

## Taller de Proyectos en Ingeniería de Sistemas e Informática

### INTEGRANTES:

| Código UC  | Apellidos y Nombres                    | Nro. celular |
|------------|--|--------------|
| 71827961   | Calderon Romero, Yersson               | 925271857    |
| 72697573   | Guevara Valdivia Alejandro<br>Humberto | 984582724    |
| 72077497   | Edwards Andree Ochoa Cutipa            | 992133112    |
| No asistio | Alejandro Rojas Aparicio               |              |

### 1. Título de la Propuesta 1:

Sistema Inteligente de Reconocimiento de Vehículos y Alerta Automática para la Gestión de Seguridad Vial en la Vía de Acceso a la Universidad Continental – Saylla, Cusco 2025

### 2. Problema / Oportunidad detectado: Describir el problema, es importante adjuntar alguna fuente estadística.

En la vía de acceso frente a la Universidad Continental, en Saylla (Cusco), se presentan con frecuencia congestionamientos, incidentes viales y fugas de vehículos que dificultan la respuesta inmediata de las autoridades, como el serenazgo o la Policía Nacional. Esa situación pone en riesgo tanto a peatones como a conductores, especialmente considerando que la infraestructura y señalización de la zona aún son novedosas para muchos.

Según el Observatorio Nacional de Seguridad Vial, en el año 2023 se registraron en todo el país **87 083 siniestros de tránsito**, con **58 000 personas heridas** y **3 316 fallecidas**.

**Además, Cusco se posicionó como la segunda región con más siniestros fatales, con 182 casos, solo detrás de Lima (370)**

## Taller de Proyectos en Ingeniería de Sistemas e Informática

### 3. Descripción de la solución Propuesta/Investigación a desarrollar:

Se propone desarrollar un **sistema inteligente de monitoreo vehicular**, instalado en la vía de acceso a la Universidad Continental – Saylla, que utilice **cámaras y visión por computadora (CNN, YOLO)** para reconocer en tiempo real las placas, tipo y características de los vehículos. Este sistema utilizará:

1. **Cámaras inteligentes** junto con modelos de visión artificial (por ejemplo, YOLO para detección de vehículos).
2. **Algoritmos de reconocimiento automático de placas (ALPR)** y clasificación de tipo de vehículo utilizando CNN u OCR como EasyOCR.
3. **Módulo de alerta automática**, que enviará información clave (placa, tipo, hora, ubicación, una fotografía del vehículo) al serenazgo o la PNP en caso de detectar:
  - Vehículos involucrados en accidentes que intentan fugarse.
  - Maniobras sospechosas o incumplimiento de normas viales.
4. **Una base de datos** centralizada para almacenar registros históricos y permitir análisis posteriores.
5. **Canales de notificación en tiempo real** (como una aplicación web) para que las autoridades puedan actuar rápidamente.

### 4. Requerimientos Funcionales de alto nivel (Indica detalladamente que podrá efectuar la solución)

Requerimientos del producto (al menos 10)

**Detectar y reconocer vehículos en tiempo real** mediante cámaras instaladas en la vía de acceso a la universidad.

**Identificar placas vehiculares automáticamente** usando algoritmos OCR e inteligencia artificial.

## **Taller de Proyectos en Ingeniería de Sistemas e Informática**

**Clasificar el tipo de vehículo** (auto, camioneta, moto, bus) a partir de las imágenes captadas.

**Registrar evidencia** (imagen, hora, ubicación, tipo de vehículo y placa) en una base de datos centralizada.

**Generar alertas automáticas** en caso de fuga tras un accidente o comportamiento sospechoso.

**Enviar notificaciones en tiempo real** a serenazgo o PNP a través de un SMS o aplicación web.

**Acceder a un panel de monitoreo web** donde las autoridades puedan visualizar los incidentes y el historial.

**Almacenar reportes históricos** para análisis de tráfico, zonas de riesgo y estadísticas.

**Soportar integración con sistemas de seguridad existentes** (como cámaras municipales o bases de datos policiales).

**Funcionar de manera continua** (24/7) con capacidad de autodiagnóstico básico (alerta si la cámara falla o no hay conexión).

