

# Proyecto Agente inteligente cortinas

Brandon Isaac Arellano Calderón<sup>[L00412284]</sup>

Universidad de las Fuerzas Armadas  
biarellano@espe.edu.ec

---

## Abstract.

En este ensayo, se presenta un programa informático para el metro de Quito, con el objetivo de mejorar la eficiencia y la experiencia de los pasajeros en el uso del servicio. El programa se basa en el lenguaje de programación Python para brindar una variedad de funciones, incluyendo el ingreso de saldo y conteo de viajes para los usuarios, y la gestión de trenes y chóferes para el personal administrativo.

---

## 1 Introducción

La optimización de los sistemas de transporte se ha vuelto una tarea crucial en la actualidad debido al creciente tráfico y congestión vehicular. Para abordar este problema, se ha desarrollado un agente inteligente de tráfico que monitorea el estado del tráfico en las carreteras y proporciona recomendaciones a los conductores para minimizar los tiempos de viaje y mejorar la seguridad vial.

Este artículo presenta una descripción detallada del agente inteligente de tráfico, incluyendo su funcionamiento y desarrollo. El agente inteligente es capaz de monitorear cuatro carreteras importantes en la ciudad de Quito: Quito-Sto. Domingo, Quito-Cayambe, Quito-Papallacta y Quito-Aloag. El sistema permite a los conductores conocer si existe o no tráfico en estas carreteras y proporciona recomendaciones en consecuencia. Además, se incluye una evaluación del costo de la acción para medir el impacto del sistema en la optimización del tráfico.

## 2 Desarrollo

ón de chóferes a los trenes, se podrá asignar chóferes a los trenes de manera El agente inteligente de tráfico se desarrolló en Python utilizando una estructura de código que permite una fácil comprensión y modificación. La funcionalidad del sistema se basa en una serie de funciones y procesos interconectados, cada uno con un propósito específico.

La función "bienvenida" proporciona una pantalla de bienvenida para el sistema, con una línea divisoria y un mensaje de bienvenida. La función agente<sub>inteligente</sub>es el corazón del sistema, donde se

El sistema funciona de la siguiente manera: el usuario es requerido para ingresar si existe o no tráfico en cada una de las cuatro carreteras. Si el usuario

ingresa un valor "1", se indica que existe tráfico en la carretera correspondiente y se proporciona una recomendación para el conductor. Si el usuario ingresa un valor "0", se indica que no existe tráfico en la carretera correspondiente y se proporciona una recomendación para el conductor.

Finalmente, el sistema proporciona una evaluación del costo de la acción, lo que permite medir el impacto del sistema en la optimización del tráfico.

**Compresión del problema** El problema abordado en este desarrollo es la monitorización del tráfico en cuatro carreteras en la ciudad de Quito. Se desarrolló un agente inteligente que permite a los usuarios conocer el estado actual del tráfico en cada una de estas carreteras y, en caso de existir tráfico, recibir recomendaciones para tomar medidas necesarias. La solución está basada en un sistema que permite ingresar la información sobre el estado del tráfico y procesarla para brindar una respuesta a los usuarios. Además, se ha implementado una función que brinda un costo a la acción de tomar medidas en caso de existir tráfico. Este sistema es de gran utilidad para aquellos que viajan en vehículos y desean conocer el estado del tráfico antes de salir a la carretera.

**Formular un modelo** un modelo en relación al código anterior, podemos decir que se está trabajando con un problema de seguimiento del tráfico en las carreteras de la ciudad de Quito, Ecuador. Se ha creado un agente inteligente que se encarga de recopilar información sobre el estado de tráfico en diferentes carreteras y proporcionar recomendaciones al usuario basadas en dicha información.

El modelo se basa en una serie de funciones que permiten la interacción con el usuario. La función "bienvenida" proporciona un membrete de bienvenida, mientras que la función "agente<sub>inteligente</sub>" *es el corazón del modelo. Aquí se lleva a cabo el procesamiento de información*

El modelo se ejecuta en un bucle infinito, lo que permite al usuario ingresar la información sobre el estado de tráfico y obtener recomendaciones de manera repetitiva hasta que decida detener la ejecución del programa; este modelo se concentra en brindar información actualizada y recomendaciones precisas sobre el estado de tráfico en la ciudad de Quito, a través de una interfaz amigable y fácil de usar.

