

## TRABALHO SBD PARTE 1

Alunos:

Marcus Vinícius Almeida Florêncio	12021BCC016
Cauã Pereira Neres	12021BCC005
Pedro Henrique de Oliveira	11921BCC010
Hendrik Abdalla Hermann	11911BCC034

### Requisitos adicionais

- 1 - Viagem é registrada após encontrar veículo.
  - 2 - Sendo um gerador de cupom de sorteios de prêmios que pode ser gerado na viagem do passageiro, cada cupom possui um identificador e uma data de validade.
  - 3 - Uma nova entidade chamada “UberMoto” que tem como atributo as cilindradas do veículo.
- 

### Motivações:

#### Agregação

A gente fez uma agregação em cima do relacionamento *Solicita* que envolve o *Usuário* e *Veículo*

#### Status

O status foi pensado de uma forma onde conseguimos ter uma *descrição*, caso haja cancelamento, *quem cancelou* e o *id* desse status. Para que assim consigamos ter um controle melhor sobre essas informações. Fizemos assim para que fique mais fácil de adicionar um novo status e também dessa forma os status já estarão pré-cadastrados.

#### Forma de pagamento

Para representar a forma de pagamento a gente criou uma entidade chamada *forma\_pagamento* que possui como atributo uma *id* e o *tipo*. Fizemos assim para que fique mais fácil de adicionar um novo método de pagamento e também dessa forma os métodos de pagamentos já estarão pré-cadastrados.

**Link do miro com o diagrama da parte 1 em alta qualidade :**

[https://miro.com/app/board/uXjVMjlZkQk=?share\\_link\\_id=151033579594](https://miro.com/app/board/uXjVMjlZkQk=?share_link_id=151033579594)

#### Definições de criação do diagrama

a. Não alteramos nem uma das solicitações do requisito, colocamos o "cpf" como nossa chave.

b. Definimos um atributo "status" para conseguir capturar a informação se o motorista está ativo ou não.

c. Definimos 1 para N, sendo 1 motorista para N carros. Assim possibilitando que o motorista tenha vários veículos.

d. Não teve nem uma alteração em relação aos requisitos, mas adicionamos o atributo "status" para que possamos validar se um veículo está inativo ou não.

e. Com o atributo "status" conseguir colocar se o motorista está disponível para viagens, no meio de uma viagem ou finalizou a corrida.

f. Colocamos um atributo multi-valorado para conseguir registrar quais acessórios cada carro tem em si.

g. Com o atributo "tipo" conseguimos declarar se um veículo é sedan ou não, com o atributo "num\_passageiro" guardamos a informação dos passageiros possíveis.

h. Pensamos em uma disjunção para assim conseguimos definir a categoria de cada carro. Categorias:

UberX sendo Última revisão para que o passageiro verifique,  
UberComfort sendo uma taxa adicional o passageiro,  
UberMoto sendo a cilindrada do veículo;

j. Para verificarmos os usuários temos como base o "nro\_celular" com possibilidade do usuário ter várias contas.

k. Foi definido um atributo de "avaliação" no motorista para disponibilizar para o usuário.

l. Definimos uma disjunção para quando o cliente consiga alcançar a meta ele vire um "ClienteVip" .

m. Definimos uma agregação para solicitações de viagens.

n. Com o atributo "horario\_corrida" sendo um atributo composto podemos capturar horário que o motorista chegou à origem e depois ao seu destino e com essa informação pode trazer o valor do atributo "duração" da corrida.

p. Na agregação definimos o valor destinado ao motorista com o atributo "valor\_motorista" e o valor destinado à UberLand com o atributo "valor\_app".

q. Nossa agregação tem uma relação de "possui" um entidade "status" com os seguintes atributos "autor\_cancelamento", "id\_status", "descricao".

r. Com o atributo multi-valorado "ponto\_parada" podemos adicionar vários pontos de para em um corrida.

s. Definimos uma relação para o pagamento da corrida.