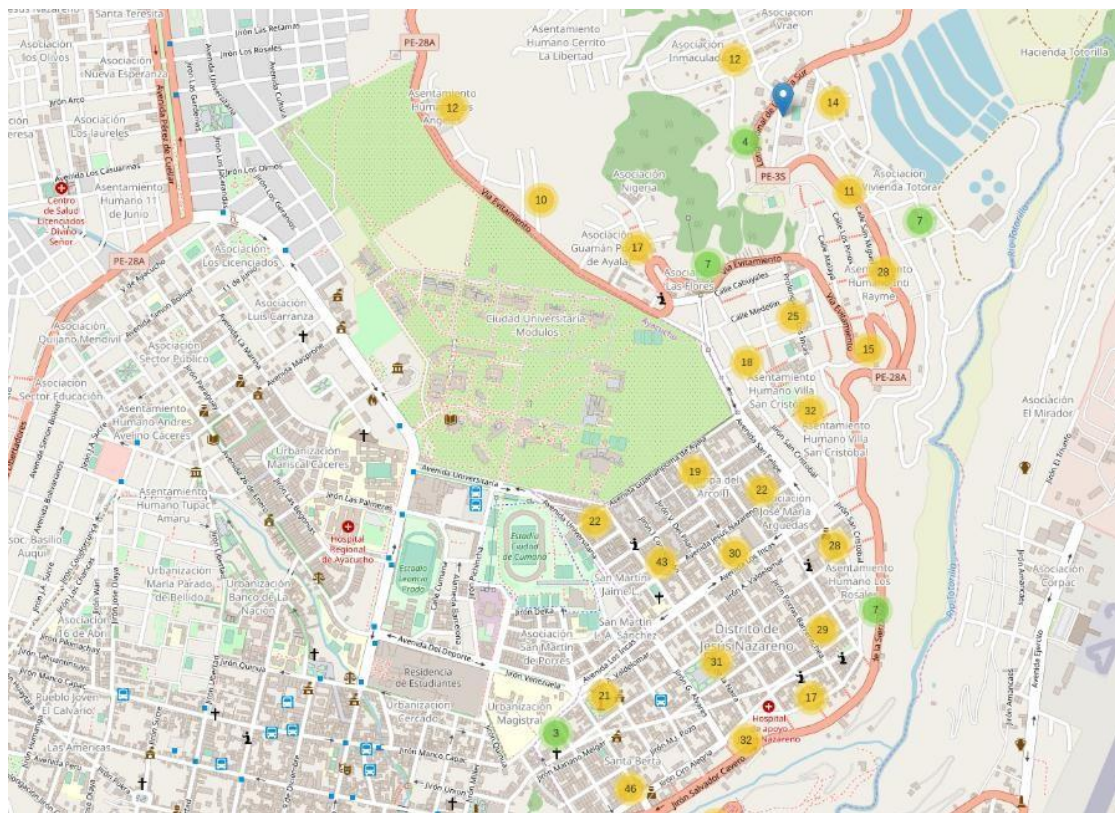
Interfaz de usuario (frontend): Mapa de incidente (filtrado – tipo delito)- Jesús Nazareno.



Elaboración propia.

Figura N° 4.35

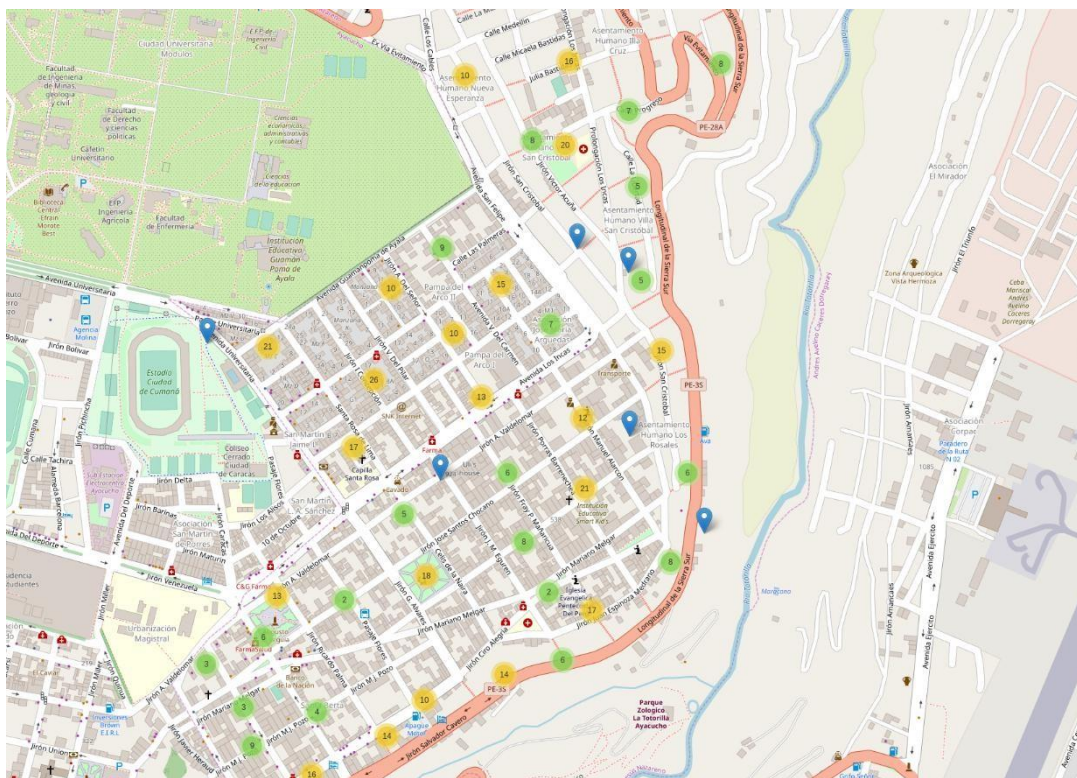
Mapa del delito del distrito de Jesús Nazareno.



Elaboración propia.

Figura N° 4.36

Mapa del delito con zoom del distrito de Jesús Nazareno.



Elaboración propia.

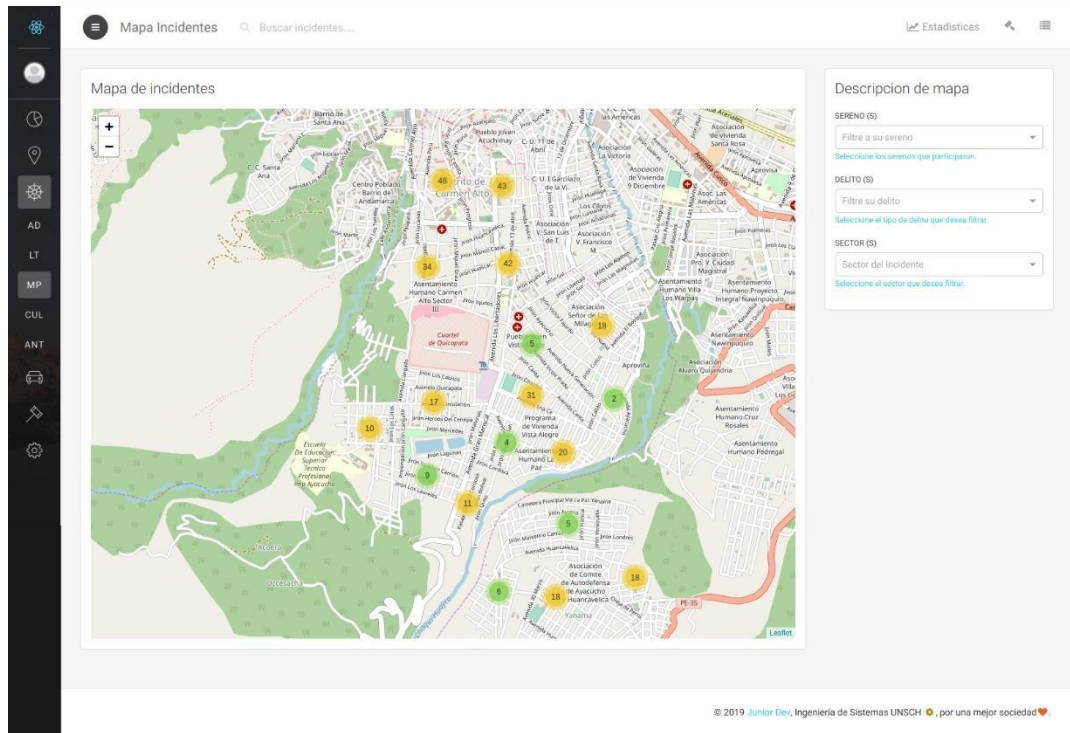
El mapa del delito del distrito de Jesús Nazareno se muestra en las figuras 4.35 “Mapa del delito del distrito de Jesús Nazareno” y 4.36 “Mapa del delito con zoom del distrito de Jesús Nazareno”, la primera figura muestra la clusterización de los incidentes delictivos dentro de los diferentes sectores, los círculos de color amarillo y verde representan los clústeres con mayor y menor incidencia delictiva, respectivamente, en la figura 4.36, se hace un zoom y los clústeres se expanden para mostrar los marcadores o nuevos clústeres.

Según los serenos, los incidentes delictivos se concentran en los sectores I, II y III por la concentración comercial del distrito y por la feria que se realizan los lunes.

El mapa del delito no analiza la cifra negra, cifra que representa los incidentes delictivos que no fueron registrados en el cuaderno de incidentes, tampoco, se analizó las zonas de riesgo, por otro lado, el mapa del delito permite identificar los puntos críticos del distrito.

Figura N° 4.37

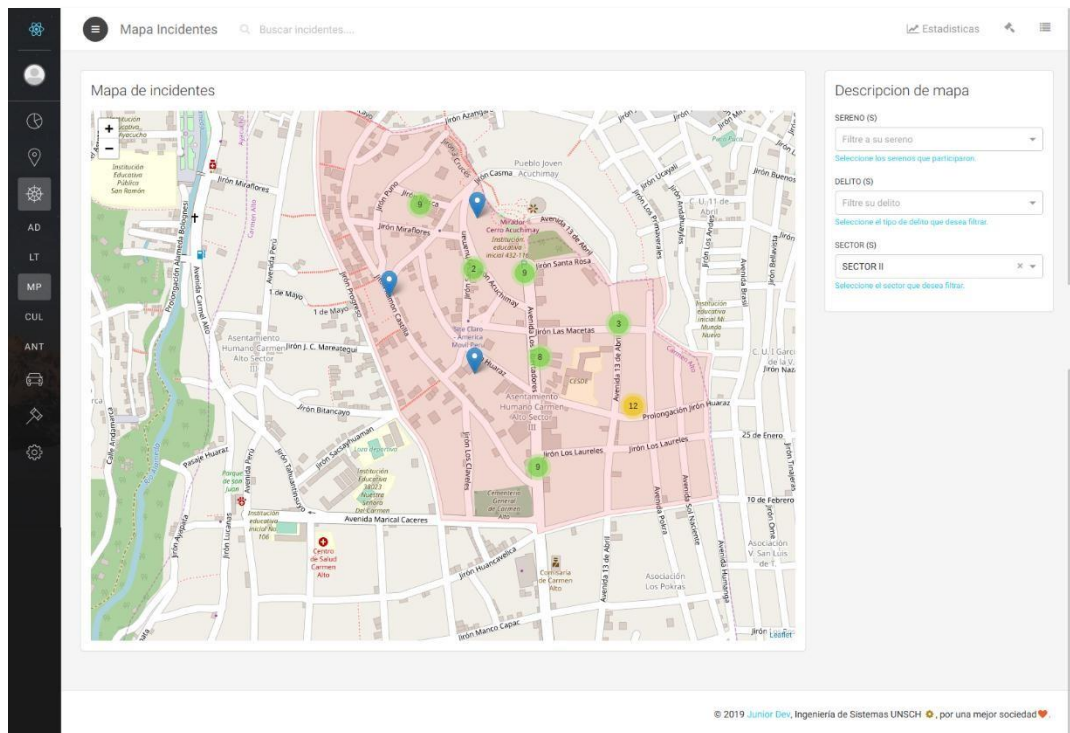
Interfaz de usuario (frontend): Mapa de incidente – Carmen Alto.



Elaboración propia.

