**INFORME DE REQUISITOS DE SOFTWARE (SRS)**

**Título del Proyecto: Sistema de Gestión de Delitos y Antecedentes Ciudadanos - "ENTERPRISE"**

**1. Introducción**

Este documento detalla los requerimientos del sistema solicitado por INTERPOLICE, con base en la entrevista realizada con el Capitán Smith. El sistema permitirá el registro, gestión, consulta y generación de reportes de antecedentes penales y delitos cometidos por ciudadanos en distintos planetas.

**2. Alcance del sistema**

El sistema permitirá:

* Acceso por roles (Administrador, General, Policía, secretaria de Juzgado).
* Registro y consulta de delitos.
* Impresión y descarga de reportes.
* Consulta de antecedentes.
* Generación automática de amonestaciones.
* Exportación de datos a Excel.
* Interfaz adaptable a distintos dispositivos (Tablet, teléfonos, terminales IBM).

**2.1 Definiciones, acrónimos y abreviaturas**

* **QR**: Código de respuesta rápida.
* **IBM**: Terminal de escritorio de gran tamaño.
* **SRS**: Software Requirements Specification.
* **IEEE 830**: Estándar para documentación de requerimientos de software.

### ****2.2 Vista General de Componentes****

* **Servicio de Autenticación y Autorización:** Gestiona el registro, inicio de sesión y control de accesos diferenciados por roles (Administrador, General, Policía, secretaria). Controla permisos para acceso a funciones específicas del sistema.
* **Servicio de Gestión de Ciudadanos:** Administra la información personal de los ciudadanos, incluyendo datos universales, códigos QR y su historial de antecedentes.
* **Servicio de Registro de Delitos:** Permite la creación, actualización y almacenamiento de delitos asociados a ciudadanos, contemplando múltiples delitos en una misma hora y lugar, con detalles como fecha, hora, ubicación y descripción.
* **Servicio de Amonestaciones:** Maneja la generación automática de amonestaciones para delitos menores, aplica reglas de reincidencia y sanciones correspondientes, como multas, cursos y trabajo cívico o cárcel.
* **Servicio de Consulta de Antecedentes:** Facilita a los policías la consulta rápida y sencilla del historial delictivo de ciudadanos, con opciones de lectura de código QR o ingreso manual del código universal.
* **Servicio de Reportes y Estadísticas:** Genera reportes gráficos y tabulares sobre delitos, ciudadanos con mayores incidencias, planetas con más delitos, etc., y permite exportar información a formatos compatibles con Excel.
* **API Gateway:** Punto de entrada unificado para todas las solicitudes del cliente (tablets, smartphones, terminales IBM), que enruta las peticiones hacia los microservicios correspondientes de forma segura y eficiente.

### ****2.2 Tecnologías Clave****

La plataforma INTERPOLICE – ENTERPRISE será desarrollada utilizando tecnologías modernas que aseguren rendimiento, seguridad, compatibilidad multiplataforma y facilidad de mantenimiento en entornos diversos como tablets transparentes, dispositivos antiguos y terminales de escritorio. Dado que la solución debe responder a criterios funcionales exigidos por el Capitán Smith y altos mandos como el comandante Jones, se adopta una arquitectura **orientada a servicios (SOA)** con enfoque en APIs RESTful.

Las tecnologías seleccionadas son las siguientes:

