



# **LaLi: Sistema de Caronas**

## **Documentação do Projeto** **Versão 1.0**

Elaborado por: Larissa Espelocin Justen  
e

Lincoln Vinicius Schreiber

Curso: Engenharia de Computação  
Disciplina: Banco de Dados  
Professora: Fabrícia Damando Santos

Guaíba / RS  
UERGS  
2018

# SUMÁRIO

<b>1. INTRODUÇÃO – APRESENTAÇÃO DO PROBLEMA</b>	<b>3</b>
<b>2. OPORTUNIDADE PERCEBIDA</b>	<b>3</b>
<b>3. SOLUÇÃO</b>	<b>3</b>
<b>4. LEVANTAMENTO DE REQUISITOS</b>	<b>4</b>
4.1 Requisitos Funcionais	4
4.2 Requisitos Não-Funcionais	5
<b>5. REGRAS DE NEGÓCIO</b>	<b>6</b>
<b>6. DIAGRAMAS DE CASO DE USO</b>	<b>7</b>
<b>7. DIAGRAMA ER</b>	<b>8</b>
<b>8. CANVAS</b>	<b>9</b>

## **1. INTRODUÇÃO – APRESENTAÇÃO DO PROBLEMA**

Dificuldade de comunicação entre os alunos que precisam de carona e os alunos que oferecem as caronas. Em razão disso, alguns motoristas fazem o percurso de ida/volta sozinhos ou com poucas pessoas.

## **2. OPORTUNIDADE PERCEBIDA**

Oportunidade de criação de um aplicativo no decorrer da cadeia de banco de dados com o conhecimento adquirido na mesma que supra o problema.

## **3. SOLUÇÃO**

Um aplicativo de estrutura simples, onde o motorista se cadastra e divulga sua disponibilidade para caronas, e o passageiro consiga contato com o motorista de forma simples e rápida. Passageiros cadastrados podem solicitar reserva (possível nova versão).

## **4. LEVANTAMENTO DE REQUISITOS**

### **4.1 Requisitos Funcionais**

Requisitos referentes ao Sistema:

RF13 – O Sistema deve permitir notificar os Passageiros sobre as caronas;

RF14 – O Sistema deve permitir notificar os Motoristas sobre a carona.

Requisitos referentes ao Visitante:

RF01 – Pode fazer cadastro através do aplicativo;

RF03 – Pode ver informações das caronas.

Requisitos referentes ao Passageiro:

RF02 – Pode logar através do aplicativo;

RF03 – Pode ver informações das caronas;

RF04 – Pode atualizar seu perfil;

RF05 – Solicitar Reserva de carona;

RF06 – Manter Agenda de carona.

RF11 – Avaliar Motorista da Carona.

Requisitos referentes ao Motorista:

Todos os requisitos do Passageiro;

RF07 – Pode Aceitar ou Recusar pedido de Reserva.

RF12 – Avaliar Passageiros da carona.

Requisitos referentes ao Administrador:

RF08 – Banir e Desbanir Usuários;

RF09 – Promover e Despromover Usuários a Administrador;

RF10 – Aceitar ou Recusar Perfil Carona.

## **4.2 Requisitos Não-Funcionais**

RNF01 – Um sistema operacional Android, a partir da versão 4.2;

RNF02 – Acesso à internet;

RNF03 – Uso da Linguagem GDScript;

## 5. REGRAS DE NEGÓCIO

RN01 – Um Passageiro ao criar a agenda de carona se torna um Motorista se aceito pelo Administrador;

RN02 – Um Motorista ao excluir sua agenda de carona, se torna um Passageiro;

RN03 – As senhas dos Passageiros devem conter de 5 a 15 caracteres;

RN04 – O login dos Passageiros deve conter apenas letras e números, e ter de 6 até 20 caracteres.

RN05 – Passageiros e Motoristas banidos pelo administrador não podem mais logar no sistema ou se cadastrar novamente até ser desbanido;

RN06 – Todas as vezes que os passageiros solicitarem uma carona, o Motorista é Notificado.

RN07 – Todas as vezes que o Motorista Aceitar ou Recusar a reserva de carona, o Passageiro é notificado.

RN08 – Motorista é notificado para confirmar cada carona.

