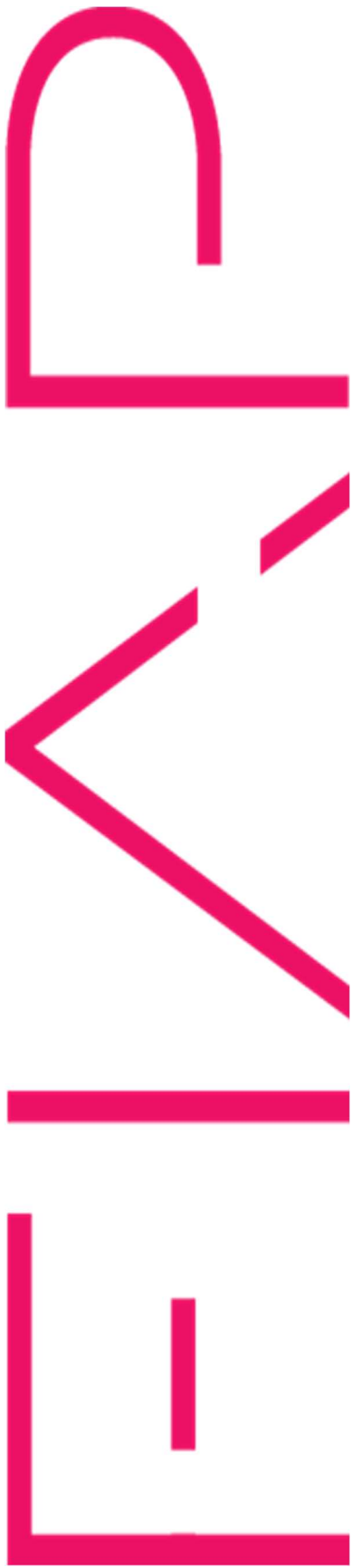


Challenge Banco PAN

2021



Sumário

<u>1</u>	<u>PROJETO</u>	<u>3</u>
<u>1.1</u>	<u>TEMA DO PROJETO</u>	<u>3</u>
<u>1.2</u>	<u>DESCRIÇÃO</u>	<u>3</u>
<u>1.3</u>	<u>TURMAS</u>	<u>3</u>
<u>1.4</u>	<u>ENTREGA</u>	<u>3</u>
<u>1.5</u>	<u>FORMAÇÃO DAS EQUIPES</u>	<u>3</u>
<u>1.6</u>	<u>LOCAL DA ENTREGA</u>	<u>3</u>
<u>1.7</u>	<u>HORÁRIOS</u>	<u>3</u>
<u>1.8</u>	<u>COMPOSIÇÃO DA MÉDIA</u>	<u>4</u>
<u>1.9</u>	<u>ENTREGA A TODOS OS PROFESSORES</u>	<u>4</u>
<u>1.10</u>	<u>ENTREGA ESPECÍFICAS PARA CADA DISCIPLINA</u>	<u>4</u>
<u>2</u>	<u>ENGENHARIA DE SOFTWARE</u>	<u>5</u>
<u>2.1</u>	<u>ENTREGA</u>	<u>5</u>
<u>3</u>	<u>CÓDIGOS DE ALTA PERFORMANCE</u>	<u>6</u>
<u>3.1</u>	<u>ENTREGA</u>	<u>6</u>
<u>4</u>	<u>NETWORKING FUNDAMENTALS AND SECURITY</u>	<u>7</u>
<u>4.1</u>	<u>ENTREGA</u>	<u>7</u>
<u>5</u>	<u>GESTÃO CORPORATIVA COM TI</u>	<u>8</u>
<u>5.1</u>	<u>ENTREGA</u>	<u>8</u>
<u>6</u>	<u>ESTATÍSTICA PARA SOLUÇÕES EM TI</u>	<u>9</u>
<u>6.1</u>	<u>ENTREGA 1</u>	<u>9</u>
<u>7</u>	<u>DESIGN E DESENVOLVIMENTO DE BANCOS DE DADOS</u>	<u>10</u>

<u>8</u>	<u>PROGRAMAÇÃO ORIENTADA A OBJETOS</u>	<u>11</u>
<u>8.1</u>	<u>ENTREGA</u>	<u>11</u>
<u>9</u>	<u>VÍDEO PITCH</u>	<u>12</u>
<u>9.1</u>	<u>ENTREGA</u>	<u>12</u>
<u>9.2</u>	<u>PROFESSORES RESPONSÁVEIS</u>	<u>14</u>

1 Projeto

1.1 Tema do projeto

O tema de projeto foi apresentado na reunião de kickoff, e está acessível nos links:

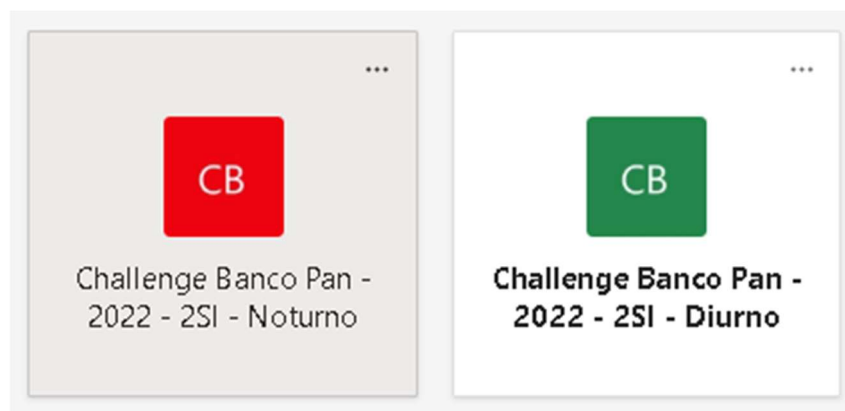
1º) Documento de apresentação do parceiro:

<https://fiapcom.sharepoint.com/:p:/s/ChallengeBancoPan-2022-2SI-Noturno/EVnD8ZE28wVBnj-xMqFW-bMBO4yldg1H0jR6j2FJlwZCQ?e=Z374jH>

2º) Vídeo gravado do evento de abertura:

https://fiapcom.sharepoint.com/:v:/r/sites/ChallengeBancoPan-2022-2SI-Noturno/Documentos%20Compartilhados/General/Material%20Kickoff/General-20220405_194032-Grava%C3%A7%C3%A3o%20de%20Reuni%C3%A3o.mp4?csf=1&web=1&e=lvOHmG

Esses arquivos estão disponíveis na sala/grupo do projeto Challenge, no TEAMS:



1.2 Descrição

O Banco PAN traz o desafio da sua área de CRM (Customer Relationship Management) quanto a qualificação de dados cadastrais dos seus clientes.

Hoje existem problemas de sincronização de dados de sistemas diversos que contém informações cadastrais dos clientes e produtos que operam.

O Banco PAN deseja o desenvolvimento de uma solução que possibilite padronizar a captura de dados, complementação, padronização e qualificação

dos dados para que o cadastro seja feito de forma correta e completa, criando harmonização de dados.

A harmonização permitirá a consulta correta da situação operacional do cliente e a oferta de produtos e serviços direcionados à necessidades desse cliente.

É de interesse do parceiro do challenge que sua equipe de desenvolvimento de projeto trabalhe uma solução envolvendo o desenho técnico e construção do software de aplicação, banco de dados e infraestrutura tecnológica para garantir alta disponibilidade, confiabilidade, integridade e desempenho nas transações que envolvem alimentação, modificação e consumo em consultas de dados cadastrais dos clientes do Banco PAN.

