

Software Design and Testing

Project Name : CryptoCortex

Team Members

202412029 : Preksha Joshi

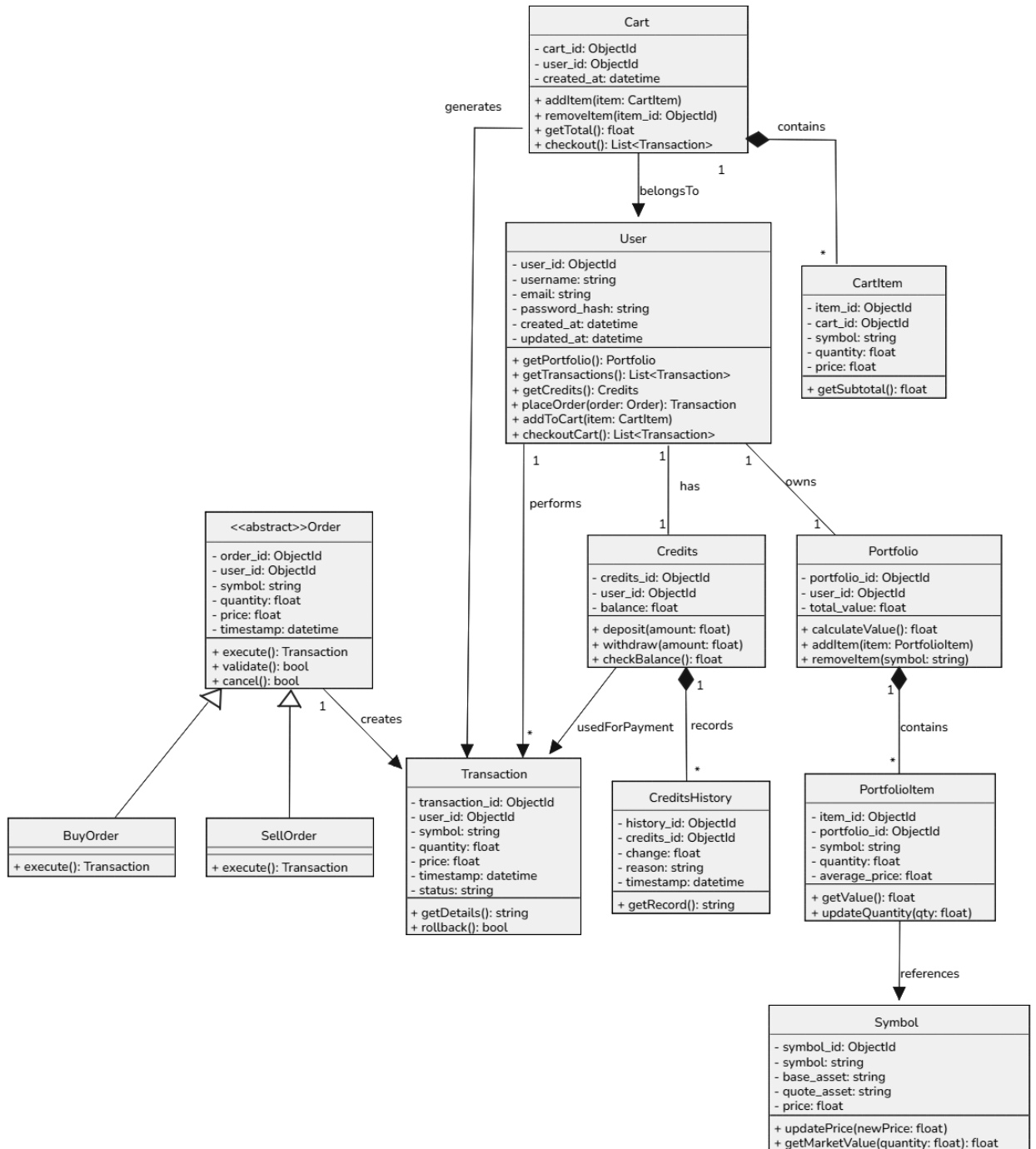
202412026 : Tisha Jain

202412032 : Kaival Shah

202412066 : Dhruv Patel

Class Diagram

CryptoCortex Class Diagram



CryptoCortex Class Diagram Documentation

1. Classes and Attributes

User

Attributes:

