

Requisitos Funcionais do MotoService

1. Painel do Atendente (Admin)

O painel do atendente é o centro de controle do sistema, oferecendo funcionalidades completas para gerenciamento de clientes, mecânicos, agendamentos e visão geral das operações.

1.1. Dashboard Geral

- Visão de Agendamentos por Dia/Semana: Exibe um resumo visual dos agendamentos programados, permitindo ao atendente ter uma visão rápida da demanda e disponibilidade.
- Lista de Serviços em Andamento: Apresenta uma lista atualizada de todos os serviços que estão sendo executados no momento, com informações essenciais como cliente, moto, mecânico atribuído e status.

1.2. Clientes

- Cadastro e Edição de Clientes: Permite ao atendente registrar novos clientes no sistema e atualizar suas informações cadastrais (nome, email, etc.).
- Histórico de Serviços de Cada Cliente: Possibilita a consulta de todos os serviços já realizados para um cliente específico, incluindo detalhes como tipo de serviço, data, mecânico e observações.

1.3. Mecânicos

- Cadastro e Edição de Mecânicos: Permite ao atendente registrar novos mecânicos e atualizar suas informações (nome, especialidade, etc.).
- Ver Agenda Individual de Cada Mecânico: Oferece uma visualização da agenda de trabalho de cada mecânico, mostrando os agendamentos atribuídos a eles.

1.4. Agendamentos

- Criar Novo Agendamento: Funcionalidade para registrar um novo agendamento, permitindo selecionar cliente, moto, tipo de serviço, data, hora e, opcionalmente, atribuir um mecânico.
- Ver e Editar Agendamentos: Permite ao atendente visualizar todos os agendamentos existentes e realizar modificações, como alterar data, hora, serviço ou mecânico atribuído.
- Atribuir Mecânicos: Funcionalidade específica para designar um mecânico a um agendamento que ainda não possui um.

2. Painel do Mecânico

O painel do mecânico é projetado para ser simples e focado na produtividade, exibindo apenas as informações relevantes para suas tarefas diárias.

2.1. Minhas Tarefas

- Lista de Serviços Atribuídos a Ele: Exibe uma lista clara de todos os serviços que foram atribuídos ao mecânico, organizados por status.

- Status: Cada serviço na lista deve indicar seu status atual: Pendente (aguardando início), Em Andamento (sendo executado) ou Concluído (finalizado).

2.2. Detalhes do Serviço

- Descrição do Problema: Exibe a descrição do problema relatado pelo cliente ou identificado no agendamento.
- Informações da Moto (Modelo, Ano): Apresenta os dados básicos da moto para a qual o serviço será realizado.
- Histórico de Reparos: Permite ao mecânico consultar o histórico de serviços anteriores daquela moto, auxiliando no diagnóstico e na execução do trabalho.

2.3. Finalizar Serviço

- Adicionar Observações: Permite ao mecânico registrar observações importantes sobre o serviço realizado, como peças trocadas, dificuldades encontradas ou recomendações futuras.
- Marcar como Concluído: Funcionalidade para alterar o status do serviço para Concluído após a finalização, liberando o mecânico para novas tarefas.

3. Painel do Cliente

O painel do cliente oferece uma visão personalizada de seus agendamentos e histórico de serviços, além de permitir o agendamento de novos serviços.

3.1. Dashboard do Cliente

- Próximos Agendamentos: Exibe os agendamentos futuros do cliente.
- Último Serviço: Mostra informações sobre o serviço mais recente realizado para o cliente.
- Total de Serviços: Apresenta o número total de serviços já realizados para o cliente.

3.2. Meus Agendamentos

- Ver Serviços Marcados: Permite ao cliente visualizar todos os seus agendamentos, tanto futuros quanto passados.
- Cancelar ou Reagendar: Oferece a opção de cancelar um agendamento existente ou solicitar um reagendamento, sujeito a confirmação da oficina.

3.3. Agendar Serviço

- Escolher Data, Tipo de Serviço e Moto: Permite ao cliente iniciar um novo agendamento, selecionando a data e hora desejadas, o tipo de serviço necessário e a moto a ser atendida.