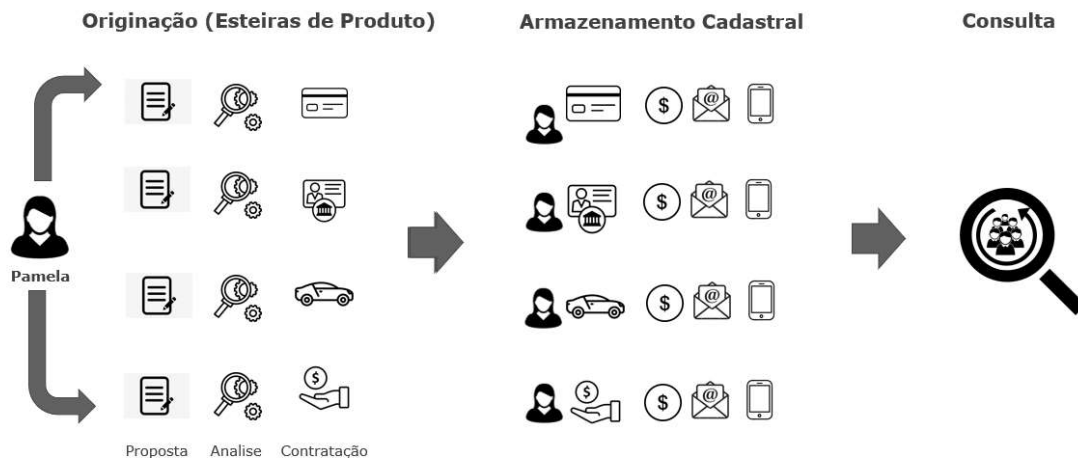
A limpeza e complementação de dados que hoje estão comprometidos em qualidade também não é alvo inicial do desafio.

Detalhes do pedido feito pelo Banco PAN:

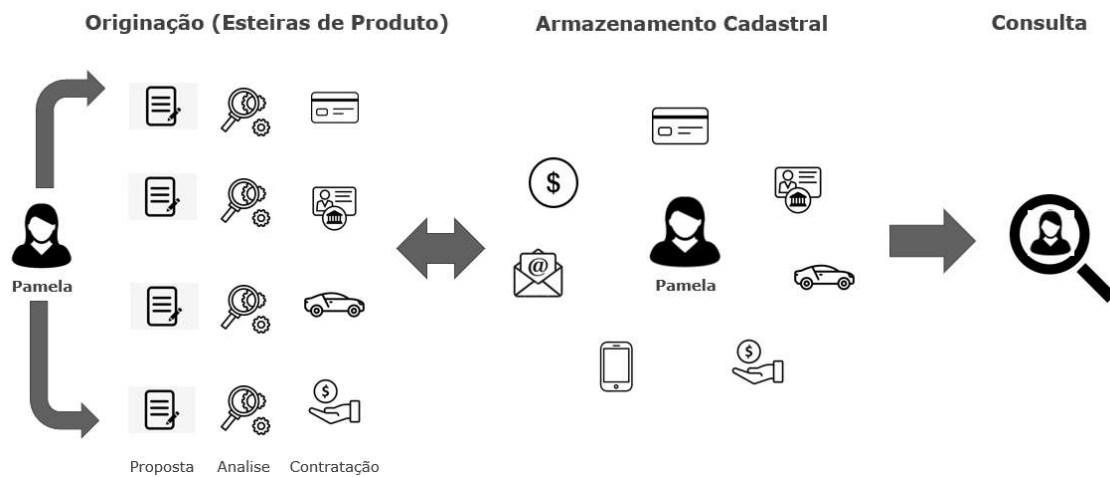
“Desejamos remodelar nosso banco de dados de modo a passar a tratar o cliente de maneira única e centralizada. O nosso sistema tem como objetivo ser o repositório cadastral de clientes do Pan, centralizando a informação em um único sistema.

Atualmente, nosso banco de dados está estruturado na visão de cliente por produto, de modo que a mesma pessoa apareça de maneira duplicada em uma visão cliente. Queremos sair de uma visão orientada ao produto, para uma visão orientada ao cliente:

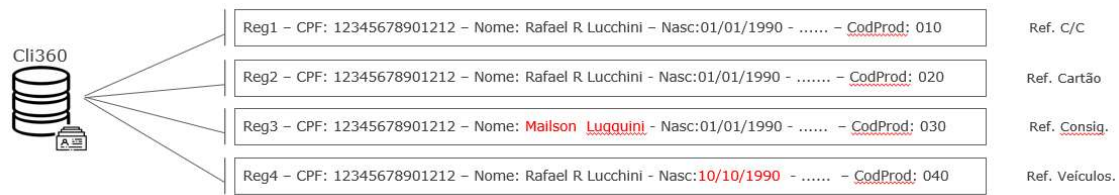
AS IS



TO BE



O dado atualmente está organizado da seguinte maneira



É importante ressaltar que temos dados voláteis (celular, email, endereço, etc.) e dados não voláteis (nome, sexo, CPF, etc.), sendo que são esses dados não voláteis que caracterizam o conceito de Pessoa dentro nosso sistema.

Uma pessoa contém CPF, nome, data de nascimento e sexo como atributos principais. As informações de contato (telefone, celular, email, etc.), endereço (rua, número, cidade, etc.) e produtos (cartões, consignado, imobiliário, etc.) são atributos desse cliente, mas que fornecem informações importantes para as áreas do banco.

Importante ressaltar que os dados de contato e endereço também precisam estar presentes na visão do produto, pois há diretivas legais que nos fazem precisar ter essa informação.

Os dados podem ser atualizados por vários produtos e chegam ao sistema de maneira online, porém assíncrona. No momento que a informação chega ao sistema, é preciso tomar a decisão sobre a qualidade da informação e sobre acatá-la ou não.

A consulta dessas informações ocorre, de maneira síncrona e online, em momentos críticos de diversos fluxos de negócio, de modo que uma indisponibilidade dessas informações traz uma perda importante para a instituição.

Perguntas a responder no desafio/linhas de solução:

Como você modelaria o banco de dados para atender aos requisitos descritos acima?

Qual seria a tecnologia usada e por quê?

Como você garantiria que a informação estaria disponível?

Como você estruturaria os serviços para receber as inclusões e/ou alterações dos dados?

Como você estruturaria os serviços para receber as consultas dos dados?

Extra-mile: Como vocês classificaria a qualidade do dado e definiria o melhor dado?”

1.3 Eventos avaliativos

Comtemplam as turmas do 2º ano do curso de Sistemas de Informação.

São divididos em dois grupos:

1º) Avaliações pelo parceiro (item 1.4 deste documento)

2º) Avaliações pelos docentes (item 2 deste documento)

1.4 Entregas gerais do primeiro e segundo semestre para avaliação pelo parceiro

Primeira semana de Maio: proposta do projeto em formato de PITCH (entrega de um documento em formato PDF contendo “slides” de explicação do projeto em formato de PITCH de negócio).

Final de Agosto: entrega do protótipo funcional (apresentação de vídeo demonstrando como a solução vai funcionar). O grupo deve apresentar telas navegáveis, simulações animadas de cargas, leituras, atualizações, exclusões de dados em processamentos batch (em lote e sem interface de frontend de usuário) – os dados usados nas simulações não precisam ser reais.

Final de Outubro: apresentação final da solução funcionando para bancas examinadoras de professores que vão selecionar os 10 (dez) melhores trabalhos para uma apresentação ao vivo para o parceiro de negócio: o Banco PAN. Essas apresentações ao vivo serão agendadas e comunicadas com antecedência.

As duas primeiras entregas (PITCH e Vídeo de protótipo funcional) serão feitas pelos alunos em uma ÁREA DE ENTREGA DE TRABALHOS criada pelo Scrum Master de cada turma. Cabe ao Scrum Master repassar os materiais para o parceiro do challenge, o qual deve retornar suas considerações (feedback) por

escrito para o Scrum Master em uma planilha de avaliação que será disponibilizada em um serviço de nuvem. Esses feedbacks serão repassados pelo Scrum Master a cada grupo, individualmente, usando a área de comentários no espaço de ENTREGA DE TRABALHOS.

1.5 Formação das Equipes

As equipes devem ser formadas no mês de abril e são permitidas até 5 (cinco) pessoas por grupo, devendo todos integrantes pertencerem à mesma turma para facilitar a evolução dos trabalhos.

1.6 Local da Entrega

Cada professor irá abrir na área de Challenge Sprints do portal do aluno uma entrega para a sua disciplina específica, basta um aluno da equipe fazer o upload da atividade, no entanto, todos os nomes dos componentes e seus respectivos RMs devem constar no documento, link, programa etc.

1.7 Horários de atendimento

As dúvidas técnicas sobre o desenvolvimento de entregas específicas para os professores devem ser sanadas no curso das aulas de cada semestre, com cada professor de cada disciplina.

