# ESCUELA POLITÉCNICA NACIONAL

## FACULTAD DE INGENIERÍA DE SISTEMAS

## EVALUACIÓN DE ALGORITMOS DE MINERÍA DE DATOS PARA DETECCIÓN Y PREDICCIÓN DE ATAQUES DE INYECCIÓN SQL EN BIG DATA

## EVALUACIÓN DEL ALGORITMO NAIVE BAYES PARA LA DETECCIÓN Y PREDICCIÓN DE ATAQUES DE INYECCIÓN SQL EN BIG DATA

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SOFTWARE**

**STEVEN JAVIER RIVERA TENELANDA**
**steven.rivera@epn.edu.ec**

**DIRECTORA: PhD. GABRIELA LORENA SUNTAXI OÑA**
**gabriela.suntaxi@epn.edu.ec**

**DMQ, septiembre 2022**

# CERTIFICACIONES

Yo, Steven Javier Rivera Tenelanda, declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

**STEVEN JAVIER RIVERA TENELANDA**

Certifico que el presente trabajo de integración curricular fue desarrollado por Steven Javier Rivera Tenelanda, bajo mi supervisión.

**PhD. Gabriela Lorena Suntaxi Oña**
**DIRECTORA DE PROYECTO**

Certificamos que revisamos el presente trabajo de integración curricular.

**Nombre1 Nombre2 Apellido1 Apellido2**
**REVISOR 1 DEL TRABAJO**
**DE INTEGRACIÓN CURRICULAR**

**Nombre1 Nombre2 Apellido1 Apellido2**
**REVISOR 1 DEL TRABAJO**
**DE INTEGRACIÓN CURRICULAR**

# DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

STEVEN JAVIER RIVERA TENELANDA

DRA. GABRIELA LORENA SUNTAXI OÑA

**Colaboradores del proyecto integrador:**

ANDRÉS MAURICIO LLUMIQUINGA GUAMBA

BRYAN ANDRÉS PALMA PONCE

EDISON JAVIER QUIMBIAMBA GUASGUA

# DEDICATORIA

Dedico este trabajo a mi madre y a mi padre que siempre me inculcaron la educación y me dieron su apoyo incondicional en todo momento, hicieron hasta lo imposible para que no me faltara nada a mí y a mis hermanos. A mis hermanos, que siempre me ayudaron de diferentes maneras a que pueda estudiar y a terminar mi carrera.

# AGRADECIMIENTOS

Agradezco a mi madre fallecida Teresa y mi padre Ángel, que siempre me inculcaron buenos valores y estuvieron conmigo en los momentos malos y buenos. A mi tía fallecida Libia, que, aunque vivía en el exterior, siempre estuvo apoyándome de diferentes formas y alentándome a que me esfuerce en mi carrera. A mis hermanos Erick y Joel, que siempre me acompañaron en todo momento de mi carrera, brindándome apoyo, consejos y que algunas veces me hacían olvidar de mis preocupaciones con su humor. A mis amigos Bryan Palma y Edison Quimbiamba, que me brindaron su amistad y su ayuda durante toda la carrera. A la Dra. Gabriela Suntaxi, mi directora de este trabajo, que tuvo paciencia para guiarme durante todo este proceso académico y darle las gracias por motivarme a culminar rápido este trabajo.

# CONTENIDO

# RESUMEN

En la época actual, los Ataques de Inyección SQL (SQLIA) están ubicados en el tercer lugar de lista OWASP (Open Web Application Security Project) TOP 10 del año 2021 [1], dado que para un atacante informático es relativamente fácil ejecutar este tipo de ataque. No obstante, cuando un atacante informático ejecuta, este tipo de ataque puede causar un gran impacto en los sistemas de información. Y teniendo el antecedente de que para combatir este tipo de ataque resulta complejo para las organizaciones, este componente se encarga de investigar de forma sistemática la literatura en cuanto a la utilización de algoritmos y técnicas de minería de datos para detección y prevención de SQLIA. Luego, tomando esta investigación como referencia, se selecciona un algoritmo de minería de datos que en este caso es Naive Bayes y se lleva a cabo la evaluación de este algoritmo, con el objetivo de determinar que tan efectivo es al momento de detectar sentencias SQL que puedan ser maliciosas. A continuación, se desarrolla un prototipo de sistema web que ayude a analizar los logs de los registros utilizando el algoritmo Naive Bayes para detectar y predecir SQLIA. El objetivo principal de este componente es mejorar la seguridad de los sistemas de información de la organización, en la cual se va a implementar el sistema desarrollado.

**PALABRAS CLAVE:**Minería de Datos, CRISP-DM, Naive Bayes, SQLIA, Seguridad de la Información

# ABSTRACT

In the current era, SQL Injection Attacks (SQLIA) are ranked third in the OWASP (Open Web Application Security Project) TOP 10 list for the year 2021 [1] since it is relatively easy for a computer attacker to execute this type of attack. However, when a computer attacker executes, this type of attack can have a great impact on information systems. Furthermore, knowing that combating this type of attack is complex for organizations, this component is responsible for systematically investigating the literature on the use of algorithms and data mining techniques for detection and prevention of SQLIA. Then, taking this research as a reference, a data mining algorithm is selected, which in this case is Naive Bayes, and the evaluation of this algorithm is carried out to determine how effective it is at detecting SQL sentences that may be malicious. Next, a prototype web system is developed to help analyze log records using the Naive Bayes algorithm to detect and predict SQLIA. The main objective of this component is to improve the security of the organization's information systems, in which the developed system will be implemented.

**KEYWORDS:** Data Mining, CRISP-DM, Naive Bayes, SQLIA, Information Security

# 1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

En la época que vivimos, los sistemas de información que utilizan las organizaciones, son cada vez son más complejos de desarrollar y mantener porque están involucradas diferentes características a tomar en cuenta. Una de ellas es la seguridad de estos sistemas. Por lo tanto, teniendo esta problemática en mente, varias organizaciones se encargan de definir y clasificar estas vulnerabilidades y amenazas con el objetivo de que las organizaciones conozcan los métodos que usan los atacantes informáticos para vulnerar los sistemas de información. Una de estas organizaciones es OWASP, que cada periodo de tiempo actualiza su lista de principales ataques y lleva el nombre de OWASP Top 10.

En esta lista elaborada por OWASP, se evidencia que los ataques de inyección SQL (SQLIA) caben como una de las amenazas más importantes dentro de la seguridad informática, dado que tienen un alto impacto en las organizaciones y posee una gran facilidad para ejecutar este tipo de ataque. Teniendo como evidencia la complejidad de los sistemas actuales y su compleja lucha por controlar manualmente este tipo de ataque, es importante aportar con soluciones que permitan mitigar esta amenaza de forma automática. Por lo tanto, es necesario realizar una revisión de la literatura para informarnos sobre como está el estado de las investigaciones con respecto a las técnicas de detección y predicción de este tipo de ataque presentado en la lista de OWASP.

En lo que respecta a las organizaciones, estas actualmente generan una gran cantidad de datos, que contienen información muy valiosa para personas externas a la organización, como pueden ser información personal, comportamientos y preferencias de los usuarios. Teniendo en cuenta el antecedente anterior, las empresas no tienen un control exhaustivo por proteger esta información y por esta razón son víctimas de ataques a sus sistemas de información. Por lo tanto, en este campo ayuda la minería de datos, que en sí su premisa es la de crear conocimiento a partir de los datos. Entonces, utilizando este contexto, se podría utilizar la minería de datos para detectar o predecir posibles ataques dentro de las bases

de datos, mediante técnicas de Machine Learning. Entonces la motivación que nace en la realización de este proyecto es la de proteger la información que se encuentra en las bases de datos, para esto se propone desarrollar una solución a esta problemática mediante la minería de datos utilizando técnicas de Machine Learning en un sistema web. En donde se va a analizar los datos que se crean en una organización en un caso de estudio real, con el objetivo de detectar posibles ataques en sus sistemas, para que la organización pueda tomar medidas correctivas oportunas para protegerse de estos ataques.

Tomando en cuenta el tiempo y la complejidad involucrada en analizar los algoritmos y técnicas que actualmente existen para la detección de ataques, esta investigación se encuentra como una parte dentro un solo proyecto general. Este proyecto está dividido en cuatro componentes que tienen la finalidad de evaluar los algoritmos más utilizados en la minería de datos para la detección y prevención de SQLIA. En cada componente se evalúa un algoritmo diferente para así cubrir la mayor cantidad de algoritmos puestos a prueba.

El presente trabajo se encarga de cubrir un componente de los cuatro totales. Es necesario mencionar que como la investigación de los algoritmos de minería de datos se la realizó en conjunto con los otros componentes, habrá secciones en las cuales son comunes para todos los otros componentes, que serán indicadas oportunamente al inicio de cada sección del presente documento.

## 1.1 OBJETIVO GENERAL

Realizar la evaluación de un algoritmo de minería de datos para la detección y predicción de ataques de inyección SQL en Big Data, aplicado a un caso de estudio real.

## 1.2 OBJETIVOS ESPECÍFICOS

- ❒ Realizar una revisión de los algoritmos más utilizados en el ámbito de la minería de datos.
- ❒ Realizar el análisis y la evaluación de un algoritmo de minería de datos para la detección de ataques de inyección SQL.
- ❒ Desarrollar un prototipo de software para la detección y predicción de ataques de

inyección SQL en Big Data utilizando un algoritmo de minería de datos.

- ❒ Evaluar el prototipo de software desarrollado en un caso de estudio utilizando datos reales.

- ❒ Realizar un análisis y comparación de los resultados obtenidos de otros algoritmos de minería de datos.

## 1.3 ALCANCE

El alcance previsto para el presente trabajo se describe a continuación, donde se utilizará una variación de la metodología descrita en [2] para el desarrollo del proyecto. Esta metodología está conformada por 5 fases las cuales se describen a continuación:

- ❒ **Fase de planificación:** se definirán los requerimientos y las características generales del proyecto, como la justificación del proyecto, recursos para el desarrollo del proyecto, cronograma, etc.

- ❒ **Fase de diseño:** en la fase de diseño se realizará una revisión sistemática de los algoritmos más utilizados en el ámbito de la minería de datos para la detección y prevención de SQLIA, utilizando una adaptación de la metodología descrita en [3]. Luego, partiendo de esta revisión, se seleccionará un algoritmo de minería de datos para ser evaluado, así como las métricas que serán consideradas para su evaluación.

- ❒ **Fase de implementación:** en esta fase se utilizará la metodología de minería de datos CRISP-DM [4] para obtener un modelo que permita definir la efectividad de este algoritmo en detección de ataques SQL en Big Data. A continuación, se desarrollará un prototipo de software utilizando la metodología de desarrollo XP [5] que permitirá detectar y predecir los SQLIA utilizando el algoritmo de minería de datos seleccionado.

- ❒ **Fase de evaluación:** en esta fase se analizará los resultados obtenidos del proceso de minería de datos en un caso de estudio real.

- ❒ **Fase de comunicación:** finalmente, en la fase de comunicación se presentará un informe con los análisis y los resultados obtenidos de la evaluación de este prototipo software.

# 2 MARCO TEÓRICO

## 2.1 SEGURIDAD EN SISTEMAS DE INFORMACIÓN

En la época actual, tanto las personas y organizaciones hacen uso de dispositivos inteligentes para interactuar entre sí, accediendo a una gran variedad de información de diferente índole. Razón por lo cual las organizaciones deben mantener la seguridad en los sistemas de información porque esta información puede ser vulnerada por personas no autorizadas y hacer uso de la misma para sus beneficios personales.

La seguridad en sistemas de información es una disciplina muy amplia que siempre está en constante evolución. La finalidad de la seguridad es permitir que la organización alcance sus objetivos de negocio, implementando sistemas que sean confiables frente a los riesgos relacionados con las TIC de la organización, sus clientes, socios comerciales, etc [6].

Cuando se habla de seguridad de la información es común que se la relacione con la triada CIA (Confidencialidad, Integridad y Disponibilidad) debido a que es la primera línea de acción para proteger los activos una organización [7].

### 2.1.1 Tríada CIA

Según un estudio sobre la historia de la triada CIA realizado en [8], nos menciona que a medida que la tecnología disminuía su costo y aumentaba su uso, la protección a los computadores y su información se volvió más importante. Por lo tanto, surgió un modelo común que se toma como base para el desarrollo de sistemas de información. El cual se centra en tres puntos focales que son independientes y que guían a los equipos de seguridad para identificar las diferentes formas en las que se puede proteger la información.

- **Confidencialidad**

  La confidencialidad es la responsable de asegurar que la información se mantenga en secreto o privada. Las medidas en las que se centra la confidencialidad es la de evitar el acceso a la información sensible por parte de personas no autorizadas.

- **Integridad**

  La integridad es la responsable de que la información sea confiable y no tenga algún tipo de manipulación por parte de personas no autorizadas. La integridad de la información se asegura solo si los datos son confiables, precisos y auténticos.

- **Disponibilidad**

  Asegurarse de que la información sea confiable e integra es inútil si esta no está disponible para las personas autorizadas que utilizan esta información en la organización. Esto significa que todo el hardware y software involucrado en estos sistemas debe funcionar todo el tiempo sin interrupciones. Además, las personas que tienen acceso a información específica puedan usarla cuando lo necesiten y con un tiempo de respuesta reducido.

Los avances tecnológicos traen consigo varias ventajas para los sistemas de información, entre una ellas es su capacidad mejorada para almacenar, analizar y procesar grandes volúmenes de datos, por lo que tener una buena gestión de los riegos asociados al Big Data es de vital importancia para las organizaciones.

## 2.2 BIG DATA

El término "Big Data"tiene su origen asociado a que cada día se están creando una gran cantidad de datos, por ejemplo, según un estudio que analiza el crecimiento de la información [9], los datos que se producen actualmente se estiman que tienen un orden de zettabytes y tienen un ritmo de crecimiento constante del 40 por ciento por año. Tomando la premisa anterior, hoy en día los analistas de datos están en la obligación de manejar grandes cantidades de datos, por lo tanto, están en búsqueda constante de nuevos algoritmos y herramientas para manejar estos datos. Según Doung Laney [10] que fue el primero en mencionar las 3 V para la gestión de Big Data propone lo siguiente:

- **Volumen:** El volumen se refiere a la gran cantidad de datos que existen actualmente

en las bases de datos de todo el mundo. Si este volumen de los datos es sumamente grande, se los puede considerar como Big Data.

- **Variedad:** La variedad hace referencia a la diversidad de los tipos de datos que existen actualmente. Las organizaciones pueden recolectar datos de diferentes fuentes, pero el verdadero desafío es la estandarización de los mismos, es decir, transformar los tipos de datos en un solo tipo y trabajar sobre este único tipo de dato.
- **Velocidad:** La velocidad hace referencia a que tan rápido se están generando los datos y que tan rápido se están moviendo. Las organizaciones necesitan que su sus datos fluyan de forma rápida y se encuentren disponibles de todo momento, con el objetivo de poder tomar las mejores decisiones comerciales.

Pero en la época actual, existen dos V más:

- **Veracidad:** La variabilidad hace referencia a la precisión y a la calidad que tiene los datos. Los datos que han sido recopilados por las organizaciones, pueden tener partes faltantes, no ser exactos, por lo tanto, se requiere un cierto nivel de confianza para la utilización de estos datos.
- **Valor:** Se refiere al valor que pueden proporcionar las grandes cantidades de datos y tiene una relación directa con lo que la organización puede realizar o sacar ventaja de esos datos. El poder extraer el valor que tienen estos datos es necesario, dado que el valor de estos datos va a aumentar si la información que se obtiene de ellos ayuda a la organización a obtener una ventaja competitiva sobre las otras organizaciones.

Tomando en cuenta que actualmente las organizaciones manejan grandes volúmenes de datos, es de vital importancia asegurar que los motores de bases de datos donde se guarda dicha información se mantenga segura, por lo tanto, es importante conocer cuáles son las principales ataques relacionados con las bases de datos.

### 2.2.1 Ataques a bases de datos

Según el estudio de "Los métodos de control de la seguridad de las bases de datos" de Malik [11], el principal activo que buscan los atacantes informáticos son las bases de datos, dado que mediante estas, ellos pueden obtener información de vital importancia para la

organización, por lo tanto, estas organizaciones se enfrentan a diversos tipos de ataques y se describen a continuación:

- **Privilegios excesivos:** En las bases de datos, cuando se produce un exceso de privilegios, el atacante informático obtiene privilegios para fines no autorizados para acceder a cierta información que no le pertenece y de estar manera obtener algún beneficio personal.

- **Malware:** consiste en la utilización de una gran variedad de ataques avanzados combinados con tácticas como el uso de correos electrónicos, de phishing dirigidos o malware, con el objetivo de ingresar a los sistemas de información para obtener información confidencial. Como el empleado de la organización no tiene conocimiento de que su equipo está infectado, este se vuelve un conductor para que el atacante acceda a su información cuando este lo desee [11].

- **Seguimiento de auditoría débil:** este tipo de ataque se produce cuando no existe un control relacionado con las políticas de auditoría de base de datos. Por ejemplo, no se lleva un control de las personas que modifican alguna tabla en una base datos, perdiéndose de esta forma el vínculo que tiene el usuario responsable de su cambio.

- **Exposición de respaldos:** según Malik [11] en las organizaciones, los medios de almacenamiento de respaldo de bases de datos por lo general están completamente desprotegidos contra ataques o hurtos. Dando como resultado que se presenten diversas brechas de seguridad que involucran el robo de discos o cintas de respaldo de las bases de datos, exponiendo información sensible de la organización.

- **Autenticación débil:** dentro de una organización, cuando no existe el debido control de los esquemas de autenticación, estos dan paso a que los atacantes puedan hacerse pasar por un usuario legítimo de una base datos, teniendo acceso a información valiosa. Algunos tipos de estrategias más comunes utilizadas por estos atacantes son la ingeniería social y los ataques de fuerza bruta [11].

- **Vulnerabilidades en la Base de datos y mala configuración:** según Malik [11] es frecuente encontrarse con bases de datos vulnerables, sin actualizaciones de los parches de seguridad, parámetros y cuentas por defecto. Por lo tanto, los atacantes se aprovechan de estas vulnerabilidades para orquestar sus ataques a la organización.

- **Datos sensibles si buena administración:** Malik [11] menciona que las organizaciones ponen todo su esfuerzo por mantener un inventario correcto de las bases de datos, pero esto no es suficiente, dado que si existen casos de que sus empleados se olvidan de la existencia de una base datos, por ejemplo en un entorno de prueba. Entonces es aquí donde los atacantes toman control de esta base de datos para acceder a la información confidencial de la organización.

- **Denegación de servicios:** este tipo de ataque consiste en negar el acceso o servicio, a un usuario legítimo de una aplicación o una base datos.

- **Experiencia y educación limitada en temas de seguridad:** consiste en la falta de experiencia y capacitación necesaria por parte de los empleados de una organización, a aplicar ciertos controles de seguridad, hacer cumplir las políticas acordadas de seguridad o no llevar a cabo los procesos de respuesta a incidentes.

- **Ataques de inyección SQL (SQLIA):** este tipo de ataque consiste en que un atacante informático interfiere en las consultas realizadas a una base de datos. Normalmente, el atacante ingresa código de consulta en un formulario y con esto puede observar o modificar cierta información en la base de datos que normalmente no está disponible para un usuario común del sistema.

Como en la presente investigación se centró en formas de mitigar los SQLIA y es el punto central que dirige este proyecto, se profundizara en este tipo de ataque.

### 2.2.2 Ataques de inyección SQL

Para comprender este tipo de ataque es necesario conocer sobre el lenguaje de consulta estructura (SQL), que básicamente es un lenguaje de alto nivel utilizado para la gestión de sistemas de bases de datos. Este lenguaje permite al usuario realizar operaciones CRUD(Create, Read, Update y Delete) sobre los datos.

Por lo, tanto los SQLIA se producen en aplicaciones web que utilizan bases de datos para guardar información y consiste en insertar una o varias consultas SQL con carácter malicioso mediante una entrada de datos, como por ejemplo un formulario en la aplicación web.

Si el ingreso de los datos suministrados por el usuario se realiza de forma insegura, entonces la aplicación web se vuelve completamente vulnerable a este tipo de ataque. Dando acceso al usuario malicioso a la base de datos y de esta forma pueda comprometer su seguridad y teniendo acceso completo a información confidencial.

En la Figura 2.2 se observa el esquema general de un ataque de inyección SQL (SQLIA).

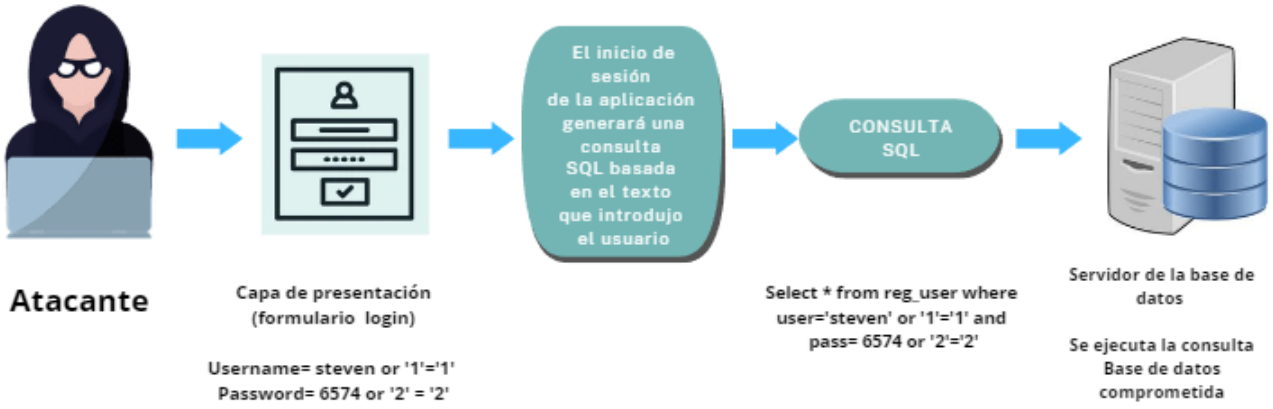


**Figura 2.1:** Esquema general de un ataque de inyección SQL (SQLIA). Fuente: Autoría propia

### 2.2.3 Tipos de ataques de inyección SQL

Según [12] los tipos de SQLIA se pueden clasificar de la siguiente forma:

**a. Tautologías**

**Objetivo del atacante:** El objetivo principal del atacante es evitar la autenticación, además de extraer la información de la organización.

**Descripción:** Las sentencias ingresadas por el atacante utilizan condicionales en los parámetros de consulta, de tal forma que esta siempre es verdadera.

**Ejemplo:**

```
Select * from emp_inf where=" or '7=7';
```

En el ejemplo anterior se observa que la consulta tiene un condicional que siempre va a ser verdadero, por lo tanto, el atacante obtendrá toda la información del los empleados de la organización.

**b. Consultas lógicamente incorrectas**

**Objetivo del atacante:** El objetivo principal del atacante es extraer información de la base datos e identificar patrones en mensajes de error que pueden inyectarse en las consultas.

**Descripción:** Consiste en que las consultas no válidas ingresadas por el atacante se ejecutan provocando que se muestren mensajes de error que brindan información sobre el tipo de datos o el nombre de la tabla a la cual se está consultando.

**Ejemplo:** Funciones de agregación que pueden ser aplicadas a diferentes tipos de datos como varchar o utilizar las cláusulas:

```
'having' o 'group by'
```

**c. Consultas de unión**

**Objetivo del atacante:** Evitar la autenticación y extraer la información de la organización.

**Descripción:** Este tipo de ataque consiste en que el atacante utiliza una consulta segura para unirla a una consulta maliciosa mediante el operador ünion".