## 6. DIAGRAMAS DE CASO DE USO

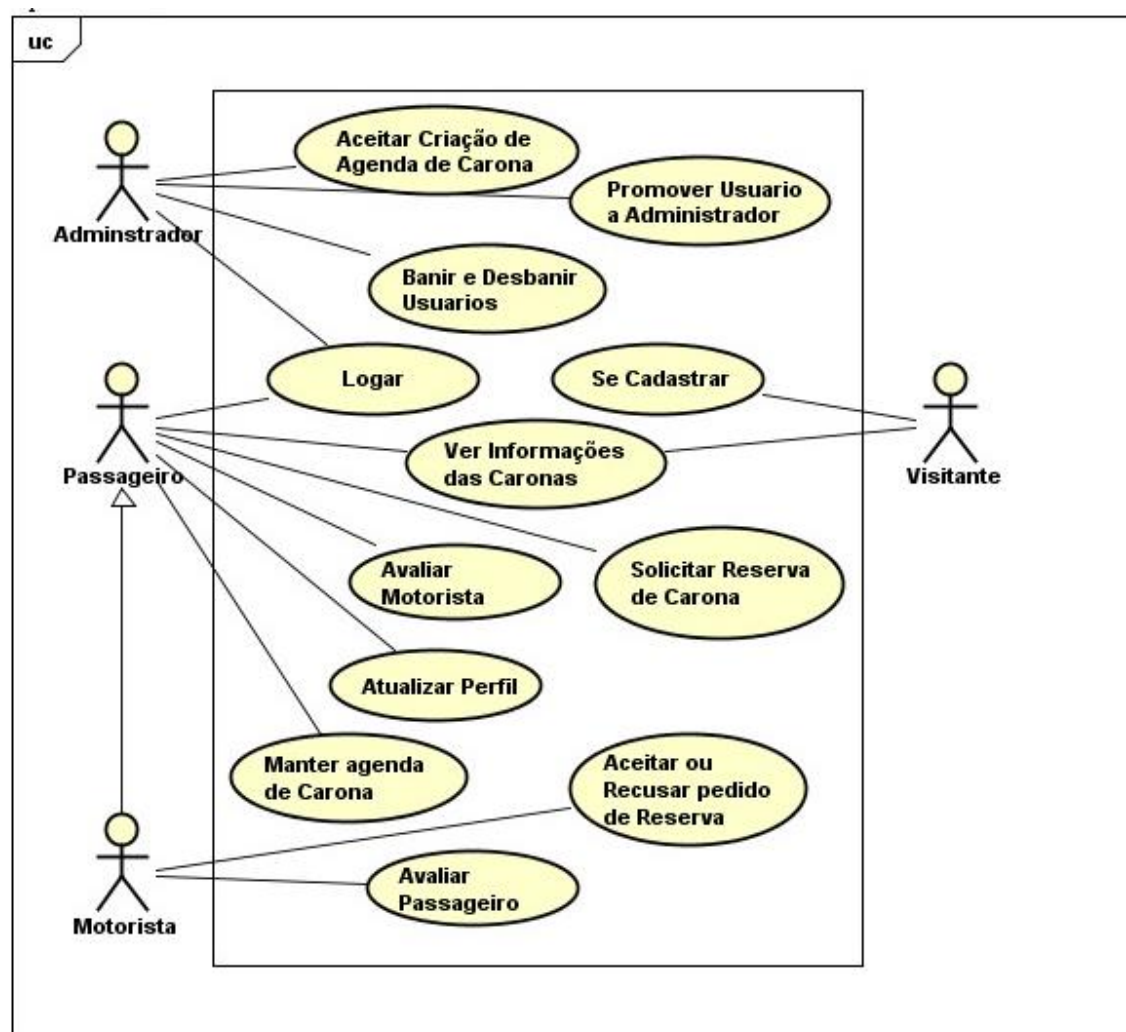


Figura 1. Diagrama de casos de uso

## 7. DIAGRAMA ER

A figura a seguir mostra diagrama ER do modelo lógico do banco de dados a ser implementado no sistema.

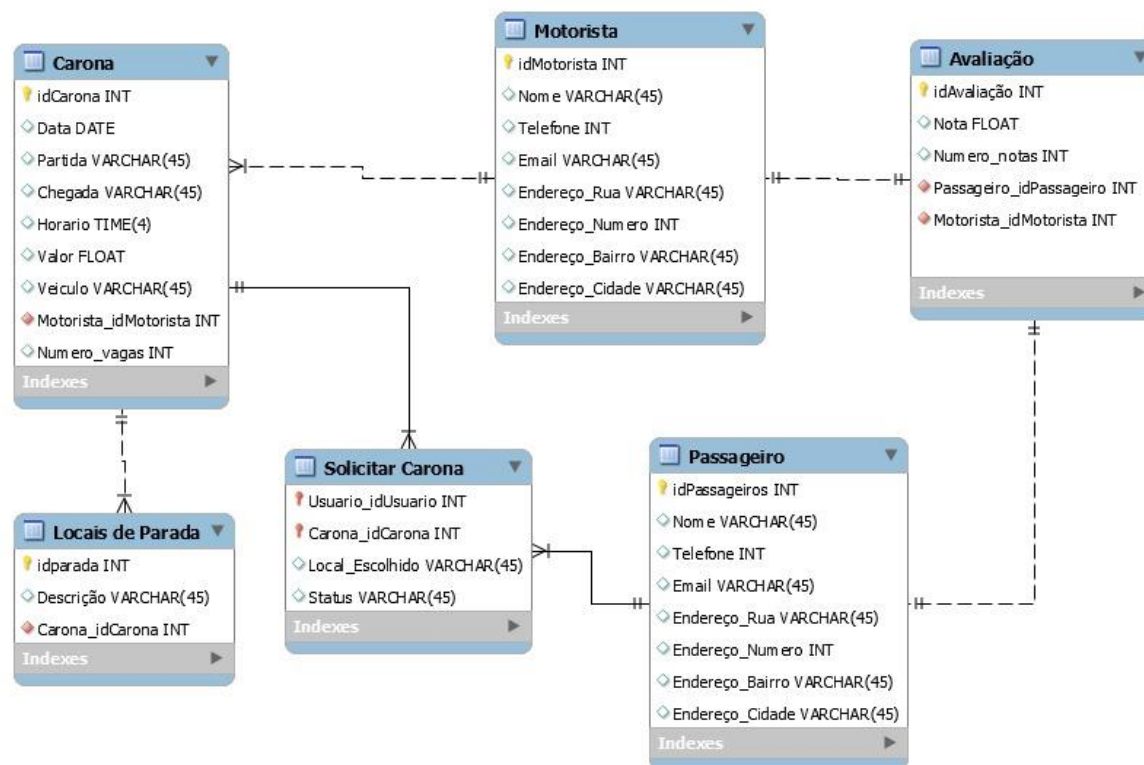


Figura 2. Diagrama ER do banco de dados do sistema



## 8. CANVAS

Aqui contem a imagem do CANVAS feito em aula.

Figura 3. CANVAS do projeto sem o esquema conceitual!

