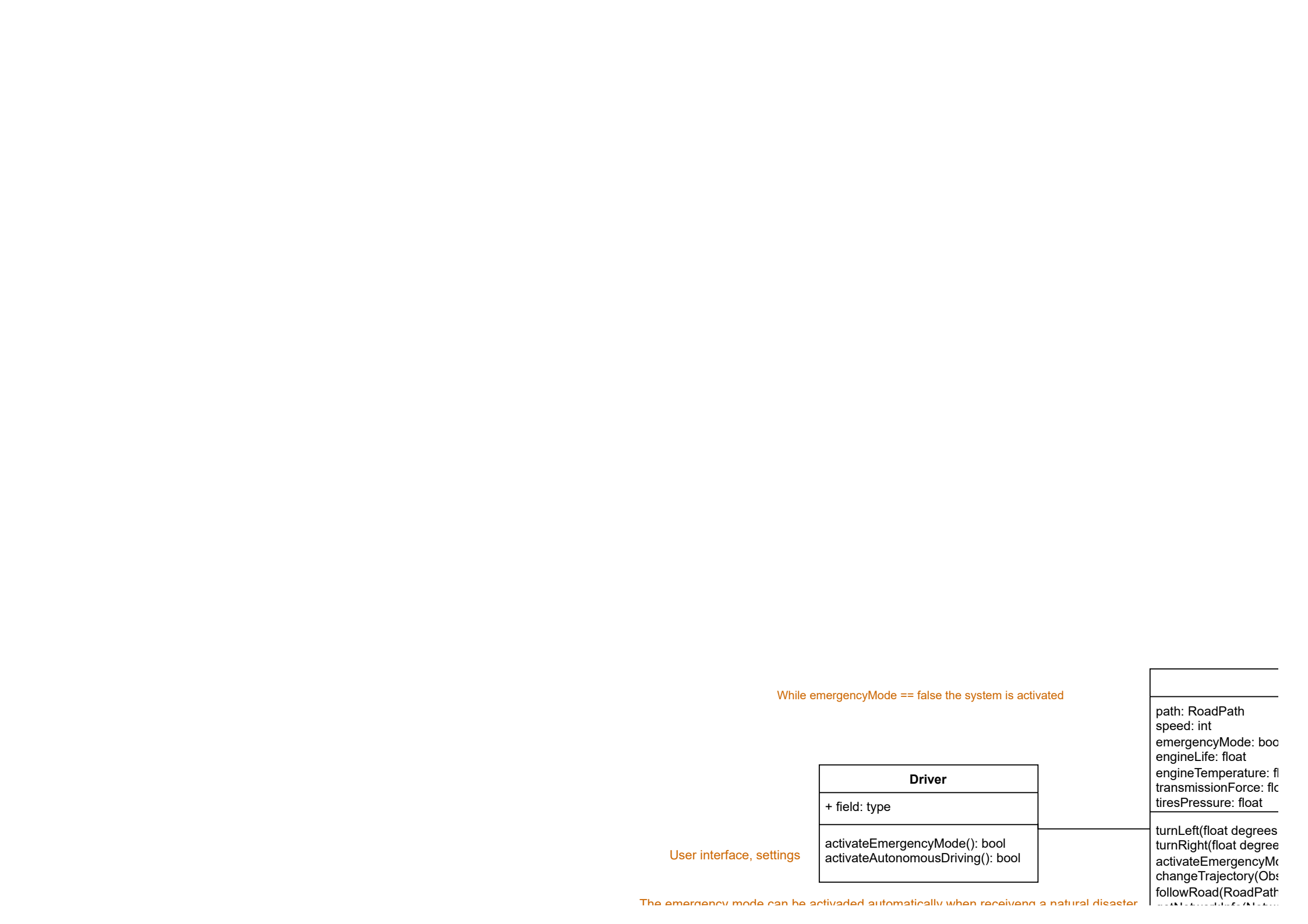
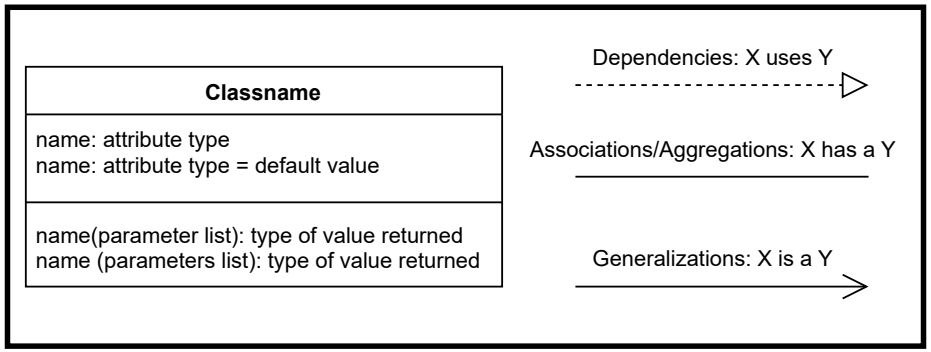
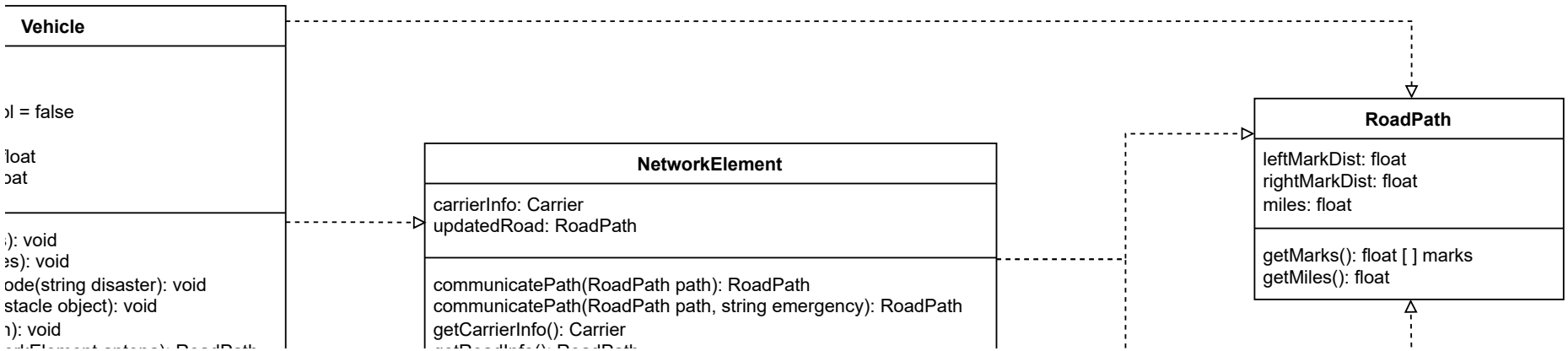