* **Backend:** Para el desarrollo de los servicios que conforman la API RESTful, se utilizará **Node.js** con enfoque en **Vanilla JavaScript** (programación imperativa), cumpliendo la restricción establecida de no utilizar programación orientada a objetos. Cada módulo seguirá el patrón arquitectónico **Modelo-Vista-Controlador (MVC)** internamente para garantizar separación de responsabilidades y claridad del código.
* **Frontend:** La aplicación cliente será desarrollada como una **aplicación web pura**, también utilizando **Vanilla JavaScript**, asegurando compatibilidad con tablets transparentes (como la Samsung SXVB45Q) y teléfonos antiguos como el iPhone 47 XPLUS SIGMA. La interfaz será intuitiva y de fácil uso, especialmente pensada para los policías en campo.
* **Base de Datos:** Se empleará **SQLite**. Su portabilidad y bajo consumo de recursos permiten integrarla fácilmente en los dispositivos existentes sin requerir servidores robustos.
* **Orquestación:** No se utilizarán contenedores ni sistemas de orquestación como Docker o Kubernetes. La solución será instalada directamente en los entornos objetivo y ejecutada en servicios o procesos nativos del sistema operativo.
* **API Gateway:** Todos los accesos desde el cliente hacia los servicios serán gestionados a través de un **API Gateway**, que controlará el enrutamiento hacia los microservicios correspondientes, validación de tokens y aplicación de políticas de seguridad.
* **Seguridad y Autenticación:** El sistema implementará autenticación por roles (Administrador, Policía, General, Secretaria) y **autenticación de dos factores (2FA)**. Los recursos y endpoints estarán protegidos mediante **tokens de autenticación** seguros, cumpliendo los estándares modernos de seguridad.
* **Colas de Mensajes:** Para funcionalidades asíncronas como notificaciones automáticas (avisos de reincidencia, generación de reportes, alertas para el comandante Jones), se utilizará **Kafka**, permitiendo un flujo eficiente y desacoplado de eventos.
* **Exportación de Datos:** Para cumplir con la preferencia del comandante Jones por el software EXCEL, se incorporará un módulo de exportación de reportes en formato **.xlsx**, lo que permitirá analizar los datos fuera del sistema según sus necesidades.
* **Control de Versiones:** Todo el código del sistema, incluyendo pruebas y configuraciones, será gestionado mediante **Git** y alojado en **GitHub**, permitiendo trazabilidad, trabajo colaborativo y control de versiones.
* **Pruebas:** Cada módulo del sistema incluirá **casos de prueba automatizados**, utilizando herramientas como **Jest** (para backend y lógica), **Selenium** (para pruebas de interfaz gráfica) y **Postman** (para validación de endpoints y APIs RESTful).

## ****3. Diseño de Bases de Datos****

### ****3.1 Base de Datos Relacional (SQLite)****

Se utilizará **SQLite** como sistema de gestión de base de datos relacional por su portabilidad, bajo consumo de recursos y compatibilidad con los dispositivos existentes, incluidos los terminales antiguos y tablets modernas. Esta base de datos almacenará información estructurada que requiere transacciones seguras, relaciones fuertes y normalización.

A continuación, se detallan las principales tablas:

#### **Tabla Usuarios**

| **Campo** | **Tipo de dato** | **Descripción** |
| --- | --- | --- |
| id\_usuario | INTEGER (PK, AUTOINCREMENT) | Identificador único del usuario |
| nombre | TEXT | Nombre completo del usuario |
| email | TEXT UNIQUE | Correo electrónico |
| password\_hash | TEXT | Contraseña en formato hash |
| rol | TEXT CHECK (rol IN 'administrador', 'general', 'policía', 'secretaria') | Rol del usuario |
| fecha\_creacion | TIMESTAMP DEFAULT CURRENT\_TIMESTAMP | Fecha de creación del usuario |

#### **Tabla Ciudadanos**

| **Campo** | **Tipo de dato** | **Descripción** |
| --- | --- | --- |
| id\_ciudadano | INTEGER (PK, AUTOINCREMENT) | Identificador del ciudadano |
| nombre\_completo | TEXT | Nombre del ciudadano |
| codigo\_qr | TEXT UNIQUE | Código QR o código universal |
| planeta\_origen | TEXT | Planeta de origen del ciudadano |

#### **Tabla Delitos**

| **Campo** | **Tipo de dato** | **Descripción** |
| --- | --- | --- |
| id\_delito | INTEGER (PK, AUTOINCREMENT) | Identificador del delito |
| id\_ciudadano | INTEGER (FK → Ciudadanos.id\_ciudadano) | Ciudadano al que se asocia el delito |
| fecha | DATE | Fecha en que ocurrió el delito |
| hora | TIME | Hora del incidente |
| lugar | TEXT | Ubicación donde ocurrió |
| descripcion | TEXT | Descripción del hecho |
| registrado\_por | INTEGER (FK → Usuarios.id\_usuario) | Usuario que registró el delito |

