CSE 522: Project

Tariq Siddiqui: 50208476

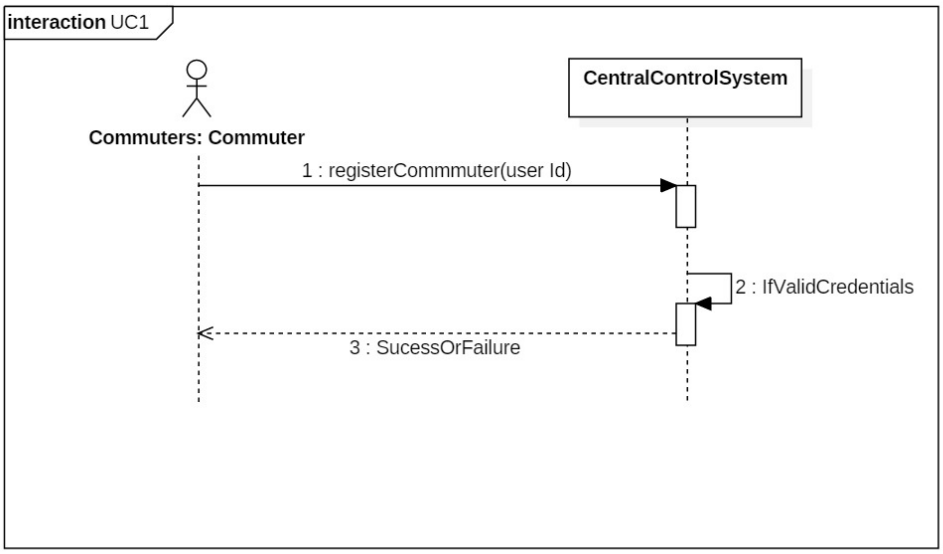
Junaid Shaikh: 50208476

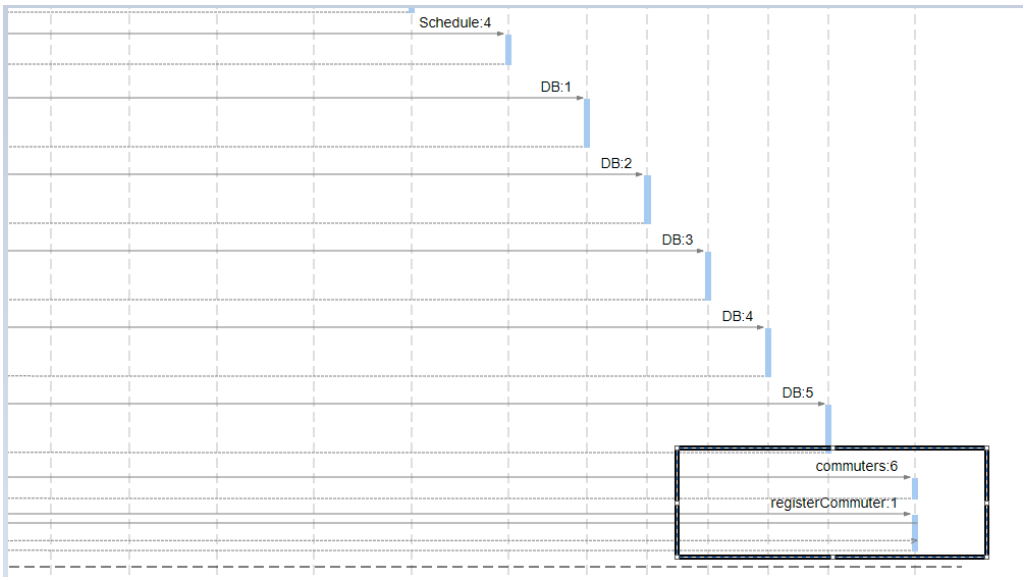
Use Case 1(UC1):

Summary:

Use Case 1 was about registering a new commuter to the database of commuters in the Central Control System. This was required since the next use case we want to implement, UC2, requires a commuter to retrieve the best route, fare, schedule and location updates for their intended trip.

