

Afranio Martins Caires
Elizabeth Barbosa de Souza
Sávio Campos Vieira
Vanderson Lopes Amaral

RELATÓRIO TÉCNICO: APLICATIVO DE CARONAS REGIONAL

Araçuaí-MG

2024

Afranio Martins Caires
Elizabeth Barbosa de Souza
Sávio Campos Vieira
Vanderson Lopes Amaral

RELATÓRIO TÉCNICO: APLICATIVO DE CARONAS REGIONAL

Relatório técnico para descrição da modelagem, codificação e demais atividades realizadas durante o Projeto Integrador em Análise e Desenvolvimento de Sistemas. Aplicativo de Caronas Regional: VemComigo

Instituto Federal de Educação, Ciência e Tecnologia do Norte de Minas Gerais (IFNMG)
Tecnologia em Análise e Desenvolvimento de Sistemas
Núcleo de Informática

Araçuaí-MG

2024

Sumário

| | | |
|------------|---|-----------|
| 1 | INTRODUÇÃO | 4 |
| 1.1 | Contextualização | 4 |
| 1.2 | Objetivos | 6 |
| 1.3 | Público-alvo e Benefícios | 7 |
| 2 | ATIVIDADES DESENVOLVIDAS | 8 |
| 2.1 | Escopo do Projeto | 8 |
| 2.2 | Organização Inicial | 8 |
| 2.2.1 | Requisitos Funcionais | 8 |
| 2.2.2 | Critérios de Aceitação | 10 |
| 2.3 | Back-end | 12 |
| 2.3.1 | Arquitetura do <i>Back-end</i> | 12 |
| 2.3.2 | Banco de Dados | 13 |
| 2.3.3 | Diagrama Entidade-Relacionamento (MER) | 13 |
| 2.3.4 | Modelo Entidade-Relacionamento (DER) | 13 |
| 2.3.5 | Dicionário de Dados | 13 |
| 2.3.6 | Relacionamentos | 18 |
| 2.4 | Front-end | 20 |
| 2.4.1 | Arquitetura do Front-End | 20 |
| 2.4.2 | Design e UX (Experiência do Usuário) | 20 |
| 2.4.3 | Componentização e Reutilização | 21 |
| 2.4.4 | Diagrama de Classes | 21 |
| 2.4.5 | Diagrama de Caso de Uso | 21 |
| 2.5 | Tecnologias e Ferramentas | 22 |
| 2.5.1 | Ambiente de Trabalho | 22 |
| 2.5.2 | Linguagem de Programação | 22 |
| 2.5.3 | <i>Framework</i> | 23 |
| 2.5.4 | Banco de Dados | 23 |
| 2.5.5 | Produtividade | 23 |
| 3 | CONSIDERAÇÕES FINAIS | 24 |
| | REFERÊNCIAS | 25 |
| | ANEXO A – MODELO ENTIDADE-RELACIONAMENTO (MER) | 26 |

| | |
|---|-----------|
| ANEXO B – DIAGRAMA ENTIDADE-RELACIONAMENTO (DER) | 28 |
| ANEXO C – DIAGRAMA DE CLASSES | 30 |
| ANEXO D – DIAGRAMA DE CASO DE USO | 32 |

1 INTRODUÇÃO

A mineração é uma atividade valiosa para a manutenção e o desenvolvimento da sociedade. O ser humano está sempre buscando novas formas de facilitar a sua sobrevivência por meio de ferramentas criadas com os minérios extraídos da terra, que vão desde um simples livro, que necessita da celulose, um polímero extraído de fontes vegetais, até às grandes invenções como aviões e computadores.

Nota-se um aumento na demanda mundial por lítio, causado por uma corrida global para a substituição da atual matriz energética. O site *Statista*, plataforma global de dados e *business intelligence*, registrou um salto crescente na demanda do mineral entre os anos de 2022 e 2023, além de expectativas maiores para a próxima década, conforme Jaganmohan (2024). No Brasil, uma região se destaca na oferta de jazidas minerais: o Vale do Jequitinhonha.

Paralelamente ao progresso, a mineração de lítio levanta questões complexas que envolvem aspectos ambientais, econômicos e sociais. Este capítulo tem a finalidade de apresentar uma contextualização sobre o desenvolvimento de um software de caronas, criado com o objetivo de resolver uma das demandas atuais causadas pelo novo ciclo econômico da região.

1.1 Contextualização

O “*Lithium Valley Brazil*” (Vale do Lítio Brasileiro) foi o nome dado ao novo projeto estadual de extrativismo em Minas Gerais. De acordo com o Governo do Estado de Minas Gerais, no dia 9 de maio, durante um evento da bolsa de valores em Nova Iorque, o então governador Romeu Zema liderou uma iniciativa responsável por atrair investidores do mundo inteiro, Gerais (2024).

Mais uma vez, o Estado de Minas surpreendeu a indústria mundial com uma recente descoberta de ricas jazidas de lítio, mineral de suma importância para a economia global, sendo utilizado em ligas metálicas, medicamentos e, principalmente, nas baterias de celulares, computadores e carros elétricos. O lítio é extraído com a finalidade de ser exportado, assim como a maioria dos minérios.

Segundo a Secretaria de Desenvolvimento Econômico de Minas Gerais, SEDE-MG (2023), as primeiras 15 mil toneladas de lítio extraídas no Vale do Jequitinhonha foram entregues no Porto de Vitória, no Estado do Espírito Santo, em julho, assim sendo o pontapé inicial do projeto coordenado pelo Governo de Minas Gerais, com a finalidade de atrair investimentos e empregos ao passo que promete desenvolver a região. A cobiça

pelo “Ouro Branco”, nome popular do mineral, está associada a uma demanda cada vez maior por fontes de energia limpa como alternativa aos combustíveis fósseis. O lítio se tornou responsável por intensas disputas geopolíticas pelo seu domínio uma vez que a Tesla, empresa norte-americana de carros elétricos gerenciada pelo bilionário Elon Musk, disputa com a gigante chinesa BYD pela prioridade na compra do lítio extraído no Vale do Jequitinhonha León (2024).

Entre as empresas de mineração que operam na região em destaque encontra-se a *Sigma Lithium*, empresa canadense que se destaca no cenário global de extração do lítio. No início de 2023, a empresa inaugurou o seu complexo, atualmente o quarto maior produtor mundial Mineração (2024). O projeto de extração da Sigma é baseado na sustentabilidade uma vez que toda a cadeia de produção não utiliza barragens de rejeito, água potável, agentes químicos nocivos ao ambiente ou carvão mineral como fonte de energia. Logo o produto final da mineradora foi batizado como Lítio Verde.

Entretanto, apesar das expectativas criadas ao redor de tal minério, nota-se alguns impactos na região brasileira mais promissora para a extração do lítio. Uma das demandas causadas pelo atual ciclo econômico surge do fato de que a região carece de estrutura urbana adequada. Em uma reportagem de Vasconcelos (2024), moradores apontam problemas como superlotação de equipamentos públicos de saúde, adoecimento mental e físico, contaminação das águas, danos nas estruturas das casas e desgastes na malha rodoviária local. Vale ressaltar que as reservas estão localizadas no Norte e Nordeste de Minas Gerais, em uma região onde muitas cidades possuem baixos níveis no Índice de Desenvolvimento Humano (IDH).

A Secretaria de Desenvolvimento Econômico de Minas Gerais afirma que o projeto Vale do Lítio é formado por 14 cidades: Araçuaí, Capelinha, Coronel Murta, Itaobim, Itinga, Malacacheta, Medina, Minas Novas, Pedra Azul, Virgem da Lapa, Teófilo Otoni e Turmalina, no Nordeste de Minas, e Rubelita e Salinas, no Norte mineiro. A reportagem de Vasconcelos (2024) deixa em evidência, que a instalação da mineradora estrangeira na Grota do Cirilo forçou uma intensa mudança nas cidades supracitadas, seja pela carência de mão de obra especializada ou pela ausência de serviços básicos em algumas cidades.