#### **Tabla TiposDelito**

| **Campo** | **Tipo de dato** | **Descripción** |
| --- | --- | --- |
| id\_tipo | INTEGER (PK) | Identificador del tipo de delito |
| nombre | TEXT | Nombre del delito (ej. Conducir ebrio) |
| gravedad | TEXT CHECK (gravedad IN 'menor', 'grave', 'muy grave') | Nivel de gravedad del delito |

#### **Tabla Delito\_TipoDelito (Relación N a N)**

| **Campo** | **Tipo de dato** | **Descripción** |
| --- | --- | --- |
| id\_delito | INTEGER (FK → Delitos.id\_delito) | Referencia al delito |
| id\_tipo | INTEGER (FK → TiposDelito.id\_tipo) | Tipo de delito asociado |
| **PK conjunta:** id\_delito, id\_tipo |  |  |

#### **Tabla Amonestaciones**

| **Campo** | **Tipo de dato** | **Descripción** |
| --- | --- | --- |
| id\_amonestacion | INTEGER (PK, AUTOINCREMENT) | Identificador de la amonestación |
| id\_ciudadano | INTEGER (FK → Ciudadanos.id\_ciudadano) | Ciudadano amonestado |
| fecha | DATE | Fecha de la amonestación |
| monto | INTEGER | Valor en dólares (por defecto: $400) |
| descripcion | TEXT | Motivo de la amonestación |
| estado | TEXT CHECK (estado IN 'primera', 'segunda', 'tercera') | Nivel de reincidencia |
| sancion\_aplicada | TEXT | Descripción de sanción (curso, trabajo cívico, cárcel) |

#### **Tabla ConsultasAntecedentes**

| **Campo** | **Tipo de dato** | **Descripción** |
| --- | --- | --- |
| id\_consulta | INTEGER (PK, AUTOINCREMENT) | Identificador de la consulta |
| id\_usuario | INTEGER (FK → Usuarios.id\_usuario) | Policía que consultó |
| id\_ciudadano | INTEGER (FK → Ciudadanos.id\_ciudadano) | Ciudadano consultado |
| fecha\_hora | TIMESTAMP DEFAULT CURRENT\_TIMESTAMP | Fecha y hora de la consulta |
| dispositivo | TEXT | Dispositivo usado (tablet, iPhone, etc.) |

## ****4. Diseño de la Interfaz de Usuario (UI/UX)****

El diseño de la interfaz de usuario se enfocará en ofrecer una experiencia clara, rápida y funcional para todos los perfiles del sistema INTERPOLICE – ENTERPRISE. Dado que los usuarios incluyen personal operativo como policías en campo y administrativos como secretarias y altos mandos, se prioriza la **simplicidad, accesibilidad y compatibilidad entre dispositivos** (tablets transparentes, terminales IBM, y móviles antiguos como el iPhone 47 XPLUS SIGMA).

Se aplicarán principios de diseño de interfaz **limpia, responsiva y centrada en tareas frecuentes**, con énfasis en accesibilidad y mínima curva de aprendizaje. No se utilizarán frameworks pesados para mantener compatibilidad con navegadores menos modernos, cumpliendo además con el requisito de usar **Vanilla JavaScript**.

### ****4.1. Componentes Clave****

#### • **Inicio del Sistema / Dashboard de Usuario**

* Visualización rápida de acciones disponibles según rol: Registrar delito, consultar antecedentes, imprimir amonestaciones, generar reportes.
* Bienvenida personalizada y registro de la última actividad.

#### • **Consulta de Antecedentes**

* Campo para escanear **código QR** o ingresar manualmente el **código universal** del ciudadano.
* Visualización en pantalla del historial delictivo con opción de **imprimir directamente** desde la interfaz.
* Compatible con pantallas pequeñas o grandes.

#### • **Registro de Delitos**

* Formulario con:
  + Fecha, hora, lugar del delito.
  + Descripción libre del hecho.
  + Asociación de **uno o varios tipos de delitos** en un solo evento.
* Registro automatizado del usuario (policía o secretaria) que ingresó la información.

#### • **Generación de Amonestaciones Automáticas**

* Activada desde el registro de un delito **menor**.
* Emisión instantánea de sanción por $400 y curso obligatorio.
* Gestión progresiva: segunda y tercera reincidencia con sanciones incrementadas.
* Botón de “Imprimir Amonestación” desde el mismo formulario.

