

# Violação ao SOLID - Responsabilidade única

Aluno: Rodrigo dos Santos Carvalho

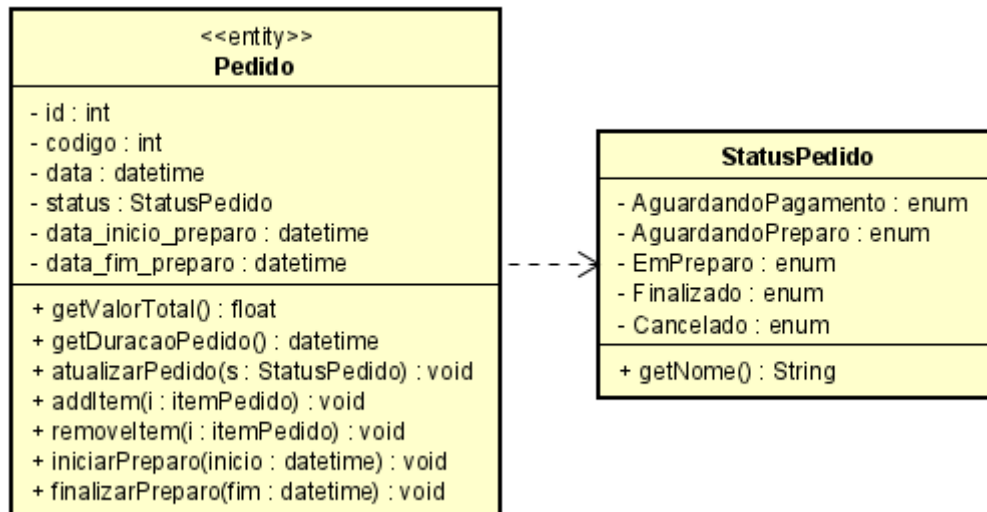


Diagrama de classes com a violação

A entidade Pedido estava violando o princípio de responsabilidade única. Estávamos realizando operações relacionadas ao preparo de um pedido dentro dessa classe.

Após o cliente realizar o pagamento do seu pedido, os cozinheiros recebem em seu painel de controle esse pedido, podendo ver os detalhes do pedido, e em sequência iniciar o preparo. Ao iniciar o preparo seria chamado o método “iniciarPreparo” passando a data e hora corrente, e mudaria o status do pedido para “EmPreparo”. E após o pedido estar pronto, o cozinheiro seleciona a opção de finalizar o preparo, que chamaria o método “finalizarPreparo” informando como parâmetro a data e hora corrente e atualizando o status do pedido para “Finalizado”.

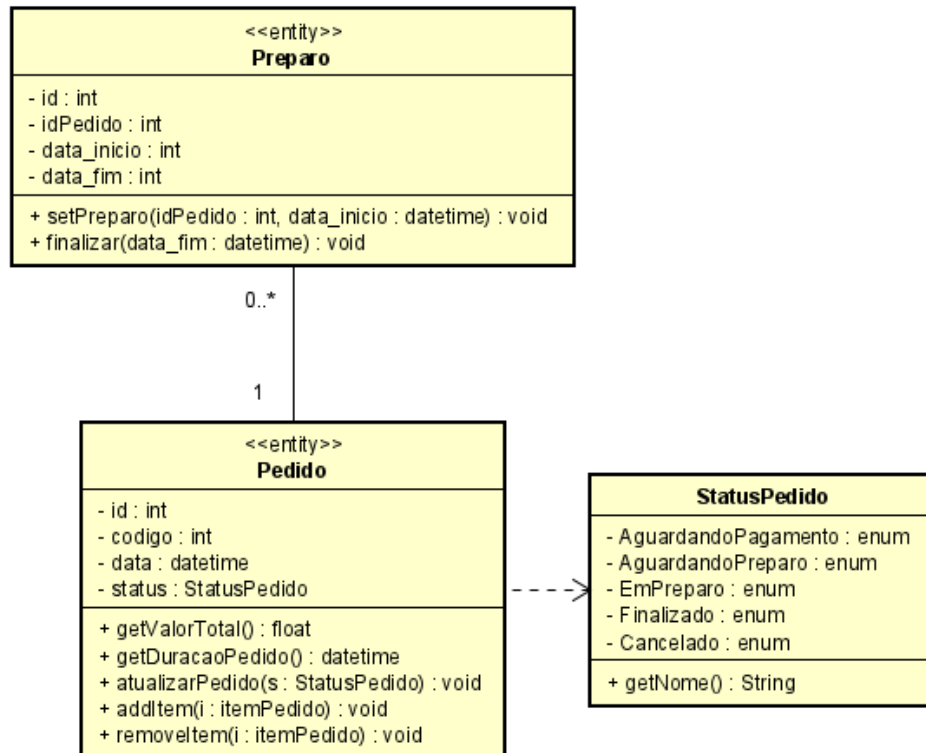


Diagrama de classes com a solução

Para resolver a violação resolvemos criar uma entidade “Preparo” para segregar as responsabilidades. O fluxo ocorre num processo muito parecido. Com o cozinheiro selecionando a opção de iniciar preparo, informando como parâmetro a data e hora corrente e dessa vez o id do pedido a ser preparado. E ao finalizar o preparo do pedido, ele deverá selecionar a opção de finalizar o preparo em seu painel, dessa forma será chamado o método “Finalizar” com a data e hora corrente sendo enviada como parâmetro.

A mudança do status do pedido será feita por um padrão de projeto observer, que estará atualizando o status de acordo com o início e o fim do preparo.