A falta de especialização local pode ter sido um dos fatores responsáveis por causar ondas de migrações de trabalhadores, muitos dos quais são de lugares diversos, atraídos com a possibilidade de bons salários e oportunidades de promoção. Seja na rede de hotelaria ou na oferta de alugueis, a região sofre com a especulação de preços causada pela elevada demanda por moradia paralela à baixa oferta de imóveis em uma única cidade. A solução encontrada por alguns dos trabalhadores foi buscar acomodações nas cidades próximas de onde trabalham, causando as chamadas migrações pendulares.

Paralelamente, a falta de estrutura urbana na região dificulta a vida dos moradores. O Vale do Jequitinhonha carece de uma rede de transporte entre as cidades devido ao

escasso número de linhas rodoviárias, que muitas das vezes operam em horários específicos, uma vez ao dia. Outra alternativa para o deslocamento seria o transporte por meio de veículos particulares, mas tal possibilidade é limitada pelo fato de que muitos ainda não possuem veículo próprio, além do péssimo estado de conservação das rodovias locais.

Portanto, entende-se que os desafios para a transformação da região por meio da mineração será um processo árduo. Muitos dos problemas enfrentados são ocorrências antigas agravadas pelas mudanças abruptas. É necessário que o poder público invista em projetos para concentrar a cadeia produtiva do lítio no país, investindo em infraestrutura nas cidades em evidência. O objetivo deste trabalho é oferecer uma possível solução por meio do desenvolvimento de uma aplicação para mitigar o problema de deslocamento na região em destaque.

1.2 Objetivos

O deslocamento entre as regiões é fundamental para trabalhadores da mineração e moradores das localidades. Uma característica do atual ciclo de extração do lítio no Vale do Jequitinhonha é o aumento no fluxo de movimentações entre as principais cidades, como Araçuaí, Itinga, Coronel Murta, Virgem da Lapa e Itaobim. Entretanto, um simples deslocamento pode se tornar difícil em algumas situações.

Primordialmente foi feita uma análise da oferta de veículos da região. Segundo dados do Transportes (2023), em 2023 o município de Araçuaí possuía uma frota de 15.667 veículos no total, entre os quais 4.436 deles são automóveis de passeio. Paralelamente, o último censo do IBGE (2022), apontou uma população de 34.297 pessoas. A partir desses números pode-se inferir que a região possui uma baixa oferta de veículos de passeio em comparação com o tamanho da população, sendo uma dedução ainda mais discrepante ao levar-se em consideração algumas possibilidades como a concentração de vários veículos à disposição de um mesmo proprietário. Tal fato se repete nas demais regiões supracitadas.

Outro fator que evidencia a necessidade de migrações pendulares está no fato de que alguns municípios possuem serviços que os outros não oferecem. Historicamente, o desenvolvimento neles ocorreu de forma distinta. Antes do ciclo de mineração do lítio, era comum que os moradores se deslocassem de cidades ou povoados em busca de tratamento médico ou para realizarem compras nos centros comerciais de outras cidades. Muitos viajavam para municípios próximos por meio de táxis, caronas ou por meio das poucas linhas de ônibus que operam em horários específicos. Atualmente, devido ao aumento da demanda de deslocamento, é comum que o acesso aos meios de transporte seja mais difícil.

Dessa forma, este trabalho tem a finalidade de apresentar um aplicativo que pode ser destinado a aumentar a segurança de uma prática que já ocorre, porém de forma informal, oferecer preços mais competitivos e uma maior oferta de horários para a realização

de traslados entre os municípios são objetivos do projeto.

1.3 Público-alvo e Benefícios

A carona é praticada na sociedade desde que os primeiros meios de transporte surgiram, por meio de cavalos e charretes. Normalmente, a carona é solicitada em ruas e estradas. No caso das estradas, utiliza-se um gesto universal: estender uma das mãos à frente do corpo com o polegar apontando na direção desejada. No entanto, um problema dessa prática é a falta de confiança entre passageiro e motorista.

Primordialmente, uma das finalidades do software é reduzir possíveis acontecimentos que prejudiquem a segurança dos envolvidos por meio da verificação do perfil de quem solicita a carona e de quem oferece a mesma, uma vez que a aplicação destina-se a todos que se deslocam constantemente entre as cidades envolvidas no complexo de mineração do lítio. Alia-se a isso a possibilidade de avaliar o perfil dos usuários conforme ocorre as viagens com uma nota e descrição.

Outro benefício do programa seria o valor final de uma corrida. Um proprietário de um carro que viaja constantemente entre os municípios poderia oferecer uma carona como forma de reduzir as despesas com combustíveis, ao passo que uma pessoa que busca a carona poderia conseguir o transporte com um valor mais competitivo do que outros meios de transporte comuns na região. Vale citar que o valor dos deslocamentos intermunicipais estão sofrendo constantes reajustes no Estado de Minas Gerais, o que motiva as pessoas a utilizarem meios alternativos.

Conforme demonstra Xavier (2024), fica evidente que o recente reajuste de 8%, realizado em setembro de 2024, no valor das passagens intermunicipais fez com que os passageiros ficassem descontentes com as empresas de viagens tradicionais. A notícia também relata que uma parte dos revoltados com a nova tributação não se importam de utilizar aplicativos de viagem, como o *Buser*, ou de aceitar caronas oferecidas em grupos de *Facebook* ou *Whatsapp*.

Caso o motorista tenha interesse e disponibilidade de espaço no seu veículo, ele poderá oferecer uma carona gratuita. A finalidade de ofertar tal recurso de forma não remunerada é preencher as vagas ociosas de seus carros em uma viagem que já ocorreria normalmente, ao passo que o motorista poderia ser beneficiado com uma companhia durante todo o trajeto.

2 ATIVIDADES DESENVOLVIDAS

Este capítulo apresentará a parte técnica de um produto na forma de software, inicialmente por meio de uma *Application Programming Interface (API)*, posteriormente adaptada com um Front-End, como possível solução da problemática em questão. Na sequência, será abordada sobre a organização do sistema, o escopo do projeto, as questões de armazenamento de dados e as tecnologias escolhidas para o desenvolvimento da aplicação. Ademais, o projeto possui apenas fins educacionais e exemplificativos até o presente momento.

2.1 Escopo do Projeto

O projeto tem como objetivo facilitar a busca e divulgação de caronas de forma ágil e organizada, melhorando uma prática já existente na região, mas que atualmente é desorganizada e de alcance limitado. Por meio de uma aplicação, será possível uma comunicação mais eficiente entre os usuários, além de oferecer maior segurança e valores competitivos para todos os envolvidos.

2.2 Organização Inicial

Primordialmente, foi fundamental criar uma organização no *GitHub* para gerenciar futuras atualizações do projeto entre os desenvolvedores, a qual pode ser acessada por meio do link: [https://github.com/projeto-integrador-tads/]

2.2.1 Requisitos Funcionais

Na sequência, foi realizado o levantamento dos requisitos técnicos, divididos em funcionais e não funcionais, com o objetivo de identificar os recursos mínimos necessários para o funcionamento inicial da aplicação, bem como visualizar possíveis necessidades futuras, visando garantir um melhor desempenho na versão final. Nesse contexto, as funcionalidades foram definidas na “Tabela 12” como requisitos funcionais do sistema e, na “Tabela 13”, como requisitos não funcionais - responsáveis por descrever as possíveis restrições do sistema até o presente momento, podendo ser implementadas futuramente.

