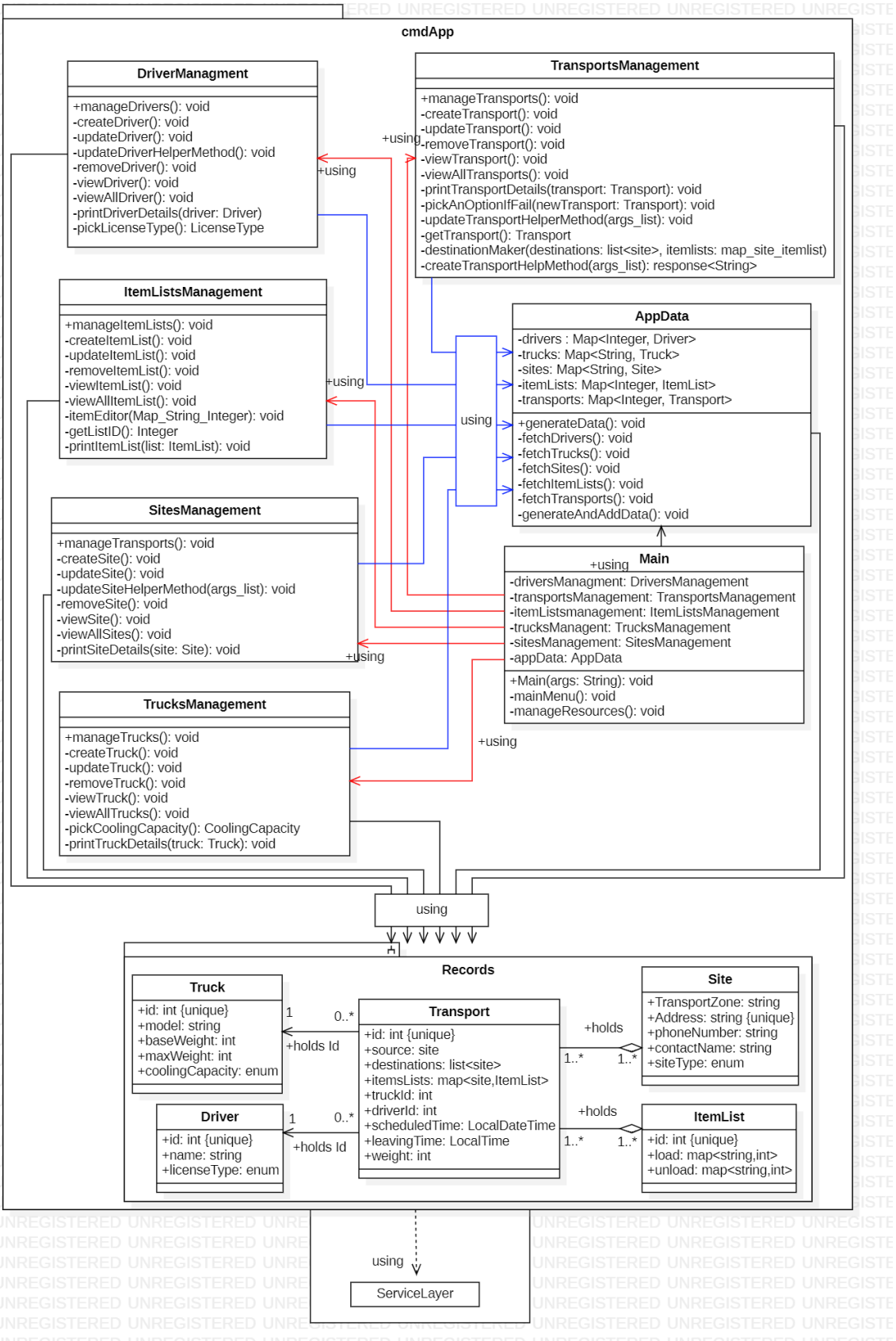