**Ejemplo:**

```
Select * from user where user='steven' union select * from admin where id='
    4567'--'pass='1=1';
```

En el ejemplo anterior se observa que la consulta legítima se una consulta maliciosa, vulnerando así al sistema dado que la primera consulta es verdadera y como la consulta maliciosa está unida con el operador ünion.[es]ta también se ejecuta.

**d. Procedimientos almacenados**

**Objetivo del atacante:** Escalar privilegios, ejecutar comandos de forma remota, DoS.

**Descripción:** Este tipo consiste en la utilización de procedimientos almacenados, que se encargan de realizar acciones maliciosas para el beneficio del atacante.

**Ejemplo:** El atacante puede ejecutar comandos sobre los procedimientos almacenados de una base de datos como:

```
DROPTABLE y SHUTDOWN
```

**e. Consultas al backend**

**Objetivo del atacante:** El objetivo del atacante es la extracción y modificación de la información de la base de datos y DoS.

**Descripción:** En el backend, la consulta maliciosa creada por el atacante se añade a una consulta legítima. Entonces, cuando se ejecuta la primera consulta, también se ejecutara la segunda consulta, que en este caso es maliciosa.

**Ejemplo:**

```
Select * from user where name= 'steven' and pass="5656';drop table user;
```

Como podemos observar en el ejemplo anterior, la consulta legítima muestra al usuario con el nombre 'steven'y la consulta maliciosa borra toda la tabla de usuarios de la organización.

**f. Codificaciones alternativas**

**Objetivo del atacante:** Consiste en evitar la detección del atacante en los sistemas de las bases de datos.

**Descripción:** Algunos sistemas de bases de datos poseen filtros que pueden detectar caracteres como %, –, etc. como caracteres no válidos. Por lo tanto, el atacante para no ser detectado codifica estos caracteres mencionados en ASCII o Unicode.

**Ejemplo:**

```
SELECT salary FROM users
WHERE login=" AND pin:
exec (char(0x73687574646j776e));
```

Como se observa en el ejemplo anterior, el atacante codifica los caracteres no válidos en la base de datos a ASCII o Unicode para poder utilizar estos caracteres que vulneran a las bases de datos.

### g. Ataques de Inferencia

**Objetivo del atacante:** El atacante busca la extracción de información, descubrir el esquema que utiliza la base de datos e identificar patrones que pueden ser inyectables.

**Descripción:** Este tipo de ataque consiste en obtener conclusiones lógicas en función de preguntas que se le hace al motor de base de datos. El tipo de preguntas es de Verdadero o Falso

### h. Inyección a ciegas

**Objetivo del atacante:** El objetivo principal del atacante es obtener el esquema de la base de datos.

**Descripción:** El atacante realiza preguntas del tipo Verdadero o Falso a la base de datos con el objetivo de obtener el esquema con el que está trabajando la base de datos.

**Ejemplo:** Un ejemplo claro de esto es que los atacantes realizan consultas para descubrir las vulnerabilidades y comprobar si existe una validación de la entrada de datos que ingresaron.

```
Select * from user where id=12" y pass="1=0';
```

### i. Ataques de tiempo

**Objetivo del atacante:** El objetivo del atacante es la obtención de la información de la base de datos.

**Descripción:** El atacante busca recolectar información sobre la base de datos mediante la observación del tiempo de respuesta involucrado en responder a las preguntas consultadas por el atacante.

**Ejemplo:** Los atacantes utilizan la inserción de ciertas palabras claves como ẅaitforp̈ara retrasar la ejecución de una consulta, si esta es verdadera. Dándoles información sobre el comportamiento de la base de datos.

## 2.3 MINERÍA DE DATOS

La minería de datos es el proceso relacionado con el descubrir ciertos patrones interesantes, modelos y otros diversos tipos de conocimiento en grandes volúmenes de datos. La identificación de estos patrones y modelos ayudan a resolver problemas que existen en la industria utilizando el análisis de datos. Para algunas personas, la minería de datos es un sinónimo de un término comúnmente usado, el descubrimiento de conocimiento a partir de los datos (KDD). Pero esta opinión está dividida. dado que para otras personas la minería de datos es un simple paso para descubrimiento de conocimiento a partir de los datos [13]. El proceso general relacionado con el descubrimiento de conocimiento se muestra a continuación como una secuencia iterativa de pasos o fases:

- **Preparación de los datos:** en este paso se empieza con la limpieza de los datos, eliminando el ruido y los datos incoherentes. Luego se integran los datos, en los cuales se combinan múltiples datos de diferentes fuentes. Después se transforman los datos, en los cuales se definen formas apropiadas para tratar los datos en la minería. Finalmente, se seleccionan los datos, es decir, se seleccionan los datos relevantes para su posterior análisis.
- **Proceso de minería de datos:** en este paso se aplican ciertas técnicas o métodos inteligentes para extraer patrones o modelos de los datos proporcionados del anterior paso.
- **Evaluación de patrones o modelos:** este paso consiste en identificar los patrones y modelos realmente relevantes para la representación del conocimiento, basándose en las áreas de interés en las que se está enfocando la minería de datos.
- **Presentación del conocimiento:** en este paso se utilizan ciertas técnicas para la presentación y visualización del conocimiento obtenido del proceso de minería de datos que ayuda a solucionar problemas en el sector industrial.

### 2.3.1 Técnicas de minería de datos

Las técnicas de minería de datos son las encargadas de identificar patrones, tendencias, modelos de los datos con el objetivo de encontrar información útil para decidir o predecir

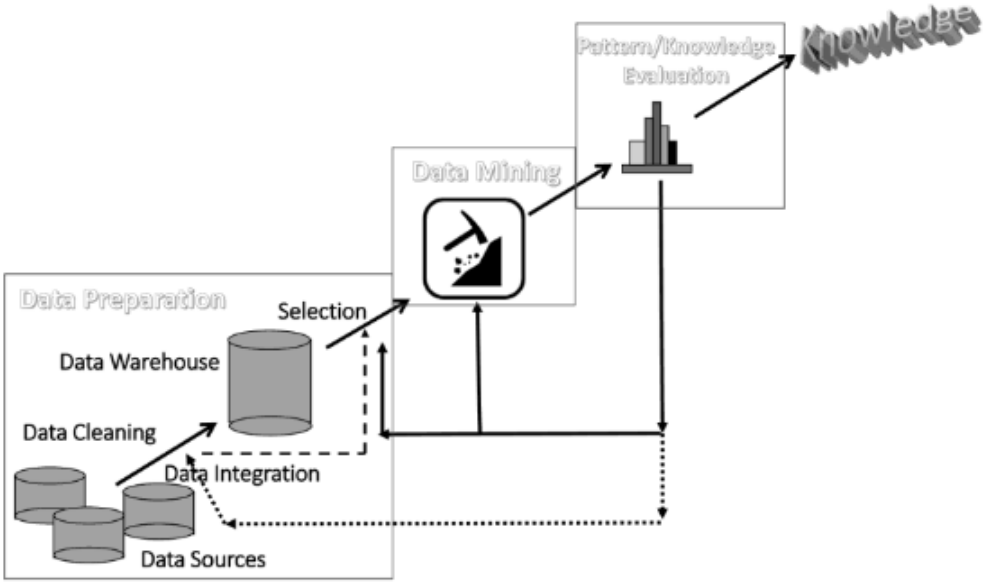


**Figura 2.2:** Esquema general del proceso de minería de datos [13].

cierto comportamiento de los datos. En la época actual se han desarrollado varias técnicas de minería de datos, entre las cuales destacan la asociación, clustering, clasificación, árboles de decisión y redes neuronales. Cada una de las técnicas mencionadas posee sus propias reglas o métodos que determinan como van a resolver el problema. A continuación se describe en que consisten cada una de estas técnicas de minería de datos [14].

#### 2.3.1.1 Técnicas de asociación

Las técnicas de asociación son las más conocidas en la minería de datos, consisten en descubrir ciertos patrones basándose en la relación que existe entre las variables de una misma operación. Además, a esta técnica también se la conoce como una técnica de relación, dado que utiliza la relación entre elementos y a su vez descubre la frecuencia con la que aparecen estos elementos dentro del conjunto de datos. Las reglas involucradas en la asociación son el uso de sentencias if-then para evidenciar la probabilidad de las relaciones existentes entre los datos o variables, dentro de un conjunto muy grande de datos [14].

#### 2.3.1.2 Técnicas de clasificación

Las técnicas de clasificación consisten en la clasificación de una colección de datos en diferentes clases o grupos, con la finalidad de obtener una predicción en un enorme conjunto

de datos. La clasificación puede ayudar a tener una ligera idea de la categoría en la que se encuentra un cliente, elemento u objeto en un conjunto de datos, todo esto mediante la descripción de varios atributos para identificar una clase en concreto [14]. Por ejemplo, podemos clasificar fácilmente los edificios en diferentes tipos basándonos en atributos como su estructura, unidad o altura.

#### 2.3.1.3 Técnicas de clustering

Este tipo de técnica consiste en el análisis de uno o varios atributos con el objetivo de identificar los datos que tienen alguna similitud entre sí. El proceso de clustering en algunas veces también se lo conoce como segmentación porque se encarga desegmentar los datos en diversas categorías para identificar un cluster de resultados relacionados entre sí. Por ejemplo, en una biblioteca podemos colocar en un lugar libros con similitudes entre sí y colocarlos una etiqueta que representa todo ese conjunto [14].

#### 2.3.1.4 Técnicas basadas en árboles de decisión

Las técnicas basadas en árboles de decisión podrían aplicarse como parte de los criterios de selección. Además, brindan apoyo en lo que respecta a la utilización y selección de datos específicos dentro de una estructura general de los datos. Esta técnica parte de una pregunta muy simple, la cual tiene dos o más respuestas. Cada una de estas respuestas conduce a otra pregunta adicional con el objetivo de apoyar la clasificación de los datos, en otras palabras podría decirse que los categoriza. En otras ocasiones estos árboles de decisión sé los puede utilizar para hacer predicciones basadas en cada respuesta anterior [14].

#### 2.3.1.5 Técnicas de predicción

Las técnicas de predicción consisten utilizar una combinación de otras técnicas de minería de datos. La predicción se encarga de la clasificación, análisis de tendencias, búsqueda de patrones y su relación. La predicción se la realiza mediante el análisis de eventos que ya pasaron, con el objetivo de predecir que pasara en años posteriores [14].

# 3 METODOLOGÍA

En esta sección, se describe de manera general las metodologías utilizadas para la realización del proyecto en cada una de las fases del proyecto. Se contemplaron las metodologías para: la revisión sistemática de la literatura, el proceso de minería de datos, y el proceso de desarrollo de un prototipo de un sistema software donde se implementará el resultado del proceso de minería de datos.

Dado que este trabajo forma parte de un proyecto integrador, las metodologías utilizadas para el desarrollo de cada componente son comunes entre los cuatro componentes del proyecto y fueron trabajadas de manera grupal.

## 3.1 METODOLOGÍA DE REVISIÓN SISTEMÁTICA DE LA LITERATURA

La metodología por la que se optó para la revisión sistemática de la literatura es la propuesta por el autor Kitchenham [3] dado que es una metodología enfocada al desarrollo de software. La estructura de esta metodología se divide en tres secciones generales: planificación, realización y presentación de informes. A continuación, se detallan cada una de estas secciones con sus respectivas fases.

### 3.1.1 Planificación de la revisión

#### 3.1.1.1 Identificación de la necesidad de una revisión

En esta fase es necesario asegurarse de que la revisión sistemática es necesaria, por lo tanto, las personas a cargo de la investigación deben empezar con la identificación y revi-

sión de cualquier revisión sistemática que existe sobre el tema de interés en función de los criterios de evaluación adecuados. Para comprobar esto, se realiza una lista de comprobación que contiene las siguientes preguntas que ayudarán a identificar la necesidad de la revisión:

- ¿Cuáles son los objetivos de la revisión?
- ¿Cuáles fueron las fuentes utilizadas para identificar los estudios primarios? ¿Existieron restricciones?
- ¿Qué criterios de inclusión y exclusión se definieron y de que manera se aplicaron?
- ¿Cuáles fueron los criterios utilizados para evaluar la calidad de los estudios primarios y cómo se aplicaron?
- ¿De qué manera se extrajeron los datos de los estudios primarios?
- ¿Cómo se sintetizaron los datos? ¿Cómo se investigaron las diferencias entre los estudios? ¿Cómo se combinaron los datos? ¿Era razonable combinar los estudios?
- ¿Las conclusiones se desprenden de las pruebas?

#### 3.1.1.2 Elaboración de un protocolo de revisión

El protocolo de revisión se encarga de especificar cuáles son los métodos a utilizarse al momento de desarrollar una revisión sistemática. Aplicando un protocolo, se disminuyen las probabilidades de generar un sesgo en la investigación. Los siguientes componentes son contemplados al momento de realizar un protocolo de revisión:

- Preguntas de investigación con las que se pretende responder con la revisión
- Estrategias que se utilizarán para buscar estudios primarios, incluyendo términos de búsqueda, recursos en los cuales se realizará la búsqueda, incluyendo bases de datos, revistas científicas, conferencias.
- Criterios y métodos para la selección de estudios. Estos criterios de selección de estudios especifican lo que se excluye o incluye dentro de la revisión.
- Listas de comprobación y procedimientos de evaluación de la calidad de los estudios.

- ❒ Definición de la manera en la que se obtendrá la información necesaria de cada estudio primario.
- ❒ Resumen de la extracción de datos, definición de la manera en la que se realizará el resumen y sus estrategias.

### 3.1.2 Realización de la revisión

Cuando los investigadores han acordado el protocolo a seguir, se puede empezar la revisión, esto implica las siguientes fases:

#### 3.1.2.1 Identificación de la investigación

El objetivo principal de una revisión sistemática es encontrar la mayor cantidad de estudios primarios con la mayor relevancia, siempre y cuando estos aporten a contestar las preguntas de investigación. Esta búsqueda debe realizarse de manera objetiva y libre de sesgos. Por lo tanto, se realizan algunos pasos adicionales comparados con las revisiones tradicionales:

**a. Generar una estrategia de búsqueda**

Una estrategia de búsqueda suele ser iterativa y se benefician de búsquedas preliminares, en las cuales se identifican las revisiones sistemáticas existentes donde se evalúan el volumen de estudios potencialmente relevantes. Un enfoque recomendado es desglosar la pregunta de investigación en fases individuales, por ejemplo, población, intervención, resultados, diseños de estudio. Luego se elabora una lista de sinónimos y abreviaturas. A continuación, construir cadenas de búsqueda sofisticadas utilizando combinaciones booleanas AND y OR.

**b. Sesgo de publicación**

El sesgo de publicación hace referencia a la tendencia que existe de seleccionar artículo que presenten un resultado particular, ya que es el investigador quien asigna un valor de que tan bueno o malo es un artículo. Por lo tanto, el investigador debe informarse sobre la problemática y explorar la literatura, conferencias o contactar con expertos e investigadores que el área de interés que puedan guiarlo en la investigación y no recaer en el sesgo de publicación.

**c. Gestión de la bibliografía y recuperación de documentos**

La gestión de la bibliografía permite gestionar un gran número de referencias que se pueden obtener de una investigación bibliografía exhaustiva. Por lo tanto, es importante tener un sistema para esto, por ejemplo, se pueden utilizar paquetes bibliográficos como Reference Manager o Endnote o simplemente un Excel con toda esta información.

**d. Documentación de la búsqueda**

Todo el proceso del desarrollo de una revisión sistemática debe mantener transparencia y ser reproducible, por lo tanto, la revisión debe: estar documentada con un detalle suficiente para su reproducción, esto incluye anotar los cambios que se realicen durante la búsqueda, así como la justificación pertinente.

#### 3.1.2.2 Selección de estudios primarios

Cuando se han conseguido los artículos principales de la búsqueda, se procede a evaluar la relevancia real de estos.

**a. Criterios para la selección de estudios**

Los criterios de selección de los estudios tienen por objetivo el de identificar los estudios primarios que aportan pruebas directas a la pregunta de investigación. Los criterios de inclusión y exclusión deben tomar como referencia la pregunta de investigación. Estos criterios deben probarse para garantizar que su interpretación es fiable y que los estudios están clasificados correctamente.

**b. Proceso de selección de estudios**

La selección de estudios es un proceso de varias etapas donde se empieza con los criterios de selección que los debe interpretar el investigador, a menos que los estudios puedan excluirse porque las copias obtenidas no están completas. Por lo tanto, una vez obtenidos los artículos a ser analizados, se realiza el proceso de aplicación de los criterios de inclusión y exclusión, y mantener una lista de estudios excluidos en los cuales se debe identificar el motivo de la exclusión.

#### 3.1.2.3 Evaluación de la calidad del estudio

Además de los criterios de inclusión y exclusión, es necesario considerar la evaluación de la calidad de los estudios primarios:

- ❒ Proporcionar criterios de inclusión/exclusión aún más detallados.
- ❒ Investigar si las diferencias de calidad explican las diferencias en los resultados de los estudios.
- ❒ Como medio para ponderar la importancia de los estudios individuales cuando se sintetizan los resultados.
- ❒ Orientar la interpretación de los resultados y determinar la fuerza de las inferencias.
- ❒ Orientar las recomendaciones para futuras investigaciones.

#### 3.1.2.4 Extracción y seguimiento de datos

En esta fase se diseñan los formularios de extracción de datos, los cuales cumplen la función de registrar con precisión la información que los investigadores han obtenido de los estudios primarios. Con el objetivo de reducir el sesgo, los formularios de extracción de datos deben definirse y probarse cuando se defina el protocolo del estudio.

**a. Diseño de formularios de extracción de datos**

Estos formularios de datos deben tener un diseño que permita la recolección de información que se necesite para abordar las preguntas de la revisión y los criterios de calidad del estudio. En la mayoría de los casos, la extracción de datos se definirá como un conjunto de valores numéricos que deben extraerse para cada estudio. Una recomendación importante es utilizar formularios electrónicos, ya que facilitan el análisis posterior.

**b. Contenido de los formularios para la recolección de datos**

Los formularios ayudan a complementar las preguntas de investigación. Dentro de estos formularios se debe proporcionar información relevante que incluya lo siguiente:

- ❒ Nombre de la revisión

- ❒ Fecha de extracción de datos
- ❒ Título, autores, revista, detalles de publicación
- ❒ Espacio para notas adicionales

**c. Procedimientos para la extracción de datos**

Para el procedimiento de extracción de datos de los estudios primarios es importante realizarlo de manera independiente por dos o más investigadores. Luego, estos datos extraídos deben compararse y solucionar los desacuerdos que podrían presentarse mediante un consenso entre los investigadores. Es recomendable utilizar un formulario aparte para marcar y corregir los errores o desacuerdos presentados.

**d. Múltiples publicaciones de los mismos datos**

Es importante tener en cuenta evitar la inclusión de múltiples publicaciones que contengan los mismos datos en la revisión sistemática, debido a que estos informes duplicados podrían sesgar gravemente cualquier resultado obtenido de la investigación. En caso de que existieren publicaciones duplicadas, es recomendable utilizar la publicación más reciente para la revisión sistemática.

**e. Datos no publicados, datos que faltan y datos que requieren manipulación**

Si existiera el caso de que se dispone de información de estudios que se están desarrollando o están en curso, debe incluirse siempre y cuando sea posible información de calidad sobre el estudio. Los informes no siempre presentan todos los datos relevantes, también pueden estar mal redactados y ser ambiguos, por lo tanto, es necesario contactar a los autores para conseguir la información necesaria. En ciertas ocasiones los estudios primarios no proporcionan todos los datos, pero en algunas situaciones se pueden recrear estos datos necesarios a partir de la manipulación de los datos publicados. Por lo tanto, si se diera el caso de manipulación de datos, es importante someterlos a un análisis de sensibilidad para su posterior uso.

#### 3.1.2.5 Síntesis de datos

La síntesis de datos consiste en cotejar y resumir los resultados obtenidos de los estudios primarios. La síntesis que se realiza sobre los resultados puede ser descriptiva (no cuantitativa). En algunos casos también es posible complementar el análisis cualitativo con una síntesis cuantitativa. El uso de técnicas que utilizan la estadística para su desarrollo se las

denomina meta-analisis.

**a. Síntesis descriptiva**

La información extraída de los estudios, como la población, el contexto, el tamaño de la muestra, los resultados y la calidad del estudio, debe ser tabulada de una forma coherente, siempre teniendo presente la pregunta de la revisión. Estas tablas deben tener una estructura tal que permita evidenciar la relación que existe entre los estudios analizados.

**b. Sensibilidad del análisis**

La realización de un análisis de sensibilidad es importante cuando se realiza un meta-analisis. El meta-analisis se utiliza para proporcionar una estimación global del efecto de tratamiento y su variabilidad en el estudio. En estos casos lo ideal es la repetición de varios subconjuntos de estudios primarios con el objetivo de determinar si estos resultados son robustos o no. Los tipos de subconjuntos pueden ser:

- ❒ Solo estudios primarios de alta calidad
- ❒ Estudios primarios de tipos particulares
- ❒ Estudios primarios para los que la extracción de datos no presento dificultades

### 3.1.3 Presentación de informes

Esta fase es una de las más importantes dado que permite comunicar de forma eficaz los resultados de la revisión realizada. Generalmente, estas revisiones son presentadas de dos maneras:

- ❒ Mediante informes técnicos dentro de documentos académicos como tesis
- ❒ En revistas o conferencias especializadas

## 3.2 METODOLOGÍA DE MINERÍA DE DATOS CRISP-DM

CRISP-DM (Cross Industry Process for Data Mining) [4], es una metodología de minería de datos que incluye descripciones de las etapas que deben seguirse dentro un proyecto, así como las tareas requeridas en cada una de estas etapas. CRISP-DM también se puede

considerar como un modelo de proceso de minería de datos que ayuda a los expertos en la materia a resolver un problema.

CRISP-DM, está estructurado en seis etapas, algunas de las cuales son bidireccionales, es decir, cada una puede retroceder para hacer revisiones y correcciones, esto implica que los segmentos de las etapas no necesariamente están dispuestos en el orden que se muestra en la Figura 3.1.

El modelo de CRISP-DM es flexible y se puede configurar fácilmente de modo que se adapta a las actividades de una organización, generando soluciones que brinden el mayor valor posible para solventar sus necesidades. Permitiendo crear un modelo de minería de datos que se adapte a sus necesidades concretas [4].