Tabela 1: Tabela de Requisitos Funcionais do Sistema.

| Referência | Requisito | Descrição |
|------------|-----------|-----------|
|------------|-----------|-----------|

Continuação da tabela 1 na próxima página

Tabela 1: Tabela de Requisitos Funcionais do Sistema.

| | | |
|------|---------------------------|---|
| RF01 | Criação de conta | Um novo usuário poderá ser cadastrado informando um nome, e-mail e número de telefone. |
| RF02 | Cadastro de veículo | Cadastrar um veículo é o que possibilitará ao usuário oferecer uma carona. |
| RF03 | Reserva de carona | Os usuários podem solicitar ou oferecer uma carona. A segunda possibilidade somente será válida para aqueles com algum veículo devidamente registrado na plataforma. |
| RF04 | Mensagens | Os interessados na carona poderão trocar mensagens entre si com o objetivo de facilitar o encontro, definir horários e decidir detalhes da viagem. |
| RF05 | Filtrar caronas na região | Todos os usuários podem ver as caronas oferecidas na região especificada. Cada anúncio terá informações de destino, hora da viagem, lugares disponíveis e, se aplicável, o valor. |
| RF06 | Avaliações | A avaliação ocorrerá entre motorista e passageiro, sendo uma etapa fundamental para garantir a segurança de todos os usuários. Poderá ser atribuída uma mensagem de caráter opinativo para o público e uma nota que varia de 1 a 5. |
| RF07 | Serviços de e-mail | Envio de e-mails específicos de notificação após determinadas ações do usuário. Será possível disparar mensagens de boas vindas, criação de corrida bem sucedida, desativação e reativação de conta, entre outras. |

Tabela 2: Requisitos Não Funcionais do Sistema.

| Referência | Requisito | Descrição |
|------------|------------------------------------|--|
| RNF01 | Verificação de documentos pessoais | Validar se os documentos cadastrados na plataforma são válidos. |
| RNF02 | Consulta veicular | Consultar se os dados do veículo informados pelo usuário estão cadastrados na base de dados do departamento de trânsito. |

Continuação da tabela 2 na próxima página

Tabela 2: Requisitos Não Funcionais do Sistema.

| | | |
|-------|------------------------------------|---|
| RNF03 | Pesquisa de antecedentes criminais | A funcionalidade poderia aumentar a segurança do usuário. |
| RNF04 | Ranking de confiança | Os usuários com a melhor pontuação poderiam ter privilégios de divulgação ao oferecer ou solicitar uma carona. |
| RNF05 | Pagamentos | Serviços de pagamento com o objetivo de monetizar a aplicação, além da possibilidade de motoristas lucrarem de forma justa. |
| RNF06 | Mensagens automatizadas | O usuário pode utilizar um recurso de inteligência artificial para gerar mensagens rápidas no chat ao solicitar ou oferecer uma carona. |
| RNF07 | Canal de Suporte | Canal dedicado para oferecer suporte para possíveis problemas e esclarecer dúvidas frequentes. |

2.2.2 Critérios de Aceitação

Os Critérios de Aceitação são utilizados para garantir que o software atenda aos requisitos estabelecidos e ofereça uma experiência segura e funcional aos usuários. Esta seção define as condições e requisitos que devem ser cumpridos para que a versão final do sistema seja considerada aprovada e esteja pronta para uso.

1. Cadastro de Usuários

- Usuários devem se cadastrar com e-mail válido e senha forte.
- O e-mail deve ser único.
- A senha deve ter no mínimo 8 caracteres.
- Usuários devem fornecer informações pessoais básicas.

2. Segurança

- Encriptação de senha no banco de dados com um *hash*.
- As fotos de perfil e documentos ficarão salvos na *Amazon S3* seguindo as normas da LGPD.
- E-mails de alerta quando dados críticos sofrerem alteração (e-mail, senha).
- Alterar a senha possui uma quantidade máxima de tentativas por *token*.

3. Perfil de Usuário

- Usuários podem atualizar seu perfil a qualquer momento.

- Possibilidade de alterar nome, telefone, foto de perfil, etc.
- Endereço de e-mail e data de nascimento não podem ser alterados sem verificação adicional.
- Usuários devem verificar seu e-mail após o cadastro.
- Envio de um e-mail de confirmação com um link para ativar a conta.

4. Cadastro de Veículos

- Motoristas devem cadastrar seus veículos para oferecer caronas.
- Informar marca, modelo, ano, placa, cor e número de assentos disponíveis.
- Motoristas devem enviar documentos comprobatórios relativos ao veículo.

5. Publicação de Viagens

- Motoristas podem criar viagens detalhando rota, data, hora e pontos de embarque/desembarque.
- Especificar preço por passageiro, se aplicável.
- Informar restrições ou preferências dos envolvidos na viagem.
- As viagens devem ser criadas com antecedência mínima.
- Definir um tempo mínimo antes do horário de partida para criação de novas viagens.

6. Reserva de Caronas

- Passageiros podem reservar vagas nas viagens disponíveis.
- Confirmar a reserva mediante pagamento, se aplicável.
- Notificação para motorista e passageiro sobre a reserva confirmada.
- Passageiros podem cancelar suas reservas.
- Definir, quando aplicável, políticas de cancelamento e reembolso.

7. Avaliação e Feedback

- Usuários podem avaliar e deixar feedback sobre as viagens.
- Motoristas e passageiros podem se avaliar mutuamente.
- Avaliações devem ser visíveis nos perfis dos usuários.

8. Suporte

- Monitoramento e resolução de problemas.
- Canal de suporte ao usuário para resolução de problemas e disputas.

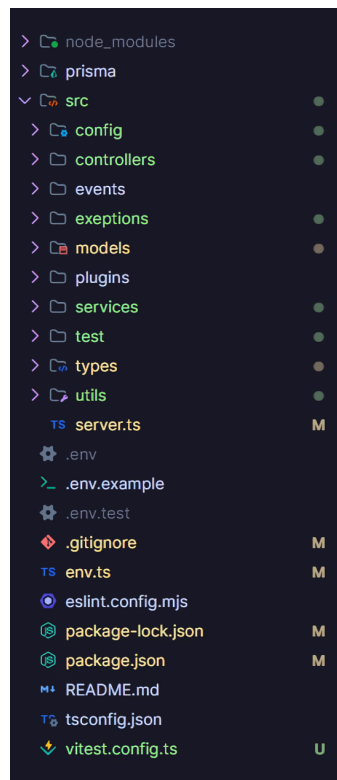
2.3 Back-end

Para alcançar o objetivo do trabalho, foi desenvolvida uma API que permitirá futuras adaptações para um aplicativo mobile. Essa API será responsável pelo gerenciamento do cadastro e autenticação dos usuários, permitindo que eles solicitem e divulguem caronas na região. Além disso, incluirá a funcionalidade de avaliação dos usuários com base no histórico de viagens, garantindo maior segurança.

2.3.1 Arquitetura do *Back-end*

Durante a produção de qualquer software é necessário garantir que o sistema seja robusto, eficiente e adaptável às necessidades do usuário. Com essa finalidade existe a arquitetura de sistema, responsável pela estrutura e organização dos componentes de um software, incluindo a maneira como esses elementos interagem entre si e com o ambiente externo. Com a finalidade de seguir os padrões de mercado, definiu-se um conjunto de regras para o *Back-end* aplicando os saberes adquiridos nas matérias de Programação Orientada a Objetos, Programação Web I e Banco de Dados I. A *API* em questão trabalha com o estilo *REST* para comunicação entre os componentes, além de separar as responsabilidades da aplicação em três partes principais, conhecidas como *Model*, *View* e *Controller*. A estrutura de pastas pode ser observada na Figura 1:

Figura 1 – Estrutura de Pastas do Projeto.



Fonte: Autoria Própria

2.3.2 Banco de Dados

A construção de um banco de dados foi necessário para fazer testes de requisição na API desenvolvida. Neste contexto, serão introduzidos o Diagrama Entidade-Relacionamento (DER) e o Modelo Entidade-Relacionamento (MER), para representar graficamente a organização das entidades e os vínculos entre os dados no sistema. Em seguida, serão descritos o esquema do banco de dados, suas tabelas e os relacionamentos estabelecidos entre elas. Essas representações visuais ajudam na compreensão da estrutura lógica e física do banco de dados, bem como facilita o processo de manutenção e expansão futura do software.

