

María Paula Calderon Jaimes, Andrés Camilo Cardona Carrasquilla, Juan Sebastian Pachón Carvajal

No. de Equipo Trabajo: 1

I. INTRODUCCIÓN

En un país como Colombia, circulan más de 14 millones de vehículos por las principales carreteras del país, cifra que termina siendo bastante preocupante si analizamos que casi la mitad de esos vehículos no cuenta con el Seguro Obligatorio Soat o la revisión Técnico- mecánica, instrumentos que se han convertido en garantes de protección tanto para los conductores como cualquier otra persona relacionada con la cadena de movilidad, pues garantizan que los vehículos que ruedan por las carreteras de nuestro país, sí están en condiciones de hacerlo y en caso de presentarse un siniestro vial, se cubran las lesiones de las personas implicadas.

Es por eso que ante la preocupante cifra de conductores y empresas que ponen a circular vehículos sin el seguro o la revisión, surge la idea de crear una aplicación, puesto que vivimos en un mundo tecnológicamente conectado, que ayude tanto a personas del común como grandes compañías de transporte a estar seguros y confiados de los vehículos que están poniendo en marcha y de esta forma puedan acceder mucho más rápido a información importante como fechas de revisión, vencimientos del seguro, entre otros aspectos.

II. DESCRIPCIÓN DEL PROBLEMA A RESOLVER

En esta sección se debe describir de manera general el problema que se propone resolver en el proyecto de clase. Se debe especificar el propósito u objetivo general.

En nuestro país, es común que las empresas de transporte o incluso las personas naturales no tengan el SOAT o la revisión técnico-mecánica al día. Según cifras del Registro Único Nacional de Tránsito (RUNT) alrededor del 46% de los vehículos que circulan en el país no poseen el seguro y el 56% no posee la revisión. Debido a esto, surge la idea de crear una aplicación móvil con el fin de brindar una herramienta que facilite la gestión de vehículos a las empresas, de tal manera que estas tengan información sobre el estado técnico y legal de sus vehículos.

III. USUARIOS DEL PRODUCTO DE SOFTWARE

En esta sección se deben mencionar las características y clasificación de los usuarios (perfiles/roles) que utilizarán el producto. La clasificación puede ser en función a la frecuencia de uso, grupo de funcionalidades utilizadas, privilegios de acceso y seguridad, nivel de experiencia y otros parámetros.

La aplicación va dirigida, principalmente, a aquellas personas que trabajan en empresas de transporte, quienes manejan diferentes cantidades de autos y muchas veces no cuentan con una herramienta que les ayude con el control de los mismos.

IV. REQUERIMIENTOS FUNCIONALES DEL SOFTWARE

Nombre: Registro de propiedades y elementos del vehículo.

Descripción: Con esta funcionalidad se crea un objeto para ser almacenado en una estructura.

Acciones indicadoras y comportamiento esperado: El usuario ingresará los atributos del auto a registrar, tanto las físicas, como las temporales. El usuario recibirá un mensaje de si el proceso se realizó con éxito.

Requerimientos funcionales:

- Creación.
- Ordenamiento.
- Almacenamiento.

Nombre: Registro de usuario y contraseña.

Descripción: Con esta funcionalidad se crea usuario para ser almacenado en una base de datos.

Acciones indicadoras y comportamiento esperado: El usuario ingresará un correo y una contraseña. El usuario recibirá un mensaje de si el proceso se realizó con éxito o no.

Requerimientos funcionales:

- Creación..
- Almacenamiento.

Nombre: Señalar fechas de mantenimiento y periodicidad de un auto.

Descripción: Con esta funcionalidad se muestra la fecha del último mantenimiento realizado y cuando debe ser el siguiente de un vehículo en particular.

Acciones indicadoras y comportamiento esperado: El usuario ingresará el identificador de un vehículo para obtener información sobre el mantenimiento y periodicidad de este. Recibirá la información mencionada.

Requerimientos funcionales:

- Almacenamiento.
- Ordenamiento.
- Búsqueda parcial de datos.

Nombre: Señalar fechas de vencimiento de seguro y periodicidad de un auto.

Descripción: Con esta funcionalidad se muestra la fecha del pago del seguro realizado y cuando debe ser el siguiente, de un vehículo en particular.

