

<p><b>Cliente</b></p> <p><b>Usuario</b></p>	<p>Ciudad de Palmira, Valle del Cauca (Colombia).</p> <p>Autoridades municipales, equipos de seguridad y ciudadanos.</p>
<p><b>Requerimientos funcionales</b></p>	<p><b>RF-01:</b> Gestión de Archivos JSON, <b>RF-02:</b> Manejo de Excepciones, <b>RF-03:</b> Almacenamiento y Manipulación mediante Listas Enlazadas, <b>RF-04:</b> Registro y Gestión de Rutas, <b>RF-05:</b> Registro y Gestión de Incidentes, <b>RF-06:</b> Búsqueda de Elementos y Determinación de la Mejor Ruta, <b>RF-07:</b> Registro y Validación de Personas, <b>RF-08:</b> Actualización y Eliminación de Personas, <b>RF-09:</b> Operaciones Complementarias sobre el Conjunto de Personas, <b>RF-10:</b> Ordenamiento y Filtrado de Personas, <b>RF-11:</b> Importación y Exportación del Conjunto de Personas, <b>RF-12:</b> Generación de Reportes y Estadísticas Integrales</p>
<p><b>Contexto del problema</b></p>	<p>La ciudad de Palmira ha experimentado un rápido crecimiento económico y poblacional, lo que ha generado serios problemas de seguridad y movilidad. Los altos índices de delincuencia, robos, accidentes y la congestión vehicular, sumados a la falta de rutas de transporte optimizadas, afectan tanto a la ciudadanía como a las autoridades. Actualmente, no se dispone de un sistema centralizado que permita el registro y monitoreo en tiempo real de incidentes de seguridad, la administración de rutas, y la gestión de información de pasajeros y conductores. Ante este escenario, el Sistema de Gestión y Monitoreo de Movilidad y Seguridad (SGMMS) se plantea como la solución para mejorar la vigilancia, agilizar la respuesta a emergencias y proporcionar información actualizada que respalde la toma de decisiones estratégicas.</p>
<p><b>Requerimientos no funcionales</b></p>	<ul style="list-style-type: none"> <li>• <b>Rendimiento:</b> El sistema debe procesar operaciones de lectura, búsqueda y ordenamiento en tiempos razonables.</li> <li>• <b>Calidad del Código:</b> El software debe ser modular, legible y contar con la documentación adecuada para facilitar su mantenimiento.</li> <li>• <b>Almacenamiento Permanente de Datos:</b> La información se conservará en archivos JSON, garantizando que los datos permanezcan disponibles entre ejecuciones de la aplicación.</li> <li>• <b>Estructuras de Datos:</b> Se utilizarán listas enlazadas (simples, dobles o circulares) para almacenar y manipular la información, evitando el uso de estructuras como ArrayList.</li> <li>• <b>Manejo de Errores:</b> Se implementará un sistema robusto de captura y gestión de excepciones que ofrezca mensajes de error claros y evite estados</li> </ul>

	inesperados en el sistema.
--	----------------------------

Identificador y nombre	<b>RF-01: Gestión de Archivos JSON</b>		
Resumen	<i>Este requerimiento consiste en permitir la lectura (importación) y escritura (exportación) de datos en formato JSON para cargar y guardar la información del sistema (rutas, incidentes y personas).</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Ruta del archivo JSON	String	<i>Debe ser una ruta válida en el sistema de archivos, con extensión. json</i>
	Contenido del archivo JSON	String (estructura JSON)	Debe cumplir con la estructura JSON esperada, incluyendo campos obligatorios (rutas, incidentes, personas).
Resultado o Postcondición	<ul style="list-style-type: none"> <li>Si se trata de <b>importación</b>, el sistema debe cargar los datos del archivo y convertirlos en objetos del modelo, almacenándolos en las estructuras de datos internas.</li> <li>Si se trata de <b>importación</b>, el sistema debe cargar los datos del archivo y convertirlos en objetos del modelo, almacenándolos en las estructuras de datos internas.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Estructura de datos en memoria (actualizada)	Objetos del modelo (Rutas, Incidentes, Personas)	
	Archivo JSON (en caso de exportación)	Archivo	<i>Archivo</i>

