

Relatório - Trabalho Final

Princípios de BD para Sistemas de Informação

Alunos: João Vitor Marins da Silva
Larissa Galvão Nascimento
Thalys Pacheco Passeri
Thiago Pereira Araújo

Requisitos

A base de dados deve incluir:

- **motoristas**, identificados por seus **cpfs, rg, endereço, data de nascimento, número da cnh e conta corrente para pagamento**,

bem como:

- **passageiros**, identificados por seus **cpfs, rg, endereço, data de nascimento e número do cartão de crédito**.

Além deles, deve-se registrar os:

- **atendentes**, que são identificados por seus **cpfs, rg, endereço, data de nascimento e formação escolar**.

- **Motoristas e atendentes podem usar os serviços de transporte como passageiros.**

- **Um atendente pode ser um motorista nas horas vagas.**

Um ou mais **veículos** são **associados** a um **motorista**, sendo que cada:

- **veículo** deve ser identificado pelo seu **número de RENAVAM, data de compra, marca, modelo, ano, preço e característica que deve ser um valor dentre {econômico, SUV, luxo, rural}**.
- Uma **seguradora** é identificada por um **CNPJ, nome, telefone e endereço postal**.
- Uma seguradora pode ser responsável por um ou mais **seguros** de veículos, que são identificados por **número e valor da apólice**. O sistema não armazena o histórico de apólices dos veículos, logo, **no máximo, apenas um único número de apólice por veículo aparece no sistema por vez**.
- Uma **corrida** é registrada como o serviço de um **motorista para vários passageiros**, sendo identificada pelo **número da corrida, endereço de destino, data início, data fim, hora início, hora fim, valor total da corrida e preço da corrida por passageiro**.

Um **atendente** é vinculado ao suporte de cada corrida, sendo que os **passageiros** podem enviar:

- **questionamentos**, identificados por um **número sequencial**, **texto da pergunta** e **texto da resposta**, ao atendente durante a corrida.
- **O sistema armazena o histórico de corridas.**

1. - Modelo Entidade Relacionamento (MER)

1.1 - Entidades

- Criação das entidades: Motoristas, Passageiros, Atendente, Veículo e Seguradora
- Para simplificar a modelagem, optamos por uma generalização/especialização disjunta em três entidades: Motorista, Passageiro e Atendente na nova entidade genérica “Usuário”, onde a participação de usuário é total para uma, e apenas uma, das entidades.
- Entidade genérica Usuário possui como atributos, CPF, RG, ENDEREÇO e DATA_NASC, onde “CPF” é a chave primária.
- A entidade “Motorista” possui como atributos, CNH e CONTA
- A entidade “Passageiro” possui como atributo apenas CART_CRED
- A entidade “ATENDENTE” possui como atributo apenas FORM_ESCOLAR
- A entidade “VEÍCULO” possui como atributos, RENAVAM, DATA_COMPRA, MARCA, MODELO, ANO, PRECO e CARACTERISTICA, onde “RENAVAM” é a chave primária.
- A entidade “SEGURADORA” possui como atributos, CNPJ, NOME, TELEFONE e ENDE_POSTAL, onde “CNPJ” é a chave primária

1.2 - Relacionamentos

- ASSOCIADO: Motorista possui um relacionamento de nome “ASSOCIADO” com Veículo, onde apenas um Motorista pode ser associado a vários Veículos (Cardinalidade 1:N)