- user_id: ObjectId
- username: string
- email: string
- password_hash: string
- created_at: datetime
- updated_at: datetime

Methods:

- getPortfolio(): Portfolio
- getTransactions(): List<Transaction>
- getCredits(): Credits
- placeOrder(order: Order): Transaction
- addToCart(item: CartItem)
- checkoutCart(): List<Transaction>

Cart

Attributes:

- cart_id: ObjectId
- user_id: ObjectId
- created_at: datetime

Methods:

- addItem(item: CartItem)
- removeItem(item_id: ObjectId)
- getTotal(): float

- checkout(): List<Transaction>

CartItem

Attributes:

- item_id: ObjectId
- cart_id: ObjectId
- symbol: string
- quantity: float
- price: float

Methods:

- getSubtotal(): float

Order (Abstract Class)

Attributes:

- order_id: ObjectId
- user_id: ObjectId
- symbol: string
- quantity: float
- price: float
- timestamp: datetime

Methods:

- execute(): Transaction
- validate(): bool
- cancel(): bool

BuyOrder (inherits Order)

Methods:

- execute(): Transaction
- SellOrder (inherits Order)

Methods:

- execute(): Transaction

Transaction**Attributes:**

- transaction_id: ObjectId
- user_id: ObjectId
- symbol: string
- quantity: float
- price: float
- timestamp: datetime
- status: string

Methods:

- getDetails(): string
- rollback(): bool

Credits**Attributes:**

- credits_id: ObjectId
- user_id: ObjectId
- balance: float

Methods:

- deposit(amount: float)
- withdraw(amount: float)
- checkBalance(): float

CreditsHistory**Attributes:**

- history_id: ObjectId
- credits_id: ObjectId
- change: float

- reason: string
- timestamp: datetime

Methods:

- getRecord(): string

Portfolio

Attributes:

- portfolio_id: ObjectId
- user_id: ObjectId
- total_value: float

Methods:

- calculateValue(): float
- addItem(item: PortfolioItem)
- removeItem(symbol: string)

PortfolioItem

Attributes:

- item_id: ObjectId
- portfolio_id: ObjectId
- symbol: string
- quantity: float
- average_price: float

Methods:

- getValue(): float
- updateQuantity(qty: float)

2. Relationships & Cardinalities

- **User ↔ Cart**

- A User *owns* **1** Cart.
- A Cart *belongs to* **1** User.

- **Cart ↔ CartItem**

- A Cart *contains* **many** CartItems.
- A CartItem *belongs to* **1** Cart.

- **User ↔ Order**

- A User *places* **many** Orders.

- **Order ↔ Transaction**

- Each Order *creates* **1** Transaction.

- **User ↔ Transaction**

- A User *performs* **many** Transactions.

- **Credits ↔ User**

- Each User *has* **1** Credits.

- **Credits ↔ CreditsHistory**

- A Credits account *records* **many** CreditsHistory entries.

- **Credits ↔ Transaction**

- Credits are *used for payment* in **many** Transactions.

- **User ↔ Portfolio**

- Each User *owns* **1** Portfolio.

- **Portfolio ↔ PortfolioItem**
 - A Portfolio *contains many* PortfolioItems.
- **PortfolioItem ↔ Symbol**
 - A PortfolioItem *references 1* Symbol.

3. Key Functionalities

- **Cart Management:** Add/remove items, calculate totals, checkout.
- **Order Execution:** Place buy/sell orders, validate and cancel.
- **Transactions:** Execute and rollback transactions, record details.
- **Credits System:** Deposit, withdraw, check balance, maintain history.
- **Portfolio Tracking:** Calculate value, add/remove items, update quantities.
- **Market Integration:** Reference symbols, update and fetch market values.
- **User Operations:** Manage profile, portfolio, credits, cart, and transactions.