2.3.3 Diagrama Entidade-Relacionamento (MER)

A construção deste diagrama conceitual foi de suma importância para a modelagem de dados, representando o mini mundo em questão de um possível aplicativo de caronas. Transformar um recorte do mundo real, para o significado dos dados e como eles se relacionam colaboram na precisão das buscas de informações armazenadas no servidor. O “Anexo A” mostra o diagrama entidade-relacionamento deste projeto.

2.3.4 Modelo Entidade-Relacionamento (DER)

Após a modelagem do diagrama anterior, foi possível utilizar o BrModelo para realizar a conversão das tabelas necessárias no banco de dados. O DER facilita a compreensão do MER, tornando a estrutura do banco de dados mais intuitiva e visual, detalhando as chaves primárias, estrangeiras e as cardinalidades. O “Anexo B” apresenta as tabelas convertidas, suas chaves e cardinalidades.

2.3.5 Dicionário de Dados

O Dicionário de Dados é uma utilizada em projetos acadêmicos e de desenvolvimento de software, pois descreve detalhadamente as tabelas que compõem o banco de dados relacional, bem como seus atributos. Esta seção tem como objetivo proporcionar uma visão clara e organizada da estrutura do banco de dados, listando as entidades e suas respectivas características. No contexto deste trabalho, as principais tabelas incluem Usuários, Endereços, Motoristas, Veículos, Viagens, Reservas, Mensagens e Avaliações. As tabelas de 1 a 11 apresentam as descrições de cada tabela do banco, incluindo suas principais colunas e uma breve explicação dos atributos mais relevantes.

Tabela 3: Descrição da Entidade Usuários.

| Usuários |
|----------|
|----------|

Continuação da tabela 3 na próxima página

Tabela 3: Descrição da Entidade Usuários.

| NOME | TIPO DE DADOS | DESCRIÇÃO |
|-------------------------|---------------|---|
| (PK) id_usuario | VARCHAR | Chave primaria Identificador único da tabela “usuarios”. |
| nome | VARCHAR | Nome do usuário. |
| segundo_nome | VARCHAR | Sobrenome do usuário. |
| email | VARCHAR | E-mail único, utilizado para login. |
| senha | VARCHAR | Senha forte, com no mínimo 8 caracteres, incluindo letras maiúsculas, minúsculas, números e caracteres especiais. |
| telefone | VARCHAR | Número de telefone válido. |
| foto_perfil | VARCHAR | Caminho para a foto de perfil do usuário. |
| eh_motorista | BOOLEAN | Diferencia o usuário do motorista. |
| ativo | BOOLEAN | Salva a informação se o usuário é ativo. |
| classificacao _media | DOUBLE | Nota média do usuário. |
| data_criacao | DATE | Armazena a data da criação do perfil. |
| data_atualizacao | DATE | Ultima atualização do usuário. |

Tabela 4: Descrição da Entidade Token.

| Token | | |
|------------------|---------------|---|
| NOME | TIPO DE DADOS | DESCRIÇÃO |
| (PK) id_token | VARCHAR | Chave primária, identificador único da tabela “token”. |
| token | VARCHAR | Armazena o token único gerado para a solicitação de recuperação de senha. |
| expiracao_em | DATETIME | Armazena a data e hora em que o token expira. |
| (FK)_usuarios | VARCHAR | Chave estrangeira, referência à tabela “usuarios”. |

Tabela 5: Descrição da Entidade Mensagens.

| MENSAGENS | | |
|-----------------------|---------------|--|
| NOME | TIPO DE DADOS | DESCRIÇÃO |
| (PK) id_mensagem | VARCHAR | Chave primária, identificador único da mensagem. |
| conteudo | VARCHAR | Armazena o conteúdo das mensagens trocadas. |
| data_envio | DATETIME | Data do envio da mensagem. |
| (FK)_remetente | VARCHAR | Chave estrangeira. |
| (FK)_viagens | VARCHAR | Chave estrangeira. |
| (FK) _destinatario | VARCHAR | Chave estrangeira. |

Tabela 6: Descrição da Entidade Viagens.

| VIAGENS | | |
|-------------------------|---------------|--|
| NOME | TIPO DE DADOS | DESCRIÇÃO |
| (PK) id_viagem | VARCHAR | Chave primária, identificador da tabela viagem. |
| status | ENUM | Armazena os possíveis status da corrida, variando entre agendado, em andamento, concluído ou cancelado. |
| data_criacao | DATETIME | Data da criação da corrida. |
| preco | DECIMAL | Preço da corrida, quando aplicável a monetização da mesma. |
| hora_termino | DATETIME | Registra a hora que a corrida acaba. |
| preferencias | VARCHAR | Preferências da corrida definidas pelos participantes antes do seu início. |
| data_atualizacao | DATETIME | Data de atualização mais recente da viagem. |
| hora_partida | DATETIME | Define a hora de início de uma viagem. |
| lugares _disponiveis | INT | Armazena a quantidade de assentos disponíveis para os passageiros de acordo com o veículo cadastrado pelo motorista. |

Continuação da tabela 6 na próxima página

Tabela 6: Descrição da Entidade Viagens.

| | | |
|----------------------------|---------|--|
| (FK)_motorista | VARCHAR | Chave estrangeira, referência à tabela “motoristas”. |
| (FK)_enderecos _destino | VARCHAR | Chave estrangeira. |
| (FK)_enderecos _partida | VARCHAR | Chave estrangeira. |
| (FK)_veiculo | VARCHAR | Chave estrangeira, referência à tabela “veículos”. |

Tabela 7: Descrição da Entidade Reservas.

| RESERVAS | | |
|----------------------|---------------|---|
| NOME | TIPO DE DADOS | DESCRIÇÃO |
| (PK) id_reserva | VARCHAR | Chave primária, identificador único da tabela “reserva”. |
| data_criacao | DATETIME | Armazena a data da criação no momento em que a reserva é feita. |
| data_atualizacao | DATETIME | |
| status | ENUM | Armazena os possíveis status da reserva, variando entre agendado, em andamento, concluído ou cancelado. |
| status _pagamento | ENUM | Armazena os possíveis status de pagamento quando aplicável. |
| (FK)_passageiro | VARCHAR | |
| (FK)_viagens | VARCHAR | Chave estrangeira, referência à tabela “viagens”. |

Tabela 8: Descrição da Entidade Avaliações.

| AVALIAÇÕES | | |
|----------------------|---------------|---|
| NOME | TIPO DE DADOS | DESCRIÇÃO |
| (PK) id_avaliacao | VARCHAR | Chave primária, identificador único da avaliação. |
| avaliacao | INT | Numeral de 1 a 5 representando a nota da avaliação. |
| comentario | VARCHAR | Mensagem de avaliação do usuário. |

Continuação da tabela 8 na próxima página

Tabela 8: Descrição da Entidade Avaliações.

| | | |
|------------------|----------|---|
| data_criacao | DATETIME | Data de criação do registro. |
| data_atualizacao | DATETIME | Data da modificação mais recente. |
| (FK)_avaliador | VARCHAR | |
| (FK)_viagens | VARCHAR | Chave estrangeira, referência à tabela “viagens”. |
| (FK)_avaliado | VARCHAR | |

Tabela 9: Descrição da Entidade Endereços.

| ENDEREÇOS | | |
|------------------------|---------------|---|
| NOME | TIPO DE DADOS | DESCRIÇÃO |
| (PK) id_endereco | VARCHAR | Chave primária, identificador único do endereço. |
| cidade | VARCHAR | Nome da cidade. |
| longitude | DOUBLE | coordenada que especifica a posição geográfica da posição norte-sul de uma cidade específica. É de suma importância nas requisições. |
| latitude | VARCHAR | coordenada geográfica que especifica a posição leste-oeste de uma cidade específica. É de suma importância nas requisições. |
| endereco _formatado | VARCHAR | Nome formatado do endereço. Importante para a precisão das pesquisas, uma vez que algumas localizações possuem nomes populares, não registrados pelos serviços de geolocalização. |
| ativo | BOOLEAN | Salva a informação se o endereço é ativo. |
| data_criacao | DATETIME | Data de criação do registro. |
| data_atualizacao | DATETIME | Data da última atualização. |
| (FK)_usuarios | VARCHAR | Chave estrangeira, referência a tabela “usuarios”. |

