

## Classes and Components:

### Vehicle

* **Attributes:**
  + id: A unique identifier for each vehicle.
  + speed: The current speed of the vehicle.
  + position: The current position of the vehicle in the simulation, represented by coordinates.
  + destination: The target destination for the vehicle.
  + state: The current state of the vehicle (e.g., moving, stopped).
* **Methods:**
  + move(): Updates the vehicle's position based on its speed and direction.
  + stop(): Halts the vehicle's movement.
  + updatePosition(): Updates the vehicle's position in the simulation based on current speed and trajectory.
* **Interactions:**
  + A Vehicle travels on a Road and interacts with TrafficSignal objects at intersections, stopping when needed and moving according to traffic signals.

### TrafficSignal

* **Attributes:**
  + id: A unique identifier for each traffic signal.
  + state: The current state of the traffic signal (e.g., red, yellow, green).
  + location: The location of the traffic signal in the simulation, represented by coordinates.
* **Methods:**
  + changeState(): Changes the state of the traffic signal (e.g., from red to green).
  + getState(): Returns the current state of the traffic signal.
* **Interactions:**
  + A TrafficSignal manages the flow of traffic at intersections. It interacts with Vehicle objects by controlling when they should stop or proceed.

### Road

* **Attributes:**
  + id: A unique identifier for the road.
  + length: The length of the road.
  + lanes: The number of lanes available on the road.
  + start: The starting coordinate of the road.
  + end: The ending coordinate of the road.
* **Methods:**
  + connectRoad(): Connects the road to another road or intersection.
  + addVehicle(Vehicle): Adds a vehicle to the road.
* **Interactions:**
  + A Road connects to Intersection objects and serves as the pathway for Vehicle objects to move along. It manages the vehicles traveling on it.

### Intersection

* **Attributes:**
  + id: A unique identifier for the intersection.
  + connectedRoads: A list of Road objects that are connected to this intersection.
  + trafficSignal: The TrafficSignal that controls the intersection.
* **Methods:**
  + manageTraffic(): Manages the flow of vehicles based on the state of the traffic signal and other conditions.
* **Interactions:**
  + An Intersection serves as a junction where multiple Road objects connect. It interacts with Vehicle objects and uses the TrafficSignal to manage vehicle flow.

### Environment

* **Attributes:**
  + vehicles: A list of all Vehicle objects present in the simulation.
  + roads: A list of all Road objects.
  + intersections: A list of all Intersection objects.
* **Methods:**
  + update(): Updates the entire environment, including vehicle positions, traffic signal states, and interactions between objects.
  + addVehicle(Vehicle): Adds a vehicle to the environment.
  + addTrafficSignal(TrafficSignal): Adds a traffic signal to the environment.
* **Interactions:**
  + The Environment class encapsulates the entire simulation. It contains all vehicles, roads, and intersections, and manages the overall behavior of the system. It updates the system at each time step, ensuring vehicles move, traffic signals change, and interactions occur.

## Component Interactions:

* **Vehicle to Road**: Vehicles travel along roads, and the road ensures the vehicle stays within its bounds and follows its path.
* **Vehicle to TrafficSignal**: Vehicles interact with traffic signals to determine whether they should stop or continue driving.
* **Road to Intersection**: Roads connect at intersections, which are controlled by traffic signals.
* **Intersection to TrafficSignal**: The intersection uses traffic signals to manage the flow of vehicles passing through it.
* **Environment to All Components**: The environment orchestrates the overall simulation, managing vehicles, roads, and intersections.

## Alignment with Project Goals:

The UML diagram and this design documentation reflect the goals of this agent-based traffic simulation model, where each vehicle (agent) interacts with other vehicles, traffic signals, and roads to mimic real-world traffic flow. The use of intersections and traffic signals ensures the simulation can model traffic congestion, signal timing, and traffic behavior. The design provides a framework for further enhancements such as adding pedestrian interactions, emergency vehicle prioritization, or variable traffic signal timing.