

1. Nombre del proyecto:

SafeFleet IA: Sistema inteligente de monitoreo de somnolencia para flotas vehiculares

2. Descripción detallada de la solución:

SafeFleet IA es un sistema de control de flota de vehículos que combina hardware en cada vehículo (Raspberry Pi, cámara, buzzer y luz de emergencia) con un backend en la nube y una aplicación web de administración.

En cada vehículo se instala una Raspberry Pi conectada a una cámara frontal enfocada al rostro del piloto. Sobre esta Raspberry corre un modelo de IA de visión por computadora capaz de detectar signos de somnolencia (ojos cerrados por cierto tiempo, mirada caída, micro sueños, etc.).

Cuando el modelo detecta que el piloto se está quedando dormido, la Raspberry:

- Activa un buzzer/alarma sonora dentro del vehículo.
- Enciende una luz de emergencia visible.
- Envía una alerta en tiempo real al backend, indicando piloto, vehículo, hora y tipo de evento.

El backend se conecta a una base de datos donde se almacena la información de:

- Flota de vehículos (placa, modelo, estado, etc.)
- Empleados/pilotos (nombre, DPI, contacto, licencia, etc.)
- Historial de alertas (quién, en qué vehículo, cuándo, qué pasó).
- Tiempos de viaje y reportes de problemas.

En el frontend web (para el encargado de flota) se podrán realizar acciones como:

- Agregar nuevos vehículos y registrar su información (placa, nombre interno, tipo, etc.).

- Registrar empleados y asignarlos a vehículos.
- Ver una planilla de vehículos de empleados (quién conduce qué).
- Consultar historial de alertas de somnolencia, filtrado por piloto, vehículo o rango de fechas.
- Ver reportes de tiempos de viaje y eventos críticos.
- Registrar y visualizar reportes de problemas con el vehículo o con el piloto.

Además, el sistema puede enviar notificaciones simultáneas al piloto (en cabina) y al encargado (por web y/o correo/notificación móvil) para que se verifique el estado del piloto y se tomen decisiones (parar, cambiar de conductor, etc.).

3. Metodologías de IA que usará:

- Supervisado / No Supervisado:
 - Principalmente Aprendizaje Supervisado, usando datos etiquetados de rostros con estados “despierto” / “somnoliento”.
- Tipo de modelo:
 - Red Neuronal Convolucional (CNN) para detección del rostro y análisis de ojos/boca (indicadores de sueño).
 - Posible uso de modelos pre-entrenados (transfer learning) para reconocimiento facial y luego una capa final para clasificar somnolencia.
- Uso de TensorFlow:
 - Entrenamiento del modelo de detección de somnolencia usando TensorFlow.
 - Exportación del modelo a TensorFlow Lite para ejecutarlo de forma optimizada en la Raspberry Pi.

4. Datos de entrada y salida del sistema:

- Input:
 - Video/frames de la cámara enfocada al rostro del piloto.
 - Identificador del piloto y del vehículo (enviado por la Raspberry al backend).
 - Datos de contexto: hora, fecha, posible ubicación (GPS si se integra), duración del viaje.
 - Eventos manuales: reportes de problemas ingresados desde el frontend.
- Output:
 - Alertas de somnolencia (tipo de alerta, nivel de riesgo, duración).
 - Activación de buzzer y luz de emergencia en el vehículo.
 - Registro en la base de datos del evento (piloto, vehículo, hora, tipo de alerta).
 - Reportes en el frontend:
 - Historial de alertas por piloto/vehículo.
 - Planilla de vehículos y conductores.
 - Tiempos de viaje y número de incidentes.
 - Notificaciones al encargado de flota (por panel web y/o correo/notificación push).

5. Usuarios y beneficiarios:

- Usuarios directos:
 - Pilotos/Conductores de la flota (reciben alertas que les ayudan a no dormirse).
 - Encargados de flota / Jefes de logística (monitorean el estado de los vehículos y pilotos).
- Beneficiarios indirectos:

- Empresa (menos accidentes, menor costo en seguros y mantenimiento).
- Clientes que dependen de entregas a tiempo y seguras.
- Peatones y otros conductores, al reducirse el riesgo de accidentes por sueño.

6. Beneficios claros:

- Reducción de accidentes causados por somnolencia de los pilotos.
- Monitoreo en tiempo real del estado de la flota y de los conductores.
- Historial trazable de alertas y eventos para auditorías y decisiones de recursos humanos (capacitación, cambios de horarios, etc.).
- Optimización de horarios y rutas, al identificar patrones de sueño en ciertos tramos o horas.
- Mejora de la seguridad laboral y cumplimiento de normativas de seguridad en transporte.

7. Pros y contras:

- Pros:
 - Sistema proactivo: detecta somnolencia antes de un accidente grave.
 - Uso de hardware relativamente económico (Raspberry, cámara, buzzer, luz).
 - Escalable a muchos vehículos: se repite el módulo de Raspberry + cámara.
 - Integración con panel web para control centralizado de la flota.
- Contras:

- Dependencia de condiciones de iluminación (puede requerir cámara IR o iluminación adicional de noche).
- Posibles falsos positivos/negativos del modelo (piloto mirando hacia abajo pero no dormido, uso de lentes, etc.).
- Requiere conectividad estable (WiFi/4G) para enviar alertas en tiempo real al backend.
- Posibles preocupaciones de privacidad del piloto al ser grabado constantemente.

8. Prototipo conceptual (descripción):

- Pantalla 1: Login y Dashboard general
 - Formulario de inicio de sesión para el encargado de flota.
 - Una vez dentro, un dashboard con:
 - Número total de vehículos.
 - Número de alertas en las últimas 24 horas.
 - Lista rápida de “vehículos con más alertas de somnolencia”.
 - Gráfico simple de alertas por día/semana.
- Pantalla 2: Gestión de flota y empleados
 - Planilla de vehículos con columnas: placa, nombre interno, tipo, estado (activo/en mantenimiento), piloto asignado.
 - Botón “Agregar nuevo vehículo” para registrar placa, nombre, marca, etc.
 - Sección de empleados/pilotos:
 - Listado de pilotos con datos básicos.
 - Botón “Registrar nuevo empleado”.
 - Opción para asignar piloto a vehículo desde un desplegable.

- Pantalla 3: Historial de alertas y reportes
 - Tabla de historial de alertas:
 - Fecha y hora.
 - Piloto.
 - Vehículo.
 - Tipo de alerta (somnolencia leve, crítica, etc.).
 - Estado (atendida / pendiente).
 - Filtros por rango de fecha, vehículo, piloto.
 - Vista detallada de un evento: mapa de ubicación (opcional), duración del evento, nota del encargado sobre qué se hizo (por ejemplo: “se detuvo el vehículo y se cambió de conductor”).
 - Sección para reportes de problemas (ej. falla mecánica, quejas del piloto) con posibilidad de adjuntar comentarios.