Acciones indicadoras y comportamiento esperado: El usuario ingresará el identificador de un vehículo para obtener información sobre el seguro y periodicidad de este. Recibirá la información mencionada.

Requerimientos funcionales:

- Almacenamiento.
- Ordenamiento.
- Búsqueda parcial de datos.

Nombre: Señalar vehículos con fechas de mantenimiento más próximas.

Descripción: Con esta funcionalidad se muestra los vehículos que tienen más próxima su fecha de mantenimiento.

Acciones indicadoras y comportamiento esperado: El sistema mostrará los vehículos con fecha más próxima de mantenimiento.

Requerimientos funcionales:

- Almacenamiento.
- Ordenamiento.
- Búsqueda parcial de datos.

Nombre: Señalar vehículos con fechas de pago de seguro más próximas.

Descripción: Con esta funcionalidad se muestra los vehículos que tienen más próxima su fecha de pago de seguro.

Acciones indicadoras y comportamiento esperado: El sistema mostrará los vehículos con fecha más próxima de pago de seguro.

Requerimientos funcionales:

- Almacenamiento.
- Ordenamiento.
- Búsqueda parcial de datos.

Nombre: Modificación de propiedades y elementos del vehículo.

Descripción: Con esta funcionalidad modifica un objeto ya almacenado una estructura.

Acciones indicadoras y comportamiento esperado: El usuario ingresará los atributos del auto a modificar, tanto las físicas, como las temporales. El usuario recibirá un mensaje de si el proceso se realizó con éxito.

Requerimientos funcionales:

- Actualización.

- Ordenamiento.
- Almacenamiento.

Nombre: Eliminar un vehículo.

Descripción: Con esta funcionalidad elimina un objeto ya almacenado una estructura.

Acciones indicadoras y comportamiento esperado: El usuario ingresará el identificador de. El usuario recibirá un mensaje de si el proceso se realizó con éxito.

Requerimientos funcionales:

- Eliminación.
- Ordenamiento.
- Almacenamiento.

V. DESCRIPCIÓN DE LA INTERFAZ DE USUARIO PRELIMINAR

A continuación se muestra un primer avance de lo que se espera para el apartado gráfico de la aplicación.

En la figura 1, se muestra el resultado de consultar la información de un vehículo, también se ve cómo el usuario tendrá las opciones de ver más vehículos que tenga registrados o agregar otro.

En la figura 2 se observa la pantalla del login, donde el usuario debe colocar sus datos para poder acceder a la aplicación.

En la figura 3 vemos cómo se desplegará las opciones que tendrá el usuario, de momento está la primera, que es consulta general de vehículos.



Figura 1.

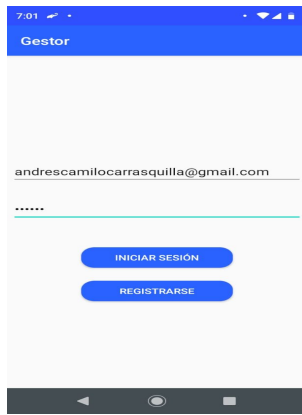


Figura 2.

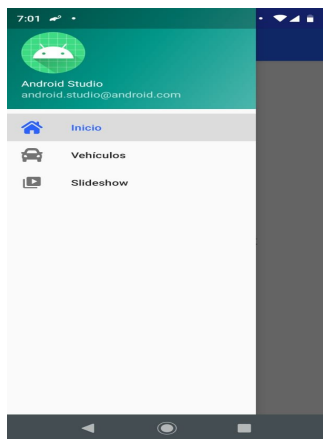


Figura 3.

VI. ENTORNOS DE DESARROLLO Y DE OPERACIÓN

Dado que el producto a desarrollar es una aplicación, el entorno de desarrollo usado será Android Studio, usando firebase como base de datos con la cual trabajar.

VII. PROTOTIPO DE SOFTWARE INICIAL

El repositorio en el cual se encuentra el código del proyecto es:

https://github.com/jupachonc/DSProject_Gestor.git

Los anexos siguiendo la estructura sugerida están disponibles en:

https://github.com/jupachonc/DSProject_Attachment/

Estructuración inicial del proyecto:

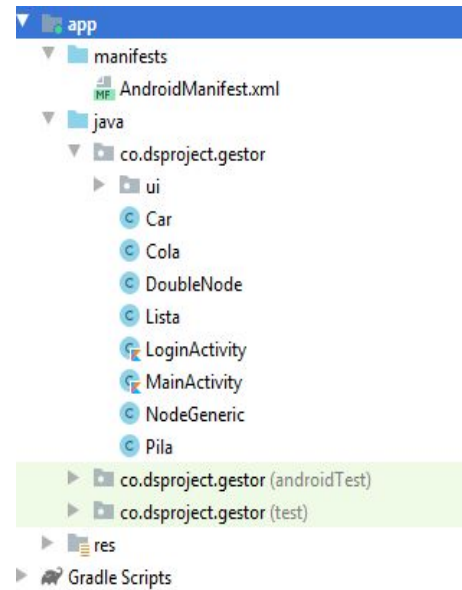


Figura 4. Organización inicial del proyecto.

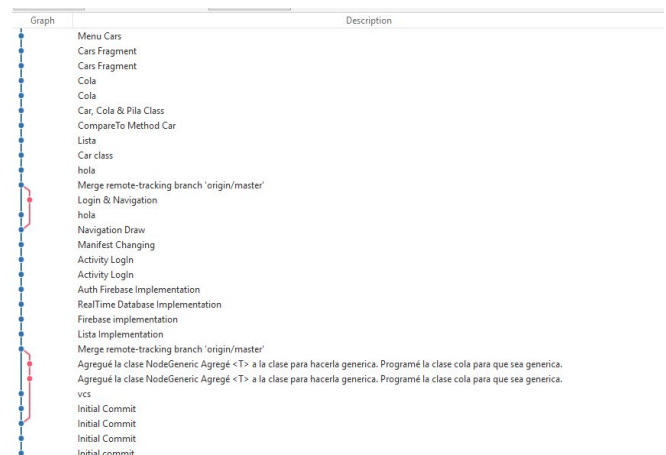


figura 4.1 Rama master inicial.

VIII. PROTOTIPO

Al realizar pruebas con el prototipo, encontramos diferentes tiempos en las diferentes funciones que tenemos y las estructuras que usamos.

Al momento de insertar los datos para ser almacenados, obtenemos los siguientes tiempos en segundos.

Cantidad de datos y estructura	Cola	Pila	Lista doblemente enlazada
10 mil datos	0.0526415	0.048036800	0.712999800

100 mil datos	0.1892552	0.1919921	289.6013102
500 mil datos	2.7512712	2.689901800	10152.9843744
1 millón de datos	5.7276427	6.0506449	Error por espacio de memoria.
9 millón de datos	18.0234867	24.8939489	Error por espacio de memoria.

Se puede observar el comportamiento de las diferentes estructuras con la siguientes gráficas:

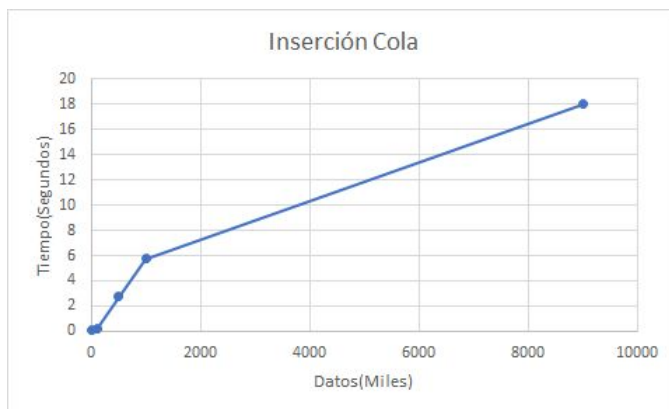


Figura 5. Gráfica de comportamiento de una cola en función de la cantidad de datos.



Figura 6. Gráfica de comportamiento de una pila en función de la cantidad de datos.



Figura 7. Gráfica de comportamiento de una pila en función de la cantidad de datos.

Al momento de liberar espacio eliminando objetos, obtenemos lo siguiente:

Cantidad de datos y estructura	Cola	Pila	Lista doblemente enlazada
10 mil datos	0.00182300	0.001474	4.659005E-4
100 mil datos	0.0072556	0.0083896	0.0081648
500 mil datos	0.03699130	0.0398443	0.038525500
1 millón de datos	0.0778405	0.0770374	Error por espacio de memoria.
9 millón de datos	0.2471933	0.2542022	Error por espacio de memoria.

Se puede observar el comportamiento de las diferentes estructuras con la siguientes gráficas:

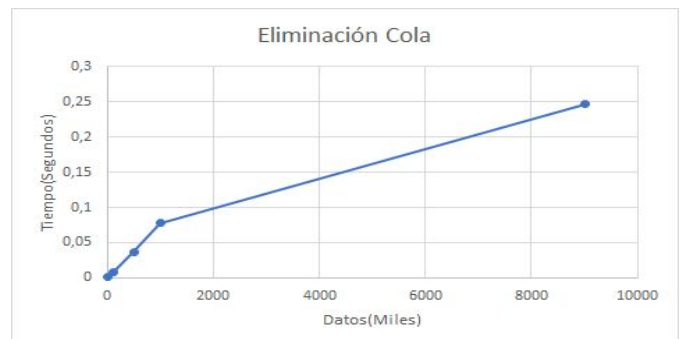


Figura 8. Gráfica de comportamiento de una cola en función de la cantidad de datos.

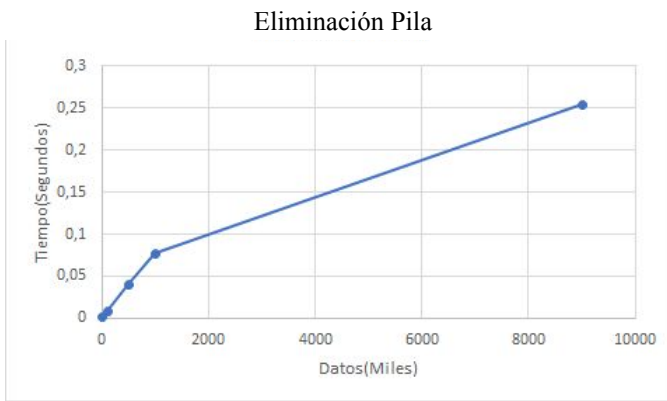


Figura 9. Gráfica de comportamiento de una pila en función de la cantidad de datos.

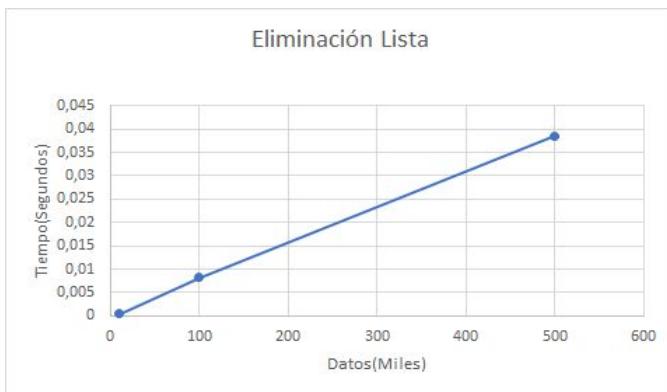


Figura 10. Gráfica de comportamiento de una pila en función de la cantidad de datos.

I. ANÁLISIS DE COMPARATIVO

Podemos observar como en la Pila y en Cola, el tiempo de ejecución para insertar es mucho menor a el tiempo de una lista, dado que en este par de estructuras, insertamos y eliminamos en un tiempo constante $O(1)$, dado que tenemos un apuntador directo a la posición donde insertamos. En el caso de la lista, la complejidad es $O(n)$, dado que debemos recorrerla completa para verificar que el dato que se use no se encuentre repetido.

II. ROLES Y ACTIVIDADES

Integrante	Roles	Actividades
María Paula Jaimes Calderón	Coordinadora, secretaria y observadora.	Creación de estructuras y presentación del proyecto.
Andres Camilo Cardona Carrasquilla	Investigador y técnico.	Creación de estructuras, documentación del proyecto y

		realizador de pruebas.
Juan Sebastián Pachón Carvajal	Líder y experto.	Creación de estructuras y del prototipo

III. DIFICULTADES Y LECCIONES APRENDIDAS

Una de las dificultades encontradas mientras se realizaba la entrega, era la relación cantidad de datos y la potencia del equipo en el que se realizaban las pruebas, dado que en algunos casos, el que fuesen tantos datos que el equipo no podía procesarlos, pues superan las limitaciones de memoria de la máquina.