- Recopilación y análisis de los datos necesarios para el proyecto, como la información sobre los trenes disponibles, los horarios de operación, la cantidad de usuarios y su frecuencia de viaje, entre otros.
- Diseño de un sistema de gestión de personal administrativo que permita la asignación de tareas, la cronometración del tiempo y la verificación de trenes disponibles.
- Implementación de un sistema de validación de códigos de entrada para los usuarios con problemas, y una función de registro y de inicio de sesión para los usuarios.
- Utilizar python para desarrollar el sistema, ya que es un lenguaje de programación con una gran cantidad de librerías y herramientas para el análisis de datos y la automatización de tareas.

- Prueba y evaluación del sistema, para asegurar su funcionamiento correcto y detectar posibles errores o problemas.
- Implementación del sistema en el metro de Quito y monitoreo continuo para asegurar su eficiencia y para identificar áreas de mejora.
- Ajustes y actualizaciones periódicas para mantener el sistema actualizado y adaptado a las necesidades cambiantes del metro.

**Desarrollar un algoritmo** Una vez que se ha comprendido el problema y se ha formulado un modelo de resolución, es necesario seguir una serie de instrucciones para poder codificar el programa. Para ayudar en este proceso se utiliza el pseudocódigo, el cual es una herramienta que permite detallar los pasos que el programa debe llevar a cabo cuando es codificado en un lenguaje de programación que la computadora pueda entender y ejecutar. El pseudocódigo es especialmente útil en la etapa temprana del desarrollo de un programa, ya que permite planificar el proceso de codificación de manera clara y ordenada, facilitando el trabajo del programador.

**Escribe el programa** En la presente La función "bienvenida" es una función que no tiene parámetros de entrada y no retorna ningún valor. La función imprime en pantalla un membrete de bienvenida que consiste en 56 caracteres '=' seguidos de una línea con el mensaje "Bienvenido al Agente inteligente Trafico" y luego otros 56 caracteres '='. Este membrete es una forma de saludar y dar una presentación visual clara para el usuario.

**Listing 1.1.** Función bienvenida

```
def bienvenida():
    '''
    Parametro
    -----
    No ingresa ningun parametro

    Retorna
    -----
    No retorna nada

    Funcion
    -----
    Nos da un membrete de bienvenida
    '''
    #Imprime el la cantidad de 56 '*'
    print("*"*56)
    #imprime una bienvenida
    print(" Bienvenido al Sistema de Transporte del Metro de Quito")
```

```
#nos da un aviso
print("A continuacion te mostraremos las diferentes
opciones que puedes hacer")
```

Por consiguiente la funcion menu se encuentra , la funcion mencionada contiene un menu principal el cual será mostrado al usuario para que el escoga el item que necesita lo cual guarda su opcion y lo retorna para que sea ocupado por alguna otra funcion.

**Listing 1.2.** Función menu

```
def agente_inteligente():
    '''
    Parametro
    -----
    Recibe ningun parametro

    Retorna
    -----
    No retorna nada

    Funcion
    -----
    En esta funcion desarrolla el estado_aspiradora de la
    aspiradora y los estados que tiene esta
    '''
    diccionario_trafico = {"Quito--Sto. Domingo" : '0',
        "Quito--Cayambe": '0', 'Quito--Papallacta': '0',
        'Quito--Aloag': '0'}

    print("Si tiene trafico ingrese 1 caso contrario 0")
    carretera1 = input("Existe trafico en la via Quito--
        Sto. Domingo: ")
    carretera2 = input("Existe trafico en la via Quito--
        Cayambe: ")
    carretera3 = input("Existe trafico en la via Quito--
        Papallacta: ")
    carretera4 = input("Existe trafico en la via Quito--
        Aloag: ")
    accion = 0

    print("="*44)
    if carretera1 == '1':
        print("En la ruta Quito--Sto. Domingo existe
            trafico")
        print("Recomendamos tomar las medidas pertinentes
            ")
```

```

        accion = accion +1

    if carretera2 == '1':
        print("En la ruta Quito--Cayambe existe trafico ")
        print("Recomendamos tomar las medidas pertinentes ")
        accion = accion +1

    if carretera3 == '1':
        print("En la ruta Quito--Papallacta existe trafico ")
        print("Recomendamos tomar las medidas pertinentes ")
        accion = accion +1

    if carretera4 == '1':
        print("En la ruta Quito--Aloag existe trafico ")
        print("Recomendamos tomar las medidas pertinentes ")
        accion = accion +1

    if carretera1 == '0':
        print("En la ruta Quito--Sto. Domingo no existe trafico ")
        print("Maneje con cuidado y un excelente viaje")

    if carretera2 == '0':
        print("En la ruta Quito--Cayambe no existe trafico ")
        print("Maneje con cuidado y un excelente viaje")
    if carretera3 == '0':
        print("En la ruta Quito--Papallacta no existe trafico ")
        print("Maneje con cuidado y un excelente viaje")
    if carretera4 == '0':
        print("En la ruta Quito--Aloag no existe trafico ")
        print("Maneje con cuidado y un excelente viaje")

    print("El costo de la accion es de: ", accion)