Figura N° 4.38

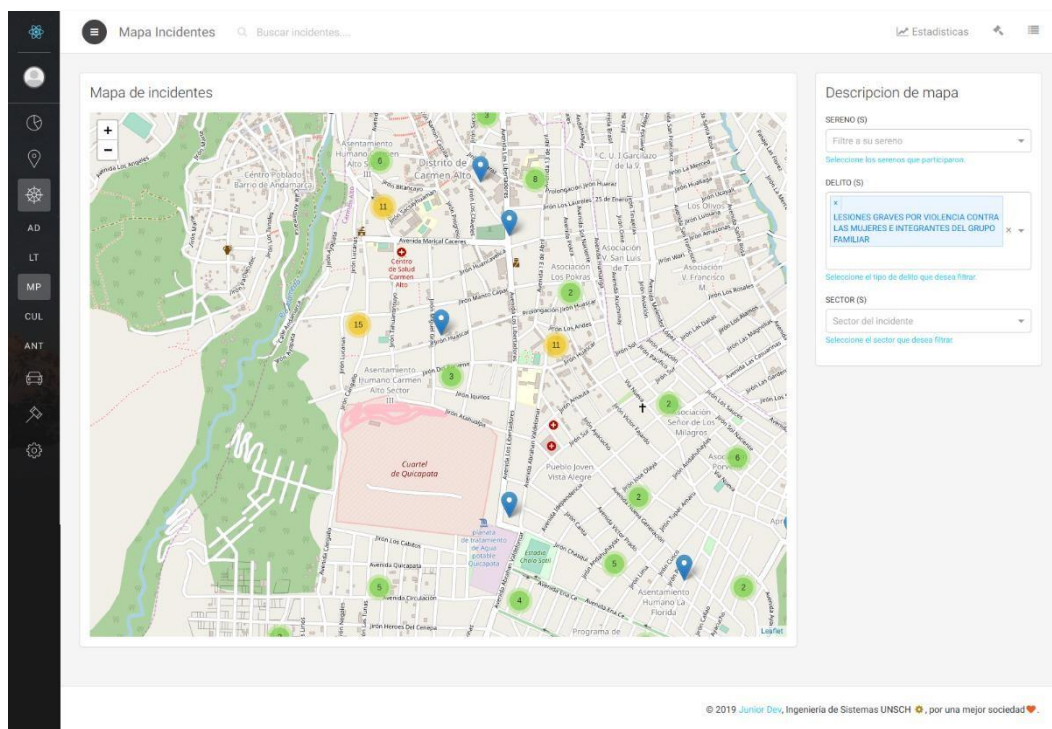
Interfaz de usuario (frontend): Mapa de incidente (filtrado- sector) – Carmen Alto.



Elaboración propia.

Figura N° 4.39

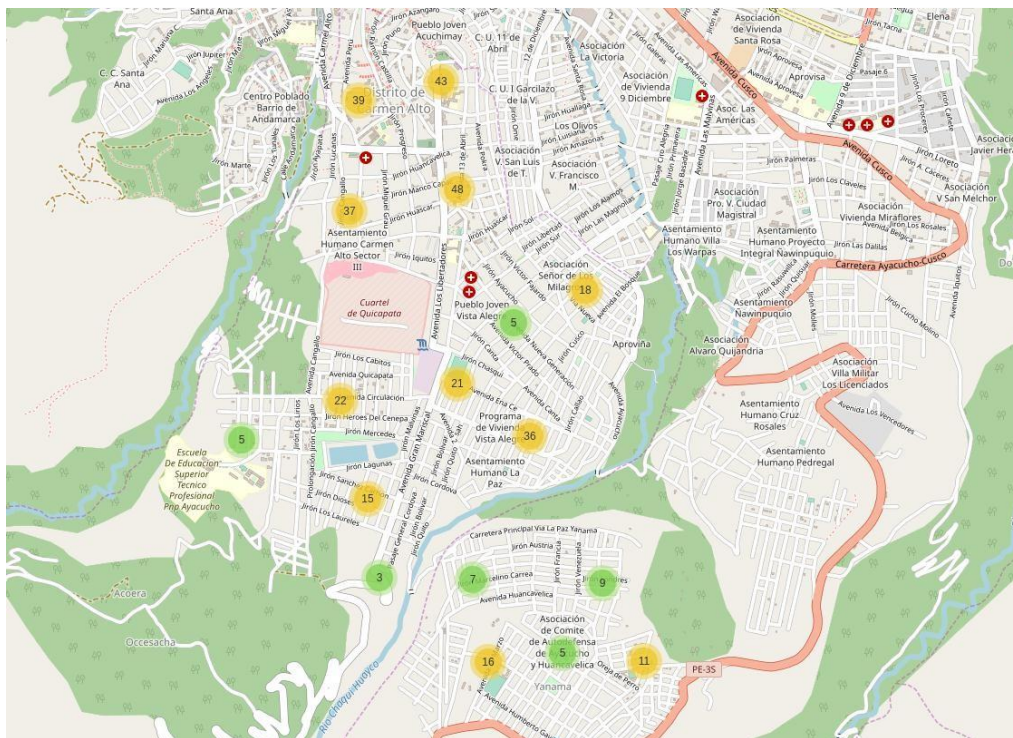
Interfaz de usuario (frontend): Mapa de incidente (filtrado- tipo delito) – Carmen Alto.



Elaboración propia.

Figura N° 4.40

Mapa del delito del distrito de Carmen Alto.



Elaboración propia.

Figura N° 4.41

Mapa del delito con zoom del distrito de Carmen Alto



Elaboración propia.

El mapa del delito del distrito de Carmen Alto se muestra en las figuras 4.40 “Mapa del delito del distrito de Carmen Alto” y 4.41 “Mapa del delito con zoom del distrito de Carmen Alto”, la primera figura muestra la clusterización de los incidentes delictivos dentro de los diferentes sectores, los círculos de color amarillo y verde representan los clústeres con mayor y menor incidencia delictiva, respectivamente, en la figura 4.41, se hace un zoom y los clústeres se expanden para mostrar los marcadores o nuevos clústeres.

Según los serenos, los incidentes delictivos se concentran en los sectores I, II, III, IV por la concentración comercial del distrito y por qué en el sector II se encuentran las discotecas y recreos.

El mapa del delito no analiza la cifra negra, cifra que representa los incidentes delictivos que no fueron registrados en el cuaderno de incidentes, tampoco, se analizó las zonas de riesgo, por otro lado, el mapa del delito permite identificar los puntos críticos del distrito.

Figura N° 4.42

Interfaz de usuario (frontend): Nube de delitos – Jesús Nazareno.



Elaboración propia.

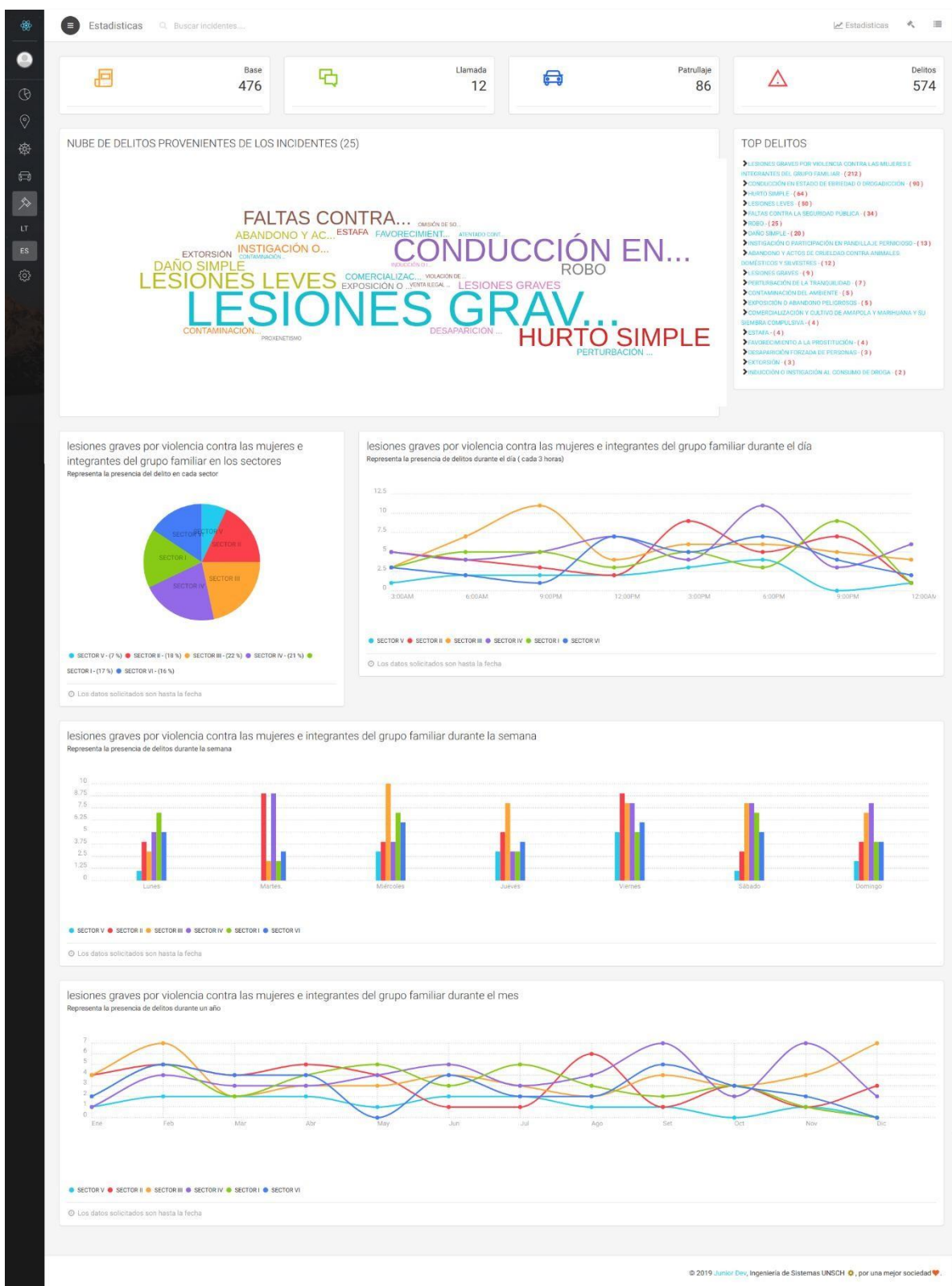
En la imagen 4.42, se muestra la interfaz inicial de la nube de delitos del distrito de Jesús Nazareno, en la parte superior se muestra la sumatoria de incidentes reportados en base, llamadas, patrullaje y finalmente la sumatoria de todos los incidentes delictivos; seguido, en la parte derecha se ve una lista con los tipos de delitos y la cantidad de apariciones (ordenados de forma descendente), en la parte izquierda se observa la nube de delitos; el color de los incidentes es aleatorio y el tamaño esta en proporción a la cantidad total de incidentes delictivos, la cabecera de la nube de delitos tiene el total de tipos de delito diferentes que se registraron en la sub gerencia.

Si se hace clic a cualquiera de los tipos de delito de la nube de delitos o la lista, el sistema muestra un detalle del tipo de delito representados en las figuras 4.43 y 4.44, el detalle del tipo de delito es una análisis que el sistema realiza sobre todos los incidentes registrados que presentan antijuridicidad y culpabilidad, el análisis se hace sobre la frecuencia de ocurrencia en los sectores, durante el día (las 24 horas del día se agruparon en bloques de 3 horas), durante los días de la semana y finalmente durante los meses del año.

La diferencia de las figuras 4.43 y 4.44 radica en el tipo de delito que se seleccionó, nótese, que en las imágenes la nube de delitos cambia de color, esto sucede porque el sistema consulta los datos al sistema backend y renderiza nuevamente el componente “mapa del delito”

Figura N° 4.43

Interfaz de usuario (frontend): Nube de delitos (lesiones graves por violencia contra las mujeres e integrantes del grupo familiar)– Jesús Nazareno.



Elaboración propia.

Interfaz de usuario (frontend): Nube de delitos (conducción en estado de ebriedad o drogadicción)– Jesús Nazareno.

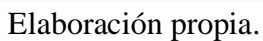


Figura N° 4.45

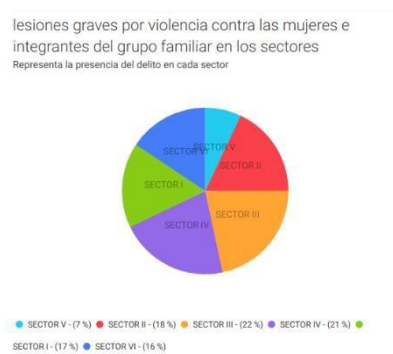
Nube de delitos del distrito de Jesús Nazareno.



Elaboración propia.