En la Figura 3.1 se muestran las fases que constan en la metodología y que son detalladas a continuación:

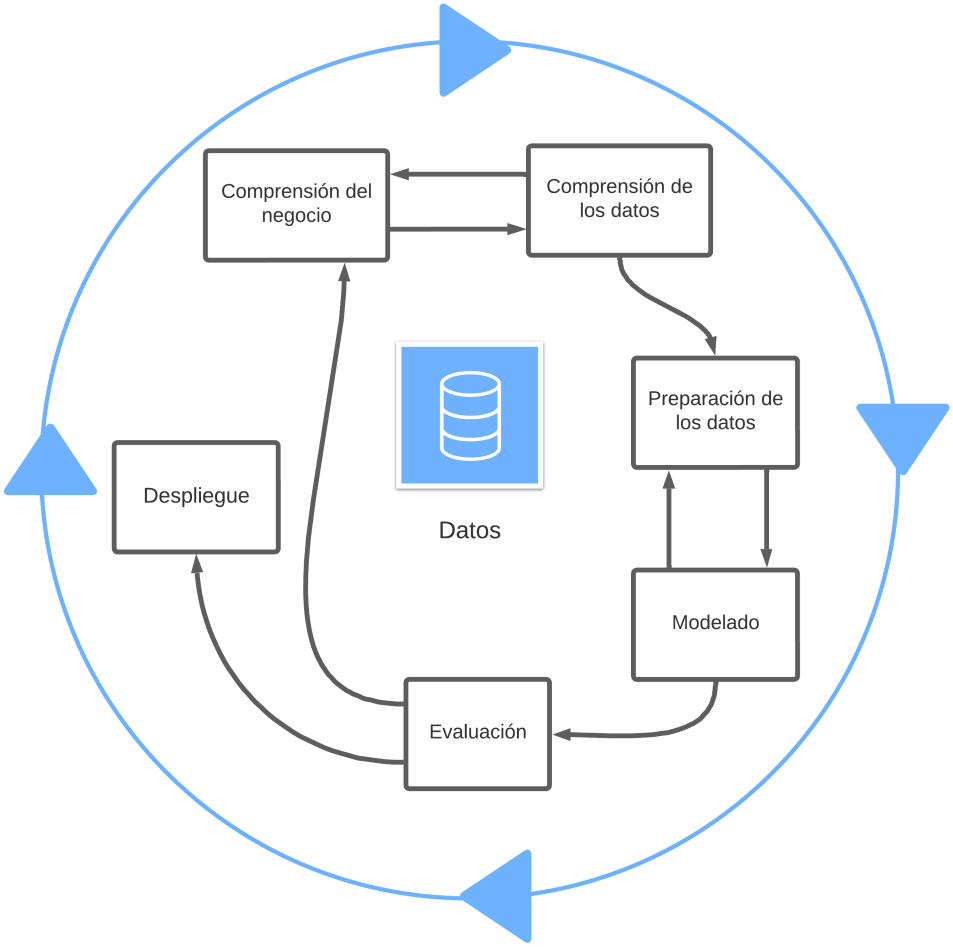


**Figura 3.1:** Fases de la metodología CRISP-DM. Adaptado de [4].

1. **Comprensión del negocio**

   En la primera fase, que es quizás la más importante, se analiza a la empresa desde una perspectiva comercial, permitiendo traducir sus necesidades y objetivos de negocio hacia requerimientos más técnicos [4]. Es decir, que si no se llega a la comprensión de los objetivos del negocio, ningún algoritmo por complejo que sea podrá lograr resultados confiables. Para la extracción de datos de manera más efectiva, es indispensable tener una buena compresión del problema que desea resolver, lo que le permitirá recopilar los datos necesarios y poder interpretar los resultados correctamente. Al final de esta etapa se obtiene un plan de proyecto donde se describen los objetivos de la empresa como un proyecto de minería de datos. Las tareas que se llevan a cabo durante esta etapa son:

   - ❒ **Determinar los objetivos de negocio**

     Esta es la primera tarea desarrollada y su objetivo es identificar el problema que necesita ser resuelto, por lo cual se debe hace la siguiente pregunta ¿Por qué usar Data Mining?, y la realidad es que en la época actual, existen muchos problemas que los datos pueden brindar información valiosa, y a partir de la minería de datos, obtener conocimiento para tomar decisiones oportunas. Un ejemplo de esto podría ser la ubicación estratégica de productos dentro de un supermercado según los hábitos de compra de un usuario. Para ello es posible utilizar datos obtenidos de facturar, analizar dichos datos, y obtener patrones que determinen que productos son comprados generalmente juntos y ubicarlos de mejor manera en las estanterías [4]. Como objetivo de la empresa debe determinar los criterios para decidir que la minería de datos se implementó correctamente o no. En el caso anterior podría ser el aumento en las ventas de un producto en particular.

   - ❒ **Evaluación de la situación**

     Es importante matizar el estado de la situación antes de proceder a realizar el proceso de minería de datos, teniendo en cuenta aspectos como: ¿Qué conocimiento tiene disponible sobre la materia?, ¿Se requiere conteo de datos?, ¿Resulta rentable realizar minería de datos?. En esta fase se identifican requerimientos de problemas, tanto de negocio como de minería de datos. El propósito de esta tarea es analizar la mayor cantidad de aspectos posibles que se deben tomar en cuenta antes de proceder a realizar la minería de datos [4]. Estos aspectos incluyen, pero no se limitan a, personal, datos, riesgos, etc.

- ❒ **Realizar el plan del proyecto** Al final de la primera etapa de CRISP-DM, es necesario desarrollar un plan de proyecto donde se detallen como se procederá con el proyecto, así como las técnicas que se utilizarán en cada paso.

2. **Compresión de los datos**

   Esta etapa inicia con la obtención de los datos que serán utilizados y sigue con tareas relacionadas con la comprensión de dichos datos, tales como identificación de anomalías, análisis de calidad, identificación de atributos, generación de hipótesis, etc.

   La comprensión de datos se encuentra fuertemente relacionada con la comprensión del negocio, ya que es indispensable comprender los datos disponibles para continuar con la ejecución del plan elaborado [4].

   - ❒ **Recolectar los datos iniciales**

     En esta tarea, los datos más importantes son recopilados en su totalidad para su procesamiento futuro. Al final de esta fase, se prepara un documento donde se detallan aspectos, los aspectos más relevantes sobre los datos, incluyendo las técnicas utilizadas para su recolección y los problemas presentados [4].

   - ❒ **Descripción de los datos** Después de haber obtenido los datos primarios, se deben describir. Este proceso incluye contabilizar el volumen de datos (recuento de datos y atributos). Asimismo, se debe brindar una explicación sobre el significado de cada atributo [4].

   - ❒ **Exploración de los datos** Esta tarea abarca la descripción estadística de los atributos de los datos. En esta descripción se obtienen tablas, gráficas, distribuciones de datos, etc. Una vez hecha la descripción de los datos, se procede a explorarlos, el propósito de esto es encontrar una estructura general para los datos. Implica aplicar pruebas estadísticas básicas para revelar las propiedades de los datos recién adquiridos, generar tablas de frecuencia y construir gráficos de distribución. Como resultado de esta tarea se obtiene un documento donde se describe un de análisis de los datos [4].

   - ❒ **Verificar la calidad de los datos** En esta tarea, se realizan pruebas en los datos para determinar la consistencia de los valores de campo individuales, el número y la distribución de ceros, y para encontrar valores fuera de rango que puedan convertirse en ruido para el proceso. La idea de cuando se llega a este punto es poder garantizar la integridad y exactitud de los datos [4].

3. **Preparación de los datos**

   En esta etapa se contemplan las actividades relacionadas con la construcción de un conjunto de datos que pueda ser analizado por herramientas especializadas para minería de datos. Esta fase abarca aspectos como la sección, procesamiento, análisis gramatical, limpieza, construcción de nuevos datos, integración, y el formato de los datos obtenidos. Esta tarea se realiza de manera iterativa, ya que es muy probable que se deba revisar varias veces los datos antes de obtener un conjunto adecuado para proceder con la fase de modelado [4].

4. **Modelado**

   En esta etapa se genera un modelo, el cual tenga la capacidad de brindar información útil para alcanzar los objetivos propuestos. En esta fase se deberá:

   - ❒ Determinar una técnica de modelado apropiada para los datos obtenidos y los objetivos planteados
   - ❒ Definir métricas para la evaluación de desempeño del modelo generado.
   - ❒ Crear un modelo utilizando las técnicas previamente definidas sobre los datos.
   - ❒ Adecuar el modelo generado a partir de los resultados obtenidos de sus métricas y su efecto en los objetivos del negocio.

5. **Evaluación** En esta etapa se centra realizar una evaluación del modelo, analizando que tan cerca se encuentra de alcanzar los objetivos de negocio antes establecidos.

   En esta fase contempla:

   - ❒ Realizar una evaluación de modelo o modelos generados
   - ❒ Realizar una retrospectiva del proceso de minería de datos que se ha realizado durante todo el tiempo.
   - ❒ Establecer los siguientes pasos a seguir. Como el proceso de CRISP-DM es iterativo, esto puede implicar regresar a fases anteriores si el modelo no se ajusta a la realidad del negocio, o continuar hacia el despliegue si se cumplen las expectativas del modelo.

6. **Despliegue o implementación** Esta fase se centra en implementar los resultados obtenidos en un ambiente real, de manera que pueda ser de utilidad en la toma de decisiones en la organización. En esta fase se definen varias tareas relativas al mantenimiento e implementación del modelo. Estas taras son:

- ❒ Diseñar un plan de despliegue de modelos
- ❒ Realizar la monitorización y mantenimiento
- ❒ Producir el informe final
- ❒ Revisar el proyecto en su totalidad

## 3.3 METODOLOGÍA DE DESARROLLO DE SOFTWARE XP

La Programación Extrema (Extreme Programing o XP) es una metodología ágil muy utilizada dentro del desarrollo de software. Esta metodología define cuatro actividades o fases principales: planeación, diseño, codificación y pruebas; así como las tareas y prácticas sugeridas para ejecutar en cada una de estas. Estas fases se muestran en la Figura 3.2.

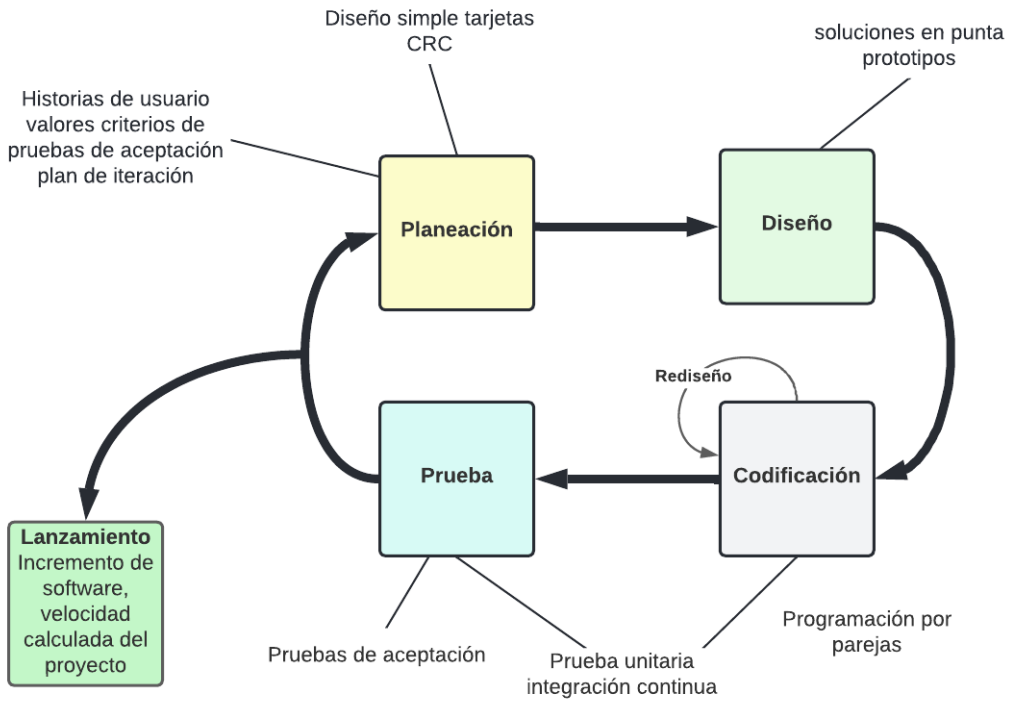


**Figura 3.2:** Fases de la programación extrema [5].

### 3.3.1 Planeación

La planeación también es denominada juego de planeación, inicia con las actividades para la elicitación de requisitos, facilitando a que el equipo de desarrollo comprendan la problemática del negocio y se puede dar solución a la misma, mediante las características y

funcionalidades principales del sistema software. En esta fase se elaboran las historias de usuario que describen el valor de los requerimientos obtenidos, así como las funcionalidades del software a desarrollarse [5].

Las historias de usuario deben contener una prioridad la cual se asignará respecto al valor que asigne el cliente a la característica o función. Cada una de las historias de usuario son evaluadas y se les asigna un costo, el cual es medido en semanas de desarrollo, es importante mencionar que las historias de usuario puede ser modificadas o escribir más historias de usuario. Una vez escritas las historias de usuario se las debe agrupar y seleccionar el orden desarrollo, posteriormente se establece una fecha de entrega y también se debe incluir otros aspectos o detalles del proyecto. Una vez que se llega a un acuerdo, se establece la fecha de entrega, tomando en cuenta algunos factores fundamentales como el hecho de que todas las historias se implementan de forma inmediata, es decir, en pocas semanas y las historias con más valor serán implementadas primero [5].

Una vez se ha realizado la primera entrega, el equipo XP debe calcular cuantas historias pudieron ser desarrolladas en esta entrega o incremento. Esta velocidad ayudará a realizar una mejor planificación y gestión de actividades y fechas de entrega durante el resto de desarrollo. Es importante mencionar que a medida que el proyecto avanza, el cliente puede añadir, modificar el valor de una historia existente o eliminar historias si es necesario. Si esto ocurre, el equipo de desarrollo debe estimar un nuevo tiempo de entrega para las historias faltantes o modificadas [5].

### 3.3.2 Diseño

El diseño XP se basa en en el principio de mantener un diseño sencillo. Mantener el diseño sencillo aporta más valor que un diseño complejo. Además, se debe considerar que el diseño guía la implementación de una historia de usuario y se debe no se debe considerar funcionalidades adicionales asumidas por el desarrollador [5]. En la metodología XP se promueve el uso de tarjetas CRC (Clase-Responsabilidad-Colaborador), como una herramienta para mantener el enfoque de orientación a objetos. Las tarjetas CRC deben contener información sobre la clase, responsabilidad y colaborador que permiten identificar y organizar las posibles clases y métodos que puedan ser implementadas en el software [5].

Es recomendable la utilización de un prototipo que permita entender de mejor manera el

diseño. Este prototipo es evaluado teniendo como objetivo disminuir el riesgo en al momento de implementar el sistema real, a la vez que valida las estimaciones en la historia donde se complica entender el diseño. Es decir, tiene un rediseño, lo que involucra cambiar un sistema software de tal forma que no se modifique el comportamiento externo del código, pero se optimice la estructura interna. Realizando el rediseño se reducen las probabilidades de que se introduzcan defectos en el código [5].

### 3.3.3 Codificación

Cuando las historias de usuario han sido desarrolladas y se ha realizado un diseño previo, lo primero que se debe realizar antes de codificar, es diseñar pruebas unitarias, y a partir de estas, generar el código necesario para su funcionamiento. Cada prueba unitaria tiene como objetivo guiar al desarrollador a enfocarse en realizar solo lo necesario para pasar la prueba unitaria y no añadir ninguna funcionalidad extra y manteniendo el principio MS. Las pruebas unitarias brindan retroalimentación inmediata a los desarrolladores, agilizando el tiempo de codificación [5].

Durante la codificación se puede utilizar la programación es parejas, que es recomendada por esta metodología, ya que se tiene la premisa de que si dos personas desarrollan las historias en conjunto, se agiliza este proceso debido a que se obtiene retroalimentación instantánea, se solucionan rápidamente los inconvenientes encontrados, además de que se aportan ideas para optimizar el código desarrollado. Finalmente, el código desarrollado se integra con el trabajo de equipo de desarrollo, utilizando la estrategia de “integración continua”para evitar los inconvenientes relacionados con la compatibilidad y descubriendo errores en etapas iniciales de desarrollo [5].

### 3.3.4 Pruebas

Dentro del enfoque XP, la realización de pruebas es un aspecto clave, ya que permiten verificar y validar el software de manera eficiente. Las pruebas deben tratar de automatizarse en la mayor medida de lo posible, de modo que puedan características funcionales generales del sistema [5].

# 4 REVISIÓN SISTEMÁTICA DE LA LITERATURA DE TÉCNICAS DE DETECCIÓN Y PREDICCIÓN DE SQLIA

Para la realización de este componente, un aspecto fundamental residió en la determinación del estado del arte en cuanto a las técnicas para la detección y predicción de SQLIA. Para ello, se realizó una revisión de literatura con respecto a este tema. A partir de los resultados obtenidos de esta revisión se procedió con el desarrollo del componente.

La revisión de literatura fue trabajada de manera conjunta entre todos los miembros del proyecto integrador. Esto fue debido a que como resultado de esta revisión, se obtuvieron los algoritmos que fueron evaluados individualmente en cada componente.

## 4.1 METODOLOGÍA

Para la realización de este estudio, se utilizó la metodología propuesta por Kitchenham en [15], cuyos pasos se detalla la sección 3.1.2. En este artículo, se describe en la sección una serie de pasos a seguir para llevar a cabo una revisión sistemática. Estos pasos se describen a continuación.

**a. Preguntas de investigación**

Las preguntas de investigación se realizaron con el objetivo de determinar las técnicas que existen actualmente para la detección y predicción de ataques de inyección SQL.

De esta manera se obtuvieron las siguientes preguntas de investigación:

RQ1 ¿Cuáles son las técnicas que se están utilizando para la detección y predicción de SQLIA?

RQ2 ¿Cuáles son las técnicas más utilizadas para detección y predicción de SQLIA?

RQ3 ¿Es posible clasificar las técnicas para la detección y predicción de SQLIA?

Mediante las preguntas RQ1 y RQ2 se realizó un análisis de publicaciones que proponen nuevas técnicas para la detección y predicción de SQLIA.

Para contestar la pregunta RQ3 se examinó los resultados obtenidos en las preguntas anteriores para proponer una clasificación de las técnicas identificadas.

**b. Proceso de búsqueda**

Para realizar la búsqueda se utilizó la siguiente cadena de búsqueda en base a las preguntas de investigación planteadas:

```
(detection OR prediction) AND (SQLIA OR (SQL AND injection)) AND NOT (survey
    OR review)
```

Esta cadena fue adaptada de manera que cumpliera con las especificaciones de búsqueda de cada librería o base de datos utilizada. Sin embargo, el esquema general de los términos y conectores utilizados se mantuvo fijo en cada búsqueda.

**c. Fuentes y bases de datos para la búsqueda**

Para la búsqueda, se utilizaron las bases de datos más conocidas dentro del área de Ciencias de la Computación. Las librerías digitales utilizadas fueron:

- ACM Digital Library
- IEEE Xplore
- ScienceDirect
- SpringerLink

Posteriormente, se realizó una búsqueda más exhaustiva en el motor de búsqueda bibliográfica Google Scholar. Esta búsqueda se la realizó con el fin de obtener artículos que no hayan sido publicados en las librerías digitales consideradas inicialmente.

**d. Criterios de inclusión y exclusión**

Para la inclusión de un artículo se tomaron en cuenta los siguientes criterios de inclusión:

- Artículos que se hayan publicado entre el 1 de enero de 2012 y el 13 de abril de 2022

- ❒ Artículos cuyos títulos cumplieran la cadena de búsqueda considerada para la búsqueda
- ❒ Artículos publicados en conferencias o revistas especializadas

Una vez determinados los artículos que cumplieron con los criterios de inclusión, se utilizaron los siguientes criterios de exclusión para la selección de artículos:

- ❒ Artículos que traten sobre la detección de ataques de inyección SQL en tecnologías o entornos específicos.
- ❒ Artículos que traten sobre la detección de otros ataques además de los ataques de inyección SQL
- ❒ Artículos que no propongan de técnicas específicas para la detección de ataques de inyección SQL. Por ejemplo, revisiones sistemáticas de la literatura o artículos científicos que presenten comparativas entre técnicas existentes.
- ❒ Artículos que no tengan DOI (Digital Object Identifier)
- ❒ Artículos que tengan menos 5 páginas de contenido sin tomar en cuenta la sección de referencias bibliográficas
- ❒ Artículos escritos en un idioma distinto al inglés

La aplicación de los criterios de exclusión se la realizó de manera manual realizando un escaneo sobre los artículos obtenidos

**e. Selección de artículos** La selección de artículos se la realizó en seis fases de búsqueda en las cuales se fueron aplicando los distintos criterios de inclusión y exclusión hasta llegar a los artículos que fueron seleccionados para formar parte de la revisión sistemática.

En la primera fase de búsqueda, se seleccionaron todos los resultados que incluyeran la cadena de búsqueda en cualquier lugar del artículo, ya sea en el título, resumen, contenido, metadatos, etc. En esta primera búsqueda se obtuvieron un total de 4048 resultados.

En la segunda fase de búsqueda se filtraron solo los resultados comprendidos entre el 1 de enero de 2012 y el 13 de abril de 2022. En esta búsqueda, los resultados disminuyeron a 2957 resultados.

En la tercera fase de búsqueda se seleccionaron los resultados en donde la cadena de búsqueda se encontrase solo en el título, así, se redujo el resultado de la búsqueda a 233 resultados.

En la cuarta fase de búsqueda se descartaron los resultados que no fuesen propiamente artículos científicos, por ejemplo, aquellos artículos que hayan sido publicados en revistas indexadas o conferencias especializadas. En esta fase se obtuvieron 198 artículos.

En la quinta fase de búsqueda, se procedió a filtrar los artículos duplicados. En este filtrado se logró obtener un total de 156 artículos.

En la sexta fase de búsqueda, fueron tomados los 156 artículos obtenidos hasta la quinta fase de búsqueda y se realizó un análisis manual de cada artículo, aplicando los criterios de exclusión definidos anteriormente. De esta búsqueda se obtuvieron finalmente 51 artículos que fueron seleccionados para la revisión.

En la Figura 4.1, se muestra de manera resumida el proceso de filtrado de los artículos mediante las diversas búsquedas en las que se fueron aplicando los criterios de inclusión y exclusión especificados anteriormente.

**Tabla 4.1:** Resumen del proceso de búsqueda y selección de artículos. Fuente: El autor.

| Fase de búsqueda | Artículos por fuente | Total |
|---|---|---|
| Primera Fase | Science Direct: 213<br>SpringerLink: 973<br>IEEE Xplore: 274<br>ACM Digital Library: 1478<br>Google Scholar:1110 | 4048 |
| Segunda Fase | Science Direct: 403<br>SpringerLink: 745<br>IEEE Xplore: 212<br>ACM Digital Library: 831<br>Google Scholar | 2957 |
| Tercera Fase | Science Direct: 213<br>SpringerLink: 973<br>IEEE Xplore: 274<br>ACM Digital Library: 1478<br>Google Scholar:766 | 233 |

| Fase de búsqueda | Artículos por fuente | Total |
|---|---|---|
| Cuarta Fase | Science Direct: 8<br>SpringerLink: 12<br>IEEE Xplore: 53<br>ACM Digital Library: 28<br>Google Scholar: 132 | 198 |
| Quinta Fase | Los artículos ya no se clasificaron según su fuente | 156 |
| Sexta Fase | Los artículos ya no se clasificaron según su fuente | 51 |

**f. Evaluación de calidad**

Para la evaluación de calidad se utilizaron los criterios propuestos en [16]. Así, se definieron las preguntas descritas a continuación:

QA1 ¿El artículo describe los objetivos de investigación de manera clara?

QA2 ¿El artículo describe una revisión de literatura, antecedentes y contexto de investigación?

QA3 ¿El artículo muestra trabajos relacionados de trabajos anteriores para mostrar la principal contribución de la investigación?

QA4 ¿El artículo describe la arquitectura propuesta o la metodología usada?

QA5 ¿El artículo tiene resultados de la investigación?

QA6 ¿El artículo muestra conclusiones que son relevantes al propósito/problema de investigación?

QA7 ¿El artículo recomienda trabajo o mejoras a realizar para el futuro?

