

Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação
Departamento de Ciências de Computação

Modelagem do MER -> MR

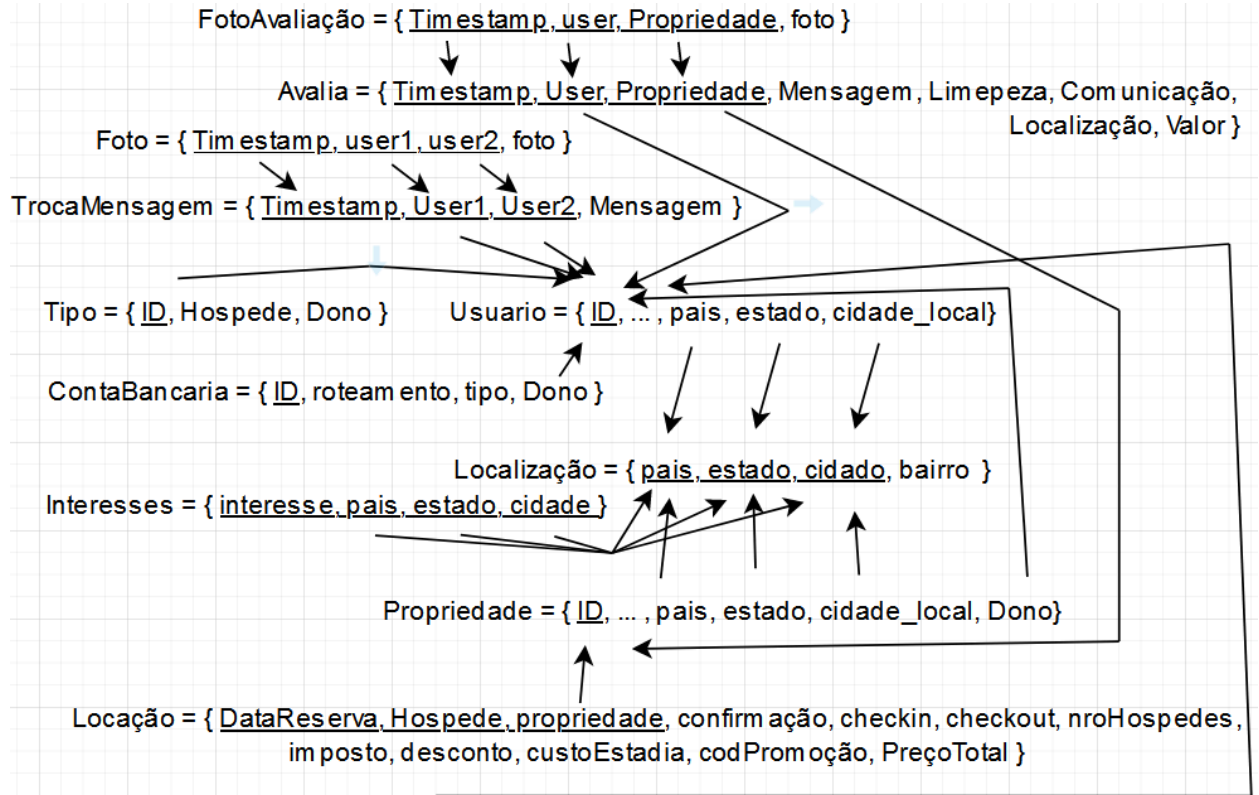
Luca Gomes Urssi - 10425396 - lucaurssi@usp.br

Vinicius Flnke Raiter José - 9791052 - vinicius.jose@usp.br

Gabriel Santos Brito - 10284250

Prof. Dr. Caetano Traina Júnior

MR



Descrição passo a passo

Passo 1: Mapear Entidades Regulares:

Localização = { País, Estado, Cidade, Bairro }

Passo 2: Mapear Entidades fracas:

ContaBancaria = { ID, Roteamento, tipo }

Propriedade = { ID, nome, número, rua, cidade, CEP, Classificação, Nro_Banheiros, Nro_Quartos, Max_Hóspedes, Diaria, Taxa_Limpeza, min_Noite, max_Noite, hr_checkin, hr_checkout, disponibilidade_inicio, disponibilidade_fim, nro&tipo_camras, comodidades, regras }

Em **Laranja**, está representado as chaves candidatas pedidas no enunciado. Não achei como fazer underline duplo no editor de texto

2a: Mapear Generalização:

Utilizando o procedimento padrão **G6**, mapeando o **Usuário**, **Hóspede** e **Dono** em uma só relação, com uma relação separada para identificar se o **Usuário** é um **dono** e/ou um **hóspede**, contendo dois atributos booleanos.

