**Public Transport Opitimization**

**Introduction :**

Efficient public transportation systems play a pivotal role in the growth and sustainability of urban areas. They offer numerous benefits, including reduced traffic congestion, lower carbon emissions, improved accessibility, and increased mobility for the community. However, many public transportation systems face challenges such as inefficient routes, overcrowding, and outdated technology, leading to a decline in overall service quality. To address these issues, we are embarking on a comprehensive Public Transportation Optimization Project.

**Platform Developed :**

We have developed in stimulation platform.

**Code :**

**# Import necessary libraries and modules**

**Import random**

**Import time**

**Class Vehicle:**

**Def \_\_init\_\_(self, vehicle\_id):**

**Self.vehicle\_id = vehicle\_id**

**Self.latitude = random.uniform(40.0, 41.0)**

**Self.longitude = random.uniform(-74.0, -73.0)**

**Def update\_location(self):**

**# Simulate GPS data updates**

**Self.latitude += random.uniform(-0.001, 0.001)**

**Self.longitude += random.uniform(-0.001, 0.001)**

**Def get\_location(self):**

**Return {“vehicle\_id”: self.vehicle\_id, “latitude”: self.latitude, “longitude”: self.longitude}**

**# Simulate multiple vehicles**

**Vehicles = [Vehicle(1), Vehicle(2), Vehicle(3)]**

**# Main loop to continuously update and send vehicle locations**

**While True:**

**For vehicle in vehicles:**

**Vehicle.update\_location()**

**Location\_data = vehicle.get\_location()**

**# Send location\_data to your IoT cloud platform or API**

**Print(location\_data) # Simulate data transmission**

**Time.sleep(10) # Update every 10 seconds**

**Passenger count:**

**# Initialize a passenger counter**

**Passenger\_count = 0**

**# Simulate passenger boarding and disembarking**

**Boarded\_passengers = [3, 2, 4, 1, 5] # Example: Passengers boarding at each stop**

**Disembarked\_passengers = [0, 1, 2, 3, 1] # Example: Passengers disembarking at each stop**

**# Main loop for tracking passenger count at each stop**

**For I in range(len(boarded\_passengers)):**

**# Update passenger count with boarding and disembarking**

**Passenger\_count += boarded\_passengers[i]**

**Passenger\_count -= disembarked\_passengers[i]**

**# Print the passenger count at each stop**

**Print(f”Passenger count at Stop {I + 1}: {passenger\_count}”)**

**Conclusion :**

This project seeks to transform public transportation into a more efficient, accessible, and sustainable mode of travel. By optimizing routes, schedules, and technology, we aim to enhance the overall transportation experience for the community while addressing congestion and environmental concerns. The success of this project will result in a more vibrant, sustainable, and accessible urban environment for all residents.