

# Esquema Relacional

Usuario(cpf, email, senha, endereco(rua, numero, cidade, estado, cep))

Funcionario(cpf\_usu\*, cargo, salario, data\_de\_contratacao, cpf\_supervisor\*)  
cpf\_usu referencia Usuario(cpf)  
cpf\_supervisor referencia Funcionario(cpf\_usu)

Cliente(cpf\_usu\*, data\_de\_adesao)  
cpf\_usu referencia Usuario(cpf)

Telefone(cpf\_usu\*, num\_telefone)  
cpf\_usu referencia Usuario(cpf)

Carro(chassi, marca, modelo, ano, preco, km, cor)

Pedido(hash, tipo, data, data\_entrega, status)

Equipamento(id, nome, descricao)

Servico(protocolo, nome)

PagamentoServico(protocolo\_serv\*, id\_pag, metodo, valor)  
protocolo\_serv referencia Servico(protocolo)

PagamentoPedido(hash\_ped, id\_pag, metodo, valor)  
hash\_ped referencia Pedido(hash)

Anuncio(chassi\_car, url)  
chassi\_car referencia Carro(chassi)

Solicita(cpf\_cli\*, cpf\_func\*, protocolo\_serv\*)  
cpf\_cli referencia Cliente(cpf\_usu)  
cpf\_func referencia Funcionario(cpf\_usu)  
protocolo\_serv referencia Servico(protocolo)

Vende(chassi\_car\*, hash\_ped\*, cpf\_func\*, cpf\_cli\*)  
hash\_ped referencia Pedido(hash)  
cpf\_func referencia Funcionario(cpf\_usu)  
cpf\_cli referencia Cliente(cpf\_usu)  
chassi\_car referencia Carro(chassi)

Envolve(protocolo\_serv\*, id Equip\*);  
protocolo\_serv referencia Servico(protocolo)  
id Equip referencia Equipamento(id)

Comenta(cpf\_cli\*, cpf\_func\*, url\_anun\*, conteudo, data)  
cpf\_cli referencia Cliente(cpf\_usu)  
cpf\_func referencia Funcionario(cpf\_usu)  
url referencia Anuncio(url)

Responde(cpf\_func, url\_anun, data, conteudo);  
cpf\_func referencia Funcionario(cpf\_usu)  
url\_anun referencia Anuncio(url\_anun)

## 1ª Forma Normal

### Entidade 1:

- Substituir atributo endereco do Usuario nos múltiplos atributos atômicos que o compõem: rua, número, cidade, estado, cep.

Usuario(cpf, email, senha, endereco(rua, numero, cidade, estado, cep))

---

### Resultado:

Usuario(cpf, email, senha, rua, numero, cidade, estado, cep)

## 2ª Forma Normal

### Entidade 1:

- Em PagamentoServico, o atributo "valor" depende parcialmente da chave composta (cada "protocolo\_serv" representa unicamente um "valor"). Portanto, criamos uma nova relação entre protocolo e valor.

PagamentoServico(protocolo\_serv\*, id\_pag, metodo, valor)  
protocolo\_serv referencia Servico(protocolo)

---

### Resultado:

PagamentoServico(protocolo\_serv\*, id\_pag, metodo)  
protocolo\_serv referencia Servico(protocolo)

ProtocoloValor(protocolo\_serv, valor)  
protocolo\_serv referencia PagamentoServico(protocolo\_serv)

## 3ª Forma Normal

### Entidade 1:

- Em PagamentoServico, o atributo "valor" depende parcialmente da chave composta (cada "protocolo\_serv" representa unicamente um "valor"). Portanto, criamos uma nova relação entre protocolo e valor.

