

Product
<ul style="list-style-type: none"> - key: string - supplier: string - price: int - critical value: int - stock: int = 0
<ul style="list-style-type: none"> + add client (c: client): void + remove client (c: client): void + notify clients (type: string, p: Product): void + add stock (amount: int): void + remove stock (amount: int): void

Notification
<ul style="list-style-type: none"> - type: string

Client
<ul style="list-style-type: none"> - key: string - name: string - address: string - points: int = 0 - purchases Made Value: int = 0 - purchases Said Value: int = 0
<ul style="list-style-type: none"> + add Notification (m: Notification): void + clear Notifications (): void + add sale (s: Sale): void

Book
<ul style="list-style-type: none"> - title: string - author: string - isbn: string

Container
<ul style="list-style-type: none"> - type: string - level: string

Box
<ul style="list-style-type: none"> - type: string

Supplier
<ul style="list-style-type: none"> - key: string - name: string - address: string - active: boolean
<ul style="list-style-type: none"> + add Order (o: Order): void + toggle Transactions (): boolean

Date
<ul style="list-style-type: none"> - date: int = 0
<ul style="list-style-type: none"> + advance Date (days: int): void

<<abstract>> Transaction
<ul style="list-style-type: none"> - next Key: int = 0 - key: int - base Value: int

Sale
<ul style="list-style-type: none"> - paid Value: int
<ul style="list-style-type: none"> + price level (d: Date): string + pay (s: Status): int

<<interface>> Status
<ul style="list-style-type: none"> + price (price level: string, original Price: int, current Date: Date, deadline: Date): int

Order

Normal

Selection

White

store

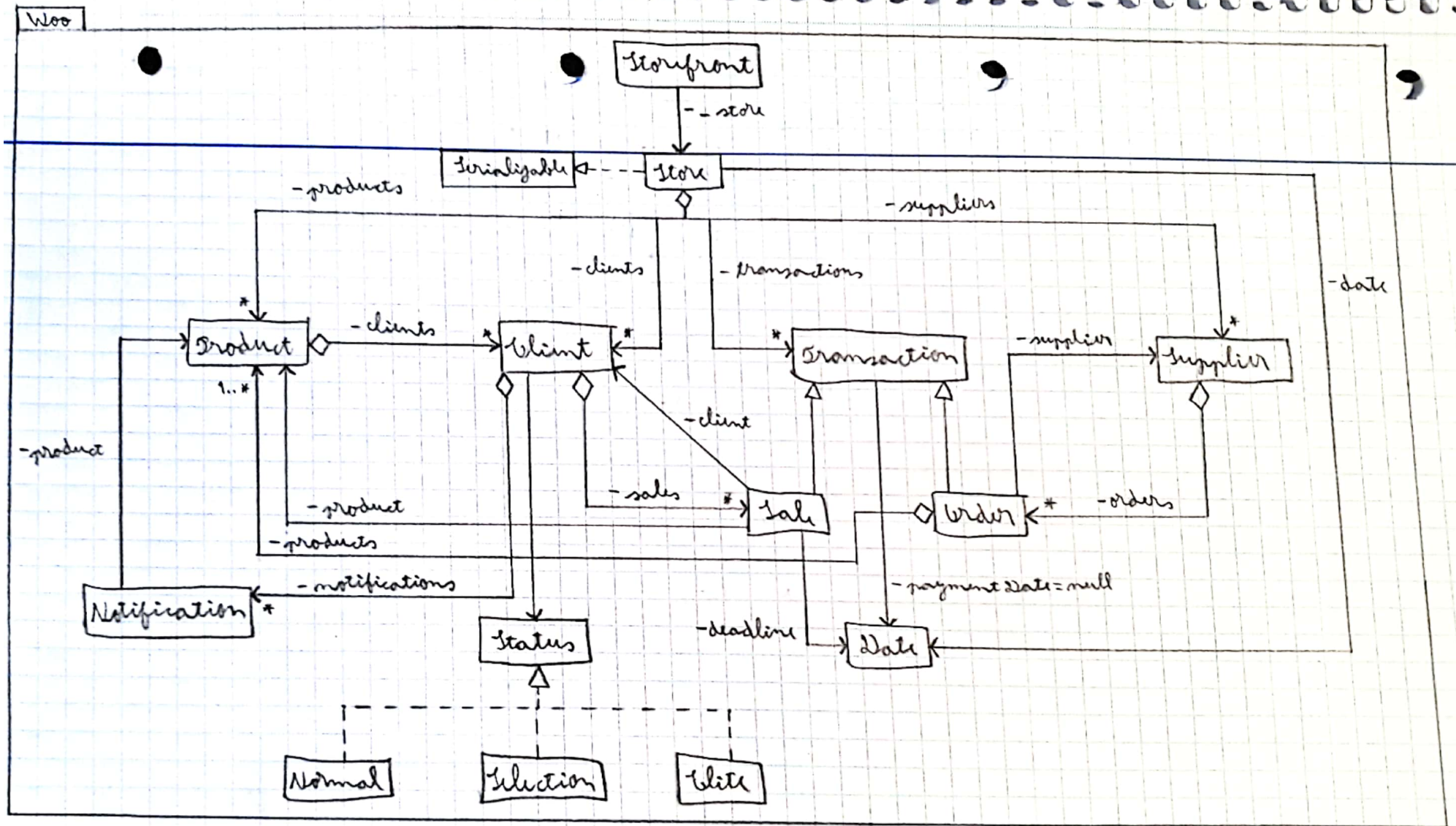
- available Balance: int = 0
- accounting Balance: int = 0