#### • **Gestión de Usuarios del Sistema**

* Listado y filtros de búsqueda.
* Creación de nuevos usuarios y asignación de **roles por jerarquía**: Administrador, Policía, secretaria, General.
* Autenticación en dos pasos (2FA).

#### • **Reportes y Estadísticas**

* Gráficas interactivas con:
  + Delitos más frecuentes.
  + Ciudadanos con mayor reincidencia.
  + Planetas con más registros delictivos.
* Opción de **descargar reportes en Excel** (.xlsx) para análisis personalizados por parte del comandante Jones.

### ****4.2. Flujos de Usuario Críticos (Ejemplos)****

#### • **Consulta de Antecedentes por Policía en Campo**

1. Policía inicia sesión desde su dispositivo.
2. Accede a la sección “Consultar antecedentes”.
3. Escanea el **código QR** o digita el código del ciudadano.
4. El sistema muestra el historial delictivo.
5. El policía puede **imprimir el reporte** si lo requiere.

#### • **Registro de un Delito Complejo**

1. Secretaria judicial accede desde su terminal.
2. Entra en la sección “Registrar Delito”.
3. Completa los datos del incidente (fecha, hora, lugar, descripción).
4. Selecciona múltiples delitos cometidos en el mismo hecho.
5. Guarda el registro y el sistema asocia los delitos al ciudadano.

#### **• Emisión Automática de Amonestación por Delito Menor**

1. Policía registra un delito menor (ej. orinar en vía pública).
2. El sistema detecta el tipo de delito y activa una amonestación.
3. Se genera la sanción por $400 y curso de normas cívicas.
4. Policía pulsa “Imprimir” para entregar al ciudadano.

#### • **Exportación de Reportes a Excel**

1. Administrador accede a la sección “Reportes”.
2. Selecciona “Delitos por planeta” y define un rango de fechas.
3. El sistema genera un gráfico interactivo.
4. Da clic en “Exportar a Excel”.
5. Se descarga el archivo .xlsx con los datos para análisis externo.

## 5. Consideraciones de Seguridad

La seguridad es un pilar fundamental en el desarrollo del sistema INTERPOLICE – ENTERPRISE. Se implementarán las siguientes medidas para proteger la información sensible y garantizar el acceso adecuado:

* **Autenticación Basada en Tokens (JWT):** Se utilizarán tokens seguros para autenticar y autorizar las peticiones a la API REST, asegurando que sólo usuarios válidos accedan a los recursos.
* **Autorización Basada en Roles (RBAC):** El control de acceso se realizará según el perfil de usuario (general, policía, secretaria, administrador), limitando el acceso y operaciones según permisos específicos.
* **Cifrado de Contraseñas:** Las contraseñas se almacenarán utilizando algoritmos de hashing seguros (por ejemplo, BCrypt) para evitar vulneraciones en caso de filtración.
* **Validación Estricta de Entradas:** Se aplicarán controles para prevenir ataques comunes como inyección SQL, Cross-Site Scripting (XSS) y otras vulnerabilidades derivadas de datos maliciosos.
* **Comunicación Segura (HTTPS):** Todas las comunicaciones entre cliente y servidor estarán cifradas mediante HTTPS para evitar interceptaciones.
* **Auditoría y Registro de Actividades:** Se mantendrá un registro detallado de acciones críticas para facilitar la detección y análisis de posibles incidentes de seguridad.
* **Políticas de Sesión:** Gestión segura de sesiones, incluyendo expiración, renovación y control de accesos concurrentes.
* **Autenticación de Dos Factores (2FA):** Para reforzar la seguridad en accesos especialmente sensibles, como el usuario administrador.

## 6. Estrategia de Pruebas

Se definirá una estrategia de pruebas exhaustiva para garantizar la calidad, confiabilidad y seguridad del sistema:

* **Pruebas Unitarias:** Se implementarán para cada módulo y función del sistema, especialmente en la API REST y lógica del backend.
* **Pruebas de Integración:** Para verificar la interacción correcta entre los diferentes módulos y servicios, como el registro de delitos y la generación de amonestaciones.
* **Pruebas de Componente (Frontend):** Validación de los componentes de la interfaz web desarrollados en Vanilla JavaScript.
* **Pruebas End-to-End (E2E):** Simulación completa de flujos críticos, como la consulta de antecedentes, registro de delitos y generación de reportes.
* **Pruebas de Rendimiento:** Evaluación del sistema bajo diferentes cargas, garantizando la respuesta rápida en dispositivos variados, incluyendo tablets y móviles antiguos.
* **Pruebas de Seguridad:** Testeo de vulnerabilidades y penetración para asegurar la protección contra ataques comunes.
* **Pruebas de Usabilidad:** Participación de usuarios reales (policías, secretarias, administradores) para validar la facilidad de uso y efectividad de la interfaz.

