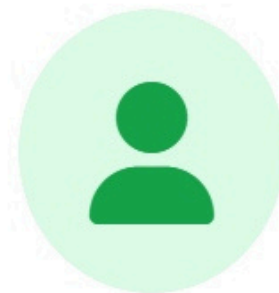


Reserva Certa

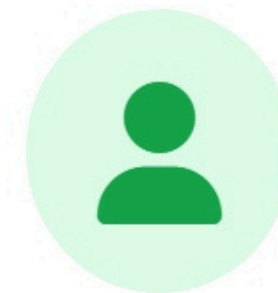
Sistema Inteligente de Gerenciamento de Espaços e Reservas Corporativas



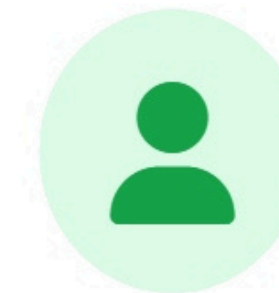
Iago Fereguetti



Gabriel Cunha



Lucas Henrique



Felipe Dias

Contexto e Motivação

O Problema

Conflitos constantes de agendamento em salas de reunião, falta de visibilidade sobre espaços livres e processos manuais descentralizados.

A Solução

Um sistema centralizado para organizar o acesso, garantir a reserva antecipada e otimizar o uso dos recursos físicos.

Requisitos e Modelagem



User Stories

Visualização de salas,
realização de reservas e
gestão administrativa.



Entidades

User: Autenticação.
Space: Salas.
Reservation: Data/hora.



Relações

1:N (Usuário -> Reservas)
N:1 (Reservas -> Sala)

Arquitetura MVC

- ✓ Rotas (Controller): Flask (``app.py``).
- ✓ Persistência (Repository): Camada isolada (``persistence/``).
- ✓ Banco de Dados: SQLite.

"A separação em camadas facilita a manutenção."

Implementação Tecnológica



Backend (Python + Flask)

Python 3.13. Padrão `connection_factory`.



Frontend (HTML5 + Jinja2)

Templates renderizados no servidor. CSS customizado.

Qualidade e Testes



Pytest

Testes unitários e de integração.



Bandit

Análise de segurança (vuln: debug=True).



Linter

Conformidade PEP8 via Flake8/Pylint.

Métricas de Código

6.49

NOTA PYLINT

100%

TESTES APROVADOS

5

CENÁRIOS TESTADOS

Melhorias: docstrings e prevenção de SQL Injection.

Demonstração Funcional

Fluxo completo:

1. Login

Autenticação segura.

2. Reserva

Verificação de conflitos.

3. Gestão

Cancelamento e agenda.

4. Relatórios

Métricas e logs.

Conclusão e Futuro

Principais Aprendizados

Repository Pattern e Análise Estática são essenciais.

Próximos Passos

- ✓ Expandir features.
- ✓ Implementar **ORM (SQLAlchemy)**.
- ✓ Autenticação via **JWT** e Mobile.



Dúvidas?

Obrigado pela atenção!

 github.com/reserva-certa/projeto