A equipe do Banco PAN definiu uma agenda periódica de disponibilidade para sanar dúvidas e dialogar com os estudantes sobre seus projetos. O horário de atendimento será às segundas-feiras, das 17h às 18h. Para consumir essa agenda cada grupo/equipe do challenge deve preencher a planilha de agendamento disponibilizada no canal geral/arquivos da equipe geral do 2º ano de SI para o Challenge Banco PAN no TEAMS. A planilha contém os dias e horários dos plantões e o grupo deve colocar o seu nome no dia e horário de interesse para que a equipe do Banco PAN possa se organizar para as conversas. Uma sala será aberta no canal geral do Challenge no TEAMS, no momento do encontro, bastando que o estudante ingresse na sala para conversar.

1.8 Composição da média

Será calculada uma média com as notas de cada professor com base nos entregáveis quando se aplicar.

1.9 Entrega específicas para cada disciplina

Em caso de desenvolvimento de projeto envolvendo tecnologia com implementação 100% elaborada pelo grupo ou junção entre solução de terceiros e implementação própria, desenvolva os entregáveis específicos a seguir:

2 Engenharia de Software

Prof. Renato Parducci e Prof. Junior Freitas do Amaral

2.1 Entrega

1ª – semestre

1ª ENTREGA: Documento de Visão e Escopo do projeto com descrição da proposta de desenvolvimento de sistema, incluindo:

Documento em formato PITCH, com slides ou texto corrido, explicando os seguintes tópicos:

- Qual o problema do Banco PAN que será resolvido com a sua solução;
- Qual a solução idealizada (descrição geral das funções que ela entrega e tecnologia que integra);
- Qual o público alvo (que vai usar/consumir a sua solução), no caso, quem será beneficiado pelas melhorias promovidas no Banco;
- Avaliação da existência de produtos/soluções similares/concorrentes;
- Diferenciação competitiva da sua solução;
- Entregue um arquivo PDF com o conteúdo.

2ª ENTREGA: Lista de requisitos do projeto, registrada em ferramenta definida pelo professor, incluindo:

- Painel (KANBAN) de controle de status de projeto;
- Cartões de requisitos com descrição de cada um, contendo todas as ideias/propostas inicialmente levantadas para o projeto;
- Identificação do tipo do requisito (funcional ou não funcional);
- Indicação dos requisitos inviabilizados e viabilizados;
- Indicação de prioridade, diferenciando requisitos obrigatórios, necessários e desejados;
- Entregue um arquivo PDF com o conteúdo.

2ª – semestre

1ª ENTREGA: Documento modelagem funcional do sistema, incluindo:

- Diagramas de Caso de Uso do seu projeto completo. Faça o diagrama de contexto e todos os detalhes necessários (Diagramas de refinamento) para explicar as funcionalidades do seu software.

- Entregue um arquivo PDF com os desenhos feitos no ASTAH (use o recurso de Impressão de Diagramas em PDF do ASTAH/Preview-print multi), demonstrado em aula.

2ª ENTREGA: Diagrama de Atividades que explica do uso do sistema com a operação das funções previstas no software/hardware:

- Siga o padrão UML ensinado e faça seu desenho usando o ASTAH.

- Entregue um arquivo PDF (use o recurso Print multi do ASTAH).

3 Códigos de Alta Performance

Profa. Patrícia Magna

3.1 Entrega

1ª – semestre

1ª ENTREGA:

3.1 Entrega

1ª – semestre

1ª ENTREGA:

Supondo que sua equipe desenvolva uma aplicação que permita uma melhor interação com clientes e o banco PAN, ou seja, seu projeto é uma solução em software (app ou aplicativo) o ideal é que sua equipe receba periodicamente feedback dos usuários de seu produto para que sua solução possa ser aprimorada.

Assim, a tarefa da disciplina nessa etapa é:

- Criar o tipo de dado PilhaMensagens que utilize a implementação sequencial de pilha para armazenar os textos das mensagens enviadas pelo usuário.
- Criar o tipo de dado Usuário que deve ter como atributos: identificador e nome do usuário e pilha de feedback e o status do atendimento. O status seria “ok” (quando não há mensagens para ser lidas) ou “não ok” (quando a pilha tiver mensagens ainda não lidas). As operações que devem compor esse tipo de dado (métodos em JAVA) são:

o construtor, que além de inserir nome e id do cliente também inicia a pilha de mensagens e status “ok”;

o enviaMsg(msg): que empilha msg lida e altera status para “não ok”

o leMsg(): que desempilha todas as mensagens, apresentando na tela de saída, e altera status para “ok”.

A classe SAC que contém o método main() (disponível para realizar testes) que utiliza os tipos de dados solicitados foi elaborado é apresentado a seguir:

```
public class SAC {

    public static void main(String[] args) {

        Scanner le = new Scanner(System.in);

        System.out.print("Digite nome do cliente: ");
```

```
String nome = le.nextLine();

System.out.print("Digite id do cliente: ");

int id = le.nextInt();

Usuario cliente = new Usuario(id, nome);

int opcao;

do {

    System.out.println(" \n 0- para sair do sistema ");

    System.out.println(" 1- para enviar mensagem de feedback");

    System.out.println(" 2- ler e responder mensagem de feedback ");

    System.out.print("Digite opção: ");

    opcao = le.nextInt();

    switch (opcao) {

        case 0:

            System.out.println("Finalizado o atendimento");

            break;

        case 1:

            System.out.println("Digite mensagem a ser enviada: ");

            le.nextLine();

            String msg = le.nextLine();

            cliente.enviaMsg(msg);

            break;

        case 2:

            if (cliente.getStatus() == "não ok")

                cliente.leMsg();

            break;
```

default:

```
System.out.println("Opção inválida!");

}

} while (opcao != 0);

}
```

Você pode aqui estar se perguntando por que pilha? O exemplo de mensagens enviadas pelo usuário mostra o que é bastante comum.

Quando você e sua equipe lerem as mensagens veriam que o problema foi resolvido, podendo responder dessa forma às mensagens:

A entrega será do projeto JAVA a ser feita pelo portal (entrega de trabalho).

Apenas 1 trabalho deve ser entregue por equipe.

2ª ENTREGA:

Na tarefa implementada no sprint 1 da challenge, o sistema de recebimento periódico de feedbacks de usuários de seu produto para o parceiro, foi projetado e construído para que 1 usuário pudesse “empilhar” mensagens e um cliente da sua equipe realizava o envio de respostas “desempilhando” todas as mensagens daquele específico usuário.

Mudando agora a abordagem para o atendimento de envio de mensagens de vários usuários de sua solução, considere a seguinte forma de implementar esse sistema outro tipo de atendimento.

Os clientes são cadastrados no seu sistema fazendo com que as informações (nome e id) de cada cliente sejam armazenadas em um vetor de clientes.

Quando um cliente quiser enviar uma mensagem, ele deve usar a sua identificação (id) para que seja permitido o envio. A mensagem com o id e nome do cliente deve então ser inserida na fila de mensagens a serem respondidas. Caso o id do cliente não exista, este deve se cadastrar no sistema (inserindo no vetor de clientes).

A simulação do atendimento das mensagens consiste em apenas retirar da fila a mensagem e escrever na tela de saída o conteúdo da mensagem enviada e as informações do usuário que a enviou.

Para essa 2ª entrega da challenge, a tarefa dessa disciplina é:

- Implementar a classe Cliente: nome e id devem ser os atributos;
- Implemente a classe Mensagem: lembrando que junto com o texto da mensagem em si devem ser incluídos id e nome do cliente que enviou a mensagem;

- Implementar a classe FilaMensagens: que armazena mensagens para serem lidas e respondidas.