Los puntajes para cada pregunta varían entre 0,0.5 y 1 de acuerdo con lo indicado en el Anexo B

**g. Extracción de datos y resultados**

Los datos extraídos de cada estudio fueron:

- ❒ Información bibliográfica (título, año de publicación, conferencia o revista, autores)
- ❒ Número de citas en el Google Scholar

- ❒ Nombre o descripción de la técnica utilizada para la detección de SQLIA
- ❒ Clasificación a la que pertenece la técnica utilizada para la detección de SQLIA
- ❒ Algoritmo utilizado para la detección de SQLIA (si aplica)
- ❒ Tamaño y fuente del conjunto de datos utilizado para la aplicación de la técnica (si aplica)
- ❒ Tipos de SQLIA que abarca la técnica descrita

## 4.2 RESULTADOS

**a. Resultados de búsqueda**

Los 51 artículos seleccionados se describen en la el Anexo A. Por cada artículo se muestran: las citas obtenidas en Google Scholar, el año de publicación, el nombre de la técnica utilizada para la detección o predicción de SQLIA, la clasificación a la que pertenece la técnica utilizada, el o los algoritmos utilizados (si aplica), el tamaño y fuente del conjunto de datos utilizado para la evaluación de la técnica propuesta (si aplica), los tipos de SQLIA que abarca la técnica descrita. Los artículos fueron ordenados por el número de citas que obtuvieron en Google Scholar. Para los artículos que no cumplan alguno de los criterios se colocarán las iniciales “N/A”, indicando que no aplica el criterio en dicho artículo.

Basándose en la Figura 4.1 se observa que la técnica de Machine Learning es utilizada en 19 artículos para la detección y predicción de SQLIA, y las técnicas menos usadas son: Sistema de detección de intrusión, técnicas híbridas y otras técnicas. Con base en la lectura de los artículos se determinó una clasificación para las técnicas de detección y predicción de SQLIA. Esta clasificación se puede apreciar en la columna “Clasificación”, del Anexo A y se explica a mayor profundidad en el apartado **c.** de la Sección 4.3.

**b. Resultados de la evaluación de calidad**

En el Anexo C, se muestran los puntajes de la evaluación de calidad de los artículos analizados

Para esta revisión se determinó que todos los artículos con una calificación mayor a 5 puntos de 7 posibles, son considerados como artículos de alta calidad.

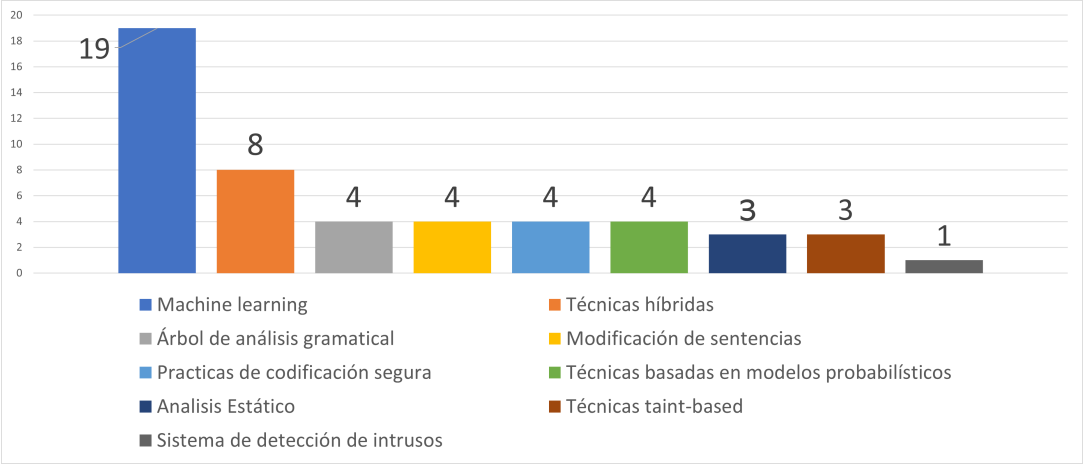


**Figura 4.1:** Distribución de clasificación de las técnicas de detección de SQLIA en los artículos analizados. Fuente: Los Autores

Como se puede observar en el Anexo C, el promedio de la evaluación de calidad es aproximadamente de 5.75, por lo que se puede determinar que de manera general, los artículos poseen una buena calidad.

En el Anexo C, se observa que a pesar de mantener una calidad relativamente alta, existen 4 artículos que muestran una baja calificación. Por contra parte, 12 artículos alcanzaron una calificación perfecta, lo que resulta en un buen indicador en cuanto a la calidad de los estudios realizados para proponer nuevas técnicas para la detección y predicción de SQLIA.

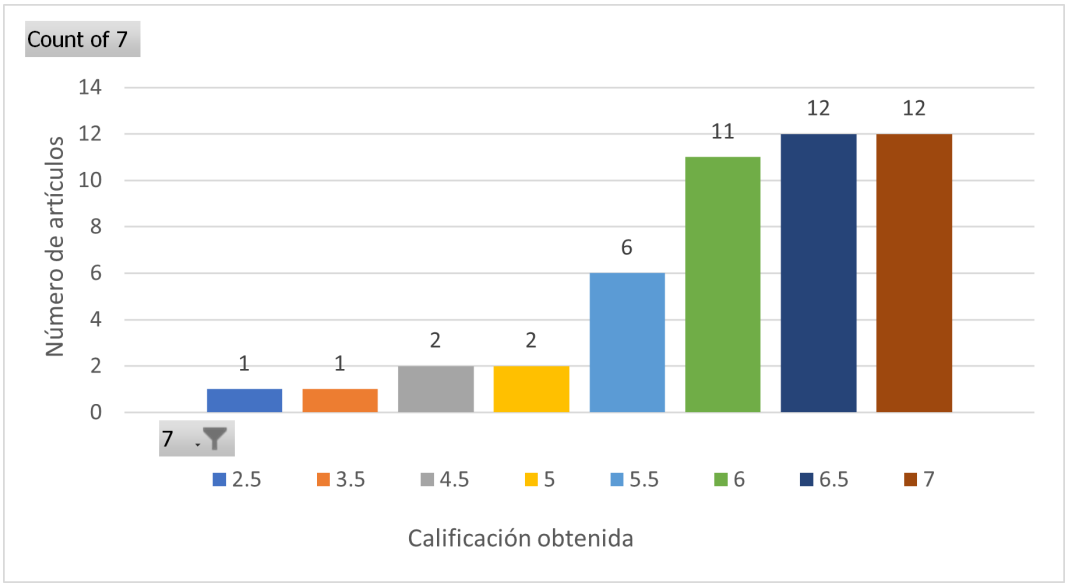


**Figura 4.2:** Distribución de calificaciones de la evaluación de calidad de los artículos analizados. Fuente: Los Autores

## 4.3 DISCUSIÓN DE LAS PREGUNTAS DE INVESTIGACIÓN

En esta sección se discuten las respuestas a las preguntas de investigación descritas en el apartado **a.** de la Sección 4.1

**a.** ***¿Cuáles son las técnicas más utilizadas para detección y predicción de SQLIA?***

De acuerdo con la investigación realizada, las técnicas con mayor impacto resultan ser aquellas que hacen usos de algoritmos de machine learning. Aproximadamente el 38 % de los artículos analizados utilizan algoritmos de machine learning.

**b.** ***¿Cuáles son las técnicas que se están utilizando para la detección y predicción de SQLIA?***

Para responder esta pregunta, se puede observar el Anexo A, donde se resumen las técnicas que han sido propuestas para la detección y predicción de SQLIA en los últimos 10 años.

**c.** ***¿Es posible clasificar las técnicas para la detección y predicción de SQLIA?***

Como se observa en el Anexo A se realizó una clasificación de las técnicas para la detección y predicción de SQLIA. En este estudio se determinó la siguiente clasificación:

- **Análisis estático:** los enfoques estáticos detectan o contrarrestan la posibilidad de un ataque de inyección SQL en la fase de compilación. Este enfoque se centra en escanear la aplicación y aprovechar el análisis del flujo de información para detectar los códigos que podrían tener vulnerabilidades [17].
- **Modificación de sentencias:**esta técnica se centra en reconstruir las consultas en tiempo de ejecución utilizando una clave criptográfica que es inaccesible para los atacantes. Esta técnica permite a los desarrolladores crear consultas SQL utilizando palabras clave aleatorias en lugar de normales, donde un proxy entre la aplicación web y la base de datos intercepta las sentencias SQL y desaleatoriza las palabras clave [17].
- **Árbol de análisis gramatical:** esta técnica comprueba en tiempo de ejecución si las consultas entrantes se ajustan a un modelo de consulta esperado. El modelo se decide en tiempo de ejecución donde examina las estructuras de la consulta

antes y después de las peticiones del cliente, es decir, se encarga de asegurar las sentencias SQL vulnerables comparándolo con un árbol de análisis sintáctico de una sentencia con el de la original y únicamente permitirá que se ejecute una sentencia con una comparación coincidente [17].

- **Técnicas Taint-based:** esta técnica aplica varias políticas de seguridad marcando los datos no fiables y rastreando sus flujos a través de los programas mediante un análisis sensible y minucioso al contexto para rechazar las consultas SQL si estas tienen una entrada no fiable [17].
- **Técnicas basadas en modelos probabilísticos:** las técnicas basadas en modelos probabilísticos se los realiza en tiempo de ejecución, donde se asume que el valor de una sentencia SQL está relacionada con la presencia o ausencia de vulnerabilidades en su estructura y de esta manera permitir la detección de un ataque de inyección SQL [18].
- **Sistemas de detección de intrusos:** los sistemas de detección de intrusos se basan en una técnica de aprendizaje automático que se entrena utilizando un conjunto de consultas típicas en aplicaciones web. La técnica empieza construyendo modelos de las consultas típicas y luego las supervisa las consultas que ingresan a la aplicación en tiempo de ejecución para identificar las consultas que no coinciden con el modelo construido [18].
- **Técnicas híbridas:** algunas técnicas combinan un análisis estático durante el desarrollo con la combinación de una supervisión dinámica en tiempo de ejecución [17], como tal es el caso de AMNESIA [19], que asocia un modelo de consulta con la ubicación de cada consulta en la aplicación y luego monitoriza la aplicación para detectar si alguna consulta se desvía del modelo esperado [18].
- **Prácticas de Codificación Segura:** las principales vulnerabilidades de inyección SQL se deben a la insuficiente validación de las entradas. Por lo tanto, la solución directa para eliminar estas vulnerabilidades es aplicar prácticas de codificación segura. Algunos ejemplos de las mejores prácticas son: comprobar el tipo de entrada en la consulta, codificación de las entradas, coincidencias positivas de patrones e identificar todas las fuentes de la entrada [18].
- **Técnicas basadas en Machine Learning:** las técnicas basadas en Machine Learning consisten en utilizar diferentes clasificadores, para detectar en una sentencia SQL los posibles ataques mediante la clasificación de los datos [20], es

decir, separar las sentencias SQL en dos grupos que contienen una etiqueta que identifique si son o no ataques. Dependiendo del clasificador que se utilice, los resultados para su detección pueden verse afectados. Unos ejemplos de estos clasificadores pueden ser Naive Bayes, Redes Neuronales Artificiales, Perceptrón Multicapa, etc.

## 4.4 CONCLUSIONES DE LA REVISIÓN SISTEMÁTICA DE LA LITERATURA

A partir de esta revisión, se pudo realizar una clasificación de los diferentes tipos de técnicas de detección de SQLIA, donde se encontró una cierta prevalencia en las técnicas que utilizan Machine Learning, particularmente los algoritmos más utilizados en esta área fueron las Redes Neuronales Artificiales (Artificial Neural Networks o ANN por sus siglas en inglés), Las Máquinas de Vectores de Soporte (Support Vector Machine o SVM por sus siglas en inglés), el Perceptrón Multicapa, y el algoritmo Naive Bayes. La gran mayoría de los artículos que fueron evaluados obtuvieron una buena calificación en la evaluación de calidad, lo que indica que se ha realizado una investigación exhaustiva, con un marco metodológico bien definido y con resultados confiables. Es importante notar el avance que ha existido en cuanto a investigaciones en el campo de las técnicas para la detección y predicción de SQLIA, ya que como se puede apreciar, existe mucha información en cuanto a la investigación dentro de esta área. En un futuro es posible profundizar en otros aspectos como los SQLIA en entornos específicos u otros ataques similares como podrían ser los ataques Cross Site Scripting (XSS). Como parte de esta investigación se pudo determinar los algoritmos más utilizados para la detección y predicción de SQLIA, y a partir de estos, realizar un análisis más a profundidad, evaluándolos con cantidades masivas de datos, utilizando datos reales en un caso de estudio como es lo que se realizará en el desarrollo de este componente

# 5 DESARROLLO E IMPLEMENTACIÓN

En esta sección se describe a detalle cada uno de los pasos seguidos para el desarrollo del componente. Como primer paso se explica el proceso de minería de datos CRISP-DM, detallando cada una de sus fases. Como siguiente paso, después de haber terminado la última fase de CRISP-DM, se empieza con el proceso de desarrollo del sistema de monitoreo, en el cual se aplica el modelo que se obtuvo de la metodología de minería de datos. Para todo el proceso que implica el desarrollo del sistema de monitoreo se aplica la metodología XP. El proceso de CRISP-DM y de la metodología XP fue realizado de manera conjunta entre todos los miembros del equipo de desarrollo, debido a que existen elementos entre los distintos componentes que son comunes para mantener consistencia al momento de realizar las evaluaciones pertinentes. Los elementos que se mantienen en común son las planificaciones, cronogramas, historias de usuario, interfaces de usuario, y de manera general, todo el proceso metodológico se lo abarca de una manera muy similar en todos los componentes del proyecto desarrollado.

## 5.1 CRISP-DM

A continuación se muestra la ejecución de la metodología CRISP-DM en cada una de sus fases:

### 5.1.1 Comprensión del negocio

En esta fase se identificaron las expectativas que tiene la “CSIRT” de la Escuela Politécnica Nacional con respecto a la implementación de algoritmos de minería de datos, que ayuden a la detección y predicción de SQLIA en los sistemas de información de esta organización.

**a. Determinación de los objetivos comerciales**

Una característica de los sistemas de información, es la de proporcionar acceso a la información, por lo que si este acceso es muy demandado, estos sistemas deben estar en la capacidad de soportar la carga que implica su uso por parte de los usuarios. Además, estos sistemas deben estar preparados para combatir la exposición a datos sensibles perpetrados por atacantes informáticos. Teniendo lo anterior como antecedente, el objetivo principal de la CSIRT es encontrar una herramienta que permita detectar y predecir los posibles ataques a sus sistemas de información y de esta manera tomar ciertas directrices que ayuden a combatir estos ataques.

**b. Evaluación de la situación**

En relación con la evaluación de la situación del CSIRT, es importante estar al tanto de los recursos que están disponibles para la elaboración de la investigación. Por lo tanto, se van a considerar los recursos descritos a continuación:

- **Personal:** En este recurso se dispone de la asesoría de personal experimentado y capacitado en el manejo de sistemas de información. El cual ayudo a solventar las dudas relacionadas con el acceso, manejo y manipulación de los registros de las bases de datos.
- **Datos:** Los datos proporcionados por la organización (CSIRT) para la minería de datos, están conformados de diversos registros, unidos en una muestra, que han sido generados por los sistemas de información utilizados en la organización.
- **Riesgos:** El principal riesgo que ha estado presente en el proyecto, fue el acceso a los datos, dado que se maneja información confidencial y es de uso exclusivo en el interior de la organización y su divulgación podría representar una amenaza a la confidencialidad de la misma. Otros riegos que tuvieron cierto impacto en el proyecto fueron el buen manejo de un cronograma y el tiempo necesario para la elaboración de la investigación.

Como el desarrollo de la investigación se lo realizó en un entorno meramente académico, no se tuvieron en cuenta aspectos relacionados con los costos y los patrocinios del proyecto.

**c. Determinación de los objetivos de minería de datos**

El objetivo principal con respecto a la minería de datos para este proyecto, es desarrollar un modelo que pueda tomar como entrada, los registros de las sentencias SQL y clasificarlas en dos tipos: datos con posibles anomalías y datos sin anomalías.

Además, es importante realizar esta clasificación lo más eficiente posible, dado que se utiliza una gran cantidad de registros para su análisis. El modelo elaborado debe tener la capacidad de reducir la cantidad de falsos negativos.

**d. Producción de un plan de proyecto**

En lo que respecta a la producción de un plan de proyecto se construyó un cronograma, en el cual se detalla cada fase de la metodología. En cada una de las fases se estableció un tiempo estimado de 4 semanas. Además, el cronograma especifica los tiempos, recursos, y riesgos asociados en cada fase. El plan del proyecto se observa en la Tabla 5.1.

**Tabla 5.1:** Plan de proyecto de minería de datos aplicando las fases de la metodología CRISP-DM

| **Fase** | **Tiempo** | **Recursos** | **Riesgos** |
|---|---|---|---|
| Comprensión del negocio | Semana 1 | Miembros del CSIRT | Falta de disponibilidad de los expertos, Falta de entendimiento del negocio. |
| Comprensión de los datos | Semana 1 y 2 | Miembros del CSIRT y equipos de desarrollo | Anomalías en los datos, confidencialidad de los datos. |
| Preparación de los datos | Semana 2 y 3 | Equipo de desarrollo | Falta de comprensión en los datos, anomalías en los datos. |
| Modelado | Semana 3 y 4 | Equipo de desarrollo | Dificultad en la implementación de los modelos requeridos. |
| Evaluación | Semana 4 | Miembros del CSIRT y equipo de desarrollo | Resultados insatisfactorios, modelos que no se adapten a las necesidades del negocio. |
| Despliegue | Semana 4 | Equipo de desarrollo | Dificultad para lograr los resultados esperados en un entorno de producción. |

En este punto ya se ha realizado el análisis de la organización, sus objetivos, necesidades, recursos y riesgos, por lo tanto, se procederá con la siguiente fase de la metodología CRISP-DM.

### 5.1.2 Comprensión de los datos

En esta fase se realiza un estudio más detallado de los datos provistos por la organización, con el objetivo de conocer como es el funcionamiento del proceso que se va a desarrollar con la minería de datos.

**a. Recopilación de datos iniciales**

En relación con la recopilación de datos iniciales, primero se identificaron los datos que se encuentran disponibles para el análisis. De tal manera que se encontró un extracto de un log constituido por sentencias SQL, que fue obtenido de las bases de datos de los sistemas de información de la organización. El log mencionado se extrajo utilizando scripts de autoría propia, motivo por el cual en todo momentos se mantiene la misma estructura dentro del archivo. La extensión que posee el archivo es de tipo JSON y su tamaño es de aproximadamente 2.5GB.

**b. Descripción de los datos**

La estructura de los logs se presenta a en el esquema 5.1:

**Listing 5.1:** Esquema del log

```
8  ...
9    "log": {
10     "file": {
11         "path": "...",
12         },
13     "offset": "...",
14     "flags": ["..."]
15     },
16     "fileset": {
17         "name": "...",
18     },
19     "message":
20         "timestamp=... ,process_id=... ,session_number=... ,user=... ,db
               =... ,app=....,client=... ,LOG: duration: ... ms bind ...:
               SELECT...
21         timestamp=... ,process_id=...,session_number=...,user=...,db=...,
```

```
            app=...,client=...,DETAIL ...
22        ...
23        ...
24        ...",
25     "fileset":{
26         "name": "...",
27     },
28     "error": {
29         "message":"Provided Grok expressions do not match field value:[
30         timestamp=... ,process_id=... ,session_number=... ,user=... ,db
               =... ,app=....,client=... ,LOG: duration: ... ms bind ...:
               SELECT...
31         timestamp=... ,process_id=...,session_number=...,user=...,db=...,
               app=...,client=...,DETAIL ...
32        ...
33        ...
34        ...]",
35     }
36   },
37   "input": {
38      "type": "...",
39   },
40 ...
```

En el esquema 5.1 se puede observar en la línea 19, la clave `"message"`, la cual representa los logs que son de interes para este estudio. En las líneas 30 y 31 que empieza con la etiqueta `"timestamp"` se muestran los diversos datos de la consulta SQL registrada y a continuación su respectiva sentencia SQL. Los registros que se encuentran en el log se dividen en consultas y parámetros. Por ejemplo, en la línea 20, el registro que termina con la sentencia `SELECT` dentro de la etiqueta `"timestamp"` representa una consulta en SQL, y de la misma forma se observa en la línea 21, un registro que termina en `DETAIL`, en el cual se especifican los valores o parámetros utilizados en esta consulta. En todo el log del esquema 5.1, la estructura mostrada se repite varias veces con un gran número de consultas SQL.

**c. Verificación de calidad de datos**

Luego de realizar un análisis en la estructura del log, se pudo encontrar un formato consistente y relativamente fácil de procesar en posteriores fases de la metodología. Entonces, como se obtuvo un correcto acceso a los datos para la evaluación y además se encontraron los campos de mayor valor para el estudio, se procede a realizar el procesamiento de estos datos.

### 5.1.3 Preparación de los datos

Cuando ya se han comprendido los datos que se van a utilizar y el proceso asociado para el desarrollo de la minería de datos, lo siguiente es la preparación de los datos, de tal manera que se pueda aplicar el modelo creado de manera efectiva. Este proceso está constituido por la limpieza de los datos, el formato e integración de los mismos en un formato estándar que no cambiara durante el desarrollo del proyecto.

**a. Selección de datos**

Para la selección de los datos se tomaron los campos más relevantes para la minería de datos, que básicamente son las consultas SQL y sus respectivos parámetros. Es necesario mencionar que para hacer uso de las consultas SQL con sus parámetros asociados, se realizó un proceso adicional que permitió unir las consultas con sus parámetros.

**b. Limpieza de datos**

En cuanto a la limpieza de los datos, se tomaron únicamente los registros del log, donde en un paso posterior se realizó un parseo de cada registro. Dicho de otra forma, si el registro es detectado como una consulta o como un parámetro, estos son separados para un mejor procesamiento de esta información.

**c. Construcción de nuevos datos**

De manera simultánea con la limpieza de los datos, cada uno de los parámetros fueron insertados en sus respectivas consultas, con el objetivo de que estas consultas se completen y puedan estar preparadas para utilizarse en fases posteriores.

**d. Integración de datos**

Como los datos proporcionados para la investigación provienen de una única fuente y en un único formato, no fue necesario realizar una integración de los datos

**e. Formato de datos**

En relación con el formato de los datos, las consultas con sus respectivos parámetros fueron almacenadas de manera secuencial en un archivo con extensión CSV para su posterior procesamiento.

Luego de la finalización de esta fase se obtuvo como resultado un archivo que contiene 3 millones de registros de sentencias SQL almacenadas en un archivo con extensión CSV.

### 5.1.4 Modelado

Tomando como punto de partida la revisión de la literatura realizada en el capítulo 3, se encontró que uno de los algoritmos más utilizados en la SQLIA es Naive Bayes. Por tal motivo se seleccionó este algoritmo para evaluar su rendimiento y efectividad en el análisis de las consultas SQL obtenidas en la fase de preparación de los datos del capítulo 5.

El algoritmo de clasificación Naive Bayes es un clasificador probabilístico que está basado en modelos que tienen con eje central la probabilidad, además se incorporan suposiciones independientes. Las suposiciones sobre la independencia por lo general no tienen impacto en la realidad. Por eso se les considera ingenuos.

Puede derivar modelos de probabilidad utilizando el teorema de Bayes (acreditado a Thomas Bayes). Según la naturaleza del modelo de probabilidad, puede entrenar el algoritmo Naive Bayes en un entorno de aprendizaje supervisado.

**a. Selección de técnica de modelado**

**Naive Bayes**

Según [21], el algoritmo Naive Bayes es un clasificador probabilístico que se basa en el Teorema de Bayes.