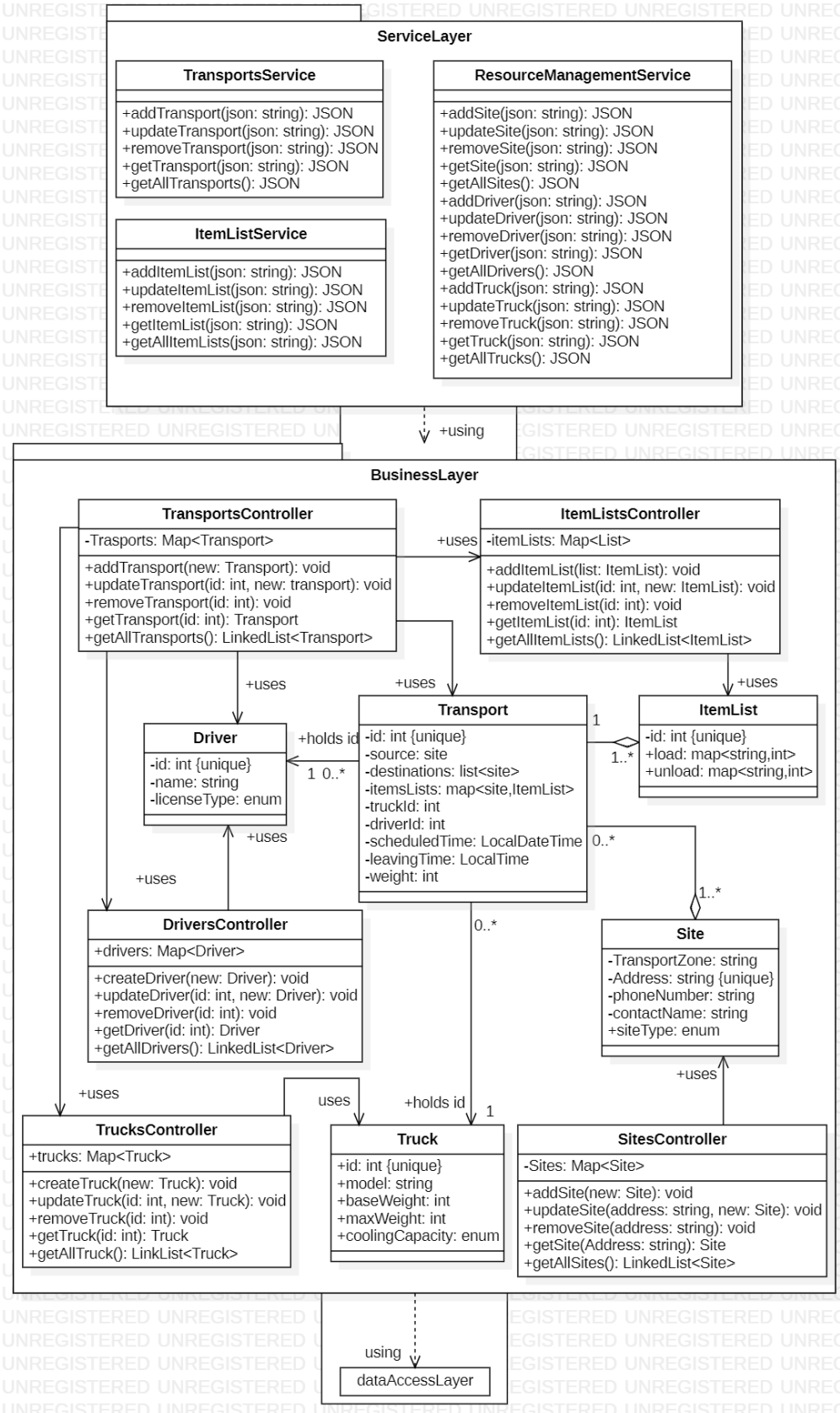
Class diagram