Tabela 10: Descrição da Entidade Veículos.

| VEÍCULOS |
|----------|
|----------|

Continuação da tabela 10 na próxima página

Tabela 10: Descrição da Entidade Veículos.

| NOME | TIPO DE DADOS | DESCRIÇÃO |
|-----------------------|---------------|---|
| (PK) id_veiculo | VARCHAR | Chave primária, identificador único do veículo. |
| data_atualizacao | DATETIME | data da última alteração dos dados do veículo. |
| cor | VARCHAR | Cor do veículo. |
| marca | VARCHAR | Descreve a marca do veículo. |
| data_criacao | DATETIME | Data de cadastro do veículo. |
| ativo | BOOLEAN | Verifica se o veículo ainda está ativo na aplicação ou não. |
| ano | INT | Ano de fabricação do veículo. |
| placa | VARCHAR | Salva o número da placa do veículo após o seu registro. |
| capacidade | INT | Quantidade máxima de passageiros que o veículo registrado deve possuir. |
| modelo | VARCHAR | Descreve o modelo do veículo. |
| (FK) _proprietario | VARCHAR | |

2.3.6 Relacionamentos

1. Usuários - Reservas

- (1,N): Um usuário pode fazer várias reservas.
- (1,1): Cada reserva é feita por um único usuário.

2. Usuários - Avaliações

- (1,N): Um usuário pode fazer várias avaliações.
- (1,1): Cada avaliação é feita por um único usuário.

3. Usuários - Mensagens

- (1,N): Um usuário pode enviar e receber várias mensagens.
- (1,1): Cada mensagem tem um remetente e um destinatário únicos, podendo envolver diferentes usuários.

4. Usuários - Endereços

- (1,N): Um usuário pode ter vários endereços.

- (1,1): Cada endereço pertence a um único usuário.

5. Usuários - Token

- (1,N): Um usuário pode ter vários TokenJwtExpirado.
- (1,1): Cada TokenJwtExpirado pertence a um único usuário.

6. Motoristas - Veículos

- (1,N): Um motorista pode ter vários veículos.
- (1,1): Cada veículo está associado a um único motorista

7. Motoristas - Viagens

- (1,N): Um motorista pode criar várias viagens.
- (1,1): Cada viagem é conduzida por um único motorista.

8. Viagens - Reservas

- (1,N): Uma viagem pode ter várias reservas.
- (1,1): Cada reserva está associada a uma única viagem.

9. Viagens - Avaliações

- (1,N): Uma viagem pode ter várias avaliações.
- (1,1): Cada avaliação é feita para uma única viagem.

10. Endereço - Viagens

- (1,1): Uma viagem tem um endereço associado.
- (1,N): Cada endereço pode estar associado a várias viagens.

11. Mensagens - Viagens

- (1,N): Uma viagem pode ter várias mensagens associadas.
- (1,1): Cada mensagem pode estar se referir a uma única viagem.

12. Veículo - Viagens

- (1,N): Um veículo pode ser utilizado em várias viagens.
- (1,1): Cada viagem utiliza um único veículo.

2.4 Front-end

O desenvolvimento do front-end do aplicativo VemComigo foi orientado para proporcionar uma experiência intuitiva e acessível aos usuários. As escolhas durante esta etapa seguem os conhecimentos adquiridos nas aulas de Banco de Dados II, Análise e Projeto de Sistemas, Práticas Profissionais Integradoras II e Desenvolvimento Web II.

2.4.1 Arquitetura do Front-End

Primeiramente, foi realizada a prototipação das possíveis telas com o auxílio de um projeto criado no Figma, o qual pode ser acessado por meio do link:[<https://abrir.link/PMhoA>]. Criou-se pastas no projeto com a finalidade de melhor organizar as projeções do visual da aplicação. Os primeiros esboços visuais da aplicação encontram-se na página “Wireframes”. O desing de alta fidelidade foi projetado na sequência e pode ser encontrado na página “Desing” do projeto. Por fim, para integrar toda a API do back-end, desenvolvida anteriormente, ao front-end escolheu-se o React Native como framework do JavaScript, visando a criação de uma aplicação multiplataforma eficiente e de forma paralela para Android e IOS. Foi necessário utilizar o Expo como uma ferramenta auxiliar nos testes das telas desenvolvidas, garantindo que os componentes estivessem bem posicionados na tela.

2.4.2 Design e UX (Experiência do Usuário)

A interface foi projetada com foco na responsividade, usabilidade e experiência do usuário, permitindo que esses, mesmo que em diferentes dispositivos móveis, tenham uma utilização intuitiva e fluida. O VemComigo é uma aplicação voltada para a mobilidade e deslocamento de pessoas, dito isso, imagina-se que o usuário final busca simplicidade para escolher rapidamente um meio viável de transporte em meio a agitação do dia a dia. Nesse contexto, decidiu-se então que as telas deveriam ser minimalistas de forma que apenas informações necessárias ocupassem espaço na tela, evitando confusões em usuários mais desatentos. As principais escolhas do desing foram:

Paleta de Cores: A escolha das cores seguiu princípios de acessibilidade e contraste, sendo composta por:

- Cor principal: #0064D2 (Azul vibrante) – representa confiança, tecnologia e mobilidade.
- Cor de Fundo: #FFFFFF (Branco) – proporciona um ambiente limpo e moderno, facilitando a leitura e destacando os elementos interativos.
- Cores secundárias e de apoio: tons de cinza e azul escuro foram utilizados para equilibrar a interface e manter um visual neutro e sofisticado.

Tipografia: A fonte escolhida para o aplicativo foi a Poppins, uma tipografia moderna e elegante, amplamente utilizada em interfaces digitais por sua excelente legibilidade em diferentes tamanhos de tela. O uso de diferentes pesos (light, regular, semibold e bold) permitiu hierarquizar informações e destacar conteúdos importantes sem comprometer a estética da interface

Inputs, Botões e Ícones: Os campos de entrada (inputs) foram projetados para serem acessíveis e de fácil identificação. Foram aplicados bordas arredondadas, espaçamento adequado e feedback visual para indicar erros ou sucessos nas interações dos usuários. Os botões seguem um estilo minimalista, utilizando o azul primário (#0064D2) para submeter ações principais. Adotou-se também uma biblioteca de ícones chamada “Tabler”, disponível em [<https://tabler.io/admin-template>].

Logo: O logotipo do aplicativo foi desenvolvido para refletir o conceito de transporte de maneira abstrata. Utilizou-se uma abordagem minimalista, incorporando elementos gráficos que remetem a um formato de uma roda.

2.4.3 Componentização e Reutilização

2.4.4 Diagrama de Classes

O diagrama de classes do VemComigo foi gerado com base no schema do Prisma, refletindo a estrutura dos dados e os relacionamentos entre as entidades da aplicação. Como o projeto utiliza um back-end baseado em Node.js e Prisma ORM, sem a implementação direta de classes, o diagrama representa apenas os modelos de dados, sem a definição de métodos. Entre as principais entidades do sistema, destacam-se Usuário, Veículo, Corrida, Reserva, Endereço e Avaliação, além das tabelas auxiliares como Token e Mensagem. As relações foram modeladas para garantir um fluxo eficiente entre motoristas e passageiros, possibilitando funcionalidades como gerenciamento de corridas, envio de mensagens e sistema de avaliação. Essa estrutura permite a manipulação otimizada dos dados, garantindo escalabilidade e performance na aplicação. O “Anexo C” apresenta uma representação gráfica do diagrama de classes.

2.4.5 Diagrama de Caso de Uso

O diagrama de caso de uso do VemComigo representa as interações entre os principais atores do sistema: Passageiro, Motorista e Administrador. Ele ilustra os serviços disponíveis para cada perfil de usuário, destacando funcionalidades como criação de conta, solicitação e gerenciamento de corridas, troca de mensagens entre usuários e sistema de avaliações. O “Anexo D” apresenta uma representação gráfica do diagrama de classes.