A classe Chat que contém o método main() (disponível para realizar testes) que utiliza os tipos de dados solicitados foi elaborado é apresentado a seguir:

```
public class Chat {

    public static void main(String[] args) {

        Scanner le = new Scanner(System.in);

        FilaMensagens fila = new FilaMensagens();

        fila.init();

        Cliente clientes[] = new Cliente[10];

        int n=0;

        int opcao;

        do {

            System.out.println(" \n 0- para sair do sistema ");

            System.out.println(" 1- para enviar mensagem pelo chat");

            System.out.println(" 2- ler e responder mensagem de chat ");

            System.out.print("Digite opção: ");

            opcao = le.nextInt();

            switch (opcao) {

                case 0:

                    System.out.println("Finalizado o atendimento");

                    break;

                case 1:

                    System.out.print("Digite seu id:");

                    int id = le.nextInt();

                    int i;
```



```

boolean achou = false;

i=0;

while (i<n && !achou) {

if (clientes[i].getId() == id)

achou = true;

else

i++;

}

if (!achou) {

System.out.println("Fazendo o cadastro deste id...");

System.out.print("Digite seu nome:");

String nome = le.next();

Cliente cli = new Cliente(id,nome);

clientes[n] = cli;

n++;

}

System.out.println("Digite mensagem a ser enviada: ");

le.nextLine();

String msg = le.nextLine();

Mensagem m = new Mensagem(clientes[i], msg);

fila.enqueue(m);

break;

case 2:

if (!fila.isEmpty()) {

m = fila.dequeue();

```

```

System.out.println("Enviado pelo cliente: "+m.getCliente());

System.out.println("Mensagem: " + m.getMsg());

}

else

System.out.println("Não há mensagens na fila");

break;

default:

System.out.println("Opção inválida!");

}

} while (opcao != 0);

}

}

```

A entrega será do projeto JAVA (.zip) a ser feita pelo portal (entrega de trabalho).

Apenas 1 trabalho deve ser entregue por equipe.

2ª – semestre

1ª ENTREGA:

O objetivo deste trabalho é elaborar um sistema para priorizar o contato do banco com os clientes visando oferecer novos serviços e aplicações bancárias mais rentáveis.

Cada cliente do banco é cadastrado nesse sistema de priorização de contato pelo gerente da conta. Este deve selecionar um certo número de clientes que considera “de interesse” para o banco. Assim, o gerente inscreve o cliente preenchendo uma ficha com seus dados, conforme mostra registro a seguir):

Nome:

CPF/CNPJ:

Número da Conta:

Tipo de Conta (Física ou Jurídica)

Saldo em Aplicações:

O registro de cada cliente é inserido na árvore de busca binária (ABB), sendo que o critério de organização dos registros na ABB é o saldo em aplicações dos clientes.

De tempo em tempos esse cadastro é usado para que um novo serviço e/ou aplicação deve ser oferecido para os clientes.

Para simulação deste sistema o programa em JAVA deve ter um menu principal com as seguintes opções:

1. Inscrição de cliente: essa opção é usada para ler do teclado todas as informações do cliente e realizar o armazenamento do registro do candidato na ABB correspondente ao tipo de conta, ou seja, ABB de pessoas físicas ou ABB de pessoas jurídicas.

2. Oferta de novo serviço e/ou aplicação: é usada para ler do teclado o saldo mínimo necessário para a oferta a ser feita e qual tipo de cliente (pessoa física ou jurídica). Em seguida, o programa deve percorrer a ABB do tipo de conta exigida pela oferta a fim de buscar todos os clientes que se adequem a oferta, gerando uma lista com os clientes que devem ser contactados (seus registros) em ordem decrescente de saldo em aplicações (obrigatoriamente a ordem deve ser gerada pelo percurso da ABB).

Com a lista de clientes aptos para receber a oferta pronta o programa deve:

I. Retirar o registro do cliente com maior saldo da lista gerada para que o setor de divulgação entre em contato com o cliente e este pode aceitar ou não a oferta do banco. Assim, se o cliente:

- Aceitar a oferta, o programa deve apresentar uma mensagem na tela de saída para avisar este fato. Em seguida, o registro do cliente deve ser retirado também da ABB, uma vez que apenas 1 oferta deve ser feita para cada cliente em um determinado período de tempo.

- Não aceitar a oferta, o programa deve apresentar uma mensagem na tela de saída para avisar a recusa por parte do cliente e o registro do cliente permanece na ABB

II. Programa prossegue retirando o próximo cliente da lista que deve ser contactado, até esvaziar a lista.

3. Encerra o programa: com essa opção se as ABBs ainda possuem registros de clientes, estas devem ser esvaziadas apresentando os registros de clientes que ainda estavam armazenados e só então o programa deve ser encerrado.

A aplicação já está parcialmente implementada e deve ser usada para realização do projeto.

```
public class DivulgaOfertas {
```

```

public static void main(String[] args) {

    Scanner le = new Scanner(System.in);

    /*

    * Cria a uma árvore de busca binária para cada tipo de conta

    * (pessoa física ou jurídica)

    */

    int opcao, op, numeroConta;

    String nome, cpfCnpj;

    String tipoConta = null;

    double saldo;

    do {

        System.out.println(" 0- Encerrar o programa");

        System.out.println(" 1- Inscrição cliente");

        System.out.println(" 2- Oferta de novo serviço e/ou aplicação");

        opcao = le.nextInt();

        switch (opcao) {

            case 1:

                System.out.print("Digite nome: ");

                nome = le.next();

                System.out.print("Digite cpf: ");

                cpfCnpj = le.next();

                System.out.print("Digite número da conta: ");

                numeroConta = le.nextInt();

            do {

                System.out.print("Digite 1- Pessoa Física 2- Pessoa Jurídica: ");

```

```

op = le.nextInt();

switch (op) {

case 1:

tipoConta = "Física";

break;

case 2:

tipoConta = "Jurídica";

break;

default:

System.out.println("Opção inválida ");

op = -1;

}

} while (op == -1);

System.out.print("Informe saldo em aplicações R$: ");

saldo = le.nextDouble();

/*

* Intancia um objeto da classe Cliente e insere na ABB correspondente a tipo de conta

*/

break;

case 2:

System.out.print("Qual tipo de conta a oferta se destina? ");

do {

System.out.print("Digite 1- Pessoa Física 2- Pessoa Jurídica: ");

op = le.nextInt();

switch (op) {

```

```
case 1:

tipoConta = "Física";

break;

case 2:

tipoConta = "Jurídica";

break;

default:

System.out.println("Opção inválida "); op = -1;

}

} while (op == -1);

System.out.print("Qual o valor de saldo mínimo exigido: R$ ");

saldo = le.nextDouble();

/*

* Fazendo uso de um método da classe ABB, desenvolvido para este problema, uma
lista de clientes

* aptos para a oferta é gerada.

* Nesse trecho de programa que tentar fazer o contato com todos os clientes presente na
lista.

*/

break;

}

} while (opcao != 0);

System.out.println("Clientes que não aceitaram ou não estavam adequados para a
oferta");

/*

* Esvazia as ABBs apresentando todos os clientes que aguardam nova oportunidade

*/
```

```
le.close();
```

```
}
```

```
}
```

3.1.1.1 Orientações para Elaboração de Itens Obrigatórios

Para implementar a classe ABB devem ser usados como base os métodos desenvolvidos em sala para construção da ABB, fazendo as alterações necessárias para as atividades explicadas no enunciado

A definição da classe Cliente tem como única obrigatoriedade conter os atributos descritos no enunciado. Fica a critério do grupo definir métodos construtores ou outros recursos da POO.

A lista utilizada para construir a opção 2 pode ser usada a estrutura de dados que o grupo achar mais conveniente, com apresentação de justificativas técnicas.

É incluído nesse enunciado o método main() da aplicação de Divulgação, sendo que é obrigatório o seu uso para inserção dos itens de programação que devem ser inseridos.

3.1.1.2 Orientações da Avaliação

Serão analisados e avaliados:

1. Criação correta da classe Cliente e seu correto uso como dado inserido na ABB (1,0)
2. Criação da classe ABB (com os métodos necessários para opção 1 do menu) com inserção organizada pelo saldo em aplicações na ABB do seu tipo de conta. (2,0)
3. Criação da classe escolhida para implementar a lista de clientes aptos para a oferta a ser divulgada ou uso de alguma estrutura de dados do JAVA (inserir comentário com justificativa pela escolha) (1,0)
4. Elaboração de um método da classe ABB que gera lista de clientes aptos para a oferta. (1,0)
5. Geração da lista de clientes (elementos da classe Cliente) contendo os clientes aptos para divulgação da oferta em ordem decrescente de saldo, fazendo uso do método elaborado no item anterior (1,0)
6. Apresentação da lista com os registros de clientes gerada no item anterior. (1,0)
7. Remoção do registro do cliente que aceitou a oferta do banco, mantendo organização da ABB (2,0)
8. Finalização do programa esvaziando ABB (1,0)

O projeto JAVA deve ser zipado e entregue como um único documento.