Cuadro N° 4.1

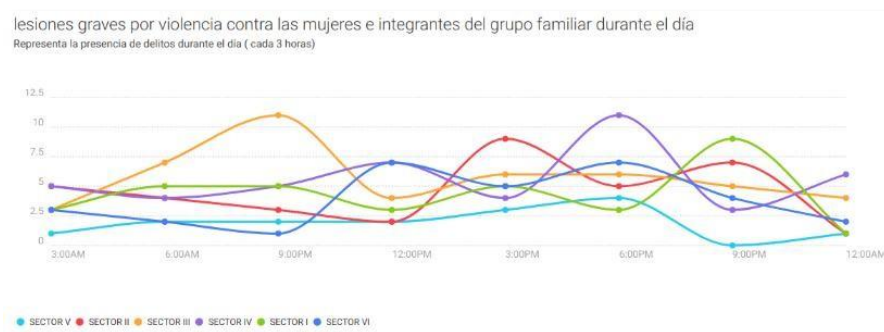
Frecuencia por sectores de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Jesús Nazareno.



Elaboración propia.

Cuadro N° 4.2

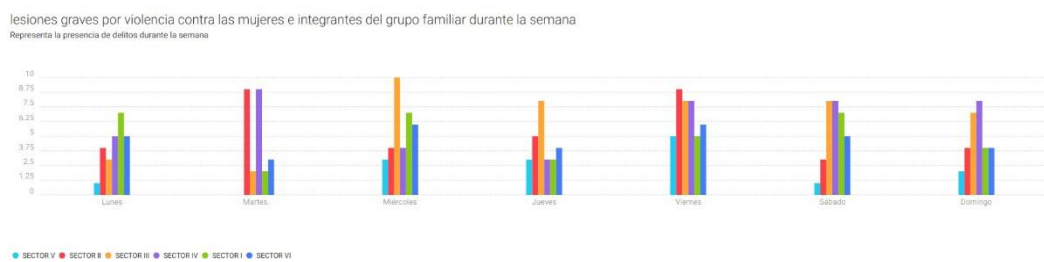
Frecuencia por horas del día de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Jesús Nazareno.



Elaboración propia.

Cuadro N° 4.3

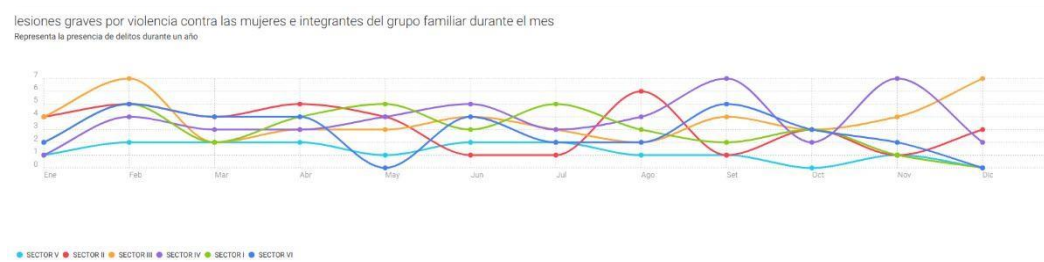
Frecuencia por días de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Jesús Nazareno.



Elaboración propia.

Cuadro N° 4.4

Frecuencia por meses de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Jesús Nazareno.



Elaboración propia.

El distrito de Jesús Nazareno tiene a las “*lesiones graves por violencia contra las mujeres e integrantes del grupo familiar*” como tipo de delito que ocurre con mayor frecuencia, con un total de 212 casos registrados que son antijurídicos y culpables, seguido, se tiene a la “*conducción en estado de ebriedad o drogadicción*” con 90 casos registrados. En los siguientes párrafos se realiza un análisis del tipo de delito con mayor ocurrencia en el distrito.

En el cuadro imagen 4.1, se observa que el sector III tiene el 22%, el sector IV el 21% y el sector II el 18% de incidentes delictivos. Según los serenos, la razón por la que se centran los incidentes en estos sectores, radica en que son los sectores con mayor densidad poblacional y comercial.

En el cuadro 4.2, se observa que entre las 6:00 am y 9:00 am, el sector III tiene mayor concentración de incidentes delictivos, pero, todos los sectores tienden a aumentar entre las

3:00 y 9:00 pm.

De los cuadros 4.3 y 4.4, se observa que los días de mayor incidencia delictiva son los viernes, sábados y domingos, en la mayoría de sectores; del mismo modo, febrero, setiembre, noviembre y diciembre son los meses que registran mayor cantidad de incidentes delictivos.

El análisis se realizó con el tipo de delito “*lesiones graves por violencia contra las mujeres e integrantes del grupo familiar*”, el sistema puede realizar análisis con cada tipo de delito que haya sido registrado en la sub gerencia.

Figura N° 4.46

Interfaz de usuario (frontend): Nube de delitos – Carmen Alto.



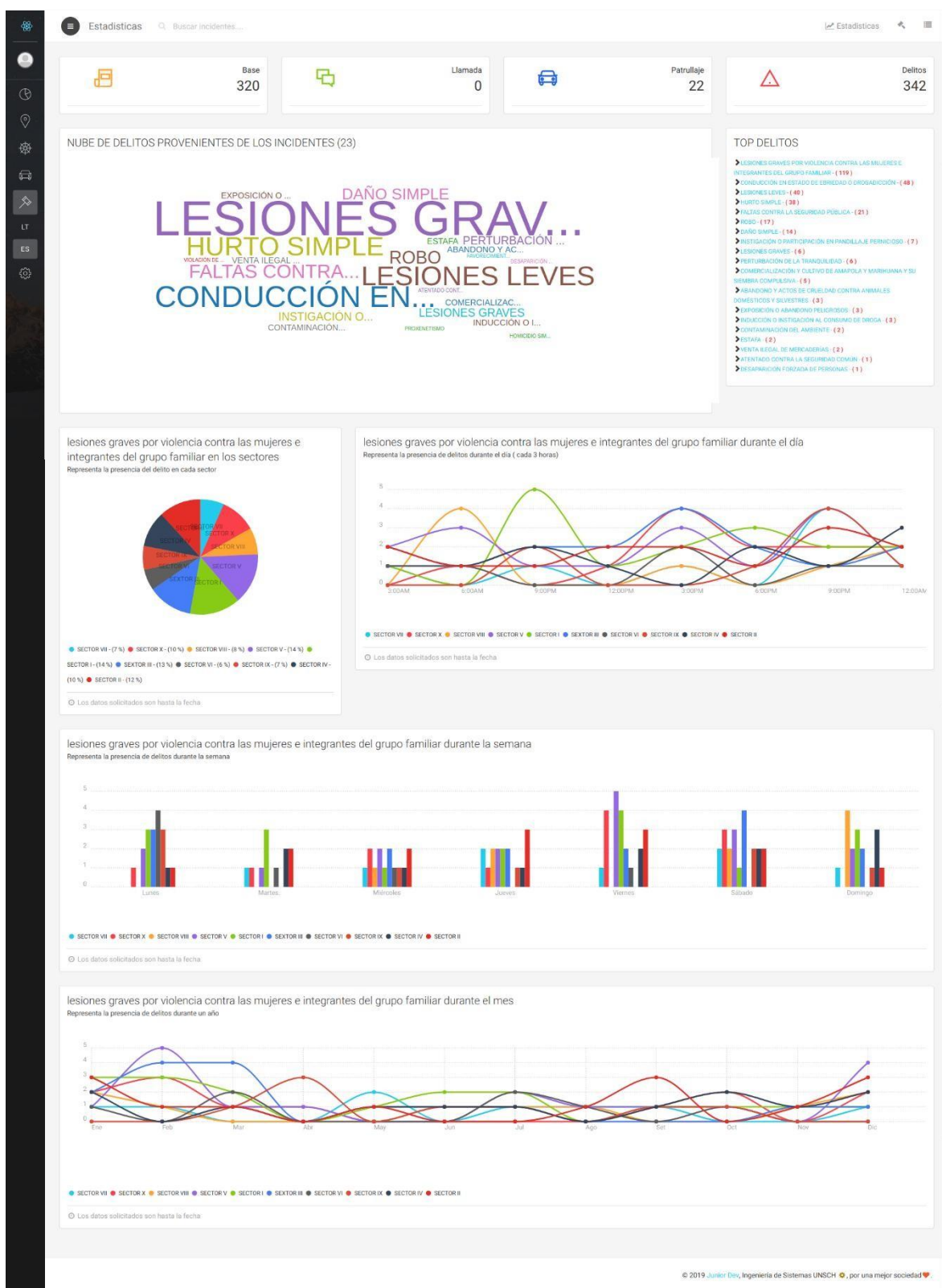
Elaboración propia.

En la imagen 4.46, se muestra la interfaz inicial de la nube de delitos del distrito de Carmen Alto, la estructura de los datos e interacción con el mapa del delito son similares a la figura 4.42 “Interfaz de usuario (frontend): Nube de delitos – Jesús Nazareno”, la diferencia es el origen de los datos.

La diferencia de las figuras 4. 47 y 4.48 radica en el tipo de delito que se seleccionó en la nube de delitos, nótese, que en las imágenes la nube de delitos cambia de color, esto sucede porque el sistema consulta los datos del sistema backend y renderiza nuevamente el componente “nube de delitos”.

Figura N° 4.47

Interfaz de usuario (frontend): Nube de delitos (lesiones graves por violencia contra las mujeres e integrantes del grupo familiar)– Carmen Alto.



Elaboración propia.

Figura N° 4.48

Interfaz de usuario (frontend): Nube de delitos (hurto simple) – Carmen Alto.



Elaboración propia.

Figura N° 4.49

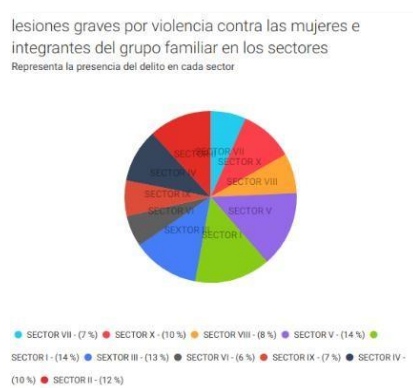
Nube de delitos del distrito de Carmen Alto.



Elaboración propia.

Cuadro N° 4.5

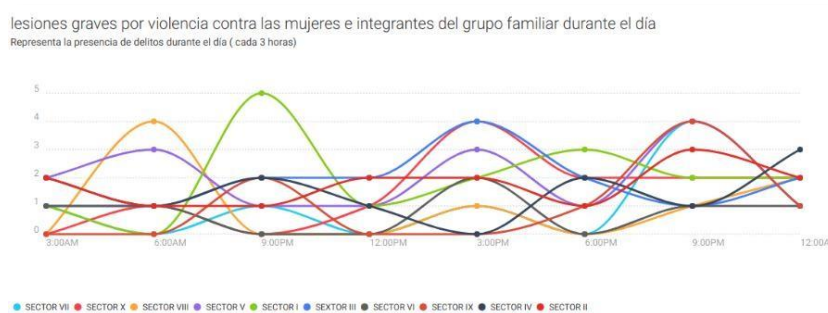
Frecuencia por sectores de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Carmen Alto.



Elaboración propia.

Cuadro N° 4.6

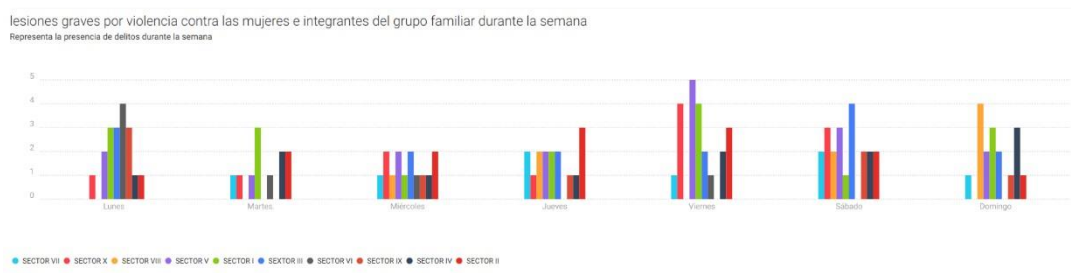
Frecuencia por horas del día de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Carmen Alto.



Elaboración propia.

Cuadro N° 4.7

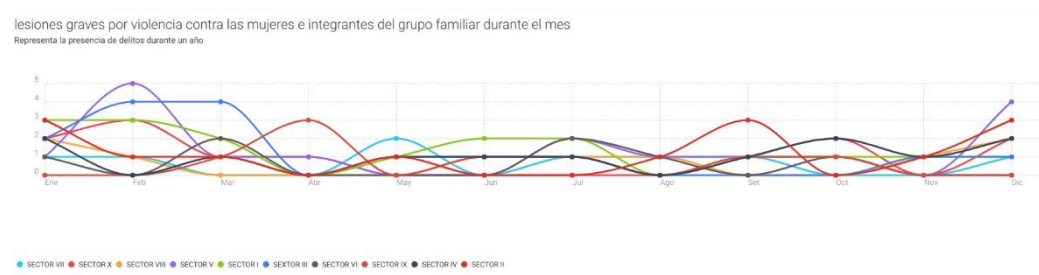
Frecuencia por días de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Carmen Alto.