## Taller de Proyectos en Ingeniería de Sistemas e Informática

### Requerimientos Tecnológicos a usar en el desarrollo de la solución

(Listado de plataformas hardware y software que se requiere para elaborar la solución)

#### - **Hardware:**

##### **Laptop con las siguientes características mínimas:**

- Procesador: Intel Core i7 (10ma gen o superior).
- Memoria RAM: 16 GB min.
- Tarjeta Gráfica (GPU): NVIDIA RTX 3070.

##### **Almacenamiento:**

- SSD de 512 GB.
- HDD/SSD adicional de 1 TB

##### **Cámara Web / Cámara IP (mín. 1080p, 30 fps)**

##### **Router y conexión a internet estable**

**UPS (Sistema de alimentación ininterrumpida)** – Para evitar pérdidas de datos en caso de cortes eléctricos.

#### - **Software:**

##### **Lenguajes de Programación:**

Python 3.10+

##### **Frameworks de Visión Artificial e IA:**

OpenCV (procesamiento de imágenes).

TensorFlow 2.x / PyTorch (entrenamiento y ejecución de CNN, ej. YOLOv8).

EasyOCR (para lectura de placas).

##### **Frameworks Web / Backend:**

Flask (para API REST y microservicios).

##### **Base de Datos:**

PostgreSQL o MySQL (para gestión de registros históricos).

##### **Frontend / Dashboard:**

HTML5, CSS3, JavaScript.

##### **Notificaciones y Comunicación:**

API de Telegram (bot para alertas).

## Taller de Proyectos en Ingeniería de Sistemas e Informática

Twilio o SMTP (SMS / correo electrónico).

### Análisis de la competencia/avances similares

Estado del arte: Nombre una aplicación/investigación/tesis (mínimamente 2)

#### 1. Sistema de Reconocimiento Automático de Placas Vehiculares (ANPR) – ParkPow

- **Nombre:** Deep License Plate Recognition
- **Versión / Año:** GitHub Open Source Project (2021, actualizado en 2023)
- **Funcionalidades:**
  - Detecta vehículos y extrae automáticamente la matrícula de imágenes o video en tiempo real.
  - Compatible con múltiples países y formatos de placas. Permite integración mediante API.
- **Limitaciones:**
  - Enfocado principalmente en la lectura de placas.
  - No incluye sistema de alertas automáticas ni clasificación avanzada del vehículo (tipo, color).
- **Ventaja de nuestra propuesta:**
  - Integramos no solo la detección de placas, sino también la clasificación de tipo de vehículo.
  - Incorporamos un módulo de alerta inmediata a serenazgo/PNP, adaptado al contexto local (Saylla – Cusco).
  - Implementamos un panel web con historial y notificaciones en tiempo real, no solo extracción de datos.

Referencia: [GitHub – parkpow/deep-license-plate-recognition](https://github.com/parkpow/deep-license-plate-recognition)

#### 2. An efficient and layout-independent automatic license plate recognition system based on the YOLO detector” — Laroca et al., 2021 (IET Intelligent Transport Systems)

## Taller de Proyectos en Ingeniería de Sistemas e Informática

**Año / versión:** 2021, enfoque layout-independent (independiente del diseño de la placa).

**Funcionalidades:** unifica detección de LP + clasificación del layout y aplica reglas de post-proceso para mejorar OCR; evalúa 8 datasets (China, Europa, Brasil, etc.) con 96.9% de tasa promedio y > 70 FPS en GPU alta, manteniendo tiempo real con múltiples vehículos.

### Dónde nuestra propuesta mejora:

- Añadimos clasificación de tipo de vehículo (auto/camión/bus/moto) y atributos (color/marca opcional) para correlacionar con infracciones o eventos.
- Entrenamiento fino con datos locales (dominios .pe) y evaluación A/B en la intersección frente a Continental para reducir falsos positivos por obras/señalización nueva.

