



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina Banco de Dados**

**AP1 1º semestre de 2018**

**Nome –**

**Assinatura –**

---

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
  2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
  3. Você pode usar lápis para responder as questões.
  4. Ao final da prova devolva as folhas de questões e as de respostas.
  5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
- 

**Questão 1 [4,0 pontos]**

Deseja-se projetar uma base de dados que dará suporte a um sistema WEB para controlar as horas trabalhadas pelos profissionais de uma empresa de desenvolvimento de software. O sistema destina-se a coletar dados para cobrança dos clientes da empresa. Através de um diagrama entidade-relacionamento, deve ser modelada esta base de dados. A base de dados não deve conter redundância de dados. O modelo ER deve ser representado com a notação vista em aula ou com outra notação de poder de expressão equivalente. O modelo deve apresentar, ao menos, entidades, relacionamentos, atributos, especializações, identificadores e restrições de cardinalidade. Não usar atributos multivalorados. O modelo deve ser feito no nível conceitual, sem incluir chaves estrangeiras.

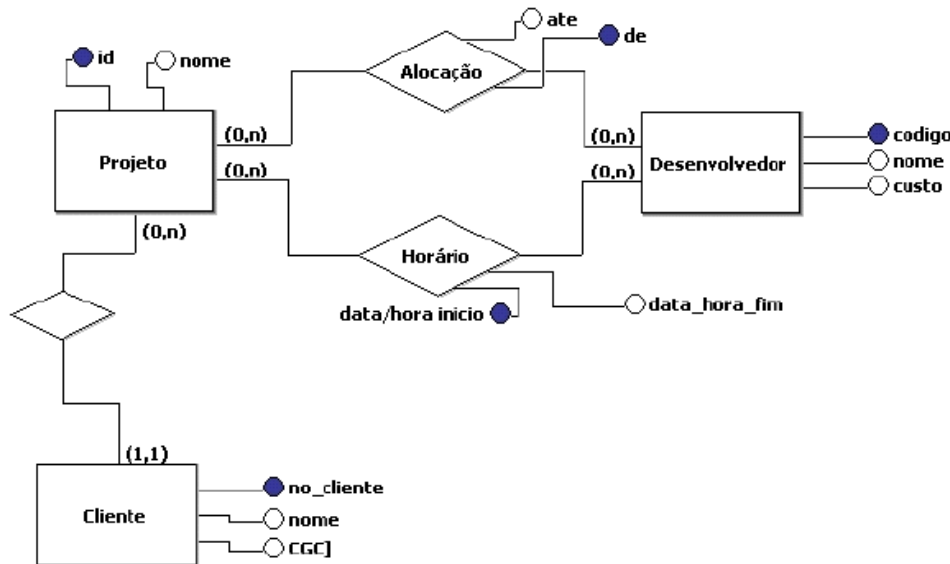
Todas atividades da empresa acontecem através de projetos. Para cada projeto, o banco de dados deve armazenar um identificador, o nome do projeto e o seu cliente. Um cliente pode ter vários projetos. Além dos projetos do cliente, o banco de dados deve armazenar o número (único) do cliente, seu CGC e seu nome.

Em cada projeto são alocados vários desenvolvedores. Um desenvolvedor é alocado a um projeto por um tempo determinado (de-até). Cada desenvolvedor tem um código identificador, um nome e um custo por hora trabalhada.

Para cada vez que um desenvolvedor trabalha em um projeto, mesmo que por alguns minutos, o banco de dados deve armazenar, além do projeto e do desenvolvedor, a

data/hora em que o desenvolvedor começou a trabalhar e a data/hora em que ele encerrou o trabalho. É preciso manter o histórico dos projetos em que um determinado empregado trabalhou.

**Resposta:**



## **Questão 2 (2 pontos)**

Que recursos um SGBD possui para:

- (a) definir uma base de dados; e (b) manipular uma base de dados.
- (b) Para definir uma base de dados o SGBD permite criar as relações que fazem parte da base de dados por meio de comandos voltados a cada nível de representação dos dados, por exemplo o lógico e o físico, como definição de atributos e chaves e índices ou representação física de atributos.
- (c) Uma das características principais de um SGBD é a execução de consultas. Para obter resultados de consultas é necessário manipular as estruturas de representação da base de dados. Para manipular uma base de dados o SGBD possui funções de gerência de estruturas de dados adequadas para cada nível de representação dos dados. Para permitir que as definições de cada nível fiquem relacionadas e consistentes, o SGBD possui funções de mapeamento entre esquemas.