Funcionario(cpf\_usu\*, cargo, salario, data\_de\_contratacao, cpf\_supervisor\*)  
 cpf\_usu referencia Usuario(cpf)  
 cpf\_supervisor referencia Funcionario(cpf\_usu)

---

#### Resultado:

Cargo(cargo\_func, salario)

Funcionario(cpf\_usu\*, cargo\*, data\_de\_contratacao, cpf\_supervisor\*)  
 cpf\_usu referencia Usuario(cpf)  
 cpf\_supervisor referencia Funcionario(cpf\_usu)  
 cargo referencia Cargo(cargo\_func)

#### Entidade 2:

- Em Usuário, os atributos "rua", "cidade", "estado" podem ser determinados pelo atributo "cep". Portanto, criamos a relação CEP agrupando esses atributos e utilizando "cep" como chave primária.

Usuario(cpf, email, senha, rua, numero, cidade, estado, cep)

---

#### Resultado:

CEP(cep, rua, cidade, estado)

Usuario(cpf, cep\*, email, senha, numero)  
 cep referencia CEP(cep)

#### Entidade 3:

- Em Carro, "modelo" é determinante de marca. Portanto, criamos a relação ModeloMarca, ligando o modelo à montadora que detém os direitos sobre o nome do modelo.

Carro(chassi, marca, modelo, ano, preco, km, cor)

---

#### Resultado:

ModeloMarca(modelo, marca)

Carro(chassi, modelo\_mc\*, ano, preco, km, cor)

modelo\_mc referencia ModeloMarca(modelo)

## Forma Normal de Boyce-Codd

Nenhum passo necessário.

### 4ª Forma Normal

Nenhum passo necessário.

### Versão final

CEP(cep, rua, cidade, estado)

Usuario(cpf, cep\*, email, senha, numero)  
cep referencia CEP(cep)

Cargo(cargo\_func, salario)

Funcionario(cpf\_usu\*, cargo\*, data\_de\_contratacao, cpf\_supervisor\*)  
cpf\_usu referencia Usuario(cpf)  
cpf\_supervisor referencia Funcionario(cpf\_usu)  
cargo referencia Cargo(cargo\_func)

Cliente(cpf\_usu\*, data\_de\_adexao)  
cpf\_usu referencia Usuario(cpf)

Telefone(cpf\_usu\*, num\_telefone)  
cpf\_usu referencia Usuario(cpf)

ModeloMarca(modelo, marca)

Carro(chassi, modelo\_mc\*, ano, preco, km, cor)  
modelo\_mc referencia ModeloMarca(modelo)

Pedido(hash, tipo, data, data\_entrega, status)

Equipamento(id, nome, descricao)

Servico(protocolo, nome)

PagamentoServico(protocolo\_serv\*, id\_pag, metodo)  
protocolo\_serv referencia Servico(protocolo)

ProtocoloValor(protocolo\_serv\*, valor)  
protocolo\_serv referencia PagamentoServico(protocolo\_serv)

PagamentoPedido(hash\_ped\*, id\_pag, metodo, valor)  
hash\_ped referencia Pedido(hash)

Anuncio(chassi\_car\*, url)  
chassi\_car referencia Carro(chassi)

Solicita(cpf\_cli\*, cpf\_func\*, protocolo\_serv\*)  
cpf\_cli referencia Cliente(cpf\_usu)  
cpf\_func referencia Funcionario(cpf\_usu)  
protocolo\_serv referencia Servico(protocolo)

Vende(chassi\_car\*, hash\_ped\*, cpf\_func\*, cpf\_cli\*)  
hash\_ped referencia Pedido(hash)  
cpf\_func referencia Funcionario(cpf\_usu)  
cpf\_cli referencia Cliente(cpf\_usu)  
chassi\_car referencia Carro(chassi)

Envolve(protocolo\_serv\*, id Equip\*);  
protocolo\_serv referencia Servico(protocolo)  
id Equip referencia Equipamento(id)

Comenta(cpf\_cli\*, cpf\_func\*, url\_anun\*, conteudo, data)  
cpf\_cli referencia Cliente(cpf\_usu)  
cpf\_func referencia Funcionario(cpf\_usu)  
url referencia Anuncio(url)

Responde(cpf\_func, url\_anun, data, conteudo);  
cpf\_func referencia Funcionario(cpf\_usu)  
url\_anun referencia Anuncio(url\_anun)