Identificador y nombre	<b>RF-02: Manejo de Excepciones</b>		
Resumen	<i>Se encarga de capturar y gestionar los errores que puedan ocurrir durante operaciones críticas (lectura/escritura de archivos, conversión de datos, procesos internos). Debe utilizar excepciones personalizadas para informar claramente al usuario y evitar estados inconsistentes en el sistema.</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Datos o condiciones de error	String	<i>Situaciones que no cumplan las condiciones esperadas (archivo no encontrado, JSON inválido, parámetros nulos, etc.)</i>
Resultado o Postcondición	<ul style="list-style-type: none"> <li>El sistema lanza excepciones personalizadas con mensajes descriptivos.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de error o excepción lanzada	String	<i>Texto en consola o en logs del sistema</i>

Identificador y nombre	<b>RF-03: Almacenamiento y Manipulación mediante Listas Enlazadas</b>		
Resumen	<i>Todos los datos (rutas, incidentes, personas) se almacenan y gestionan en estructuras de listas enlazadas (simples, dobles o circulares). No se debe utilizar ArrayList. Esto facilita operaciones de inserción, eliminación y recorrido sin depender de estructuras de la biblioteca estándar.</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Datos de rutas, incidentes y personas	Objetos del modelo	<i>Objetos creados correctamente (IDs válidos, campos requeridos completos, etc.)</i>
Resultado o Postcondición	<ul style="list-style-type: none"> <li>Los datos quedan organizados en listas enlazadas, permitiendo operaciones subsecuentes de búsqueda, ordenamiento, filtrado y reportes.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Estructura de listas enlazadas en memoria	Listas enlazadas	

Identificador y nombre	<b>RF-04: Registro y Gestión de Rutas</b>		
Resumen	<i>Permite registrar nuevas rutas (ID, distancia, tiempo estimado, puntos de inicio y fin) y mantenerlas en la estructura de datos. Incluye la posibilidad de ordenarlas por distancia o tiempo estimado.</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Datos de la ruta	Objetos Ruta (ID, distancia, tiempo, puntos)	<i>Distancia y tiempo deben ser valores numéricos positivos. ID único para cada ruta.</i>
Resultado o Postcondición	<ul style="list-style-type: none"> <li>Las rutas se almacenan en la lista enlazada correspondiente.</li> <li>Se dispone de métodos para ordenarlas según el criterio requerido (distancia o tiempo).</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Lista de rutas	Lista enlazada de objetos Ruta	

Identificador y nombre	<b>RF-05: Registro y Gestión de Incidentes</b>		
Resumen	<i>Permite ingresar y gestionar incidentes de seguridad (ID, tipo, ubicación, fecha/hora, descripción, estado), ordenándolos por fecha/hora de reporte (más recientes primero).</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Datos del incidente	Objeto Incidente (ID, tipo, ubicación, fecha/hora, descripción, estado)	<i>La fecha/hora debe tener un formato válido (por ejemplo, dd-MM-yyyy HH:mm). El ID debe ser único.</i>
Resultado o Postcondición	<ul style="list-style-type: none"> <li>Los incidentes se registran en la lista enlazada correspondiente.</li> <li>El sistema puede ordenarlos cronológicamente de más reciente a más antiguo.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Lista de incidentes	Lista enlazada de objetos Incidente	

Identificador y nombre	<b>RF-06: Búsqueda de Elementos y Determinación de la Mejor Ruta</b>		
Resumen	Facilita la localización de elementos específicos en el sistema (incidentes por ID, conductores por nombre) y la determinación de la mejor ruta según criterios predefinidos (por ejemplo, menor distancia y tiempo).		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Parámetro de búsqueda (ID o nombre)	String	Debe existir en la estructura de datos para que la búsqueda sea exitosa
	Criterios de la mejor ruta	Int	Valores positivos para distancia y tiempo
Resultado o Postcondición	<ul style="list-style-type: none"> <li>Se retorna el elemento buscado o un indicador de “no encontrado”.</li> <li>Se identifica la ruta óptima en función de los criterios establecidos.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Resultado de la búsqueda (incidente, conductor o ruta)	Objeto Incidente, Conductor o Ruta	
	Mensaje de resultado	String	Texto en consola (p. ej., “Elemento no encontrado”)

Identificador y nombre	<b>RF-07: Registro y Validación de Personas</b>		
Resumen	<i>Permite registrar nuevos pasajeros y conductores, asegurando que no se dupliquen IDs. Incluye la verificación de unicidad para cada persona que ingresa al sistema.</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Datos de la persona (ID, nombre, contacto, etc.)	Objeto Persona (Pasajero o Conductor)	<i>ID único. Nombre y contacto con el formato adecuado (p. ej., número de teléfono válido).</i>
Resultado o Postcondición	<ul style="list-style-type: none"> <li>La persona se almacena en la lista enlazada si no existe duplicado.</li> <li>Se garantiza la unicidad de cada registro.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Conjunto de personas actualizado	Lista enlazada de personas	