OBS: Trabalhos iguais serão zerados

2ª ENTREGA:

3.1.1.3 Descrição do projeto

Esse projeto tem como objetivo realizar uma comparação de eficiência para localizar registro de cliente usando uma AVL e uma ABB.

Deverão ser criadas tanto a ABB como a AVL contendo como dado os registros dos clientes. Cada cliente será identificado apenas por seu CPF, número da conta e saldo (dado de cada nó da árvore).

Supondo que o banco em questão tenha 1 mil clientes, o programa iniciará gerando um vetor com 1 mil números de contas aleatórios (para evitar valores repetidos gerar valores de 0 a 1 milhão), para o saldo também devem ser gerados valores aleatórios e, finalmente, para os CPFs dos clientes podem ser usados (cpf1, cpf2, etc) e esses são inseridos nas árvores AVL e ABB.

Supondo que diariamente são feitos acessos à 300 clientes cadastrados nas árvores, o programa deve fazer que até 300 números de contas sejam procurados em ambas as árvores e a quantidade total de comparações necessárias para pesquisar todos os 300 clientes em cada tipo de árvore deve ser apresentada.

Um texto contendo uma análise sobre os resultados obtidos e apresentação de pontos positivos e negativos de cada tipo de árvore deve ser entregue. É importante observar que para uma análise mais correta, devem ser realizadas várias medidas para fazer a análise em relação às médias das medidas obtidas.

3.1.1.1 Orientações para Elaboração de Itens Obrigatórios

Para implementar as classes ABB e AVL devem ser usados como base os métodos desenvolvidos em aula para suas elaborações.

Os números de contas devem ser gerados aleatoriamente para serem inseridos nas árvores, bem como alguma forma de selecionar os 300 clientes que irão ser pesquisados em cada dia.

Use a criatividade e descrevam como construíram o programa.

3.1.1.1 Orientações da Avaliação e Entrega

Serão analisados e avaliados:

1. Criação da classe ABB com inserção organizada pelo número da conta do cliente.
(1,0)
2. Criação da classe AVL com inserção organizada pelo número da conta do cliente.
(1,0)

3. Elaboração do método de consulta do número da conta do cliente., retornando o número de comparações para encontrar (1,0)
4. Elaboração do programa que implementa o que foi descrito na seção da descrição do projeto. (3,0)
5. Redação do documento que analisa os resultados obtidos e apresenta discussão sobre os pontos positivos de cada tipo de árvore (4,0)

O projeto JAVA e o documento (formato pdf) com a análise dos resultados devem ser zipados e entregues como um único documento.

OBS: Trabalhos iguais serão zerados

4 Networking Fundamentals and Security

Prof. Mauro Bernardes

4.1 Entrega

1ª – semestre

1ª ENTREGA:

Definição de uma topologia de redes com identificação dos elementos intermediários que irão compor a infraestrutura de comunicação a ser utilizada pela solução. A entrega deverá ser no formato de um arquivo gerado no software Cisco Packet Tracer e deverá ocorrer utilizando o Portal da FIAP, na área da disciplina de 'Network Fundamentals and Security'. **IMPORTANTE:** para a correção desta atividade o arquivo entregue deverá estar no formato .PKT.

2ª ENTREG A:

Apresentação da Topologia de redes, com detalhes do esquema de endereçamento em cada camada do modelo TCP/IP. A topologia a ser entregue deverá propor um ambiente de rede para a implantação da solução proposta. A entrega deverá ser no formato de um arquivo gerado no software Cisco Packet Tracer. A topologia deverá possibilitar a análise e a configuração do endereçamento em cada dispositivo que compõe a topologia. A entrega deverá ocorrer utilizando o Portal da FIAP, na área da disciplina de 'Network Fundamentals and Security'. **IMPORTANTE:** para a correção desta atividade o arquivo entregue deverá estar no formato .PKT.

2ª – semestre

1ª ENTREGA: Construa uma topologia de redes de comunicação utilizando o software Packet Tracer e que possa representar parte da rede de comunicação da empresa parceira.

A topologia a ser construída por sua equipe deverá utilizar como ponto de partida um endereço de rede Classe C Público (cada equipe deverá propor livremente o endereço de rede Classe C Público que irá utilizar para configuração da sua topologia). Um vez escolhido o endereço Classe C Público pela equipe, a topologia deverá conter, **NO MÍNIMO**, os seguintes elementos:

- a. 1 (uma) nuvem representando a internet;
- b. 1 (um) roteador que conecte a empresa à nuvem internet;

c. (pelo menos) 3 segmentos de redes locais (3 subredes) conectadas ao roteador. (Aqui a equipe deverá partir do endereço de rede Classe C Público escolhido e utilizar a técnica de CIDR para a divisão desse endereço de rede em subredes para configuração de cada segmento);

d. alocação de endereços IPs para os hosts utilizando DHCP;

e. um texto na topologia informando:

- o endereço de Rede Classe C Público escolhido
- a máscara de subrede que será utilizada após a divisão utilizando CIDR
- a faixa de endereços IP utilizada em cada segmento

A versão final a ser entregue deverá estar configurada para permitir a troca de informações entre todos os hosts (isso poderá ser verificado por meio de testes com o comando Ping).

A entrega deverá ser de um arquivo .pkt, com a topologia configurada e compactado na versão .zip

(IMPORTANTE: para a correção desta atividade o arquivo entregue deverá estar no formato .PKT).

2ª ENTREGA:

Construa uma topologia de redes de comunicação utilizando o software Packet Tracer e que possa representar parte da rede de comunicação da empresa parceira do challenge.

A topologia a ser construída por sua equipe deverá utilizar como ponto de partida um endereço de rede Classe C Público (cada equipe deverá propor livremente o endereço de rede Classe C Público que irá utilizar para configuração da sua topologia). Um vez escolhido o endereço Classe C Público pela equipe, a topologia deverá conter, NO MÍNIMO, os seguintes elementos:

- a. 1 (uma) nuvem representando a internet;
- b. 1 (um) roteador que conecte a empresa à nuvem internet;
- c. (pelo menos) 3 segmentos de redes locais (3 subredes) conectadas ao roteador. (Aqui a equipe deverá partir do endereço de rede Classe C Público escolhido e utilizar a técnica de CIDR para a divisão desse endereço de rede em subredes para configuração de cada segmento);

d. alocação de endereços IPs para os hosts utilizando DHCP;

e. um texto na topologia informando:

- o endereço de Rede Classe C Público escolhido
- a máscara de subrede que será utilizada após a divisão utilizando CIDR
- a faixa de endereços IP utilizada em cada segmento

A versão final a ser entregue deverá estar configurada para permitir a troca de informações entre todos os hosts (isso poderá ser verificado por meio de testes com o comando Ping).

A entrega deverá ser de um arquivo .pkt, com a topologia configurada e compactado na versão .zip

(IMPORTANTE: para a correção desta atividade o arquivo entregue deverá estar no formato .PKT).

5 GESTÃO CORPORATIVA COM TI

Prof. Junior Freitas do Amaral e Prof. Paulo Sampaio

5.1 Entrega

1ª – semestre

1ª ENTREGA:

O Planejamento é a atividade mais importante em quaisquer instituições, e o seu maior foco é na otimização dos resultados das organizações, por meio da melhoria dos processos do negócio.

Com essa afirmação, pensando no challenge do parceiro, e que vocês devem encarar como uma atividade da Organização.

Solicito que a equipe construa para cada uma das disciplinas (do seu curso), que solicitaram atividade, as seguintes entregas:

O planejamento de atividades;

Usando a metodologia:

- Matriz 5W2H como plano de ação.

Lembrando que há mais de uma atividade dentro de cada um dos trabalhos solicitados pelos professores.

2ª ENTREGA:

Solicito que a equipe construa para cada uma das disciplinas (do seu curso), que solicitaram atividade, as seguintes entregas:

O planejamento de atividades;

Usando a metodologia:

- Ciclo PDCA de controle e melhoria contínua de processos e produtos.

Lembrando que há mais de uma atividade dentro de cada um dos trabalhos solicitados pelos professores.

2ª – semestre

1ª ENTREGA:

A análise de desempenho também é um processo importante em qualquer iniciativa, e o foco está sempre na otimização dos resultados das organizações, por meio da melhoria dos processos do negócio.

Com essa afirmação, pensando no challenge do parceiro, e que vocês devem encarar como uma atividade da Organização.

Solicito que a equipe construa para cada uma das disciplinas (do seu curso), que solicitaram atividade, as seguintes entregas:

A medição e análise de desempenho do projeto.

Para tanto deverão utilizar o método da Análise de Pareto como ferramenta de medição.

É importante que sua equipe anote todas as dificuldades encontradas em cada uma das disciplinas apontando a quantidade de vezes que ela ocorreu e o tipo da dificuldade.

2ª ENTREGA:

As equipes deverão apresentar um relatório de Lições Aprendidas (Lessons Learned) utilizando modelo a ser disponibilizado pelo professor em tempo hábil.

O principal intuito desta ferramenta é coletar de forma sistêmica e organizada o que deu certo e o que poderia ter sido feito de outras maneiras, dentre outras análises, para que possam ser empregados em iniciativas futuras num ambiente de aprendizado e melhoria contínua (também denominado por vezes como Kaizen)

6 ESTATÍSTICA PARA SOLUÇÕES EM TI

Prof Ismael de Araujo Silva

6.1 Entrega

1ª – semestre

1ª ENTREGA:

Organizar uma Distribuição de Frequências e um Gráfico de Colunas, associado a uma das variáveis de pesquisa envolvidas no projeto, trabalhando dados em defesa da importância do seu projeto.

- a) Apresentar uma distribuição de frequências, com frequências absolutas, relativas, acumuladas e acumuladas relativas.
- b) Determinar o valor da média aritmética para a variável de pesquisa escolhida.

2ª ENTREGA:

Organizar uma Distribuição de Frequências e um Gráfico de Colunas, associado a uma das variáveis de pesquisa envolvidas no projeto, trabalhando dados em defesa da importância do seu projeto.

- a) Apresentar uma distribuição de frequências, com frequências absolutas, relativas, acumuladas e acumuladas relativas.
- b) Determinar o valor da média aritmética para a variável de pesquisa escolhida.

2ª – semestre

1ª ENTREGA:

A partir de uma das variáveis de pesquisa envolvidas no projeto do grupo:

- (I) organizar uma **Distribuição de Frequências Sem Classes**;
- (II) calcular o valor da **Média Aritmética**;
- (III) calcular o valor da **Variância**;
- (IV) calcular o valor do **Desvio Padrão**.

2ª ENTREGA:

A partir de uma das variáveis de pesquisa envolvidas no projeto do grupo:

(I) organizar duas Distribuições de Frequências Com Classes;

(II) calcular o Coeficiente de Variação de cada uma delas (indicar o cálculo da média aritmética e do desvio padrão);

(III) verificar e justificar qual das duas distribuições apresenta dados mais regulares (menor nível de dispersão).

7 Design e Desenvolvimento de Bancos de Dados

Prof. Alexandre Barcelos, Prof. Luciano Inácio Melo, Prof. Milton Goya e Profa. Rita Rodrigues

7.1 Entrega

1ª – semestre

1ª ENTREGA:

1. Descrição do Projeto e Regras de Negócio

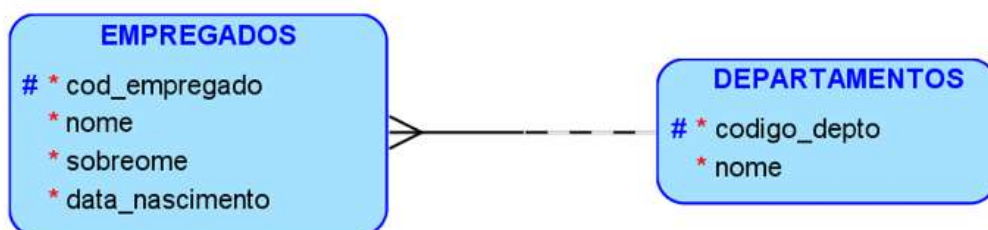
Descreva brevemente as regras de negócio. A descrição deve conter informações relevantes para a definição e compreensão dos requisitos necessários para o projeto de banco de dados.

2. Projeto Lógico do Banco de Dados

Entregar o modelo lógico do seu projeto de banco de dados. Lembrando que o modelo lógico é focado em negócio e contém alto grau de abstração. Mostre que você sabe o que é um modelo lógico de banco de dados. As regras são as seguintes:

- ENTREGAR O MODELO EM FORMATO PDF. Exporte o modelo para PDF ou IMAGEM
- Notação: Use obrigatoriamente Barker ou Engenharia de Informação. Nenhuma outra notação será aceita
- Ferramenta: Pode usar a que quiser, desde que esteja em uma das notações citadas acima

Exemplo



ENTREGAS QUE CONTENHAM OS ARQUIVOS DE PROJETO SERÃO REJEITADAS POR NÃO SEGUIR AS REGRAS .

2ª ENTREGA:

1. Entregar o projeto lógico do banco de dados revisado e:

- Normalizado com pelo menos a 3a Forma Normal.
- Use obrigatoriamente especialização e/ou associação (entidade associativa).
- **IMPORTANTE:** Use a notação de Barker ou Engenharia de Informação

2. Entregar o modelo relacional referente ao projeto lógico revisado. Este é o modelo que contém as tabelas que serão criadas no Banco de Dados.

Dicas:

- Revise o modelo relacional
- Garanta que os nomes das tabelas e dos atributos assim como das constraints estão ok.
- Que não haja inconsistência com o modelo lógico apresentado

3. Entregar o dicionário de dados do projeto criado.

O dicionário de dados de um projeto é parte importante da documentação. Ele contém as tabelas e colunas com suas respectivas descrições entre outras informações que julgar relevante para compreensão das mesmas.

Exemplo:

Tabela		EMPREGADOS		
Descrição		Tabela de funcionários. Relaciona-se com as tabelas departamentos e cargos.		
Coluna	Tipo de Dados	Tamanho	Constraint	Descrição
cod_empregado	NUMBER	5	PK	Identificador único do empregado
nome	VARCHAR	50	NN	Nome do empregado
sobrenome	VARCHAR	50	NN	Sobrenome do empregado
salario	NUMBER	5,2	NN,CK	Salário mensal do empregado. Deve ser maior do que

				zero. (constraint: emp_salario_min)
cod_cargo	NUMBER	5	NN, FK	Cargo atual do empregado. Chave estrangeira que referencia a coluna codigo_cargo da tabela cargos.
cod_departamento	NUMBER	5	FK	Código do departamento em que o funcionário é alocado. Chave estrangeira que referencia a coluna cod_departamento da tabela departamentos

Tabela	DEPARTAMENTOS			
Descrição	Tabela de funcionários. Relaciona-se com as tabelas departamentos e cargos.			
Coluna	Tipo de Dados	Tamanho	Constraint	Descrição
cod_departamento	NUMBER	5	PK	Identificador único do departamento
nome	VARCHAR	50	NN	Nome do departamento
cod_localizacao	VARCHAR	50	FK	Código do local do departamento.

				Chave estrangeira que referencia a coluna cod_localizacao da tabela localizacoes.
--	--	--	--	---

2ª – semestre

1ª ENTREGA:

1. **Entregar o Modelo Relacional do Banco de Dados:** Será a base para validação do script em SQL para criação do banco de dados (DDL). (5 pontos)

==> É a oportunidade de revisarem o projeto feito no primeiro semestre e arrumarem conforme feedbacks e comentários feitos em aula. Lembrem-se que é uma avaliação e, portanto, deve mostrar o que aprenderam. Não se espera um modelo com menos de 5 tabelas.

