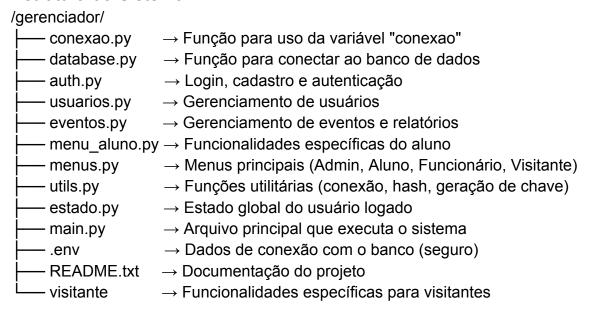
## DOCUMENTAÇÃO DO SISTEMA GERENCIADOR DE CREDENCIAIS, ACESSOS E EVENTOS

#### Descrição Geral

O sistema tem como objetivo gerenciar usuários, credenciais de acesso, eventos e registros de participação.

É uma aplicação de terminal que se conecta a um banco de dados MySQL e permite o controle de acessos e eventos por diferentes tipos de usuários.

#### Estrutura do Sistema



## Estrutura do Banco de Dados

| Tabela: Usuários |          |                                  |
|------------------|----------|----------------------------------|
| Campo            | Tipo     | Descrição                        |
| id_usuario       | INT (PK) | ID único do usuário              |
| id_tipo          | INT (FK) | Chave estrangeira para tipo      |
| nome_usuario     | VARCHAR  | Nome completo do usuário         |
| email_usuario    | VARCHAR  | Email (único)                    |
| senha_usuario    | VARCHAR  | Senha (hash SHA-256)             |
| id_responsavel   | INT      | ID de responsável à um visitante |
| prioridade_admin | INT      | Define a prioridade entre admins |

| Tabela: tipos_acesso |          |                         |
|----------------------|----------|-------------------------|
| Campo                | Tipo     | Descrição               |
| id_tipo              | INT (PK) | Identificador do tipo   |
| nome_tipo            | VARCHAR  | Nome do tipo de usuário |

| Tabela: credenciais |             |                              |
|---------------------|-------------|------------------------------|
| Campo               | Tipo        | Descrição                    |
| id_credencial       | INT (PK)    | Identificador da credencial  |
| id_usuario          | INT (FK)    | Usuário dono da credencial   |
| id_tipo             | INT (FK)    | Tipo de acesso da credencial |
| documento           | VARCHAR(11) | Documento (CPF)              |
| chave_acesso        | VARCHAR(8)  | Código de acesso gerado      |
| data_registro       | DATETIME    | Registra a data de criação   |

| Tabela: eventos |          |                               |
|-----------------|----------|-------------------------------|
| Campo           | Tipo     | Descrição                     |
| id_evento       | INT (PK) | Identificador do evento       |
| nome_evento     | VARCHAR  | Nome do evento                |
| data_evento     | DATE     | Data de realização            |
| local           | VARCHAR  | Local do evento               |
| descricao       | VARCHAR  | Detalhes sobre o evento       |
| ativo           | BOOLEAN  | Se o evento está ativo ou não |

| Tabela: acessos |          |                          |
|-----------------|----------|--------------------------|
| Campo           | Tipo     | Descrição                |
| id_acesso       | INT (PK) | ID do registro de acesso |
| id_credencial   | INT (FK) | Credencial usada         |
| id_tipo         | INT (FK) | Tipo de usuário          |
| data_hora       | DATETIME | Momento do acesso        |

| Tabela: participacoes |          |                           |
|-----------------------|----------|---------------------------|
| Campo                 | Tipo     | Descrição                 |
| id_participacao       | INT (PK) | ID da participação        |
| id_acesso             | INT (FK) | Relacionamento com acesso |
| id_evento             | INT (FK) | Evento relacionado        |
| data_inscricao        | DATETIME | Momento da inscrição      |

## Tipos de Usuários

- Administrador: Pode gerenciar usuários e eventos, visualizar relatórios.
- Aluno: Pode ver sua credencial, inscrever-se em eventos e registrar visitantes.
- Visitante: Pode ver sua credencial e tem acesso limitado aos eventos disponíveis para participação.

#### Módulos principais

- 1. Funcionalidades de Autenticação (auth.py)
- Login com verificação de senha hash (SHA256)
- Atualização automática de senhas não hash
- Cadastro de usuário e credencial

#### 2. Funcionalidades do Administrador

- Gerenciar usuários (listar, alterar, excluir, adicionar credencial)
- Gerenciar eventos (criar, listar, relatório de participações)
- Relatório geral de acessos

#### 3. Funcionalidades do Aluno

- Perfil: ver credencial, inscrever visitante, excluir visitante
- Eventos: listar eventos, inscrever-se, cancelar inscrição, ver eventos inscritos

#### 4. Funcionalidades Utilitárias

- Hash de senhas (SHA256)
- Verificação e geração de senha hash
- Geração de chaves de acesso aleatórias
- Conexão MySQL segura via dotenv
- Controle de sessão com usuario logado

#### Arquivo .env (Exemplo)

DB HOST=localhost

DB USER=seu usuario

DB PASSWORD=sua senha

DB NAME=gerenciador

#### Segurança

- Dados sensíveis protegidos com .env
- Verificação de senhas legadas e atualização automática

As senhas dos usuários são protegidas com hash SHA-256, utilizando a biblioteca hashlib. Durante o login:

- A senha digitada é convertida para hash e comparada com o valor armazenado no banco.
- Se o sistema detectar uma senha em texto puro (usuários antigos), ele a converte automaticamente para hash e atualiza o banco de dados.

# Execução do Programa python main.py

#### Melhorias Futuras

- Recuperação de senha
- Interface gráfica ou web
- Notificações e dashboards
- Versão mobile

| Desenvolvido por:                    |                    |  |
|--------------------------------------|--------------------|--|
| Nome                                 | Função             | Atividades principais  |
| Rodrigo de Carvalho Pereira          | Product Owner (PO) | Planejamento,<br>funcionalidades, conexão<br>com banco de dados,<br>organização. |
| Júlia Beatriz Nascimento da<br>Silva | Dev Front-End      | Interface, menus, interação com usuário.   |
| Chrystyan Jorge Santos<br>Dornelas   | Dev Back-End       | Lógica, revisão, teste de sistema.   |
| Vanuza Emília de Carvalho            | Dev Segurança      | Hash de senha.   |

<sup>-</sup> Projeto acadêmico — Análise e Desenvolvimento de Sistemas - 1º Semestre

<sup>-</sup> Ano: 2025