

Cenário: Desenvolvendo o Sistema FlexEntregas

A startup FlexEntregas quer criar uma plataforma para planejar e simular entregas em diferentes regiões do país. O sistema precisa:

1. Lidar com múltiplos meios de entrega: moto, bicicleta, drone, cavalo (em áreas rurais).
2. Calcular o tempo estimado, custo de combustível (se aplicável), e restrições (ex: drones não podem operar em dias de chuva).
3. Permitir simulações combinadas, comparando diferentes estratégias de entrega em diferentes regiões.
4. Estar aberto a extensões futuras, como novos meios de transporte e regras logísticas.

A equipe de produto deseja modularidade, extensibilidade, testabilidade e baixo acoplamento.

Tarefa das Equipes

Cada equipe deve propor uma arquitetura de código para resolver o problema, considerando:

- Representação dos meios de entrega com lógica própria para cálculo de tempo e restrições.
- Simulação flexível, que permita passar diferentes estratégias de transporte, de forma que seja possível trocar ou adicionar novas sem reescrever o código principal (**polimorfismo**).
- Requisitos de baixa duplicação, fácil manutenção e possibilidade de testes automatizados.

A solução deve demonstrar claramente o uso de **polimorfismo** — seja via herança, interfaces, funções de ordem superior — e justificar sua escolha.

Análise Crítica

Com base no seu projeto, discutam:

1. Como o polimorfismo foi aplicado na sua solução?
2. Foi por herança? Interface? Função como parâmetro? Outro?
3. A linguagem usada facilitou ou dificultou o uso de estratégias polimórficas?
4. Como sua abordagem permite adicionar novos meios de entrega no futuro?

Entrega

- Código (pode ser um esboço funcional simplificado).

- Uma explicação de como aplicaram o polimorfismo.
- Data de entrega 23/06/2025 até as 9h.
- Data de apresentação: 23/06/2025