

| Integrante                            | Módulo                        | Principais classes e responsabilidades   |
|---------------------------------------|-------------------------------|--|
| <b>1</b> — Cliente / Autenticação     | Módulo de Usuário             | Criar classes: Cliente, Administrador, Usuario, Autenticacao. • Implementar login, cadastro, e autenticação de sessão. • Métodos: login(), logout(), registrar(), autenticar(). • Controlar permissões de acesso (cliente vs admin). |
| <b>2</b> — Catálogo / Carrinho        | Módulo de Produtos e Carrinho | Criar classes: Produto, Categoria, Carrinho. • Métodos: listar_produtos(), adicionar_ao_carrinho(), remover_do_carrinho(), calcular_total(). • Lidar com persistência de dados dos produtos.   |
| <b>3</b> — Pedido e Checkout          | Módulo de Pedidos             | Criar classes: Pedido, ItemPedido, Checkout. • Métodos: finalizar_compra(), acompanhar_pedido(), gerar_nota_fiscal(). • Fazer a ponte entre carrinho, pagamento e envio.   |
| <b>4</b> — Pagamento                  | Módulo de Pagamento           | Criar classes: Pagamento, Transacao, NotificacaoPagamento. • Métodos: processar_pagamento(), verificar_status(), notificar_falha(). • Simular integração com API de pagamento (ex: Stripe, PagSeguro ou PayPal).                     |
| <b>5</b> — Entrega / Logística        | Módulo de Envio               | Criar classes: Transportadora, Envio, Rastreio. • Métodos: enviar_produto(), atualizar_status_entrega(). • Pode simular integração com API de transporte (ex: Correios).   |
| <b>6</b> — Administração e Relatórios | Módulo Administrativo         | Criar classes: Relatorio, GerenciamentoUsuarios, GerenciamentoPedidos, GerenciamentoProdutos. • Métodos: gerar_relatorios(), visualizar_vendas(), gerenciar_cadastros(). • Responsável por funções exclusivas do Administrador.      |