Projeto Final

Aluna: Mariana Raquel Damasceno da Silva

Curso: Fullstack – sexta-feiras 19h



Documentação da Aplicação de Gerenciamento de Funcionários, Veículos e Equipamentos

Visão Geral

Esta aplicação web foi desenvolvida para realizar o **gerenciamento de funcionários, veículos, equipamentos e segurança** da empresa. O sistema também contempla suporte técnico, controle de acessos e permissões por tipo de usuário.

Tecnologias utilizadas:

• **Backend**: Flask (Python)

• **Frontend**: HTML5, CSS3, Bootstrap Icons

• Banco de Dados: MySQL

Requisitos do Sistema

•Python 3.x

•MySQL 5.x ou superior

•Flask 2.x

- •Bibliotecas do Python:
 - Flask
 - Flask-Session

- flask-wtf
- Flask-Flash
- Flask-url_for
- mysql-connector-python
- datetime
 - date
 - datetime
- Functools-wraps

Funcionalidades

1. Cadastro e Consulta

- Funcionários
- Veículos
- Equipamentos

2. Controle de Segurança

- Cadastro de usuários com permissões
- Primeiros acessos e redefinição de senha
- Acompanhamento de segurança digital

3. Movimentação

- Entrada e saída de veículos
- Entrega, devolução e envio de equipamentos
- Histórico de manutenções

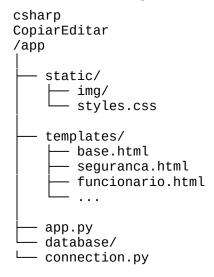
4. Chamados de Suporte

- Registro de problemas por usuários
- Classificação por setor e tipo

5. Permissões de Usuário

- Gerente (acesso completo)
- Analista (acesso limitado por área)
- Visualização

Estrutura do Projeto



Estrutura de Banco de Dados

Tabela Funcionarios

- id_matricula: ID do funcionário (chave primária).
- nome_completo: Nome completo
- data_nascimento: data
- rg: RG
- cpf: CPF
- data_admissao: Data de admissão
- cargo: Cargo
- email_corporativo: E-mail corporativo
- telefone_corporativo: Telefone corporativo
- data_demissao: Data de demissão

Tabela usuarios

- id_usuario: ID do usuário (chave primária).
- usuario: Nome de usuário.
- senha: Senha do usuário.

- primeiro_acesso: Indicador se é o primeiro acesso.
- Data de Criação: data
- Data de alteração: data
- Permissão: permissões de usuários

Tabela veiculos

- id_veiculo: ID do veículo (chave primária).
- placa: Placa do veículo.
- Marca: marca do veiculo
- modelo: Modelo do veículo
- Ano: ano
- chassi: chassi do veiculo
- renavam: renavam
- data_compra: data de compra
- fornecedor: fornecedor
- valor_compra: valor da compra
- documento fiscal compra: documento fiscal de compra
- comprador:comprador
- valor_venda: data de venda
- documento_fiscal_venda: documento fiscal de venda

Tabela controle_veiculos

- id_controle: ID do controle (chave primária).
- id_veiculo: ID do veículo (chave estrangeira).
- id_funcionario: ID do funcionário (chave estrangeira).
- data_entrada: Data e hora de entrada do veículo.
- data_saida: Data e hora de saída do veículo.
- Motivo: motivo da saída
- observação de entrada (avarias, necessidade manutenção etc)

Tabela equipamentos

- id_equipamento: ID do equipamento (chave primária).
- descrição do equipamento.
- Marca: marca do veiculo
- modelo: Modelo do veículo
- serie: serie do equipamento
- imei: imei no caso de aparelho celular
- data_compra: data de compra
- fornecedor: fornecedor
- valor_compra: valor da compra
- documento_fiscal_compra: documento fiscal de compra
- comprador:comprador
- valor_venda: data de venda
- documento_fiscal_venda: documento fiscal de venda

Tabela controle_equipamentos

- id_controle: ID do controle (chave primária).
- id_equipamento: ID do equipamento (chave estrangeira).
- id_funcionario: ID do funcionário (chave estrangeira).
- Setor: setor de localização do equipamento
- data registro: data da entraga ao setor ou/e funcionário responsavel
- motivo: motivo
- observação: observações

Rotas principais (Flask)

```
python
CopiarEditar
@app.route('/')
def index()
@app.route('/funcionario-pesquisa', methods=['GET'])
@app.route('/seguranca-pesquisa', methods=['GET', 'POST'])
@app.route('/seguranca-editar', methods=['POST'])
```

```
@app.route('/veiculo-pesquisa')
@app.route('/equipamento-pesquisa')
@app.route('/suporte-chamados')
```

Segurança

- Primeiro acesso obriga o usuário a criar uma nova senha.
- Data de criação e alteração de senha armazenadas no banco.
- Em caso de redefinição de senha o usuário reporta ao suporte.
- Suporte altera o campo de primeiro acesso de false (0) para true (1)

Como Executar

1. Crie o ambiente virtual:

```
bash
CopiarEditar
python -m venv venv
source venv/bin/activate # Windows: venv\Scripts\activate
```

2. Instale os requisitos:

```
bash
CopiarEditar
pip install flask mysql-connector-python
```

3. Execute a aplicação:

bash CopiarEditar flask run