Usuário = { ID, nome, sobrenome, telefone, DataNasc, sexo, email, senha, numero, rua, cidade, CEP }

Tipo_Usuario = { ID, Hóspede, Dono }

2b: Mapear Agregações:

Troca_Mensagem = { Timestamp, User1, User2, mensagem }

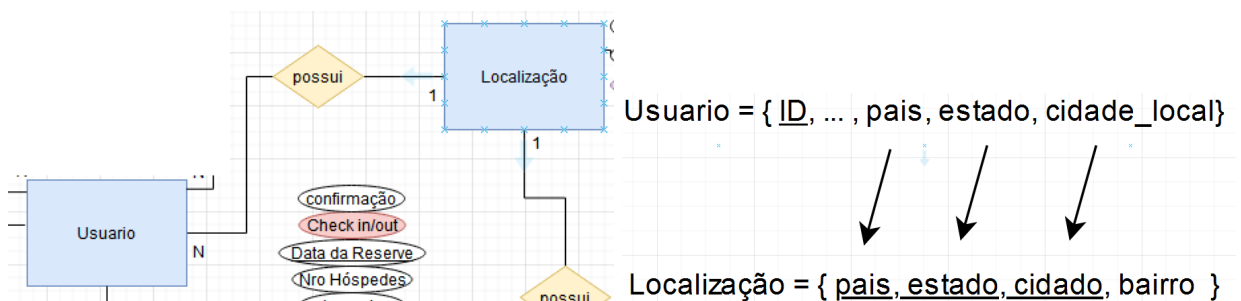
Avalia = { Timestamp, Hóspede, Propriedade, mensagem, limpeza, comunicação, localização, valor }

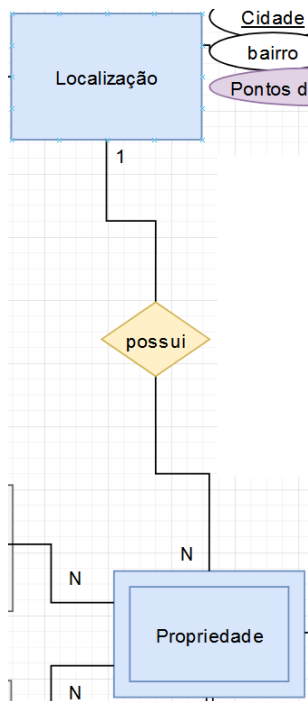
Locação = { DataReserva, Hóspede, Propriedade, confirmação, checkin, checkout, nro_hóspedes, imposto, desconto, custo_estadia, codPromoção, preçoTotal }

Passo 3,4,5: Mapear Relações 1-1, 1-N, N-N

Não temos 1-1 no modelo.

1-N :

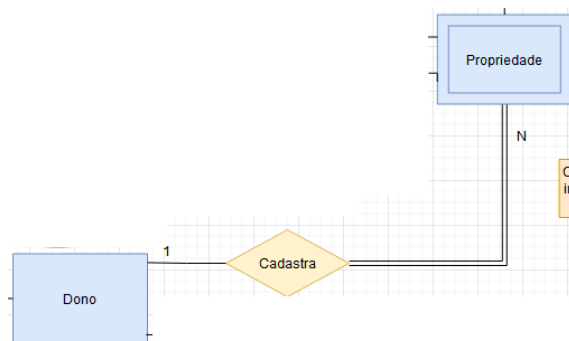




Usuario = { ID, ... , pais, estado, cidade_local }

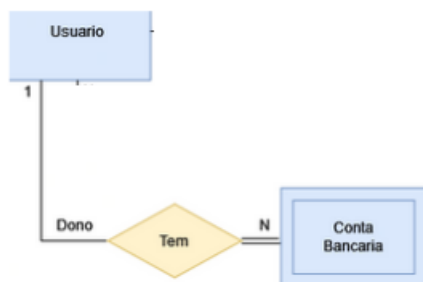
Localização = { pais, estado, cidade, bairro }

Propriedade = { ID, ... , pais, estado, cidade_local }



Usuario = { ID, ... }

Propriedade = { ID, ..., Dono }

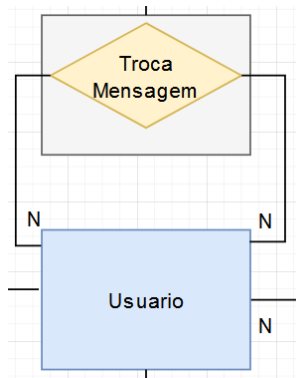


ContaBancaria = {ID, Roteamento, tipo, Dono}

Usuário = { ID, ... }

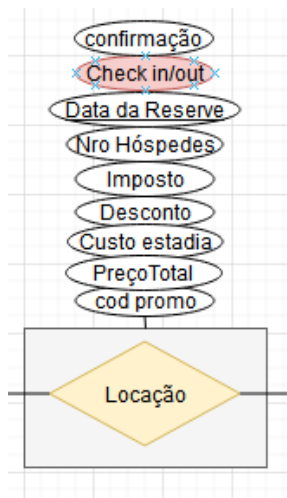
Propriedade { ID, ..., Dono }

N-N :