Elaboración propia.

Cuadro N° 4.8

Frecuencia por meses de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Carmen Alto.



Elaboración propia.

El distrito de Carmen Alto tiene a las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar como tipo de delito que ocurre con mayor frecuencia, con un total de 119 casos registrados que son antijurídicos y culpables.

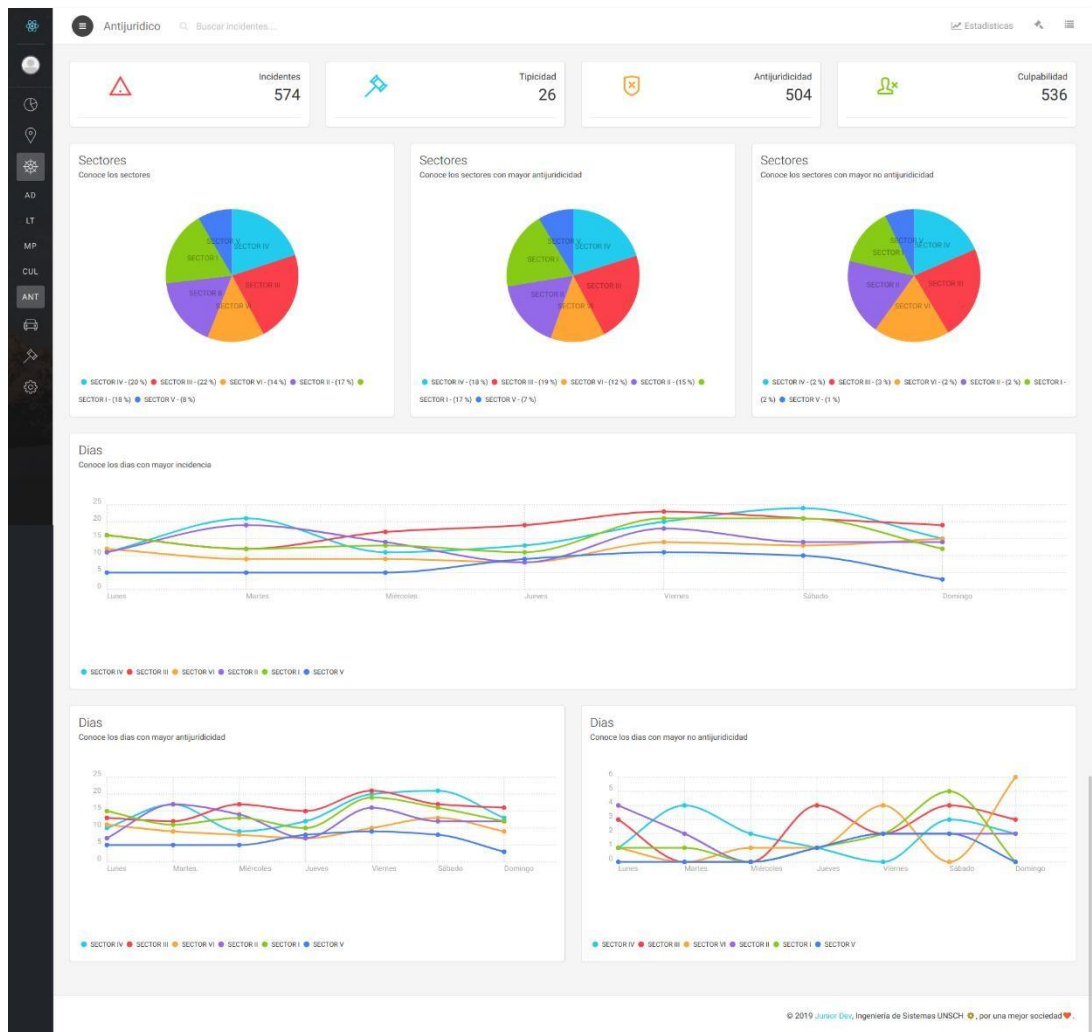
En el cuadro 4.5, se puede observar cuales son los sectores que concentran la mayor cantidad de incidentes delictivos. Según los serenos, la razón por la que se centran los incidentes en estos sectores, radica en que son los sectores con mayor densidad poblacional y comercial.

En el cuadro 4.6, se puede observar en que intervalo del día ocurre más incidentes delictivos, del mismo modo, conocer cuál es la endentecía entre los diez sectores del distrito.

De los cuadros 4.7 y 4.8, se puede conocer que días y meses concentran la mayor cantidad de incidentes delictivos.

Figura N° 4.50

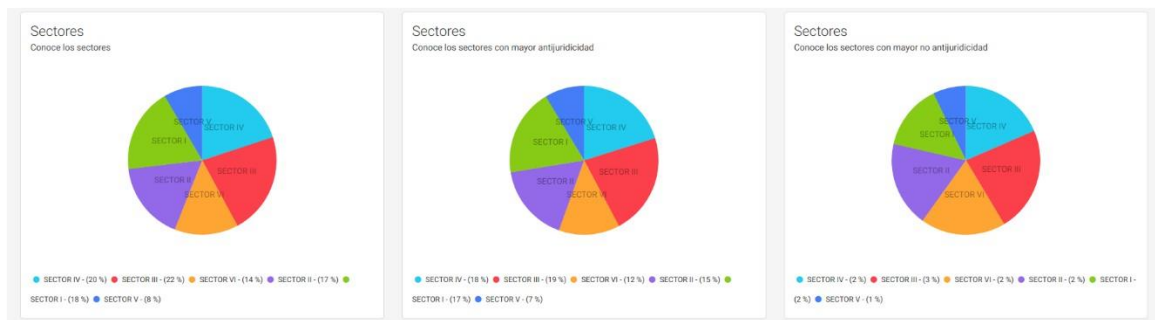
Interfaz de usuario (frontend): Dashboard Antijuridicidad – Jesús nazareno



Elaboración propia.

Figura N° 4.51

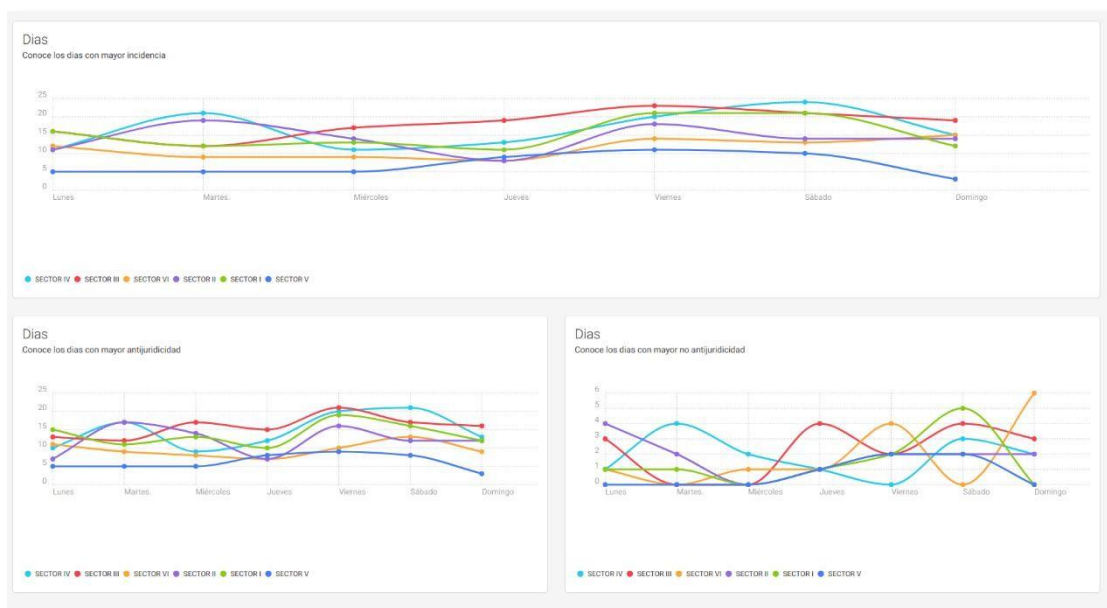
Frecuencia por sectores de la antijuridicidad del distrito de Jesús nazareno.



Elaboración propia.

Figura N° 4.52

Frecuencia por días de la antijuridicidad del distrito de Jesús nazareno.



Elaboración propia.

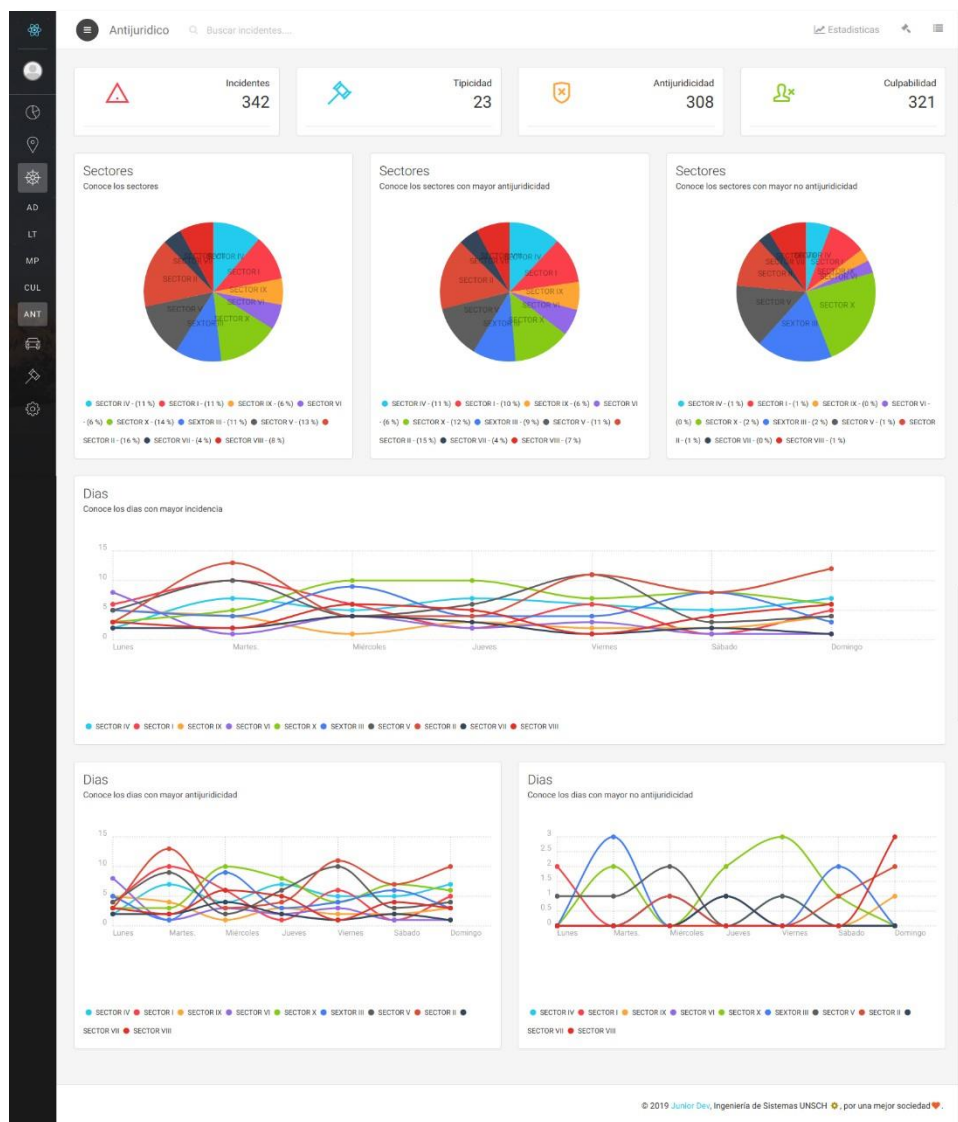
En el distrito de Jesús Nazareno, el 87% de incidentes delictivos son antijurídicos, los cuales se encuentran clasificados en los diferentes sectores del distrito (figura 4.51); de izquierda a derecha, la primera representación muestra cual es la frecuencia de los incidentes delictivos (antijuridicidad indiferente), la representación del medio muestra la frecuencia de ocurrencia de los incidentes antijurídicos, finalmente, la última representación muestra la frecuencia de ocurrencia de los incidentes jurídicos.

