Delivery System Project UML Class Diagram

Observer Pattern

Abstract Factory Pattern

Client

Potential Other Client Usage

<<interface>> PublisherBase

+ removeSubscriber(SubscriberBase): void
+ notifySubscriber(): void
+ registerSubscribers(SubscriberBase): void

<<interface>> SubscriberBase

+ updateSelf(DeliveryRequest): void

0..*

1

Shop

- name: String
- deliveryRequest: DeliveryRequest
- suscriberBases: List<SubscriberBase>

+ Shop(String): void
+ Shop(Shop): void
+ registerSubscriber(SubscriberBase): void
+ removeSubscriber(SubscriberBase): void
+ notifySubscribers(): void

Driver

- latestDeliveryRequest: DeliveryRequest
- name: String
- vehicleTyle: String
- availability: Boolean = true

+ Driver(): void
+ setName(String): void
+ getName(): String
+ setVehicleType(String): void
+ getVehicleType(): String
+ setAvailability(Boolean): True
+ getAvailability(): Boolean
+ getLatestDeliveryRequest(): DeliveryRequest
+ updateSelf(DeliveryRequest): void

DriverFactory {Abstract}

+ DriverFactory(): void
+ createDriver(String): Driver

Extends | Extends | Extends

TaxiDriverFactory

+ TaxiDriverFactory(): void
+ createDriver(String): Driver

ScooterDriverFactory

+ ScooterDriverFactory(): void
+ createDriver(String): Driver

VanDriverFactory

+ VanDriverFactory(): void
+ createDriver(String): Driver

DeliveryRequest

- shop: Shop
- customerAdress: String

+ DeliveryRequest(Shop, String): void
+ getShop: Shop
+ getCustomerAddress(): String

Extends

Extends | Extends

TaxiDriver

+ TaxiDriver(String): void
+ updateSelf(DeliveryRequest): void

ScooterDriver

+ ScooterDriver(String): void
+ updateSelf(DeliveryRequest): void

3

VanDriver

+ VanDriver(String): void
+ updateSelf(DeliveryRequest): void

Main

+ shop: Shop = new Shop ("Boston Seafood)
+ vanDriverFactory: DriverFactory
+ vanDriver: Driver
+ taxiDriverFactory: DriverFactory
+ taxiDriver: Driver
+ scooterDriverFactorty DriverFactory
+ scooterDriver1: Driver
+ scooterDriver2: Driver
+ scooterDriver3: Driver
+ deliveryRequest: DeliveryRequest

1

Main Entry

* demonstrating a scenario