3.4. Histórico

- Ver Serviços Anteriores: Exibe uma lista completa de todos os serviços que o cliente já realizou na oficina, com detalhes relevantes.

4. Tela de Login

- Entrada com Email e Senha: A tela de login permite que usuários (clientes ou funcionários) acessem o sistema utilizando seu email e senha cadastrados.

- Botão de Cadastrar Cliente: Um botão dedicado para que novos clientes possam iniciar o processo de cadastro diretamente da tela de login. O cadastro de funcionários é restrito ao atendente.

5. Tela de Cadastro

- Campos de Cadastro: Esta tela possui quatro campos essenciais para o registro de novos clientes: Nome, Email, Senha e Confirmar Senha.

Tecnologias Usadas

1. Frontend (Interface do Usuário)

HTML5 e CSS3

- Descrição: As linguagens fundamentais para a estruturação e estilização de conteúdo na web. HTML5 oferece novos elementos semânticos e APIs para multimídia e armazenamento local, enquanto CSS3 permite designs modernos e responsivos.
- Justificativa: Essenciais para a base de qualquer interface web.

Canva

- Justificativa: Usado para fazer a logo.

Chat GPT e Manus IA

- Justificativa: Foram essenciais para organizar, tirar dúvidas e solucionar dúvidas .

2. Banco de Dados

MySQL

- Descrição: Outro popular sistema de gerenciamento de banco de dados relacional de código aberto, conhecido por sua velocidade, confiabilidade e facilidade de uso.
- Justificativa: Uma alternativa viável ao PostgreSQL, especialmente se a equipe já tiver familiaridade com ele. É amplamente utilizado e possui um bom desempenho para a maioria das aplicações web.

3. Hospedagem

GitHub:

Utilizado para versionamento do código, colaboração e portfólio.

Permite salvar o projeto na nuvem e acompanhar o histórico de alterações.

4. Responsividade e Acessibilidade

Media Queries (CSS):

Utilizados para adaptar o layout para diferentes tamanhos de tela (celular, tablet, computador).

Google Fonts e Ícones (ex: Font Awesome, fontes do GOOGLE):

Para tornar o design mais moderno e com boa usabilidade.