En el grafico 4.52, se muestra la frecuencia de los incidentes antijurídicos durante los días de la semana, las tres representaciones siguen el patrón delos sectores (figura 4.51).

Para que un incidente delictivo sea considerado jurídico o antijurídico, se evalúa la responsabilidad penal del incidente y si este cumple el requisito base del tipo de delito (tipicidad). El artículo 20 del código penal menciona las causas que eximen o atenúan la responsabilidad, como ejemplo que apareció en los registros del distrito se tiene un caso “*El que en defensa de bienes jurídicos propios o de terceros*” en un caso en donde un hombre maltrato a un perro porque el perro lo atacó.

Figura N° 4.53

Interfaz de usuario (frontend): Dashboard Antijuridicidad – Carmen Alto



Elaboración propia.

Figura N° 4.54

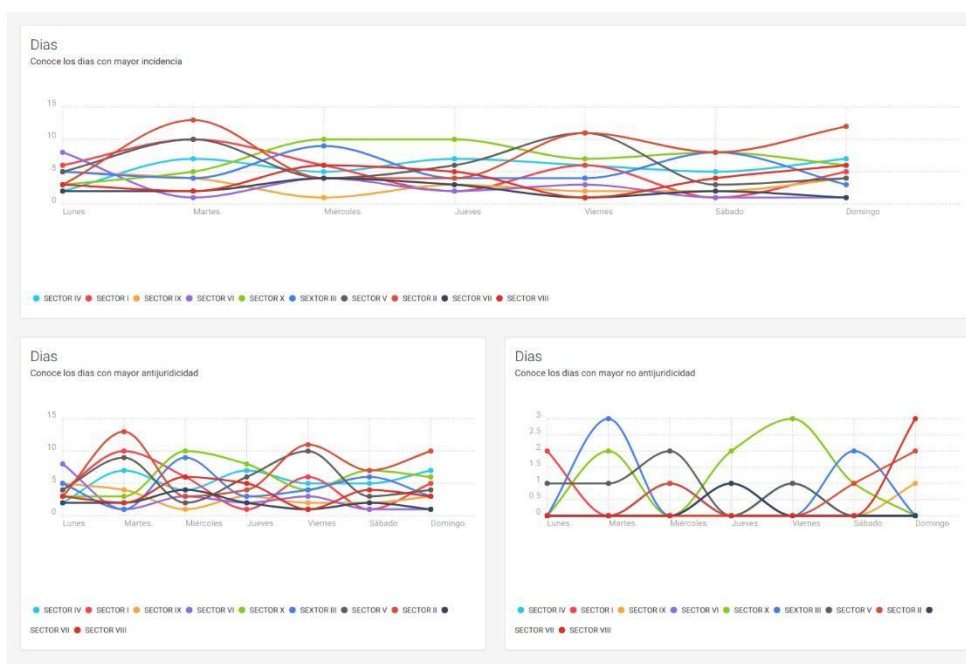
Frecuencia por sectores de la antijuridicidad del distrito de Carmen Alto



Elaboración propia.

Figura N° 4.55

Frecuencia por días de la antijuridicidad del distrito de Carmen Alto



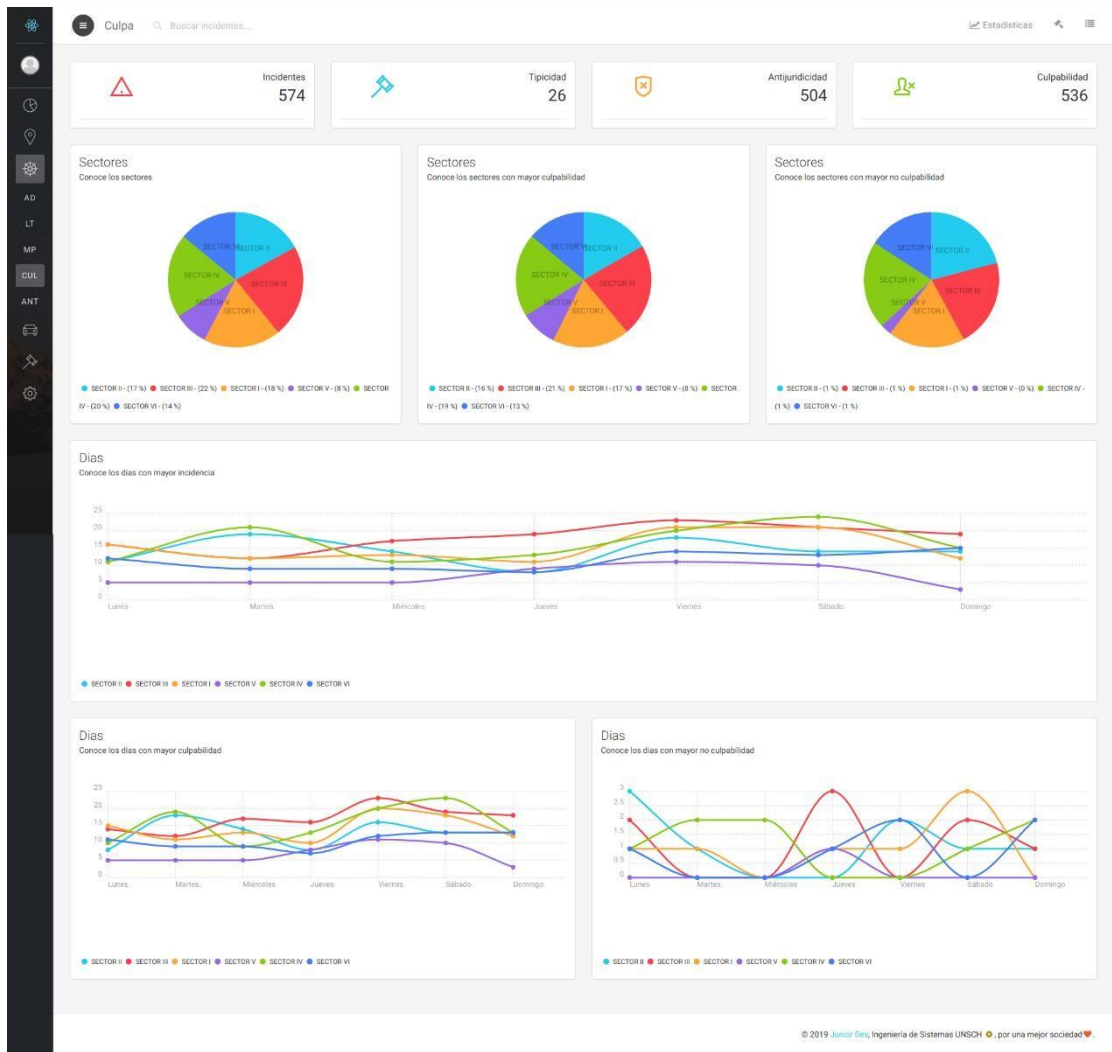
Elaboración propia.

En el distrito de Carmen Alto, el 90% de incidentes delictivos son antijurídicos, los cuales se encuentran clasificados en los diferentes sectores del distrito (figura 4.54); de izquierda a derecha, la primera representación muestra cual es la frecuencia de los incidentes delictivos (antijuridicidad indiferente), la representación del medio muestra la frecuencia de ocurrencia de los incidentes antijurídicos, finalmente, la última representación muestra la frecuencia de ocurrencia de los incidentes jurídicos.

Para que un incidente delictivo sea considerado jurídico o antijurídico, se evalúa la responsabilidad penal del incidente y si este cumple el requisito base del tipo de delito (tipicidad). El artículo 20 del código penal menciona las causas que eximen o atenúan la responsabilidad, como ejemplo que apareció en los registros del distrito se tiene un caso “*El que en defensa de bienes jurídicos propios o de terceros*” en un caso en donde un hombre que causo lesiones leves a un joven, el motivo registrado fue “*El joven lesionado intento robar el celular en la ruta*”.

Figura N° 4.56

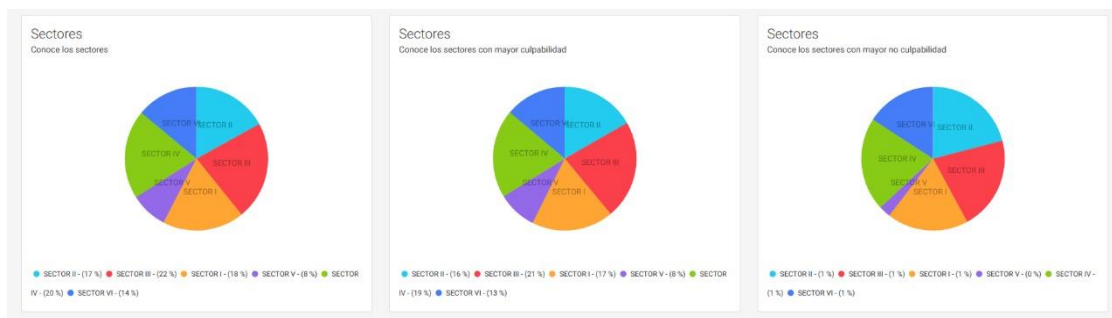
Interfaz de usuario (frontend): Dashboard Culpabilidad – Jesús Nazareno



Elaboración propia.

Figura N° 4.57

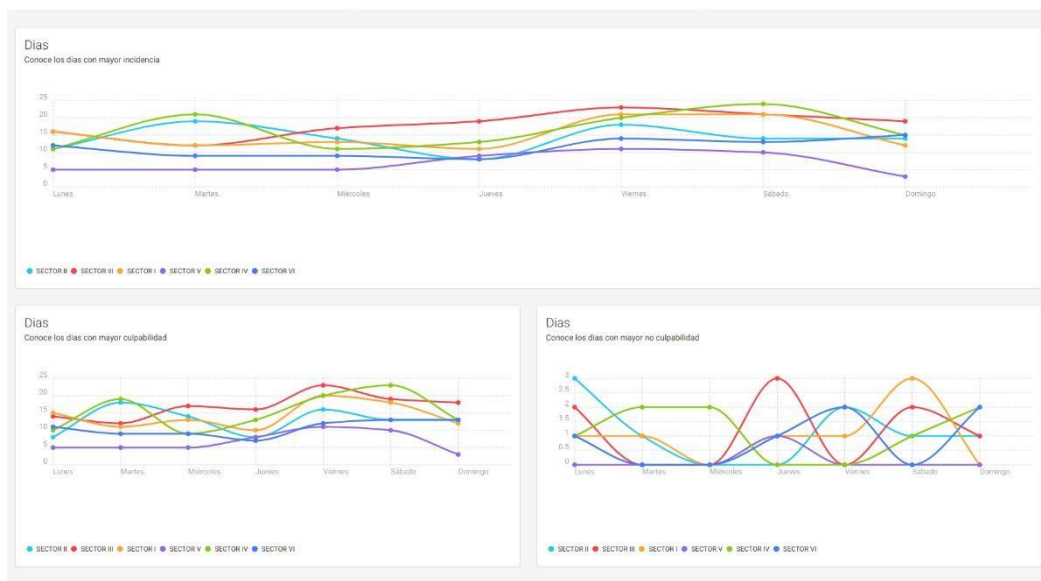
Frecuencia por sectores de la culpabilidad del distrito de Jesús Nazareno.



Elaboración propia.

Figura N° 4.58

Frecuencia por días de la culpabilidad del distrito de Jesús Nazareno.