- ASSEGURA: Veículo possui um relacionamento de nome “ASSEGURA” com Seguradora, onde vários Veículos podem ser assegurados por uma Seguradora (Cardinalidade 1:N).
Também existem os atributos de relacionamento, NUMERO_APOLICE e VALOR_APOLICE.
- CORRIDA: Motorista possui um relacionamento de nome “CORRIDA” com Passageiro, onde vários passageiros podem pegar uma corrida com um motorista (Cardinalidade 1:N)
- Agregação “HISTÓRICO DE CORRIDAS”: Com o intuito de armazenar informações sobre as corridas realizadas, foi criada a agregação do relacionamento CORRIDA, que possui como atributos: NUM_CORRIDA, CPF_PASSAGEIRO_RESPONSAVEL, CPF_PASSAGEIRO_CARONANTE, ENDE_DESTINO, DATA_INICIO, DATA_FIM, HORA_INICIO, HORA_FIM, VALOR_TOTAL, PRECO_CORRIDA, onde NUM_CORRIDA é chave primária e PRECO_CORRIDA é atributo derivado.
- SUPORTE: Atendente possui um relacionamento de nome “SUPORTE” com a agregação “HISTÓRICO DE CORRIDAS”, onde um Atendente dá suporte a várias corridas (Cardinalidade 1:N).
- Agregação ‘CONVERSA’: Como suporte só tem atributo número sequencial a relação só poderia identificar um texto_pergunta para um texto_resposta. Então foi criada uma agregação herdando o número da corrida da agregação histórico e CPF_ATENDENTE para que uma corrida pudesse identificar uma conversa, onde NUMERO_SEQ e NUM_CORRIDA correspondem a chave primária composta, também todos os atributos(NUMERO_SEQ, TXT_PERGUNTA, TXT_RESPOSTA) do relacionamento suporte foram passados para a agregação conversa.

Normalização

- USUARIO está na 3FN porque não contém atributos multivalorados, não tem chave composta para ter dependências de atributos a parte da chave primária e não existe dependência de atributos não chave para atributos não chave.
- MOTORISTA está na 3FN porque não contém atributos multivalorados, não tem chave composta para ter dependências de atributos a parte da chave primária e não existe dependência de atributos não chave para atributos não chave.
- PASSAGEIRO está na 3FN porque não contém atributos multivalorados, não tem chave composta para ter dependências de atributos a parte da chave

primária e existe apenas um atributo não chave, não existindo dependência de atributos não chave para atributos não chave.

- ATENDENTE está na 3FN porque não contém atributos multivalorados, não tem chave composta para ter dependências de atributos a parte da chave primária e existe apenas um atributo não chave, não existindo dependência de atributos não chave para atributos não chave.
- VEICULO está na 3FN porque não contém atributos multivalorados, não tem chaves composta para ter dependências de atributos a parte da chave primária e não existe dependência de atributos não chave para atributos não chave.
- SEGURADORA está na 3FN porque não contém atributos multivalorados, não tem chave composta para ter dependências de atributos a parte da chave primária e não existe dependência de atributos não chave para atributos não chave.
- ASSEGURA está na 3FN porque não contém atributos multivalorados, não tem chave composta para ter dependências de atributos a parte da chave primária e não existe dependência de atributos não chave para atributos não chave.
- HISTORICO_CORRIDAS: Não está na 2FN porque os atributos: CPF_MOTORISTA, ENDE_DESTINO, DATA_INICIO, DATA_FIM, HORA_INICIO, HORA_FIM, VALOR_TOTAL; dependem apenas de parte da chave primária, NUM_CORRIDA. E para atingir a 2FN deixamos a tabela HISTORICO_CORRIDAS apenas com NUM_CORRIDA E CPF_PASSAGEIRO_RESPONSÁVEL como chaves primárias e o restante de seus atributos CPF_MOTORISTA, ENDE_DESTINO, DATA_INICIO, DATA_FIM, HORA_INICIO, HORA_FIM, VALOR_TOTAL. E por fim criamos uma nova tabela CARONANTES_CORRIDA com NUM_CORRIDA, CPF_PASSAGEIRO_RESPONSÁVEL como uma chave primária composta, e composta com CPF_PASSAGEIRO_CARONANTE, NUM_CORRIDA, CPF_PASSAGEIRO_RESPONSÁVEL referenciando HISTORICO_CORRIDAS e CPF_PASSAGEIRO_CARONANTE referenciando PASSAGEIRO. E a pela dependência de CPF_PASSAGEIRO_CARONANTE para NUM_CORRIDA, CPF_PASSAGEIRO_RESPONSÁVEL, é criado uma tabela para a relação dos caronantes com um responsável em uma corrida, chamada CARONANTES_CORRIDA, onde NUM_CORRIDA, CPF_PASSAGEIRO_RESPONSÁVEL são uma chave composta, composta com CPF_PASSAGEIRO_CARONANTE.
- CONVERSA: Não está na 2FN, porque o atributo: CPF_ATENDENTE depende apenas de parte da chave primária, NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSÁVEL. E os atributos TXT_PERGUNTA, TXT_RESPOSTA dependem de toda a chave, NUM_CORRIDA, CPF_PASSAGEIRO_RESPONSÁVEL e NUMERO_SEQ. Então para a passagem para a 2FN é criado uma tabela ATENDENTE_CORRIDA contendo NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSÁVEL como chaves primárias referenciando HISTORICOS_CORRIDAS e CPF_ATENDENTE referenciando ATENDENTE. Então a tabela CONVERSA fica com NUM_CORRIDA,

