

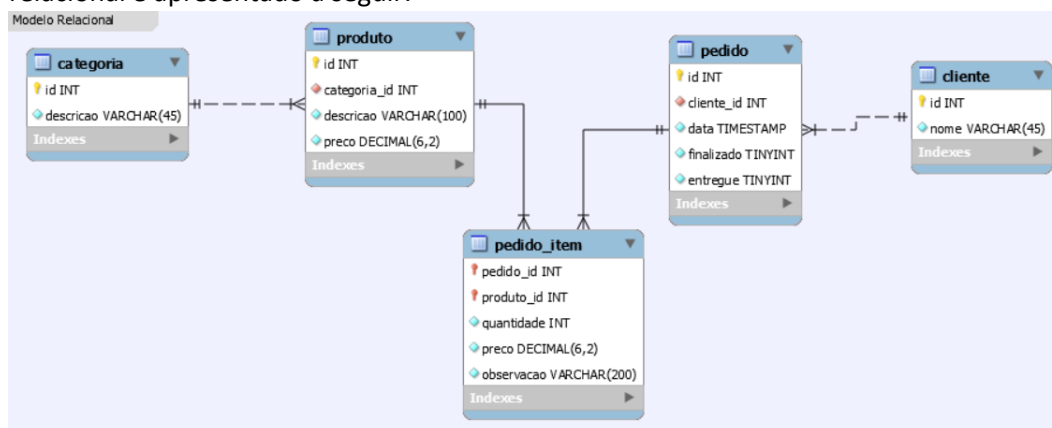
Leia com atenção:

1. Este trabalho vale 25% da nota final, deve ser realizado grupos de dois ou três integrantes;
2. A formação dos grupos deve ocorrer no dia da especificação do trabalho;
3. O código fonte deve ser entregue no ambiente virtual no **link Trabalho-Enviar**. **A RESPONSABILIDADE pela integridade do arquivo é sua**, ou seja, você deve verificar se o arquivo foi enviado com sucesso e se ele poderá ser aberto pelo professor. Arquivo corrompido terá nota zero.
4. A **apresentação** ocorre durante a aula do dia da entrega quando **cada integrante** do grupo deve **demonstrar e explicar** todas as funcionalidades implementadas. Na apresentação, todos os integrantes devem demonstrar uma parte do projeto, mostrando código fonte e funcionalidades.
5. O problema descrito a seguir deve ser resolvido mediante desenvolvimento de um programa em java utilizando conceitos de Orientação a Objetos vistos em aula, como herança, composição, separação em camadas, GUI, acesso a banco de dados, threads, entre outros.

Especificação

A cada dia mais pessoas escolhem a praticidade de pedir comida utilizando celular. Por outro lado, alguns usuários de Smartphones estão cansados de terem que instalar aplicativos diferentes o tempo todo, então alguns restaurantes optam por interagir com os seus clientes por meio dos aplicativos de trocas de mensagens instantâneas, o que demanda de um funcionário somente para isso. Sabendo disso, sua empresa enxerga uma oportunidade de negócio e resolve disponibilizar um sistema com um ChatBot para fazer pedidos pelo Telegram.

ChatBot é um programa de computador que tenta simular um ser humano na conversação com as pessoas. Seu grupo deve criar um Bot no aplicativo Telegram (forme o grupo de modo que ao menos um integrante tenha o aplicativo). É preciso abrir o Telegram, localizar o perfil “BotFather” e enviar o comando `/newbot`. Depois, precisa especificar um nome, como por exemplo “LancheriaMilliway” e um nome de usuário (que termine com a palavra bot, por exemplo “LancheriaMilliwaybot”). Por fim, você receberá um token para utilizar no seu programa Java o qual acessa o Bot via API e responde as requisições do usuário. A documentação de uso da API está disponível em <https://core.telegram.org/api>. Você pode utilizar como base o código disponibilizado no virtual para fazer a parte da comunicação devendo desenvolver o restante do sistema. Os dados serão armazenados em um banco de dados cujo modelo relacional é apresentado a seguir:



A tabela cliente vai armazenar os dados de todos os usuários que entrarem em contato pelo aplicativo (inserção automática). As tabelas categoria e produto terão seus dados mantidos pelo dono do restaurante de forma que você precisa fazer as telas de cadastro. A seguir são apresentados exemplos de registros nessas tabelas.

Exemplos de registros na tabela categoria	
Id	descricao
1	Hamburguer
2	Pastel
3	Pizza
4	Sushi

Exemplo de registros na tabela produto (lanches)			
id	categoria_id	descricao	preco
1	3	Pizza Pequena	30,00
2	3	Pizza Média	35,01
3	3	Pizza Grande	40,00
4	1	Hamburger Normal	20,00
5	1	Hamburger Gourmet	31,01

As tabelas pedido e pedido_item serão preenchidas durante a interação do usuário com o Bot. As tabelas a seguir exibem exemplos de registros.

