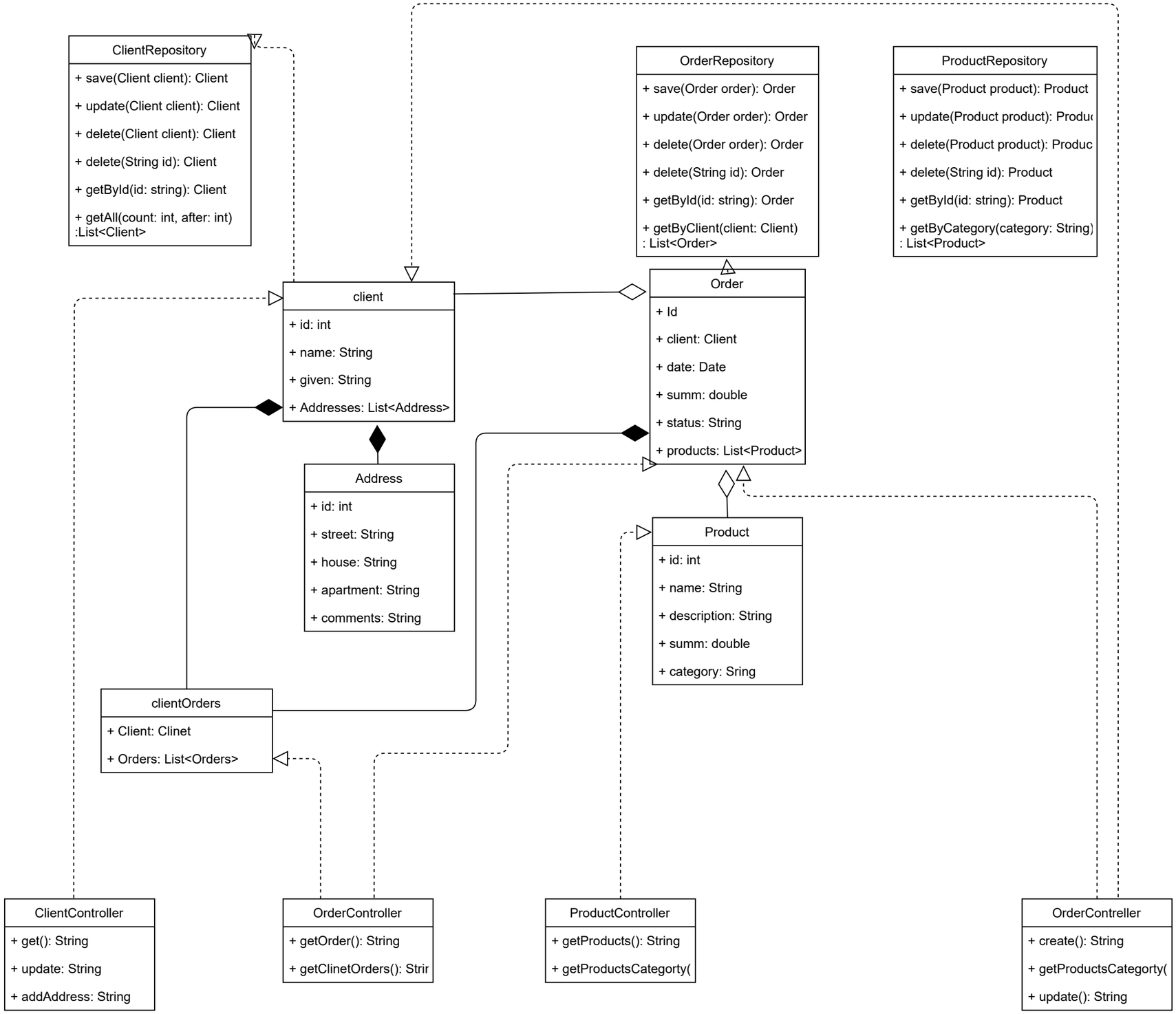
ClientRepository
+ save(Client client): Client
+ update(Client client): Client
+ delete(Client client): Client
+ delete(String id): Client
+ getById(id: string): Client
+ getAll(count: int, after: int) :List<Client>
OrderRepository
+ save(Order order): Order
+ update(Order order): Order
+ delete(Order order): Order
+ delete(String id): Order
+ getById(id: string): Order
+ getByClient(client: Client) : List<Order>
ProductRepository
+ save(Product product): Product
+ update(Product product): Produc
+ delete(Product product): Produc
+ delete(String id): Product
+ getById(id: string): Product
+ getByCategory(category: String) : List<Product>
client
+ id: int
+ name: String
+ given: String
+ Addresses: List<Address>
Order
+ Id
+ client: Client
+ date: Date
+ summ: double
+ status: String
+ products: List<Product>
Address
+ id: int
+ street: String
+ house: String
+ apartment: String
+ comments: String
Product
+ id: int
+ name: String
+ description: String
+ summ: double
+ category: Sring
clientOrders
+ Client: Clinet
+ Orders: List<Orders>
ClientController
+ get(): String
+ update: String
+ addAddress: String
OrderController
+ getOrder(): String
+ getClinetOrders(): Strir
ProductController
+ getProducts(): String
+ getProductsCategorty(
OrderContreller
+ create(): String
+ getProductsCategorty(
+ update(): String