

 <p>INSTITUTO FEDERAL NORTE DE MINAS GERAIS Campus Montes Claros</p>	INSTITUTO FEDERAL DO NORTE DE MINAS GERAIS	
	Aluno(a):	
	Curso: Bacharelado em Ciência da Computação	
	Ano: 2023	
	Professor: Lucio F Dutra Santos	
	Disciplina: Banco de Dados	
	Data: 05/06/2023	Valor: 12 pontos

Projeto Parte 3 – Implementação do Projeto de BD e sistema de acesso

Entrega: 30/06 até às 11:00 hs.

1) Condições:

- Sistema de Jogos online similar ao *Steam* desenvolvido durante o semestre Sistema
- Ferramentas a serem utilizadas:
 - SGBD: Oracle como usado na parte final do curso.
 - Linguagem de programação: escolha livre; sugere-se Java.
- **NÃO DEVEM** ser utilizados componentes que implicitamente gerem comandos *SQL*; ou seja, todas as operações no SGBD devem ser feitas por comandos *SQL* explícitos.

2) Requisitos

- Efetuar as correções indicadas na segunda parte do projeto;
- Implementar um protótipo para a manipulação do banco de dados;
- Criar uma interface gráfica intuitiva de uso.

→ Só serão considerados projetos completos, até a definição do protótipo, e satisfazendo aos requisitos e às condições descritas.

3) Funcionalidades do projeto

- (2 pontos) Escolha uma relação contendo atributos de pelo menos dois tipos de dados distintos (por exemplo: número e texto; texto e data; ou data e número) e pelo menos uma chave estrangeira para manipulação de dados via **interface gráfica intuitiva**. Seu protótipo deverá gerenciar (inserir, atualizar, listar, e remover) as tuplas da relação escolhida.
 - ⇒ O cadastro de valores de cada chave estrangeira deve ser intuitivo. Para tanto, deve-se apresentar ao usuário uma lista de valores válidos e permitir que o usuário selecione o valor desejado sem precisar digitá-lo. Essa lista deve conter todos os valores de chave primária da relação referenciada pela chave estrangeira, incluindo qualquer outro atributo da relação referenciada que possa facilitar a escolha do usuário.
 - ⇒ Caso se aplique, apresente também uma lista de valores válidos para qualquer atributo que tenha cláusula *CHECK(atributo IN ('valor 1','valor 2',...,'valor n'))*, ou seja, ilustre a situação em que você lidaria com questões de integridade por meio da aplicação.
- (2 pontos) Considerando uma Generalização/Especialização de seu modelo ER para manipulação de dados via **interface gráfica intuitiva**, tal que esta: i) **não** tenha sido mapeada no Modelo Relacional como uma única relação; ii) contenha tanto atributos genéricos quanto atributos específicos, e; iii) alguma de suas relações tenha pelo menos uma chave estrangeira. Seu protótipo deverá gerenciar (inserir, atualizar, listar, e remover) as entidades da Generalização/Especialização escolhida, inserindo/atualizando/excluindo/consultando em conjunto tanto atributos genéricos como específicos.

- ⇒ Deve ser utilizada alguma estratégia na aplicação para garantir que cada inserção/atualização/remoção seja realizada em todas as relações envolvidas, e nunca em apenas parte delas.
 - ⇒ O cadastro de valores de cada chave estrangeira também deve ser intuitivo, assim como descrito previamente no **Item 1**.
 - ⇒ Caso se aplique, apresente também uma lista de valores válidos para qualquer atributo que tenha cláusula *CHECK(atributo IN ('valor 1','valor 2',... 'valor n'))*.
3. (3 pontos) Apresente em **interface gráfica intuitiva** um relatório com os dados da seguinte consulta: “Recupere as contas de usuários na base de dados que tenham conquistados os mesmos troféus que uma dada conta informada e que pelo menos um amigo tenha comprado ou instalado ou jogado o mesmo jogo.”. A consulta SQL correspondente deve ser escrita explicitamente.

4) Cada representante de grupo deve submeter na tarefa relacionada ao trabalho na sala de aula (Google Classroom), em um único arquivo zip, os seguintes artefatos:

- (3 pontos) Os *scripts* para criação das tabelas no banco de dados, incluindo todas as definições de *PRIMARY KEY*, *UNIQUE*, *FOREIGN KEY* (com e sem *ON DELETE*), *CHECK*, *NOT NULL* e *DEFAULT* que forem relevantes para o sistema;
 - O *script* entregue deve definir todas as relações de seu modelo, e não apenas as que foram utilizadas na interface gráfica;
 - O *script* entregue deve contemplar todas as correções na segunda parte do trabalho. As correções não atendidas terão novo desconto em nota;
- (2 pontos) Os *scripts* de inserção de dados usados para os testes com no mínimo 10 tuplas por relação;
 - O *script* entregue deve incluir dados em todas as relações de seu modelo, e não apenas as que foram utilizadas na interface gráfica;
 - Os dados inseridos devem ter sentido semântico (não utilizem geradores do tipo *Lorem Ipsum*);
- Arquivos fonte com o código do sistema e o executável; incluir os *drivers* utilizados;
- Relatório com uma breve documentação externa:
 - Lista dos softwares, ferramentas, linguagens e APIs utilizadas, incluindo as versões;
 - Manual de consideração sobre a execução do sistema “*how-to*”.
 - Breves comentários, discussões e explicações necessárias para a correção, incluindo comentários sobre as funcionalidades extras, caso existam;
 - Não incluir código na documentação externa.
- As apresentações aconteceram no dia 30/06 no horário da aula.