As explained in the Use Case diagram, the commuter object calls the Central Control System to request to register himself to the database. The commuter is registered to the database by the CCS and control is passed back.

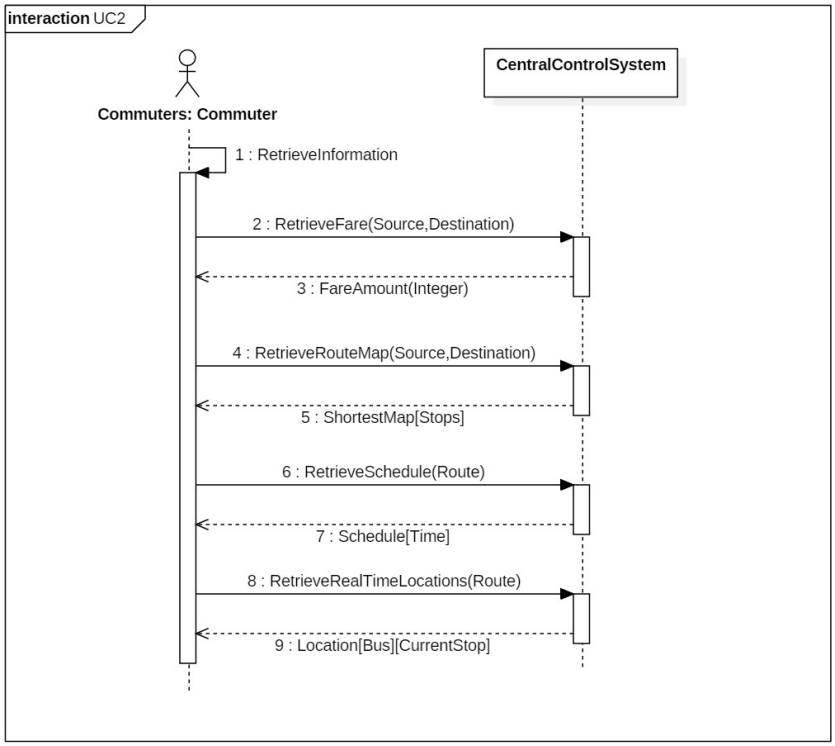




Use Case 2(UC2):

Summary:

The registered commuter wants to enquire about the best route from his source to intended destination. The object commuter sends a retrieveRouteMap, retrieveFare, retrieveSchedule and retrieveRealTimeLocation request to the CCS along with the source and destinations as shown in the use case diagram below.