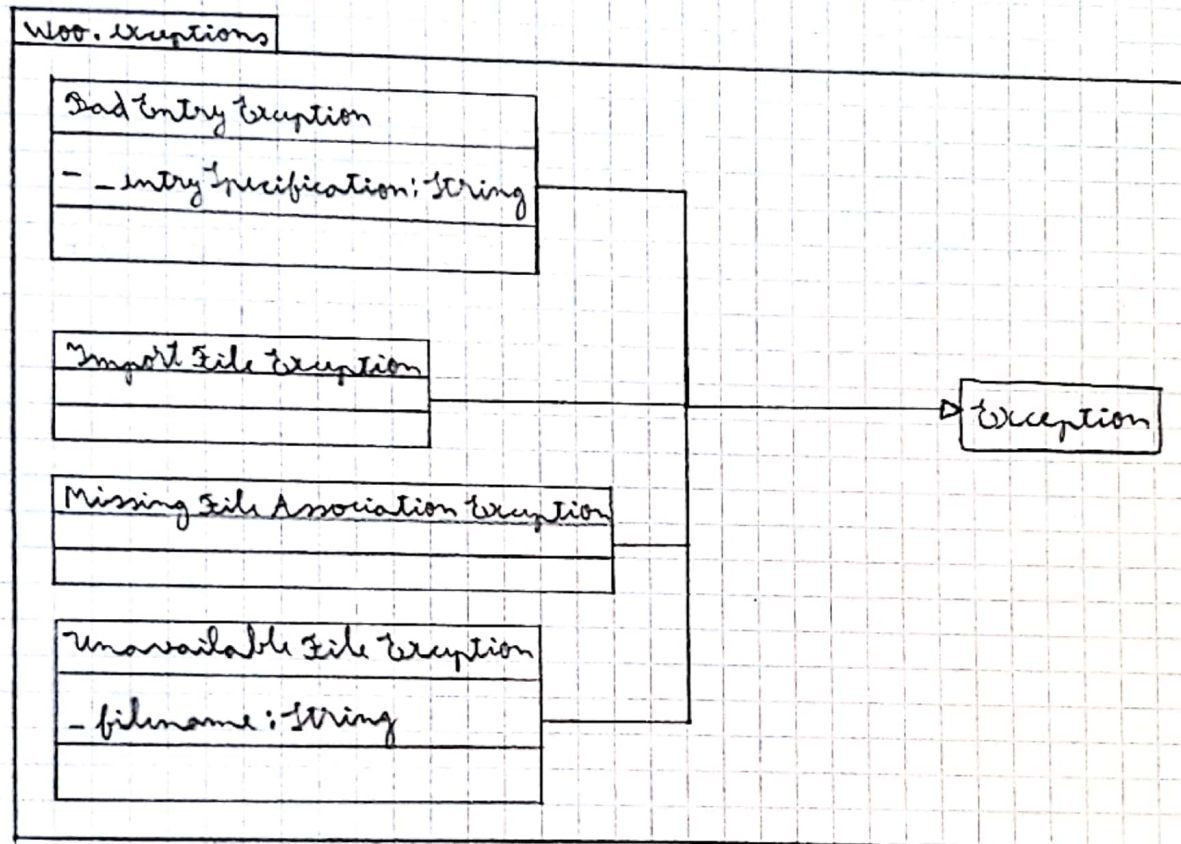
+ get Current Date (): int
+ advance Date (days: int): void
+ add Client (key: String, c: Client): void
+ get Client (key: String): Client
+ get Client Transactions (key: String): Collection<Sale>
+ toggle Product Notifications (client: String, product: String): boolean
+ activate All Product Notifications (c: Client): void
+ add Product (key: String, p: Product): void
+ get Product (key: String): Product
+ change Price (product: String, price: int): void
+ add Supplier (key: String, s: Supplier): void
+ get Supplier (key: String): Supplier
+ get Supplier Transactions (key: String): Collection<Order>
+ toggle Transactions (supplier: String): boolean
+ add Transaction (key: int, t: Transaction): void
+ get Transaction (key: int): Transaction
+ pay (key: int): void
+ lookup Payments by Client (key: String): Collection<Sale>
+ lookup Products under Top Price (limit: int): Collection<Product>
import File (textfile: String): void

storefront

- filename: String = ""

+ advance Date (days: int): void
+ get Current Date (): int
+ get Available Balance (): int
+ get Accounting Balance (): int
+ add Client (key: String, name: String, address: String): void
+ get All Clients (): Collection<Client>
+ get Client (key: String): Client
+ get Client Transactions (key: String): Collection<Sale>
+ toggle Product Notifications (client: String, product: String): boolean
+ add Book (key: String, title: String, author: String, isbn: String, price: int, critical Value: int, supplier: String): void
+ add Bot (key: String, price: int, critical Value: int, supplier: String, type: String): void
+ add Container (key: String, price: int, critical Value: int, supplier: String, type: String, level: String): void
+ get All Products (): Collection<Product>
+ change Price (product: String, price: int): void
+ add Supplier (key: String, name: String, address: String): void
+ get All Suppliers (): Collection<Supplier>
+ get Supplier Transactions (key: String): Collection<Order>
+ toggle Transactions (supplier: String): boolean
+ add Order (supplier: String, products: Hash Map<String, int>): void
+ add Sale (client: String, deadline: int, product: String, amount: int): void
+ get Transaction (key: int): Transaction
+ pay (key: int): void
+ save (): void
+ save As (filename: String): void
+ load (filename: String): void
+ import File (textfile: String): void





Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo de projecto.
Jorge Luis Guerra