Atores:

- Passageiro – Usuário que solicita caronas.
- Motorista – Usuário que oferece caronas.
- Administrador – Responsável por gerenciar o sistema.

Casos de Uso:

1. O passageiro pode solicitar caronas, gerenciar reservas e avaliar motoristas.
2. O motorista pode cadastrar veículos, oferecer corridas e interagir com passageiros.
3. O administrador, por sua vez, possui permissões para gerenciar usuários e monitorar avaliações.

2.5 Tecnologias e Ferramentas

O desenvolvimento de software exige uma variedade de recursos durante a produção. Escolher cuidadosamente as ferramentas a serem utilizadas no ambiente de desenvolvimento garante a qualidade, funcionalidade, eficácia, escalabilidade e eficiência do sistema. Nesse contexto, as tecnologias empregadas durante o desenvolvimento do trabalho serão descritas a seguir.

2.5.1 Ambiente de Trabalho

1. *Visual Studio Code* - editor de código-fonte gratuito que permite a integração com *Git*, facilitando *commits*, *pushes*, *pulls* e *merges*, além de possibilitar o uso do *intelliSense* para melhorar a produtividade no ambiente de trabalho.
2. *GitHub* - plataforma de hospedagem de código-fonte que permite o versionamento *Git*. Foi importante para que cada colaborador trabalhasse na implementação das mudanças nos repositórios da organização.
3. *Node.js* - Ferramenta de execução e interpretação da linguagem *JavaScript* que permite o seu uso no ambiente de desenvolvimento, sendo utilizado para executar os códigos criados ao lado do servidor com tal linguagem.

2.5.2 Linguagem de Programação

1. *JavaScript* - linguagem de programação escolhida devido a sua versatilidade, facilidade de uso, sintaxe limpa e grande oferta de *frameworks*.

2. *TypeScript* - é o superset do *JavaScript* que adiciona tipagem estática à linguagem, permitindo com que o desenvolvedor possa definir os tipos de dados das suas variáveis, funções e objetos com a finalidade de tornar o código mais seguro, previsível e escalável, além de facilitar futuras refatorações

2.5.3 Framework

1. *Fastify* - Uma das melhores opções entre os frameworks para *Node.js*, sendo rápido, flexível e com uma excelente experiência de desenvolvimento. Foi utilizado para construir aplicações web escaláveis e de alto desempenho, além de oferecer uma boa integração com o *TypeScript*.

2.5.4 Banco de Dados

1. *Prisma* - *Object-Relational Mapper (ORM)* escolhido para as interações com o banco de dados, sendo necessário para criar migrações, assim criar, ler, atualizar e deletar dados no banco de dados se tornou mais rápido e com menos código, reduzindo a possibilidade de erros.
2. *MySQL* - Sistema Gerenciador de Banco de Dados (SGBD) responsável por armazenar, organizar e gerenciar dados. É conhecido pela confiabilidade e ampla utilização nos mais variados ambientes de desenvolvimento.
3. *BrModelo* - foi uma ferramenta importante na modelagem do banco de dados, permitindo a elaboração de diagramas entidade-relacionamento (ER) e facilitando a sua visualização antes da implementação da versão final do banco.

2.5.5 Produtividade

1. *Trello* - Utilizado para o gerenciamento do projetos baseado em metodologia visual, por meio de um sistema de quadro de *Kanban*, dividindo uma tarefa em várias ações para que todos os integrantes do grupo participem do projeto de forma coesa.
2. *Notion* - Necessário para os desenvolvedores centralizarem as informações importantes, bem como anotações desenvolvidas ao longo do trabalho. Entre as suas vantagens destaca-se a facilidade de uso e a integração com outras ferramentas.

3 CONSIDERAÇÕES FINAIS

A mobilidade entre as cidades do Vale do Jequitinhonha, especialmente no contexto do atual ciclo de extração de lítio, apresenta desafios significativos devido à escassez de veículos e à oferta limitada de serviços de transporte. A análise da frota de veículos e a comparação com o tamanho da população evidenciam a necessidade de soluções práticas para facilitar o deslocamento entre os municípios.

Além disso, a desigualdade na oferta de serviços em cada cidade intensifica as migrações pendulares, o que sobrecarrega as opções de transporte disponíveis atualmente. Nesse cenário, a implementação de um aplicativo de caronas pode ser uma solução inovadora, capaz de formalizar uma prática já existente, porém de forma mais segura e eficiente. Tal proposta visa não apenas garantir maior acessibilidade ao transporte, mas também promover preços justos e aumentar a oferta de viagens, contribuindo para o desenvolvimento socioeconômico da região.

Durante o desenvolvimento do projeto foi possível criar vários *endpoints* importantes para garantir um funcionamento inicial adequado, o que facilitará possíveis integrações com o *frontend* futuramente.

Poderão ser implementadas novas funcionalidades no *Back-End*, como integração com serviços de pagamento e validação de documentos. O projeto será desenvolvido com foco em escalabilidade, segurança de dados e alta disponibilidade, assegurando uma experiência fluida e segura tanto para motoristas quanto para passageiros.

As funcionalidades de criação de conta, cadastro de motorista, autenticação de usuário e reserva de carona estão plenamente funcionais, além de seguirem o conceito do *Create, Read, Update, Delete (CRUD)*, essencial na criação de sistemas.

Referências

GERAIS, G. de M. *Governo de Minas lança em Nova Iorque projeto Vale do Lítio*. 2024. Agência Minas. Disponível em: <<https://www.agenciaminas.mg.gov.br/multimedia/galeria/governo-de-minas-lanca-em-nova-iorque-projeto-vale-do-litio>>. Acesso em: 17 set 2024. Citado na página 4.

IBGE. *População no último Censo*. 2022. IBGE. Disponível em: <<https://cidades.ibge.gov.br/brasil/mg/aracuai/panoram>>. Acesso em: 16 set 2024. Citado na página 6.

JAGANMOHAN, M. *Lithium global demand forecast 2025*. 2024. Statista. Disponível em: <<https://www.statista.com/statistics/452025/projected-total-demand-for-lithium-globally/>>. Acesso em: 17 set 2024. Citado na página 4.

LEÓN, L. P. *Níquel, lítio e satélites: conheça interesses de Musk no Brasil*. 2024. Agência Brasil. Disponível em: <<https://agenciabrasil.ebc.com.br/economia/noticia/2024-04/niquel-litio-e-satelites-conheca-interesses-de-musk-no-brasil>>. Acesso em: 25 set 2024. Citado na página 5.

MINERAÇÃO, R. *Grota do Cirilo é 4º maior complexo industrial de beneficiamento de lítio no mundo*. 2024. Revista Mineração. Disponível em: <<https://revistamineracao.com.br/2024/02/02/grota-do-cirilo-quarto-maior-complexo-beneficiamento-litio-mundo/>>. Acesso em: 17 set 2024. Citado na página 5.

SEDE-MG. *Minas Gerais começa o envio de lítio do Vale do Jequitinhonha para fora do Brasil*. 2023. SEDEMG. Disponível em: <[https://desenvolvimento.mg.gov.br/inicio/noticias/noticia/2206/minas-gerais-comeca-o-envio-de-litio-do-vale-do-jequitinhonha-para-fora-do-brasil#:~:text=Nesta%20quinta%2Dfeira%20\(27/7\)%2C%20o%20embarque%20das%20primeiras,Ara%C3%A7ua%C3%AD%2C%20esse%20mineral%20%C3%A9%20utilizado%20em%20aplica%C3%A7%C3%B5es](https://desenvolvimento.mg.gov.br/inicio/noticias/noticia/2206/minas-gerais-comeca-o-envio-de-litio-do-vale-do-jequitinhonha-para-fora-do-brasil#:~:text=Nesta%20quinta%2Dfeira%20(27/7)%2C%20o%20embarque%20das%20primeiras,Ara%C3%A7ua%C3%AD%2C%20esse%20mineral%20%C3%A9%20utilizado%20em%20aplica%C3%A7%C3%B5es)>. Acesso em: 17 set 2024. Citado na página 4.