Este modelo se basa en la probabilidad y la incorporación de fuertes supuestos de independencia. En términos más simples, este algoritmo asume que la existencia de una característica en específico de una clase, no tiene relación alguna con la existencia de otra característica. Por ejemplo, consideramos que una fruta es un limón, solo si esta es verde, redonda y mide 4 centímetros de diámetro. Entonces, tomando en cuenta que si incluso estas características dependen entre sí o también la existencia

de otras características adicionales, todas y cada una de estas características contribuyen de manera totalmente independiente a la probabilidad de que esta sea un limón y por esta justificación se lo conoce como "Naive"

El modelo Naive Bayes no es complejo de desarrollar debido a su simplicidad. Además, este modelo en particular es muy útil cuando se tiene conjuntos de datos muy grandes.

El teorema de Bayes ayuda a expresar la probabilidad de que pueda suceder el evento A, dado que ya ha pasado el evento B, todo esto en función de la probabilidad de que suceda el evento B dado que ha sucedido el evento A, de la probabilidad del evento A y de la probabilidad del evento B [21]. A continuación se describe el teorema de Bayes que es el eje central de este algoritmo.

$$P(A|B) = \frac{P(A) * P(B|A)}{P(B)} \tag{5.1}$$

Donde:

- A y B son eventos y P(B) es diferente de 0.
- P(A|B) es la probabilidad de que suceda el evento A, teniendo en cuenta que ha sucedido el evento B.
- P(B|A) es la probabilidad de que suceda el evento B, teniendo en cuenta que ha sucedido el evento A.
- P(A) es la probabilidad de que suceda el evento A
- P(B) es la probabilidad de que suceda el evento B

**b. Generación de un diseño de comprobación**

Cuando ya se ha seleccionado la técnica del modelo, el siguiente paso consiste en la definición de los datos con los cuales se trabajará y comprobara el algoritmo seleccionado. Para la definición de los datos se utilizó un dataset obtenido de internet, el cual está compuesto de 100 mil sentencias, clasificadas entre legales e ilegales de manera uniforme. En la época actual, existen varias métricas que ayudan a medir el desempeño de los algoritmos de minería de datos. En las siguientes investigaciones [22]-[25], existe un consenso en relación con las métricas que más se utilizan para la evaluación de los modelos creados. Por lo tanto, se definieron las métricas que se utilizarán para medir el rendimiento del algoritmo seleccionado. A continuación, se describen las métricas utilizadas para su evaluación:

❒ **Matriz de confusión:** La matriz de confusión es una medida de rendimiento que está compuesta de 4 espacios donde se presentan 4 valores que ayudan al análisis de los resultados, utilizando una clasificación binaria. En esta matriz se presentan 4 combinaciones de diferentes valores entre reales y previstos.

- ✧ **Verdaderos Positivos o True Positives (TP):** estos valores representan el número de datos clasificados correctamente de forma positiva y se encuentran ubicados en la esquina superior izquierda de la matriz.
- ✧ **Verdaderos Negativos o True Negatives (TN):** estos valores representan el número de datos clasificados correctamente de forma negativa y se encuentran ubicados en la esquina inferior derecha de la matriz.
- ✧ **Falsos Negativos o False Negatives (FN):** estos valores representan el número de datos clasificados incorrectamente de forma negativa y se encuentran ubicados en la esquina superior derecha de la matriz.
- ✧ **Falsos Positivos o False Positives (FP):** estos valores representan el número de datos clasificados incorrectamente de forma positiva y se encuentran ubicados en la esquina inferior izquierda de la matriz.

En la Figura 5.1 se presenta un ejemplo de una matriz de confusión.

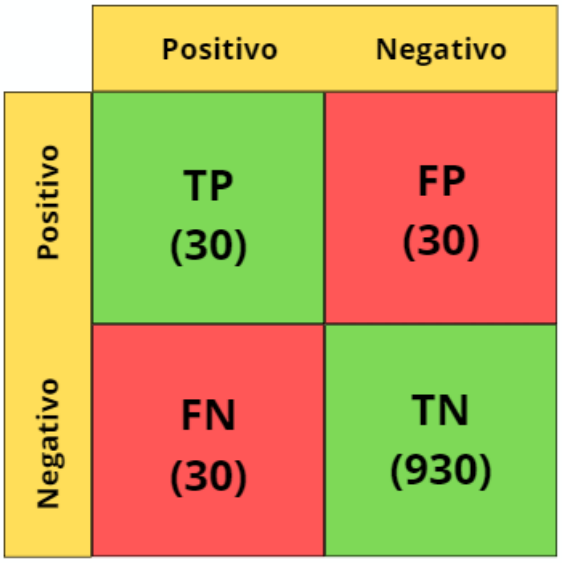


**Figura 5.1:** Ejemplo de una matriz de confusión. Fuente: El autor

❒ **Sensibilidad (SN):** Esta medida hace referencia a la proporción que existe de clasificaciones positivas realizadas de forma correctamente con respecto a los datos que realmente son positivos. Esta medida se encuentra representada por

la siguiente ecuación:

$$SN = \frac{TP}{TP + FN} \tag{5.2}$$

- **Exactitud (AC):** Esta medida representa la proporción que existe de predicciones correctas, tomando como referencia el total de predicciones que se han realizado. Esta medida se encuentra representada por la siguiente ecuación:

$$AC = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.3}$$

- **Precisión (P):** Esta medida representa la proporción que existe de clasificaciones positivas realizadas correctamente, tomando como referencia todas las clasificaciones consideradas como positivas. Esta medida se encuentra representada por la siguiente ecuación:

$$P = \frac{TP}{TP + FP} \tag{5.4}$$

- **Especificidad (SP):** Esta medida representa la proporción de clasificaciones negativas realizadas correctamente, tomando como referencia todas las clasificaciones consideradas como negativas. Esta medida se encuentra representada por la siguiente ecuación:

$$SP = \frac{TN}{TN + FP} \tag{5.5}$$

- **F1-Score:** Esta medida es muy importante dado que representa la armonía que existe entre la sensibilidad y precisión. Cuando esta medida toma un valor aproximado de 1, significa que existe una mayor armonía entre la sensibilidad y precisión, además el nivel de los valores de los falsos positivos (FP) o falsos negativos (FN) se ven bastante reducidos.

$$F1 - Score = \frac{2 * SN * P}{SN + P} \tag{5.6}$$

### c. Generación de modelos

Para el entrenamiento del modelo se utilizo un conjunto de datos obtenidos de diversas fuentes de internet y fueron etiquetados como se presenta en la siguiente tabla:

| Descripción | Etiqueta | Cantidad |
|---|---|---|
| Sentencias SQL Normales | 1 | 11382 |
| Sentencias SQL maliciosas | 0 | 19537 |

**Tabla 5.2:** Etiquetado de sentencias SQL

En lo que respecta a la generación del modelo, se realizaron varios pasos que se describen a continuación:

- ❒ **Procesamiento de los datos:** Se empezó realizando un procesamiento de los datos que se utilizaran para el entrenamiento del algoritmo, que en este caso es Naive Bayes. En relación con el procesamiento de los datos se utilizó la técnica CountVectorizer, que se encarga de transformar una compilación de texto en una matriz que cuenta la aparición de un token en específico dentro de todo el texto.
- ❒ **División del conjunto de datos:** Cuando ya se han tratado los datos, estos se dividieron de dos conjuntos grandes de datos: un conjunto de entrenamiento y otro para realizar las pruebas.
- ❒ **Entrenamiento del algoritmo:** En este paso se empieza a realizar el entrenamiento del algoritmo utilizando los datos de entrenamiento para generar un modelo.
- ❒ **Predicción de los ataques de inyección SQL:** Una vez que se ha realizado el entrenamiento, se procede a realizar la predicción de los ataques de inyección SQL utilizando los datos de prueba.
- ❒ **Evaluación del modelo:** Finalmente, todo este proceso dio como resultado un modelo que posteriormente será evaluado bajo las métricas definidas en la fase anterior llamada la generación de un diseño de comprobación.

**d. Evaluación del modelo**

Cuando ya se ha generado el modelo y se han realizado las predicciones correspondientes, se obtuvieron los siguientes resultados de las métricas definidas:

- ❒ **Matriz de confusión:**

  De la Figura 5.2, se obtuvieron los siguientes resultados

  - ✧ Verdaderos positivos o True Positives (TP) = 4142
  - ✧ Falsos Positivos o False Positives (FP) = 1688
  - ✧ Falsos Negativos o False Negatives (FN) = 49
  - ✧ Verdaderos Negativos o o True Negatives (TN) = 3397

  A continuación se detallan los cálculos de las diferentes métricas utilizadas para evaluar el modelo:

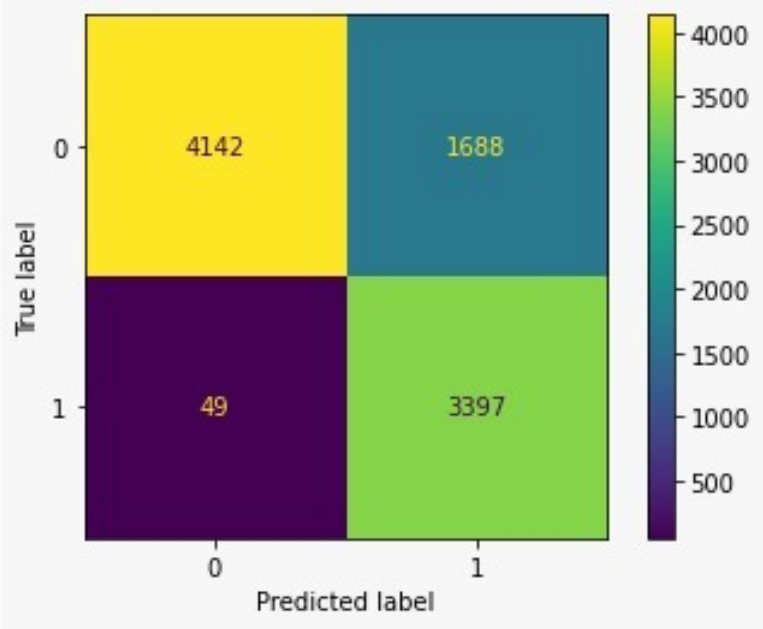


**Figura 5.2:** Matriz de confusión del algoritmo Naive Bayes. Fuente: El Autor

❒ **Sensibilidad (SN):**

$$SN = \frac{4142}{4142 + 49} = 0.9858 = 98.58\,\% \tag{5.7}$$

❒ **Exactitud (AC):**

$$AC = \frac{4142 + 3397}{4142 + 3397 + 1688 + 49} = 0.8127 = 81.27\,\% \tag{5.8}$$

❒ **Precisión (P):**

$$P = \frac{4142}{4142 + 1688} = 0.6680 = 66.80\,\% \tag{5.9}$$

❒ **Especificidad (SP):**

$$SP = \frac{3397}{3397 + 1688} = 0.7105 = 71.05\,\% \tag{5.10}$$

❒ **F1-Score:**

$$F1 - Score = \frac{2 * 0.9883 * 0.7104}{0.9883 + 0.7104} = 0.7964 = 79.64\,\% \tag{5.11}$$

A partir de los resultados presentados se puede decir que el modelo tiene resultados poco favorables, dado que el umbral mínimo obtenido de las métricas de especificidad es del 71.05 %. Además, es importante mencionar que la sensibilidad tiene un valor bastante alto

del 98.58 % haciendo alusión a que la proporción que existe de clasificaciones positivas realizadas de manera correcta con respecto a los datos que son realmente positivos.

### 5.1.5 Evaluación

De la etapa anterior se pudo obtener un modelo que fue evaluado con los registros del log proporcionados por la CSIRT de la Escuela Politécnica Nacional. Luego, con base en los resultados obtenidos, se puso a prueba el desempeño del modelo tomando en cuenta los aspectos comerciales de la organización. Con base en los resultados obtenidos, se determinó el desempeño del modelo para los aspectos de negocio.

- **Evaluación de los resultados**

  En relación con la evaluación de los resultados, el primer paso fue la de evaluar los datos proporcionados por la CSIRT en el modelo que se ha generado de la fase de modelado. Para la realización de esta evaluación, se empezó tomado una muestra de las sentencias SQL reales. Esta muestra esta divida entre sentencias SQL válidas y sentencias maliciosas, es decir, sentencias que incluyen algún tipo de SQLIA. Por lo tanto, los resultados que se obtuvieron de la evaluación son presentados en la figura 5.3.

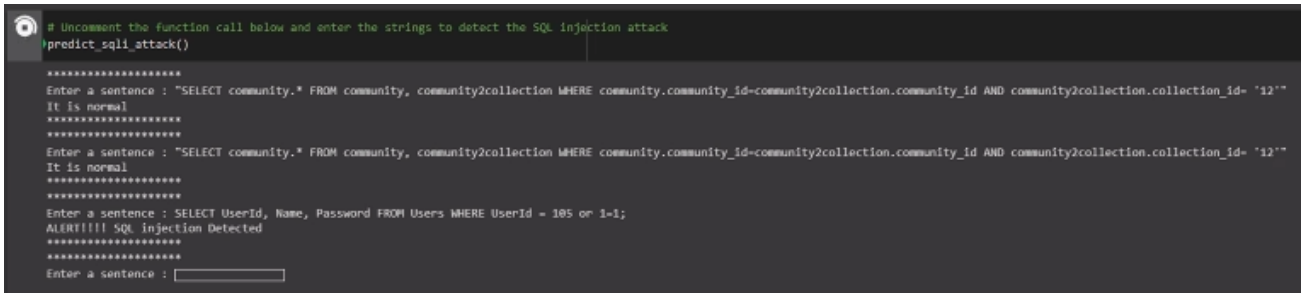


**Figura 5.3:** Resultados obtenidos de la evaluación del modelo utilizando una muestra de los registros reales provistos por la CSIRT. Fuente: Autoría propia

- **Proceso de revisión**

  En lo que respecta al proceso de revisión, se pudo observar que en los resultados obtenidos de la evaluación del modelo con los datos reales fue favorable. Dado que cuando se introdujeron sentencias SQL legítimas, el modelo no las detecto como un SQLIA. Teniendo la premisa anterior, como evidencia la podemos constatar en el bajo nivel de valores con falsos negativos que se obtuvo en la matriz de confusión asociada

a este modelo. Entonces este modelo nos da cierta confianza en lo que respecta a evitar que se filtren posibles SQLIA dentro de los registros.

❒ **Determinación de los pasos siguientes**

Teniendo como antecedente que uno de los objetivos de este proyecto es la implementación del modelo en un sistema real, sin tomar en cuenta los resultados obtenidos de su evaluación. Todo esto debido a que el sistema real debe tener la capacidad de integrarse con otros modelos utilizando otros algoritmos para su entrenamiento. Con el objetivo de comparar cada uno de estos modelos en un entorno real de producción.

### 5.1.6 Despliegue

En la fase de despliegue se explica un resumen de los descubrimientos y resultados obtenidos a partir de todo el proceso relacionado con la minería de datos. Además, también se detalla todo el plan a seguir para la implementación del modelo, su mantenimiento y una revisión final de todo el proceso de despliegue.

❒ **Planificación de despliegue**

Luego de todo el proceso involucrado en la realización de la minería de datos, es necesario explicar a mejor detalle los resultados obtenidos para que la implementación dentro de la organización (CSIRT) fuera más simple. Los artefactos obtenidos del proceso de minería de datos fueron los siguientes:

- ✧ **Algoritmo de pre-procesamiento de datos:** El algoritmo de pre-procesamiento de datos fue creado con el objetivo de facilitar el manejo de los datos de los log facilitados por la organización (CSIRT). Este algoritmo se encarga de transformar y limpiar todos los datos de los logs.
- ✧ **Modelo de machine learning:** En este apartado se obtuvo un modelo de machine learning aplicando el algoritmo Naive Bayes. El modelo obtenido está en la capacidad de poder ser aplicado en cualquier sistema, el cual se encarga de evaluar las consultas SQL y determinar cuál de estas es una consulta legítima o maliciosa.

❒ **Planificación del control y mantenimiento**

El manejo del modelo desarrollado es simple debido a que el algoritmo se encarga

de realizar una clasificación binaria, es decir, solo existen dos clasificaciones, es un ataque o no es un ataque. Por lo que se puede observar claramente si los resultados obtenidos de la evaluación reflejan o no la realidad de los datos.

El modelo desarrollado en este proyecto puede ser mejorado en trabajos futuros, dado que las medidas de desempeño pueden mejorarse aún más con el objetivo de tener un mejor resultado en relación con más métricas.

Un aspecto fundamental que es necesario mencionar es que los logs de la organización a la cual se vaya a someter a la evaluación debe tener el mismo formato, es decir, si los datos no cuentan con el formato original que se utilizan en el proyecto, estos deben ser pasados por el algoritmo de pre-procesamiento mencionado en el apartado anterior.

- ❒ **Revisión final del proyecto** Esta fase es importante dado que nos permite analizar como fue el proceso de minería de datos, los problemas que se presentaron, que fue lo que se aprendió y los trabajos futuros. Por lo tanto, en lo que respecta al manejo de los datos, se aprendió que es necesario realizar primero un análisis de los datos para comprenderlos mejor. De no realizar este paso no hubiera sido posible la limpieza de los datos. Actualmente, existen diferentes algoritmos que se utilizan para la minería de datos, pero no todos resultan efectivos para el propósito de esta investigación. Por lo tanto, el detenerse a analizar que algoritmo es mejor basado en nuestros objetivos a alcanzar fue de vital importancia.

El modelo que se obtuvo de la minería de datos requiere ser evaluado por diferentes usuarios, por lo tanto, se ha optado por realizar una página web. En la interfaz de la página web se realiza la evaluación del modelo generado utilizando un gran volumen de datos con el objetivo de poner a prueba el modelo en un entorno real. Por lo tanto, el desarrollo de esta página web y su evaluación se presentan en los siguientes capítulos.

## 5.2 DESARROLLO DEL PROTOTIPO

Para evaluar del modelo obtenido del proceso de minería de datos, se desarrolló un prototipo de un sistema web con el objetivo de que los datos de los logs reales facilitados por la organización (CSIRT) puedan ser evaluados por los usuarios mediante una interfaz gráfica de manera intuitiva. Observando los resultados en dicha interfaz. El prototipo de un sistema

web fue desarrollado aplicando la metodología XP, dado que esta se centra en las pruebas y favorece la escalabilidad del sistema por si en futuras iteraciones se requiere agregar nuevas funcionalidades. Este prototipo de un sistema web permite a los usuarios observar como se desempeñan los modelos obtenidos del proceso de minería de datos, haciendo énfasis en sus principales diferencias y desempeños frente a los otros modelos que están disponibles en el sistema.

### 5.2.1 Planeación

En esta primera fase de la metodología XP, se detallan aspectos como los requerimientos funcionales y no funcionales del sistema. Además, se definen los roles que van a estar en el proyecto y finalmente el plan de entregas que se tiene preparado para cada una de las historias de usuario.

#### 5.2.1.1 Historias de usuario

Para desarrollar el prototipo de un sistema web se empezó realizando la identificación de las principales funcionalidades con las que debe contar el sistema. Esto se lo realizo con la ayuda de entrevistas a las partes interesadas, en las cuales nos detallan el funcionamiento general de la organización y sus necesidades. Tener claro las funcionalidades del sistema cuando se está desarrollando el sistema ayuda a facilitar el uso y manipulación de los usuarios finales.
Las historias de usuario obtenidas del proceso anterior se detallan en la Tabla 5.3 donde podemos observar el código, el título, la descripción, la prioridad y el esfuerzo implicado en su desarrollo de cada una de las historias de usuario. Es necesario mencionar que el apartado de prioridad significa cuál historia de usuario es primordial en completar su desarrollo frente a las otras.

**Tabla 5.3:** Historias de usuario para el desarrollo del sistema web

| Código | Título | Descripción | Prioridad | Esfuerzo |
|---|---|---|---|---|
| HU-01 | Control de acceso | Como administrador deseo poder mantener un control de acceso dentro del sistema para evitar la divulgación de información sensible de la organización | 2 | 1 |
| HU-02 | Carga de datos | Como usuario deseo poder realizar la carga de un log de cualquier tamaño, para que pueda ser procesado y limpiado, de manera que pueda ser utilizado para detectar sentencias de ataque de inyección SQL | 4 | 2 |
| HU-03 | Análisis | Como usuario deseo evaluar diferentes modelos de machine learning para detectar ataques de inyección SQL | 3 | 3 |
| HU-04 | Visualización de logs anómalos | Como administrador del sistema, deseo visualizar las sentencias SQL detectadas como posibles ataques de inyección SQL para aumentar la seguridad de los sistemas de donde se obtuvo la información | 1 | 1 |

#### 5.2.1.2 Planificación de las entregas del proyecto

Cuando ya se han definido las historias de usuario y tomando en cuenta la prioridad y el esfuerzo estimado en su desarrollo, se elaboró la planificación de las entregas, el cual consiste en dos iteraciones de dos semanas cada una y se detallan en la Tabla 5.4.

**Tabla 5.4:** Planificación de entregas de las historias de usuario

| Código | Iteración |
|---|---|
| HU-01 | 1 |
| HU-02 | 2 |
| HU-03 | 2 |
| HU-04 | 1 |

### 5.2.2 Diseño

En relación con la fase de diseño, se define la arquitectura utilizada para el proyecto y las interfaces del sistema que va a utilizar el usuario final.

#### 5.2.2.1 Arquitectura del sistema

El prototipo del sistema web está diseñado bajo la arquitectura cliente-servidor, en la cual el cliente envía peticiones mediante el protocolo HTTP a un servidor que también responde a las solicitudes del cliente mediante el mismo protocolo. En lo que respecta a la implementación de la aplicación, se decidió utilizar el patrón arquitectónico Modelo Vista Controlador o MVC porque permite tener una mayor escalabilidad y modularidad del sistema, el cual nos permite separar toda la lógica del negocio (modelo), de las interfaces (vista) y las operaciones encargadas de la interacción con el usuario (controlador). En la figura 5.4 podemos observar la arquitectura del sistema basada en el patrón arquitectónico Modelo Vista Controlador o MVC

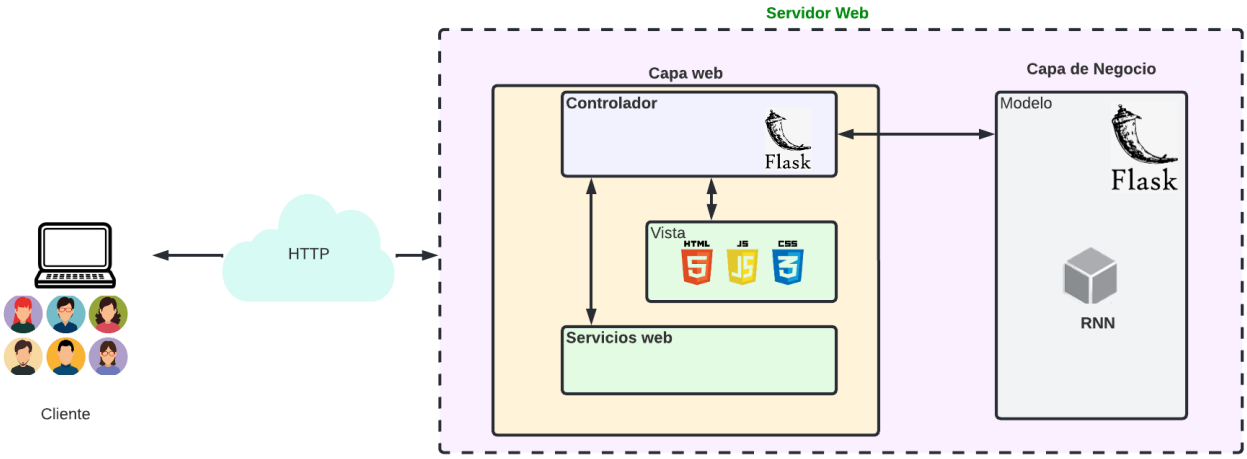


**Figura 5.4:** Diseño arquitectónico del sistema. Fuente: Los autores

#### 5.2.2.2 Diagrama de actividades del sistema

Para que se comprenda mejor de como es el flujo de actividades en el interior del sistema, se elaboró un diagrama de actividades que muestra como se está comportando el sistema cuando un usuario lo está utilizando. El diagrama de actividades del sistema se muestra en la Figura 5.5.