While emergencyMode == false the system is activated
Driver
+ field: type
activateEmergencyMode(): bool
activateAutonomousDriving(): bool
User interface, settings
path: RoadPath
speed: int
emergencyMode: boo
engineLife: float
engineTemperature: fl
transmissionForce: flo
tiresPressure: float
turnLeft(float degrees
turnRight(float degree
activateEmergencyMo
changeTrajectory(Obs
followRoad(RoadPath
The emergency mode can be activaded automatically when receiveng a natural disaster


Classname
name: attribute type
name: attribute type = default value
name(parameter list): type of value returned
name (parameters list): type of value returned
Dependencies: X uses Y
Associations/Aggregations: X has a Y
Generalizations: X is a Y


Vehicle
NetworkElement
carrierInfo: Carrier
updatedRoad: RoadPath
communicatePath(RoadPath path): RoadPath
communicatePath(RoadPath path, string emergency): RoadPath
getCarrierInfo(): Carrier
RoadPath
leftMarkDist: float
rightMarkDist: float
miles: float
getMarks(): float [ ] marks
getMiles(): float

The emergency mode can be activaded automatically when receiveng a natural disaster

saveAnalytics will store the information of the car in a database

getNetworkInfo(Netwo
saveAnalytics(): void
throttle(): void
break(): void

Bool to return if an obstacle is considered as a danger

Det

scannedObstacl
sensors: Sensor

getScannedObs
isDanger(Obsta
comunicateDan
scanEnvironmer

width: flo
height: fl
distance

getWidth
setWidth
getHeigh
setHeigh

Camera

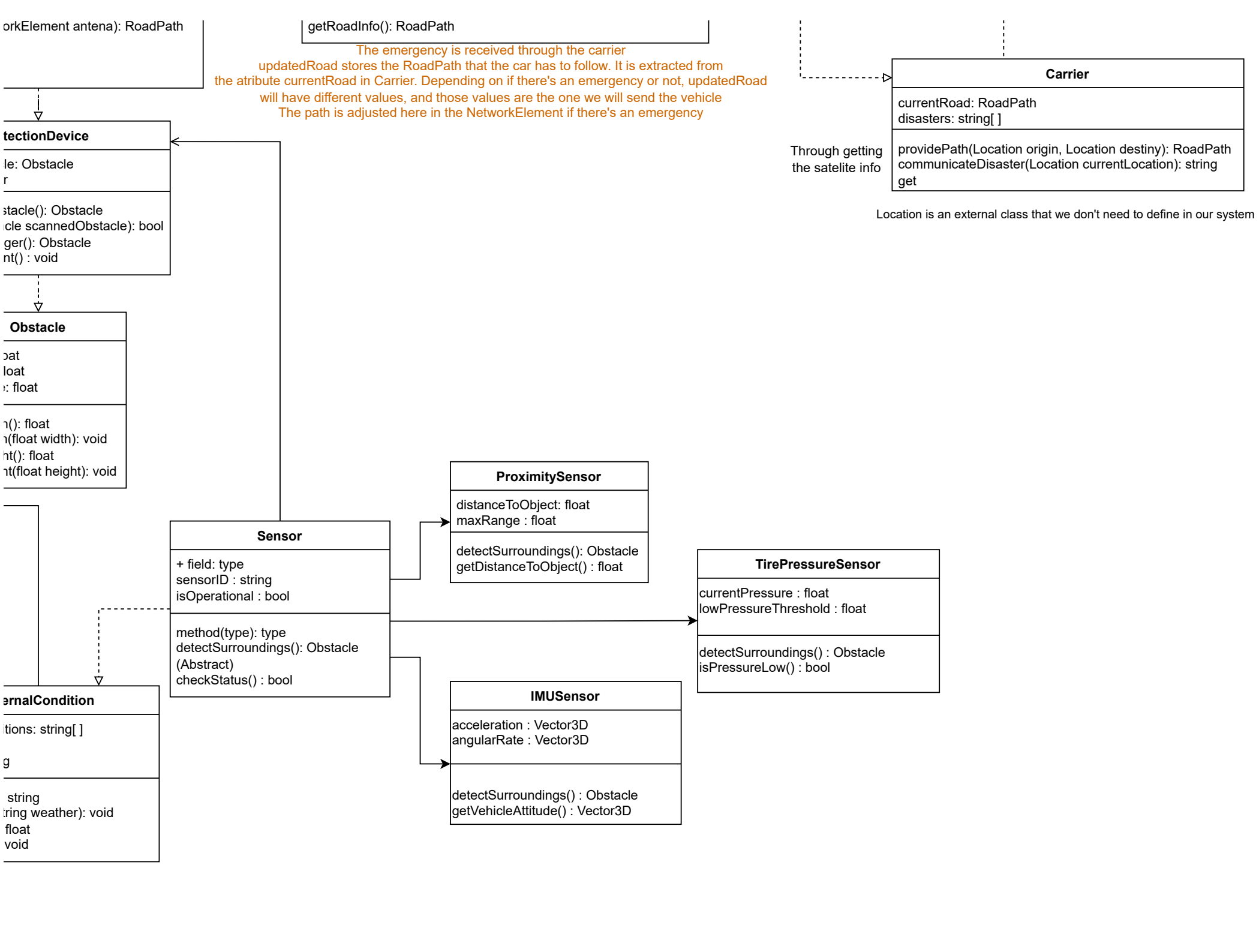
+ field: type
esolution : string
fieldOfView : float

detectSurroundings(): Obstacle