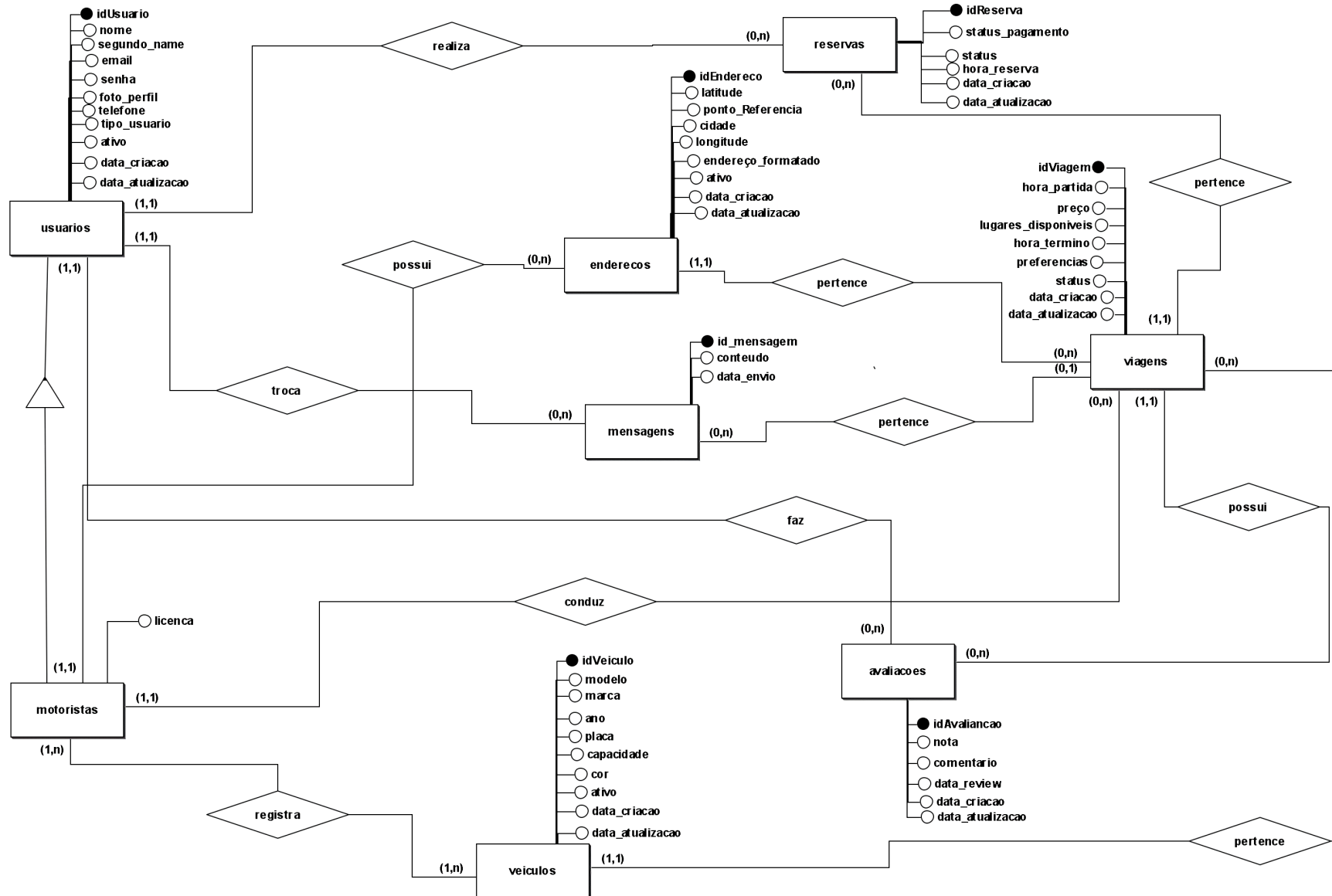
TRANSPORTES, M. dos. *Frota de Veículos 2023 - gov.br*. 2023. Ministério dos Transportes. Disponível em: <<https://www.gov.br/transportes/pt-br/assuntos/transito/conteudo-Senatran/frota-de-veiculos-2023>>. Acesso em: 18 set 2024. Citado na página 6.

VASCONCELOS, A. C. *Comunidades do Vale do Jequitinhonha denunciam violações de direitos com a mineração de lítio*. 2024. Brasil de Fato - Minas Gerais. Disponível em: <<https://bit.ly/3BjQIKz>>. Acesso em: 17 set 2024. Citado na página 5.

XAVIER, V. *Com reajuste de 8% nas passagens intermunicipais em Minas, alternativa pode ser ônibus por aplicativo ou caronas combinadas*. 2024. DiviNews. Disponível em: <<https://bit.ly/3XDw0Nd>>. Acesso em: 17 set 2024. Citado na página 7.

ANEXO A – MODELO ENTIDADE-RELACIONAMENTO (MER)

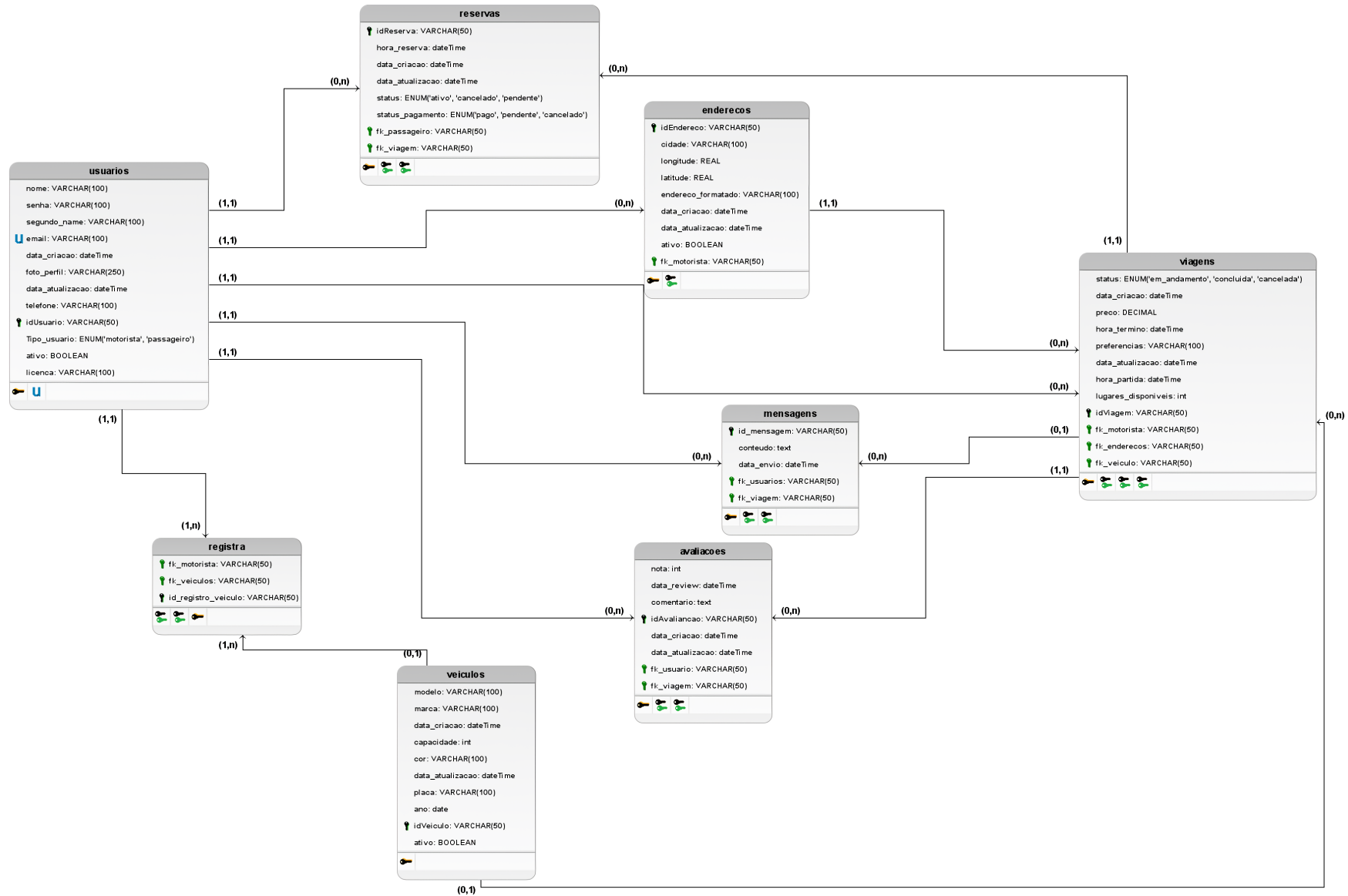
Figura 2 – Modelo Conceitual - Aplicativo de Caronas.



