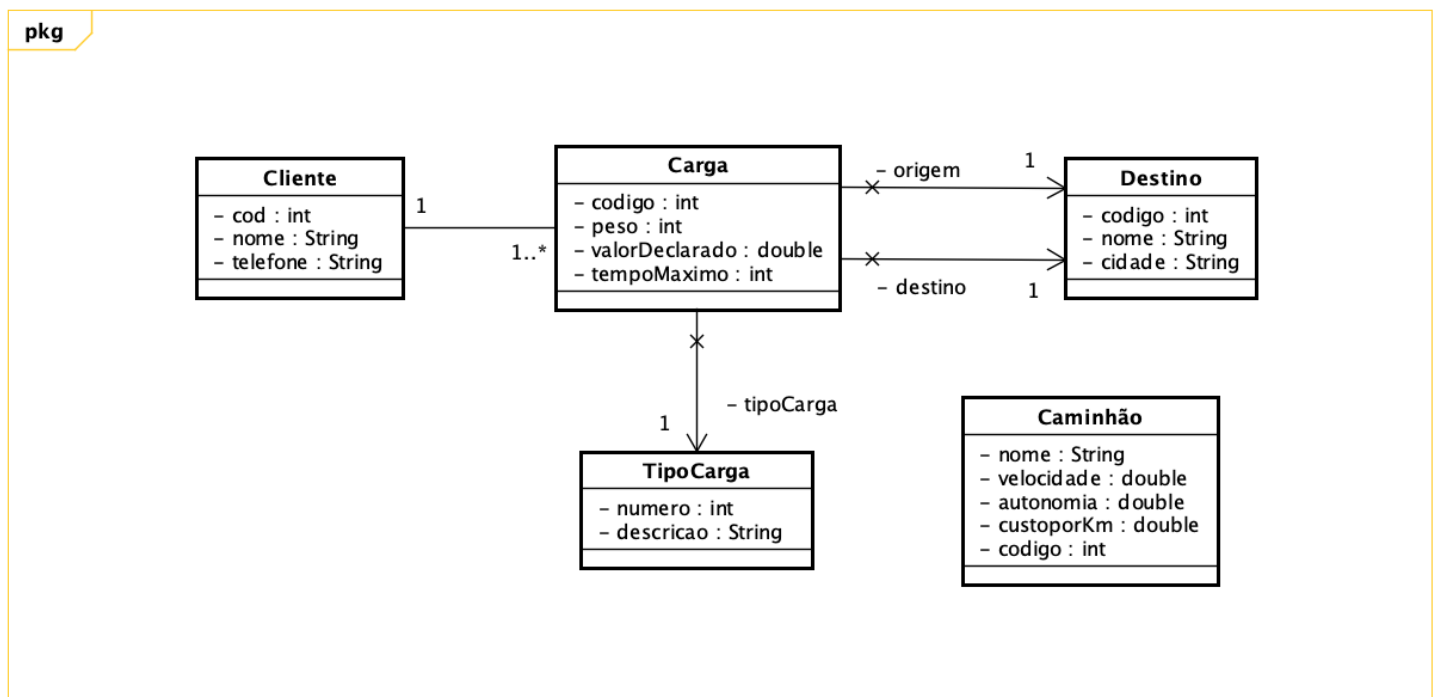


O ramo da empresa KonPOO é transporte de cargas terrestres. Este divide suas funções nas seguintes etapas: administração do transporte, conferência da carga, roteirização da entrega, controle do transporte de carga, descarregamento, roteirização e entrega da carga, análise de indicadores e atendimento ao cliente. Com base em algumas das funções descritas, a empresa solicita que seja desenvolvido um sistema para gerenciar: seus clientes, sua frota (caminhões) e seus fretes relacionados com as cargas transportadas.

Para o desenvolvimento do sistema algumas características foram listadas:

- ❑ Caminhões (frota): cada um possui código, nome único, autonomia e o valor do custo por km rodado.
- ❑ Destino: possui um código, um nome e a cidade.
- ❑ Cliente: possui um código, um nome e um número de telefone.
- ❑ Carga: um código único, tipo de carga, local de origem, local de destino, cliente, peso (toneladas), o valor declarado pelo cliente e o tempo máximo (em dias) para o frete.
- ❑ Tipo de carga: possui numero e descrição (Perecível: possui uma origem (String) e um tempo máximo de validade (int); Durável: que possui um setor (String) e o material principal (String).
- ❑ Possíveis situações da carga: Pendente, Locada, Cancelada, Finalizada. Ao ser cadastrada uma carga fica na situação Pendente; definindo um caminhão para a carga muda para a situação: Locada; carga foi entregue a situação muda para finalizada (liberando o caminhão). Para uma carga não atendida, a situação é cancelada.
- ❑ O cálculo do valor de um frete = *Preço por distância + Preço por peso*
- ❑ O preço por distância = *Distância entre origem e destino X Custo por Km*
- ❑ O preço por peso depende do tipo da carga: PERECIVEL, é: *peso X 2*; DURÁVEL, é: *peso X 1,5*.