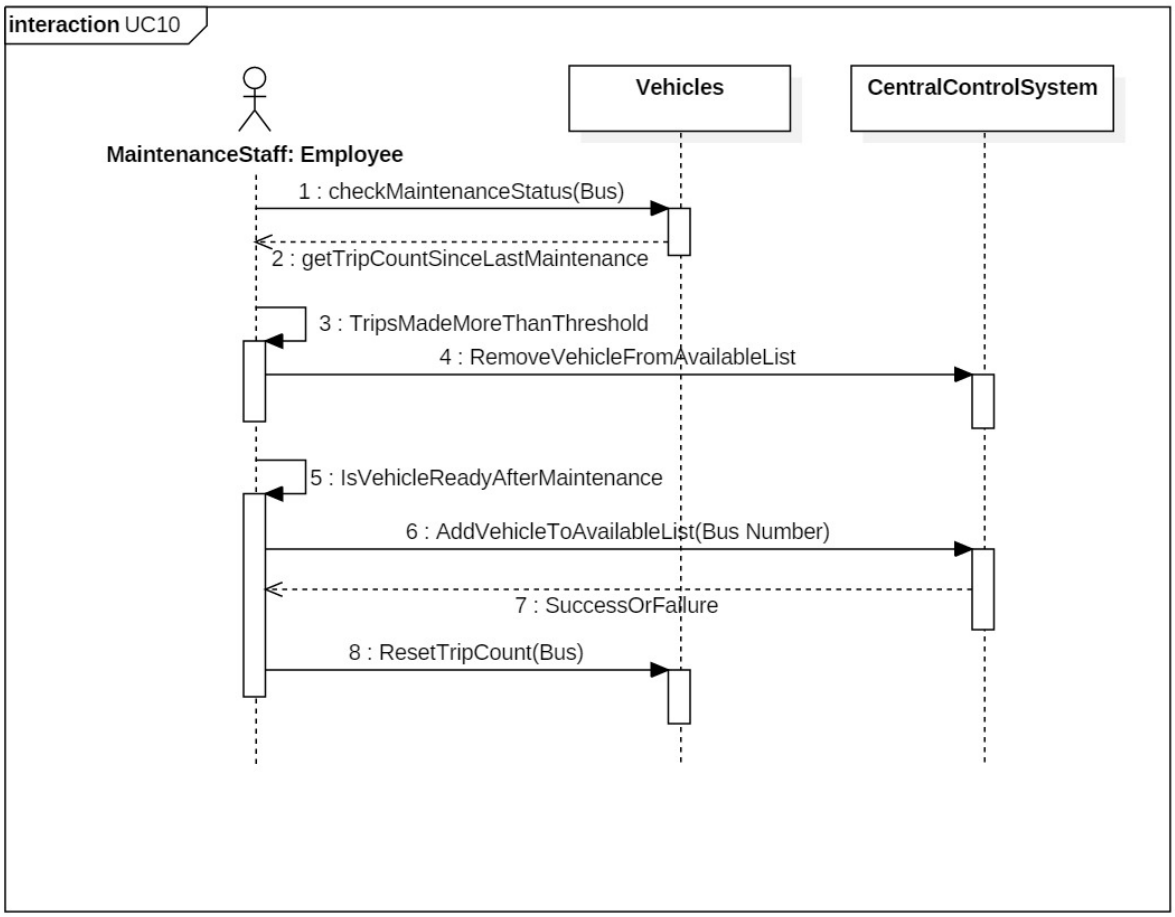
We have implemented the code best route retrieval and the fare by defining the routes as a collection of stops and searching for the source and destination stops among the routes and returned the best route available (i.e. least no of stops). The fare is calculated as $2 per stop. Schedule and realTimeLocation updates will be implemented in Phase4.

Use Case10(UC 10):

Summary:

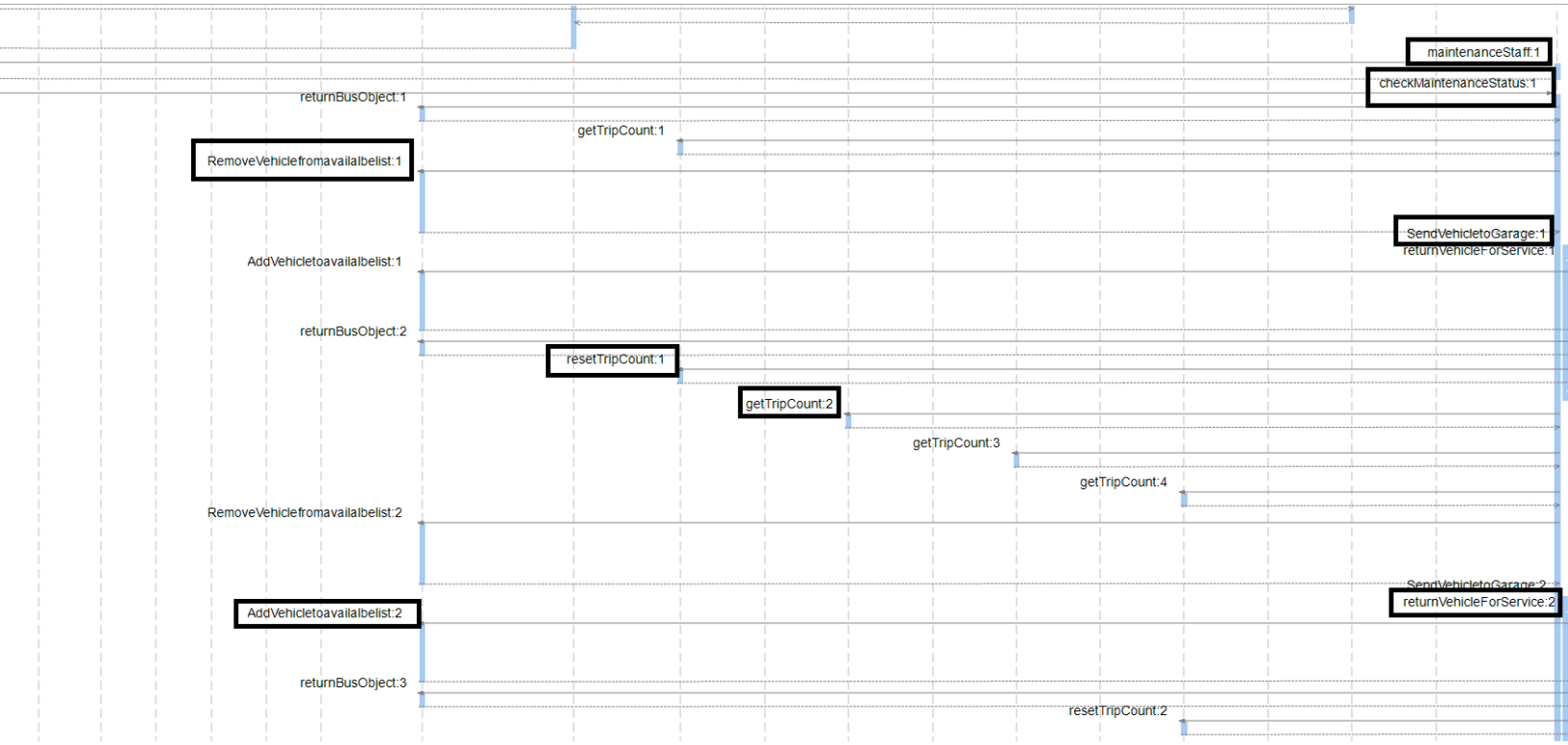
First Use Case we implemented(UC10) is related to periodic checking of TripCount of each vehicle since last it was Serviced in Garage. As show below, maintenance Staff (who implements Employee interface), periodically inquires TripCount for each Bus. Whenever, this TripCount increases beyond some threshold, Vehicle is called to Dockyard/Garage and is removed from Available Service List.

Once Servicing for Vehicle is Complete, it is sent to Field for service again. Since bus is serviced again, TripCount for this bus is reset. Also, Bus is added to available Service list.



Details:

1. **checkMaintenanceStatus :** periodically check for TripCount of buses since last service periodically(Say every night when all buses are in DockYard)
2. **getTripCount :** Retrieves TripCount since last maintenance for all Vehicles.
3. **RemoveVehicleFromavailableList:** Removes vehicle from available vehicle list when Bus/Van is called to Garage for Servicing.
4. **SendVehicleToGarage:** Vehicle is kept in Garage for duration till it is ready for Service again. NOTE: We have used timer value of 3 seconds (Thread.sleep(3000)) to simulate the duration which is needed to Service the Bus.
5. **AddVehicletoavailablelist:** Once Vehicle is ready for Service after servicing, add this vehicle to available service list.
6. **resetTripCount:** In addition to point 5 above, also reset “trip count since last maintenance” for that vehicle


maintenanceStaff:1
checkMaintenanceStatus:1
returnBusObject:1
getTripCount:1
RemoveVehiclefromavailalbelist:1
SendVehicletoGarage:1
returnVehicleForService:1
AddVehicletoavailalbelist:1
returnBusObject:2
resetTripCount:1
getTripCount:2
getTripCount:3
getTripCount:4
RemoveVehiclefromavailalbelist:2
SendVehicletoGarage:2
returnVehicleForService:2
AddVehicletoavailalbelist:2
returnBusObject:3
resetTripCount:2