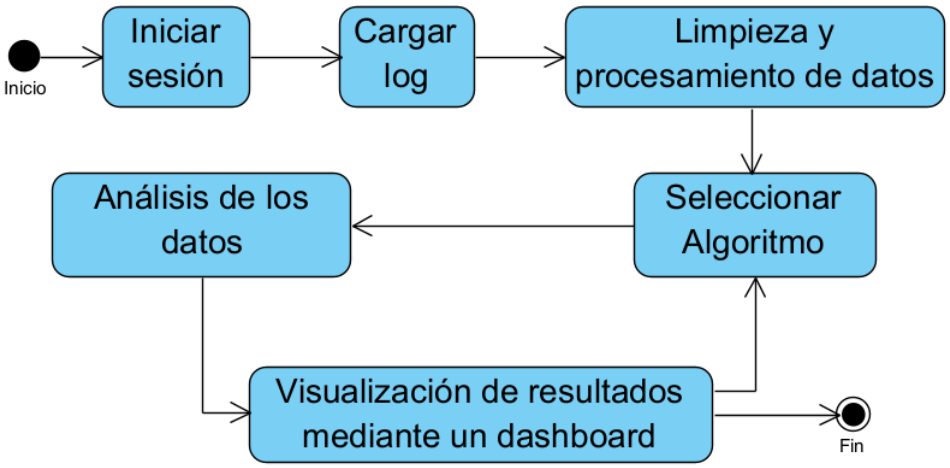


**Figura 5.5:** Diagrama de actividades del sistema. Fuente: Los autores

#### 5.2.2.3 Diseño de interfaces del sistema

Cuando ya se ha comprendido el funcionamiento del flujo de actividades que ocurren en el sistema. El siguiente paso es la elaboración de mockups, que son bosquejos de como idealizamos cada una de las interfaces. Que en pasos posteriores se plasmaran estos bosquejos a interfaces que van a utilizar el usuario. En las Figuras 5.6, 5.7, 5.8, 5.9 se muestran los bosquejos o mockups que se desarrollaron para la elaboración de las interfaces.

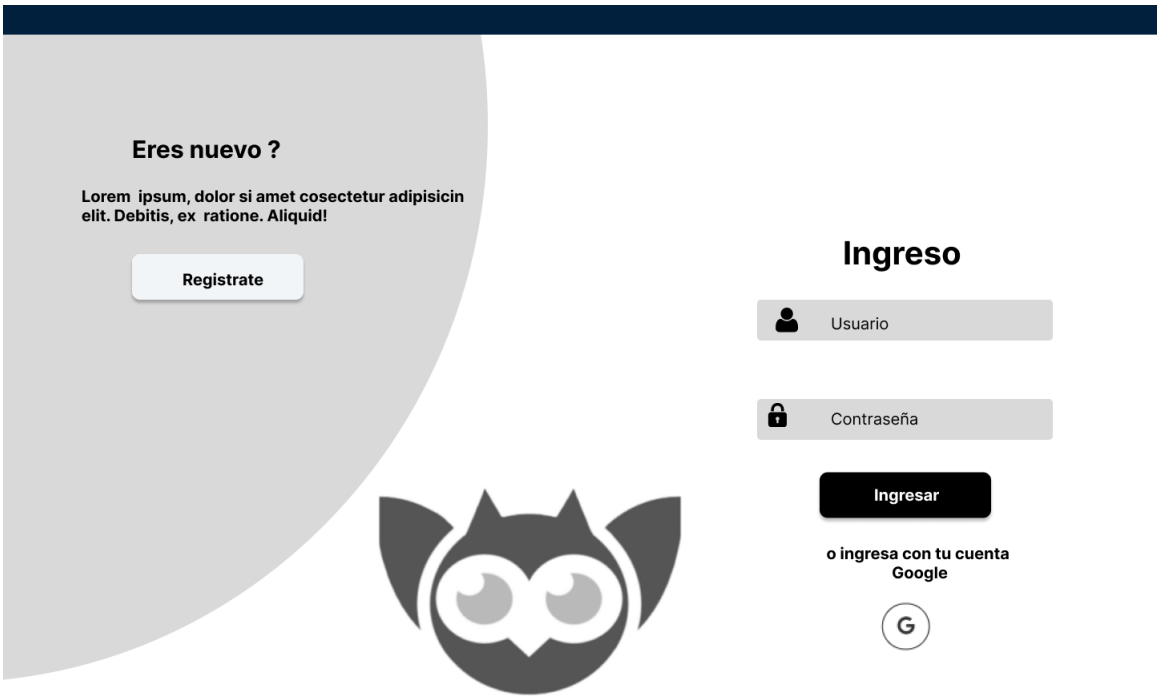


**Figura 5.6:** Mockup de la pantalla de inicio de sesión Fuente: Los autores

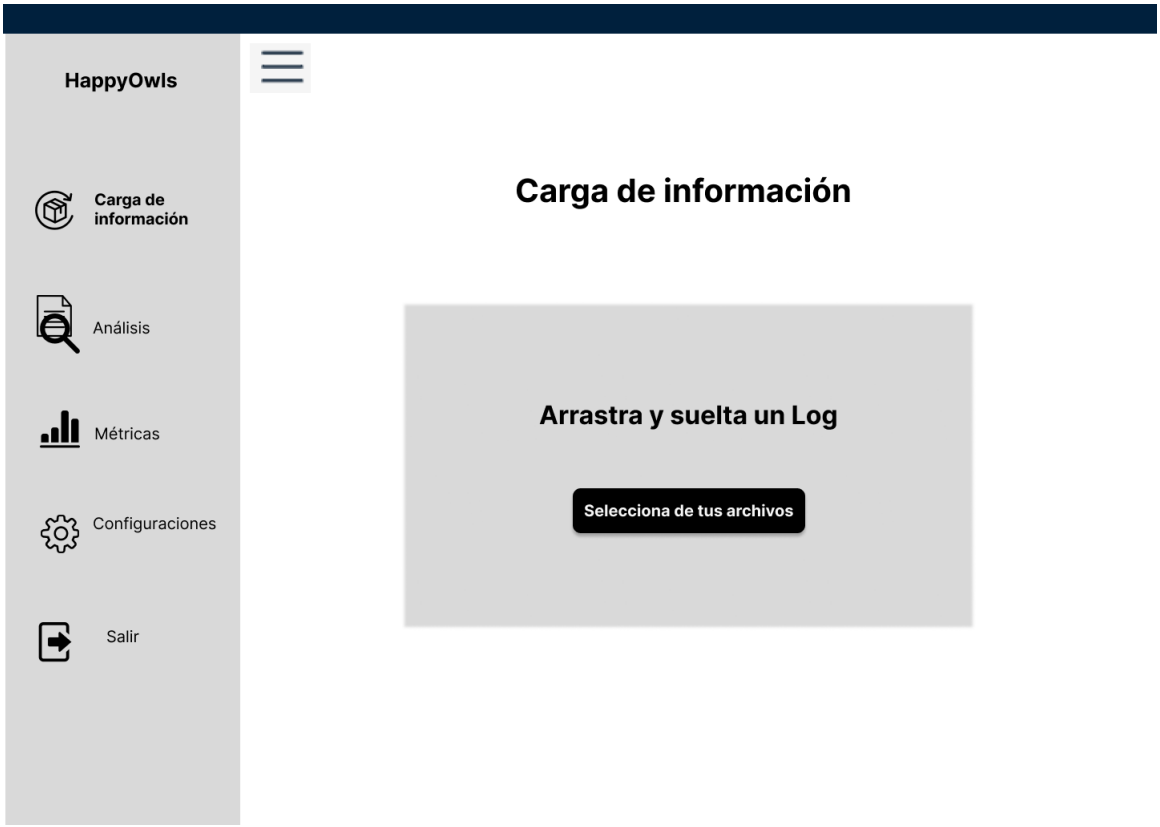


**Figura 5.7:** Mockup de la pantalla de selección y carga del archivo de logs. Fuente: Los autores

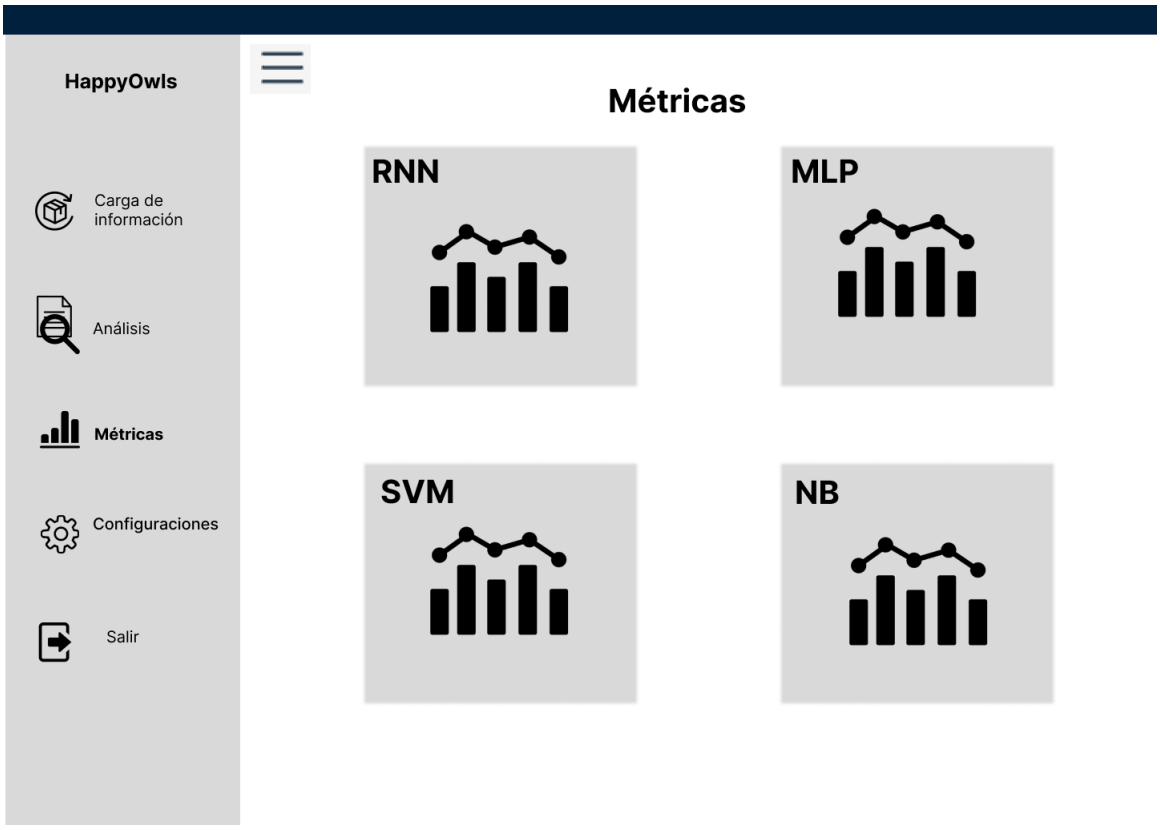


**Figura 5.8:** Mockup de la pantalla de selección del algoritmo para evaular el log


**Figura 5.9:** Mockup del dashboard de resultados del análisis del log

### 5.2.3 Codificación del sistema

En la fase se describe como se realizó la codificación. Este proyecto se lo realizo en dos iteraciones. Estas iteraciones se las describe a continuación:

#### 5.2.3.1 Primera iteración

En la primera iteración se contempló el desarrollo de las historias de usuario HU-01 y HU-04. El desarrollo de estas historias de usuario en esta primera iteración se debe a que en colaboración con las partes interesadas se llegó al mutuo acuerdo de que estas historias aportan el mayor valor para la organización y se deben desarrollar primero. En la historia de usuario (HU-01) se desarrolla el control de acceso de los usuarios en la aplicación. En la historia de usuario (HU-02) se desarrolla el modelo que permite la detección de SQLIAs.

El desarrollo del control de acceso de los usuarios en la aplicación se muestra en la Figura 5.10.

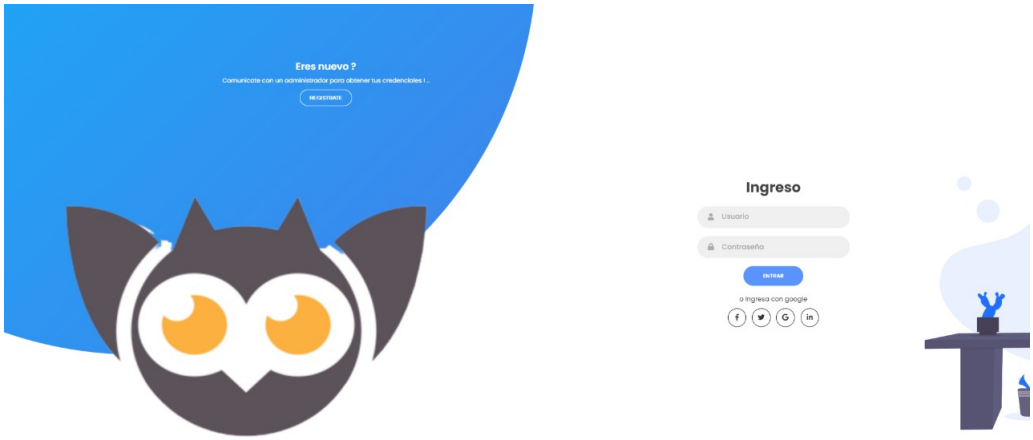


**Figura 5.10:** Interfaz del login del sistema. Fuente: Los autores

El desarrollo del modelo que permite la detección de los ataques de inyección SQL se muestra en la Figura 5.11.

#### 5.2.3.2 Segunda iteración

En la segunda iteración se desarrolló la integración con los otros modelos basados en diferentes algoritmos que se encuentran definidos en los otros componentes del proyecto. El

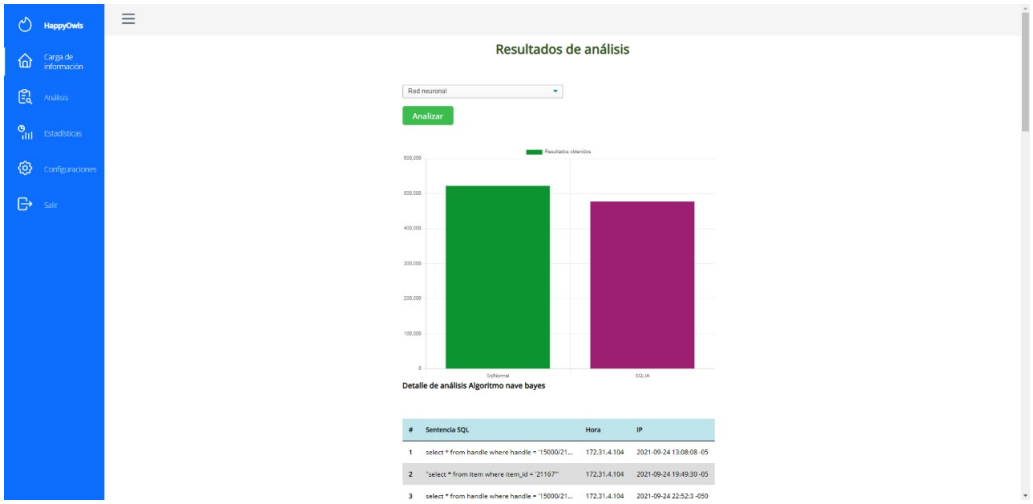


**Figura 5.11:** Interfaz de análisis de logs del sistema. Fuente: Los autores

objetivo de la integración de todos los modelos en un mismo sistema es la de que el usuario pueda seleccionar entre cada uno de estos y poder desarrollar un análisis y comparativa entre estos. Además, se realizó la funcionalidad que permite al usuario poder ingresar un log sin procesar, para que el sistema pueda procesar y limpiar estos logs y puedan ser analizados por el modelo. Esta funcionalidad ayuda al usuario a que no deba procesar los logs manualmente, sino simplemente ingresar los logs directamente en el sistema.

La interfaz en donde se permite al usuario realizar la carga de un log sin procesar se muestra en la Figura 5.12.

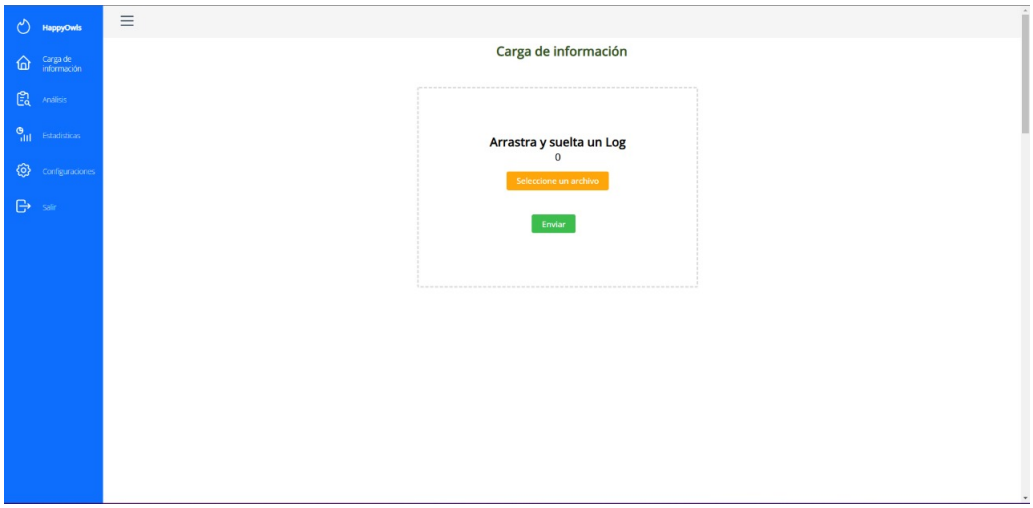


**Figura 5.12:** Interfaz carga log sin procesar en el sistema. Fuente: Los autores

### 5.2.4 Pruebas

Cuando ya se ha desarrollado el sistema web, utilizando el modelo que se generó del proceso de minería de datos utilizando la metodología CRISP-DM, se empezó a realizar las pruebas del sistema. En las pruebas del sistema se utilizan datos reales facilitados por la organización (CSIRT).
En lo que respecta a las pruebas se realizó lo siguiente:

- **Carga del log:** en este paso se cargó un log con los datos facilitados por la organizacion, comprimido en formato ZIP El algoritmo de pre-procesamiento de datos fue creado con el objetivo de facilitar el manejo de los datos de los log facilitados por la organización (CSIRT). Este algoritmo se encarga de transformar y limpiar todos los datos de los logs.

- **Proceso de descompresión:** en esta paso el sistema procede a descomprimir todos los logs.

- **Procesado y analisis mediante el modelo generado:** en este paso el sistema se encarga de analizar y procesar los logs y los va a ir clasificando en sentencias SQL legítimas y sentencias SQL maliciosas.

- **Presentación de resultados:** en este paso el sistema muestra esta clasificación de los logs en una gráfica que pueda entender el usuario.

En la Figura 5.13 se puede observar la presentación de resultados al usuario del sistema.


**Figura 5.13:** Resultado del análisis de un millón de sentencias SQL dentro del log cargado en el sistema. Fuente: El autor

Como se puede apreciar, el sistema aún muestra una gran tendencia a detectar sentencias SQL válidas como posibles SQLIAs, por lo que no resulta en un modelo muy efectivo para detectar estos ataques en este caso de estudio. Sin embargo, el modelo fue correctamente implementado en el sistema y la funcionalidad se encuentra integrada de manera adecuada.

# 6 ANALISIS DE RESULTADOS, CONCLUSIONES, RECOMENDACIONES

En este punto del trabajo ya se ha realizado el componente en todas sus fases, por lo tanto, en esta sección se detallará el análisis de resultados obtenidos luego de la implementación del componente. Además, se realiza una comparación del desempeño de los algoritmos utilizados en los otros componentes que conforman este proyecto. Utilizando las métricas establecidas en la evaluación del modelo del capítulo 5. Para de esta manera concluir de estos resultados, brindar recomendaciones y detallar las áreas de mejora de este proyecto para trabajos futuros.

## 6.1 ANÁLISIS Y COMPARACIÓN DE RESULTADOS

En esta sección se detalla el análisis del algoritmo evaluado en este componente, que en este caso es Naive Bayes. Además, se presenta una comparación de los resultados obtenidos de la evaluación de los algoritmos de los otros componentes, que en este caso son los algoritmos Perceptrón multicapa (MLP), Máquinas de vectores de soporte (SVM), Red neuronal recurrente (RNN) y Naive Bayes.

### 6.1.1 Análisis de resultados

Según los resultados obtenidos a partir de la evaluación del modelo en el prototipo del sistema web, se logró observar que el modelo presenta un bajo desempeño, dado que el número de falsos positivos fue bastante alto, dejando como evidencia que las sentencias SQL ingresadas para el análisis que realmente son un SQLIA, el modelo las muestra como que no es un SQLIA. Por lo tanto, es importante realizar un nuevo análisis a más detalle con el objetivo de encontrar la razón por la cual el desempeño del algoritmo Naive Bayes se ve

bastante reducido cuando se lo evalúa en un ambiente real. Un punto importante que puede repercutir directamente en el desempeño del modelo puede ser el conjunto reducido de datos que se utilizaron para su entrenamiento debido a cuestiones de poder computacional. Por lo tanto, se podría mejorar este modelo, utilizando una mayor cantidad de datos para tener una mayor precisión, dado que el algoritmo Naive Bayes se basa en el número de los vecinos más cercanos en un conjunto de datos para clasificar un nuevo dato entrante. Entonces, si tenemos más datos para el entrenamiento, el número de vecinos cercanos a un dato que poseen las mismas características aumenta, dando como resultado un modelo más preciso.

### 6.1.2 Comparación de resultados

Como parte de este componente, se contempla la comparación de los resultados obtenidos en la evaluación de los algoritmos realizada tanto en el proceso de CRISP-DM, como al final del desarrollo del prototipo bajo la metodología de desarrollo de software XP.

#### 6.1.2.1 Comparación de resultados del proceso CRISP-DM

A continuación se comparan los resultados obtenidos de la evaluación de los algoritmos en los diferentes componentes durante la ejecución de la metodología CRISP-DM. Los resultados obtenidos en esta fase en cada uno de los componentes se resumen en la Tabla 6.1.