Exemplo de registros na tabela pedido		
id	cliente_id	data
1	9999999	01/10/2018 19:00
2	8888888	01/10/2018 19:30

Exemplo de registros na tabela pedido_item				
pedido_id	produto_id	quantidade	preco	observacao
1	2	1	35,00	Borda de catupiri
1	5	2	28,00	Sem tomate

O programa deve ter no mínimo as seguintes telas:

- **(0,3 pts) Manutenção de Categoria:** permitir incluir, alterar e excluir categorias de lanches (somente será possível excluir se ainda não houver produto vinculado);
- **(0,7 pts) Manutenção de Produtos (lanches):** permitir incluir, alterar e excluir produtos (somente será possível excluir se ainda não estiver em algum pedido). Deve ser possível escolher a categoria em uma JComboBox.
- **(3,0 pts) Consulta de pedidos:** O sistema deve apresentar uma tela em formato mestre-detalle. Na primeira tabela exibir todos os pedidos que foram finalizados pelo cliente, mas ainda não foram entregues. Na segunda tabela listar os itens do pedido selecionado. Ao clicar no botão “Atualizar a Tela” deve ser refeita a consulta no BD. O botão “Finalizar Entrega” deve atualizar o campo entregue=1 para o pedido

selecionado e recarregar a tela (de forma que o pedido saia da tela). O layout deve ser semelhante ao apresentado a seguir:

Consulta de pedidos

Pedidos por Entregar

Nº Pedido	Cliente	Data/Hora	Valor Total
1	Linus Torvals	01/10/2019 19:05	97,03
2	Douglas Adams	01/10/2019 20:00	42,99
3	Bill Gates	02/10/2019 18:00	95,59
4	Ada Lovelace	03/10/2019 20:00	50,55

Itens do Pedido Selecionado

Cód. Produto	Descrição	Observação	Qtde	Vi. Unit.	Vi. Total
2	Pizza Média	Borda de cheddar	1	35,01	35,01
5	Hamburguer gourmet	Sem tomate	2	31,01	62,02

Atualizar a Tela

Finalizar Entrega

(2,0 pts) A interação do cliente com o Bot ocorre da seguinte forma:

1. Cliente envia uma mensagem qualquer;
2. Bot insere o registro na tabela cliente (se ainda não houver esse id) e responde com uma saudação padrão perguntando ao cliente qual categoria de lanches ele gostaria de olhar (listando as opções);
3. Supondo que o cliente escolha ver a categoria 3 – pizzas;
4. O Bot responde com a lista de produtos desta categoria (código, descrição e preço);
5. Supondo que o cliente escolha a opção 2 – Pizza Média;
6. O Bot pergunta a quantidade;
7. Supondo que o cliente responda 1;
8. O Bot pergunta se tem alguma observação;
9. Supondo que o cliente responda “Com borda de Cheddar”;
10. O Bot insere o corpo do pedido e o item de pedido e em seguida pergunta se o cliente quer adicionar mais algum produto no pedido;
11. Se o cliente responder sim, exibe novamente as categorias e repete os passos a partir do 3;
12. Se o cliente responder não, então atualiza o pedido como finalizado=1 e retorna uma mensagem agradecendo o pedido e informando que poderá retirar em 40 minutos.

Lembre-se que a cada interação o usuário pode informar uma entrada que não corresponde a um comando esperado. Então se o cliente informar na quantidade que quer “uma” ao invés de “1”, o Bot deveria ser inteligente o suficiente para tratar isso. Caso seja impossível interpretar o comando, retornar uma mensagem avisando que não entendeu e pedir a resposta novamente.

Lembre-se também que enquanto o Bot está respondendo a requisição do cliente 9999999, um outro cliente pode iniciar uma interação. Cada sequência de comandos deve ser gerenciada em um thread separado.

(2.0 pts) Avaliação geral da interface gráfica

- Utilizar menus para acessar as telas
- Layout deve permitir redimensionar a tela de forma adequada;
- Utilizar ícones nos botões;
- Utilizar teclas de atalho (Ctrl + I para incluir, Ctrl + S para salvar);
- Utilizar textos de ajuda para documentar os controles e comandos da tela;

(2.0 pts) Avaliação geral do sistema;

- Correta separação em camadas;
- Correta utilização do BD;
- Correto tratamento de exceções;
- Utilização de threads;
- Documentação e organização do código;

Funcionalidades opcionais

O aluno que não entregou algum exercício de aula poderá substituir até 2 notas de exercício por meio das atividades a seguir:

- Exportar a consulta de pedidos em um arquivo PDF (dica: relatório iReport ou iText);
- Fotos dos produtos enviadas no Telegram.

Bom trabalho