DIAGRAMA DE CASO DE USO

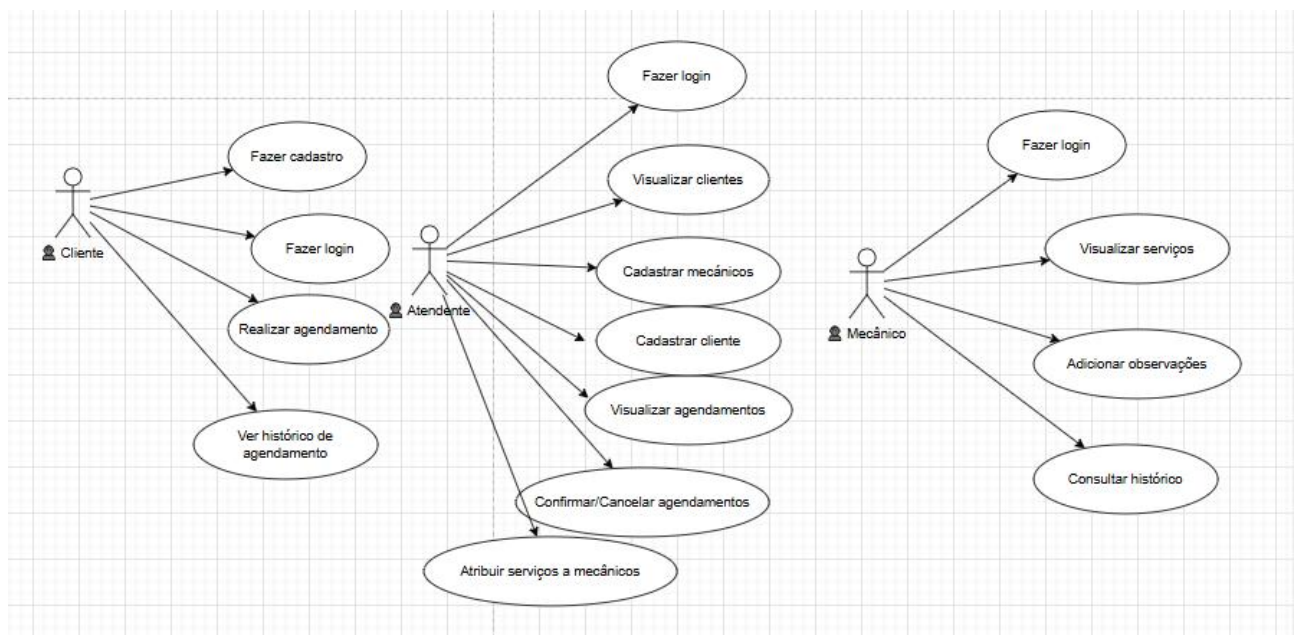
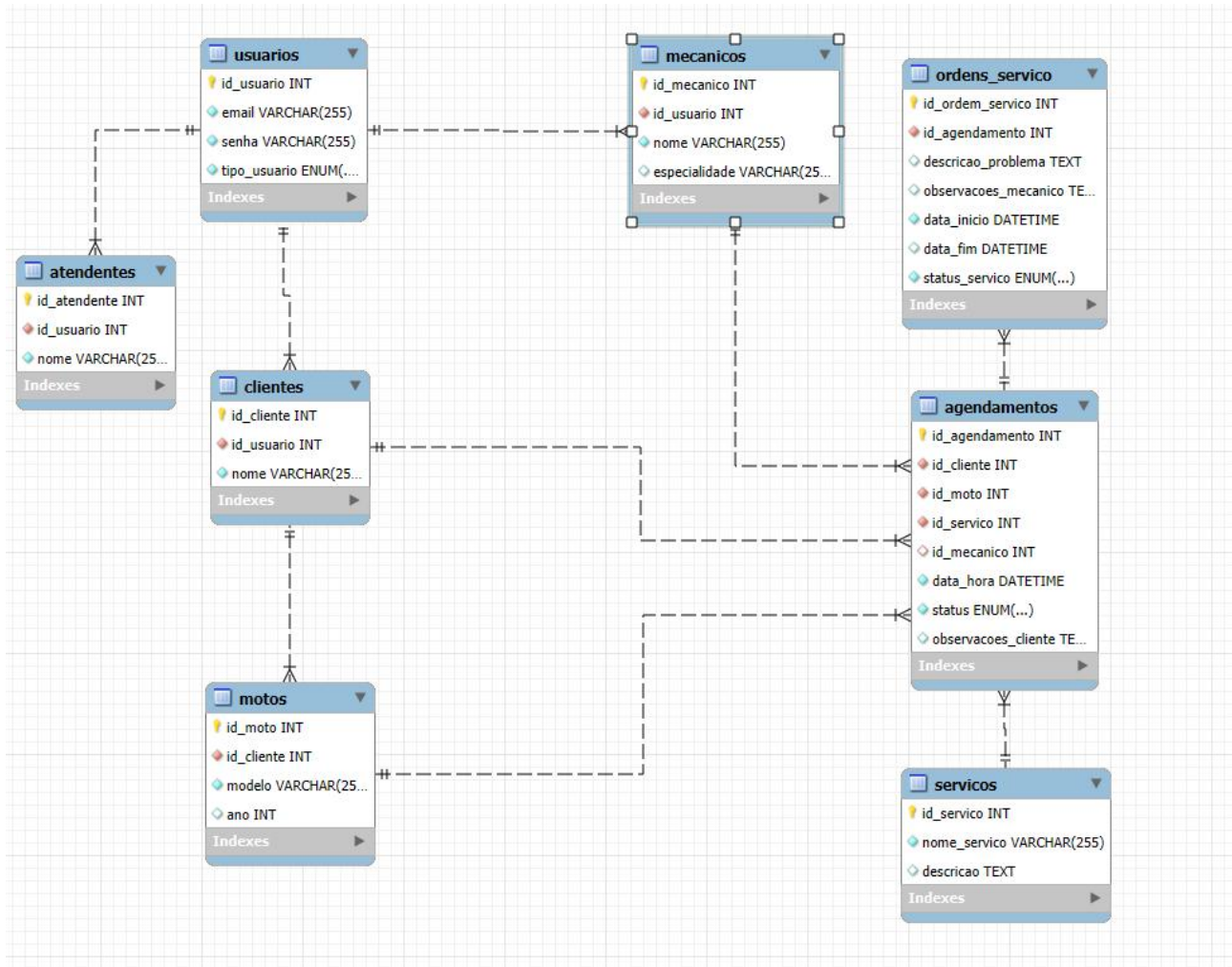