**Tabla 6.1:** Comparación de métricas de los algoritmos evaluados.

| Métricas | MLP | SVM | RNN | Naive Bayes |
|---|---|---|---|---|
| **Matriz de confusión** | TP:3179<br>TN:5740<br>FP:90<br>FN:267 | TP:1032<br>TN:5829<br>FP:1<br>FN:230 | TP:3216<br>TN:5712<br>FP:118<br>FN:49 | TP: 3397<br>TN: 4142<br>FP: 1688<br>FN: 49 |
| **Exactitud (AC)** | 96.15 % | 73.97 % | 96.25 % | 81.27 % |
| **Especificidad (SP)** | 98.46 % | 99.98 % | 97.98 % | 71.05 % |
| **Sensibilidad (SN)** | 92.25 % | 29.95 % | 93.33 % | 98.58 % |
| **Precisión (P)** | 97.25 % | 99.90 % | 96.46 % | 66.80 % |
| **F1-Score** | 94.68 % | 46.08 % | 94.87 % | 79.64 % |

De los resultados obtenidos del entrenamiento de los modelos, se analizaron las métricas para determinar cuáles algoritmos tienen un mejor desempeño durante este entrenamiento en cada una de las métricas seleccionadas. Tanto los modelos obtenidos de los algoritmos MLP y RNN muestran un gran desempeño (mayor al 90 %) en todas las métricas, por lo que se determina que son algoritmos muy equilibrados; esto puede comprobarse con su puntaje en el F1-Score, donde ambos destacan muy por encima de los otros algoritmos, por lo que son muy buenos candidatos para implementar en un ambiente real. En cuanto al algoritmo evaluado en este componente, Naive Bayes, este posee una gran sensibilidad detectando SQLIA. Sin embargo, esto se descompensa en su precisión, generando problemas de ser implementado en un ambiente real, debido a que muchas de las sentencias ingresadas serían detectadas como SQLIA aunque no lo sean. Por último, el algoritmo SVM posee mucha dificultad detectando SQLIA, afectando drásticamente su puntaje. Un aspecto mucho más preocupante, es el hecho de la seguridad, ya que, si el modelo no es capaz de detectar ataques, no es apto para ser implementado en un sistema donde es crucial que estos sean detectados oportunamente. En la generación y evaluación del modelo dentro del proceso de CRISP-DM, tanto los algoritmos MLP y RNN resultan buenos candidatos para ser implementados en un sistema real, donde el algoritmo RNN posee una ligera ventaja debido a que posee más sensibilidad al momento de detectar posibles SQLIA.

#### 6.1.2.2 Comparación de resultados de la metodología XP

Al evaluar estos cuatro algoritmos en un ambiente real utilizando un dataset de un millón de datos, se obtuvieron los resultados que se pueden observar a continuación:


**Figura 6.1:** Resultado de la evaluación del algoritmo Naive Bayes en el prototipo desarrollado. Fuente: El autor

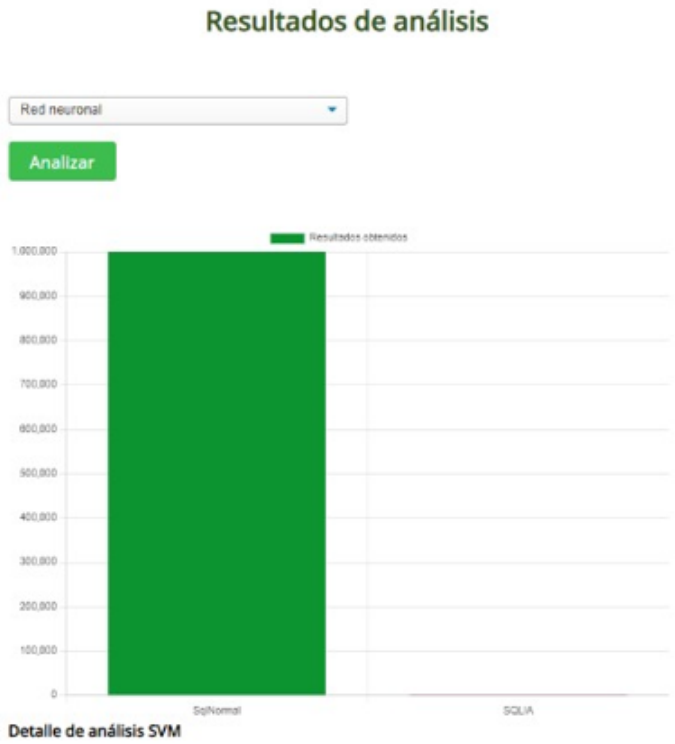


**Figura 6.2:** Resultado de la evaluación del algoritmo SVM en el prototipo desarrollado. Fuente: El autor

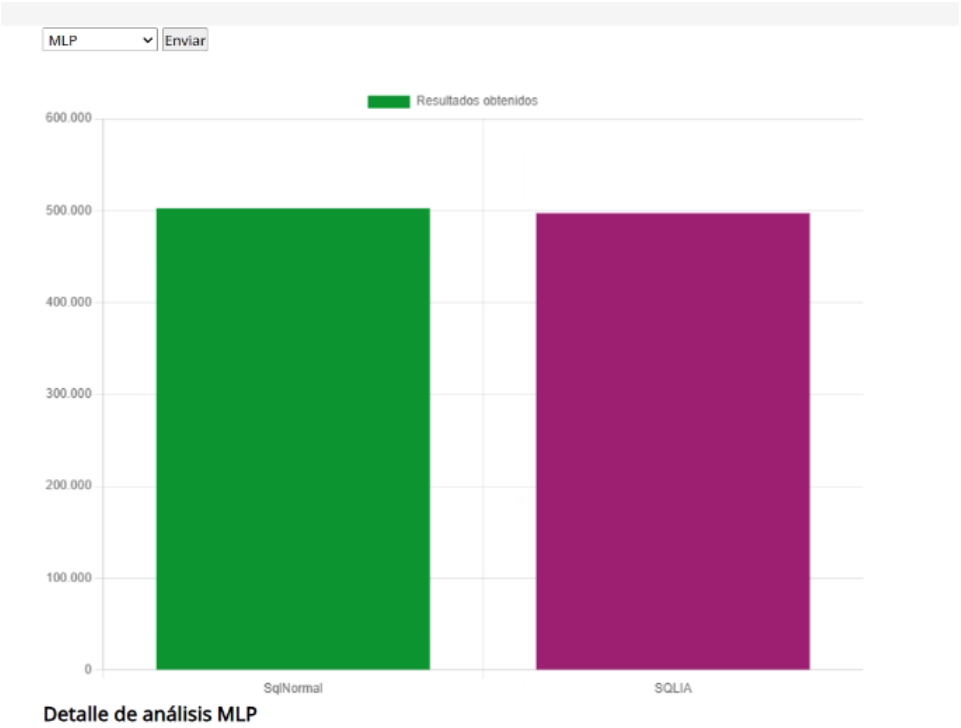


**Figura 6.3:** Resultado de la evaluación del algoritmo MLP en el prototipo desarrollado. Fuente: El autor


**Figura 6.4:** Resultado de la evaluación del algoritmo RNN en el prototipo desarrollado. Fuente: El autor

Los resultados obtenidos de esta evaluación muestran particularidades con respecto a los obtenidos cuando se generaron los modelos. En cuanto a los algoritmos Naive Bayes y SVM, no se presentan muchas novedades; Naive Bayes presenta una fuerte tendencia a clasificar sentencias como posibles ataques a pesar de no serlos, y, por otra parte, el algoritmo SVM, a pesar de no detectar ningún posible ataque, este resultado no es muy confiable dado su desempeño dentro del entrenamiento. El algoritmo MLP muestra dificultad al momento de clasificar correctamente los ataques, ya que detecta que aproximadamente el 50 % de las sentencias ingresadas son SQLIAs, cuando basta tomar una pequeña muestra estadística para darse cuenta de que este porcentaje difiere mucho de la realidad. Por último, el algoritmo RNN mostró un gran desempeño en el entrenamiento, y en su evaluación dentro del prototipo muestra un desempeño muy similar, detectando una cantidad mínima de posibles SQLIAs; sin embargo, este resultado resulta más confiable que el obtenido por el algoritmo SVM puesto como se observó en los resultados del proceso CRISP-DM, este modelo tiene mucha dificultad detectando efectivamente los SQLIAs. De este análisis se puede determinar que el algoritmo más eficaz para la detección de posibles SQLIAs en un ambiente real, con datos reales que difieren de los datos que fueron utilizados en el entrenamiento, es el algoritmo RNN.

## 6.2 CONCLUSIONES

- ❒ En relación con la revisión de la literatura, se logró realizar la misma basada en la detección y predicción de SQLIAs. Además, en esta revisión se pudo recabar las técnicas más utilizadas actualmente para realizar la predicción de SQLIA, donde en su mayoría destacan las técnicas basadas en Machine Learning. Adicional a esto, se obtuvieron los algoritmos más utilizados en el área del Machine Learning. Por lo tanto, para el desarrollo de este componente se seleccionó el algoritmo Naive Bayes para evaluar su desempeño en la detección y predicción SQLIAs.

- ❒ Mediante la metodología CRISP-DM, se puedo realizar el proceso de minería de datos para obtener un modelo basado en Machine Learning con el algoritmo Naive Bayes. En lo que respecta al entrenamiento del modelo, se pudo observar resultados prometedores basados en las métricas definidas para evaluar su desempeño. Pero cuando se utilizó este modelo para evaluar su desempeño con datos reales, los resultados obtenidos no fueron los adecuados. Motivo por el cual sería necesario realizar un nuevo

análisis de los datos que han sido evaluados para identificar cuáles fueron los motivos de la reducción de su desempeño.

- En lo que respecta a la evaluación del modelo basado en el algoritmo Naive Bayes, se logró desarrollar un prototipo de sistema web mediante la metodología XP con éxito. En este prototipo se logró que las partes interesadas (usuarios), puedan evaluar este modelo mediante una interfaz intuitiva, facilitando su uso.

- El modelo obtenido del proceso de minería de datos utilizando la metodología CRISP-DM y puesto en marcha en un prototipo de sistema web. Se evidencia que cuando se utiliza una muestra bastante grande de datos, el modelo presenta resultados poco favorables para los objetivos de este estudio, por lo tanto, este modelo no logro cumplir con las expectativas.

- Se logró realizar una comparativa entre los algoritmos evaluados en cada componente del trabajo integrador. De esta comparativa se determinó que si bien los algoritmos MLP y RNN presentan ambos un buen desempeño, al final de la aplicación de la metodología CRISP-DM, solo el algoritmo RNN mantiene este desempeño dentro de un ambiente real, al momento de evaluarlo en el prototipo creado con la metodología XP. Por esta razón se determinó que el algoritmo RNN es el más eficiente para la detección y predicción de SQLIAs en este caso de estudio.

## 6.3 RECOMENDACIONES

- El punto central en el desarrollo de este proyecto fue la utilización de varias metodologías que guiaban cada una de las fases. Por lo tanto, se recomienda que cuando se desarrolle algún tipo de proyecto en cualquier área de investigación, se haga primero un estudio de las metodologías que se pueden utilizar dentro del proyecto, con el objetivo de seleccionar la más adecuada basada en los objetivos que requieren cumplirse dentro del proyecto.

- Las metodologías utilizadas para la minería de datos (CRISP-DM) y de desarrollo de software (Programación Extrema), tiene la característica de ser iterativa. Por lo tanto, el equipo de trabajo involucrado en el proyecto debe tener esta característica presente y estar preparados para realizar cambios en el proyecto a medida que van surgiendo problemas y de esta manera solucionarlos oportunamente.

- ❒ Antes de empezar a realizar el modelo, el entrenamiento y las pruebas del modelo, es importante que se realice un estudio y análisis de los datos con los cuales va a trabajar el modelo. Motivo por el cual si no se comprende como están estructurados los datos facilitados por la organización, puede que el modelo no presente los resultados esperados cuando se evalúe su desempeño.

## 6.4 TRABAJO FUTURO

Teniendo en cuenta que el alcance del componente que fue desarrollado es limitado, es importante hacer énfasis en las opciones de mejora o extensiones que pueden desarrollarse a partir de esta investigación para futuros trabajos.

- ❒ Como se evidenció en la fase de análisis de resultados, el algoritmo Naive Bayes desarrollado dentro en el componente no fue óptimo en cuanto a las métricas propuestas para su evaluación. Por lo tanto, queda como investigación el determinar cuál fue la causante de que este algoritmo no funcione correctamente para la detección y predicción de SQLIAs.
- ❒ Una característica dentro del basto campo del procesamiento de grandes cantidades de datos es la velocidad. Tomando la premisa anterior, actualmente existen algunas tecnologías que se centran en el Big Data como por ejemplo Apache Spark, Hadoop o el multiprocesamiento. Pero su utilización en el desarrollo de este componente no fue favorable, dado que existen varias limitaciones en cuanto al poder computacional necesario. Por lo tanto, todavía puede mejorarse este componente mediante la utilización de estas tecnologías con el objetivo de obtener un resultado más eficiente en cuando al procesamiento de datos.

# 7 REFERENCIAS BIBLIOGRÁFICAS


[1] OWASP, *OWASP top 10:2021*, en, https://owasp.org/Top10/, Accessed: 2022-1-3, 2021.

[2] M. Figueroa y A. Gustavo, «La metodologı´a de elaboración de proyectos como una herramienta para el desarrollo cultural,» 2005.

[3] B. Kitchenham, «Procedures for performing systematic reviews,» *Keele, UK, Keele University*, vol. 33, n.º 2004, págs. 1-26, 2004.

[4] R. Wirth y J. Hipp, «CRISP-DM: Towards a standard process model for data mining,» en *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, Manchester, vol. 1, 2000, págs. 29-39.

[5] R. S. Pressman y J. M. Troya, «Ingenierı´a del software,» 1988.

[6] F. E. Arevalo-Cordovilla, A. E. Cortez-Lara, I. B. Ordoñez-Sigcho y J. E. Solı´s-Gaibor, *Importancia de la seguridad en los Sistemas de Información*, 2020.

[7] *Vulnerabilidades y violaciones digitales en productos de software: análisis de regresión logística*. DOI: 10.1109/ICCWS48432.2020.9292397.

[8] S. Samonas y D. Coss, «The CIA strikes back: Redefining confidentiality, integrity and availability in security.,» *Journal of Information System Security*, vol. 10, n.º 3, 2014.

[9] M. Chen, S. Mao e Y. Liu, *Big data: A survey*, 2014.

[10] W. Fan y A. Bifet, *Mining big data: current status, and forecast to the future*, 2013.

[11] M. Malik y T. Patel, «Database security-attacks and control methods,» *International Journal of Information*, vol. 6, n.º 1/2, págs. 175-183, 2016.

[12] B. Nagpal, N. Chauhan y N. Singh, «A survey on the detection of SQL injection attacks and their countermeasures,» *Journal of Information Processing Systems*, vol. 13, n.º 4, págs. 689-702, 2017.

[13] J. Han, J. Pei y H. Tong, *Data mining: concepts and techniques*. Morgan kaufmann, 2022.

[14] A. S. Osman, «Data mining techniques,» 2019.

[15] B. Kitchenham y S. Charters, *Guidelines for performing Systematic Literature Reviews in Software Engineering*, 2007.

[16] R. K. Jamra, B. Anggorojati, D. I. Sensuse, R. R. Suryono et al., «Systematic Review of Issues and Solutions for Security in E-commerce,» en *2020 International Conference on Electrical Engineering and Informatics (ICELTICs)*, IEEE, 2020, págs. 1-5.

[17] Y.-C. Chung, M.-C. Wu, Y.-C. Chen y W.-K. Chang, «A Hot Query Bank approach to improve detection performance against SQL injection attacks,» *computers & security*, vol. 31, n.º 2, págs. 233-248, 2012.

[18] W. G. Halfond, J. Viegas, A. Orso et al., «A classification of SQL-injection attacks and countermeasures,» en *Proceedings of the IEEE international symposium on secure software engineering*, IEEE, vol. 1, 2006, págs. 13-15.

[19] W. G. J. Halfond y A. Orso, «AMNESIA: Analysis and Monitoring for NEutralizing SQL-Injection Attacks,» en *Proceedings of the 20th IEEE/ACM International Conference on Automated Software Engineering*, ép. ASE '05, Long Beach, CA, USA: Association for Computing Machinery, 2005, págs. 174-183, ISBN: 1581139934. DOI: 10.1145/1101908.1101935. dirección: https://doi.org/10.1145/1101908.1101935.

[20] M. Hasan, Z. Balbahaith y M. Tarique, «Detection of SQL injection attacks: a machine learning approach,» en *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*, IEEE, 2019, págs. 1-6.

[21] I. Rish et al., «An empirical study of the naive Bayes classifier,» en *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, 2001, págs. 41-46.

[22] M. Hossin y M. N. Sulaiman, «A review on evaluation metrics for data classification evaluations,» *International journal of data mining & knowledge management process*, vol. 5, n.º 2, pág. 1, 2015.

[23] M. Grandini, E. Bagli y G. Visani, «Metrics for multi-class classification: an overview,» *arXiv preprint arXiv:2008.05756*, 2020.

[24] Z. Vujović, «Classification model evaluation metrics,» *International Journal of Advanced Computer Science and Applications*, vol. 12, n.º 6, págs. 599-606, 2021.

[25] B. J. Erickson y F. Kitamura, «Magician's corner: 9. Performance metrics for machine learning models,» *Radiology: Artificial Intelligence*, vol. 3, n.º 3, 2021.

[26] I. Lee, S. Jeong, S. Yeo y J. Moon, «A novel method for SQL injection attack detection based on removing SQL query attribute values,» *Mathematical and Computer Modelling*, vol. 55, n.º 1, págs. 58-68, 2012, Advanced Theory and Practice for Cryptography and Future Security, ISSN: 0895-7177. DOI: https://doi.org/10.1016/j.mcm.2011.01.050. dirección: https://www.sciencedirect.com/science/article/pii/S0895717711000689.

[27] C. I. Pinzón, J. F. De Paz, Á. Herrero, E. Corchado, J. Bajo y J. M. Corchado, «idMAS-SQL: Intrusion Detection Based on MAS to Detect and Block SQL injection through data mining,» *Information Sciences*, vol. 231, págs. 15-31, 2013, Data Mining for Information Security, ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2011.06.020. dirección: https://www.sciencedirect.com/science/article/pii/S0020025511003148.

[28] M.-Y. Kim y D. H. Lee, «Data-mining based SQL injection attack detection using internal query trees,» *Expert Systems with Applications*, vol. 41, n.º 11, págs. 5416-5430, 2014, ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2014.02.041. dirección: https://www.sciencedirect.com/science/article/pii/S0957417414001171.

[29] H. Shahriar y M. Zulkernine, «Information-Theoretic Detection of SQL Injection Attacks,» en *2012 IEEE 14th International Symposium on High-Assurance Systems Engineering*, 2012, págs. 40-47. DOI: 10.1109/HASE.2012.31.

[30] S. Som, S. Sinha y R. Kataria, «Study on sql injection attacks: Mode detection and prevention,» *International Journal of Engineering Applied Sciences and Technology*, vol. 1, n.º 8, págs. 23-29, 2016.

[31] I. Balasundaram y E. Ramaraj, «An Efficient Technique for Detection and Prevention of SQL Injection Attack using ASCII Based String Matching,» *Procedia Engineering*, vol. 30, págs. 183-190, 2012, International Conference on Communication Technology and System Design 2011, ISSN: 1877-7058. DOI: https://doi.org/10.1016/j.proeng.2012.01.850. dirección: https://www.sciencedirect.com/science/article/pii/S1877705812008600.

[32] T. Latchoumi, M. S. Reddy y K. Balamurugan, «Applied machine learning predictive analytics to SQL injection attack detection and prevention,» *European Journal of Molecular & Clinical Medicine*, vol. 7, n.º 02, pág. 2020, 2020.

[33] P. Tang, W. Qiu, Z. Huang, H. Lian y G. Liu, «Detection of SQL injection based on artificial neural network,» *Knowledge-Based Systems*, vol. 190, pág. 105 528, 2020, ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2020.105528. dirección: https://www.sciencedirect.com/science/article/pii/S0950705120300332.

[34] Q. Li, W. Li, J. Wang y M. Cheng, «A SQL Injection Detection Method Based on Adaptive Deep Forest,» *IEEE Access*, vol. 7, págs. 145 385-145 394, 2019. DOI: 10.1109/ACCESS.2019.2944951.

[35] N. M. Sheykhkanloo, «Employing Neural Networks for the Detection of SQL Injection Attack,» en *Proceedings of the 7th International Conference on Security of Information and Networks*, ép. SIN '14, Glasgow, Scotland, UK: Association for Computing Machinery, 2014, págs. 318-323, ISBN: 9781450330336. DOI: 10.1145/2659651.2659675. dirección: https://doi.org/10.1145/2659651.2659675.

[36] D. Kar, S. Panigrahi y S. Sundararajan, «SQLiDDS: SQL Injection Detection Using Query Transformation and Document Similarity,» en *Distributed Computing and Internet Technology*, R. Natarajan, G. Barua y M. R. Patra, eds., Cham: Springer International Publishing, 2015, págs. 377-390, ISBN: 978-3-319-14977-6.

[37] A. Ghafarian, «A hybrid method for detection and prevention of SQL injection attacks,» en *2017 Computing Conference*, 2017, págs. 833-838. DOI: 10.1109/SAI.2017.8252192.

[38] X. Xie, C. Ren, Y. Fu, J. Xu y J. Guo, «SQL Injection Detection for Web Applications Based on Elastic-Pooling CNN,» *IEEE Access*, vol. 7, págs. 151 475-151 481, 2019. DOI: 10.1109/ACCESS.2019.2947527.

[39] Y. Wang y Z. Li, «SQL injection detection via program tracing and machine learning,» en *International Conference on Internet and Distributed Computing Systems*, Springer, 2012, págs. 264-274.

[40] H. Gu, J. Zhang, T. Liu et al., «DIAVA: A Traffic-Based Framework for Detection of SQL Injection Attacks and Vulnerability Analysis of Leaked Data,» *IEEE Transactions on Reliability*, vol. 69, n.º 1, págs. 188-202, 2020. DOI: 10.1109/TR.2019.2925415.

[41] R. A. Katole, S. S. Sherekar y V. M. Thakare, «Detection of SQL injection attacks by removing the parameter values of SQL query,» en *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, 2018, págs. 736-741. DOI: 10.1109/ICISC.2018.8398896.

[42] K. Ross, M. Moh, T.-S. Moh y J. Yao, «Multi-Source Data Analysis and Evaluation of Machine Learning Techniques for SQL Injection Detection,» en *Proceedings of the ACMSE 2018 Conference*, ép. ACMSE '18, Richmond, Kentucky: Association for Computing Machinery, 2018, ISBN: 9781450356961. DOI: 10.1145/3190645.3190670. dirección: https://doi.org/10.1145/3190645.3190670.

[43] K. N. Durai, R. Subha y A. Haldorai, «A Novel Method to Detect and Prevent SQLIA Using Ontology to Cloud Web Security,» *Wireless Personal Communications*, vol. 117, n.º 4, págs. 2995-3014, 2021.

[44] J. O. Atoum y A. J. Qaralleh, «A hybrid technique for SQL injection attacks detection and prevention,» *International Journal of Database Management Systems*, vol. 6, n.º 1, pág. 21, 2014.

[45] N. M. Sheykhkanloo, «A learning-based neural network model for the detection and classification of SQL injection attacks,» *International Journal of Cyber Warfare and Terrorism (IJCWT)*, vol. 7, n.º 2, págs. 16-41, 2017.

[46] S. Bangre, A. Jaiswal et al., «SQL Injection Detection and Prevention Using Input Filter Technique,» *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 1, n.º 2, págs. 145-150, 2012.

[47] M. Hasan, Z. Balbahaith y M. Tarique, «Detection of SQL Injection Attacks: A Machine Learning Approach,» en *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*, 2019, págs. 1-6. DOI: 10.1109/ICECTA48151.2019.8959617.

[48] Z. Xiao, Z. Zhou, W. Yang y C. Deng, «An approach for SQL injection detection based on behavior and response analysis,» en *2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN)*, 2017, págs. 1437-1442. DOI: 10.1109/ICCSN.2017.8230346.

[49] D. Kar, K. Agarwal, A. K. Sahoo y S. Panigrahi, «Detection of SQL injection attacks using Hidden Markov Model,» en *2016 IEEE International Conference on Engineering and Technology (ICETECH)*, 2016, págs. 1-6. DOI: 10.1109/ICETECH.2016.7569180.