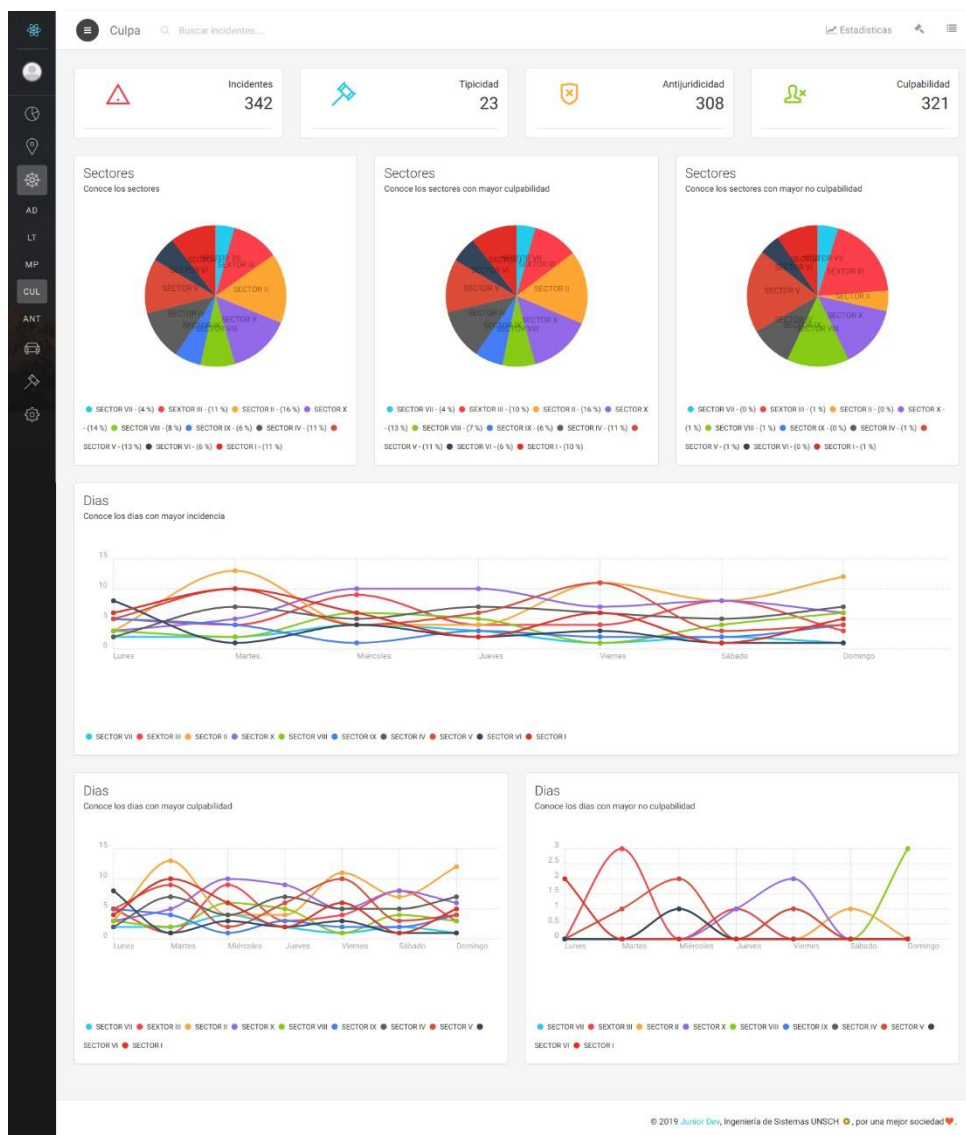
Elaboración propia.

En el distrito de Jesús Nazareno, el 93% de incidentes delictivos tienen culpabilidad, los cuales se encuentran clasificados en los diferentes sectores del distrito (figura 4.57); de izquierda a derecha, la primera representación muestra cual es la frecuencia de los incidentes delictivos (culpabilidad indiferente), la representación del medio muestra la frecuencia de ocurrencia de los incidentes con culpabilidad, finalmente, la última representación muestra la frecuencia de ocurrencia de los incidentes con ex culpabilidad.

Para que un incidente delictivo tenga culpabilidad, se evaluó la responsabilidad penal del incidente y si este cumple el requisito base del tipo de delito (tipicidad) y la antijuridicidad. El artículo 20 del código penal menciona las causas que eximen o atenúan la responsabilidad, como ejemplo que apareció en los registros del distrito se tiene el caso *“El que por anomalía psíquica, grave alteración de la conciencia o por sufrir alteraciones en la percepción, que afectan gravemente su concepto de la realidad, no posea la facultad de comprender el carácter delictuoso de su acto o para determinarse según esta comprensión”* en un caso en donde una mujer golpea a un transeúnte provocando lesiones leves en la vía pública, la mujer no es culpable de sus actos, ya que ella padece de una enfermedad mental que la exculpa de sus actos.

Figura N° 4.59

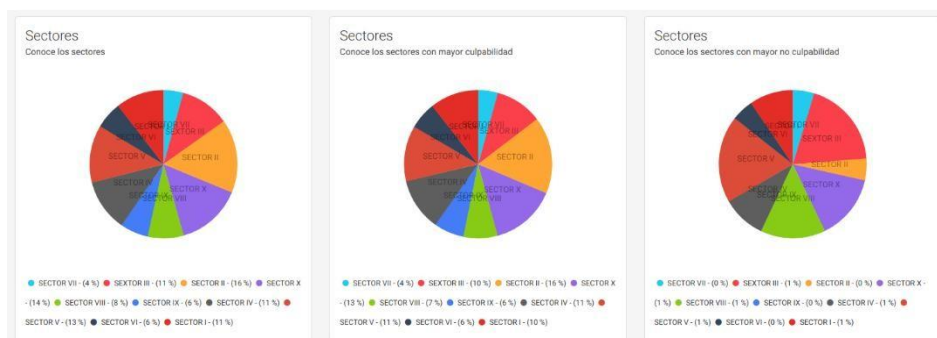
Interfaz de usuario (frontend): Dashboard Culpabilidad – Carmen Alto



Elaboración propia.

Figura N° 4.60

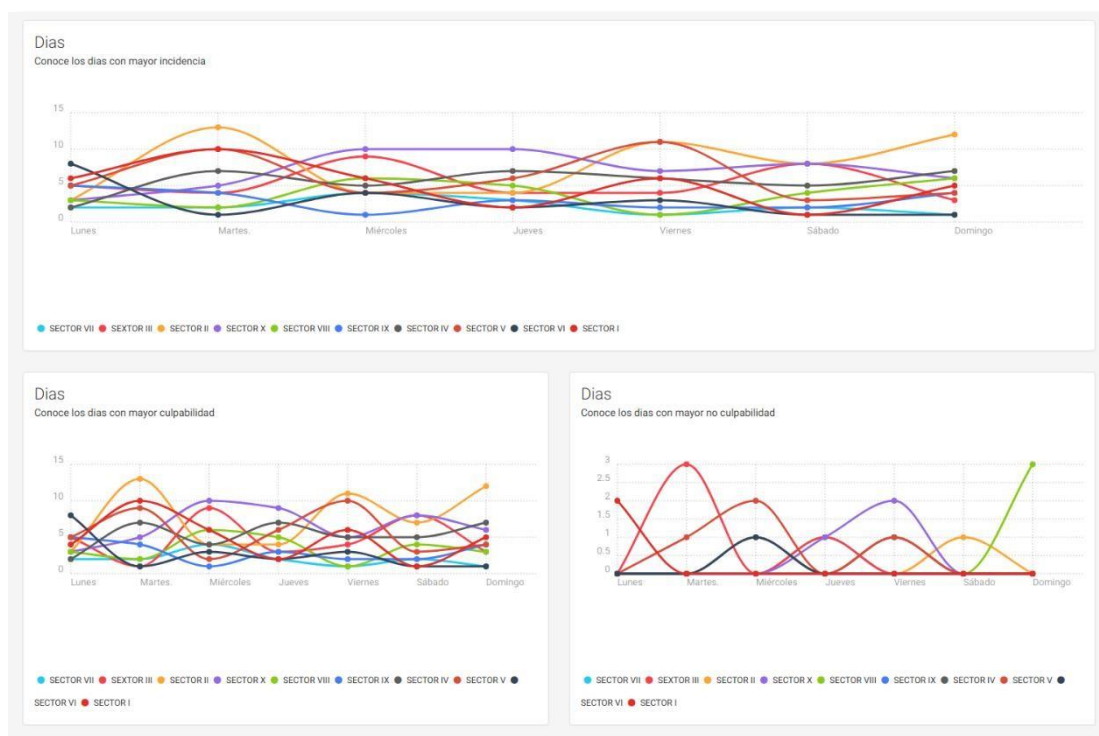
Frecuencia por sectores de la culpabilidad del distrito de Carmen Alto



Elaboración propia.

Figura N° 4.61

Frecuencia por días de la culpabilidad del distrito de Carmen Alto



Elaboración propia.

En el distrito de Carmen Alto, el 94% de incidentes delictivos tienen culpabilidad, los cuales se encuentran clasificados en los diferentes sectores del distrito (figura 4.60); de izquierda a derecha, la primera representación muestra cual es la frecuencia de los incidentes delictivos (culpabilidad indiferente), la representación del medio muestra la frecuencia de ocurrencia de los incidentes con culpabilidad, finalmente, la última representación muestra la frecuencia de ocurrencia de los incidentes con ex culpabilidad.

Para que un incidente delictivo tenga culpabilidad o no, se evaluó la responsabilidad penal del incidente y si este cumple el requisito base del tipo de delito (tipicidad) y la antijuridicidad. El artículo 20 del código penal menciona las causas que eximen o atenúan la responsabilidad, como ejemplo que apareció en los registros del distrito se tiene el caso *“El que por anomalía psíquica, grave alteración de la conciencia o por sufrir alteraciones en la percepción, que afectan gravemente su concepto de la realidad, no posea la facultad de comprender el carácter delictuoso de su acto o para determinarse según esta comprensión”* en un caso en donde una mujer golpea a un cliente en una tienda, la mujer no es culpable de sus actos, ya que ella padece de una enfermedad mental que la exculpa.

E. TARJETAS CRC (CLASE – RESPONSABILIDAD - COLABORADOR)

Tabla N° 4.82
Tarjeta CRC USUARIO_USER

CLASE: USUARIO_USER	
RESPONSABILIDAD Registrar datos del usuario Extraer datos del sereno Extraer datos de la sgsc Extraer último cargo Autenticar usuarios	COLABORADOR SGSC_SERENO SGSC_SGSC

Elaboración propia

Tabla N° 4.83
Tarjeta CRC AUTH_PERMISSION

CLASE: AUTH_PERMISSION	
RESPONSABILIDAD Registrar permisos del usuario Registrar grupos de permisos a los usuarios	COLABORADOR SGSC_USUARIO

Elaboración propia

Tabla N° 4.84
Tarjeta CRC TOKEN_BLACKLIST_OUTSTANDINGTOKEN

CLASE: TOKEN_BLACKLIST_OUTSTANDINGTOKEN	
RESPONSABILIDAD Registrar token del usuario Generar token del usuario Decodificar token del usuario	COLABORADOR TOKEN_BLACKLIST_BLACKLISTEDTOKEN

Elaboración propia

Tabla N° 4.85
Tarjeta CRC TOKEN_BLACKLIST_BLACKLISTEDTOKEN

CLASE: TOKEN_BLACKLIST_BLACKLISTEDTOKEN	
RESPONSABILIDAD Registrar token del usuario a la blacklist Validar token Decodificar token	COLABORADOR SGSC_USUARIO

Elaboración propia.

Tabla N° 4.86
Tarjeta CRC LEGAL_CATEGORIADELITO

CLASE: LEGAL_CATEGORIADELITO	
RESPONSABILIDAD Registrar categoría delito Clasificar las instancias en delito o falta	COLABORADOR LEGAL_SUBCATEGORIADELITO

Elaboración propia.

Tabla N° 4.87
Tarjeta CRC LEGAL_SUBCATEGORIADELITO

CLASE: LEGAL_SUBCATEGORIADELITO	
RESPONSABILIDAD Registrar sub categoría delito Registrar delito o falta	COLABORADOR LEGAL_TIPODELITO

Elaboración propia

Tabla N° 4.88
Tarjeta CRC LEGAL_TIPODELITO

CLASE: LEGAL_TIPODELITO	
RESPONSABILIDAD Registrar tipo delito Registrar tipo delito incidente sgsc Adicionar contador tipo delito Disminuir contador tipo delito Obtener tipos de delito más usados	COLABORADOR SGSC_INCIDENTE SGSC_INCIDENTE_TIPODELITO

Elaboración propia

Tabla N° 4.89
Tarjeta CRC SGSC_SGSC

CLASE: SGSC_SGSC	
RESPONSABILIDAD Registrar sgsc Registrar sgscID Obtener total delitos incidente Obtener serenos sgsc	COLABORADOR SGSC_INCIDENTE SGSC_INCIDENTE_TIPODELITO SGSC_PUESTO SGSC_SECTOR

Elaboración propia

Tabla N° 4.90
Tarjeta CRC SGSC_SECTOR

CLASE: SGSC_SECTOR

RESPONSABILIDAD Registrar sector Obtener total delitos incidente sector Validar puesto en sector Validar sub sector incluye sector Validar intersección en sector	COLABORADOR SGSC_INCIDENTE SGSC_PUESTO
---	---

Elaboración propia

Tabla N° 4.91
Tarjeta CRC SGSC_PUESTO

CLASE: SGSC_PUESTO	
RESPONSABILIDAD Registrar puesto Obtener total delitos por cercanía a puesto Validar puesto sector	COLABORADOR SGSC_INCIDENTE SGSC_PATRULLAJE

Elaboración propia

Tabla N° 4.92
Tarjeta CRC SGSC_TURNO

CLASE: SGSC_TURNO	
RESPONSABILIDAD Registrar turno Obtener total delitos por turno Obtener serenos por turno Validar turno	COLABORADOR SGSC_INCIDENTE SGSC_INCIDENTETIPODELITO SGSC_PATRULLAJE