ServiceLayer
TransportsService
+addTransport(json: string): JSON
+updateTransport(json: string): JSON
+removeTransport(json: string): JSON
+getTransport(json: string): JSON
+getAllTransports(): JSON
ItemListService
+addItemList(json: string): JSON
+updateItemList(json: string): JSON
+removeItemList(json: string): JSON
+getItemList(json: string): JSON
+getAllItemLists(json: string): JSON
ResourceManagementService
+addSite(json: string): JSON
+updateSite(json: string): JSON
+removeSite(json: string): JSON
+getSite(json: string): JSON
+getAllSites(): JSON
+addDriver(json: string): JSON
+updateDriver(json: string): JSON
+removeDriver(json: string): JSON
+getDriver(json: string): JSON
+getAllDrivers(): JSON
+addTruck(json: string): JSON
+updateTruck(json: string): JSON
+removeTruck(json: string): JSON
+getTruck(json: string): JSON
+getAllTrucks(): JSON
+using
BusinessLayer
TransportsController
-Trasports: Map<Transport>
+addTransport(new: Transport): void
+updateTransport(id: int, new: transport): void
+removeTransport(id: int): void
+getTransport(id: int): Transport
+getAllTransports(): LinkedList<Transport>
+uses
ItemListsController
-itemLists: Map<List>
+addItemList(list: ItemList): void
+updateItemList(id: int, new: ItemList): void
+removeItemList(id: int): void
+getItemList(id: int): ItemList
+getAllItemLists(): LinkedList<ItemList>
Driver
-id: int {unique}
-name: string
-licenseType: enum
+holds id
1 0..*
Transport
-id: int {unique}
-source: site
-destinations: list<site>
-itemsLists: map<site,ItemList>
-truckId: int
-driverId: int
-scheduledTime: LocalDateTime
-leavingTime: LocalTime
-weight: int
1
1..*
ItemList
-id: int {unique}
+load: map<string,int>
+unload: map<string,int>
0..*
1..*
DriversController
+drivers: Map<Driver>
+createDriver(new: Driver): void
+updateDriver(id: int, new: Driver): void
+removeDriver(id: int): void
+getDriver(id: int): Driver
+getAllDrivers(): LinkedList<Driver>
Site
-TransportZone: string
-Address: string {unique}
-phoneNumber: string
-contactName: string
+siteType: enum
TrucksController
+trucks: Map<Truck>
+createTruck(new: Truck): void
+updateTruck(id: int, new: Truck): void
+removeTruck(id: int): void
+getTruck(id: int): Truck
+getAllTruck(): LinkList<Truck>
uses
+holds id
1
Truck
+id: int {unique}
+model: string
+baseWeight: int
+maxWeight: int
+coolingCapacity: enum
SitesController
-Sites: Map<Site>
+addSite(new: Site): void
+updateSite(address: string, new: Site): void
+removeSite(address: string): void
+getSite(Address: string): Site
+getAllSites(): LinkedList<Site>
using
dataAccessLayer