captureFrame() : ImageData

Exte

weatherCondi
visibility: float
weather: string

getWeather():
setWeather(st
getVisibility():
setVisibility():


orkElement antena): RoadPath
getRoadInfo(): RoadPath
The emergency is received through the carrier
updatedRoad stores the RoadPath that the car has to follow. It is extracted from
the atribute currentRoad in Carrier. Depending on if there's an emergency or not, updatedRoad
will have different values, and those values are the one we will send the vehicle
The path is adjusted here in the NetworkElement if there's an emergency
Carrier
currentRoad: RoadPath
disasters: string[ ]
providePath(Location origin, Location destiny): RoadPath
communicateDisaster(Location currentLocation): string
get
Through getting
the satelite info
Location is an external class that we don't need to define in our system
tectionDevice
le: Obstacle
r
stacle(): Obstacle
cle scannedObstacle): bool
ger(): Obstacle
nt() : void
Obstacle
oat
loat
: float
(): float
(float width): void
ht(): float
nt(float height): void
Sensor
+ field: type
sensorID : string
isOperational : bool
method(type): type
detectSurroundings(): Obstacle
(Abstract)
checkStatus() : bool
ProximitySensor
distanceToObject: float
maxRange : float
detectSurroundings(): Obstacle
getDistanceToObject() : float
TirePressureSensor
currentPressure : float
lowPressureThreshold : float
detectSurroundings() : Obstacle
isPressureLow() : bool
IMUSensor
acceleration : Vector3D
angularRate : Vector3D
detectSurroundings() : Obstacle
getVehicleAttitude() : Vector3D
ernalCondition
tions: string[ ]
g
string
ring weather): void
float
void