Elaboración propia

Tabla N° 4.93
Tarjeta CRC SGSC_CARGO

CLASE: SGSC_CARGO	
RESPONSABILIDAD Registrar cargo Obtener serenos por cargo Obtener serenos que pueden hacer patrullaje	COLABORADOR SGSC_INCIDENTE SGSC_SERENO SGSC_PATRULLAJE

Elaboración propia

Tabla N° 4.94
Tarjeta CRC SGSC_SERENO

CLASE: SGSC_SERENO

RESPONSABILIDAD Registrar sereno Validar sereno cargo Obtener último cargo Validar alias Obtener incidentes registrados Registrar serenoID	COLABORADOR SGSC_INCIDENTE SGSC_BASEPATRULLAJE SGSC_SERENOCARGO
---	---

Elaboración propia

Tabla N° 4.95
Tarjeta CRC SGSC_SERENOCARGO

CLASE: SGSC_SERENOCARGO	
RESPONSABILIDAD Registrar sereno cargo Actualizar último cargo sereno Actualizar incidente, patrullaje sereno	COLABORADOR SGSC_INCIDENTE SGSC_BASEPATRULLAJE

Elaboración propia

Tabla N° 4.96
Tarjeta CRC SGSC_BASEPATRULLAJE

CLASE: SGSC_BASEPATRULLAJE	
RESPONSABILIDAD Registrar base patrullaje Registrar base patrullaje detalle Validar último base patrullaje	COLABORADOR SGSC_BASEPATRULLAJE

Elaboración propia

Tabla N° 4.97
Tarjeta CRC SGSC_INCIDENTE

CLASE: SGSC_INCIDENTE	
RESPONSABILIDAD Registrar incidente Generar código serenoID Generar código sgscID Validar registro persona involucradas Validar registro archivos adicionales Validar registro ubicación Obtener incidentes por filtro dashboard	COLABORADOR SGSC_INCIDENTESERENO SGSC_INCIDENTETIPODELITO

Elaboración propia

Tabla N° 4.98

Tarjeta CRC SGSC_INCIDENTETIPODELITO

CLASE: SGSC_INCIDENTETIPODELITO	
RESPONSABILIDAD Registrar incidente tipo delito Validar sectores Validar puesto Validar unidades Validar fecha incidente Obtener incidentes por filtro nube de delitos	COLABORADOR SGSC_SERENO SGSC_INCIDENTE SGSC_TIPODELITO

Elaboración propia

Tabla N° 4.99

Tarjeta CRC SGSC_PERSONAINVOLUCRADA

CLASE: SGSC_PERSONAINVOLUCRADA	
RESPONSABILIDAD Registrar persona involucrada Validar dni de persona involucrada Validar existencia persona involucrada Registrar persona involucrada incidente	COLABORADOR SGSC_SGSC SGSC_INCIDENTE

Elaboración propia

Tabla N° 4.100

Tarjeta CRC SGSC_INCIDENTEARCHIVO

CLASE: SGSC_INCIDENTEARCHIVO	
RESPONSABILIDAD Registrar incidente archivo Redimensionar imagen Almacenar en el servidor de archivos estáticos (S3)	COLABORADOR SGSC_INCIDENTE SGSC_SGSC

Elaboración propia

F. DIAGRAMA LÓGICO DE LA BASE DE DATOS

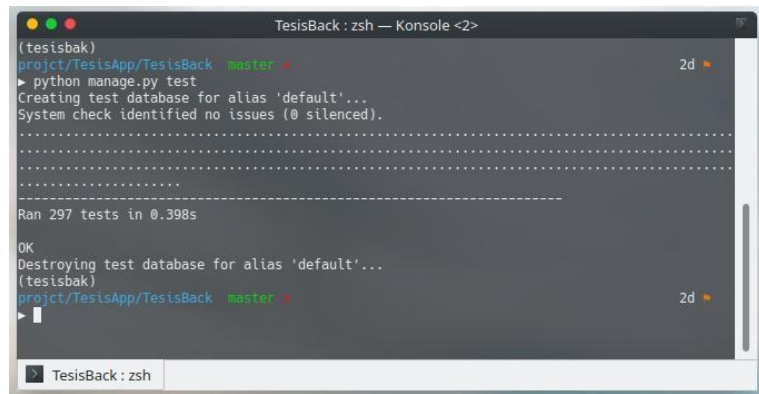
Se muestra en el anexo 3.

G. PRUEBAS UNITARIAS

Para realizar las pruebas unitarias se utilizó el archivo **test.py** de cada aplicación (sgsc, legal, usuario y crimeblog) y el paquete de unittest proporcionado por python 3. Las pruebas unitarias fueron implementadas a la par en todo el proceso de desarrollo del sistema;

En total se hicieron 297 pruebas unitarias y todas fueron superadas; una forma de comprobar es entrar al entorno y correr el script «**python manage.py test**».

Figura N° 4.62



Elaboración propia.

H. PRUEBA DE INTEGRACIÓN

Tabla N° 4.101

Nº H.U.	Nº T. I.	TAREA DE INGENIERÍA	RESULTADO
08	01	Registrar sub gerencia	Satisfactorio
	02	Registrar polígono sub gerencia	Satisfactorio
10	03	Registrar sector	Satisfactorio
	04	Registrar polígono sector	Satisfactorio
	05	Registrar sub sector	Satisfactorio
	06	Registrar polígono sub sector	Satisfactorio
12	07	Registrar puesto	Satisfactorio
	08	Registrar punto para el puesto	Satisfactorio

09	09	Mantener sub gerencia	Satisfactorio
11	10	Mantener sector	Satisfactorio
	11	Mantener sub sector	Satisfactorio
13	12	Mantener puesto	Satisfactorio

Elaboración propia.

Tabla N° 4.102

Reporte pruebas de integración (segunda iteración).

N° H.U.	N° T. I.	TAREA DE INGENIERÍA	RESULTADO
06	13	Estructurar código penal	Satisfactorio
	14	Registrar código penal	Satisfactorio
14	15	Registrar turno	Satisfactorio
16	16	Registrar cargo	Satisfactorio
07	17	Mantener marco legal	Satisfactorio
15	18	Mantener turno	Satisfactorio
17	19	Mantener cargo	Satisfactorio

Elaboración propia.

Tabla N° 4.103

Reporte pruebas de integración (tercera iteración).

N° H.U.	N° T. I.	TAREA DE INGENIERÍA	RESULTADO
04	20	Registrar usuario	Satisfactorio
	21	Registrar sereno	Satisfactorio
01	22	Login usuario	Satisfactorio
	23	Logout usuario	Satisfactorio
	24	Refresh token	Satisfactorio
	25	Integrar redux con el token	Satisfactorio

02	26	Registrar permiso	Satisfactorio
03	27	Mantener permiso	Satisfactorio
05	28	Mantener usuario	Satisfactorio
	29	Mantener sereno	Satisfactorio

Elaboración propia.

Tabla N° 4.104

Reporte pruebas de integración (cuarta iteración).

N° H.U.	N° T. I.	TAREA DE INGENIERÍA	RESULTADO
18	30	Registrar incidente	Satisfactorio
	31	Registrar delito (s)	Satisfactorio
	32	Registrar punto geográfico	Satisfactorio
	33	Registrar archivo adicional (s)	Satisfactorio
	34	Registrar persona involucrada	Satisfactorio
19	35	Mantener incidente	Satisfactorio
	36	Detallar incidente	Satisfactorio

Elaboración propia.

Tabla N° 4.105

Reporte pruebas de integración (quinta iteración).

N° H.U.	N° T. I.	TAREA DE INGENIERÍA	RESULTADO
21	37	Registrar base patrullaje	Satisfactorio
	38	Integrar redux con el árbol de base patrullaje	Satisfactorio
22	39	Mantener base patrullaje	Satisfactorio

Elaboración propia.

Tabla N° 4.106

Reporte pruebas de integración (sexta iteración).

N° H.U.	N° T. I.	TAREA DE INGENIERÍA	RESULTADO
20	40	Generar mapa de incidente	Satisfactorio
	41	Filtrar mapa de incidente	Satisfactorio
	42	Integrar redux con el mapa de incidente	Satisfactorio
23	43	Generar nube de delitos	Satisfactorio
	44	Filtrar nube de delitos (dashboard)	Satisfactorio
	45	Integrar redux con la nube de delitos	Satisfactorio
	46	Generar cuadros de antijuridicidad	Satisfactorio
	47	Integrar redux a los cuadros de antijuridicidad	Satisfactorio
	48	Generar cuadros de culpabilidad	Satisfactorio
	49	Integrar redux a los cuadros de culpabilidad	Satisfactorio

Elaboración propia.

I. PRUEBAS DE ACEPTACIÓN

Tabla N° 4.107

Reporte de casos de prueba de aceptación.

N° H.U.	REQUISITOS	N° C.P.	CASO DE PRUEBA (C. P.)	RESULTADO
01	Acceso al sistema	01	El sistema permite la autenticación de usuarios registrados, la autenticación requiere del ingreso del username y password.	Satisfactorio
		02	El sistema permite eliminar la sesión del usuario, eliminando el estado de autenticado y redirigiendo a la interfaz inicial.	Satisfactorio
02	Registro de permisos	03	El sistema permite registrar permisos y asignarlos a los usuarios, los permisos pueden ser por individual o grupal.	Satisfactorio
		04	El sistema verifica en cada el nivel de acceso del usuario.	
03	Gestión de permisos	05	El sistema permite actualizar permisos, los permisos determinan el acceso de los usuarios al sistema.	Satisfactorio
		06	El sistema permite eliminar permisos	Satisfactorio

04	Registro de usuarios	07	El sistema permite registrar serenos, se valida que el sereno no repita el alias ni el nombre de usuario.	Satisfactorio
		08	El sistema permite registrar usuarios, previo registro de serenos, un sereno puede o no ser un usuario, un usuario siempre es un sereno.	Satisfactorio
05	Gestión de usuarios	09	El sistema permite actualizar serenos	Satisfactorio
		10	El sistema permite actualizar usuarios	Satisfactorio
		11	El sistema permite eliminar serenos	Satisfactorio
		12	El sistema permite eliminar usuarios	Satisfactorio
06	Registro de marco legal	13	El sistema estructura el código penal	Satisfactorio
		14	El sistema permite registrar el código penal (tipo delito)	Satisfactorio
07	Gestión del marco legal	15	El sistema permite eliminar el marco legal (tipo delito)	Satisfactorio
		16	El sistema permite actualizar el marco legal (tipo delito)	Satisfactorio
08	Registro de sub gerencias	17	El sistema permite registrar sub gerencia de seguridad ciudadana (sgsc)	Satisfactorio
		18	El sistema permite registrar el área de una sgsc (polígono geográfico)	Satisfactorio
09	Gestión de sub gerencias	19	El sistema permite actualizar sub gerencia de seguridad ciudadana (sgsc)	Satisfactorio
		20	El sistema permite eliminar sub gerencia de seguridad ciudadana (sgsc)	Satisfactorio
10	Registro de sectores y sub sectores	21	El sistema permite registrar sectores	Satisfactorio
		22	El sistema permite registrar un sector de la instancia sector (polígono geográfico)	Satisfactorio
		23	El sistema permite registrar sub sectores	Satisfactorio
		24	El sistema permite registrar un sector de la instancia sub sector (polígono geográfico)	Satisfactorio
11	Gestión de sectores y sub sectores	25	El sistema permite actualizar sectores	Satisfactorio
		26	El sistema permite eliminar sectores	Satisfactorio
		27	El sistema permite actualizar sub sectores	Satisfactorio
		28	El sistema permite eliminar sub sectores	Satisfactorio
12	Registro de puestos	29	El sistema permite registrar puestos	Satisfactorio
		30	El sistema permite registrar una ubicación de la instancia puesto (punto geográfico)	Satisfactorio
13	Gestión de puestos	31	El sistema permite actualizar puestos	Satisfactorio
		32	El sistema permite eliminar puestos	Satisfactorio

14	Registro de turnos	33	El sistema permite registrar turnos	Satisfactorio
15	Gestión de turnos	34	El sistema permite actualizar turnos	Satisfactorio
		35	El sistema permite eliminar turnos	Satisfactorio
16	Registro de cargos	36	El sistema permite registrar cargo	Satisfactorio
17	Gestión de cargos	37	El sistema permite actualizar cargo	Satisfactorio
		38	El sistema permite eliminar cargo	Satisfactorio
18	Registro de incidentes	39	El sistema permite registrar incidentes	Satisfactorio
		40	El sistema permite registrar los tipos de delito	Satisfactorio
		41	El sistema permite registrar los archivos adicionales	Satisfactorio
		42	El sistema permite registrar las personas involucradas.	Satisfactorio
19	Gestión de incidentes	43	El sistema permite actualizar incidentes	Satisfactorio
		44	El sistema permite eliminar incidentes	Satisfactorio
20	Gestión mapa de incidentes	45	El sistema permite generar mapa de incidentes los tipos de delito, sectores y serenos participantes.	Satisfactorio
		46	El sistema permite realizar múltiples filtros sobre los datos cargados en memoria.	Satisfactorio
21	Registro de base patrullaje	47	El sistema permite registrar base patrullaje, asignando el cargo, turno y los días de descanso.	Satisfactorio
22	Gestión de base patrullaje	48	El sistema permite actualizar base patrullaje	Satisfactorio
		49	El sistema permite eliminar base patrullaje	Satisfactorio
23	Gestión de tipos de delito	50	El sistema permite generar la nube de delitos	Satisfactorio
		51	El sistema permite generar cuadros estadísticos sobre los tipos de delito (delitos por sector, hora y día)	Satisfactorio
		52	El sistema permite generar cuadros estadísticos sobre la antijuridicidad de los incidentes delitos (sector y día)	Satisfactorio
		53	El sistema permite generar cuadros estadísticos sobre la culpabilidad de los incidentes delitos (sector y día)	Satisfactorio

Elaboración propia.

CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- a.** Se describió la tipicidad presentando el mapa del delito, representadas en las figuras 4.35 “Mapa del delito del distrito Jesús Nazareno”, 4.36 “Mapa del delito con zoom del distrito de Jesús Nazareno”, 4.40 “Mapa del delito del distrito de Carmen Alto” y 4.41 “Mapa del delito con zoom del distrito de Carmen Alto”; que muestran los incidentes delictivos de sus respectivos distritos; del mismo modo, en las figuras 4.45 “Nube de delitos del distrito de Jesús Nazareno”, cuadro 4.1 “Frecuencia por sectores de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Jesús Nazareno”, cuadro 4.2 “Frecuencia por horas del día de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Jesús Nazareno”, cuadro 4.3 “Frecuencia por días de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Jesús Nazareno”, cuadro 4.4 “Frecuencia por meses de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Jesús Nazareno”, figura 4.49 “Nube de delitos del distrito de Carmen Alto”, cuadro 4.5 “Frecuencia por sectores de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Carmen Alto”, cuadro 4.6 “Frecuencia por horas del día de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Carmen Alto”, cuadro 4.8 “Frecuencia por días de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Carmen Alto” y cuadro 4.8 “Frecuencia por meses de las lesiones graves por violencia contra las mujeres e integrantes del grupo familiar del distrito de Carmen Alto”, que muestran el detalle del tipo de delito seleccionado de sus respectivos distritos.
- b.** Se describió la antijuridicidad de los incidentes delictivos presentados en las figuras 4.51 “Frecuencia por sectores de la antijuridicidad del distrito de Jesús nazareno”, 4.52 “Frecuencia por días de la antijuridicidad del distrito de Jesús nazareno”, 4.54 “Frecuencia por sectores de la antijuridicidad del distrito de Carmen Alto” y la figura 4.55 “Frecuencia por días de la antijuridicidad del distrito de Carmen Alto”; que muestran los incidentes delictivos que son típicos y antijurídicos de sus respectivos

distritos

- c. Se describió la culpabilidad de los incidentes delictivos presentados en las figuras 4.57 “Frecuencia por sectores de la culpabilidad del distrito de Jesús nazareno”, 4.58 “Frecuencia por días de la culpabilidad del distrito de Jesús nazareno”, 4.60 “Frecuencia por sectores de la culpabilidad del distrito de Carmen Alto” y la figura 4.61 “Frecuencia por días de la culpabilidad del distrito de Carmen Alto”; que muestran los incidentes delictivos que son típicos, antijurídicos y con culpabilidad de sus respectivos distritos

5.2 RECOMENDACIONES

- a. Los gobiernos locales, en especial las sub gerencias de seguridad ciudadana, deberían apoyarse más en las tecnologías de información, a fin de tomar estrategias más acertadas al momento de abordar la inseguridad ciudadana.
- b. Es importante tomar conocimiento y estrategias para abordar la cifra negra, cifra que representa los incidentes delictivos que no son reportados; en la presente investigación no se tomó consideración sobre la cifra negra.
- c. Capacitar a los serenos en materia legal para la correcta descripción de los incidentes delictivos, considerando el código penal como marco de referencia para los tipos de delito; del mismo modo, crear un marco de referencia compartido por las demás sub gerencias de seguridad ciudadana para cubrir los puntos débiles y casos excepcionales que el código penal no cubre.
- d. Es importante el uso de tecnologías geográficas para la descripción del mapa del delito, ya que facilita la integración de datos espaciales con datos estructurados y no estructurados; permitiendo realizar consultas convencionales y geográficas.
- e. Al estructurar los datos sobre la actividad delictiva se abre la posibilidad de desarrollar aplicaciones móviles (apps) que permitan un trato más cercano por parte del sereno al vecino.

REFERENCIA BIBLIOGRÁFICA

1. Andrea Marcela Behar y Patricia Lucilli (2003). *Mapa del delito de la ciudad autónoma de buenos aires*.
2. ARCGIS (2013). *Georreferenciación y sistemas de coordenadas*. recuperada de: <http://resources.arcgis.com/es/help/getting-started/articles/026n00000000s0000000.htm>.
3. Bell, D., Parr, M. (2003). *Java*. México: Pearson educación.
4. Borja, y. (2013). *metodología ágil de desarrollo de software – xp. espe*, MEVAST.
5. Burrough, p.a. (1986). *Principles of geographic information systems for land resource assessment. monographs on soil and resources survey no. 12*, oxford science publications, new york.
6. Bustamante, D., Rodríguez, J. (2014). *Metodología actual, metodóloga xp*. Barinas, marzo 2014.
7. Cafassi, E. (1998). *Internet: políticas y comunicaciones*. Buenos Aires, Argentina: biblos.
8. Carlos Vázquez González y Carles Soto Urpina (2013). *El análisis geográfico del delito y los mapas de la delincuencia*.
9. Cecilia Riveros Cáceres y Mario Alonso Zevallos Yapias (2016). *Business intelligence para el área de seguridad ciudadana en el distrito de villa el salvador mediante la metodología de ralph kimball*. (tesis de grado). Universidad Nacional Mayor de San Marcos.
10. Colección Esencial (2011). *Esencial internet explorer 9*. Catalauña, España: Editions ENI.
11. Comité distrital de seguridad ciudadana (2018). *Plan local de seguridad ciudadana Andrés Avelino Cáceres Dorregaray, 2018*.
12. Dávila Martínez, Francisco Javier; Camacho Arranz, Elena (2012). *Georreferenciación de documentos cartográficos para la gestión de archivos y cartotecas 'propuesta metodológica*. revista catalana de geografía, recuperada de: <http://www.rcg.cat/articles.php?id=252>.
13. Duran, F., Gutiérrez, F., Pimentel, E. (2007). *Programación orientada a objetos con java*. Madrid, España: Paraninfo s. a.
14. Edmund Mezger (1955). *Derecho penal parte general*, Editorial bibliográfica

Argentina

15. Environmental Systems Research Institute Inc (ESRI, 1998). *Esri educational services*
16. Flórez, H. (2012). *Programación orientada a objetos usando java*. Bogotá, Colombia: Ecoe Ediciones.
17. Francisco Muñoz Conde; Mercedes Garcia Aran (2002). *Mercedes, derecho penal parte general (5ª ed)*, valencia, España: Tirant lo blanch.
18. Fray León Osorio Rivera (2008). *Lógica y programación orientada a objetos: un inicio al desarrollo de software*. Editorial: Itm.
19. Groussard, T. (2012). *Java 7: los fundamentos del lenguaje java*. Barcelona: ediciones ENI.
20. Gutierrez, J. y Tena, J. (2003). *Protocolos criptográficos y seguridad en redes*. Cantabria, España: graficas calima.
21. Haithcoat, T., L. Warnecke, y Z. Nedovic-Budic (2001). *Geographic information technology in local government: experience and issues. the municipal year book 2001*. Washington, D.C.: International city/county management association.
22. Hanantek (2010). *Documentación generada para programación extrema*, recupera el 23 de mayo del 2018 de: <http://www.hanantek.com/es/documentacion-programacion-extrema>.
23. Ignacio Pervesi (2007). *Aplicación de minería de datos para la exploración y detección de patrones delictivos en argentina*.
24. Instituto Nacional de Estadística e Informática (INEI, 2017). *Estadísticas de seguridad ciudadana de seguridad ciudadana (mayo - octubre 2017) informe técnico no 6 - noviembre 2017*.
25. Instituto nacional de estadística y geografía (INEGI, 2014). *Sistema de información geográfica*, México. México.
26. Jorge Julio Jaulis Rua y Jonathan Renatto Vilcarromero Giraldo (2015). *Sistema de predicción de hechos delictivos para la mejora del proceso de prevención del delito en el distrito de la molina utilizando minería de datos*. (tesis de grado). Universidad de San Martín de Porres.
27. José Antonio de Urbina (2001) *El gran libro del protocolo*. Madrid, España: temas de hoy.

28. Joskowicz, J. (2008). *Reglas y prácticas en extreme programming*. Universidad de Vigo, España.
29. Juan Carlos Portocarrero Z (2017). *Teoría del delito, procurador de la procuraduría publica especializada en delitos de orden público*.
30. Kroenke, d. (2003). *Procesamiento de base de datos: fundamentos, diseño e implementación*. (8a ed.). Juárez, México: Pearson.
31. Laura Vozmediano (2011). *Criminología ambiental: ecología del delito y de la seguridad*.
32. Lopez Barja De Quiroga, Jacobo (2004). *Derecho penal. parte general: introducción a la teoría jurídica del delito*.
33. Lujan, S. (2001). *Programación en internet: cliente web. alicante*. España: club universitario.
34. Mario Pablo Rodríguez Hurtado, Ángel Fernando Ugaz Zegarra, Lorena Mariana Gamero Calero, Horst Schönbohm (2012). *Manual de casos penales, la teoría general del delito y su importancia en el marco de la reforma procesal penal*. Editorial Nova.
35. Meléndez, S., Gaitán, M. y Pérez, N. (2016). *Sistema web de evaluación al desempeño docente unan-managua, empleando la metodología ágil programación extrema, 2015* (tesis de pregrado). Universidad Nacional Autónoma de Nicaragua.
36. Municipalidad distrital de Carmen Alto (2018). *Plan local de seguridad ciudadana plsc Carmen Alto, 2017*.
37. Municipalidad distrital de San Juan Bautista (2018). *Plan local de seguridad ciudadana de San Juan Bautista, 2018*.
38. Municipalidad provincial de Huamanga (2018). *Plan distrital de seguridad ciudadana Ayacucho, 2018*.
39. Óscar Peña, Gonzales (2010). *Teoría del delito, manual práctico para su aplicación en la teoría del caso*.
40. Percy Nicolas Mora Luque (2015). *Uso de tecnologías para sistematización de la información sobre el crimen (usos, problemas de georreferencia y demás)*. (tesis de grado). Universidad Pontificia la Católica del Perú.
41. Ramos, A. (2007). *Operaciones con base de datos ofimáticas y corporativas*. recuperado de <https://books.google.com.pe/books/operaciones-con-base-de>

datos-ofimaticas .

42. Romero, L. (1997). *Publicar en internet: guía práctica para la creación de documentos html*. Cantabria, España: universidad de Cantabria.
43. Roxin, Claus. (1997). *Derecho penal. parte general. tomo i. fundamentos: la estructura de la teoría del delito*. (d.- m. luzón peña, m. díaz y garcía conlledo y javier de vicente remezal, trad. y notas). Madrid, España: Editorial Civitas.
44. Ruiz García (2012), *Crimen y seguridad. análisis, predicción y prevención del fenómeno criminal (máster en tecnologías de la información geográfica)*.
45. Sherman (1995). *General deterrent effects of police patrol in crime "hot spots"*
46. Somerville I. (2005). *Ingeniería de software (7^{ma} ed.)*. España. Pearson addison wesley
47. Stangeland y Garrido (2005). *El mapa del crimen. herramientas informáticas para policías y criminólogos*. Tirant lo blanch, (issn 1696-9219).
48. Zaffaroni Eugenio Raúl, Plagia Alejandro Y Slokar Alejandro (2005). *Manual de derecho penal. parte general. (2 da reimpresión)*. Buenos Aires - Argentina.