Vehicle Simulator and API Requirements and Design

(Updated as of 3/8/19) (Code reflected in BitBucket in *Vehicle Supply* Repository)

**Vehicle Simulator:**

Requirements:

* Receive travel specifications from Vehicle API (Format: Vehicleinfo, Route *as of 3/8/19*)
  + Reflected in startEngine(vehicleSend) function
* Create/enact dynamic route
  + Set vehicle information
    - State “deployed” in Vehicle object
    - Initialize variables and position for vehicle
  + Parse occurs in startEngine function of both vehicle info and map routing from vehicleSend (Creates legLat, legLong,legTime Format: Array/List (float,int) *as of 3/8/19*)
  + Begin dynamic travel (Run travelRoute(legpos, legtime))
* Dynamically travel route
  + Iterate in time based intervals
    - Change Vehicle object position in legTime[index] amount of time to legLat[index], legLong[index].
    - Update also occurs every approximate minute of elapsed time
    - Update through updateVApi(vId, vIn, vType, vLat, vLong, vState, fId) (Fomat: Int,Int,String,Float,Float,String,Int *as of 3/8/19*)
  + Update on arrival
    - State “arrived” in Vehicle object
    - Reset
  + Reflected in travelRoute function
* Update Vehicle API during travel and destination
  + Run updateDispatch(vId, vIn, vType, vLat, vLong, vState, fId) function reflected in Vehicle API (Fomat: Int,Int,String,Float,Float,String,Int *as of 3/8/19*)

**Vehicle API:**

Requirements:

* Receive and process order information from Dispatch API
  + Retrieve order information through dispatch database (Every approximate minute time interval) (SELECT VID FROM DispatchRecord WHERE isRouteDone LIKE false *as of 3/8/19*)
  + Reflected in deploy() function
* Deploy simulated vehicle
  + Send request to VehicleSimulator.startEngine() with information retrieved (Format: HTTP Request format with vehicle info & route *as of 3/8/19*)
  + Reflected in deploy()
* Update dispatch record on Vehicle information
  + Intialize SQL query (UPDATE Vehicle SET VID = " + vId + ", VIN = " + vIn

+ ", VType = " + vType + ", VLat = " + vLat + ", VLong = " + vLong + ", VState = " + vState + ", FID = " + fId + " WHERE VID = " + vId *as of 3/8/19*)

* + Query dispatch record
  + Time interval same as updateVApi reflected in Vehicle Simulator

**Sequence of Events:**

**Dispatch Supply**

SQL Query(deploy())

**Vehicle API**

updatedispatch() SQL Query

* deploy()
* updatedispatch()

\

updatevapi() HTTP Request

startengine() HTTPS Request

**Vehicle Simulator**

* startengine()
* travelroute()
* updatevapi()