## **Questão 3 [3 pontos]**

Considere a seguinte base de dados, usada por uma sapataria. O cliente entrega um sapato para conserto e o retira algum tempo depois de o serviço ser prestado. As chaves primárias estão sublinhadas.

```

/* tabela de clientes cadastrados na sapataria */
CLIENTE (cpf, nomeCli, celular)
  
```

```

/* tabela com dados dos sapatos dos clientes da sapataria – ao chegar na sapataria, cada sapato
recebe um código de barras único para facilitar a sua identificação*/
SAPATO (codBarra, numero, modelo, cor, cpf);
(cpf) referencia CLIENTE

/* tabela com os reparos feitos */
REPARO (codBarra, dataExecutada, dataUltimoConserto)
(codBarra) referencia SAPATO

/* tabela com os materiais usados em cada reparo */
MATERIAL_REPARO (codBarra, dataExecutada, codMat, quantidade)
(codBarra, dataExecutada) referencia REPARO
(codMat) referencia MATERIAL

/* tabela com as descrições dos materiais */
MATERIAL (codMat, descricao)

```

Sobre esta base de dados, resolver as consultas a seguir usando álgebra relacional. Não usar mais tabelas do que o estritamente necessário.

(a) Faça uma consulta que retorna os modelos dos sapatos do cliente chamado “João Moreira” [1 ponto].

$\pi_{\text{modelo}} (\text{Sapato} \bowtie (\sigma_{\text{nomeCli} = \text{“João Moreira”}} \text{Cliente}))$

(b) Faça uma consulta que retorna a descrição do material utilizado no reparo realizado no dia 01/03/2014 do sapato cujo código de barras é 000111 [1 ponto].

$\pi_{\text{descricao}} (\text{Material} \bowtie (\text{MaterialReparo} \bowtie (\sigma_{\text{dataExecutada} = \text{“01/03/2014”}} \text{Reparo} \bowtie (\sigma_{\text{codBarra} = \text{“000111”}} \text{Sapato})))$

(c) Faça uma consulta que mostre a descrição dos materiais que já foram utilizados em algum reparo [1 ponto].

$\pi_{\text{descricao}} (\text{Material} \bowtie (\text{MaterialReparo}))$

Note que aqui a tabela Reparo não é necessária.

#### **Questão 4 [1 ponto]**

Utilizando o esquema da questão 3, analise as consultas a seguir e diga qual o esquema da tabela retornada por cada uma das consultas. O esquema deve ser informado utilizando a seguinte sintaxe:

$\text{Tab} (\text{TabOrigem}_1.\text{Atrib}_1, \text{TabOrigem}_1.\text{Atrib}_2, \dots, \text{TabOrigem}_N.\text{Atrib}_1, \text{TabOrigem}_N.\text{Atrib}_2, \dots)$

Nesta notação,  $\text{tabOrigem}_i$  é o nome da tabela de onde veio o atributo  $\text{Atrib}_j$  originalmente.

(a)  $\sigma_{\text{nomeCli}=\text{"Vanessa"}}(\text{Cliente})$

$\text{Tab} (\text{Cliente.cpf}, \text{Cliente.nomeCli}, \text{Cliente.celular})$

(b)  $(\pi_{\text{nomeCli}, \text{cpf}}(\text{Cliente})) * \text{Sapato} * (\sigma_{\text{dataUltimoConserto} < \text{"10/03/2014"}} \text{Reparo})$

$\text{Tab} (\text{Cliente.nomeCli}, \text{Cliente.cpf}, \text{Sapato.codBarra}, \text{Sapato.numero}, \text{Sapato.modelo}, \text{Sapato.cor}, \text{Reparo.dataExecutada}, \text{Reparo.dataUltimoConserto})$