

Warehouse Manager

-- warehouse: Warehouse

-- filename: String

- + save(): void
- + saveAs(filename: String): void
- + load(filename: String): void
- + importFile(textfile: String): void
- + showDate(): String
- + advanceDate(delta: int): void
- + viewAllProducts(): String
- + viewAllBatches(): String
- + viewAllBatchesByPartner(id: String): String
- + viewAllBatchesByProduct(id: String): String
- + viewPartner(id: String): String
- + viewAllPartners(): String
- + registerPartner(id: String, address: String): void
- + toggleNotification(partnerId: String, productId: String): void
- + showBuyTransactionsByPartner(id: String): String
- + showSellTransactionsByPartner(id: String): String
- + viewTransaction(id: int): String
- + registerBreakdown(partnerId: String, productId: String, productQuantity: int): void
- + registerSale(partnerId: String, paymentDeadline: String, productId: String, productQuantity: int): void
- + registerPurchase(partnerId: String, productId: String, productPrice: float, productQuantity: int): void
- + addRecipe(componentQuantity: int, alpha: float, productId: String, components: String): void
- + registerPayment(id: int): void
- + lookupBatchesByPrice(limit: float): String
- + lookupPaymentsByPartner(id: String): String

Warehouse

- products: Product[]
- partners: Partner[]
- transactions: Transaction[]
- transactionCounter: int
- balance: float
- date: Date
- notifySystem: NotificationSystem

- + importFile (filePath: String): void
- + showDate (): String
- + advanceDate (date: int): void
- + viewAllProducts (): String
- + viewAllBatches (): String
- + viewAllBatchesByPartner (id: String): String
- + viewAllBatchesByProduct (id: String): String
- + viewPartner (id: String): String
- + viewAllPartners (): String
- + registerPartner (id: String, address: String): void
- + toggleNotification (partnerId: String, productId: String): void
- + showBuyTransactionsByPartner (id: String): String
- + showSellTransactionsByPartner (id: String): String
- + viewTransaction (id: int): String
- + registerBreakdown (partnerId: String, productId: String, productQuantity: int): void
- + registerSale (partnerId: String, paymentDeadline: String, productId: String, productQuantity: int): void
- + registerPurchase (partnerId: String, productId: String, productPrice: float, productQuantity: int): void
- + addRecipe (componentQuantity: int, alpha: float, productId: String, components: String[]): void
- + registerPayment (id: int): void
- + lookupBatchesByPrice (limit: float): String
- + lookupPaymentsByPartner (id: String): String

Date
-- systimdate: int
+ advance Date (delta: int): void

Notification System
- notifications: Notification History []
+ send Notifications (product: Product)

Notification History
- blacklisted Products: Product []
- notifications: Notification []
+ toggle Product Notifications (product: Product): void
+ clear Notifications (): void

Notification
-- description: String
-- delivery: Delivery Method

Delivery Method
-- description: String

Breakdown Transaction
-- base price: float

Buy Transaction
-- price per unit: float

Sell Transaction
-- payment deadline: Date
-- base price: float
-- payed: boolean

Partner
-- name: String
-- address: String
-- status: Status
-- points: int
-- notification History: Notification History
+ clear Notifications (): void
+ toggle Notifications (productId: String): void
+ promote Status: void
+ demote Status: void

Status
-- status: String
+ promote Status: void
+ demote Status: void

<<abstract>> Transaction
-- id: int
-- payment date: Date
-- partner: Partner
-- product quantity: int
-- product: Product
-- price: float

<<abstract>> Product

- name: String
- batches: Batch[]

- + viewAllBatches(): Batch[]
- + getMaxPrice(): float
- + get current Stock(): int
- + get Batch by Price Limit (price: float): Batch[]
- + view All Partner Batches(): Batch[]