**Referencia:** [raysonlaroca.github.io](https://raysonlaroca.github.io)

### Autoevaluación:

La solución que presentamos es (Elija únicamente una y sustente)

|                       | Sustento   |
|-----------------------|--|
| innovación            |  |
| innovación disruptiva | Nuestro sistema no solo reconoce placas como lo hacen otros, sino que integra clasificación de vehículos, registro de evidencia y envío de alertas automáticas en tiempo real al serenazgo de Saylla. Esto lo convierte en una propuesta diferente y con impacto directo en la seguridad vial frente a la Universidad Continental. |



## **Taller de Proyectos en Ingeniería de Sistemas e Informática**

## Taller de Proyectos en Ingeniería de Sistemas e Informática

### 1. Título de la Propuesta 2:

Detección de vulnerabilidades en redes informáticas utilizando técnicas de Machine Learning en instituciones educativas de Cusco, Perú, 2025

### 2. Problema / Oportunidad detectado: Describir el problema, es importante adjuntar alguna fuente estadística.

Las instituciones educativas, tanto a nivel de colegios como universidades, se han convertido en objetivos cada vez más frecuentes de ciberataques por parte de actores como APT que son (Amenazas Persistentes Avanzadas) y ciberdelincuentes con motivaciones económicas

En universidades del Reino Unido, **el 85 % identificó ataques maliciosos en el último año**, mientras que en colegios superiores fue el **82 %**. Incluso en escuelas primarias y secundarias la incidencia fue del **41 % y 63 %**, respectivamente, (2023).

### 3. Descripción de la solución Propuesta/Investigación a desarrollar:

El objetivo principal será diseñar y validar un sistema capaz de analizar el tráfico de red y los registros de seguridad, identificando de manera rápida y temprana los comportamientos anómalos que puedan representar intentos de ataque o vulnerabilidades explotables y la solución será, Reducir el tiempo de detección de incidentes en redes educativas, Minimizar riesgos de seguridad ante amenazas como ransomware, phishing o intrusiones externas y por ultimo Proveer un sistema accesible y adaptable para instituciones que no cuentan con grandes recursos en ciberseguridad.



## Taller de Proyectos en Ingeniería de Sistemas e Informática

### 4. Requerimientos Funcionales de alto nivel (Indica detalladamente que podrá efectuar la solución)

#### Requerimientos del producto (al menos 10)

##### 1. Captura y procesamiento de tráfico de red

- El sistema deberá ser capaz de recopilar información en tiempo real o desde registros históricos de red (logs, paquetes, conexiones).
- Podrá filtrar y organizar los datos en variables relevantes para el análisis (IP, puertos, protocolos, tiempo de conexión, intentos de acceso, etc.).

##### 2. Análisis automatizado mediante Machine Learning

- El sistema aplicará algoritmos de Machine Learning (supervisados y/o no supervisados) para clasificar el tráfico de red como normal o sospechoso/malicioso.
- El modelo será capaz de detectar anomalías y patrones inusuales que puedan representar vulnerabilidades o intentos de ataque.

##### 3. Generación de alertas de seguridad

- Cuando se detecte una posible vulnerabilidad o ataque, el sistema deberá emitir alertas en tiempo real.
- Las alertas incluirán información relevante como: origen, tipo de tráfico, gravedad del riesgo y posibles acciones sugeridas.

##### 4. Visualización y reportes

- El sistema deberá contar con un panel o interfaz que muestre de forma clara los resultados del análisis.
- Permitirá generar reportes periódicos (diarios, semanales, mensuales) con estadísticas sobre intentos de intrusión, vulnerabilidades detectadas y tendencias.

##### 5. Compatibilidad con entornos educativos

- La solución deberá adaptarse a redes de tamaño pequeño y mediano, como las utilizadas en universidades, institutos y laboratorios académicos.
- Deberá ser escalable para integrarse en diferentes escenarios (laboratorios de cómputo, servidores académicos, plataformas de e-learning).