DIAGRAMA DE ENTIDADE E RELACIONAMENTO



CÓDIGO MYSQL

```
create database MotoService;
```

```
use MotoService;
```

```
update table motos;
```

```
CREATE TABLE usuarios (
```

```
    id_usuario INT PRIMARY KEY AUTO_INCREMENT,
```

```
    email VARCHAR(255) UNIQUE NOT NULL,
```

```
    senha VARCHAR(255) NOT NULL,
```

```
    tipo_usuario ENUM('cliente', 'atendente', 'mecanico') NOT NULL
```

```
);
```

```
CREATE TABLE clientes (
```

```
    id_cliente INT PRIMARY KEY AUTO_INCREMENT,
```

```
    id_usuario INT UNIQUE NOT NULL,
```

```
    nome VARCHAR(255) NOT NULL,
```

```
    FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario)
```

```
);
```

```
CREATE TABLE mecanicos (
```

```
    id_mecanico INT PRIMARY KEY AUTO_INCREMENT,
```

```
    id_usuario INT UNIQUE NOT NULL,
```

```
    nome VARCHAR(255) NOT NULL,
```

```
    especialidade VARCHAR(255),
```

```
    FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario)
```

```
);
```

```
CREATE TABLE atendentes (  
    id_atendente INT PRIMARY KEY AUTO_INCREMENT,  
    id_usuario INT UNIQUE NOT NULL,  
    nome VARCHAR(255) NOT NULL,  
    FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario)  
);
```

```
CREATE TABLE motos (  
    id_moto INT PRIMARY KEY AUTO_INCREMENT,  
    id_cliente INT NOT NULL,  
    modelo VARCHAR(255) NOT NULL,  
    ano INT,  
    FOREIGN KEY (id_cliente) REFERENCES clientes(id_cliente)  
);
```

```
CREATE TABLE servicos (  
    id_servico INT PRIMARY KEY AUTO_INCREMENT,  
    nome_servico VARCHAR(255) UNIQUE NOT NULL,  
    descricao TEXT  
);
```

```
CREATE TABLE agendamentos (  
    id_agendamento INT PRIMARY KEY AUTO_INCREMENT,  
    id_cliente INT NOT NULL,  
    id_moto INT NOT NULL,  
    id_servico INT NOT NULL,  
    id_mecanico INT,  
    data_hora DATETIME NOT NULL,  
    status ENUM('pendente', 'confirmado', 'cancelado', 'concluido') NOT NULL,
```

```
observacoes_cliente TEXT,  
FOREIGN KEY (id_cliente) REFERENCES clientes(id_cliente),  
FOREIGN KEY (id_moto) REFERENCES motos(id_moto),  
FOREIGN KEY (id_servico) REFERENCES servicos(id_servico),  
FOREIGN KEY (id_mecanico) REFERENCES mecanicos(id_mecanico)  
);
```

```
CREATE TABLE ordens_servico (  
    id_ordem_servico INT PRIMARY KEY AUTO_INCREMENT,  
    id_agendamento INT UNIQUE NOT NULL,  
    descricao_problema TEXT,  
    observacoes_mecanico TEXT,  
    data_inicio DATETIME NOT NULL,  
    data_fim DATETIME,  
    status_servico ENUM('pendente', 'em_andamento', 'concluido') NOT NULL,  
    FOREIGN KEY (id_agendamento) REFERENCES agendamentos(id_agendamento)  
);
```