CPF_PASSAGEIRO_RESPONSAVEL e NUMERO_SEQ como chaves primárias, onde NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL referencia ATENDENTE_CORRIDA, e com os atributos TXT_PERGUNTA, TXT_RESPOSTA). Agora as tabelas já estão automaticamente na 3FN.

Scripts

Schema

Tabela MOTORISTA:

A tabela motorista tem as restrições de integridade:

```
PRIMARY KEY(CPF_MOTORISTA),  
CONSTRAINT MOTORISTA_ibfk_1 FOREIGN KEY (CPF_MOTORISTA)  
REFERENCES USUARIO (CPF) ON UPDATE CASCADE
```

Pois, motorista usa chave de USUARIO, CPF, como chave estrangeira. E ao ser alterado em usuário deve ser alterado em motorista também, não deve ser apagado, se apagado de usuário, porque é de nossa escolha que os motoristas que fizeram corridas continuem no histórico de corridas.

Tabela PASSAGEIRO:

A tabela PASSAGEIRO possui as seguintes restrições de integridade:

- A coluna CPF_PASSAGEIRO é definida como PRIMARY KEY.
- A coluna CPF_PASSAGEIRO possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna CPF da tabela USUARIO, com a opção ON UPDATE CASCADE. Isso significa que, se o valor do CPF for atualizado na tabela USUARIO, o valor correspondente em CPF_PASSAGEIRO também será atualizado.
-

Tabela ATENDENTE:

A tabela ATENDENTE possui as seguintes restrições de integridade:

- A coluna CPF_ATENDENTE é definida como PRIMARY KEY.
- A coluna CPF_ATENDENTE possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna CPF da tabela USUARIO, com a opção ON UPDATE CASCADE. Isso significa que, se o valor do CPF for atualizado na tabela USUARIO, o valor correspondente em CPF_ATENDENTE também será atualizado

Tabela VEICULO:

A tabela VEICULO possui as seguintes restrições de integridade:

- A coluna RENAVAL é definida como PRIMARY KEY.
- A coluna CPF_MOTORISTA possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna CPF_MOTORISTA da tabela MOTORISTA, com a opção ON DELETE CASCADE e ON UPDATE CASCADE. Isso significa que, se um registro relacionado em MOTORISTA for excluído ou seu CPF_MOTORISTA for atualizado, os registros correspondentes em VEICULO também serão excluídos ou atualizados.

- A coluna CARACTERISTICA possui uma restrição de verificação (CHECK) que permite apenas os valores 'econômico', 'SUV', 'luxo' ou 'rural'.

Tabela SEGURADORA:

A tabela SEGURADORA possui a coluna CNPJ definida como PRIMARY KEY.

Tabela ASSEGURA:

A tabela ASSEGURA possui as seguintes restrições de integridade:

- A coluna RENAVAL é definida como PRIMARY KEY.
- A coluna RENAVAL possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna RENAVAL da tabela VEICULO, com a opção ON DELETE CASCADE e ON UPDATE CASCADE. Isso significa que, se um registro relacionado em VEICULO for excluído ou seu RENAVAL for atualizado, os registros correspondentes em ASSEGURA também serão excluídos ou atualizados.
- A coluna CNPJ_SEGURADORA possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna CNPJ da tabela SEGURADORA, com a opção ON DELETE CASCADE e ON UPDATE CASCADE. Isso significa que, se um registro relacionado em SEGURADORA for excluído ou seu CNPJ for atualizado, os registros correspondentes em ASSEGURA também serão excluídos ou atualizados.

