



Model

Usuario - email: String - senha: String - nome: String - foto: String - cartoes: List<CartaoDeCredito> - endereços: List<Endereço> - pedidos: List<Pedido> - itensAnunciados: List<Item> | Categoria | |- id: Int | |- nome: String | |- imagem: String | |- id: Int | |-

- titulo: String

- data: String

- preco: Double

descricao: Stringimagem: String

vendedor: Usuario

- categoria: Categoria

- avaliacoes: List<Avaliacao>

- numero: String - validade: String - cvc: String - tipoCartao: TipoCartao

CartaoDeCredito

- nomeTitular: String

+ connectToChatWith(userId: Int): void

+ createOrder(order: Order): void

+ getOrder(id: Int): Order

+ getChat(): Chat

+ sendMessageTo(message: Message, userId: Int): void

OrderController

«enummeration» TipoCartão DEBITO CREDITO

Endereco - id: Int - rua: String - cep: String - numero: Int - estado: String - cidade: String

Pedido - id: Int - quantidade: Int - data: String - item: Item - enderecoDeEntrega: Endereco - statusEntrega: StatusEntrega - metodoPagamento: MetodoPagamento

«enummeration»
MetodoPagamento
CARTAO
BOLETO
PIX

- id: Int - avaliacao: Int - comentario: String

+ getCategory(id: Int): Categoria

+ deleteCategory(id: Int): void

+ getReview(id: Int): Avaliacao

+ deleteReview(id: Int): void

+ listCategories(): List<Categoria>

+ listCategoryItens(id: Int): List<item> + createReview(review: Avaliacao): void

+ listReviews(itemId: Int): List<Avaliacao> + updateReview(review: Avaliacao): Avaliacao

+ updateCategory(category: Categoria): Categoria

«enummeration»
StatusEntrega

PREPARANDO_PARA_ENVIO
ENVIADO
ENTREGUE

Chat
- id: Int
- mensagens: List<Mensagem>

Mensagem

- idUsuario: Int
- horario: String
- mensagem: String

O projeto usará a biblioteca Lombok, que gera automaticamente os getters e os setters das classes com uso de anotações.