## Taller de Proyectos en Ingeniería de Sistemas e Informática

### 6. Aprendizaje y mejora continua

- El modelo deberá actualizarse y mejorar su rendimiento conforme procese nuevos datos de tráfico de red.
- Permitirá la retroalimentación del administrador de la red para afinar la precisión en la detección de incidentes.

## Taller de Proyectos en Ingeniería de Sistemas e Informática

### Requerimientos Tecnológicos a usar en el desarrollo de la solución

(Listado de plataformas hardware y software que se requiere para elaborar la solución)

#### - **Hardware:**

##### 1. Servidor o PC de pruebas con especificaciones mínimas:

- Procesador: Intel i5 / Ryzen 5 o superior.
- Memoria RAM: 8–16 GB (para entrenamiento y pruebas de modelos).
- Almacenamiento: 256 GB SSD o más (para datasets y registros de red).
- Tarjeta de red compatible con captura de tráfico.

##### 2. Opcional: Acceso a máquinas virtuales o entorno en la nube (AWS, Google Cloud, Azure) para pruebas de escalabilidad.

#### - **Software:**

##### 1. **Sistemas Operativos**

- Linux (Ubuntu, Kali Linux para pruebas de seguridad).
- Windows (para compatibilidad y pruebas en entornos educativos).

##### 2. **Lenguajes de Programación y Frameworks de Machine Learning**

- **Python 3.x** (lenguaje principal).
- Librerías de ML:

- Scikit-learn (modelos clásicos de ML).
- TensorFlow / PyTorch (para redes neuronales y deep learning).
- Pandas y NumPy (para manejo y análisis de datos).

##### 3. **Herramientas de Seguridad y Redes**

## Taller de Proyectos en Ingeniería de Sistemas e Informática

- Wireshark (captura y análisis de tráfico de red).
- Nmap (escaneo de puertos y servicios).
- Snort o Suricata (como referencia de IDS tradicionales).
- Tcpdump (captura de paquetes en CLI).

### 4. Bases de Datos

- MySQL o PostgreSQL (para almacenar registros de red y resultados).

### 5. Entornos de Desarrollo

- Visual Studio Code (IDE principal).
- Jupyter Notebook (para experimentación y pruebas de modelos ML).
- Git / GitHub

### 6. Herramientas de Virtualización / Simulación de Redes

- Cisco Packet Tracer / GNS3 (simulación de entornos de red educativa).
- VirtualBox o VMware (máquinas virtuales para pruebas controladas).

## Análisis de la competencia/avances similares

**Estado del arte: Nombre una aplicación/investigación/tesis (mínimamente 2)**

Estado del Arte – Proyectos Relevantes

#### 1. "Estudio de técnicas de aprendizaje automático utilizadas en ciberseguridad"

- Tipo: Tesis de posgrado (2023)
- Autor: William Andrés Villavicencio Benalcázar, Universidad Politécnica Salesiana
- Descripción: Este estudio analiza cómo diversas técnicas de Machine Learning se aplican en la ciberseguridad, incluyendo detección de malware, redes sociales, pruebas de penetración e intrusiones en redes informáticas.

## Taller de Proyectos en Ingeniería de Sistemas e Informática

- Relevancia: Está muy alineado con tu enfoque de utilizar ML para detectar intrusos en redes, aunque de forma más general. Sirve como base para entender qué métodos ML son más comunes en seguridad informática.