Tabela HISTORICO_CORRIDAS:

A tabela HISTORICO_CORRIDAS possui as seguintes restrições de integridade:

- As colunas NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL são definidas como PRIMARY KEY.
- A coluna CPF_MOTORISTA possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna CPF_MOTORISTA da tabela MOTORISTA, com a opção ON UPDATE CASCADE. Isso significa que, se o valor do CPF_MOTORISTA for atualizado na tabela MOTORISTA, o valor correspondente em CPF_MOTORISTA também será atualizado.
- A coluna CPF_PASSAGEIRO_RESPONSAVEL possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna CPF_PASSAGEIRO da tabela PASSAGEIRO, com a opção ON UPDATE CASCADE. Isso significa que, se o valor do CPF_PASSAGEIRO for atualizado na tabela PASSAGEIRO, o valor correspondente em CPF_PASSAGEIRO_RESPONSAVEL também será atualizado.

Tabela CARONANTES_CORRIDA:

A tabela CARONANTES_CORRIDA possui as seguintes restrições de integridade:

- As colunas NUM_CORRIDA, CPF_PASSAGEIRO_RESPONSAVEL e CPF_PASSAGEIRO_CARONANTE são definidas como PRIMARY KEY.
- A combinação das colunas NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL possui uma FOREIGN KEY (chave estrangeira) referenciando a combinação das colunas NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL da tabela HISTORICO_CORRIDAS, com a opção ON UPDATE CASCADE. Isso significa que, se os valores de NUM_CORRIDA ou CPF_PASSAGEIRO_RESPONSAVEL forem

atualizados na tabela HISTORICO_CORRIDAS, os valores correspondentes em NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL também serão atualizados.

- A coluna CPF_PASSAGEIRO_CARONANTE possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna CPF_PASSAGEIRO da tabela PASSAGEIRO, com a opção ON UPDATE CASCADE. Isso significa que, se o valor do CPF_PASSAGEIRO for atualizado na tabela PASSAGEIRO, o valor correspondente em CPF_PASSAGEIRO_CARONANTE também será atualizado.

Tabela ATENDENTE_CORRIDA:

A tabela ATENDENTE_CORRIDA possui as seguintes restrições de integridade:

- As colunas NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL são definidas como PRIMARY KEY.
- A combinação das colunas NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL possui uma FOREIGN KEY (chave estrangeira) referenciando a combinação das colunas NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL da tabela HISTORICO_CORRIDAS, com a opção ON UPDATE CASCADE. Isso significa que, se os valores de NUM_CORRIDA ou CPF_PASSAGEIRO_RESPONSAVEL forem atualizados na tabela HISTORICO_CORRIDAS, os valores correspondentes em NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL também serão atualizados.
- A coluna CPF_ATENDENTE possui uma FOREIGN KEY (chave estrangeira) referenciando a coluna CPF_ATENDENTE da tabela ATENDENTE, com a opção ON DELETE CASCADE e ON UPDATE CASCADE. Isso significa que, se um registro relacionado em ATENDENTE for excluído ou seu CPF_ATENDENTE for atualizado, os registros correspondentes em ATENDENTE_CORRIDA também serão excluídos ou atualizados.

Tabela CONVERSA:

A tabela CONVERSA possui as seguintes restrições de integridade:

- As colunas NUM_CORRIDA, CPF_PASSAGEIRO_RESPONSAVEL e NUMERO_SEQ são definidas como PRIMARY KEY.
- A combinação das colunas NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL possui uma FOREIGN KEY (chave estrangeira) referenciando a combinação das colunas NUM_CORRIDA e CPF_PASSAGEIRO_RESPONSAVEL da tabela HISTOR

Consultas

1. Dado uma seguradora, identificar quais são os veículos com seguros sob sua responsabilidade e listar as vigências das apólices.

```
SELECT RENAVAM, NUMERO_APOLICE FROM ASSEGURA WHERE  
CNPJ_SEGURADORA = '<insira o CNPJ_SEGURADORA>';
```

Essa consulta retorna os números de RENAVAM dos veículos e os números das apólices dos seguros associados à seguradora especificada.