```

Y por finaliar en donde es la cabeza del proyecto el main , aquí se encuentra llamada a las funciones para el funcionamiento del código.

**Listing 1.3.** Main

```

if __name__ == "__main__":

```

```

diccionario_trafico = {"Quito_--Sto._Domingo" : '0',
                        "Quito_--Cayambe": '0', 'Quito_--Papallacta': '0',
                        'Quito_--Aloag': '0'}
#llama a la fncion bienvenida
bienvenida()

while True:
    #llama a la funcion accion y como parametro la
    funcion menu
    agente_inteligente()
    print("="*44)
    input("presione_enter_para_continuar")

    os.system('cls')

```

### Pruebe el programa

**Evalué la solución** Después de evaluar el modelo de resolución propuesto, se puede concluir que es una estrategia viable para optimizar la operación del metro de Quito y mejorar la experiencia de los usuarios. El diseño del sistema de gestión de personal administrativo, la implementación de un sistema de validación de códigos de entrada y una función de registro y inicio de sesión para los usuarios son medidas necesarias para garantizar una operación eficiente del metro. Utilizar Python para desarrollar el sistema es una buena opción debido a las numerosas librerías y herramientas disponibles para el análisis de datos y la automatización de tareas. La realización de pruebas y evaluaciones para asegurar el correcto funcionamiento del sistema, así como el monitoreo continuo y los ajustes periódicos son medidas necesarias para garantizar su eficiencia y adaptabilidad a las necesidades cambiantes del metro de Quito. Por lo que puedo afirmar que este modelo de resolución es una estrategia sólida y viable para mejorar la operación del metro de Quito.

A continuacion se mostrara las siguien figuras las cuales mostraran las pruebas del programa.

### 3 Discusión

En la ciudad de Quito, un problema importante es el tráfico en las carreteras. Para solucionar este problema, se desarrolló un agente inteligente para monitorear el tráfico en cuatro carreteras específicas. Este agente permite a los usuarios obtener información actualizada sobre el estado del tráfico y, en caso de haber congestión, recibir recomendaciones para tomar medidas apropiadas.

El sistema se basa en una entrada de información sobre el tráfico y un procesamiento de dicha información para brindar una respuesta útil a los usuarios. Además, la implementación de una función que asigna un costo a la acción de

tomar medidas en caso de congestión de tráfico añade un elemento de incenti-vación para que los usuarios actúen.

En general, este sistema es muy útil para aquellos que desean planificar sus viajes en vehículos y tener una idea del estado actual del tráfico antes de salir a la carretera. De esta manera, pueden tomar decisiones informadas y reducir la posibilidad de encontrar congestión de tráfico inesperada.

Para más detalles sobre el proyecto se puede visitar el siguiente link: [https://github.com/BrandonArellanoC/Agente\\_Inteligente](https://github.com/BrandonArellanoC/Agente_Inteligente), donde se encuentra subido a Github con el código fuente y documentación para su consulta y estudio. En general, el programa informático propuesto para el metro de Quito tiene el potencial de mejorar significativamente la eficiencia y la experiencia de los pasajeros en el uso del servicio de metro.

## 4 Conclusiones

A partir del desarrollo de un agente inteligente de monitoreo de tráfico en cuatro carreteras de la ciudad de Quito, se puede concluir que esta solución ofrece una mayor eficiencia y comodidad para los usuarios que desean conocer el estado actual del tráfico antes de iniciar su viaje en vehículo. La implementación del sistema permite ingresar la información sobre el tráfico y procesarla para brindar una respuesta precisa y útil a los usuarios. Además, la función que brinda un costo a la acción de tomar medidas en caso de existir tráfico, es un añadido valioso que permite tomar decisiones más informadas.

El agente inteligente de monitoreo de tráfico en la ciudad de Quito es una solución innovadora y útil que mejora la experiencia de los usuarios y les brinda información valiosa para tomar decisiones más informadas sobre su viaje en vehículo.