

## **Evoluindo o Strategy para Template Method**

Garantir que o fluxo do algoritmo possa ser separado nem sempre é possível. A vezes as regras que você precisa aplicar não vão permitir que o algoritmo seja quebrado em fluxos diferentes. Uma ideia para começar e validar essa possibilidade é tentar duplicar o código entre as estratégias.

Se o algoritmo separado precisar de muita duplicação, talvez a separação em estratégias não seja a melhor opção. Os ganhos obtidos com a flexibilidade da solução não vão compensar os gastos com a manutenção de código repetido. Nesses casos, outros padrões podem ajudar a resolver o problema, como o Template Method.

Ao aplicar o Template Method definimos uma estrutura base, que será comum a todas as variações de fluxos, além de vários pontos gancho onde podemos variar a implementação. Dessa forma os pontos comuns ficam centralizados na classe mãe e as classes filhas podem implementar sua própria lógica se beneficiando do algoritmo base.

Caso fosse necessário aplicar uma mesma regra em todas as diferentes estratégias (por exemplo, incluir um bônus de 10% para primeiras compras), essa regra precisaria ser duplicada. Poderíamos definir um template que calcula a pontuação baseado na distância, regra que seria definida pelas classes filhas, e sempre adiciona essa bonificação extra ao final.