Regras:

1. Criar o modelo físico de dados baseando-se no modelo lógico de dados revisado, conforme feedback do 1o. semestre.
2. O modelo relacional deve apresentar além das restrições (CONSTRAINTS) chave primária (PRIMARY KEY), chave estrangeira (FOREIGN KEY) e obrigatória (NOT NULL), pelo menos uma restrição única/exclusiva UNIQUE e uma restrição de verificação/validação CHECK, de acordo com regras de negócio da solução proposta.
3. Todos os nomes devem estar padronizados, conforme nomenclatura trabalhada nas aulas: nomes de tabelas, nome de colunas e nome de restrições.
4. É obrigatório utilizar a ferramenta Oracle **Data Modeler** para construir o MER (modelo relacional). A não utilização dessas ferramentas irá acarretar desconto na nota.
5. É obrigatório utilizar uma das seguintes notações: Barker ou Engenharia da Informação.
6. O arquivo com o modelo DEVE ser entregue em PDF e seguir a seguinte padronização de nomenclatura:

{ANO_TURMA}_2021_MODELO_RELACIONAL_{RM}.PDF

Onde:

ANO_TURMA = ANO e TURMA onde o grupo está inserido.

RM: RM do aluno que fez a entrega no Portal da FIAP

Exemplo: 2SIR_2021_MODELO_RELACIONAL_RM12345.PDF

ATENÇÃO: SOMENTE UM ALUNO DO GRUPO DEVE POSTAR O ARQUIVO NO PORTAL DO ALUNO FIAP

2. SCRIPT DDL PARA IMPLEMENTAÇÃO DO BD: Aqui você deve apresentar o script em linguagem SQL com os comandos DDL para criação do banco de dados proposto pelo projeto apresentado no item 1 acima.

Você pode gerar o código por meio da ferramenta Oracle SQL Developer, incluindo os comandos DROP TABLE com opção CASCADE CONSTRAINTS.

REVISE O SCRIPT e garanta que ele esteja apenas com o que é necessário para a criação do Banco de Dados.

Requisitos obrigatórios do script:

1. O arquivo com o script DDL, deve ser criado na seguinte ordem:
 - DROP TABLE, com a opção CASCADE CONSTRAINTS no início do arquivo **em comentários de bloco**.
 - CREATE TABLE, contendo colunas, tipo de dado (DATATYPE), tamanho e as restrições NOT NULL, UNIQUE, PRIMARY KEY E CHECK , conforme nós fizemos em nossas aulas.
 - No final do arquivo, após os comandos CREATE TABLE, deverão ser criadas as chaves estrangeiras (FOREIGN KEY) com o comando ALTER TABLE
2. Todos os nomes devem estar padronizados, conforme as boas práticas trabalhadas nas aulas: nomes de tabelas, nome de colunas e restrições e devem estar de acordo (Tabelas, PKs, FKs e UKs) com os apresentados no projeto do banco de dados.
3. O arquivo DEVE ser gravado no formato **.SQL**.
4. Inserir no início do arquivo, em **forma de comentário**, o nome e RM de cada componente do grupo (**OBRIGATÓRIO**).
5. O arquivo entregue deve ser texto com extensão SQL, seguindo obrigatoriamente o seguinte padrão:

{ANO_TURMA}_2021_MODELO_RELACIONAL_{RM}.SQL

Onde:

ANO_TURMA = ANO e TURMA onde o grupo está inserido.

RM: RM do aluno que fez a entrega no Portal da FIAP

Exemplo: 2SIR_2021_MODELO_RELACIONAL_RM12345.SQL

ATENÇÃO: SOMENTE UM ALUNO DO GRUPO DEVE POSTAR O ARQUIVO NO PORTAL DO ALUNO FIAP

ATENÇÃO: O script deve executar sem nenhum erro e criar o banco de dados exatamente como proposto pelo projeto. Erros de execução do script quer dizer que o banco de dados não pode ser criado e este item 2 (que vale 5 pontos) será zerado.

==> A avaliação será feita levando em conta a qualidade do projeto de banco de dados apresentado, do script SQL e do cumprimento de todas as regras estabelecidas acima.

2ª ENTREGA:

SCRIPT DML E DQL (SELECT) PARA POPULAR E CONSULTAR AS TABELAS PARA OS TESTES DA APLICAÇÃO – VALOR: 10,0 (DEZ) PONTOS

1. Arquivo com as instruções **DMLs** referente a carga de dados para testes.
2. Cada tabela deve ser preenchida com no mínimo 10 linhas. As tabelas associativas, devem ser preenchidas com no mínimo 20 linhas.
3. A massa de dados, deve ser composta por dados válidos, ou seja, não devem ser inseridos: xxxx,11111,teste, ou similar. Trabalhar com dados fictícios, mas coerentes.
4. Em seguida, apresentar comandos DQLs (SELECTS) para selecionar os dados de acordo com as orientações a seguir:

(a) UMA SELEÇÃO UTILIZANDO JOIN DE DUAS TABELAS.

(b) UMA SELEÇÃO UTILIZANDO AGRUPAMENTO (GROUP BY)

(c) UMA SELEÇÃO UTILIZANDO JOIN DE NO MÍNIMO 3 TABELAS, COM WHERE E ORDENANDO O RESULTADO POR NO MÍNIMO DUAS COLUNAS.

(d) UMA SELEÇÃO UTILIZANDO JOIN, WHERE, GROUP BY E HAVING, E QUE EXIBA OS RESULTADOS EM ORDEM DECRESCENTE POR ALGUMA DAS COLUNAS

1. CADA SELECTE APRESENTADO ACIMA DEVE TER UMA DESCRIÇÃO DIZENDO QUAL A FINALIDADE DO COMANDOS.

Exemplo: Gerar um relatório que mostre e que traga todos agrupados por ... em ordem ...

1. O arquivo DEVE ser gravado no formato **.SQL**
2. Inserir no início do arquivo, em **forma de comentário**, o nome e RM de cada componente do grupo (**OBRIGATÓRIO**).
3. **O arquivo DEVE seguir a seguinte padronização de nomenclatura:**

{ANO_TURMA}_2021_SCRIPT_DML_DQL{RM}.SQL

Onde:

ANO_TURMA = ANO e TURMA em onde o grupo está inserido.

RM = RM do aluno que fez a entrega no portal da FIAP

Exemplo: 2ECR_2021_SCRIPT_DML_DQL_RM12345.SQL

ATENÇÃO: SOMENTE UM ALUNO DO GRUPO DEVE POSTAR O ARQUIVO NO PORTAL DO ALUNO FIAP

ATENÇÃO: Os comandos apresentados no script devem ser executados sem erros. Qualquer erro na execução do comando resultará na perda de pontos. Se o erro impactar no resultado de comandos posteriores, pontos também serão descontados.

A simples execução do comando sem erros de sintaxe não garante que a questão esteja correta. Por exemplo, se algum dado que deveria ser retornado por um relatório não o for, será considerado completamente errado.

Todas as regras devem ser cumpridas impreterivelmente. O não cumprimento de qualquer uma delas acarretará no desconto de pontos.

Sugestões de sites para criação da massa de dados online (podem ser utilizados outros sites similares):

<http://www.generatedata.com>

<http://mockaroo.com>

<http://freedatagenerator.com/sql-data-generator>

<http://databasetestdata.com>

8 Programação Orientada a Objetos

Prof. Antonio Marcos Selmini e Prof. Emerson Rodolfo Abraham

8.1 Entrega

1ª – semestre

1ª ENTREGA:

O grupo deverá enviar (documento de texto ou pdf):

1. a descrição do problema que abordará. A descrição deve ser detalhada para que a avaliação seja coerente.
2. identificar os objetos envolvidos na aplicação, ou seja, aqueles que serão utilizados na resolução do problema.
3. identificar as classes e seus atributos. Faça a descrição das classes como feita em sala de aula (diagrama de classe): o retângulo dividido em três partes.

Não esqueçam de colocar o nome e o RM de todos os integrantes no documento que será enviado para avaliação.

2ª ENTREGA:.

O grupo deverá fazer uma nova análise na problemática abordada e identificar:

1. há aplicação de herança? Se tiver, identificar onde. Caso não tenha, apenas informar que não haverá.
2. há aplicação de polimorfismo? Se tiver, identificar onde. Caso não tenha, apenas informar que não haverá.
3. enviar o documento com as novas alterações nas classes.
4. projeto em java contendo as classes implementadas de acordo com o documento no item 3.