Fonte: Autoria Própria

ANEXO B – DIAGRAMA ENTIDADE-RELACIONAMENTO (DER)

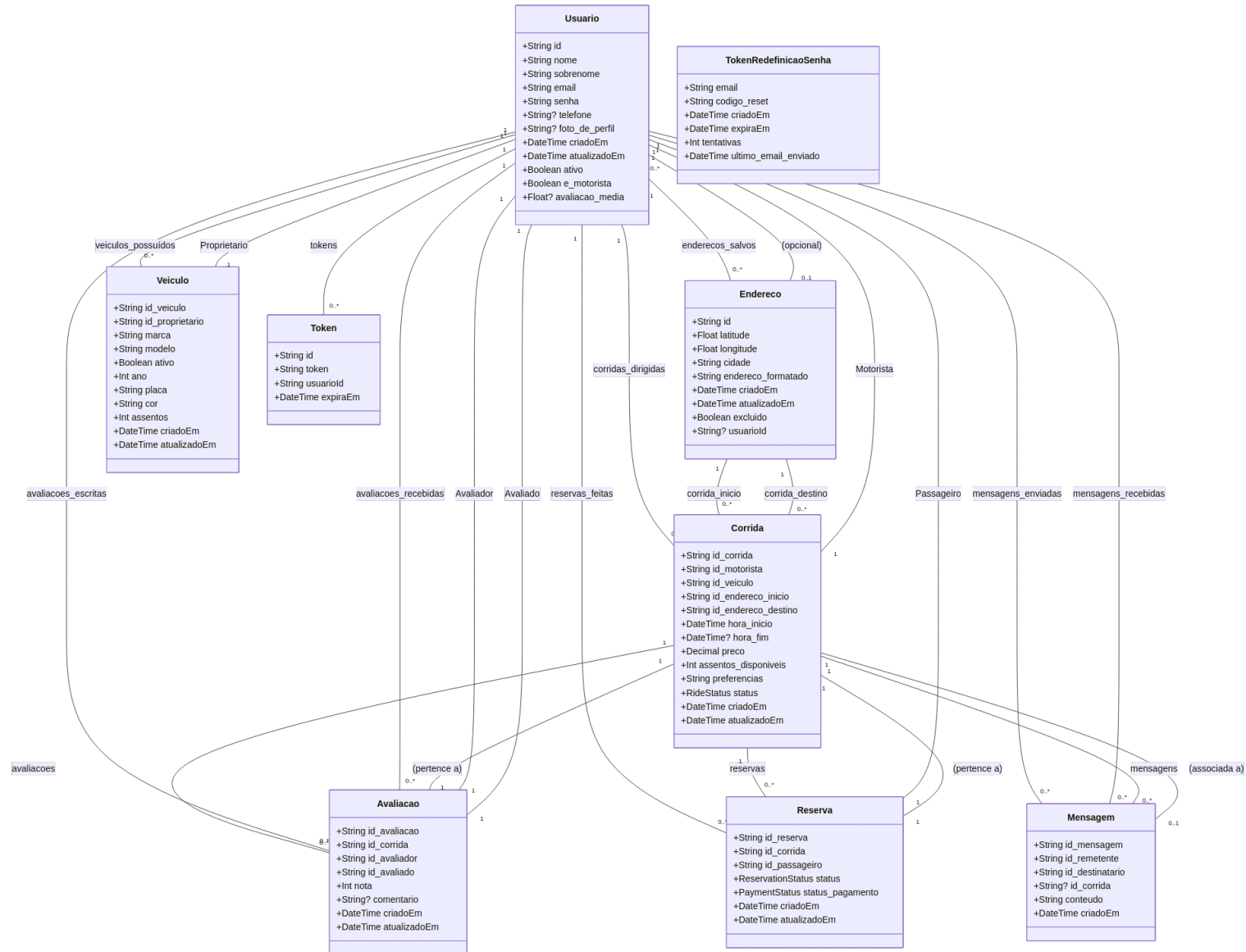
Figura 3 – Modelo Lógico - Aplicativo de Caronas.



Fonte: Autoria Própria

ANEXO C – DIAGRAMA DE CLASSES

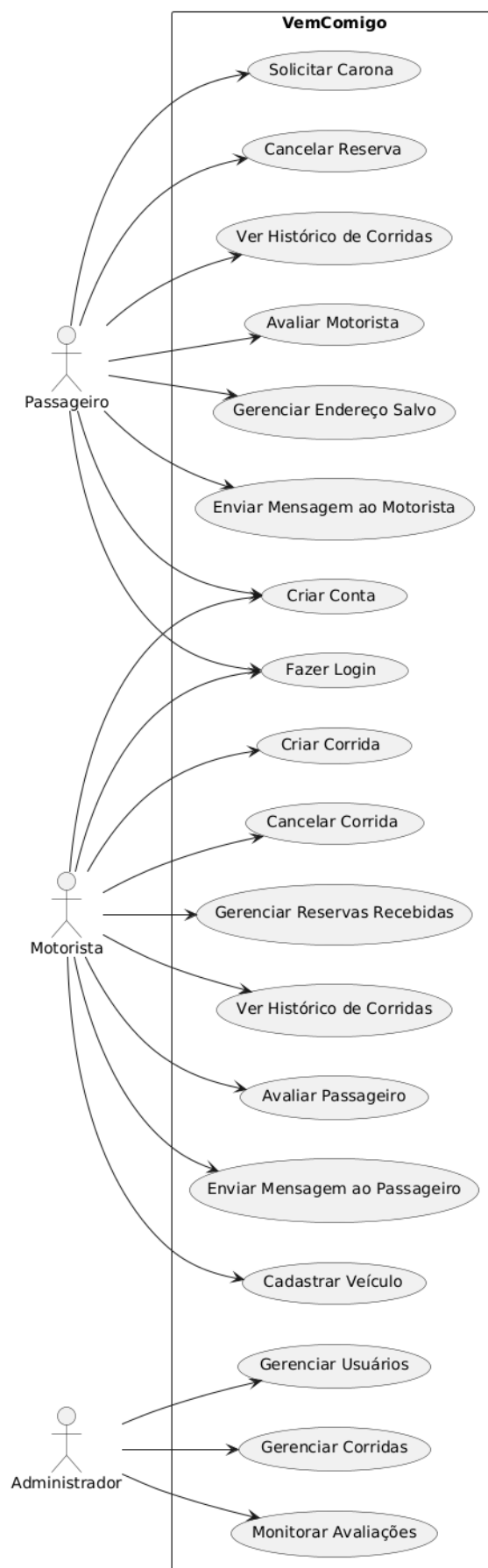
Figura 4 – Diagrama de Classes - VemComigo.



Fonte: Autoria Própria

ANEXO D – DIAGRAMA DE CASO DE USO

Figura 5 – Diagrama de Caso de Uso - VemComigo.



Fonte: Autoria Própria