RecipeComponent

- productType: Simple Product
- productQuantity: int

Simple Product

Batch

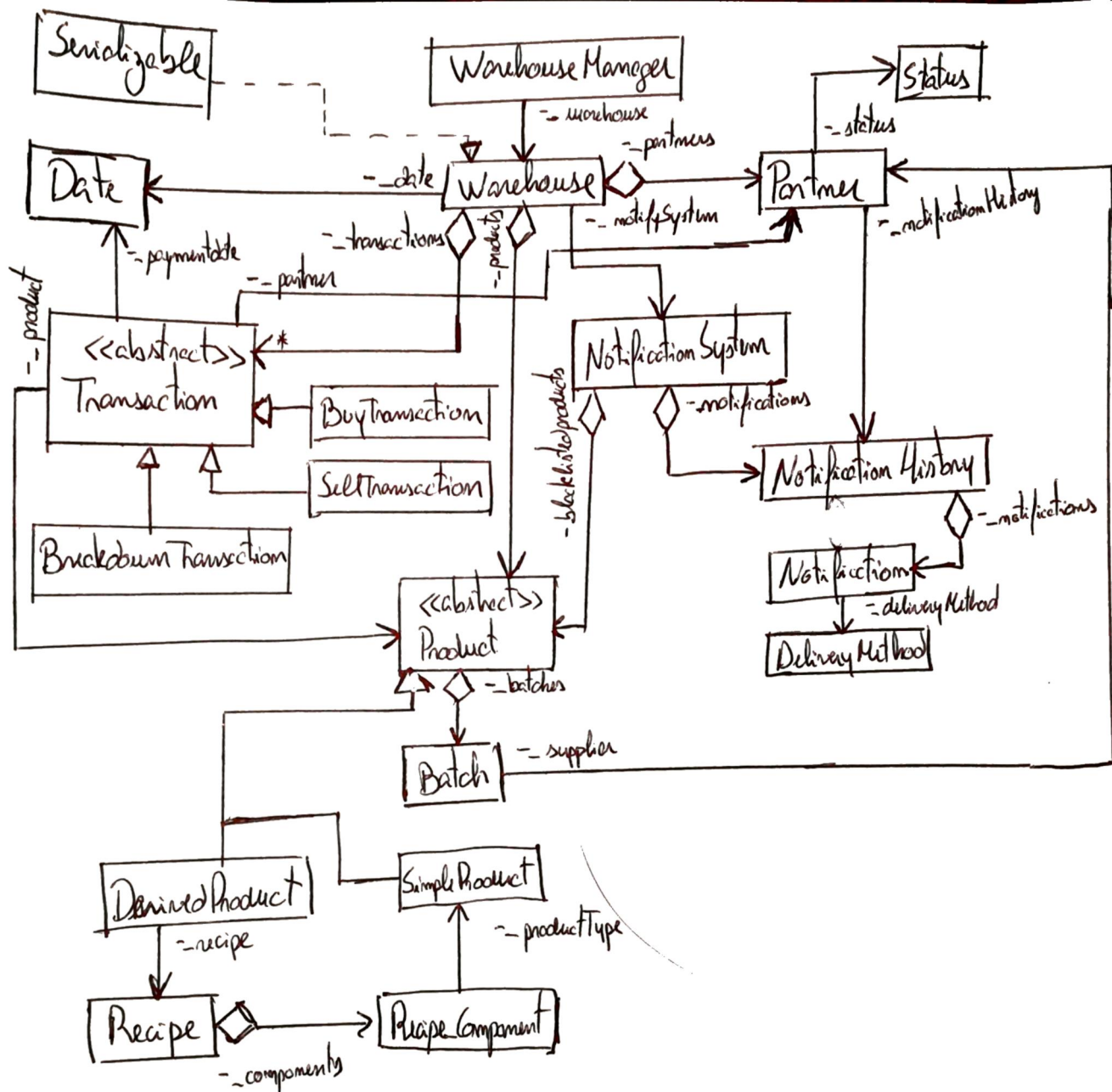
- supplier: Partner
- quantity: int
- price Per Unit: float
- get Partner Name(): String

Recipe

- alpha: float
- components: Recipe components
- + calculate Recipe(): String

Derived Product

- recipe: Recipe



Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo de projeto (grupo 107).

Tig. Rodrigo Viana de Siqueira

Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que ~~o~~ constituem o grupo de projeto. (grupo 107)

Rafael Grãu