2ª – semestre

1ª ENTREGA:

Nesta etapa a sua equipe deverá trabalhar com a linguagem Java e acesso a banco de dados. A equipe deverá:

- Criar um projeto Java.
- Escolher pelo menos três tabelas da aplicação (aquela que será entregue para a disciplina de banco de dados) para implementar.
- Implementar as classes (entidades da aplicação).
- Implementar as classes DAO para realizar as operações de banco de dados (inserir, pesquisar e alterar).

A equipe deverá entregar para avaliação:

1. Projeto java.
2. Script de banco de dados.
3. Modelo de banco de dados (o mesmo que será entregue para a disciplina de banco de dados)

2ª ENTREGA:.

Nesta etapa a sua equipe deverá trabalhar com a linguagem Java e acesso a banco de dados. A equipe deverá:

- Criar um projeto Java.
- Escolher pelo menos três tabelas da aplicação (aquela que será entregue para a disciplina de banco de dados) para implementar.
- Implementar as classes (entidades da aplicação).
- Implementar as classes DAO para realizar as operações de banco de dados (inserir, pesquisar e alterar).

A equipe deverá entregar para avaliação:

1. Projeto java.
2. Script de banco de dados.
3. Modelo de banco de dados (o mesmo que será entregue para a disciplina de banco de dados)

9 Vídeo de apresentação do protótipo funcional

9.1 Entrega

Um vídeo pitch com exibição do protótipo funcional, de no máximo 3 minutos sobre seu projeto, deve ser realizado e entregue para os professores, use e abuse de sua criatividade. Escolha o colega que tenha mais desenvoltura que gravará o vídeo pitch.

Para ajudá-los a fazer um vídeo mais interessante, aqui estão algumas dicas do Time Estúdio Fiap:

1. Antes do REC: lembre-se da configuração!

Talvez você não saiba, mas antes de qualquer gravação, é necessário configurar os equipamentos. E com seu smartphone ou computador não é diferente!

A primeira configuração é o ajuste da resolução:

IDEAL: 1920x1080 (qualidade full HD)

ESTÁ VALENDO: 1280x720 (qualidade HD)

A segunda recomendação é esquecer o mundo das selfies. Como padrão, a gravação deve ser 16:9, por isso, grave com o seu celular na horizontal.

IMPORTANTE: Use a câmera traseira do seu smartphone, a resolução dela é melhor, combinado?

A DICA DE OURO: Lembre-se de, quando for gravar, manter o celular no modo avião e com o modo “Não Perturbe” ativado. Assim, nenhuma ligação indesejada ou alarme vão interromper a produção do seu vídeo.

2. Capriche no cenário!

O background do seu vídeo pode fazer toda a diferença no resultado final! Por isso, escolha um cenário com elementos que conversem com o tema do seu vídeo. Na dúvida, vá para o local seguro: um escritório com livros e plantas, que são sempre coringas.

EVITE: Portas, ambientes domésticos e janelas.

NUNCA: Jamais se posicione com fontes de luz atrás de você.

ESTÁ VALENDO: Caso o ambiente não tenha elementos, tudo bem! Uma parede colorida já garante o profissionalismo necessário.

3. Atenção ao enquadramento! Se está bonito, enquadre!

Definido o cenário, é hora de escolher o enquadramento. Lembre-se de que enquadramento é tudo aquilo que será visto pelas pessoas! Pense sobre o que é importante que apareça na tela!

Posicione-se no centro da telinha, da cintura para cima e deixe uma margem de segurança sobre a sua cabeça.

4. Estabilize o seu celular durante a gravação!

Sabemos que vocês têm superpoderes, mas até heróis e heroínas precisam de um apoio! O objetivo aqui é evitar que você acabe tremendo durante a gravação! Se não tiver um tripé à disposição, apoie o seu celular em uma superfície fixa! É sucesso garantido!

EVITE: Apoiar seu aparelho em um lugar que possa tombar durante a filmagem.

NUNCA: Não grave com o celular em pé (na vertical), combinado?

ESTÁ VALENDO: Olhe para a lente do seu smartphone enquanto estiver gravando.

5. Vá para a luz!

A luz é um dos elementos mais importantes do vídeo. Por isso, escolha um ambiente bem iluminado, lembrando-se de se posicionar em frente às fontes de luz, sejam elas janelas ou luzes artificiais.

NUNCA: Jamais grave de costas para uma janela.

ESTÁ VALENDO: Se o ambiente estiver um pouco escuro, vale recorrer a abajures e luminárias para preencher a luz.

6. Cuidado com o áudio!

Encontre um lugar silencioso para a sua gravação e fique atento à acústica do ambiente! Locais muito grandes e com o pé direito alto acabam gerando ECO. Assim como o carro dos ovos ou do gás podem vazar no vídeo.

DICA: Na busca pelo ambiente perfeito para a sua gravação, bata uma palma e preste atenção ao som! Se a palma ecoar, ou seja, se o som se repetir algumas vezes, é sinal de que o mesmo poderá acontecer com a sua voz! Nesse caso, procure outro ambiente.

7. Crie um roteiro!

Parece coisa de cinema, mas o script é fundamental para que você consiga abordar todos os pontos propostos durante a sua gravação. Organize seu conteúdo em bullet points para ensaiar o pitch (mas não coloque os textos nos slides) e arrase!

EVITE: O improviso leva ao esquecimento. Anote tudo que lembrar antes da filmagem.

NUNCA: Vale o roteiro, não vale ler sua “colinha”. Pense nele nela como um material de apoio.

ESTÁ VALENDO: Colar post-its com termos-chave e mencione-os ao longo do vídeo.

8. Com que roupa?

O figurino também é parte importante do vídeo.

EVITE: Estampas que possam ficar estranhas com sua movimentação.

JAMAIS: O black-tie está dispensado, mas nada de pijamas. Um tapa no penteado também é bem-vindo :D

9. Você está no comando!

Lembre-se de que, assim como em uma aula, você é o protagonista, não o seu PPT!

O apoio visual é sempre bem-vindo, mas você deve ser a estrela principal do vídeo. Busque o equilíbrio entre as suas aparições na telinha e o PPT.

EVITE: Deixar sua didática apoiada no PPT.

NUNCA: Nada de ficar apenas lendo ou só deixar mostrar a tela, você está no comando.

ESTÁ VALENDO: O dinamismo entre você e o conteúdo sempre vale. :D

10. Postura, respiração e olho na câmera

As pessoas poderão ver a movimentação dos seus olhos e pescoço. Por isso, evite ler grandes trechos durante a sua gravação! Seja natural e tente esquecer a presença da câmera.

EVITE: Tentar soar sério ou descontraído demais. Dê o tom certo de acordo com o conteúdo e o público.

NUNCA: Faça um sermão lendo, passando muito tempo sem olhar para a câmera.

ESTÁ VALENDO: Ficou nervoso? Pense em alguma fala para descontrair.

Ufa! Depois de tudo pronto, é só salvar um arquivo .doc com o link do seu vídeo no Youtube e postar na área de trabalhos do site da Fiap. Lembre-se de deixar esse link público para que possamos assistir. Boa sorte 😊

QUANDO SOLICITADO, faça os envios de vídeos via área de entrega de trabalhos (envie apenas o link do vídeo publicado na WEB em um canal como o Youtube) definida pelo Scrum Master:

-Para todas as turmas do 2º ano de SI, o Scrum Master é o professor Renato Jardim Parducci
profrenato.parducci@fiap.com.br

9.2 Professores responsáveis

Scrum Master é o professor Renato Parducci para as turmas 2SIA, 2SIF, 2SIG, 2SIH, 2SIR e o professor Junior Amaral para as turmas 2SIB e 2 SIS.

Foco do trabalho do Scrum Master:

- Trata comunicação com professores e parceiro quando a montagem de agendas de eventos e avaliações
- Auxilia em caso de impedimento ou dificuldade na comunicação direta do time de projeto com algum professor ou representantes do parceiro

Demais professores

- Combinam e avaliam as entregas específicas das suas disciplinas

LEMBRE-SE QUE O PROJETO É GERENCIADO E DE TOTAL RESPONSABILIDADE DAS EQUIPES! Os professores são facilitadores e o parceiro é o responsável por avaliar as propostas de soluções.