Changes to the UML Diagram:
The main change to the UML diagram is that rather than having a concrete creator for every single type of vehicle and staff, there is only a single creator, so it is more of a simple factory implementation.

Observer Pattern

Subject
`<<interface>>`
register()
remove()
notify()

Observer
`<<abstract>>`

FNCDMoney
moneyAdded

JobOutcomes
totalSales

RaceResults
results

StaffMoney
totalMoneyChange

Tracker
Singleton Pattern
GetInstance()
display()

Logger
Singleton Pattern
GetInstance()
display()

Strategy Pattern

WashBehavior
csProbability
cdProbability
dsProbability
dcProbability

Chemical
brokenChance
wash()

ElbowGrease
likeNewChance
wash()

Detailed
wash()

Staff
name
daysworked
WashBehavior wb
earned
totalPay

Driver
injured
id

Salesperson
id
Action()

Mechanic
id

Intern
id

Driver
injured
id

Salesperson
id
Action()

Mechanic
id

Intern
id

Client

GetName
undo()
execute()

GetTime
undo()
execute()

GetSalesperson
undo()
execute()

GetInventory
undo()
execute()

GetInventoryDetails
undo()
execute()

BuyItem
undo()
execute()

EndInteraction
undo()
execute()

<<interface>>
Command
undo()
execute()

Command
Pattern

Invoker
SetCommand()

FNCD
performanceCars : ArrayList<Vehicle>
cars: ArrayList<Vehicle>
pickups: ArrayList<Vehicle>
salespeople: ArrayList<Staff>
mechanics: ArrayList<Staff>
interns: ArrayList<Staff>
soldVehicles: ArrayList<Vehicle>
departedStaff: ArrayList<Staff>
budget: int
dailySales: int

ModifyBudget()

Actions
opening()
racing()
washing()
repairing()
selling()
ending()

Buyer
buyType
vehicleType

Decorator Pattern

AddOns
- addChance
- cost

RoadRescueCoverage
- Vehicle v

SatelliteRadio
- Vehicle v

ExtendedWarranty
- Vehicle v

Undercoating
- Vehicle v

Vehicle
- saleBonus
- repairBonus
- washBonus
- cost
- saleprice
- condition
- state
- name
- type
- sold

Vehicle
- saleBonus
- repairBonus
- washBonus
- cost
- saleprice
- condition
- state
- name
- type
- sold

PerformanceCar
- id

Car
- id

Pickup
- id

ElectricCar
- range
- id

Motorcycle
- engineSize
- id

MonsterTruck
- stageName
- id

SemiTruck
- id

Limo
- id

Hybrid
- id

VehicleCreator
- CreateVehicle()

Factory
Pattern