## Taller de Proyectos en Ingeniería de Sistemas e Informática

**Autoevaluación:**

**La solución que presentamos es (Elija únicamente una y sustente)**

|                              | <b>Sustento</b>   |
|------------------------------|---|
| <b>innovación</b>            | La solución propuesta se clasifica como innovación, ya que integra técnicas de Machine Learning en el ámbito de la seguridad informática, específicamente en instituciones educativas. Si bien existen sistemas tradicionales de detección de intrusos, la aplicación de modelos de aprendizaje automático permite mejorar la precisión, anticipar amenazas y reducir tiempos de respuesta. Esto representa una mejora significativa frente a las herramientas convencionales, adaptada a un contexto donde los recursos en ciberseguridad son limitados. |
| <b>innovación disruptiva</b> |   |

### 5. Título de la Propuesta 3

**Aplicación móvil para recordatorio de tareas escolares y organización de horarios para estudiantes de secundaria.**

**2. Problema / Oportunidad detectado: Describir el problema, es importante adjuntar alguna fuente estadística.**

Muchos estudiantes de secundaria tienen dificultades para organizar su tiempo, recordar tareas y cumplir con sus horarios académicos. Esto se agrava cuando no cuentan con una herramienta sencilla que les ayude a gestionar sus deberes escolares. La falta de organización

## **Taller de Proyectos en Ingeniería de Sistemas e Informática**

puede llevar a bajo rendimiento académico, estrés y retraso en las entregas.

### **3. Descripción de la solución Propuesta/Investigación a desarrollar:**

**Registrar tareas, trabajos y exámenes con fechas de entrega.**

**Organizar su horario de clases semanal.**

**Recibir recordatorios automáticos antes de cada actividad.**

**Visualizar su calendario académico de forma clara y sencilla.**

**Personalizar colores y categorías (tareas, exámenes, prácticas, etc.).**

### **4. Requerimientos Funcionales de alto nivel (Indica detalladamente que podrá efectuar la solución)**

**Requerimientos del producto (al menos 10)**

**Registrar tareas y actividades con fecha y descripción.**

**Crear un horario semanal de clases.**

**Generar recordatorios automáticos según la fecha programada.**

**Mostrar calendario con vista semanal y mensual.**

**Editar o eliminar tareas fácilmente.**

**Clasificar actividades por colores y tipos.**

## **Taller de Proyectos en Ingeniería de Sistemas e Informática**

**Acceder sin necesidad de internet (modo offline).**

**Enviar notificaciones en el celular.**

**Interfaz adaptada para jóvenes de secundaria.**

**Posibilidad de marcar tareas como "completadas".**

**Requerimientos Tecnológicos a usar en el desarrollo de la solución**

**(Listado de plataformas hardware y software que se requiere para elaborar la solución)**

**- Hardware:**

**Smartphone con Android (versión 8.0 o superior).**



## **Taller de Proyectos en Ingeniería de Sistemas e Informática**

**Conexión Wi-Fi (opcional, para sincronización).**

**- Software:**

**Lenguaje: Java o Kotlin (para Android).**

**Framework: Android Studio.**

**Base de datos local: SQLite o Room.**

**Librerías: Firebase (opcional para sincronización), AlarmManager (para notificaciones).**

**Análisis de la competencia/avances similares**

**Estado del arte: Nombre una aplicación/investigación/tesis (mínimamente 2)**

**Caso 1:**

**Aplicación: MyHomework Student Planner**

**Funcionalidades: Registro de tareas y calendario escolar.**

**Limitación: Interfaz en inglés, no adaptada a currículos latinoamericanos.**

**Ventaja de la propuesta actual: App sencilla, en español y adaptada a estudiantes de secundaria en Perú.**

## Taller de Proyectos en Ingeniería de Sistemas e Informática

### Caso 2:

**Aplicación:** Google Calendar

**Funcionalidades:** Organización de eventos y recordatorios.

**Limitación:** Poca personalización para tareas escolares.

**Ventaja de la propuesta actual:** Enfoque educativo, más simple y con funciones específicas para el día a día del estudiante.

### Autoevaluación:

**La solución que presentamos es (Elija únicamente una y sustente)**

#### **Sustento**

**innovación** Es una innovación educativa que adapta funciones simples de agenda digital a las necesidades específicas de estudiantes INNOVACIÓN escolares, ayudándolos a ser más organizados, mejorar su rendimiento académico y reducir el estrés por olvidos.

**innovación disruptiva**

## Taller de Proyectos en Ingeniería de Sistemas e Informática

---

### 6. Título de la Propuesta 4: