

**UNIVERSIDAD TÉCNICA NACIONAL  
INGENIERÍA DEL SOFTWARE**

**CURSO:  
PRINCIPIOS DE PROGRAMACIÓN**

**PROYECTO DENOMINADO:  
“EVENTOS-COLISIÓN DE VEHÍCULOS”**

**INTEGRANTES:  
JOSÉ RAÚL HIDALGO PÉREZ  
ISABEL MARÍA MARTÍNEZ VINDAS  
ANGÉLICA VALERA VARGAS**

**DOCENTE:  
FREDDY GERARDO ROCHA BOZA**

**II CUATRIMESTRE 2024**

## **Descripción del proyecto:**

Este proyecto tuvo como objetivo crear un programa de colisión de vehículos. La principal función que ejecuta este código es que se encuentra dentro de "vehicleCollision (generalFunctions.loginMenu)" la cual se encarga de procesar toda la demás programación implementada en cada función, en otras palabras, almacena las variables de: "adminFuntions, citizenFuntions, judgeFuntions, officerFuntions, globalVariables".

La función de variables globales contiene la creación del administrador con más detalle, al igual que las listas y diccionarios donde se almacenará toda la información (alguna ya predeterminada) y la escala de colores que se utiliza el texto de visualización.

La función de administrador es nuestro bloque principal de trabajo, ya que es el único usuario por defecto que debe utilizarse al ingresar al sistema, y será el encargado de crear los otros usuarios que se registren posteriormente, al igual que es el que le asigna cada rol (dicho rol maneja un menú en específico y no puede cambiar), los disponibles son el del juez, ciudadano y el policía.

Con la función ciudadano se dispone del manejo de vehículos (utiliza un CRUD) y eventos (utiliza un CRUD). Y cada cambio que realice se guardará para este usuario. Los CRUD se refieren a crear, leer, actualizar y eliminar.

La función del juez puede utilizar la visualización de la lista de incidentes por aprobar y manejar el número de registro del incidente.

La función del oficial puede ver la lista de eventos sucedidos y agregar el número de la multa.

Estas son las funciones más importantes que ejecutan todo este código de la manera más eficiente posible.

### Dificultades del proyecto:

Estudiante	Problema	Solución
Raúl	1. Asignar el rol sin afectar el código.	Utilizar la función global y el for con los datos de currentUser, currentUserName, currentUserProfile, currentUserID
Isabel	2. Guardar los datos de registro del nuevo usuario.  3. Actualizar las provincias y cantones	Utilizar un diccionario vacío y ingresarle atributos uno por uno  Ej:  newUser["id"] = newUserID  -----  Con el "for i in" y el "index" se pudo ingresar a las provincias verificar que no se repita y actualizar
Angélica	4. Crear el menú de opciones del juez.  5. Localizar un archivo para poder modificarlo.	Crear una función, que utilice un ciclo y valide la variable opción.  -----  Utilizar este formato para poder acceder a el archicvo  __location__ = os.path.realpath(  os.path.join(os.getcwd(), os.path.dirname(__file__)))

### Conclusiones:

Estudiante	Conclusiones
Isabel	<ol style="list-style-type: none"><li>1. Fue bastante complicado implementar algunos registros de datos.</li><li>2. Algunas instrucciones fueron complejas de entender.</li><li>3. La mayoría de las funciones que utilizamos se trataron de hacer lo más sencillas y comprensibles.</li></ol>
Raúl	<ol style="list-style-type: none"><li>1. El proyecto implementa un manejo robusto de datos para los usuarios ciudadanos mediante operaciones CRUD.</li><li>2. La función de administrador se implementó como el bloque central, encargada de gestionar la creación de otros usuarios y asignar roles.</li><li>3. El proyecto logró una organización eficiente mediante la creación de funciones específicas para diferentes roles (administrador, ciudadano, juez, policía).</li></ol>
Angélica	<ol style="list-style-type: none"><li>1. Identificar y optimizar las partes del código que afectaban el rendimiento del sistema fue un poco tedioso y sumamente importante para su funcionamiento.</li><li>2. La correcta administración de variables globales y la estructuración clara del código han sido fundamentales para mantener el proyecto organizado y fácilmente escalable.</li><li>3. En este proyecto he podido ver como aplica la programación en la resolución de problemas del mundo real.</li></ol>

**Bitácora:**

Aspecto	Raúl	Isabel	Angélica
Documentación interna	35%	35%	30%
Documentación externa	10%	80%	10%
Sección Administrativa			
• Uso de menús	33.4%	33.4%	33.4%
• Perfiles	40%	30%	30%
• Estados	33.4%	33.4%	33.4%
• CRUD Provincia y Cantones	0%	100%	0%
• Consulta 1	25%	25%	50%
• Consulta 2	25%	50%	25%
• Consulta 3	50%	25%	25%
• Consulta 4	25%	50%	25%
• Consulta 5	0%	100%	0%
• Consulta 6	50%	25%	25%
• Consulta 7	33.4%	33.4%	33.4%
• Consulta 8	0%	0%	100%
• Consulta 9	100%	0%	0%

• Consulta 10	33.4%	33.4%	33.4%
Sección Usuario			
Registro Sistema	33.4%	33.4%	33.4%
Inicio de sesión	33.4%	33.4%	33.4%
CRUD Vehículos	33.4%	33.4%	33.4%
CRUD Evento Ciudadano	100%	0%	0%
Evento Oficial de Transito	33.4%	33.4%	33.4%
Evento Oficina de Juzgado	0%	0%	100%