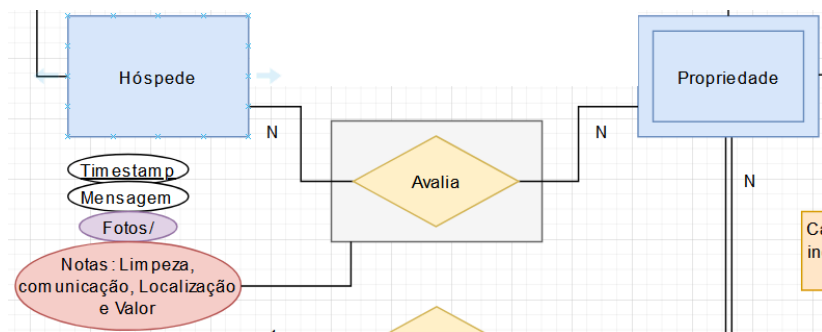


$TrocaMensagem = \{ \underline{Timestamp}, \underline{User1}, \underline{User2}, Mensagem \}$

$Usuario = \{ \underline{ID}, \dots \}$

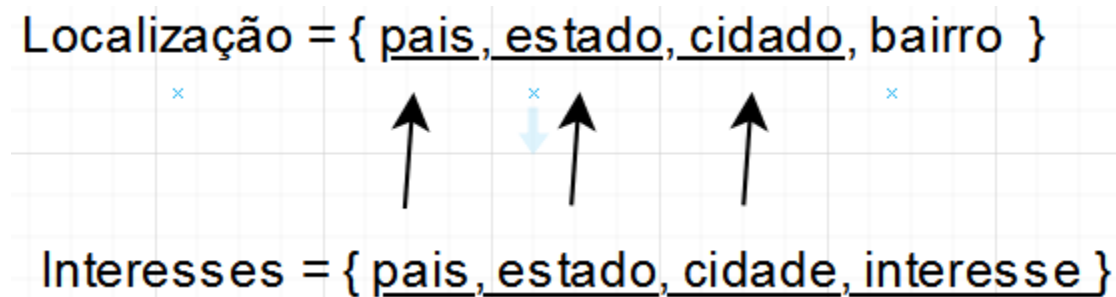
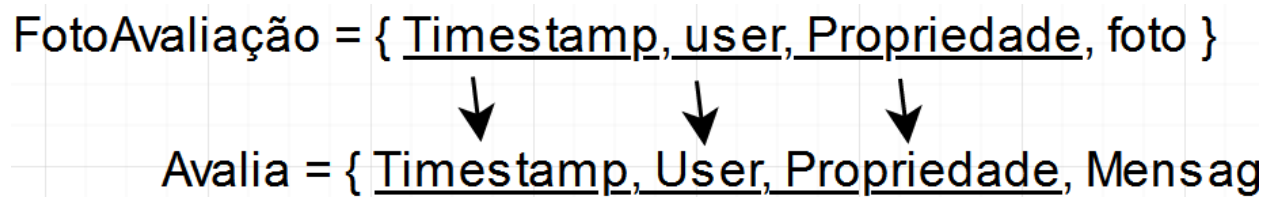
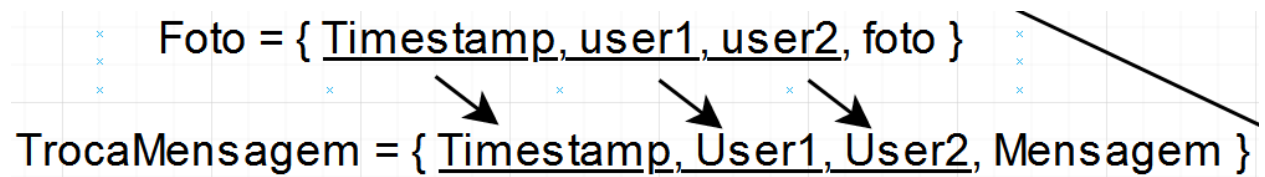


$Locação = \{ \underline{DataReserva}, \underline{Hospede}, \underline{propriedade}, confirmação, checkin, checkout, nroHospedes, imposto, desconto, custoEstadia, codPromoção, PreçoTotal \}$



$Avalia = \{ \underline{Timestamp}, \underline{User}, \underline{Propriedade}, Mensagem, Lim epeza, Comunicação, Localização, Valor \}$

Passo 7: Mapear atributos multivalorados



Restrições:

Na relação de Locação temos os atributos de check in e check out. Esses atributos tem duas restrições necessárias para evitar que a ocorra erros na locação :

- Estar dentro do período de disponibilidade da propriedade.
- Não conflitar com check in e check out de outra Locação na mesma Propriedade.

Ainda na relação de Locação, temos que o número de Hóspedes deva ser menor que o número de Hóspedes máximo da Propriedade.

Atributos não nulos: (chaves primárias são todas não nulas)

Usuario:

Nome, Sobrenome, Telefone

Locação:

Check In, check out, confirmação, nroHóspedes, preço Total

ContaBancaria:

Roteamento, tipo

Propriedade:

Nome, rua, número, cidade, CEP, máx hóspedes, Diária, classificação