Identificador y nombre	<b>RF-08: Actualización y Eliminación de Personas</b>		
Resumen	<i>Permite modificar la información de una persona existente o eliminarla del conjunto de personas.</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Identificador de la persona a actualizar/eliminar	String (ID)	<i>Debe existir en el conjunto</i>
	Nuevos datos (solo en caso de actualización)	Objeto Persona con datos actualizados	Campos deben mantener la consistencia (ID existente, formato de nombre, etc.)
Resultado o Postcondición	<ul style="list-style-type: none"> <li>• Los datos de la persona se modifican en el sistema o se elimina el registro de la lista enlazada.</li> <li>• El conjunto de personas refleja los cambios de manera inmediata.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de confirmación	String	<i>Texto en consola (p. ej., "Persona actualizada correctamente" o "Persona eliminada")</i>

Identificador y nombre	<b>RF-09: Operaciones Complementarias sobre el Conjunto de Personas</b>		
Resumen	<i>Incluye funcionalidades para administrar el conjunto de personas: imprimir la lista completa, contar el número de registros, verificar si está vacío, limpiar el conjunto y convertirlo a un arreglo.</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Conjunto de personas	Lista enlazada de personas	<i>Debe existir y estar correctamente inicializada</i>
Resultado o Postcondición	<ul style="list-style-type: none"> <li>El sistema realiza las operaciones solicitadas (por ejemplo, limpieza total, conteo de registros, etc.) y retorna la información solicitada o actualiza la estructura.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Lista impresa o valor numérico	String	<i>Texto en consola (p. ej., "Hay 10 personas registradas")</i>
	Arreglo resultante (si se solicita conversión)	Array de objetos Persona	<i>Estructura de arreglo en memoria</i>

Identificador y nombre	<b>RF-10: Ordenamiento y Filtrado de Personas</b>		
Resumen	<i>Permite organizar el conjunto de personas según criterios como el orden alfabético por nombre, así como aplicar filtros (por ejemplo, por edad o por rol –pasajero/conductor–).</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Conjunto de personas	Lista enlazada de personas	<i>Debe estar correctamente cargada</i>
	Criterio de ordenamiento/filtrado	String	Por ejemplo, “nombre”, “edad>30”, “rol=conductor”
Resultado o Postcondición	Se obtiene una lista ordenada o filtrada según el criterio solicitado.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Lista de personas resultante	Lista enlazada de personas	

Identificador y nombre	<b>RF-11: Importación y Exportación del Conjunto de Personas</b>		
Resumen	<i>Permite importar registros de personas desde un archivo (por ejemplo, JSON) y exportar el conjunto actual a un archivo para persistir los datos.</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Archivo de importación/exportación	String	<i>Debe existir (en importación) o crearse (en exportación), con extensión .json</i>
	Conjunto de personas (para exportar)	Lista enlazada de personas	Debe existir y contener datos válidos
Resultado o Postcondición	<ul style="list-style-type: none"> <li>• En la importación, los datos del archivo se cargan al conjunto.</li> <li>• En la exportación, el archivo queda actualizado con los datos del conjunto.</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Conjunto de personas actualizado (tras importación)	Lista enlazadas de personas	
	Archivo JSON (tras exportación)	Archivo	<i>.json</i>

**Universidad Icesi**  
**Departamento de Computación y Sistemas Inteligentes**  
**Identificación del problema y análisis de requerimientos**  
**Algoritmos y Programación II**

Identificador y nombre	<b>RF-12: Generación de Reportes y Estadísticas Integrales</b>		
Resumen	<i>Genera reportes en consola (o en un formato determinado) que integren la información de rutas, incidentes y personas, y calcula estadísticas relevantes (por ejemplo, totales, promedios, distribuciones) para la toma de decisiones.</i>		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	Listas de datos procesados (rutas, incidentes, personas)	Listas enlazadas	<i>Deben contener datos válidos para el reporte</i>
Resultado o Postcondición	<ul style="list-style-type: none"> <li>Se generan informes que muestran la información de forma organizada (rutas ordenadas, incidentes ordenados, datos de personas) y se calculan indicadores (cantidad total, promedio de edad, número de incidentes recientes, etc.).</li> </ul>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Reporte en consola o archivo	String	<i>Estructura legible (p. ej., tabla, lista, etc.)</i>
	Estadísticas calculadas	string	<i>Texto numérico (p. ej., "Promedio de edad: 32.5")</i>