Com base nos requisitos levantados, foram definidas as seguintes classes (inicialmente):



A partir disso, é necessário desenvolver as seguintes funcionalidades (tela cíclica com o usuário):

- ❑ **Cadastrar novo destino:** cadastra os dados de um destino. Se já houver destinos cadastrados, assume-se a distância de 50 Km entre cada um deles. Os destinos são mantidos em ordem crescente de identificador.
- ❑ **Cadastrar novo caminhão:** cadastra os dados de um caminhão. Se houver caminhão com o nome indicado mostrar uma mensagem de erro. Os caminhões são mantidos em ordem crescente de nome.
- ❑ **Cadastrar novo cliente:** solicita os dados de um cliente. Se o código já existir, retorna mensagem de erro. Os clientes são mantidos em ordem crescente de código.
- ❑ **Cadastrar novo tipo de carga:** solicita os dados de um tipo de carga. Se o número já existir, mostra uma mensagem de erro. Os tipos de carga são mantidos em ordem crescente de número.
- ❑ **Cadastrar nova carga:** solicita os dados de uma carga (se o código existir, mostra mensagem de erro); colocar a nova carga em uma fila de cargas pendentes; ao final mostra na tela os dados da nova carga cadastrada, incluindo os dados da origem e destino, cliente e tipo de carga. As cargas são mantidas em ordem crescente de código.
- ❑ **Consultar todas as cargas:** mostra todas as cargas cadastradas e todos os dados das cargas (origem e destino, cliente, do tipo de carga, situação, se possui caminhão designado, mostrar os dados do caminhão e o valor final do frete).
- ❑ **Alterar a situação de uma carga:** solicita o código de uma carga, mostra os dados da carga, solicita a nova situação [se não há carga com o código indicado, mostra uma mensagem de erro; se a carga estiver na situação FINALIZADO, não pode ser alterado e mostra uma mensagem de erro]).
- ❑ **Carregar dados iniciais:** carrega os dados num arquivo para o sistema; as cargas devem ser carregadas em uma fila de cargas pendentes; ao final da carga de dados, mostra todos os dados de destinos, caminhões, clientes, cargas. O formato do arquivo é de livre escolha.
- ❑ **Fretar cargas:** a partir da fila de cargas pendentes, verifica se é possível designar algum caminhão disponível para cada carga, e atualiza a sua situação. Se há algum caminhão com capacidade de fazer o frete, mas já está designado para outra carga, a carga retorna para a fila de fretes pendentes. Se não há nenhum caminhão com capacidade de fazer o frete, a carga muda para a situação CANCELADO [se não há cargas na fila de cargas pendentes, mostra uma mensagem de erro]).
- ❑ **Salvar dados:** salvar todos os dados cadastrados em um ou mais arquivos. Se houver algum problema para salvar deve mostrar uma mensagem de erro.
- ❑ **Carregar dados:** carrega todos os dados de um ou mais arquivos. Se houver algum problema no carregamento mostra uma mensagem de erro. A forma como irá carregar o arquivo é de livre escolha.
- ❑ **Finalizar sistema:** termina a execução.

**Definição do trabalho:**

O objetivo do trabalho é implementar o sistema acima descrito, e que atenda as seguintes restrições:

- ☐ É permitida a criação de novas classes, métodos, atributos e relacionamentos, mas as informações definidas no diagrama de classes original não podem ser removidas.
- ☐ Deve haver tratamento de exceções para que não haja falhas na execução.
- ☐ Deve utilizar coleções da API Java para o armazenamento de objetos.
- ☐ Deve possuir interface gráfica com o usuário.
- ☐ Deve ser entregue um relatório contendo uma breve descrição do que foi desenvolvido e:  
**Diagrama de classes** – diagrama com as classes do sistema contendo as alterações realizadas;  
**Coleções de dados** – descrição de coleções (listas, filas, pilhas, etc.) que foram utilizadas e em que partes do sistema;  
**Armazenamento de dados** – descrição da forma escolhida de armazenamento de dados em arquivos texto ou serialização.

**Critérios de avaliação:**

O trabalho será avaliado conforme os seguintes critérios:

**Nota = 15% apresentação (individual) + 85% implementação**

Implementação:

- Relatório: 1 pt.
- Armazenamento de dados em arquivos: 1 pt.
- Tratamento de exceções: 1 pt.
- Interface gráfica com o usuário: 1 pt.
- Uso de herança, polimorfismo e coleções: 1 pt.
- Implementação e execução do sistema conforme a descrição: 4.5 pt.

**Entrega:**

- ☐ Códigos-fonte do sistema (e demais arquivos necessários para a compilação do mesmo).
- ☐ Relatório do projeto.
- ☐ Deverá ser gerado um arquivo compactado (.zip ou .rar), com os itens acima e deve ser postado no link “Trabalho Final” no card TDE do Moodle.
- ☐ Data: **22/11/2023**, até 17h30min.

**Apresentação:**

- ☐ Data: **22/11/2023 (NP)**.

**Alguns Lembretes:**

- ☐ Modalidade: individual, dupla ou trio.
- ☐ Não serão aceitos trabalhos com erros de compilação. Programas que não compilarem corretamente terão nota zerada.