Object Diagram 1 :

The following diagram describes the transport that take place on

14-05-23, performe by driver named Moshe (his id = 0), the transport will be carried out by a Toyota truck whose identification number is 4354 and will be at 09:00 in the morning .

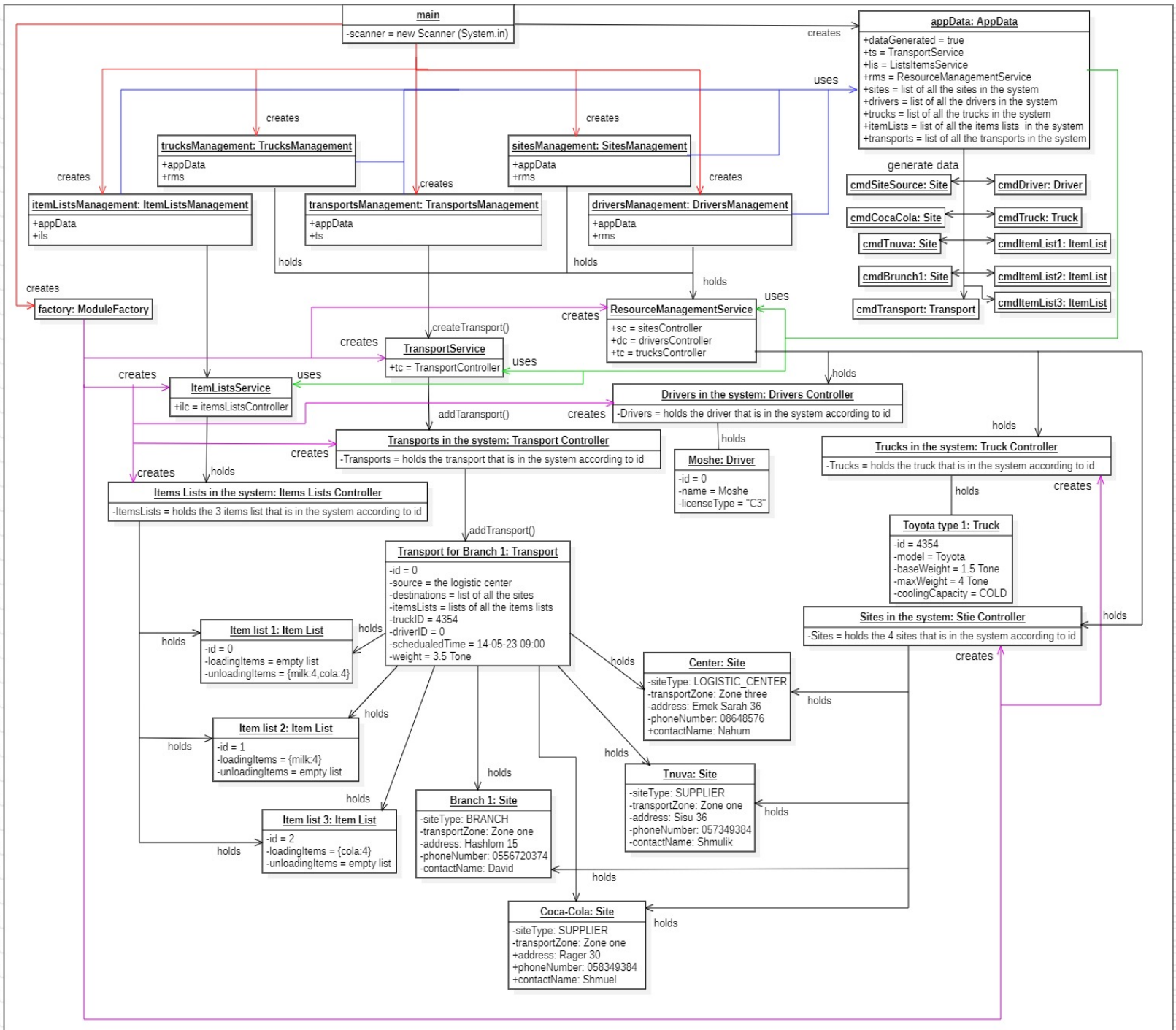
The following assumptions must be made about the scenario:

1) The driver has a driver's license suitable for the truck

2) The transport start from the logistics center

3) The system is described at the end of the transport - describes a snapshot of the system

