



Programação II
2021/22
1º Trabalho Prático

Número de alunos por grupo: 2

Data limite de entrega: 29-4-2022 até às 23h55

(1/5 para Trab. Est. e Dir. Asso.)

ComEST

Com a pandemia, o negócio de entrega de comida ao domicílio cresceu bastante. O grupo de investidores, AlbinvEST, responsável pelo sucesso de várias empresas, resolveu investir na criação de uma empresa deste tipo: a ComEST. Para o desenvolvimento do sistema informático da empresa, contactaram a sua parceira de sempre, a EST. Nesta primeira fase, o sistema irá simular a escolha dos pratos pelos clientes e o cálculo do preço final, incluindo a taxa de entrega. Nesta fase não se irá ter em conta a distância entre restaurante e o cliente. No futuro, mas não neste trabalho, pretende-se implementar o sistema no seu todo, pelo que as decisões tomadas para este trabalho devem ser escaláveis e facilmente alteráveis.

Funcionamento do sistema

A aplicação deve permitir a um cliente escolher qual o(s) prato(s) que deseja dentro da oferta de um dos vários restaurantes suportados pelo sistema. O primeiro passo é escolher o restaurante. Depois serão apresentados vários pratos e o cliente terá de escolher qual o prato, ou pratos, que deseja. Cada prato pode ter opções que aumentam (ou diminuem) o preço do prato, as quais o cliente pode selecionar.

Depois de confirmar cada prato, o cliente pode confirmar o pedido. O pedido é então aceite e é gerado um código aleatório de 6 caracteres, que identifica univocamente o pedido dentro do sistema. Numa próxima fase de desenvolvimento, este código permitirá ao cliente ver o estado do pedido (se está em preparação, a caminho, tempo estimado para entrega, etc). Nesta fase apenas permite ver um resumo do pedido.

Os operadores da empresa podem ver todos os pedidos ou ver só os de um restaurante.

Interface do sistema

O sistema tem duas interfaces, ambas simuladas numa consola. As interfaces são: uma que simula o sistema central e outra que simula a aplicação do cliente. É importante enfatizar este ponto: estas interfaces SIMULAM o sistema, não são as interfaces finais.

A interface que representa o sistema permite a visualização de todos os pedidos e também a visualização dos pedidos de um dado restaurante. O operador pode ver um pedido em particular, introduzindo o seu código.

A interface do cliente permite realizar um novo pedido ou visualizar o estado de um pedido. Neste trabalho não se guardam informações sobre o cliente que fez o pedido, pelo que não há necessidade de o identificar.

Aspetos técnicos a ter em conta

Geração de códigos

Cada código gerado pelo sistema terá letras maiúsculas, minúsculas e algarismos. Cada um deles é gerado aleatoriamente e terá 6 caracteres (exemplo: H6ttY3). Para a geração de códigos devem usar a classe `GeradorCodigos`.

Unidades a usar

Os preços serão sempre em euros e devem ser armazenados em float. Os pesos devem ser em gramas e guardados em inteiros.

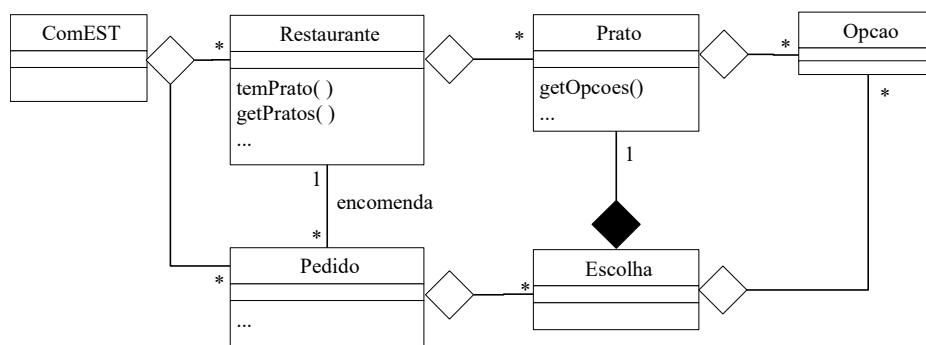
Taxas de entrega a usar

A taxa de entrega a usar depende do peso da comida transportada. Cada prato (e opção) terá um peso indicativo. O peso a calcular será a soma do preço de todos os pratos e respectivas opções. As taxas a usar estão indicadas na tabela seguinte:

Peso total (g)	Tarifa a usar (€)
[0, 1500]	2.5
]1500, 3000]	4.5
]3000, 4000]	5.0
> 4000	6.0 base + 1.0 por kilo acima dos 4kg

Diagrama de classes

O diagrama de classes, incompleto, para este trabalho é o seguinte:



Código fornecido

Para este trabalho já são disponibilizadas várias classes que, contudo, estão incompletas. Há muitas variáveis a serem declaradas e métodos a serem implementados. Todos os métodos apresentados no código DEVEM ser implementados, mesmo que durante a implementação do sistema não venham a ser necessários. **Além dos métodos mencionados poderão ter de desenvolver novos métodos. Além dos campos indicados podem ser necessários novos campos.** Quer os métodos quer os campos devem ser colocados na classe certa.

O grupo responsável pelo desenvolvimento das interfaces já cumpriu todas as suas tarefas. O grupo do sistema (você) é que ainda não cumpriu com a sua parte e terá de o fazer até à data de entrega. Nas classes das interfaces estão assinalados (com `// TODO`) os locais onde terão de acrescentar o vosso código, de modo a pôr todo o sistema a funcionar. Podem acrescentar código em mais locais que não os assinalados, se a vossa solução assim o exigir.

Classes a desenvolver

Estas são as classes que se devem desenvolver. Algumas classes têm, no código fornecido, métodos declarados que terão de ser implementados. Podem ter de implementar mais métodos além dos que estão indicados no código.

Prato

Esta classe representa um prato. Deve ter o nome, uma descrição, o preço e o peso (preço e peso devem ser positivos), além de uma lista com as opções associadas.

Opcao

Esta classe representa uma opção de um prato. Deve ter o nome, o preço e o peso (ambos podem ser negativos). Deve ter construtores, getters e setters relevantes.

Restaurante

Esta classe representa um restaurante. Deve ter o nome, uma descrição, a lista de pratos que oferece e uma lista com os pedidos já realizados.

Escolha

Esta classe representa um prato escolhido pelo utilizador. Deve ter a indicação de qual o prato escolhido bem como a lista das opções que o utilizador selecionou.

Pedido

Classe que representa um pedido. Deve ter a indicação de qual o restaurante onde foi feito o pedido e qual a lista de escolhas feita pelo cliente.

ComEST

Classe que representa o sistema. Deve armazenar todos os restaurantes e todos os pedidos feitos pelos clientes.

AppMain

Esta classe tem o método main e é ela que inicia os vários componentes do sistema, bem como define a configuração de teste.

Configurações de teste

No código da classe AppMain deve ser colocado o código de teste. **O software deve estar preparado para qualquer configuração, não deve ficar limitado à configuração de teste.** Para facilitar os testes, devem-se criar os elementos referidos em comentário no código.

Objetivos

O objetivo deste trabalho é implementar o sistema tal como está descrito, dando-se especial atenção à correta implementação da arquitetura do sistema.

Sugestão dos passos a seguir	Critérios de avaliação	Conceitos a utilizar
Estudar as teóricas e as fichas práticas Identificar as classes presentes, as relações entre elas e a sua interface; Implementar a classe Opcao Implementar a classe Prato Implementar a classe Restaurante Implementar a classe Escolha Implementar a classe Pedido Implementar a classe ComEST Ajustar a interação com o utilizador	Estrutura de packages Estrutura de classes Interface das classes Implementação das classes Interligação entre os objectos Funcionamento do sistema Relatório final Defesa do trabalho	Análise de problemas Declaração de classes Relações entre classes

Notas:

Todas as classes devem pertencer a um package que não o por defeito. **A correta criação e colocação dos packages É um dos parâmetros de avaliação.**

A otimização de algumas das operações será beneficiada. Por outro lado, soluções pouco eficazes (desperdício de memória exagerado, código redundante, etc) serão penalizadas.

O código deve seguir as regras de codificação estabelecidas na disciplina.

O que se deve entregar:

- As regras para entrega serão anunciadas mais tarde.

Dúvidas:

Eventuais dúvidas podem ser esclarecidas junto dos docentes da disciplina. Pode também ser usada a plataforma de e-learning que terá um fórum onde poderão colocar dúvidas.

Os trabalhos ficarão sujeitos a uma defesa na aula prática/teórica da última semana de aulas.

Trabalhos idênticos serão anulados.