## 7. Plan de Despliegue y Operaciones

### 7.1. Entornos

* **Desarrollo:** Entorno local para los desarrolladores con herramientas de control de versiones (Git y GitHub).
* **Pruebas/QA:** Entorno que replica las condiciones de producción para pruebas funcionales e integración.
* **Pre-producción/Staging:** Ambiente para validación final antes del despliegue a producción.
* **Producción:** Entorno definitivo donde estará disponible la aplicación para los usuarios finales, considerando las limitaciones de hardware (terminales IBM, tablets transparentes, móviles antiguos).

### 7.2. Herramientas de CI/CD

* **GitHub Actions:** Automatización del ciclo de construcción, pruebas y despliegue continuo para asegurar entregas rápidas y confiables.

### 7.3. Monitorización y Logging

* **Registro Centralizado:** Logs de eventos críticos y errores serán almacenados y revisados periódicamente para mantener la estabilidad y seguridad del sistema.
* **Alertas:** Configuración de notificaciones ante fallos o actividades anómalas para respuesta inmediata.

### 7.4. Escalabilidad

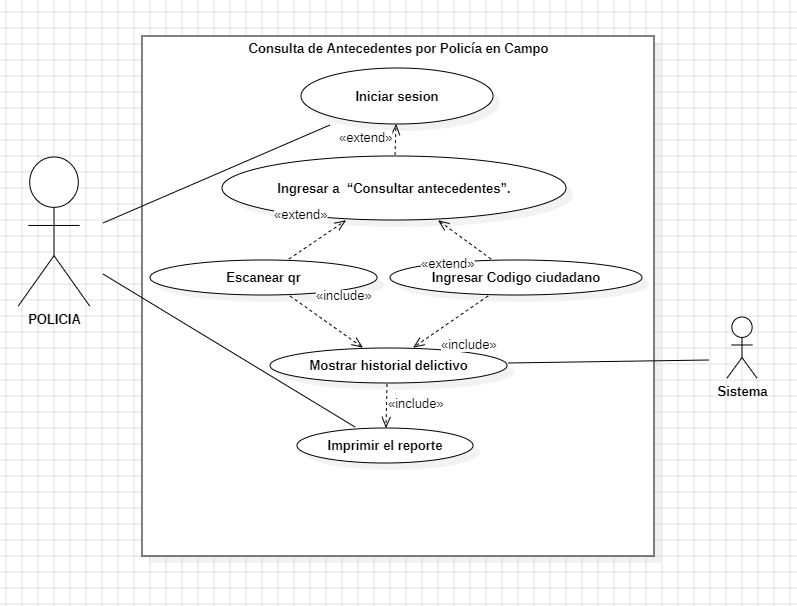
* **Escalado Horizontal:** Se podrá añadir nuevas instancias del sistema para soportar más usuarios si la demanda lo requiere.
* **Escalado Vertical:** Incremento de recursos del servidor para mejorar el rendimiento en caso de necesidades puntuales.

## 8. Conclusiones y Próximos Pasos

Este informe ha presentado la arquitectura, tecnologías y consideraciones clave para el desarrollo del sistema INTERPOLICE – ENTERPRISE, asegurando una solución segura, accesible y alineada a los requerimientos del cliente.

**Los próximos pasos incluyen:**

* Refinar y validar el modelo de datos con el cliente.
* Implementar los módulos de autenticación y autorización con seguridad robusta.
* Desarrollar progresivamente las funcionalidades críticas, comenzando por el registro y consulta de delitos.
* Crear la interfaz web con especial atención en la experiencia de usuario para policías y secretarias.
* Establecer la integración continua con pruebas automatizadas para mantener la calidad.
* Preparar la documentación y capacitación para los usuarios finales.

****