4)It is assumed that all objects added to the transport exist in the system



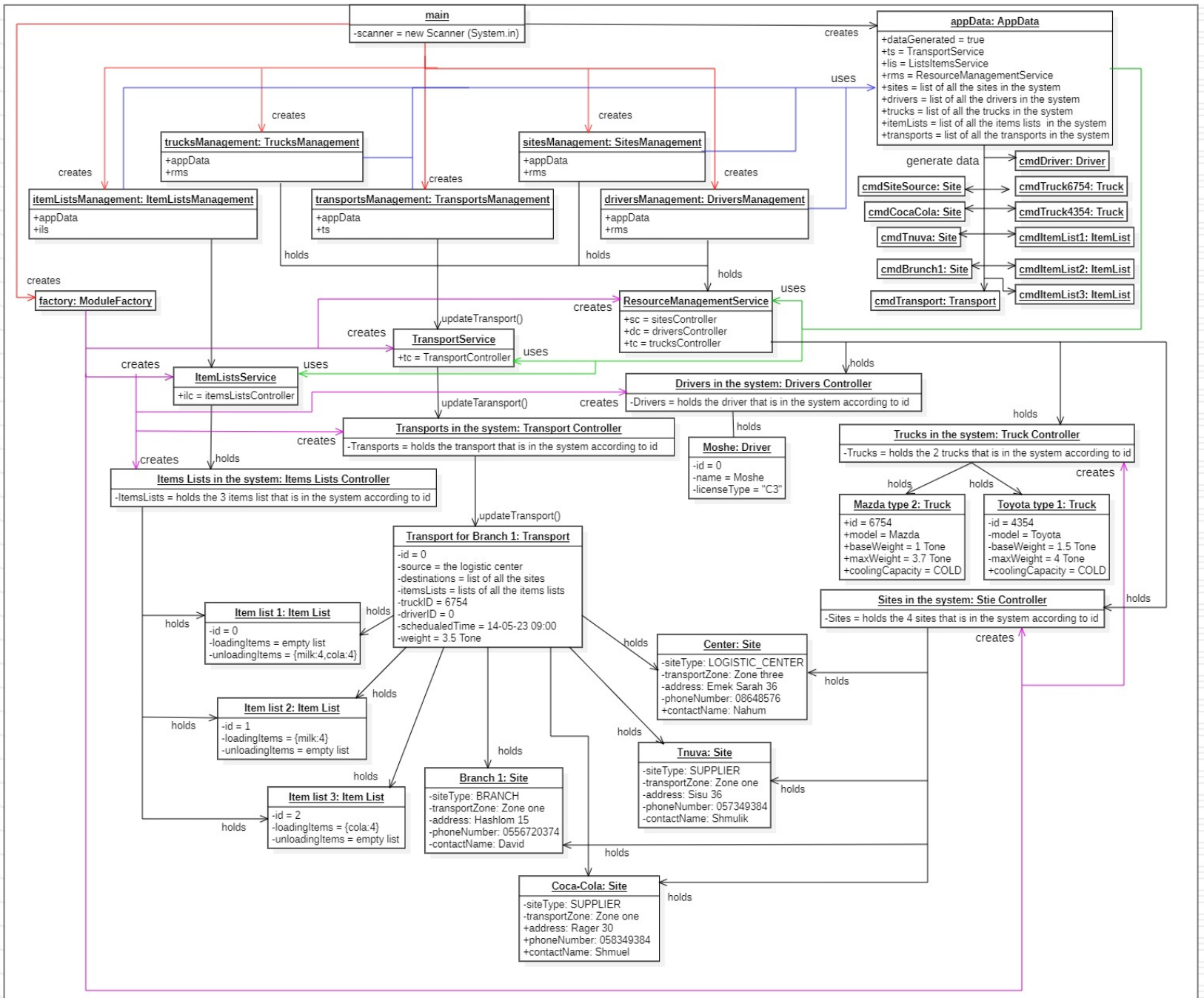
Object Diagram 2 :

On 13-05-23 the logistics center called Dani the transport manager and informed him that truck identification number 4354 had a flat tire, Dani looked in the system to see if there a truck that suitable for the transport that should be carried out on 14-05-23 and indeed found that truck number 6754 was suitable for that transport, so he updated in the system the new data about this transport.

The following assumptions must be made about the scenario:

1) The driver has a driver's license suitable for the new truck

2) The transport start from the logistics center

3) The system is described after the update - describes a snapshot of the system

4)It is assumed that truck number 6754 is exist in the system in addition to all the objects of the existing transport