2. Dado um atendente, descobrir se ele já usou o serviço de corridas e quando.
- ```
SELECT NUM_CORRIDA, DATA_INICIO FROM HISTORICO_CORRIDAS WHERE
NUM_CORRIDA IN (SELECT NUM_CORRIDA FROM CARONANTES_CORRIDA
WHERE CPF_PASSAGEIRO_RESPONSAVEL = '<insira o CPF do atendente>' OR
CPF_PASSAGEIRO_CARONANTE = '<insira o CPF do atendente>');
```

A subconsulta (SELECT NUM\_CORRIDA FROM CARONANTES\_CORRIDA WHERE CPF\_PASSAGEIRO\_RESPONSAVEL = '<insira o CPF do atendente>' OR CPF\_PASSAGEIRO\_CARONANTE = '<insira o CPF do atendente>') retorna os números de corrida em que o atendente especificado participou, seja como passageiro responsável ou como passageiro caronante.

A cláusula WHERE NUM\_CORRIDA IN (...) filtra as linhas da tabela HISTORICO\_CORRIDAS, selecionando apenas as corridas cujos números estão presentes na lista retornada pela subconsulta.

3. Listar quais foram os passageiros que nunca fizeram uma corrida.
- ```
SELECT CPF_PASSAGEIRO, CART_CREDIT FROM PASSAGEIRO  
WHERE CPF_PASSAGEIRO NOT IN ( SELECT  
CPF_PASSAGEIRO_RESPONSAVEL FROM HISTORICO_CORRIDAS UNION  
SELECT CPF_PASSAGEIRO_RESPONSAVEL FROM CARONANTES_CORRIDA)  
AND CPF_PASSAGEIRO NOT IN ( SELECT CPF_PASSAGEIRO_CARONANTE  
FROM CARONANTES_CORRIDA);
```

A subconsulta (SELECT CPF_PASSAGEIRO_RESPONSAVEL FROM HISTORICO_CORRIDAS UNION SELECT CPF_PASSAGEIRO_RESPONSAVEL FROM CARONANTES_CORRIDA) retorna todos os CPFs dos passageiros responsáveis em histórico de corridas e caronantes em caronas já realizadas.

A cláusula WHERE CPF_PASSAGEIRO NOT IN (...) filtra os passageiros da tabela PASSAGEIRO que não estão presentes na lista de CPFs dos passageiros responsáveis em histórico de corridas ou caronantes.

A subconsulta SELECT CPF_PASSAGEIRO_CARONANTE FROM CARONANTES_CORRIDA retorna todos os CPFs dos passageiros caronantes em caronas já realizadas.

A cláusula AND CPF_PASSAGEIRO NOT IN (...) filtra ainda mais os passageiros da tabela PASSAGEIRO, excluindo aqueles que também estão presentes na lista de CPFs dos passageiros caronantes.

o.

4. Dado uma corrida, listar todas as perguntas e respostas feitas ao atendente responsável.
- ```
SELECT TXT_PERGUNTA, TXT_RESPOSTA FROM CONVERSA WHERE
NUM_CORRIDA = <insira o número da corrida> AND
CPF_PASSAGEIRO_RESPONSAVEL = '<insira o CPF do passageiro responsável>';
```



Essa consulta lista todas as perguntas e respostas feitas ao atendente responsável por uma determinada corrida. É necessário especificar o número da corrida e o CPF do passageiro responsável.

- 5.1. Dado um motorista, descobrir quantas corridas ele já realizou.

```
SELECT COUNT(*) FROM HISTORICO_CORRIDAS WHERE CPF_MOTORISTA =
'<insira o CPF do motorista>;
```

Essa consulta retorna o número total de corridas realizadas por um motorista específico.

- 5.2. Dada uma data, listar todas as corridas.

```
SELECT * FROM HISTORICO_CORRIDAS WHERE DATA_INICIO = '<insira a
data>;
```

Essa consulta retorna todas as informações das corridas que ocorreram em uma data específica. É necessário fornecer a data de início das corridas desejada.