[50] L. Yan, X. Li, R. Feng, Z. Feng y J. Hu, «Detection Method of the Second-Order SQL Injection in Web Applications,» en *Proceedings of the Third International Workshop on Structured Object-Oriented Formal Language and Method - Volume 8332*, Berlin, Heidelberg: Springer-Verlag, 2013, págs. 154-165, ISBN: 9783319049144. DOI: 10.

1007/978-3-319-04915-1_11. dirección: https://doi.org/10.1007/978-3-319-04915-1_11.

[51] Z. C. S. S. Hlaing y M. Khaing, «A Detection and Prevention Technique on SQL Injection Attacks,» en *2020 IEEE Conference on Computer Applications(ICCA)*, 2020, págs. 1-6. DOI: 10.1109/ICCA49400.2020.9022833.

[52] P. Li, L. Liu, J. Xu et al., «Application of Hidden Markov Model in SQL Injection Detection,» en *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2, 2017, págs. 578-583. DOI: 10.1109/COMPSAC.2017.64.

[53] T. Oosawa y T. Matsuda, «SQL injection attack detection method using the approximation function of zeta distribution,» en *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2014, págs. 819-824. DOI: 10.1109/SMC.2014.6974012.

[54] K. Wang e Y. Hou, «Detection method of SQL injection attack in cloud computing environment,» en *2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 2016, págs. 487-493. DOI: 10.1109/IMCEC.2016.7867260.

[55] Y.-C. Chung, M.-C. Wu, Y.-C. Chen y W.-K. Chang, «A Hot Query Bank approach to improve detection performance against SQL injection attacks,» *Computers & Security*, vol. 31, n.º 2, págs. 233-248, 2012, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2011.11.007. dirección: https://www.sciencedirect.com/science/article/pii/S016740481100143X.

[56] P. Kumar, «The multi-tier architecture for developing secure website with detection and prevention of sql-injection attacks,» *International Journal of Computer Applications*, vol. 62, n.º 9, 2013.

[57] D. Chen, Q. Yan, C. Wu y J. Zhao, «SQL Injection Attack Detection and Prevention Techniques Using Deep Learning,» *Journal of Physics: Conference Series*, vol. 1757, n.º 1, pág. 012 055, ene. de 2021. DOI: 10.1088/1742-6596/1757/1/012055. dirección: https://doi.org/10.1088/1742-6596/1757/1/012055.

[58] G. Singh, D. Kant, U. Gangwar y A. P. Singh, «Sql injection detection and correction using machine learning techniques,» en *Emerging ICT for Bridging the Future-Proceedings of the 49th Annual Convention of the Computer Society of India (CSI) Volume 1*, Springer, 2015, págs. 435-442.

[59] C.-c. Shi, T. Zhang, Y. Yu y W. Lin, «A new approach for SQL-injection detection,» en *Instrumentation, Measurement, Circuits and Systems*, Springer, 2012, págs. 245-254.

[60] R. M. Nadeem, R. M. Saleem, R. Bashir y S. Habib, «Detection and prevention of SQL injection attack by dynamic analyzer and testing model,» *International Journal of Advanced Computer Science and Applications*, vol. 8, n.º 8, págs. 209-214, 2017.

[61] L. Xiao, S. Matsumoto, T. Ishikawa y K. Sakurai, «SQL Injection Attack Detection Method Using Expectation Criterion,» en *2016 Fourth International Symposium on Computing and Networking (CANDAR)*, 2016, págs. 649-654. DOI: 10.1109/CANDAR.2016.0116.

[62] R. M. Nadeem, R. M. Saleem, R. Bashir y S. Habib, «Detection and Prevention of SQL Injection Attack by Dynamic Analyzer and Testing Model,» *International Journal of Advanced Computer Science and Applications*, vol. 8, n.º 8, 2017. DOI: 10.14569/IJACSA.2017.080827. dirección: http://dx.doi.org/10.14569/IJACSA.2017.080827.

[63] M. S. Aliero e I. Ghani, «A component based SQL injection vulnerability detection tool,» en *2015 9th Malaysian Software Engineering Conference (MySEC)*, 2015, págs. 224-229. DOI: 10.1109/MySEC.2015.7475225.

[64] H. Zhang, B. Zhao, H. Yuan, J. Zhao, X. Yan y F. Li, «SQL Injection Detection Based on Deep Belief Network,» en *Proceedings of the 3rd International Conference on Computer Science and Application Engineering*, ép. CSAE 2019, Sanya, China: Association for Computing Machinery, 2019, ISBN: 9781450362948. DOI: 10.1145/3331453.3361280. dirección: https://doi.org/10.1145/3331453.3361280.

[65] G. Bafghi, «A Simple and Fast Technique for Detection and Prevention of SQL Injection Attacks (SQLIAs),» *International Journal of Security and Its Applications*, vol. 7, n.º 5, págs. 53-66, 2013.

[66] Sangeeta, S. Nagasundari y P. B. Honnavali, «SQL Injection Attack Detection using ResNet,» en *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2019, págs. 1-7. DOI: 10.1109/ICCCNT45670.2019.8944874.

[67] T.-Y. Wu, J.-S. Pan, C.-M. Chen y C.-W. Lin, «Towards SQL injection attacks detection mechanism using parse tree,» en *Genetic and Evolutionary Computing*, Springer, 2015, págs. 371-380.

[68] L. Saoudi, K. Adi e Y. Boudraa, «A rejection-based approach for detecting SQL injection vulnerabilities in web applications,» en *International Symposium on Foundations and Practice of Security*, Springer, 2019, págs. 379-386.

[69] R. Kozik, M. Choraś y W. Hołubowicz, «Hardening Web Applications against SQL Injection Attacks Using Anomaly Detection Approach,» en *Image Processing & Communications Challenges 6*, Springer, 2015, págs. 285-292.

[70] N. M. Sheykhkanloo, «A Pattern Recognition Neural Network Model for Detection and Classification of SQL Injection Attacks,» *International Journal of Computer and Information Engineering*, vol. 9, n.º 6, págs. 1436-1446, 2015, ISSN: eISSN: 1307-6892. dirección: https://publications.waset.org/vol/102.

[71] O. Hubskyi, T. Babenko, L. Myrutenko y O. Oksiiuk, «Detection of sql injection attack using neural networks,» en *International scientific-practical conference*, Springer, 2020, págs. 277-286.

[72] D. E. Nofal y A. A. Amer, «SQL Injection Attacks Detection and Prevention Based on Neuro-Fuzzy Technique,» en *International Conference on Advanced Intelligent Systems and Informatics*, Springer, 2019, págs. 722-738.

[73] N. Gandhi, J. Patel, R. Sisodiya, N. Doshi y S. Mishra, «A CNN-BiLSTM based Approach for Detection of SQL Injection Attacks,» en *2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, 2021, págs. 378-383. DOI: 10.1109/ICCIKE51210.2021.9410675.

[74] R. A. Dalimunthe y S. Sahren, «Intrusion detection system and modsecurity for handling sql injection attacks,» en *International Conference on Social, Sciences and Information Technology*, vol. 1, 2020, págs. 187-194.

[75] A. O. Agbakwuru y D. O. Njoku, «SQL Injection Attack on Web Base Application: Vulnerability Assessments and Detection Technique,» *International Research Journal of Engineering and Technology*, vol. 8, n.º 3, págs. 243-252, 2021.

# 8 ANEXOS

En esta sección se presentan las tablas obtenidas de la revisión sistemática de la literatura detallada en el Capítulo 4

# A ARTÍCULOS SELECCIONADOS PARA LA REVISIÓN

**Tabla 8.1:** Artículos seleccionados para la revisión

| ID | Ref. | Citas | Año | Técnica | Clasificación | Algoritmo(s) | Dataset | Tipos de SQLIA |
|---|---|---|---|---|---|---|---|---|
| 1 | [26] | 166 | 2012 | Técnica basada en Removing Sql query attribute values | Machine learning | — | — SQL 5.0 | — |
| 2 | [27] | 75 | 2013 | Técnica basada en idMas-SQL | Machine learning | CNN | Base de datos SQL 5.0 | Todos |
| 3 | [28] | 66 | 2014 | Técnica basada en Data-mining based Sql injection attacj deteccion in internal query tress | Machine learning | CNN | PostgreSql v 9.2.3 | Todos |
| 4 | [29] | 52 | 2012 | Técnica basada en Detection of SQL injection attacks | Machine learning | CNN | PostgreSql v 9.2.3 | Todos |
| 5 | [30] | 43 | 2016 | Técnica basada en análisis estático | Prácticas de codificación segura | N/A | N/A | Todos |

| ID | Ref. | Citas | Año | Técnica | Clasificación | Algoritmo(s) | Dataset | Tipos de SQLIA |
|---|---|---|---|---|---|---|---|---|
| 6 | [31] | 42 | 2016 | Técnica basada en basada ASCII based string matching | Técnicas híbridas | String Matching | Generador de claves basado en texto, graficos SQL utilizando FMS | Todos |
| 7 | [32] | 38 | 2020 | Técnica basada en Machine learning predictive analytics to SQL injection attac | Machine Learning | SVM | N/A | Todos |
| 8 | [33] | 28 | 2020 | Técnica basada en Artificial neural network | Machine Learning | CNN | Generador de URL | Todos |
| 9 | [34] | 27 | 2019 | Técnica basada en Adaptive Deep Forest | Machine Learning | AdaBoost | Exploit-Db y wooyun-Db | Todos |
| 10 | [35] | 27 | 2019 | Técnica basada en Neural networks | Machine Learning | CNN | Generador de URL | Todos |
| 11 | [36] | 26 | 2015 | Técnica basada en SQLiDDs | Técnicas taint-based | K-meas | N/A | Todos |
| 12 | [37] | 24 | 2017 | Técnica basada en análisis estático y dinámico | Técnicas Hibridas | N/A | N/A | Todos |
| 13 | [38] | 24 | 2019 | Técnica basada en Elastic pooling - CNN | Machine learning | CNN | Registros de web reales en entorno de producción | Todos |

| ID | Ref. | Citas | Año | Técnica | Clasificación | Algoritmo(s) | Dataset | Tipos de SQLIA |
|---|---|---|---|---|---|---|---|---|
| 14 | [39] | 23 | 2012 | Técnica basada en Progam tracing and machine learning | Técnicas híbridas | N/A | N/A | Todos |
| 15 | [40] | 23 | 2019 | Técnica basada en Diava | Modificación de sentencias | N/A | Almacenamiento en la nube, análisis de trafico de red | Todos |
| 16 | [41] | 22 | 2019 | Técnica basada en Removing the parameter values of SQL query | Modificación de sentencias | N/A | Aplicaciones web vulnerables | Todos |
| 17 | [42] | 22 | 2019 | Técnica basada en Machine learning for SQL injection detection | Modificación de sentencias | N/A | Aplicaciones web vulnerables | Todos |
| 18 | [43] | 17 | 2020 | Técnica basada en Ontology to cloud web security | Practicas de codificación segura | N/A | Información guardada en la nube | Todos |
| 19 | [44] | 16 | 2014 | Técnica basada en análisis estático y dinámico | Técnicas híbridas | N/A | N/A | Todos |
| 20 | [45] | 15 | 2017 | Técnica basada en redes neuronales | Machine learning | CNN | Generador y clasificador de URLs | Todos |

| ID | Ref. | Citas | Año | Técnica | Clasificación | Algoritmo(s) | Dataset | Tipos de SQLIA |
|---|---|---|---|---|---|---|---|---|
| 21 | [46] | 14 | 2012 | Técnicas basadas en filtrado de atributos | Prácticas de códificación | N/A | N/A | Todos |
| 22 | [47] | 14 | 2019 | Técnica basada en clasificadores de machine learning | Machine learning | 23 clasificadores | Ejemplos de sentencias SQL de W3School (benignos) y sentencias del OWASP SecLists Project | Todos |
| 23 | [48] | 13 | 2017 | Técnica basada en analisis de comportamiento | Otras técnicas | N/A | N/A | Solo los 6 tipos de SQLIA más básicos |
| 24 | [49] | 13 | 2016 | Técnica basada en el modelo oculto de Markov | Técnicas basadas en modelos probabilísticos | HMM | Datos reales de una configuración de prueba | Todos |
| 25 | [50] | 13 | 2014 | Tecnica basada en análisis estático y dinamico | Técnicas híbridas | N/A | N/A | Todos |
| 26 | [51] | 12 | 2020 | Técnica basada en creacion de lexicos y tokenizacion de cadenas | Árbol de análisis gramatical | N/A | N/A | Todos |

| ID | Ref. | Citas | Año | Técnica | Clasificación | Algoritmo(s) | Dataset | Tipos de SQLIA |
|---|---|---|---|---|---|---|---|---|
| 27 | [52] | 12 | 2017 | Técnica basada en el modelo oculto de Markov | Técnicas basadas en modelos probabilísticos | HMM | Datos reales de una configuración de prueba | Todos |
| 28 | [53] | 12 | 2014 | Técnica basada en la funcion de distribución Zeta | Técnicas basadas en modelos probabilísticos | N/A | Datos de ejemplo | Todoss |
| 29 | [54] | 12 | 2016 | Técnica basada en creación de reglas | Técnicas taint-based | N/A | N/A | Todos |
| 30 | [55] | 12 | 2012 | Técnica asado en creacioón de banco de consultas | Árbol de análisis gramatical | N/A | N/A | Todos |
| 31 | [56] | 11 | 2013 | Tecnica basada en análisis estático y dinamico | Técnicas híbridas | N/A | N/A | Todos |
| 32 | [57] | 11 | 2021 | Técnica basada en deep learning | Machine learning | CNN y MLP | Datos de ejemplo obtenidos de internet | Todos |
| 33 | [58] | 11 | 2015 | Técnica basada en machine learning | Machine learning | K-means | No especifica | Todos |
| 34 | [59] | 10 | 2015 | Técnica basada en creación de librerias de conocimiento | Árbol de análisis gramatical | N/A | N/A | Todos |

| ID | Ref. | Citas | Año | Técnica | Clasificación | Algoritmo(s) | Dataset | Tipos de SQLIA |
|---|---|---|---|---|---|---|---|---|
| 35 | [60] | 9 | 2017 | Técnica basada en Dynamic Analyzer and Testing Model | Taint-based Technique | N/A | datos reales de una configuración de prueba | Todos |
| 36 | [61] | 9 | 2016 | Técnica basada en Expectation Criterion | Probabilístico | N/A | datos de ejemplo | Todos |
| 37 | [62] | 9 | 2013 | Técnica basada en la detección del lado del cliente utilizando cuatro métricas de entropía condicional | Practicas de Codificación Segura | N/A | datos reales de una configuración de prueba | Todos |
| 38 | [63] | 8 | 2015 | Técnica basada en herramientas de detección de vulnerabilidades basada en Rastreo de la web, análisis de los ataques y elaboración de informes, análisis de los ataques y elaboración de informes | Análisis Estático | N/A | datos reales de una configuración de prueba | Todos |
| 39 | [64] | 7 | 2019 | Técnica basada en Deep Belief Network | Machine Learning | Deep Belief Network (DBN) | datos de ejemplo | Todos |

| ID | Ref. | Citas | Año | Técnica | Clasificación | Algoritmo(s) | Dataset | Tipos de SQLIA |
|---|---|---|---|---|---|---|---|---|
| 40 | [65] | 6 | 2013 | Técnica basada en modelos de consulta válidos obtenidos de una aplicación web | Análisis Estático y Dinámico | N/A | datos reales de una configuración de prueba | Todos |
| 41 | [66] | 6 | 2019 | Técnica basada en ResNet | Machine Learning | ResNet | uso de una herramienta (no especifica) y datos de internet | Todos |
| 42 | [67] | 6 | 2015 | Técnica basada en Dynamic SQLIA Detection (DSD) | Parse Tree | Dynamic SQLIA Detection (DSD) | datos reales de una configuración de prueba | Todos |
| 43 | [68] | 5 | 2019 | Técnica basada en rechazo | Análisis Estático | N/A | datos reales de una configuración de prueba | Todos |
| 44 | [69] | 4 | 2015 | Técnica basada en anomalias de rechazo | Analisis Estático | Linear Discriminant Analysis(LDA) | datos generados por un servicio HTTP | todos |
| 45 | [70] | 4 | 2015 | Técnica basada en una Red Neuronal | Machine Learning | Red Neuronal | datos de ejemplo | Todos |
| 46 | [71] | 4 | 2020 | Técnica basada en Artificial Neural Networks | Machine Learning | Artificial Neural Networks | datos obtenidos de sitios de internet | Todos |

| ID | Ref. | Citas | Año | Técnica | Clasificación | Algoritmo(s) | Dataset | Tipos de SQLIA |
|---|---|---|---|---|---|---|---|---|
| 47 | [72] | 3 | 2019 | Técnica basada en Neuro-Fuzzy | Machine Learning | Adaptive Neuro-Fuzzy Inference System (ANFIS) / Fuzzy C-Means (FCM) / ScaledConjugate Gradient (SCG)" | datos reales de una configuración de prueba | Todos |
| 48 | [73] | 1 | 2021 | Técnica basada en CNN-BiLSTM | Machine Learning | CNN-BiLSTM | datos obtenidos de sitios de internet | Todos |
| 49 | [74] | 1 | 2020 | Técnica basada en un Sistema de Detección de Intrusos y Cortafuegos(ModSecurity) | Sistema de detección de intrusos | N/A | Datos de ejemplo | Todos |
| 50 | [75] | 1 | 2021 | Técnica basada en fuzzy rule-based classification system (FRBCS) | Machine Learning | Algoritmo Genético Simple | datos reales de una configuración de prueba | Todos |

# B CRITERIOS DE CALIFICACIÓN PARA LAS PREGUNTAS DE EVALUACIÓN DE CALIDAD

**Tabla 8.2:** Criterios de calificación para las preguntas de evaluación de calidad

| **Pregunta** | Puntajes posibles | | |
|---|---|---|---|
| QA1 | 0: Si no se indica en el abstract o en la introducción de manera la técnica a desarrollar en el artículo. en el artículo. | 0.5: Si se indica en el abstract o en la introducción de manera implícita la técnica a desarrollar en el artículo. | 1: Si se indica en el abstract o en la introducción de manera explícita la técnica a desarrollar |
| QA2 | 0: Si no se describe ningún tema que dé contexto de la investigación en el artículo. | 0.5: si se describe un solo tema que dé contexto de la investigación en el artículo. | 1: Si se describen al menos dos temas diferentes que den contexto de la investigación en el artículo. |
| QA3 | 0: Si no existe ningún indicio o mención a trabajos relacionados dentro del artículo. | 0.5: Si existe un indicio o breve referencia a trabajos relacionados en el artículo pero no se los describe de manera detallada. | 1: Si existe una sección de trabajos relacionados en el artículo o se describen de manera detallada algunos trabajos relacionados. |
| QA4 | 0: Si no existe una descripción de la arquitectura o metodología propuesta en el artículo. | 0.5: Si existe una descripción inconsistente, incompleta o ambigua de la arquitectura o metodología propuesta en el artículo. | 1: Si existe una descripción clara, completa y detallada de la arquitectura o metodología propuesta en el artículo |
| QA5 | 0: Si no se muestran ni la evaluación ni los resultados de la metodología o técnica propuesta en el artículo. | 0.5: Si se solo se evalúa la metodología o técnica propuesta en el artículo sin mostrar los resultados o solo se muestran los resultados de la metodología o técnica propuesta sin mostrar la evaluación. | 1: Si se evalúa de manera detallada la metodología o técnica propuesta en el artículo y se muestran los resultados de dicha evaluación. |

| **Pregunta** | Puntajes posibles | | |
|---|---|---|---|
| QA6 | 0: Si la conclusión de la investigación difiere completamente de los objetivos propuestos en el artículo o si no existen conclusiones en el artículo. | 0.5: Si la conclusión de la investigación difiere un poco de los objetivos propuestos en el artículo. | 1: Si la conclusión de la investigación muestra concordancia con los objetivos propuestos en el artículo. |
| QA7 | 0: Si no se indican trabajos futuros en el artículo. | 0.5: Si los trabajos futuros no se detallan claramente o no se relacionan directamente con la investigación principal del artículo. | 1: si los trabajos futuros se detallan claramente y se relacionan directamente con la investigación principal del artículo. |

# C PUNTAJE DE LA EVALUACIÓN DE CALIDAD DE LOS ARTÍCULOS ANALIZADOS

**Tabla 8.3:** Puntaje de la evaluación de calidad de los artículos analizados.

| # | QA1 | QA2 | QA3 | QA4 | QA5 | QA6 | QA7 | Total |
|---|---|---|---|---|---|---|---|---|
| [26] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [27] | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 6 |
| [28] | 1 | 1 | 1 | 1 | 1 | 0.5 | 0 | 5.5 |
| [29] | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 6 |
| [30] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [31] | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 6 |
| [32] | 1 | 0.5 | 0.5 | 1 | 1 | 0.5 | 1 | 5.5 |
| [33] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [34] | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 6 |
| [35] | 1 | 1 | 1 | 0 | 1 | 1 | 0.5 | 5.5 |
| [36] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [37] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [38] | 1 | 1 | 1 | 1 | 1 | 1 | 0.5 | 6.5 |
| [39] | 1 | 1 | 1 | 1 | 1 | 1 | 0.5 | 6.5 |
| [40] | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 6 |
| [41] | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 5 |
| [42] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [43] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [44] | 1 | 1 | 0.5 | 1 | 1 | 1 | 1 | 6.5 |
| [45] | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 6 |
| [46] | 1 | 1 | 0.5 | 1 | 1 | 1 | 1 | 6.5 |
| [47] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [48] | 0.5 | 1 | 1 | 1 | 1 | 1 | 1 | 6.5 |
| [49] | 1 | 0.5 | 1 | 1 | 1 | 1 | 1 | 6.5 |
| [50] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [51] | 1 | 1 | 0 | 1 | 1 | 0.5 | 0 | 4.5 |
| [52] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [53] | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 6 |
| [54] | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 5 |
| [55] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [56] | 1 | 1 | 1 | 1 | 1 | 0.5 | 1 | 6.5 |

| # | QA1 | QA2 | QA3 | QA4 | QA5 | QA6 | QA7 | Total |
|---|---|---|---|---|---|---|---|---|
| [57] | 1 | 1 | 0.5 | 1 | 1 | 1 | 1 | 6.5 |
| [58] | 1 | 1 | 0.5 | 1 | 1 | 0.5 | 1 | 6 |
| [59] | 1 | 1 | 0 | 1 | 1 | 0.5 | 1 | 5.5 |
| [60] | 1 | 1 | 0.5 | 1 | 1 | 1 | 0.5 | 6 |
| [61] | 1 | 0,5 | 1 | 0,5 | 1 | 1 | 1 | 6 |
| [62] | 1 | 0 | 0 | 0.5 | 0 | 0.5 | 0.5 | 2.5 |
| [63] | 1 | 0.5 | 1 | 1 | 1 | 1 | 1 | 6.5 |
| [64] | 1 | 1 | 1 | 0.5 | 1 | 1 | 0 | 5.5 |
| [65] | 1 | 1 | 1 | 1 | 1 | 1 | 0.5 | 6.5 |
| [66] | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 6 |
| [67] | 1 | 0.5 | 0.5 | 1 | 1 | 1 | 0.5 | 5.5 |
| [68] | 0.5 | 0.5 | 0 | 0.5 | 1 | 1 | 1 | 4.5 |
| [70] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [71] | 1 | 0.5 | 0.5 | 1 | 1 | 1 | 0 | 5 |
| [72] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [73] | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 |
| [74] | 1 | 0.5 | 0 | 1 | 1 | 0 | 0 | 3.5 |
| [75] | 1 | 1 | 1 | 1 | 1 | 1 | 0.5 | 6.5 |